User's Manual

RZ/G3S Group

User's Manual: Hardware

Renesas Microprocessor
RZ Family / RZ/G Series

arm

Renesas Electronics
www.renesas.com

Rev.1.10 Nov, 2023

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Trademarks (continued)

For the “Cortex” notation, it is used as follows;
— Arm® Cortex®-A55
— Arm® Cortex®-M33
Note that after this page, they may be noted as Cortex-A55 and Cortex-M33 respectively.

Examples of trademark or registered trademark used in the RZ/G series, 2nd Generation User’s Manual: Hardware;
AMBA®: AMBA is a registered trademark of Arm Limited.
CoreLink™: CoreLink is a trademark of Arm Limited.
CoreSight™: CoreSight is a trademark of Arm Limited.
Mali™: Mali is a trademark of Arm Limited.
TrustZone ®: TrustZone is a registered trademark of Arm Limited.
HyperFlash™: HyperFlash is a trademark of Cypress Semiconductor Corporation.
MIPI®: MIPI is a registered trademark of MIPI Alliance, Inc.
Neon™: Neon is a trademark of Arm Limited.
OpenGL ES™: OpenGL ES is a trademark of Khronos Group, Inc.
DynamIQ™: DynamIQ is a trademark of Arm Limited.
OctaFlash™: OctaFlash is a trademark of Macronix International Co., Ltd.
eMMC™: eMMC is a trademark of MultiMediaCard Association.

Note that in each section of the Manual, trademark notation of ® and TM may be omitted.
All other trademarks and registered trademarks are the property of their respective owners.

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)
   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on
   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state
   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins
   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals
   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin
   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses
   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products
   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Table of Contents

1. Overview

1.1 Introduction

This LSI is a single-chip microprocessor that includes a single Arm® Cortex®-A55 core, which operates at speeds of up to 1.1GHz and two Cortex®-M33 250MHz cores. One Cortex®-M33 has FPU function. This LSI includes a 32-Kbyte L1 instruction cache, a 32-Kbyte L1 data cache, and a 256-Kbyte L3 cache. The following are the features of this LSI.

- RZ/G3S
  - 1.1 GHz Arm® Cortex®-A55 MPCore
  - Two 250 MHz Arm® Cortex®-M33 cores (One Cortex®-M33 has FPU function)
  - Memory controller for DDR4-1600 / LPDDR4-1600 with 16 bits
  - Octa-Flash/Octa-RAM interface
  - USB2.0 host / function interface
  - Gigabit Ethernet interface
  - SD/MMC host interface
  - CAN interface
  - Sound interface
  - On The Fly Decryption / Encryption
  - Tamper Detection
  - PCI Express Gen 2.0 interface (option)

NOTE

Arm and Cortex are registered trademark of Arm Limited. All other brands or product names are the property of their respective holders.

1.2 List of Specifications

1.2.1 CPU Core

Item	Description
System CPU Cortex-A55	• Arm Cortex-A55 Single Core 1.1 GHz • L1 I-cache: 32 Kbytes (Parity) / D-cache : 32 Kbytes (ECC) • L2 cache: Not included • L3 cache: 256 Kbytes (ECC) • NEON™ / FPU supported • Cryptographic Extension supported • Arm® v8.2-A architecture
System CPU Cortex-M33	• Arm Cortex-M33 Processor 250 MHz × 2 cores • Security Extension supported • Floating Extension supported*1 • Arm® v8-M architecture
Boot	• Bootable CPU: Cortex-A55, Cortex-M33*4 • Boot device: Boot Mode 0: Booting from eSD Boot Mode 1: Booting from eMMC Boot Mode 2: Booting from a serial flash memory (Single / Quad /Octal)*3 Boot Mode 3: Booting from SCIF download • Boot device voltage: 1.8 V*2, 3.3 V
Debug Interface	• Arm® CoreSight™ architecture • JTAG / SWD interface supported • ETF 16 Kbytes for program flow trace (each cluster) • JTAG Disable supported

Note 1. One core supports Floating Extension

Note 2. 1.8 V is supported in booting from eMMC and serial flash memory.

Note 3. A serial flash memory (Octal) is supported when Boot device voltage is 1.8 V.

Note 4. Cortex-M33 booting is supported from a serial flash memory and SCIF download.

1.2.2 CPU Peripheral

Item	Description
Clock Pulse Generator (CPG)	• Generates the clocks from external clock (EXCLK 24 MHz). Maximum Arm Cortex-A55 clock: 1.1 GHz Maximum Arm Cortex-M33 clock: 250 MHz Maximum DDR clock: 800 MHz (DDR4-1600 / LPDDR4-1600) Maximum AXI-bus clock: 200 MHz Maximum APB-bus clock: 100 MHz • SSC (Spread Spectrum Clock) supported
Direct Memory Access Controller (DMAC)	• 2 modules, 16 channels per module • Transfer request: On-chip peripheral request / auto request (software trigger) • A specific DMA transfer interval can be specified to adjust the bus occupancy. • LINK mode (DMA transfer under descriptor control) supported • Transfer information can be automatically reloaded
Interrupt Controller	• Arm® CoreLink™ Generic Interrupt Controller (GIC-600) for Arm Cortex-A55 • Nested Vectored Interrupt Controller (NVIC) for Arm Cortex-M33 • External Interrupt pins (NMI, IRQ7 to IRQ0, TINT31-0) • On-chip peripheral Interrupts: Priority level set for each module
Message Handling Unit (MHU)	• Message handling function between Arm Cortex-A55 and Arm Cortex-M33 • Assert interrupt to inform message and response from/to Arm Cortex-A55, Cortex-M33
General-purpose I/O (GPIO)	• General-purpose I/O ports
Thermal Sensor Unit (TSU)	• 1 channel

1.2.3 Internal Memory

Item	Description
On-chip RAM	• RAM of 1 Mbytes (ECC)

1.2.4 External Memory Interface

Item	Description
External Bus Controller for DDR4 / LPDDR4 SDRAM (DDR)	• Support DDR4-1600 / LPDDR4-1600 • Bus Width: 16-bit • In line ECC supported (Support error detection interrupt) • Memory Size: Up to 4 Gbytes (DDR4), 1 Gbytes (LPDDR4) • Auto Refresh / Self Refresh supported • On-The-Fly Decryption / Encryption supported
eXpanded Serial Peripheral Interface (xSPI)	• 1 channel • Up to 2 serial flash memories can be connected • Connectable with 2 Quad-SPI flash memories • Connectable with 2 Octal-SPI flash memories • Connectable with 2 Octal-RAMs • External address space read mode (built-in read cache) • SPI operation mode • Maximum Clock Frequency: 66 MHz (Single-SPI / Quad-SPI, SDR, 1.8 V / 3.3 V), 133 MHz (Octal-SPI / OctaFlash / OctaRAM, DDR, 1.8 V) • On-The-Fly Decryption / Encryption supported
Octa Memory Controller	• Macronix Serial Multi I/O (MXSMIO®) Octa Peripheral Interface (OPI) for high-end consumer applications is supported. • One each of an OctaFlash device and an OctaRAM device compliant with the OPI specifications are connectable. • A chip select signal is assigned to each memory device (OM_CS0#: OctaFlash; OM_CS1#: OctaRAM). • Supported device interfaces SPI: Serial peripheral interface (OctaFlash, SPI mode) SOPI: Single Octa I/O (8 bits) (OctaFlash, single data rate) DOPI: Double Octa I/O (8 bits) (OctaFlash and OctaRAM, double data rate) • On-The-Fly Decryption / Encryption supported
SD Card Host Interface / Multimedia Card Interface (SD/MMC)	• 3 channels • Channel 0 supports SDHI / e-MMC (boot supported) • Channel 1 and 2 support SDHI (Channel 1: Dedicated pin, Channel 2: Multiplexed pin, 3.3 V only) • SD memory I/O card interface (1-bit/4-bit SD bus) • SD, SDHC and SDXC SD memory card access supported • Compliant with SD 3.0 • Default, high-speed, UHS-I/SDR50, SDR104 transfer modes supported • Error check function: CRC7 (Command/response), CRC16 (Data) • Card detection function, write protect supported • MMC interface (1-bit/4-bit/8-bit MMC bus) • e-MMC device access supported • Compliant with eMMC 4.51 • High-speed, HS200 transfer modes supported (SD clock (SD_CLK) frequency: Up to 125 MHz)

1.2.5 Sound Interface

Item	Description
Serial Sound Interface (SSI)	• 4 channels bidirectional serial transfer • 2 external clock sources available • Full Duplex communication • Support of I2S / Monaural / TDM audio formats • Support of master and slave functions • Generation of programmable word clock and bit clock • Multi-channel formats • Support of 8, 16, 18, 20, 22, 24, and 32-bit data formats • Support of 32-stage FIFO for transmission and reception • Support of LR-clock continue function in which the LR-clock signal is not stopped
Pulse Density Modulated (PDM) Interface	• 3 channels • Capable of filtering 1-bit digital input data PDM_DATn (n = 0,..,2) and converting them into 20-bit or 16-bit digital data. • Support of stereo microphone (L/R sampling by rising/falling clock edge). • Support of sound activity detector. • Support of programmable filters: 4th order sinc filter, high-pass filter (for suppression of DC bias), correction filter (for sinc passband distortion), half-band decimation filter (for aliasing distortion). • Internal buffer: Capable of storing voice data during low power mode
SPDIF	• Supports the IEC 60958 standard (stereo and consumer use modes only). • Supports sampling frequencies of 32 kHz, 44.1 kHz, and 48 kHz. • Supports audio word sizes of 16 to 24 bits per sample. • Biphase mark encoding. • Double buffered data. • Parity encoded serial data. • Simultaneous transmit and receive • Receiver autodetects IEC 61937 compressed mode data
Sampling Rate Converter (SRC)	• 1 channel • Data format: 16-bit (stereo / monaural) • Sampling Rate Input: Selectable from 8 kHz, 11.025 kHz, 12 kHz, 16 kHz, 22.05 kHz, 24 kHz, 32 kHz, 44.1 kHz, 48 kHz Output: Selectable from 8 kHz*, 16 kHz*, 32 kHz, 44.1 kHz, 48 kHz (*: can select in 44.1 kHz input mode) • SNR: More than or equal to 80 dB

1.2.6 Storage and Network

Item	Description
USB2.0 Host / Function (USB)	• 2 channels (ch0: Host-Function ch1: Host only) • Compliance with USB2.0 • Supports On-The-Go (OTG) Function • Supports Battery Charging Function • Internal dedicated DMA
Gigabit Ethernet Interface (GbE)	• 2 channels • Supports transfer at 1000 Mbps and 100 Mbps, 10 Mbps • Supports filtering of Ethernet frames • Supports interface conforming to IEEE802.3 PHY RGMII (Reduced Gigabit Media Independent Interface) • Supports interface conforming to IEEE802.3 PHY MII (Media Independent Interface)
Controller Area Network Interface (CAN)	• 2 channels • CAN-FD ISO 11898-1 (CD2015) compliant • Up to 1 Mbps for arbitration phase and up to 8 Mbps for data phase • Message buffer – Up to 64 × 2-channel receive message buffer: Shared among all channels – 16 transmit message buffers per channel
PCI Express Gen2 (option)	• PCI Express Base Specification 4.0 compliant • PCI Express Gen1(2.5[GT/s])/Gen2(5.0[GT/s]) • Root Complex, Type1 Configuration Register • Lane implementation ×1 • Support Polarity inversion • Maximum data payload of 256 bytes, Maximum read request size 512 bytes • Not support for Virtual channels (support VC0 only) • Number of outstanding 1-8 • Dynamic control of speed/width up/down configuration • Not support for Clock Power Management (not support P1.CPM, P2.CPM) • Power Management (ASPM L1-Substate Support (Support Power Down Sequence only)) • Error handling/logging (AER Support) • Replay FIFO with ECC • Internal Memory without Parity • Number of Support Functions 1

1.2.7 Timer

Item	Description
Multi-function Timer Pulse Unit 3 (MTU3a)	• 9 channels (16 bits × 8 channels, 32 bits × 1 channel) • Module clock frequency: 100 MHz • Maximum 28 lines of pulse inputs/outputs and 3 lines of pulse inputs • 14 types of count clocks selectable • Input capture function • 39 outputs compare and input capture registers • Counter clear operation (Simultaneous counter clearing by Compare match or Input capture is available) • Simultaneous writing to multiple timer counters (TCNT) • Synchronous input/output of each register due to synchronous operation of the counter • Buffered operation • Cascade-connected operation • 43 types of interrupt sources • Automatic transfer of register data • Pulse output modes Toggle, PWM, complementary PWM, and reset-synchronized PWM modes • Synchronization of multiple counters • Phase counting mode – 16-bit mode (channel 1 and 2) – 32-bit mode (channel 1 and 2) • Counter function of dead time compensation • Digital filter functions for the input capture and external count clock pin
Port Output Enable 3 (POE3)	• Control of the high-impedance state of the MTU3a waveform output pins • Activation with four input pins • Activation on detection of short-circuited outputs (detection of simultaneous PWM output to the active level) • Activation by register write • Additional programming of output control target pins is possible.
General PWM Timer (GPT)	• 32 bits × 8 channels • Counting up or down (sawtooth wave), counting up and down (triangular wave) selectable for all channels • Independent selectable for each channel • 2 input/output pins per channel • 2 output compare / input capture registers per channel • For the 2 output compare / input capture registers of each channel, 4 registers are provided as buffer registers and are capable of operating as comparison registers when buffering is not in use • In output compare operation, buffer switching can be at peaks or troughs, enabling the generation of laterally asymmetrically PWM waveforms • Registers for setting up frame intervals on each channel (with capability for generating interrupts on overflow or underflow) • Generation of dead times in PWM operation • Synchronous start / stop / clear of counters on arbitrary channels • Starting, stopping, and clearing up/down counters in response to a maximum of eight events • Starting, stopping, and clearing up/down counters in response to input level comparison • Starting, stopping, and clearing up/down counters in response to a maximum of four external triggers • Output pin invalidation functions due to dead time error or detection of short circuit between output pins • Digital filter functions for the input capture and external trigger pins
Port Output Enable for GPT (POEG)	• Output prohibition control of the GPT waveform output pin • Activation with up to four input pins • Activation by dead time error detection or output short detection • Activation by register write

Item	Description
Watchdog Timer (WDT)	• 3 channels • A counter overflow can reset the LSI • CPU parity error can reset the LSI
General Timer (GTM)	• 32 bits × 8 channels • Two operating modes – Interval timer mode – Free-running comparison mode
RTC	• Clock sources: 32 kHz clock (RTXIN) • Count mode: Calendar count mode / binary count mode • Interrupt sources: Alarm interrupt, periodic interrupt and carry interrupt • Time capture function

1.2.8 Peripheral Module

Item	Description
I2C Bus Interface (I2C)	• 4 channels (ch0,1 = Dedicated pin, ch2,3 = Multiplexed pin) • Master mode and slave mode supported • Support for 7-bit and 10-bit slave address formats • Support for multi-master operation • Timeout detection
I3C Bus Interface (I3C)	• 1 channel • Master (Main Master/Secondary Master) mode and Slave mode selectable • SDR (I3C Single Data Rate) Mode – Private Message – Broadcast Message (Common Command Code) – Direct Message (Common Command Code) • Legacy I2C Message – Fast-mode (Fm): Up to 400 kbit/s – Fast-mode Plus (Fm+): Up to 1 Mbit/s • Slave Interrupt Request • Master Ship Request (Secondary Master only) • Support for 7-bit slave address formats • Synchronous Timing Control – Sync Mode: Synchronous Basic Mode • Asynchronous Timing Control – Async Mode 0: Asynchronous Basic Mode – Async Mode 1: Asynchronous Advanced Mode • Error Detection
Serial Communication Interface with FIFO (SCIFA)	• 6 channels • Clock synchronous mode or asynchronous mode selectable • Simultaneous transmission and reception (full-duplex communication) supported • Dedicated baud rate generator • Separate 16-byte FIFO registers for transmission and reception • Modem control function (channel 0, 1 and 2 in asynchronous mode)
Serial Communication Interface (SCIg)	• 2 channels • Clock synchronous mode, asynchronous mode, or smart card interface mode is selectable • Simultaneous transmission and reception (full-duplex communication) supported • Dedicated baud rate generator • LSB first / MSB first selectable • Modem control function • Encoding and decoding of IrDA communications waveforms in accord with version 1.0 of the IrDA standard (on channel 0)
Renesas Serial Peripheral Interface (RSPI)	• 5 channels • SPI operation • Master mode and slave mode supported • Programmable bit length, clock polarity, clock phase can be selected • Consecutive transfers • LSB first / MSB first selectable

1.2.9 Security

Item	Description
Renesas Security IP (RSIP-E01B) [option]	• Security algorism – Common key encryption: AES (compliant with NIST FIPS PUB 197) – Non-common key encryption: RSA, ECC • Other features – TRNG (true-random number generator) – Hash value generation: SHA-1, SHA-224, SHA-256 – Support of Unique ID
One Time Programmable memory (OTP)	• A nonvolatile memory that can be written only once • Security setting, authentication setting are possible • Support one time read function (128 bytes)
Battery Backup Function	• Realtime clock • Backup register • Tamper detection

1.2.10 Analog

Item	Description
A/D Converter (ADC)	• 8 channels • Resolution: 12-bit • Input Range: 0 V to 1.8 V • Conversion Time: 1.0 µs • Operation Mode: Select mode / scan mode • Conversion Mode: Single mode / repeat mode • Condition for A/D conversion start – Software trigger – Asynchronous trigger: External trigger supported – Synchronous trigger: MTU and PWM timer

1.2.11 Others

Item	Description
Boundary Scan	• Boundary scan based on IEEE 1149.1 via JTAG interface is supported. Note that some module pins are not available on this boundary scan.

1.2.12 Power Supply Voltage

Item	Description
Power supply voltage	• VBATT_VDD: 1.50 to 1.95 V • VDD, PLL16_AVDD, PLL23_AVDD, PLL4_AVDD: 0.905 to 0.99 V • PVDD33: 3.0 to 3.6 V • PVDD18, ADC_AVDD18, OTP_AVDD18: 1.65 to 1.95 V • JTAG_PVDD: 1.65 to 1.95 V • XSPI_PVDD: 1.65 to 1.95 V / 3.0 to 3.6 V • I3C_PVDD: 1.1 to 1.3 V / 1.65 to 1.95 V • VDD_ISO, PCIE_VDD09: 0.905 to 0.99 V • SDn_PVDD (n = 0, 1): 1.65 to 1.95 V / 3.0 to 3.6 V • PVDD182533_n (n = 0, 1): 1.65 to 1.95 V / 2.3 to 2.7 V / 3.0 to 3.6 V • USB_VDD33: 3.0 to 3.6 V • USB_AVDD18, USB_VDD18, PCIE_VDD18: 1.65 to 1.95 V • DDR_VAA: 1.65 to 1.95 V • DDR_VDDQ: 1.06 to 1.17 V(LPDDR4) / 1.14 to 1.26 V (DDR4)

1.2.13 Temperature Range

Item	Description
Temperature range	• T_a: −40°C to +85°C*1 • T_j: −40°C to +125°C

Note 1. If wider temp is required than this range, use case has to be investigated.

1.2.14 Quality Level

Item	Description
Quality level	• Industrial usage, etc.

1.2.15 Package

Item	Description
Package	• PBGA, 13-mm square, 0.5mm pitch (w/o PCIe) • PBGA, 14-mm square, 0.5mm pitch (w/ PCIe)

1.3 Product Lineup

Table 1.1 Product Lineup

Group	Package	Part Number	CPU	Security	PCIe
RZ/G3S	14 mm BGA	R9A08G045S37GBG	1 × Cortex-A55, 2 × Cortex-M33	Available	Available
		R9A08G045S17GBG	1 × Cortex-A55, 1 × Cortex-M33		
		R9A08G045S33GBG	1 × Cortex-A55, 2 × Cortex-M33	Not supported	
		R9A08G045S13GBG	1 × Cortex-A55, 1 × Cortex-M33		
	13 mm BGA	R9A08G045S35GBG	1 × Cortex-A55, 2 × Cortex-M33	Available	Not supported
		R9A08G045S15GBG	1 × Cortex-A55, 1 × Cortex-M33		
		R9A08G045S31GBG	1 × Cortex-A55, 2 × Cortex-M33	Not supported	
		R9A08G045S11GBG	1 × Cortex-A55, 1 × Cortex-M33		

1.4 Pin

This section describes the pins of this LSI.

1.4.1 Pin Assignment

Refer to another excel file for the “pin function list” about pin assignment of this LSI.

1.4.2 External Pins

Refer to another excel file for the “pin function list” about information of external pins of this LSI.

2. System CPU Cortex-A55

The Cortex-A55 system CPU is a core block equipped with single Cortex-A55 core. For details on the functions including interrupts of the Cortex-A55, see the Arm® Cortex®-A55 Core Technical Reference Manual and related documents in **Section 2.4, Function Reference**. For details of interrupt type and control, see **Section 8, Interrupt Controller**.

2.1 Features

The Cortex-A55 incorporates an extensively redesigned microarchitecture system that improves performance across the board while being extremely competitive in area and power efficiency.

Table 2.1 Function Summary

	Description
CPU	• 1.1 GHz Single Cortex-A55 r2p0 (Armv8-A) core
Cache memory	• L1$ (I/D) = 32 KB(Parity) / 32 KB (ECC) • L2$ = None • L3$ = 256 KB (ECC)

Table 2.2 Core Configuration

	Value	Description
L1 instruction cache size	32 KB	L1 instruction cache = 32 KB (Parity)
L1 data cache size	32 KB	L1 data cache = 32 KB (ECC)
L3 cache size	256 KB	L3 cache = 256KB (ECC)
ECC or parity core cache protection	Included	Support for core cache ECC
Advanced SIMD and floating-point support (including dot product instruction support)	Included	Support for NEON™ FPU engine
Cryptographic extension	Included (option)	Support for cryptography engine
Big endian	N/A	Not supported

2.2 Resets

Table 2.3 shows the supported types of reset and the areas that are reset by the respective types. For the procedures of handling the individual resets, see **Section 2.2.1, Cold Reset** and the subsequent parts. See the Arm® DynamIQ™ Shared Unit Technical Reference Manual and Arm® Cortex®-A55 Core Technical Reference Manual.

Table 2.3 Areas to be Reset

Reset Function	Areas to be Reset
Cold reset	Entire area
Cluster warm reset	Entire area of the cluster except for the debug, RAS, and ETM registers + core 0
Core warm reset	Entire area of the target core except for the debug, RAS, and ETM registers
Debug reset	Debug Block

The P-Channels can be used to control safety in resetting. For details on the P-Channels, see the AMBA® Low Power Interface Specification Arm® Q-Channel and P-Channel Interfaces. For details on registers for controlling the P-Channel and reset pins, see the chapter on the clock pulse generator (CPG). **Table 2.4** shows the supported power modes for each P-Channel. Transitions to power modes other than those listed in the table below are not supported.

Table 2.4 Supported Power Modes

PSTATE	Power Mode
CLUSTERPSTATE[6:0]	• 100_1000b (ON) • 000_0000b (OFF)
COREPSTATE0[5:0]	• 00_1000b (ON) • 00_0001b (OFF_EMU) • 00_0000b (OFF)

Reset address vector can be set by below Reset Vector Address Configuration Registers. For details, see the chapter of System Controller.

CA55 Core0 Reset Vector Address High/Low Configuration Register:
SYS_CA55_CFG_RVAH0 / SYS_CA55_CFG_RVAL0

To change reset vector address, follow the procedure below.

1. store codes to new reset address.
2. Write the new reset address to CA55 Reset Vector Address Configuration Registers.
3. Issue reset to target core.

2.2.1 Cold Reset

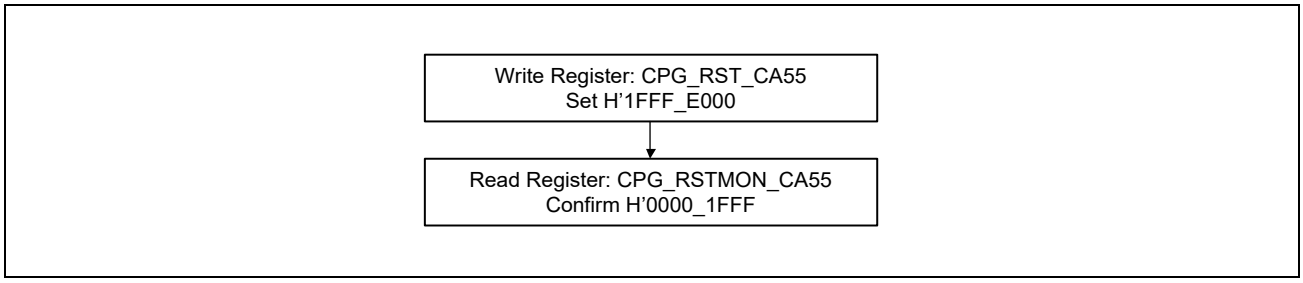
A cold reset is applied at any of the following cases.

(a) The sequence of a power-on reset by the CPG is executed.

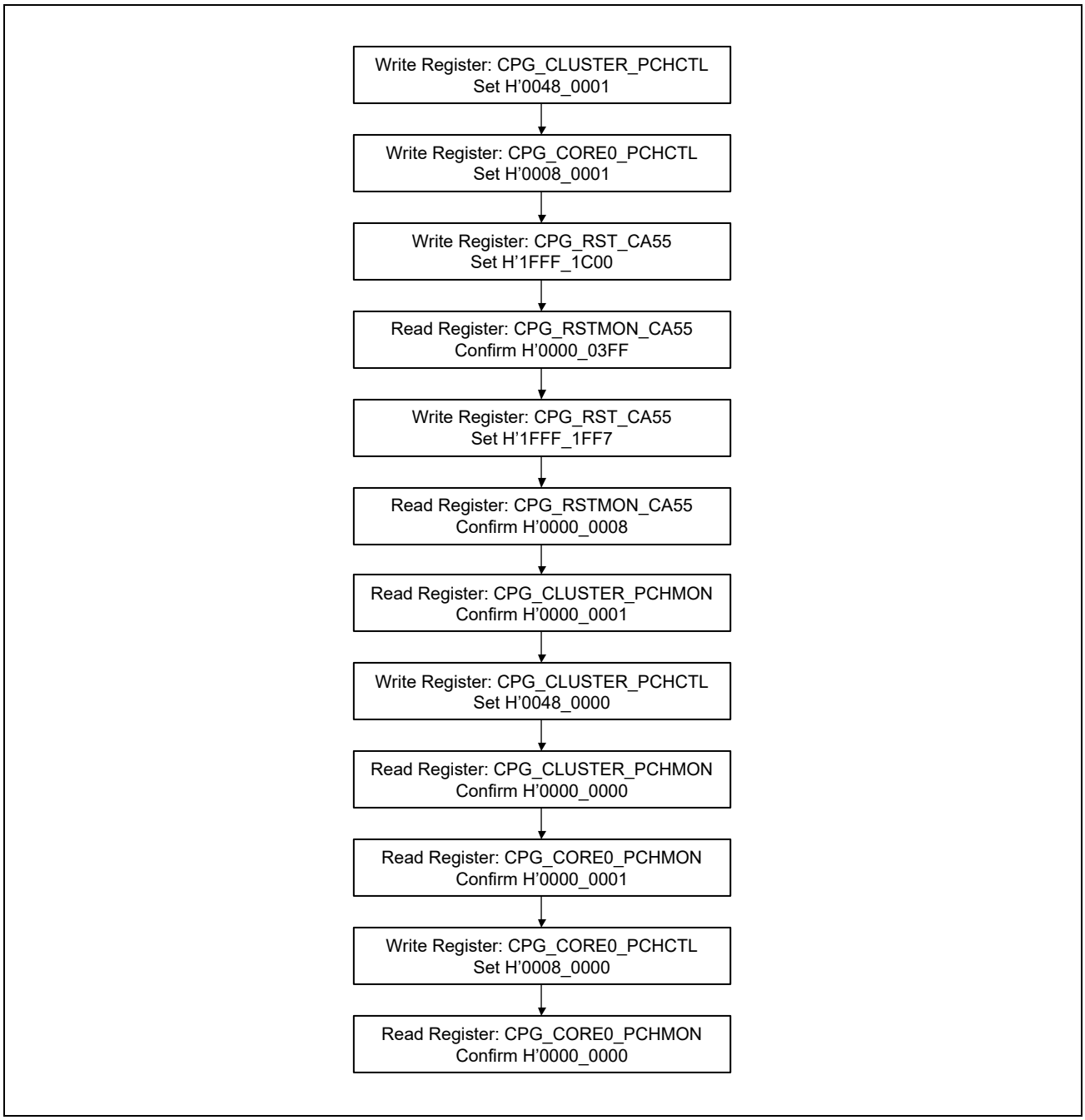
(b) The WDT counter has overflowed or a non-correctable error has been detected.

(c) Software control by the Cortex-M33 system CPU is applied.

For details on a and b reset, see the chapters on the clock pulse generator (CPG) and watchdog timer (WDT), respectively. To apply a cold reset by case c reset, follow the procedure below.



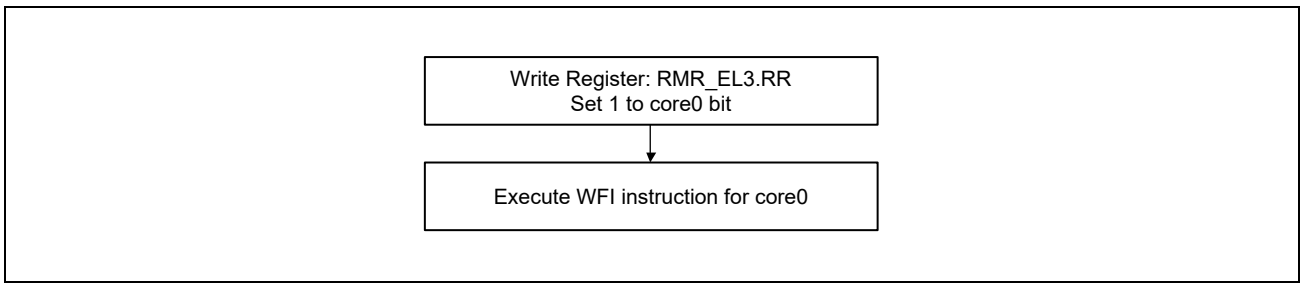
To release cold reset and start core 0 by case c reset, follow the procedure below.



2.2.2 Core Warm Reset

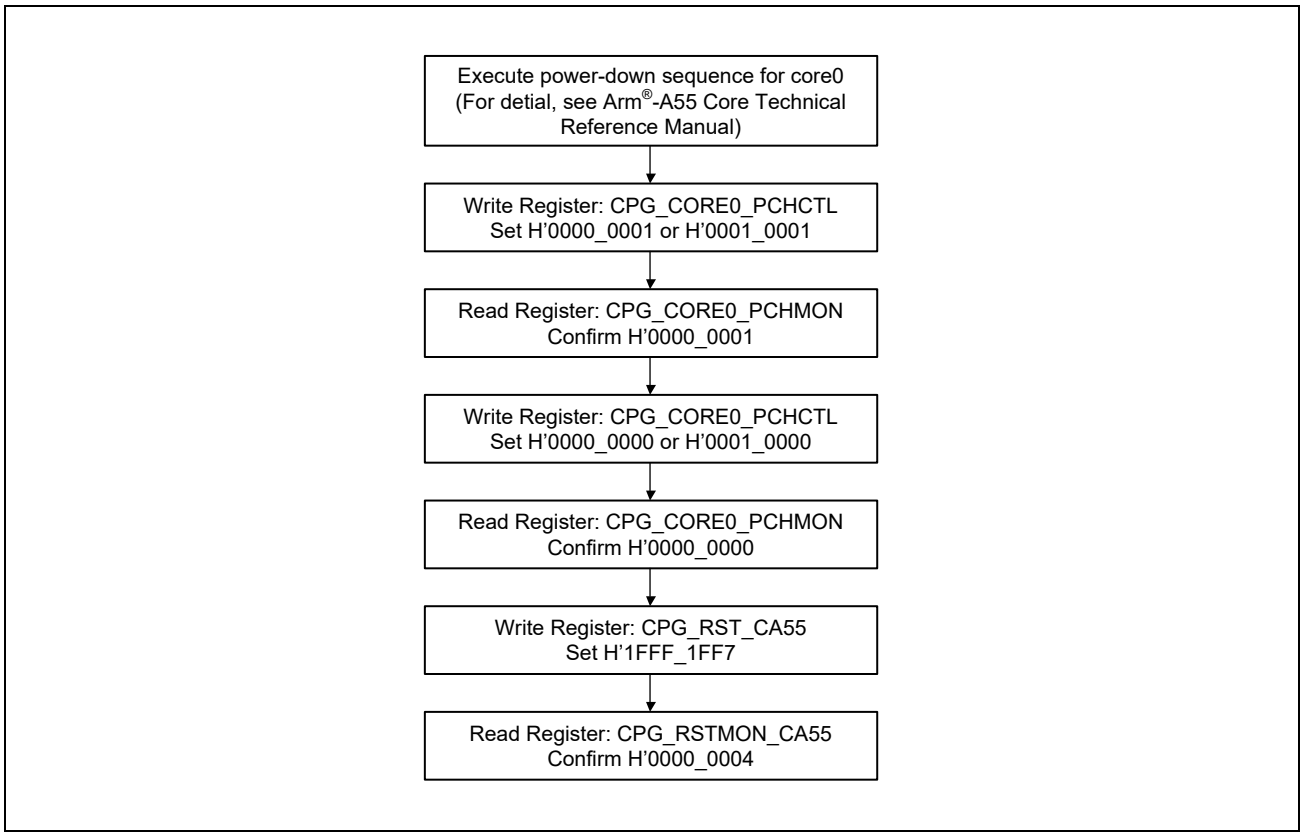
There are the two types of core warm reset: internal and external resets. The former requires software control by the Cortex-A55 system CPU itself and the latter requires software control by the Cortex-A55 and Cortex-M33 system CPUs. Though the Cortex-A55 system CPU handles an internal reset in the same way as an external reset, if you intend to reset a single core, we recommend using an internal reset because handling this requires fewer steps than an external reset. An external reset is mainly used as part of a cluster warm reset, which is described in **Section 2.2.3, Cluster Warm Reset**.

To apply an internal reset to core 0, follow the procedure below.

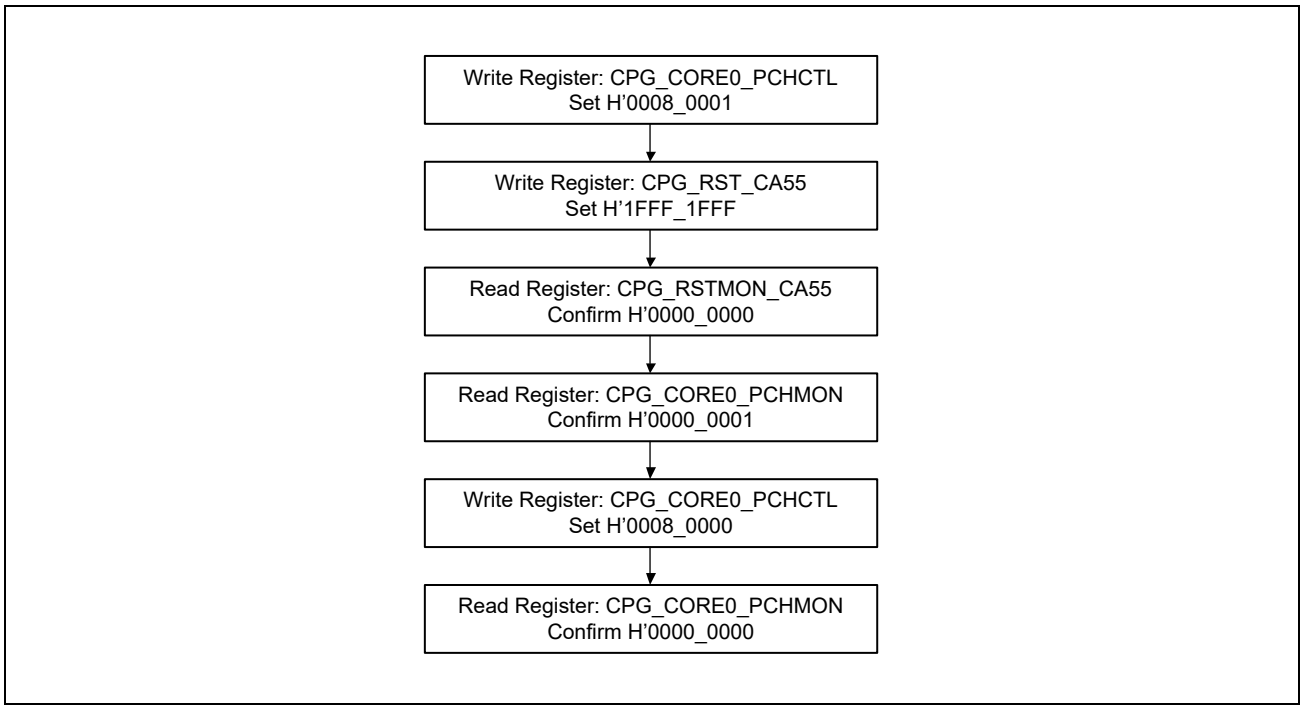


Release from an internal reset is automatic following application of the reset. No specific processing is required. For details, see the Arm® DynamIQ™ Shared Unit Technical Reference Manual and Arm® Cortex®-A55 Core Technical Reference Manual.

To apply an external reset to core 0, follow the procedure below.



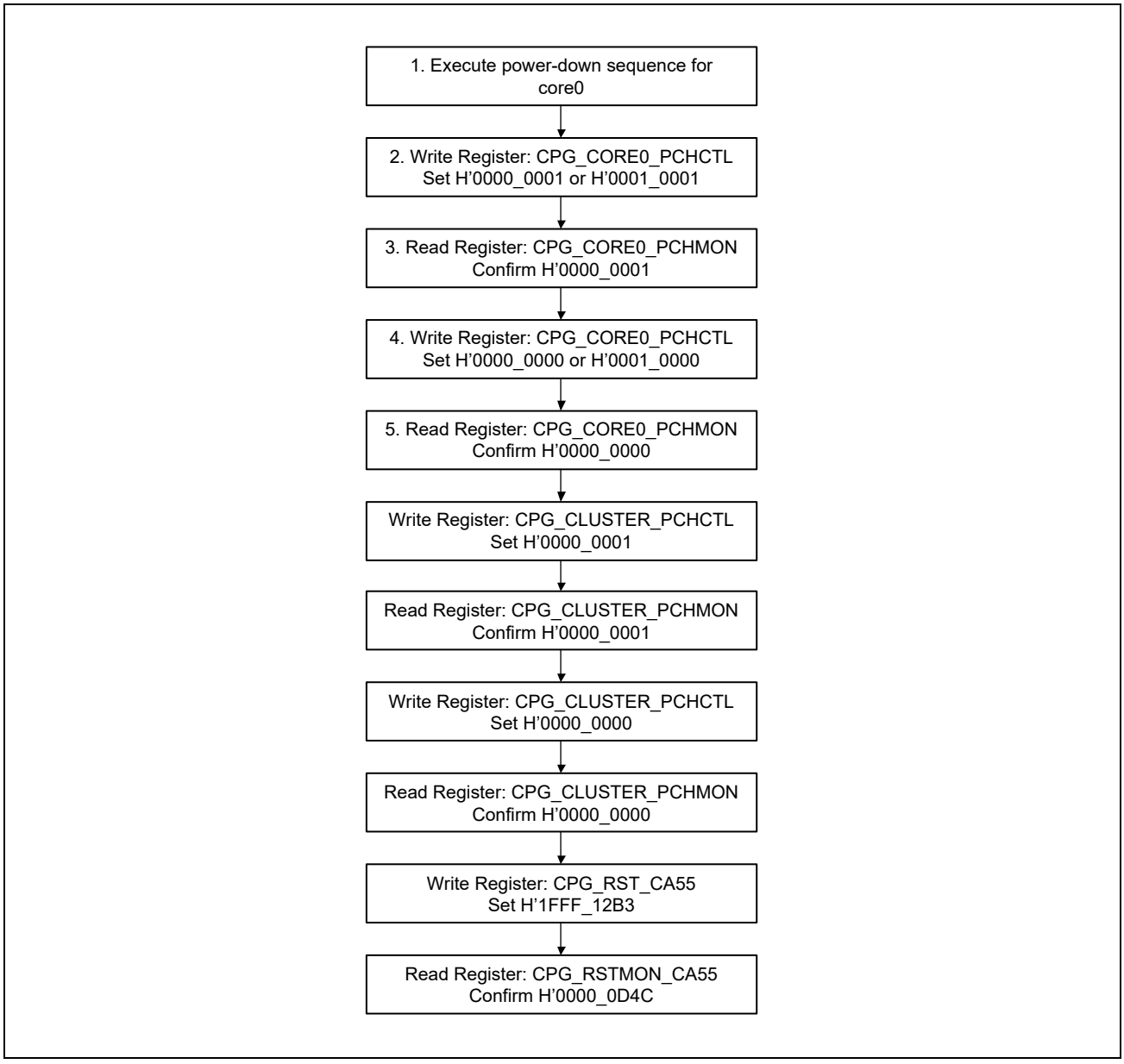
To release core 0 from the core warm reset state and start the core, follow the procedure below.



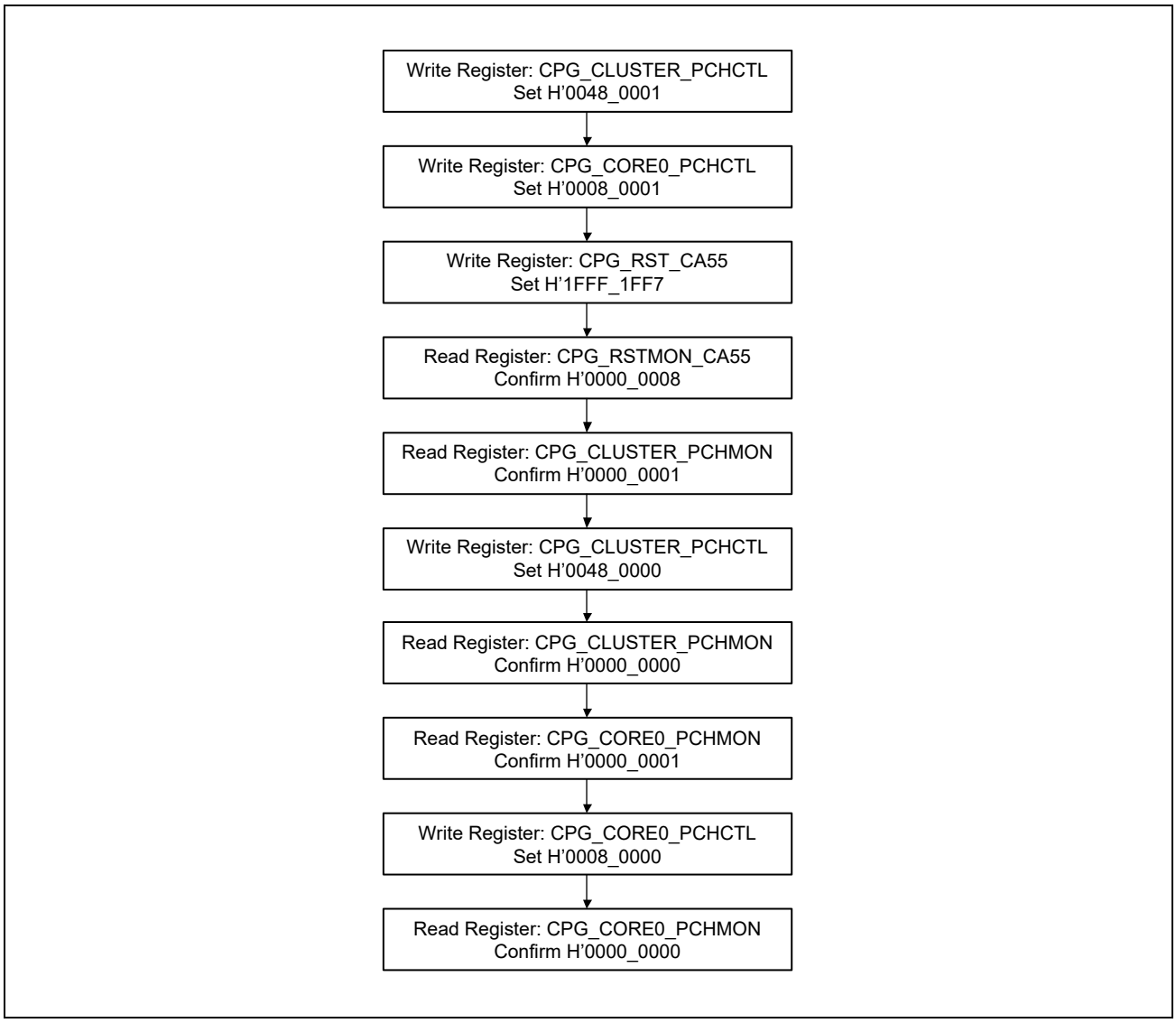
2.2.3 Cluster Warm Reset

This reset requires software control by the Cortex-A55 and Cortex-M33 system CPUs.

To apply this reset for core0, follow the procedure below.



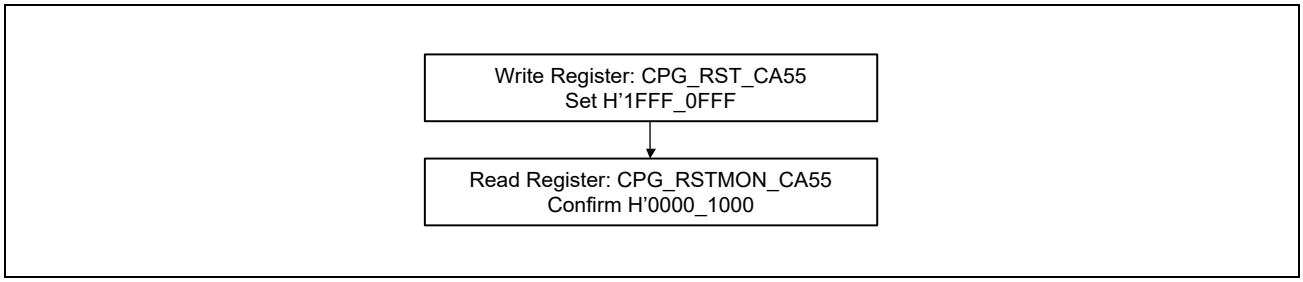
To release from the cluster warm reset state and start core 0, follow the procedure below.



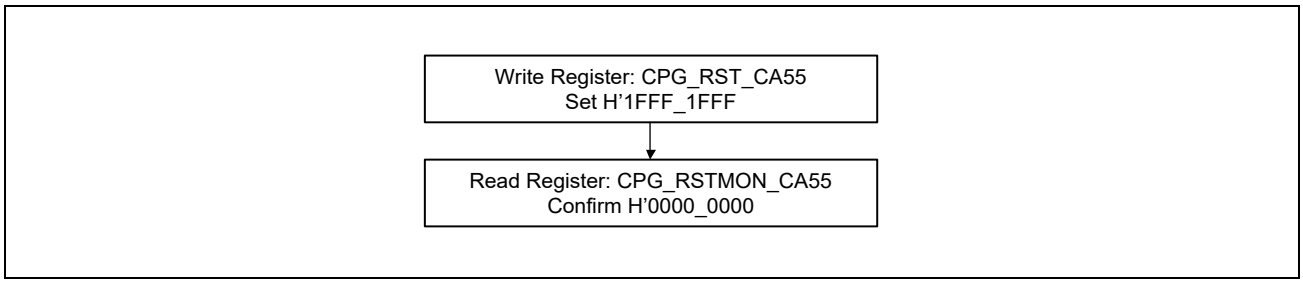
2.2.4 Debug Reset

Debug resets can only be applied under software control.

To apply a debug reset, follow the procedure below.



To release from the debug reset state, follow the procedure below.

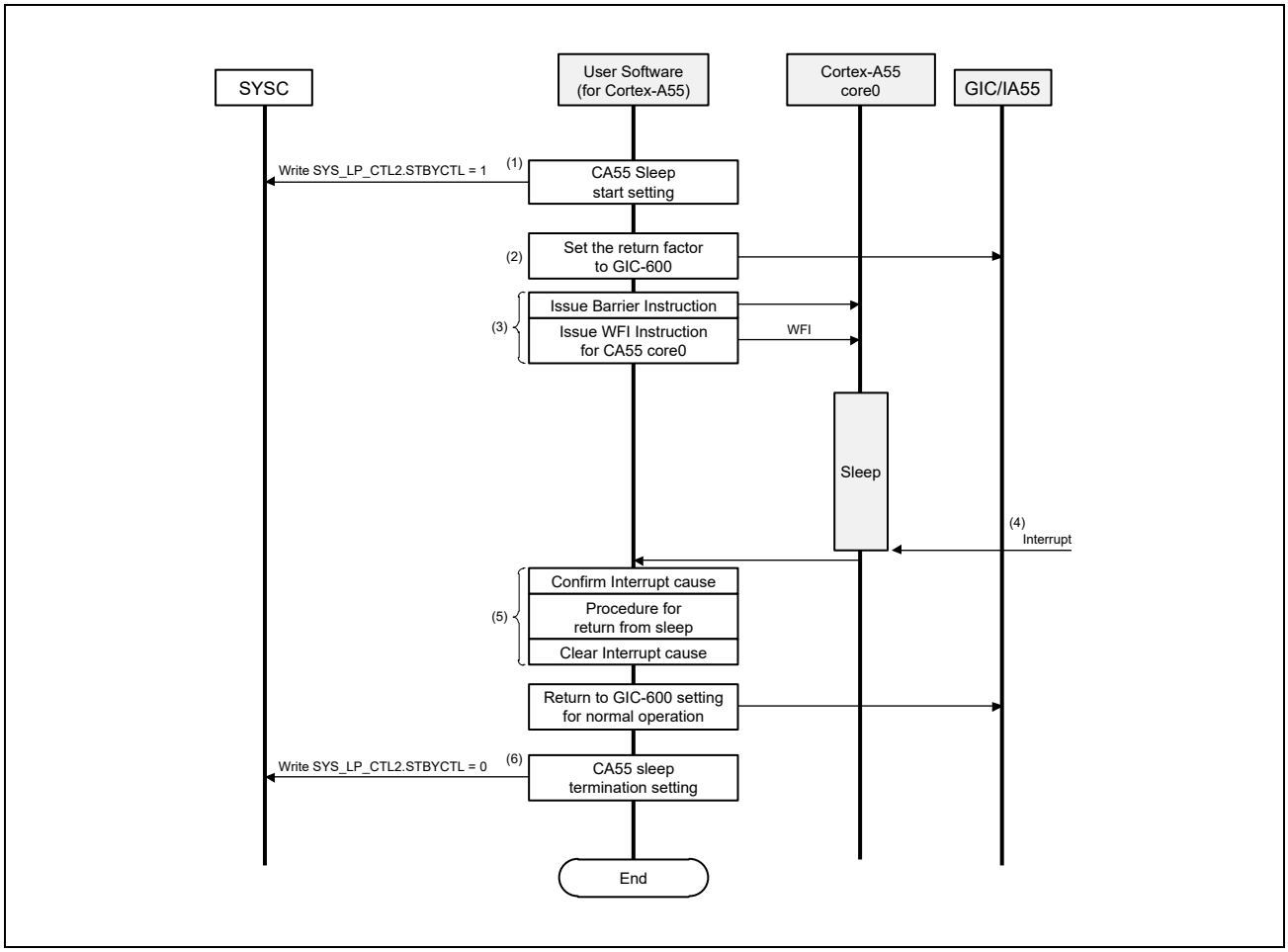


2.3 Low Power Consumption Mode

2.3.1 Cortex-A55 Sleep Mode

Cortex-A55 Sleep Mode of this LSI is a low power consumption mode obtained by having a core execute the WFI instruction. For details on the WFI instructions, see the Arm® Cortex®-A55 Core Technical Reference Manual and Arm® DynamIQ Shared Unit Technical Reference Manual.

The following procedure is the example sequence of a transition to Cortex-A55 Sleep Mode (core0).



(1) Set the STBYCTL bit of the SYS_LP_CTL2 register to 1 to start the Cortex-A55 Sleep Mode.

(2) Set the return factor to GIC. Enables only the interrupts used for return from the sleep mode and disables the interrupts used for the normal operation.

(3) Issue Barrier instruction.
Issue the WFI instruction for Cortex-A55 core0. The Cortex-A55 core0 is put into the sleep mode.

(4) Return from the sleep mode when the event occurs or the interrupt set in GIC occurs.

(5) Confirm the interrupt cause and perform the processing required for returning from the sleep mode.
And then clear the interrupt cause.
Set the end of interrupt to GIC.
Set the interrupt causes for the normal operation to GIC.

(6) Clear the STBYCTL bit of the SYS_LP_CTL2 register.

2.4 Function Reference

For details on the functions of the DynamIQ™ Shared Unit and Cortex-A55, see the documents listed below.

- Arm® DynamIQ Shared Unit Technical Reference Manual
- Arm® Cortex®-A55 Core Technical Reference Manual
- Arm® Cortex®-A55 Core Advanced SIMD and Floating-point Support Technical Reference Manual
- Arm® Cortex®-A55 Core Cryptographic Extension Technical Reference Manual

For details on clock and the controls, see the chapter on the clock pulse generator (CPG).

3. System CPU Cortex-M33/Cortex-M33_FPU

This LSI has the Cortex-M33 system CPU and the Cortex-M33_FPU system CPU.

The Cortex-M33 system CPU is a core block equipped with an Arm Cortex-M33 processor without floating-point.

The Cortex-M33_FPU system CPU is a core block equipped with an Arm Cortex-M33 processor with floating-point.

The Cortex-M33 processor is a highly energy-efficient processor that is intended for a microcontroller and deeply embedded applications. The processor is based on the Armv8-M architecture.

Refer to **Section 5.2.2, Cortex-M33/Cortex-M33_FPU Address Space** in **Section 5, LSI Internal Bus**.

3.1 Features

The following table shows the Cortex-M33/Cortex-M33_FPU Configuration.

For details on the functions of the Cortex-M33/Cortex-M33_FPU, see the Arm® Cortex®-M33 Processor Technical Reference Manual.

Table 3.1 Cortex-M33/Cortex-M33_FPU Configuration

Description	Configuration	
	Cortex-M33	Cortex-M33_FPU
Floating-point	Without floating-point	With floating-point
CPU	250 MHz Cortex-M33 r0p4 (ARMv8-M)	
DSP extension	No Armv8-M DSP extension	
Security extension	Armv8-M security extension	
Non-secure protected memory regions	16 regions	
Secure protected memory regions	16 regions	
Security attribution unit (SAU)	8 regions	
Interrupts	480 interrupts	
Number of bits of interrupt priority	8 bits are supported. 256 levels of priority are implemented.	
Debug watchpoints and breakpoints	Full set 4 data watchpoint comparators and 8 breakpoint comparators	
ITM and data watchpoint and trace (DWT) trace functionality	Complete ITM and DWT trace	
Embedded trace macrocell (ETM)	ETM instruction execution trace	
Micro trace buffer (MTB)	MTB is not supported.	
Cross trigger interface (CTI)	CTI is included.	
Wakeup interrupt controller (WIC)	WIC is included.	
External coprocessor interface	Coprocessor hardware is not supported.	

3.1.1 Interrupts

Table 3.2 Cortex-M33 System CPU Interrupt

Name	Description	Type	Active Level
CTIIRQ [1:0]	CTI interrupt output	Level	High

Table 3.3 Cortex-M33_FPU System CPU Interrupts

Name	Description	Type	Active Level
CTIIRQ [1:0]	CTI interrupt output	Level	High
FPIXC	Inexact	Level	High
FPIDC	Input denormal	Level	High
FPOFC	Overflow	Level	High
FPUFC	Underflow	Level	High
FPDZC	Divide-by-zero	Level	High
FPIOC	Invalid operation	Level	High

3.2 Address Space

For details on the address space, refer to the Arm® Cortex®-M33 Processor Technical Reference Manual.

3.2.1 IDAU Setting

The Cortex-M33/Cortex-M33_FPU system CPU uses the IDAU. **Table 3.4** shows the security attributes assigned by the IDAU.

Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. After registers are set, release Cortex-M33/Cortex-M33_FPU Reset.

Default status:

0: Secure (Lower), Non-Secure (Upper)
1: Secure (Upper), Non-Secure (Lower)

Table 3.4 IDAU Security Attributes

Area No.	Security Attribute	Lower Limit of Address Range	Upper Limit of Address Range
0	Non-secure	H'0000_0000	H'0FFF_FFFF
1	Secure	H'1000_0000	H'1FFF_FFFF
2	Non-secure	H'2000_0000	H'2FFF_FFFF
3	Secure	H'3000_0000	H'3FFF_FFFF
4	Non-secure	H'4000_0000	H'4FFF_FFFF
5	Secure	H'5000_0000	H'5FFF_FFFF
6	Non-secure	H'6000_0000	H'6FFF_FFFF
7	Secure	H'7000_0000	H'7FFF_FFFF
8	Non-secure	H'8000_0000	H'8FFF_FFFF
9	Secure	H'9000_0000	H'9FFF_FFFF
10	Non-secure	H'A000_0000	H'AFFF_FFFF
11	Secure	H'B000_0000	H'BFFF_FFFF
12	Non-secure	H'C000_0000	H'CFFF_FFFF
13	Secure	H'D000_0000	H'DFFF_FFFF
14	Non-secure	H'E000_0000	H'EFFF_FFFF
15	Secure	H'F000_0000	H'FFFF_FFFF

3.3 Register Descriptions

For details on registers, refer to the Arm® Cortex®-M33 Processor Technical Reference Manual.

For the device-specific Cortex-M33/Cortex-M33_FPU system CPU control registers, see the section on the system controller (SYSC).

3.4 Description of Functions

For the registers and functions of Cortex-M33/Cortex-M33_FPU, refer to the Arm® Cortex®-M33 Processor Technical Reference Manual and Arm®v8-M Architecture Reference Manual.

3.4.1 Startup Sequence

The startup sequence of the Cortex-M33/Cortex-M33_FPU system CPU is shown in the figure below.

It is designed to execute by other processor (Cortex-A55) for the function and sequence of Cortex-M33/Cortex-M33_FPU control here.

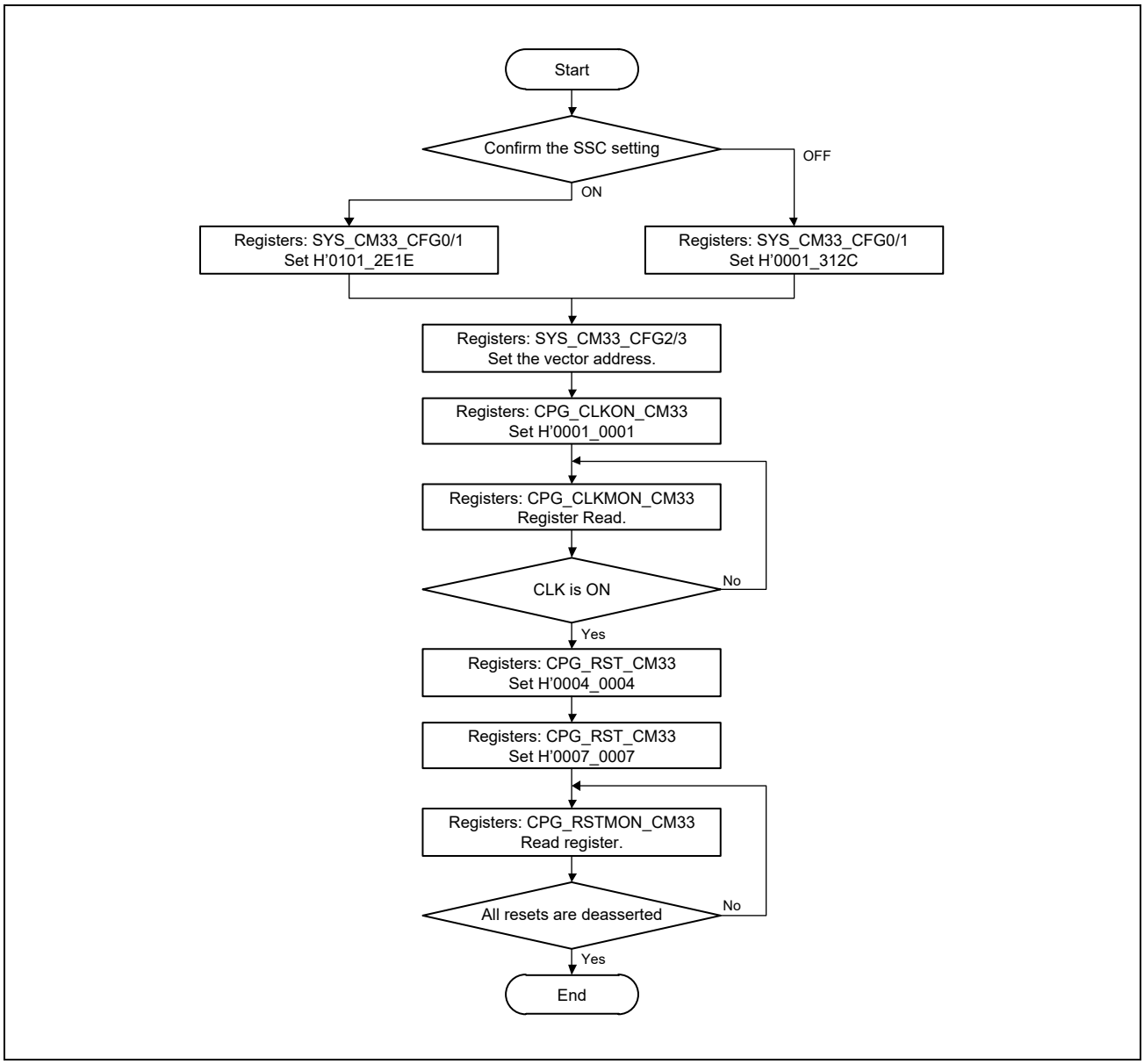


Figure 3.1 Startup Sequence (Cortex-M33)

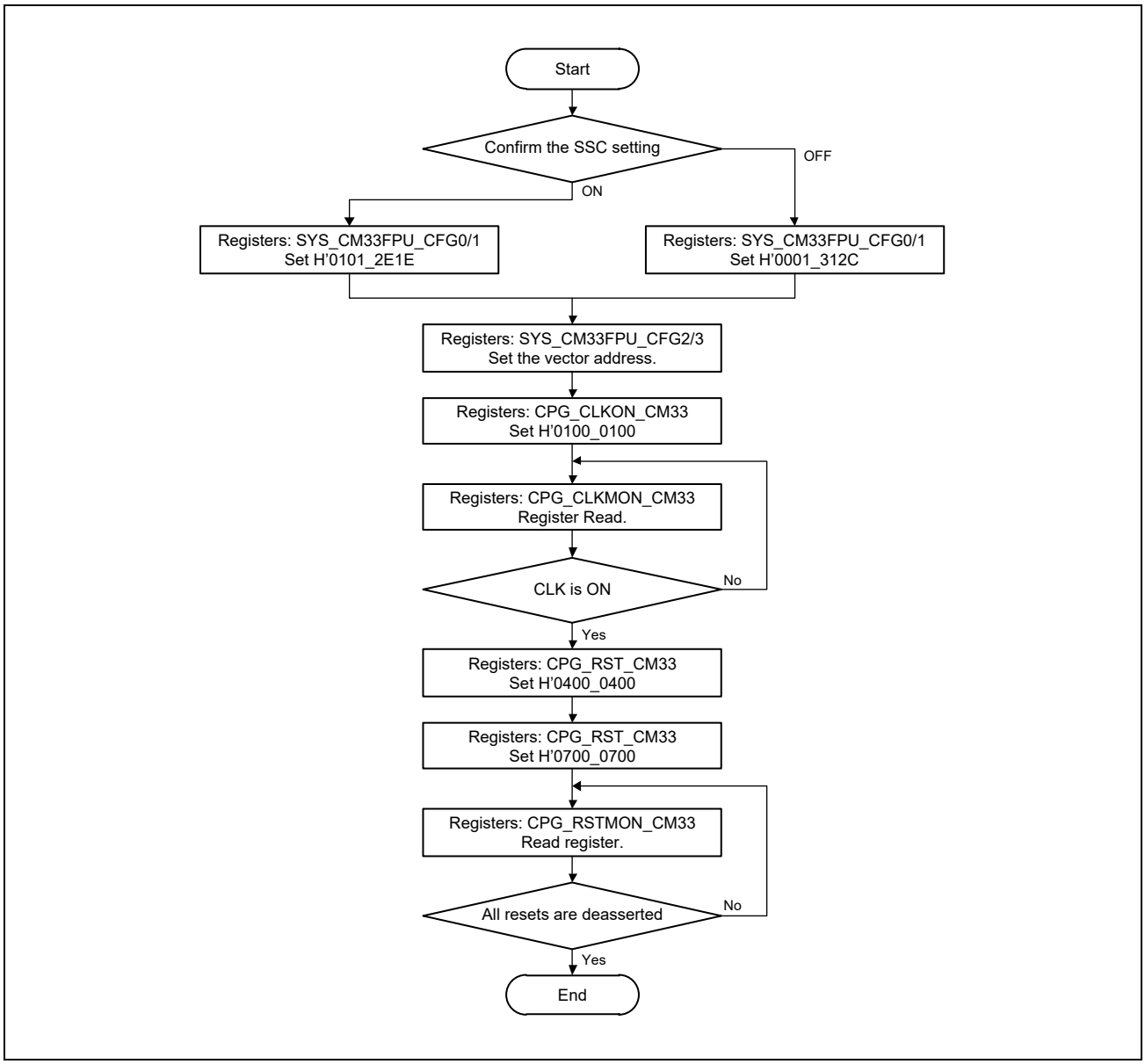


Figure 3.2 Startup Sequence (Cortex-M33_FPU)

3.4.2 Control by the SYSC

In this LSI, the system controller (SYSC) is used to control the functions of the Cortex-M33/Cortex-M33_FPU system CPU. For details of SYSC registers, see the section on the system controller (SYSC).

Table 3.5 List of Functions Controlled by the SYSC

No.	Description
1	Cortex-M33/Cortex-M33_FPU control registers • CM33/CM33_FPU Config registers (SysTick timer, reset vector address): (SYS_CM33_CFGx/SYS_CM33FPU_CFGx) • Low-power Sequence CM33/CM33_FPU Control Register0: (SYS_LP_CM33CTL0/SYS_LP_CM33_CTL1/SYS_CM33_CTL, SYS_LP_CM33FPUCTL0/SYS_LP_CM33FPUCTL1/SYS_CM33FPU_CTL)
2	Warm reset request Cooperation with the SYSC is necessary for executing a warm reset. For details, see **Section 3.4.3, Warm Reset**.
3	Control of low power consumption function The Cortex-M33/Cortex-M33_FPU system CPU supports two low power modes. For details on how to control Cortex-M33/Cortex-M33_FPU Sleep Mode, see **Section 3.4.4, Cortex-M33/Cortex-M33_FPU Sleep Mode**.

3.4.3 Warm Reset

When executing a warm reset, the WFI instruction is used to guarantee that no transaction is issued.

The warm reset sequence is shown below. For a details of SYSC registers, see the section on the system controller (SYSC).

1. Set the SYSRESETREQ bit in the Cortex-M33/Cortex-M33_FPU register AIRCR to 1.
2. Confirm that the value of the SYSRESETREQ bit in the SYSC register SYS_LP_CM33CTL0/SYS_LP_CM33FPUCTL0 is 1.
3. Set the IM33_MASK/IM33FPU_MASK bit in the SYSC register SYS_LP_CTL7 to 1.
4. Issue a WFI instruction.*[1]

The above procedure is used to execute a warm reset. Clearing the IM33_MASK/IM33FPU_MASK bit and release from the warm reset state are controlled by hardware.

Note 1. If the WFE instruction is executed during a warm reset in this LSI, correct operation cannot be guaranteed.

3.4.4 Cortex-M33/Cortex-M33_FPU Sleep Mode

In this LSI, the sleep mode for the Cortex-M33/Cortex-M33_FPU system CPU is Cortex-M33/Cortex-M33_FPU Sleep Mode. In this mode, some modules (SysTick timers, NVIC, etc.) of the core can operate.

The following procedure is the example sequence of a transition to Cortex-M33/Cortex-M33_FPU Sleep Mode.

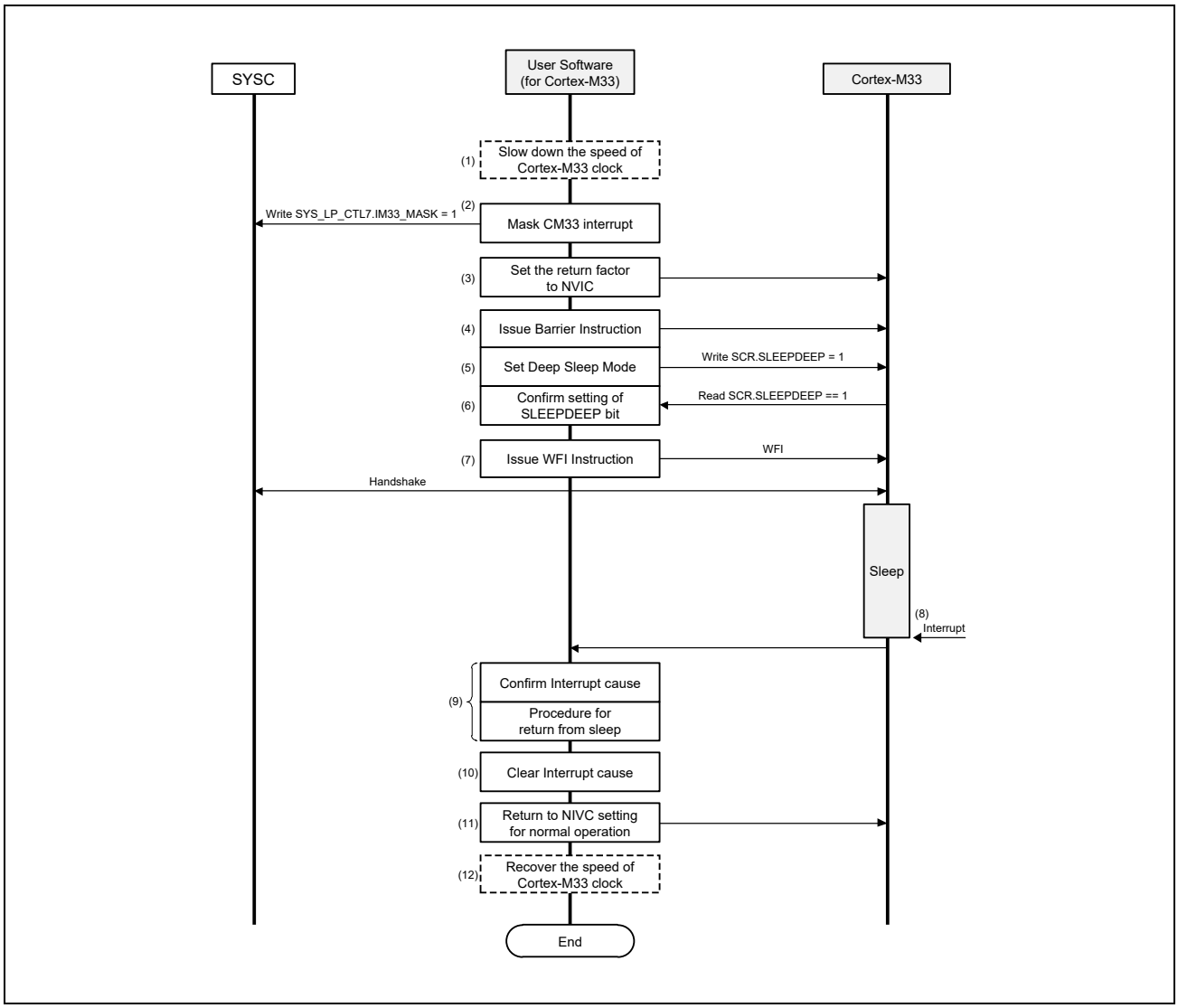


Figure 3.3 Cortex-M33 Sleep Mode Procedure

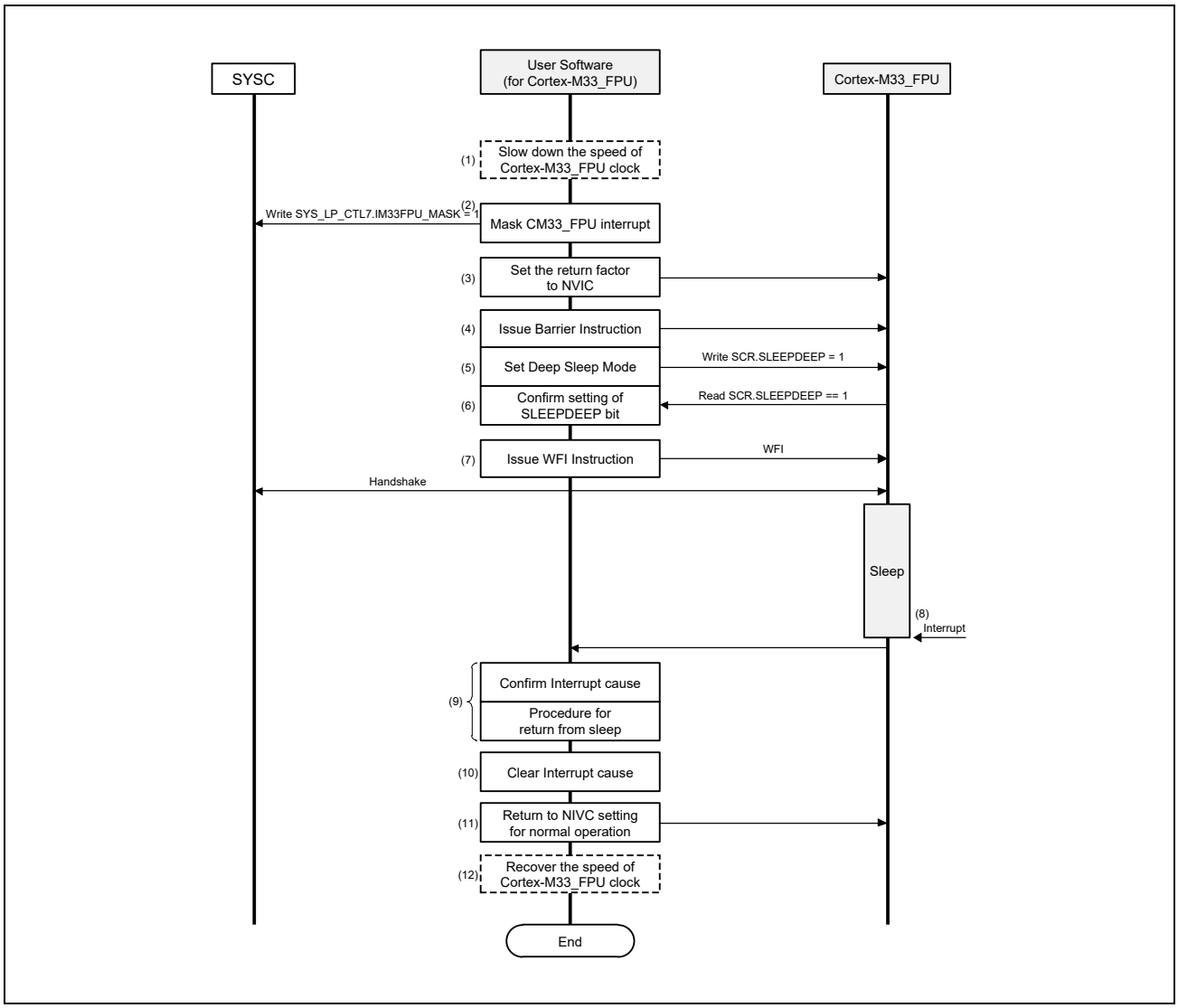


Figure 3.4 Cortex-M33_FPU Sleep Mode Procedure

(1) Slow down the speed of Cortex-M33/Cortex-M33_FPU clock.*1

(2) Set the IM33_MASK/IM33FPU_MASK bit of the SYS_LP_CTL7 register to 1.
Interrupts to IM33/IM33FPU are masked and interrupts from IM33/IM33FPU to Cortex-M33/Cortex-M33_FPU are not accepted.
The IM33_MASK/IM33FPU_MASK bit is automatically cleared after completing the transition to Cortex-M33/Cortex-M33_FPU Sleep Mode.
Refer to the SYS_LP_CTL7 register in the SYSC section.

(3) Set the return factor to NVIC. Enables only the interrupts used for return from Cortex-M33/Cortex-M33_FPU Sleep Mode and disables the interrupts used for the normal operation.*2

(4) Issue Barrier instruction.

(5) Set the SLEEPDEEP bit of the SCR register in Cortex-M33/Cortex-M33_FPU to 1.*3

(6) Read SCR register to confirm whether the SLEEPDEEP bit is set to 1.

(7) Issue a WFI instruction to enter Cortex-M33/Cortex-M33_FPU Sleep Mode.

(8) Return from Cortex-M33/Cortex-M33_FPU Sleep Mode when the event occurs or the interrupt set in NVIC occurs.

(9) Confirm the interrupt cause and perform the processing required for returning from Cortex-M33/Cortex-M33_FPU Sleep Mode.*2

(10) And then clear the interrupt cause.*2

(11) Set the interrupt causes for the normal operation to NVIC.*2

(12) Recover the speed of Cortex-M33/Cortex-M33_FPU clock.*1

Note 1. Switching clock (I2ϕ/I3ϕ, all Cortex-M33/Cortex-M33_FPU clock systems) frequency is controlled by software in this LSI. Refer to the CPG_PL3B_DDIV register in the CPG section.
Note that the step of slowing down clock (I2ϕ/I3ϕ) is not mandatory.

Note 2. See Arm® Cortex®-M33 Processor Technical Reference Manual for NVIC.

Note 3. To make SLEEPDEEP bit 1 in the case of non-secure mode, SLEEPDEEPS bit is required to be 0. See Armv8-M Architecture Reference Manual for details.

3.4.5 SysTick Timers

The SysTick timers in the Cortex-M33/Cortex-M33_FPU are described below.

- There is one Secure SysTick timer and one Non-secure SysTick timer.
- The count cycle of SysTick timers is common to Secure and Non-secure.
- A reference clock for counting by a SysTick timer is implemented in this LSI.
- Though the cycle of the reference clock is constant regardless of the clock(I2ϕ/I3ϕ) cycle, the accuracy is degraded when the clock(I2ϕ/I3ϕ) frequency is switched. Therefore, the accuracy of counting is not guaranteed when there is clock(I2ϕ/I3ϕ) frequency switchover.
- The configuration of SysTick timers can be set in the SYSC registers SYS_CM33_CFG0, SYS_CM33_CFG1, SYS_CM33FPU_CFG0 and SYS_CM33FPU_CFG1*[1]. Before the Cortex-M33/Cortex-M33_FPU system CPU is started up, set the following values in accordance with the PLL3 SSCG being turned on or off*[2]. SSCG clock is used for SysTick timer when turned on.

  When turned on: H'0101_2E1E
  When turned off: H'0001_312C

The above settings are reflected in the Cortex-M33/Cortex-M33_FPU register SYST_CALIB*[3]. The settings of the 24 lower-order bits and the 25th bit are reflected in the TENMS and SKEW bits, respectively. When the SSCG is ON, set the SKEW bit to 1. The setting of the 26th bit is reflected in the NOREF bit. Set the bit to 0 because a reference clock is implemented in this LSI.

Note 1. The register settings are reflected after release from the cold reset state.
Note 2. For details on turning on and off the SSCG, see the section on the clock pulse generator (CPG).
Note 3. For details on the SYST_CALIB register, refer to the Arm®v8-M Architecture Reference Manual.

3.4.6 Restrictions of Functions

The Cortex-M33 system CPU has some restrictions on the functions due to specification of this LSI. **Table 3.6** lists the restrictions on the functions.

Table 3.6 List of Restrictions of Functions

No.	Description
1	WFE instruction is not supported in a warm reset. In this LSI, operation of the WFE instruction is not guaranteed while a warm reset is being executed.
2	SysTick timer accuracy Though the clock(I2ϕ/I3ϕ) frequency can be switched dynamically in this LSI, the timer accuracy cannot be guaranteed when there is a dynamic frequency switchover during the measurement period.

4. Boot Mode

4.1 Overview

This LSI has two cold boots and four boot modes.

Boot mode 0: Booting from 3.3-V eSD*[1]

Boot mode 1: Booting from 1.8-V/3.3-V eMMC*[2]

Boot mode 2: Booting from the 1.8-V/3.3-V Single, Quad, or Octal*[3] serial flash memory

Boot mode 3: Booting from the program downloaded through the serial communications with FIFO (SCIF)

Note 1. Embedded SD (eSD) defined in the SD Specification Part 1 eSD Addendum (Version 2.10)
Note 2. eMMC supporting the boot operation mode prescribed in JEDEC STANDARD JESD84 A44 (MMCA 4.4)
Note 3. This LSI supports Boot from Octa Flash.

This LSI has two boot methods: Cortex-M33 cold boot (Cortex-M33 w/o FPU) and Cortex-A55 cold boot. Which CPU to cold boot from is controlled by BOOTCPUSEL. BOOTCPUSEL is described in **Table 4.1**.

Table 4.1 Selection of Boot CPU

Signal Name	Function	Description
BOOTCPUSEL	BOOT CPU Select	Select the CPU to cold boot. 0: Cortex-M33 cold boot 1: Cortex-A55 cold boot

During the Cortex-M33 cold boot, you can choose to boot from either a Serial Flash Memory device or an SCIF download boot. During the Cortex-A55 cold boot, you can choose to boot from one of the following devices: eSD, eMMC, Serial Flash Memory, or SCIF download boot.MD_BOOT0/1 controls which device to boot from. For booting from eMMC and Serial Flash Memory, set the voltage to 3.3 V or 1.8 V in MD_BOOT2.MD_BOOT0/1/2 are shown in **Table 4.2** and **Table 4.3**. After the reset is released, the Cortex-A55 or Cortex-M33 (w/o FPU) starts after performing the prescribed procedures for clock and reset by the CPG described later. After booting the other CPUs, select Operation ON/OFF in the user program as necessary.

Table 4.2 Selection of Boot Mode

Signal Name		Boot Mode	Interface Module	Connected Device	Cortex-A55 Cold Boot	Cortex-M33 Cold Boot
MD_BOOT1	MD_BOOT0					
0	0	Boot mode 0	SDHI0	eSD (3.3 V at startup)	○	×
0	1	Boot mode 1	SDHI0	1.8-V/3.3-V eMMC	○	×
1	0	Boot mode 2	xSPI	1.8-V/3.3-V Single, Quad, or Octal serial flash memory	○	○
1	1	Boot mode 3	SCIF0	Downloading through SCIF	○	○

Table 4.3 Boot Device I/F Voltage Setting Pin

Signal Name	Function	Description
MD_BOOT2	Boot Device I/F Voltage Settings	Boot Device I/F Voltage Settings This pin is valid only for boot mode 1 (eMMC) and boot mode 2 (Serial Flash Memory) 0: 3.3 V 1: 1.8 V

4.2 Operation

4.2.1 Boot Mode 0 (eSD)

Table 4.4 shows the interface signals used for connection with the external device in boot mode 0 (eSD). This LSI supports the embedded SD (eSD) defined in the SD Specification Part 1 eSD Addendum (Version 2.10).

Table 4.4 External Interface Signals Used in Boot Mode 0

Interface Module	Pin Name	I/O	Function	Pin Type
SDHI/eMMC (SDHI0)	SD0_CLK	Output	SD clock	Dedicated pins
	SD0_CMD	Input/output	SD command or response	
	SD0_DATA0	Input/output	SD data 0	
	SD0_DATA1	Input/output	SD data 1	
	SD0_DATA2	Input/output	SD data 2	
	SD0_DATA3	Input/output	SD data 3	
	SD0_CD*1	Input	SD card detection	Multiplexed pins
	SD0_WP*1	Input	SD write protection	

Note 1. This terminal is not used at boot time.

4.2.1.1 External Connections

Figure 4.1 shows the connections with the eSD device.

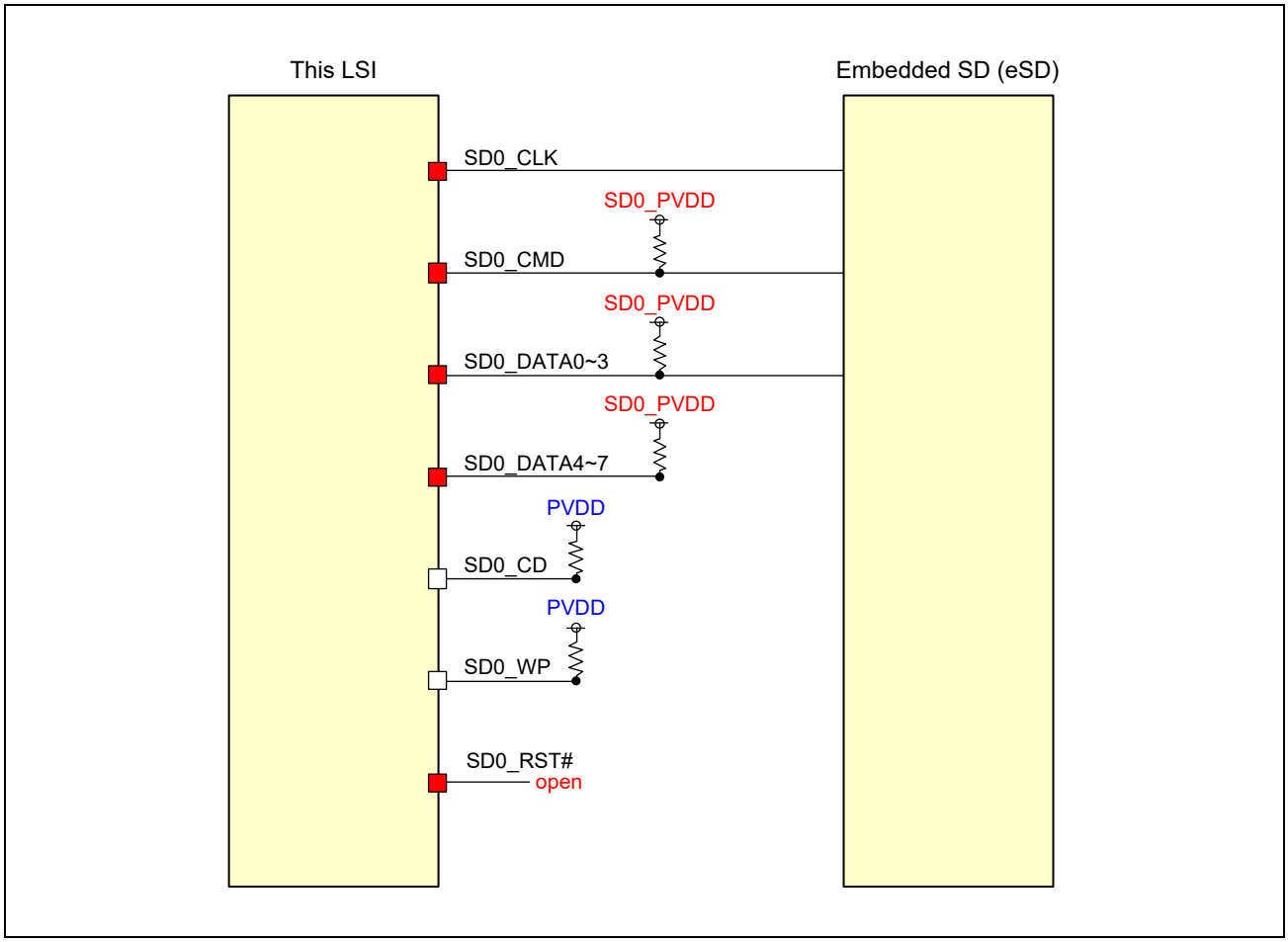


Figure 4.1 Connections in Boot Mode 0

4.2.1.2 Overview of Operation

This mode is used to boot this LSI from the user program stored in the eSD device. The operating voltage is set to 3.3 V and the width of the data bus connected to the eSD device is fixed to four bits in this mode. SDHI0 used as the interface controller in this mode is placed in the module standby mode at the startup of the LSI. Once SDHI0 is released from the module standby mode, the boot program in this LSI sets up the CPG so that the frequency of the external output clock becomes 16.6 MHz, initializes SDHI0, and then begins the booting process with the I/O buffers set to the maximum driving ability.

4.2.1.3 Operation of Booting from eSD

In boot mode 0 (booting from eSD), this LSI is booted from the loader program in the eSD device connected to channel 0 of the SD host interface (SDHI) as follows.

The boot program executes the following processing to access the eSD device.

1. Setting up the necessary peripheral modules (SDHI channel 0 and GPIO)
2. Set drive capability value
3. Mounting the eSD device
4. Issue a read command to the eSD device via SDHI to retrieve the loader program size block from sector 1 of the selected partition and store it in the H'0_000A_1E00 of the on-chip RAM

Obtain the loader program size, loader program load address, and loader program destination address from the stored loader program size block.

After acquiring the loader program size block, SDHI issues a read command to the eSD device to transfer the loader program data from the loader program load address (sector conversion) of the selected partition to the loader program destination address of the on-chip RAM corresponding to the loader program size.

After the loader program transfer is completed, SDHI channel 0 is returned to its initial state, and device information and boot results are stored in the on-chip RAM.

Then, by branching to the first address of the loader program transferred to the on-chip RAM (loader program destination address), the loader program stored in the eSD device is executed.

If the boot program has failed to read data, it reads data from a reserved area. If reading has failed for all reserved areas, the boot program enters fail-safe mode (SCIF downloading mode). If the fail-safe processing has failed, execution enters an infinite loop in the on-chip ROM and the boot processing is terminated.

4.2.1.4 Allocation of the Loader Programs in the eSD Device

(1) Allocation in eSD V2.0 (Single Partition)

Figure 4.2 shows the allocation of the loader program size blocks and loader programs in the eSD device conforming to the eSD V2.0 standard. To prevent read-disturb errors, up to seven loader program size blocks and loader programs can be multiplexed and written to the eSD device.

The loader program size blocks are allocated to sectors 1 to 7. In the loader program size block, place the loader program size in the first 4 bytes, the loader program load address in 4 bytes at offset H'10, and the loader program destination address in 4 bytes at offset H'20. Place the signature H'AA55 in the last two bytes. For more information about the loader program size block, refer to **Figure 4.2** Loader Program Size Block. The configuration of the loader program size block is shown in **Figure 4.3**. Note that the loader program size, loader program load address, and loader program destination address are assumed to be stored in 4-byte little-endian.

The following describes the operation of transferring the loader program.

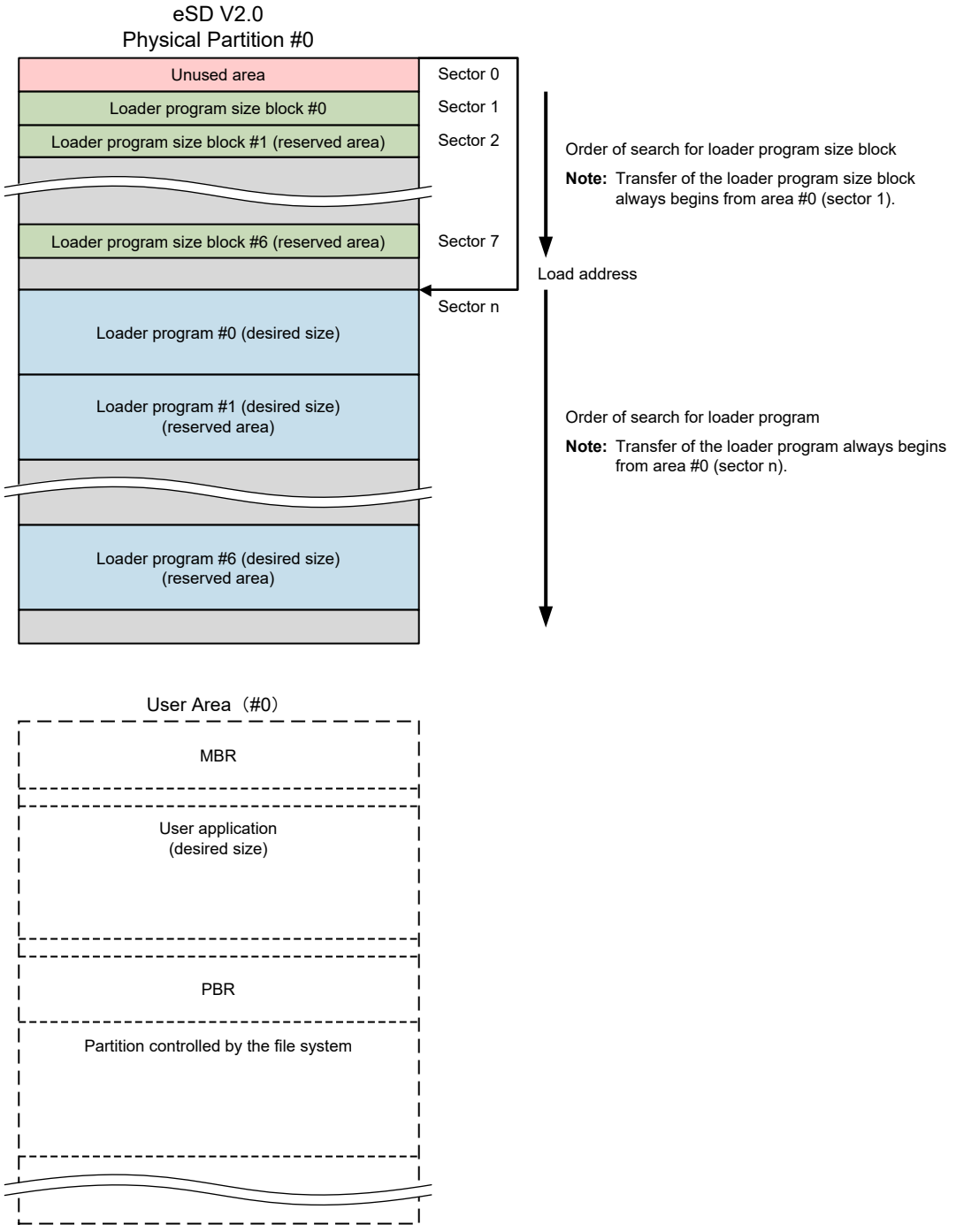
(1) The loader program size block is transferred from sector 1 in physical partition #0 of the eSD device to the on-chip RAM.
If a communication error occurs during data transfer, read access is retried up to three times per area. If the retry processing is repeated three times during the transfer of the loader program size block in a single area, another loader program size block is transferred from the next multiplexed area.

(2) The signature in the transferred loader program size block is checked. When the signature matches the expected value (H'AA55), the size of the loader program data is obtained from the loader program size block and the processing for transferring the loader program is executed.
If the signature does not match the expected value, another loader program size block is transferred from the next multiplexed area.

(3) Transfer the loader program to the on-chip RAM in order from the loader program load address (in sectors) of Physical Partition #0 of the eSD device to the loader program data size.
If a communication error occurs during data transfer, read access is retried up to three times per area. If the retry processing is repeated three times during the loader program transfer in a single area, another loader program is transferred from the next multiplexed area.

NOTE

Even when the loader size block is transferred from a multiplexed (reserved) area, the transfer of the loader program always begins from the area for loader program #0.



Note 1. Write the same data to loader program size block areas #0 to #6.

Note 2. Write the same data to loader program areas #0 to #6.

Note 3. The start sector location of the loader program is calculated as follows.
Loader program load address (in sectors) + retrieved size (in sectors) × n

Figure 4.2 Allocation of Loader Programs in the eSD Device Conforming to the eSD V2.0 Standard

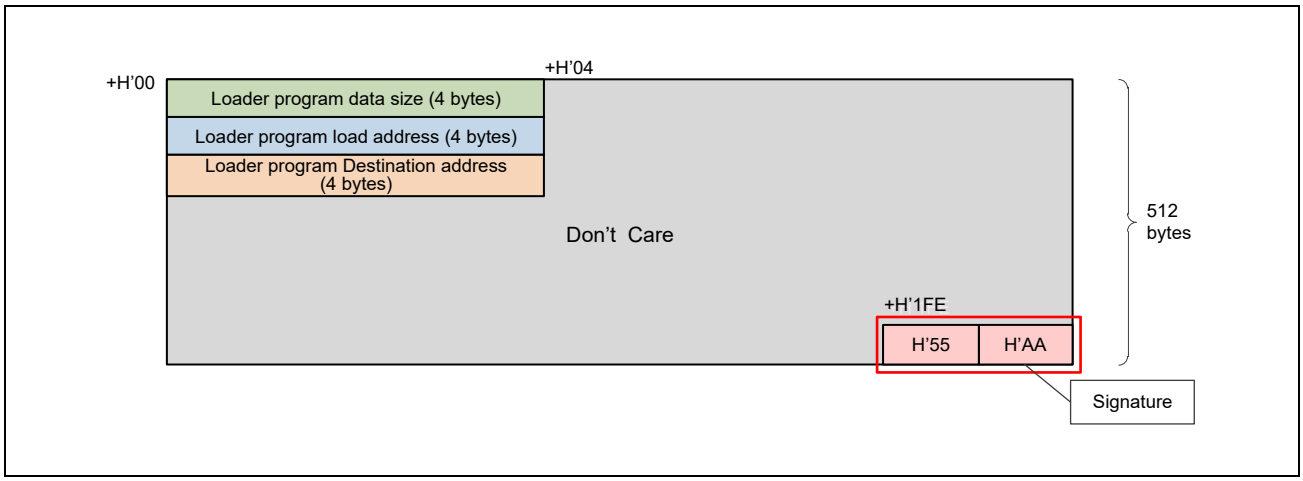


Figure 4.3 Structure of the Loader Program Size Block

NOTE

The loader program data size, loader program load address, and loader program destination address are expected to be stored in 4-byte little-endian.

(2) Allocation in eSD V2.1 (Multi Partitions)

Figure 4.4 shows the allocation of the loader program size blocks and loader programs in the eSD device conforming to the eSD V2.1 standard. To prevent read-disturb errors, up to seven loader program size blocks and loader programs can be multiplexed and written to the eSD device

The specifications and allocation of loader program size blocks are the same as those in eSD V2.0. For the structure of the loader program size block, refer to **Figure 4.3**.

The following describes the operation of transferring the loader program.

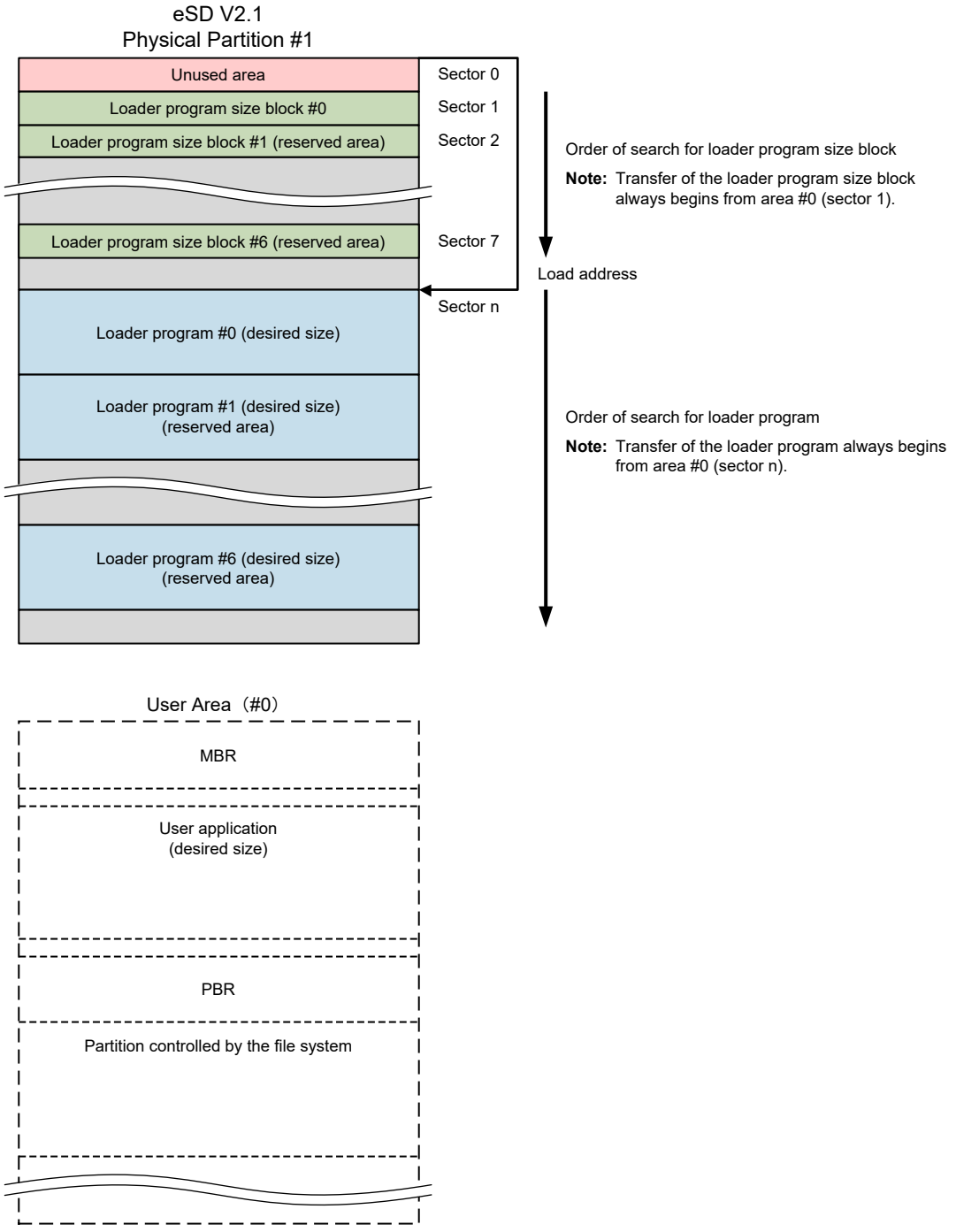
(1) The loader program size block is transferred from sector 1 in physical partition #1 of the eSD device to the on-chip RAM.
If a communication error occurs during data transfer, read access is retried up to three times per area. If the retry processing is repeated three times during the transfer of the loader program size block in a single area, another loader program size block is transferred from the next multiplexed area.

(2) The signature in the transferred loader program size block is checked. When the signature matches the expected value (H'AA55), the size of the loader program data is obtained from the loader program size block and the processing for transferring the loader program is executed. If the signature does not match the expected value, another loader program size block is transferred from the next multiplexed area.

(3) Transfer the loader program to the on-chip RAM in order from the loader program load address (in sectors) of Physical Partition #1 of the eSD device to the loader program data size.
If a communication error occurs during data transfer, read access is retried up to three times per area. If the retry processing is repeated three times during the loader program transfer in a single area, another loader program is transferred from the next multiplexed area.

NOTE

Even when the loader size block is transferred from a multiplexed (reserved) area, the transfer of the loader program always begins from the area for loader program #0.



Note 1. Write the same data to loader program size block areas #0 to #6.

Note 2. Write the same data to loader program areas #0 to #6.

Note 3. The start sector location of the loader program is calculated as follows.
Loader program load address (in sectors) + retrieved size (in sectors) × n

Figure 4.4 Allocation of Loader Programs in the eSD Device Conforming to the eSD V2.1 Standard

(3) How to Distinguish between eSD V2.0 (Single Partition) and V2.1 (Multi Partitions)

This boot program first assumes that an eSD device supporting multi partitions is connected and begins loader program transfer from physical partition #1. If an error regarding a command for multi partitions shown in the following table occurs, the boot program assumes that an eSD device with a single partition is connected and transfers the loader program from physical partition #0.

Table 4.5 Errors Regarding Commands for Multi Partitions

No.	Issued Command	Command Function	Error	Remarks
1	CMD43 (SELECT_PARTITION)	Selects physical partition #1.	Multi partitions are not supported.	
2	CMD45 (QUERY_PARTITION)	Obtains the physical partition ID.	The specified partition does not exist.	

4.2.1.5 Note

The shared bus and 8-bit SD bus of the embedded SDIO cannot be used in this mode.

4.2.2 Boot Mode 1 (1.8-V / 3.3-V eMMC)

Table 4.6 shows the interface signals used for connection with the external device in boot mode 1 (eMMC). This LSI supports the eMMC that operates in the boot operation mode prescribed in JEDEC STANDARD JESD84 A44 (MMCA 4.4).

Table 4.6 External Interface Signals Used in Boot Mode 1

Interface Module	Pin Name	I/O	Function	Pin Type
SDHI/eMMC (SDHI0)	SD0_CLK	Output	eMMC clock	Dedicated pins (1.8 V/3.3 V)
	SD0_CMD	Input/output	eMMC command or response	
	SD0_DATA0	Input/output	eMMC data 0	
	SD0_DATA1	Input/output	eMMC data 1	
	SD0_DATA2	Input/output	eMMC data 2	
	SD0_DATA3	Input/output	eMMC data 3	
	SD0_DATA4	Input/output	eMMC data 4	
	SD0_DATA5	Input/output	eMMC data 5	
	SD0_DATA6	Input/output	eMMC data 6	
	SD0_DATA7	Input/output	eMMC data 7	
	SD0_CD*1	Input	SD card detection	Multiplexed pins
	SD0_WP*1	Input	SD write protection	
	SD0_RST#	Output	eMMC reset	Dedicated pins (1.8 V/3.3 V)

Note 1. This terminal is not used at boot time.

4.2.2.1 External Connections

Figure 4.5 shows the connections with the eMMC device. The boot program in this LSI does not monitor the state of the SD0_CD and SD0_WP pins.

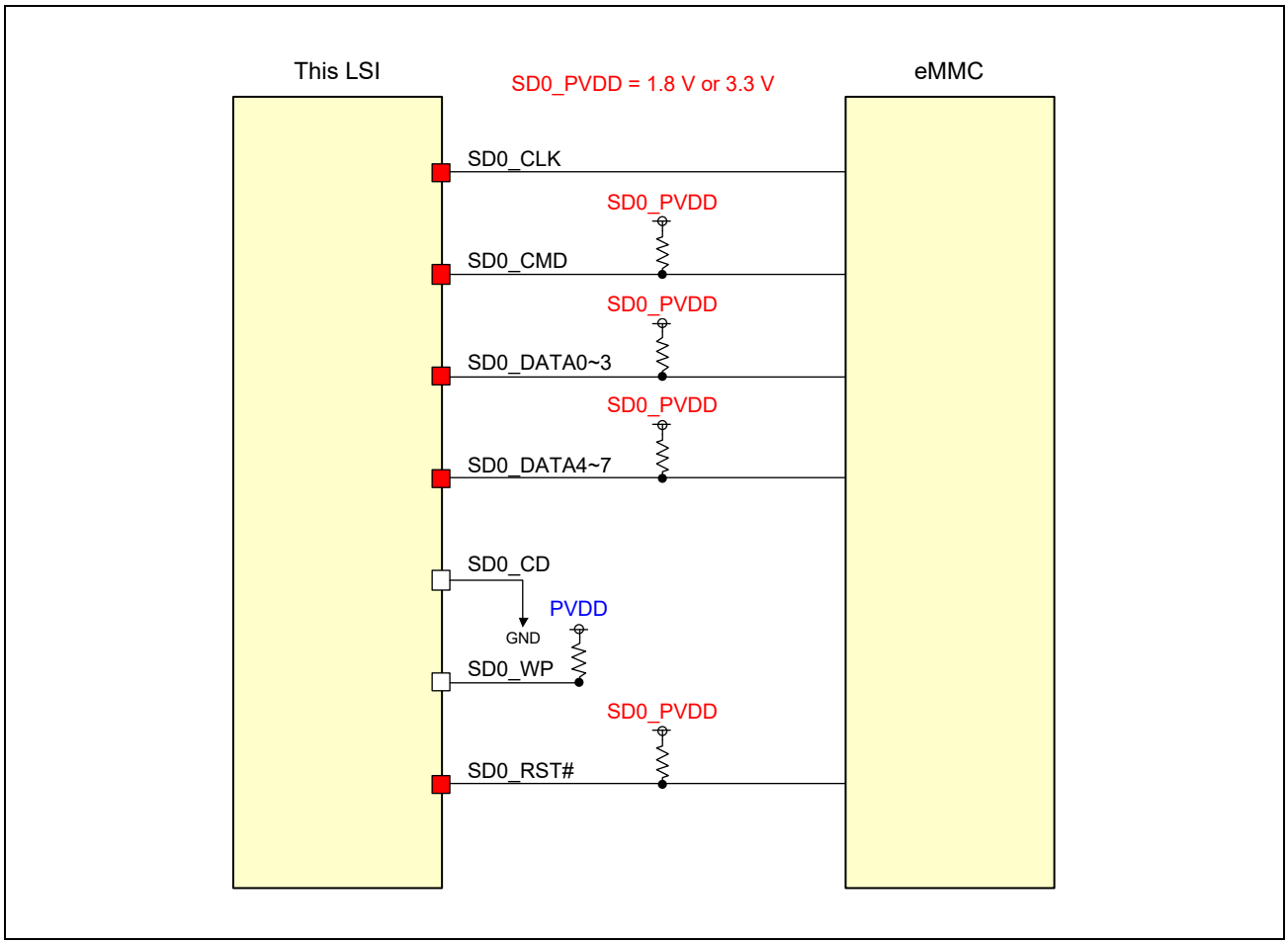


Figure 4.5 Connections in Boot Mode 1

4.2.2.2 Overview of Operation

This mode is used to boot this LSI from the user program stored in the eMMC device. The operating voltage is set to 1.8 V or 3.3 V and the width of the data bus connected to the eMMC device is fixed to eight bits in this mode. SDHI0 used as the interface controller in this mode is placed in the module standby mode at the startup of the LSI. Once SDHI0 is released from the module standby mode, the boot program in this LSI sets up the CPG so that the frequency of the external output clock becomes 16.6 MHz, initializes SDHI0, and then begins the booting process.

4.2.2.3 Operation of Booting from 1.8-V eMMC

In boot mode 1 (booting from 1.8-V eMMC), this LSI is booted from the loader program in the 1.8-V eMMC device connected to channel 0 of the SD host interface (SDHI) as follows.

The boot program executes the following processing to access the eMMC device.

1. Setting up the necessary peripheral modules (SDHI channel 0 and GPIO)
2. Set drive capability value
3. Starting the alternative boot operation mode of the eMMC device
   Read access begins from sector 0 of the partition selected by the [179] field (PARTITION_CONFIG) of the extended CSD (EXT_CSD) register in the eMMC device. For details, refer to **Section 4.2.2.5, Alternative Boot Operation** and **Section 4.2.2.5(1), Partitions**.
4. Data in sector 0 is a dummy read, the loader program size block is obtained from sector 1, and stored in the H'0_000A_1E00 of the on-chip RAM

Obtain the loader program size, loader program load address, and loader program destination address from the stored loader program size block.

After acquiring the loader program size block, the sector is read blankly from sector 2 to the loader program load address (sector conversion), and the loader program data is transferred from the loader program load address (sector conversion) to the loader program destination address of the built-in RAM for the data size.

If the boot program has failed to read the loader program, it enters fail-safe mode (SCIF downloading mode). If the fail-safe processing has failed, execution enters an infinite loop in the on-chip ROM and the boot processing is terminated.

Note that the data bus width is set to eight bits for booting from the eMMC device.

4.2.2.4 Allocation of the Loader Program in the eMMC Device

Figure 4.6 shows the allocation of the loader program in the eMMC device conforming to the MMCA 4.4 standard.

In the loader program size block, place the loader program size in the first 4 bytes, the loader program load address in 4 bytes at offset H'10, and the loader program destination address in 4 bytes at offset H'20.

Place the signature H'AA55 in the last two bytes. For more information about the loader program size block, refer to **Section 4.3, Loader Program Size Block**.

The configuration of the loader program size block is shown in **Figure 4.7**. Note that the loader program size, loader program load address, and loader program destination address are assumed to be stored in 4 bytes of little-endian.

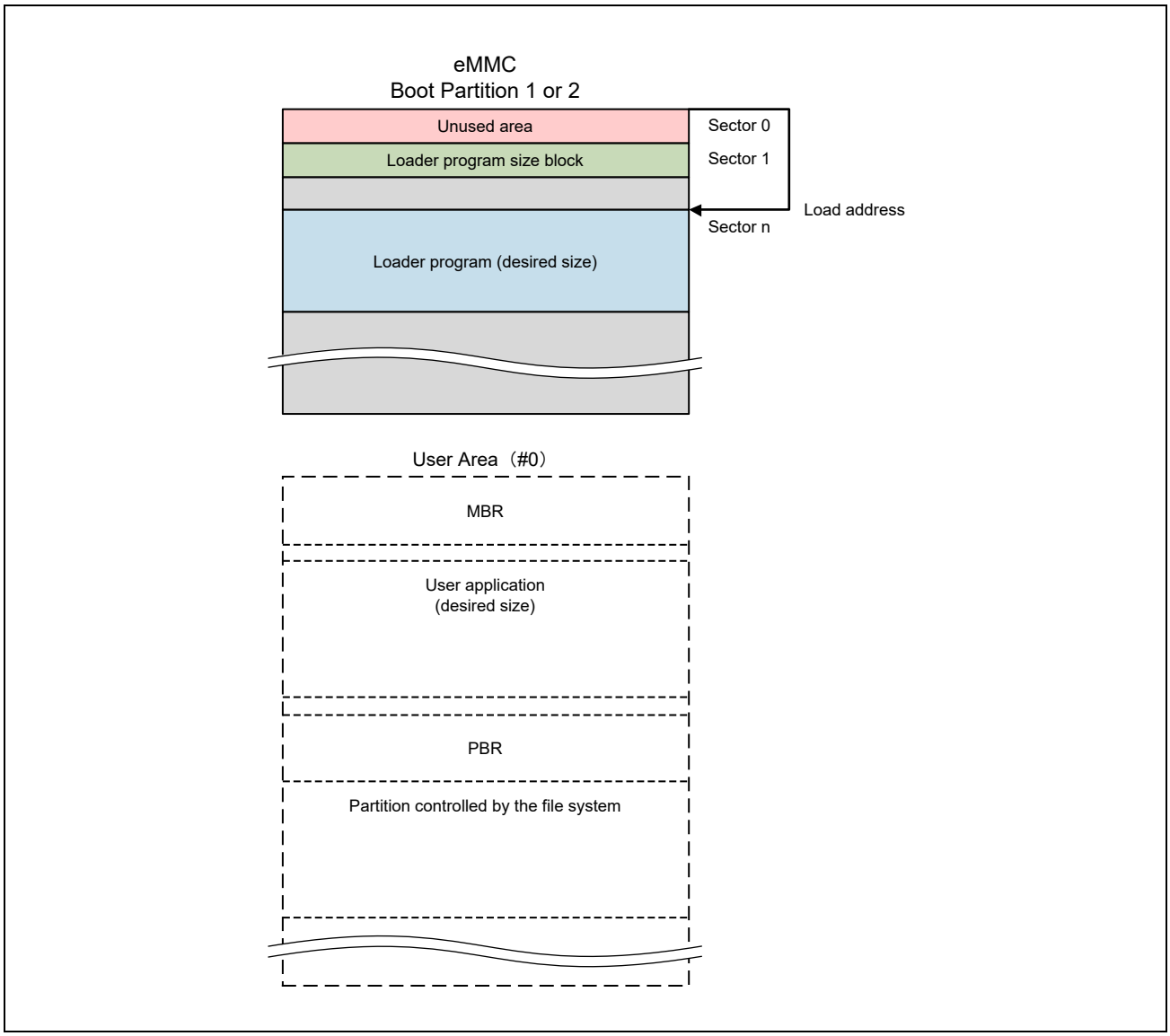


Figure 4.6 Allocation of Loader Program in the eMMC Device

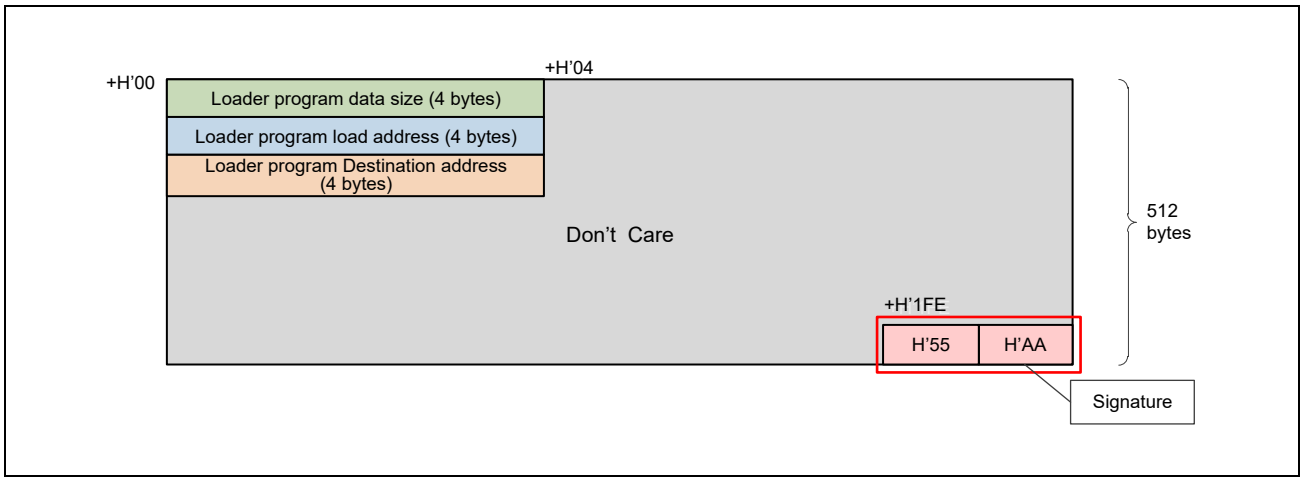


Figure 4.7 Structure of the Loader Program Size Block

NOTE

The loader program data size, loader program load address, and loader program destination address are expected to be stored in 4-byte little-endian.

In the eMMC device, the loader program is read from the partition specified by the BOOT_PARTITION_ENABLE bits in the PARTITION_CONFIG (ECSD[179]) field of the extended CSD register. Write the loader program to the partition selected by this register. **Table 4.7** shows the specifications of the extended CSD register.

Table 4.7 Specifications of the Extended CSD Register

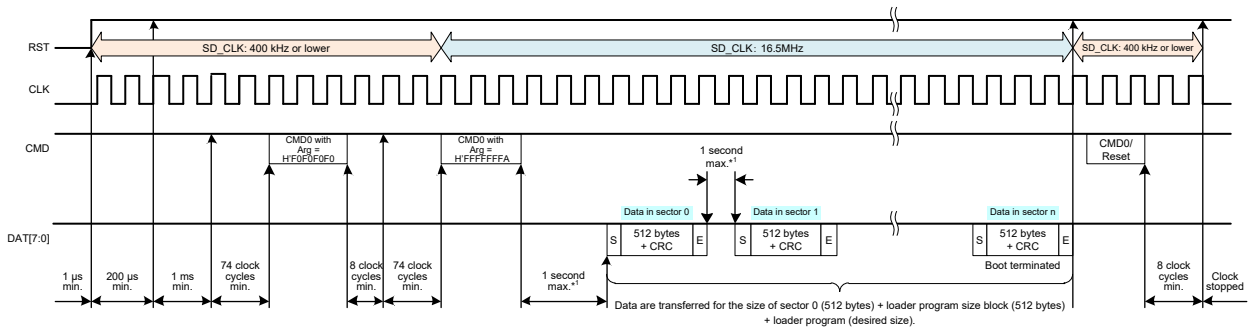
No.	Field Name	Bit Name	Setting	Remarks
1	PARTITION_CONFIG (ECSD[179])	BOOT_PARTITION_ENABLE bits [5:3]	H'0: Device boot is disabled (default) H'1: Boot partition 1 enabled for boot H'2: Boot partition 2 enabled for boot H'7: User area enabled for boot	

Allocate partitions as follows.

- Store the loader program in boot partition 1.
- Store the application program in boot partition 2.
- Allocate the user area to a partition controlled by the file system.

4.2.2.5 Alternative Boot Operation

Figure 4.8 shows the alternative boot operation to read the loader program. This boot program sets up the registers in SDHI channel 0 so that the SDHI operates with an 8-bit bus width when booting from the eMMC device. On the eMMC device side, the data bus width in the alternative boot operation is determined by the setting of the BOOT_BUS_WIDTH[177] field in the extended CSD register. To boot up from the eMMC device correctly, set up the extended CSD register in the eMMC device as shown in **Table 4.8** when writing a program to the eMMC device.



- This CPU accesses the eMMC device with a data bus width of 8 bits in the alternative boot operation. The data bus width of the eMMC device can be specified in the BOOT_BUS_WIDTH[177] field of the extended CSD register. When writing a loader program to the eMMC device, specify the 8-bit bus width in this field.
- The eMMC device sequentially outputs data from sector 0 of the partition selected in the PARTITION_CONFIG[179] field of the extended CSD register in the alternative boot operation.
- This CPU does not detect Boot Ack.
  The output of Boot Ack from the eMMC device can be specified in the PARTITION_CONFIG[179] field. Be sure to stop the output of Boot Ack by using this field.

Note 1. For a maximum of one second, the BRE, ERR3, and ERR1 bits of the SD_INFO2 register are polled to check whether data have been received, a timeout has occurred, and a CRC error has occurred, respectively.
<Normal operation>
SD_INFO2.BRE = 1
Data are transferred from SD_BUF to the on-chip RAM.
<Error>
CRC error: SD_INFO2.ERR1 = 1
The loader program is read from a reserved area.
Timeout error: SD_INFO2.ERR3 = 1
Execution of the boot program for eMMC is terminated.

Figure 4.8 Alternative Boot Operation to Read the Loader Program

Table 4.8 Setting of the Data Bus Width in Alternative Boot Operation

Field in Extended CSD Register of eMMC Device	Setting
BOOT_BUS_WIDTH [177]	Bits 1 and 0: BOOT_BUS_WIDTH = 10b (The bus width is set to 8 bits in alternative boot operation.)

(1) Partitions

An eMMC device has multiple partitions (**Figure 4.9**). The partition used for booting can be specified in the PARTITION_CONFIG[179] field of the extended CSD (EXT_CSD) register. To boot up correctly, specify the boot partition in the extended CSD (EXT_CSD) register of the eMMC device as shown in **Table 4.9** when writing a program to the eMMC device.

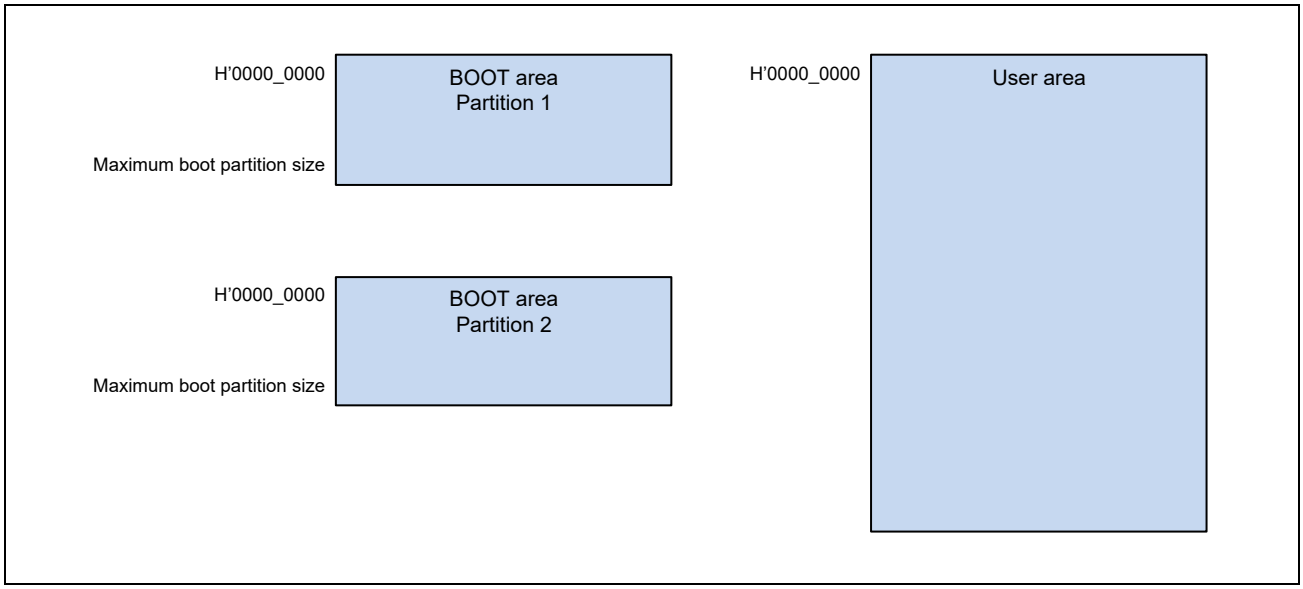


Figure 4.9 Partitions in the eMMC Device

Table 4.9 Boot Partition Setting in the Alternative Boot Operation

Field in Extended CSD Register of eMMC Device	Setting
PARTITION_CONFIG [179]	• Set bit 6 (BOOT_ACK) to 0b (the eMMC device does not output a boot acknowledge in the alternative boot operation). • Specify the boot partition in bits 5 to 3 (BOOT_PARTITION_ENABLE). 000b: Booting is disabled. 001b: Booting from sector 0 in boot partition 1 in the boot operation. 010b: Booting from sector 0 in boot partition 2 in the boot operation. 011b to 110b: Reserved 111b: Booting from sector 0 in the user area in the boot operation.

4.2.2.6 Operation of Booting from 3.3-V eMMC

The eMMC boot (3.3 V) differs from the eMMC boot (1.8 V) only in the voltage it supplies.

4.2.2.7 Notes

- This mode does not support booting from an MMC card.
- The width of the data bus is fixed to eight bits; 1-bit or 4-bits data bus cannot be used in this mode.

4.2.3 Boot Mode 2 (Serial Flash Memory (Single / Quad / Octal))

Table 4.10 shows the interface signals used for connection with the external device in boot mode 2 (1.8-V serial flash memory).

Table 4.10 External Interface Signals Used in Boot Mode 2

Interface Module	Pin Name	I/O	Function	Pin Type
xSPI	XSPI_SPCLK	Input/output	XSPI clock	Dedicated pins
	XSPI_IO0	Input/output	XSPI data 0	(1.8 V/3.3 V)
	XSPI_IO1	Input/output	XSPI data 1	
	XSPI_IO2	Input/output	XSPI data 2	
	XSPI_IO3	Input/output	XSPI data 3	
	XSPI_IO4	Input/output	XSPI data 4	
	XSPI_IO5	Input/output	XSPI data 5	
	XSPI_IO6	Input/output	XSPI data 6	
	XSPI_IO7	Input/output	XSPI data 7	
	XSPI_DS	Input/output	XSPI data strobe	
	XSPI_CS0	Input/output	XSPI CS0	
	XSPI_CS1	Input/output	XSPI CS1	
	XSPI_RESET#	Output	XSPI reset	
	XSPI_WP#	Output	XSPI write protect	

4.2.3.1 External Connections

Figure 4.10, Figure 4.11, and **Figure 4.12** show the connections with the Single SPI memory and Quad SPI memory, respectively.

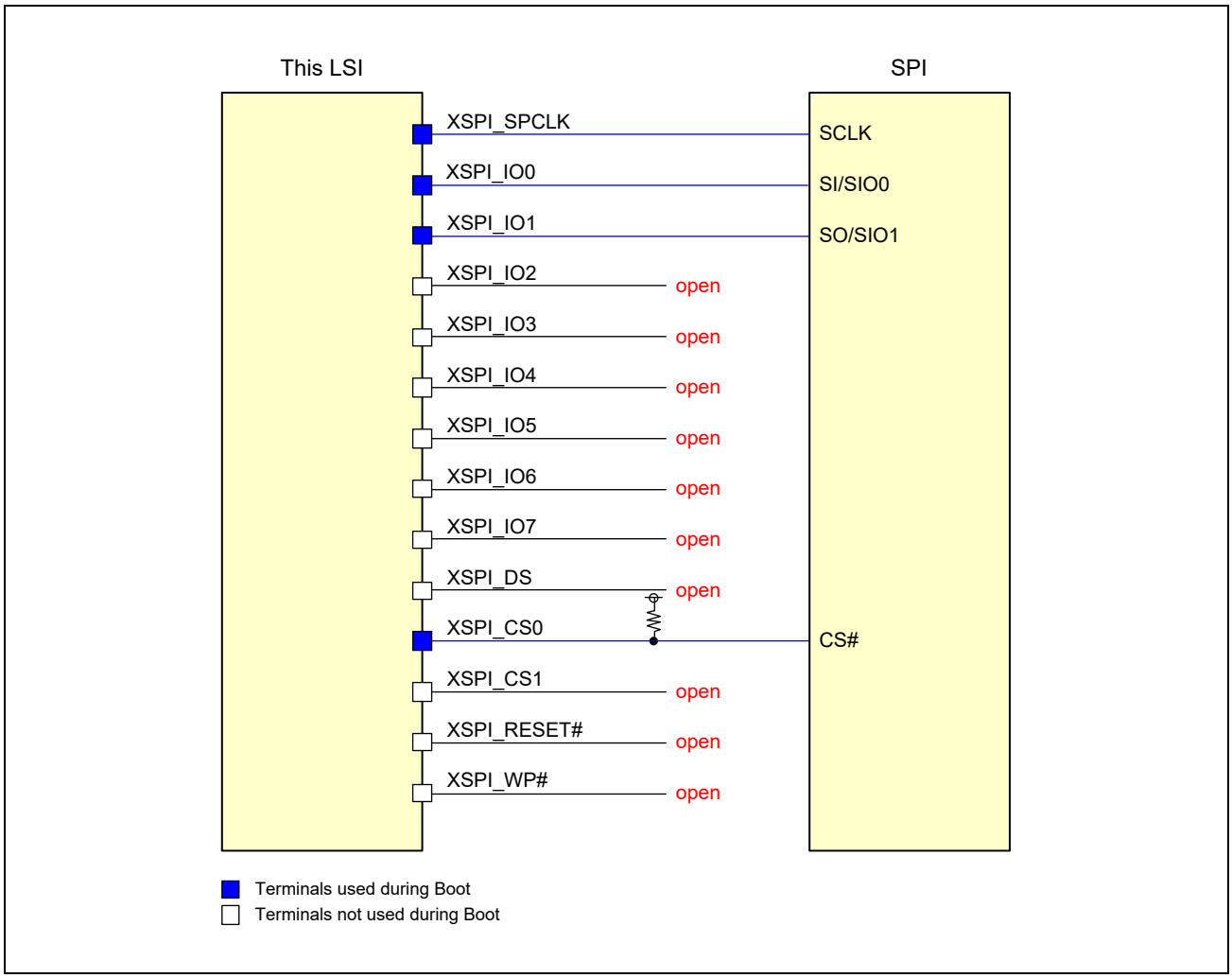


Figure 4.10 Connections with Single SPI Memory in Boot Mode 2

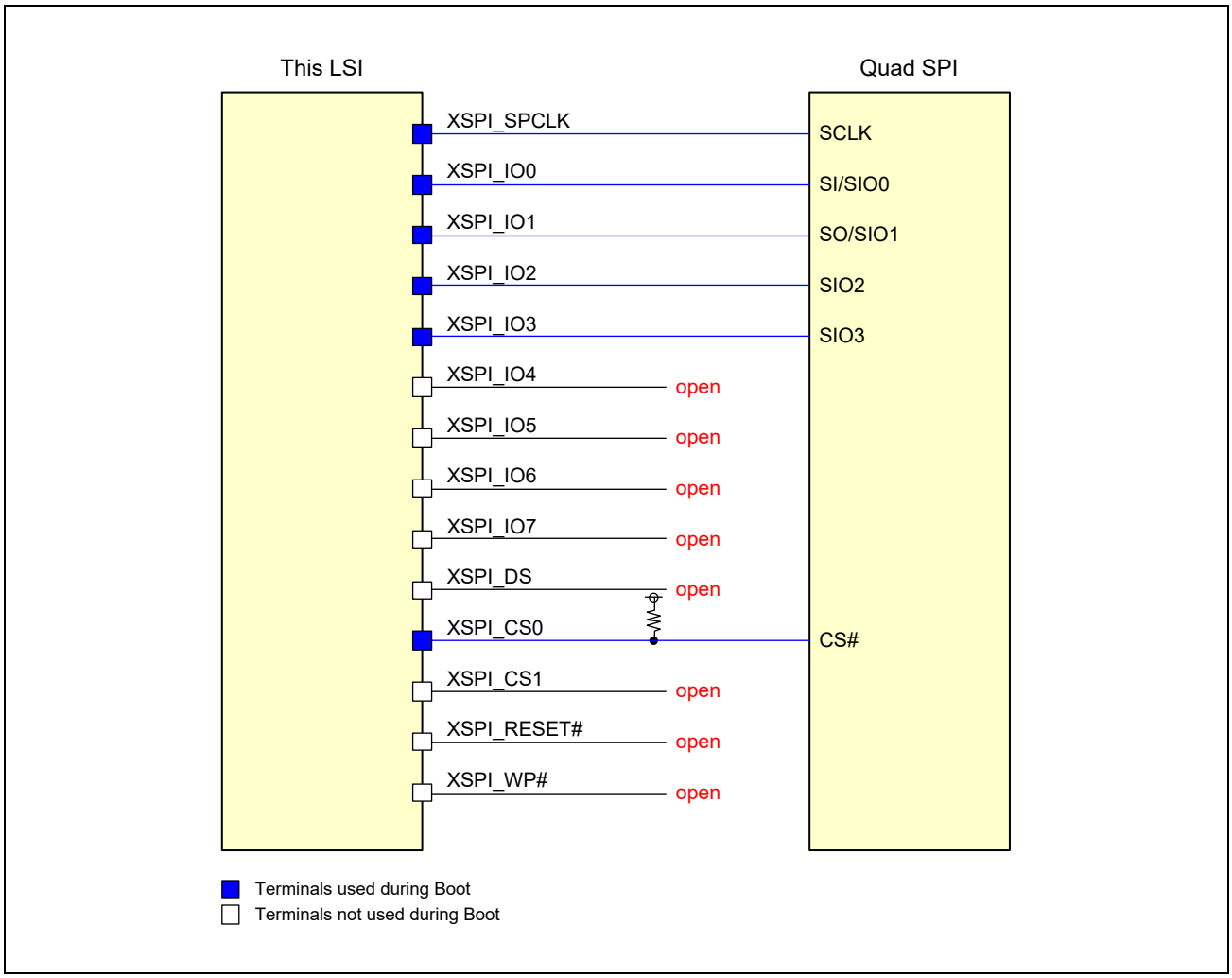


Figure 4.11 Connections with Quad SPI Memory in Boot Mode 2

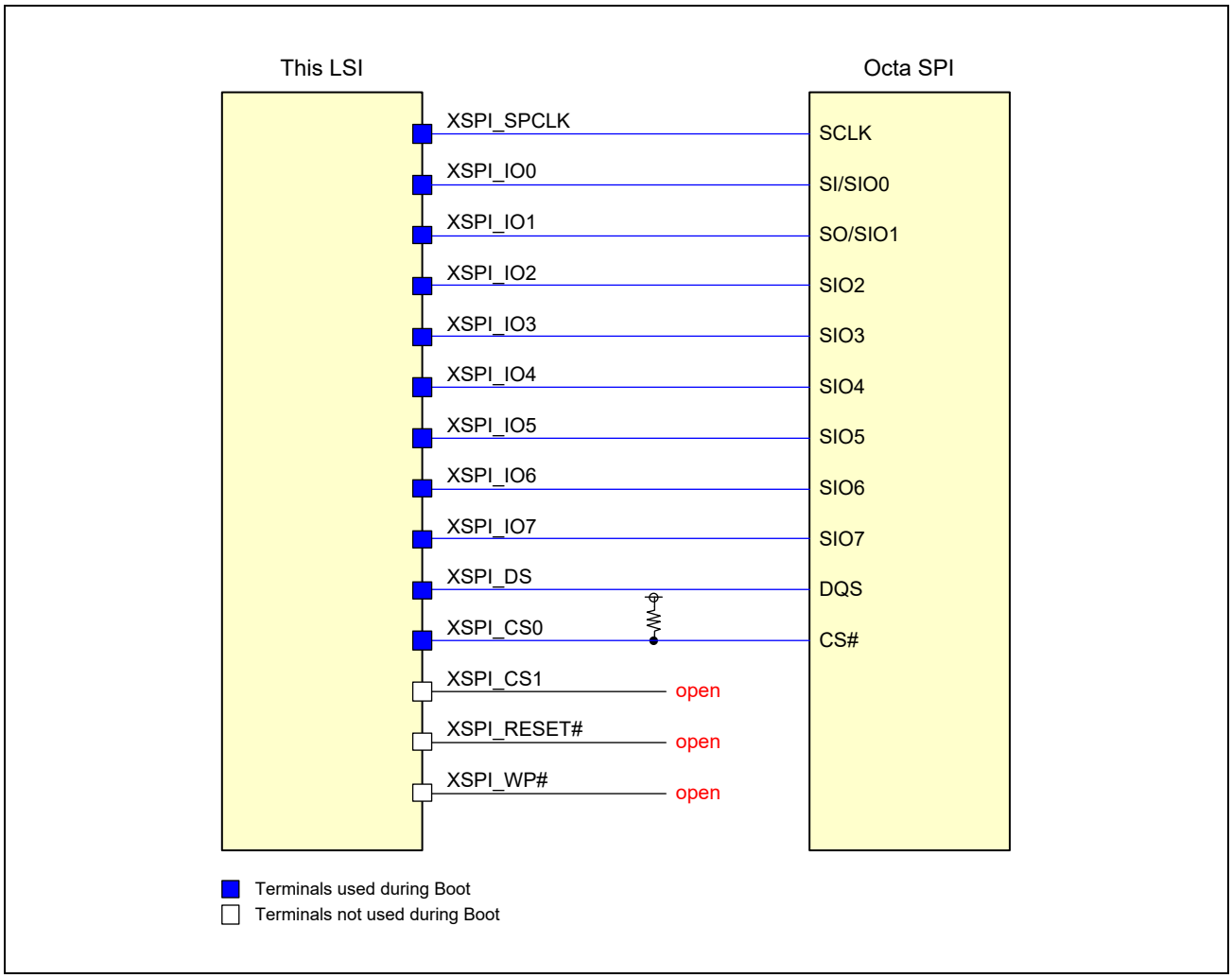


Figure 4.12 Connections Octa Memory in Boot Mode 2

4.2.3.2 Overview of Operation

This mode is used to boot this LSI from the user program stored in the Single or Quad flash memory device.

The serial flash memory mounted in advance on the board is used in this mode. In this mode, the MD_BOOT2 pin is controlled by the voltage (1.8 V/3.3 V) of the target device. The clock frequency at the startup is set to 16.6 MHz. The LSI is booted up through an SPI handshake with the Single, Quad, or OCTA flash memory device.

Upon detecting boot mode 2 according to the value read from the SYSC, the boot program makes the necessary settings of the xSPI bus controller to access memory. The boot program handles the following steps to control the booting process.

1. The CPG is set up to supply clocks, and the xSPI bus controller receives the operating clock and is released from the reset state.
2. The necessary information to control the I/O buffers is read from the registers in the SYSC.
   The given information is copied from the OTPC to the SYSC registers in advance.
3. The properties such as the driving ability of the I/O buffers of the signals to be used are set up.
4. After the above setup, the xSPI bus controller performs a handshake process with the target memory device.

4.2.3.3 Operation of Booting from 1.8-V Serial Flash Memory

xSPI (1.8 V) boot boots from the loader program in serial flash memory (1.8 V) connected to xSPI channel 0.

xSPI (1.8 V) boot does the following to access the external address space:

1. Releasing the xSPI from module standby mode
2. Setting up the registers in the necessary peripheral modules (xSPI channel0 and GPIO)
3. Set the drive capability value during 1.8 V operation
4. Retrieves the loader program size block stored at the first address of the external address space (H'0_2000_0000*[1]) and stores it in the H'0_000A_1E00*[2] of the on-chip RAM

Obtain the loader program size, loader program load address, and loader program destination address from the stored loader program size block.
The loader program stored at the first address of the external address space (H'0_2000_0000*[1]) + loader program load address of the loader program is transferred to the loader program destination address of the on-chip RAM for the size of the loader program.

After that, the loader program stored in the serial flash device is executed by branching to the first address of the loader program transferred to the on-chip RAM (loader program destination address).

If the boot program has failed to read data, the boot program enters fail-safe mode (SCIF downloading mode). If the fail-safe processing has failed, execution enters an infinite loop in the on-chip ROM and the boot processing is terminated.

Note 1. Cortex-A55 (H'0_2000_0000), Cortex-M33 (H'0_8000_0000)
Note 2. Cortex-A55 (H'0_000A_1E00), Cortex-M33 (H'0_2002_1E00)

4.2.3.4 Allocation of the Loader Program in the Serial Flash Memory

Figure 4.13 shows the allocation of the loader program size block and loader program in the serial flash memory.

Place the loader program size in the first 4 bytes, the load address in the next 4 bytes, and the destination address in the next 4 bytes. The termination 4 bytes are filled with the signature H'AA55. For more information about the loader program size block, Refer to **Section 4.3, Loader Program Size Block**.

The loader program data structure is shown in **Figure 4.14**. Note that the loader program size, loader program load address, and loader program destination address are assumed to be stored in 4 bytes of little-endian.

Transfer the loader program starting from the H'0_2000_0000*1 + loader program load address to the loader program destination address in the on-chip RAM for the loader program size.

Note 1. Cortex-A55 (H'0_2000_0000), Cortex-M33 (H'0_8000_0000)

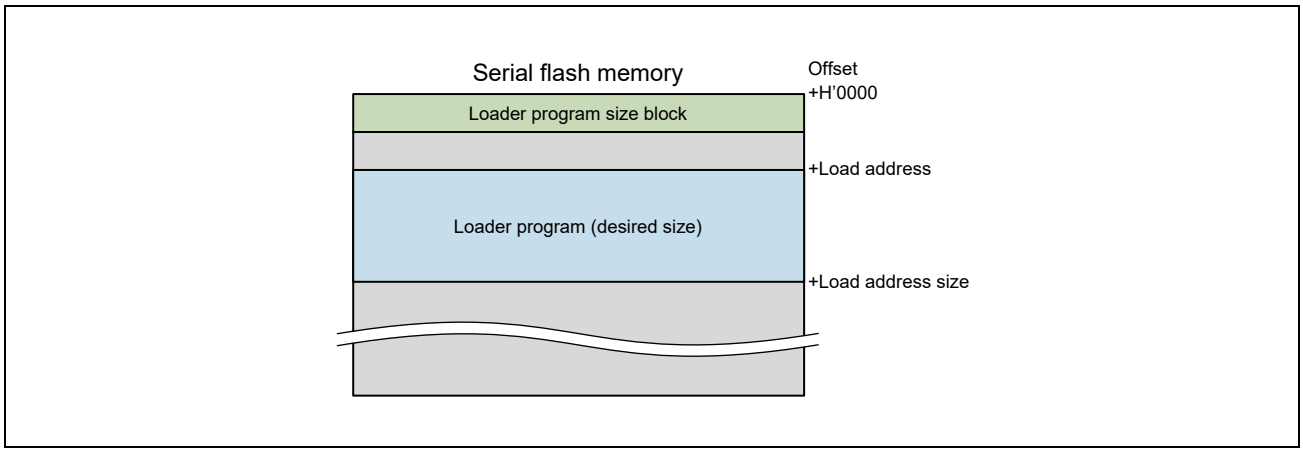


Figure 4.13 Allocation of Loader Program in the Serial Flash Memory Device

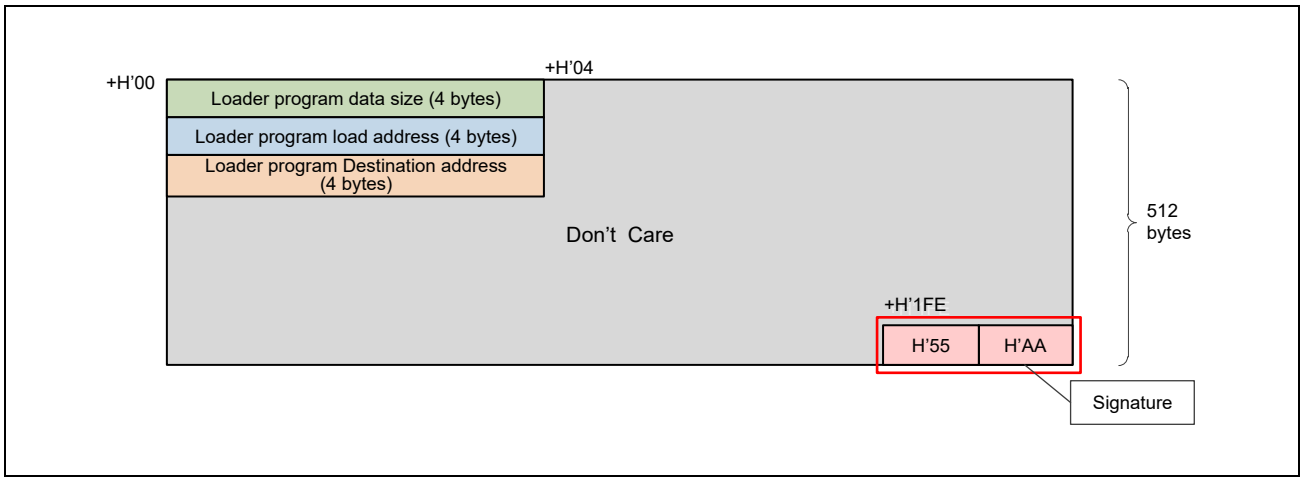


Figure 4.14 Structure of the Loader Program Size Block

NOTE

The loader program data size, loader program load address, and loader program destination address are expected to be stored in 4-byte little-endian.

4.2.3.5 Operation of Booting from 3.3-V Serial Flash Memory

The xSPI boot (3.3 V) differs from the xSPI boot (1.8 V) only in the voltage it supplies.

4.2.3.6 Notes

- It supports 1.8 V/3.3 V SPI, 1.8 V/3.3 V QSPI Flash memory, and 1.8 V OctaFlash (3.3 V OctaFlash is not supported).
- xSPI supports two slaves, but Boot mode is for devices connected to CS0.
- xSPI supports On-The-Fly Encode/Decode, but this function is turned off during Boot.

4.2.4 Boot Mode 3 (SCIF Downloading)

Table 4.11 shows the interface signals used for connection with the external device in boot mode 3 (SCIF downloading).

Table 4.11 External Interface Signals Used in Boot Mode 3

Interface Module	Pin Name	I/O	Function	Pin Type
SCIF0	SCIF0_SCK*1	Input/output	SCIF0 serial clock	Multiplexed pins (3.3 V)
	SCIF0_RXD	Input	SCIF0 receive data	
	SCIF0_TXD	Output	SCIF0 transmit data	
	SCIF0_CTS#*1	Input/output	SCIF0 transmission enable	
	SCIF0_RTS#*1	Output	SCIF0 transmission request	

Note 1. This terminal is not used at boot time.

4.2.4.1 External Connections

Figure 4.15 shows the external connections for downloading through the SCIF.

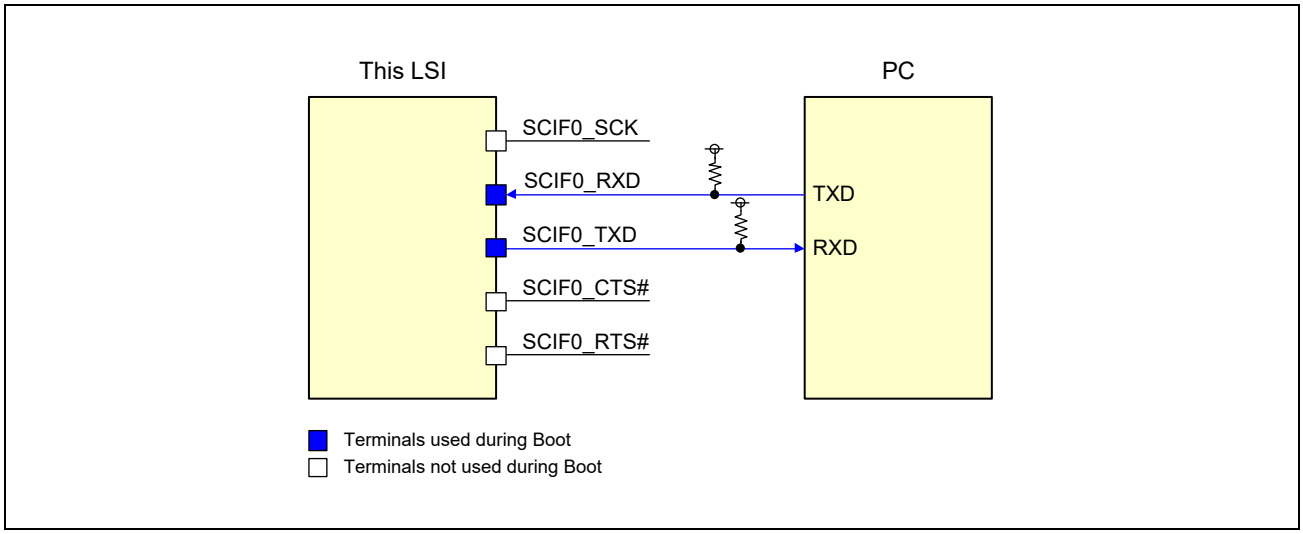


Figure 4.15 Connections for Downloading through the SCIF in Boot Mode 3

4.2.4.2 Overview of Operation

This mode is used to download the user program from the host PC. Upon detecting boot mode 3 according to the value read from the SYSC, the boot program makes the necessary settings of the SCIF0 module to access memory. The boot program handles the following steps to control the booting process.

1. The SCIF0 module is set up to handle communications as shown below. Asynchronous communications proceed in this mode. The SCIF0_RXD and SCIF0_TXD pins are used.
   Baud rate: 115200 bps
   Data length: 8 bits
   Stop bit: 1 bit
   Flow control: None
   Data format: Motorola S-record
2. The sequence shown in **Figure 4.16** is used to download the user program through the SCIF.

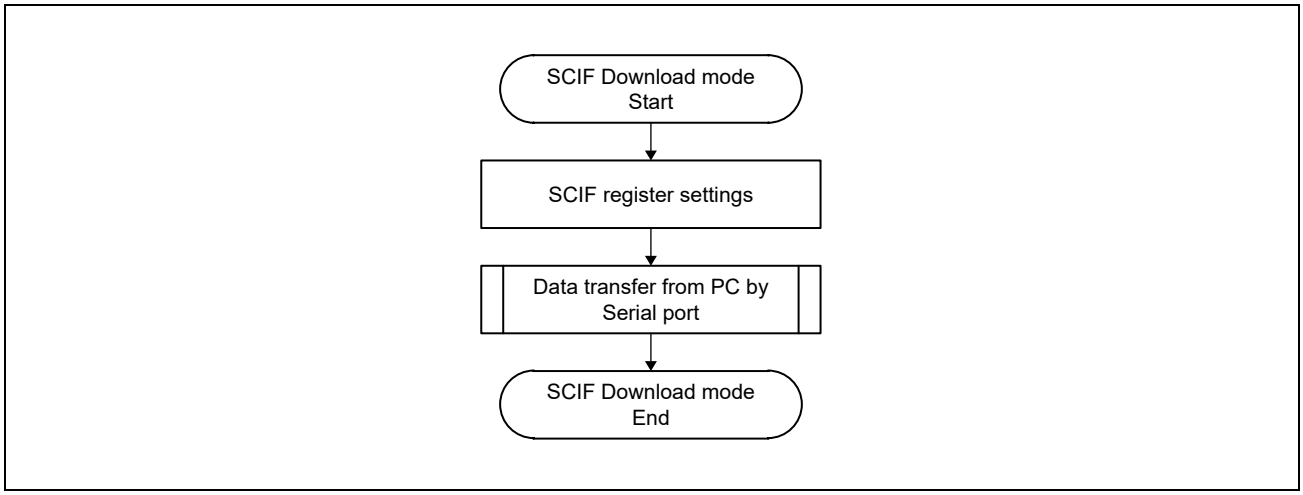


Figure 4.16 Sequence of Downloading through the SCIF0

4.2.4.3 Operation of Booting through SCIF Downloading

SCIF download boot is launched from a loader program stored on an external PC via the FIFO built-in serial communication interface (SCIFA).

The boot program executes the following processing.

1. Set peripheral modules (SCIFA channel 0, GPIO) and set drive capability values.

   When the LSI becomes ready for data reception, the boot program outputs the following message to the SCIFA.
   "SCI Download mode (Normal SCI boot)"

   If execution enters boot mode 3 due to the fail-safe processing, the boot program outputs the following message to the SCIFA.
   "SCI Download mode (due to parameter error)"

2. When the data of the Motorola S0 record sent from an external PC via SCIFA is received, Output the following message to SCIFA.
   "-- Load Program to System RAM ----------"

   Note: The S0 record must always be generated because the boot program requires an S0 record as a trigger for the reception processing.

3. When receiving an S3 record, it is converted to binary and transferred to the on chip RAM.

   The first 512 bytes of records are treated as loader program size blocks and verified for signatures. If the value is H'55AA, it is judged that the correct data is received, and the load program size and destination address are obtained.

   For subsequent S3 records, expand to on-chip RAM only if the addressing is within the destination address and load program size.

4. When an S7 record is received, the following message is output to SCIFA to complete the deployment to the on-chip RAM, and then branch to the destination address of the on-chip RAM.
   "-- Start Boot Program on SRAM ---------"

   If the data in the Motorola S records sent from the external host PC satisfies any of the following error conditions, the corresponding error message is output to the SCIFA and copying to the on-chip RAM is aborted.

   — An S1 or S2 record is received: "Invalid Record Type Error!!!"

   — An address outside the on-chip RAM area is specified: "Address Error!!!"

   — An illegal character code is found: "Invalid Character Error!!!"

   — An illegal byte count is found: "Invalid Byte Count Error!!!"

The data transmission and reception operation in SCIF communications (asynchronous) is shown in **Figure 4.17**.

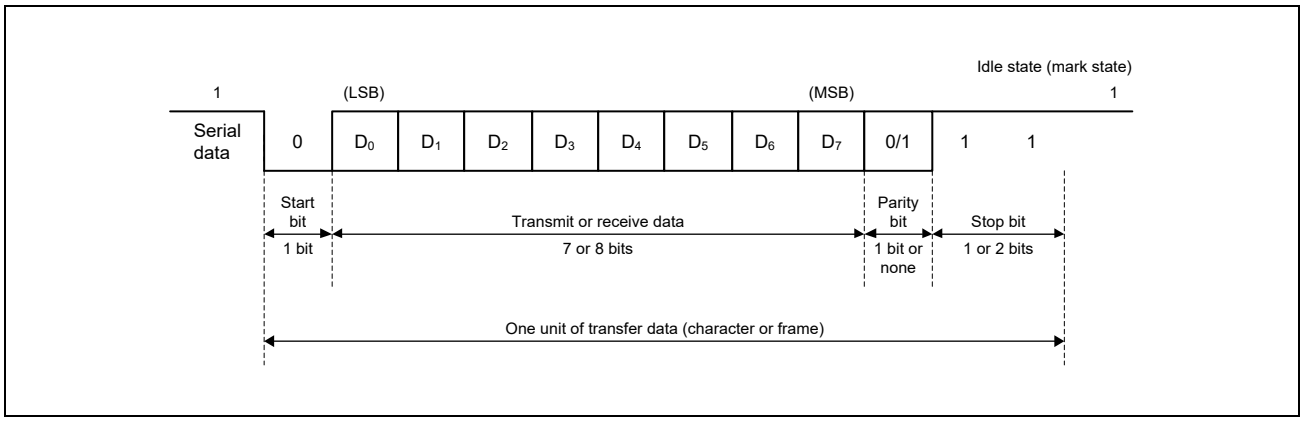


Figure 4.17 Data Transmission and Reception Operation in SCIF Communications

4.2.4.4 Allocation of the Loader Program for SCIF Downloading

Figure 4.18 shows the allocation of the loader program copied to the on-chip RAM.

In the loader program size block (512 bytes), place the loader program size in the first 4 bytes, the load address in 4 bytes at offset H'10, and the destination address in 4 bytes at offset H'20. Place the signature H'AA55 in the last two bytes. Note that the loader program size is assumed to be stored in 4 bytes little endian.

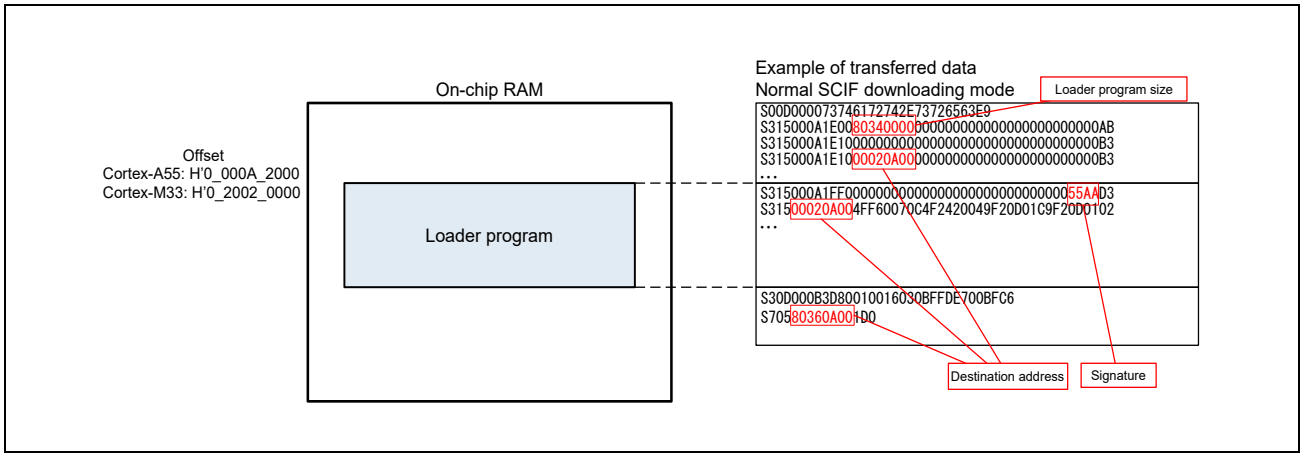


Figure 4.18 Allocation of the Loader Program for SCIF Downloading Copied to the on-chip RAM

4.2.4.5 Note

The pins assigned to the SCIF0_SCK, SCIF0_CTS, and SCIF0_RTS signals, which are not used in this mode, must not be changed from the initial settings (GPIO pins). If such a pin is set to operate as an SCIF0 pin, it becomes an input pin or an input/output pin. In this case, when the pin is externally placed in the Hi-Z state, this LSI cannot communicate with the host PC correctly.

To use the SCIF0 for normal serial communications after booting, the SCIF0_SCK, SCIF0_CTS, and SCIF0_RTS pins should be pulled up or down in accordance with the communication mode and the driver software specifications.

4.3 Loader Program Size Block

The loader program size block defines information about where the loader program is stored and deployed.

It has a fixed size of 512bytes and is stored in a fixed location for each boot device.

The configuration of the loader program size block is shown in **Figure 4.19**, and the details of each field are shown in **Table 4.12**.

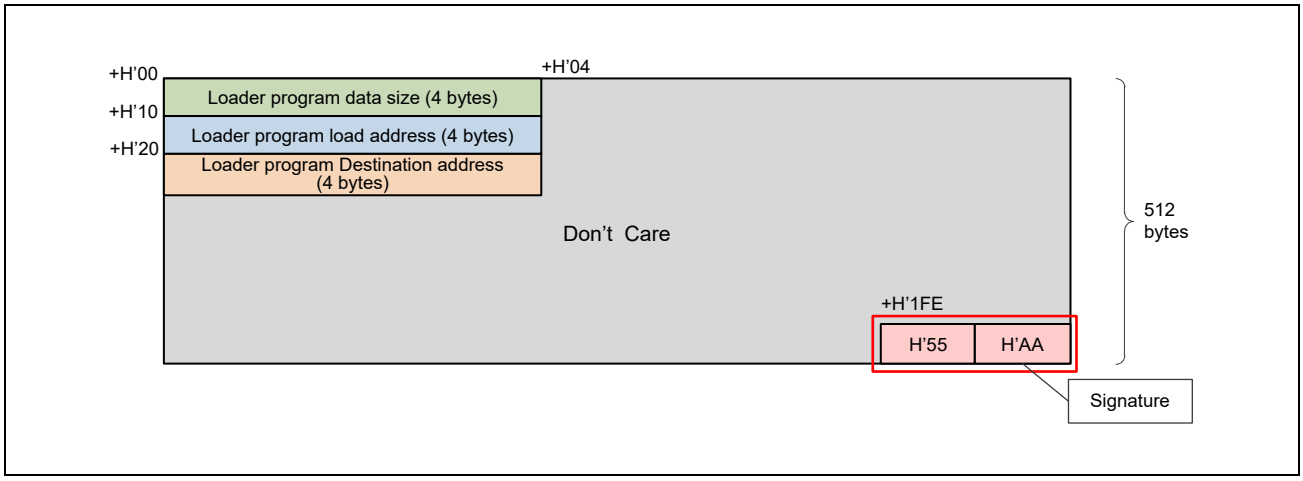


Figure 4.19 Configuring the Loader Program Size Block

Table 4.12 Loader Program Size Block Field List

No.	Field Name	Offset	Size	Description
1	Loader program data size	H'0000	4 bytes*1	Loader program size (byte units)
2	Loader program load address	H'0004	4 bytes*1	Loader program offset value (byte units)
3	Loader program destination Address	H'0008	4 bytes*1	The starting address of the loader program placement destination (byte units)
4	Signature	H'01FE	2 bytes*2	Signature indicating that the loader program supports this LSI. Place the H'AA55.

Note 1. It is arranged in little-endian units of 4 bytes.

Note 2. It is arranged in little-endian units of 2 bytes.

The boot program reads the loader program size block from a fixed position for each boot device into the loader size block area (Cortex-A55: H'000A_1E00, Cortex-M33: H'2002_1E00) of the on-chip RAM.

Check the signature (H'01FE) after reading the loader program size block (512 bytes).

If the signature value is H'AA55, the correct loader is determined to be stored, and the loader program size, load address, and destination address are obtained from the loader size block area.

Check the destination address and destination address + loader size, and if it is within the loader program expandable area, transfer the loader program from the load address to the destination address. The destination address and loader program size must be set so that the loader program fits within the loader load area (Cortex-A55: H'0_000A_2000 – H'0_0011_FFFF, Cortex-M33: H'0_2002_2000 – H'0_2009_FFFF).

Figure 4.20 shows an image of a load program deployment.

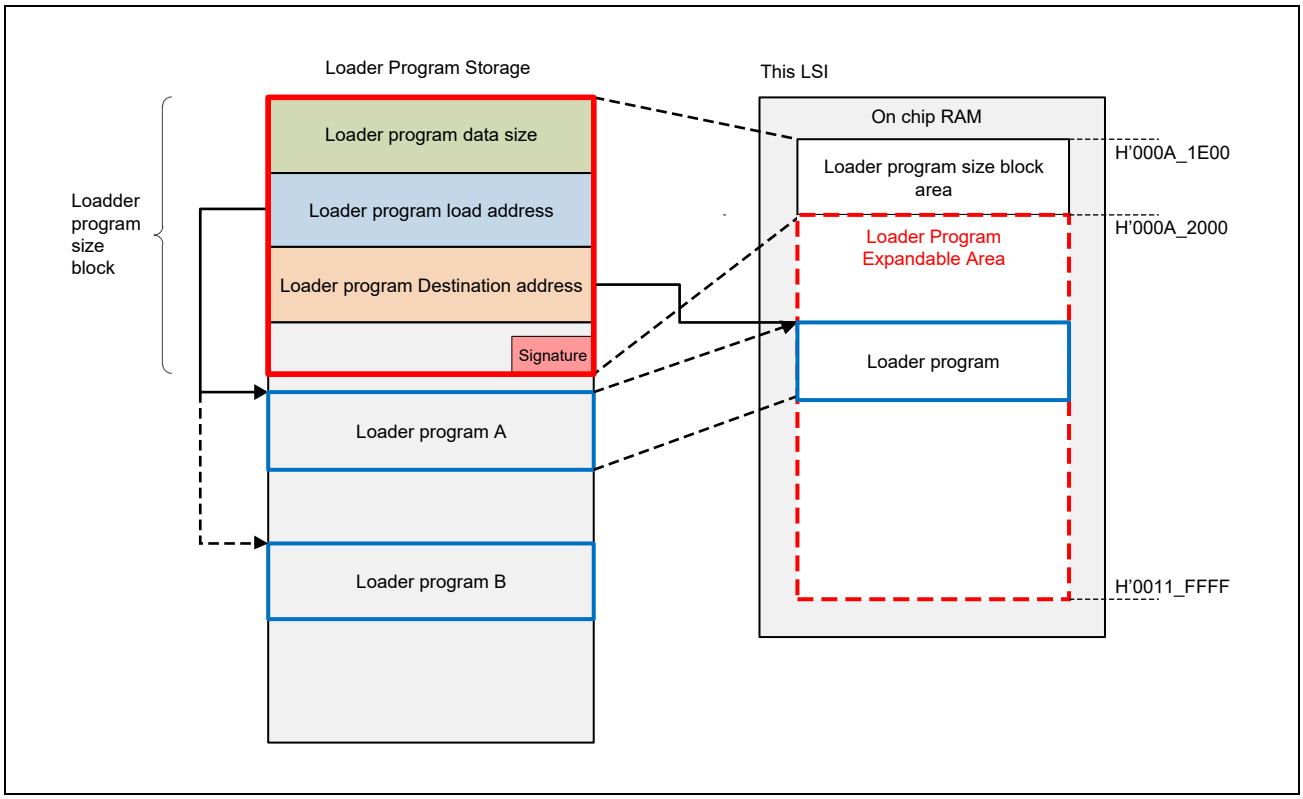


Figure 4.20 Loader Program Deployment Image

NOTE

For SCIF downloads, S3-Record addressing stores only records within the range of destination address and destination address + loader size, and discards other records. It does not consider the load address. For xSPI boot, the maximum load address + loader size is H'1000_0000.

For eSD, eMMC boot, the load address should be in sectors (512 bytes). Also, the maximum value of the destination address is H'FFFF_FFFF.

The address shown in the figure is the Cortex-A55 address as an example.

5. LSI Internal Bus

5.1 Overview

5.1.1 Features

The bus system of this LSI provides a physical address space of 16 Gbytes (address bus width of 34 bits). The LSI internal bus of this LSI incorporates Arm CoreLink NIC-400, etc., and controls the following bus functions.

Security control:

Security attribute re-setting, Security level determination

Address translation:

34-bit address space access

Interrupt generation:

AXI bus error interrupt generation

Unit state detection:

Slave unit stop state detection

Address map switching function:

XSPI area and OCTA area switching function

Encryption/decryption function:

Encryption/decryption function for external memory device (Flash, DDR)

NOTE

The security control function is valid only for secure products.
For information on secure products, please contact our sales.

5.1.2 Block Diagram of LSI Internal Bus

The LSI internal bus of this LSI consists of the ACPU bus, MCPU bus, and system bus. **Figure 5.1** shows the configuration of the buses.

ACPU bus:

A bus connected to Cortex-A55, Cortex-M33_FPU, DDR memory controllers and Storage and Network

MCPU bus:

A bus connected to Cortex-M33 and serial interface units

System bus:

A bus connected to the control registers of each unit

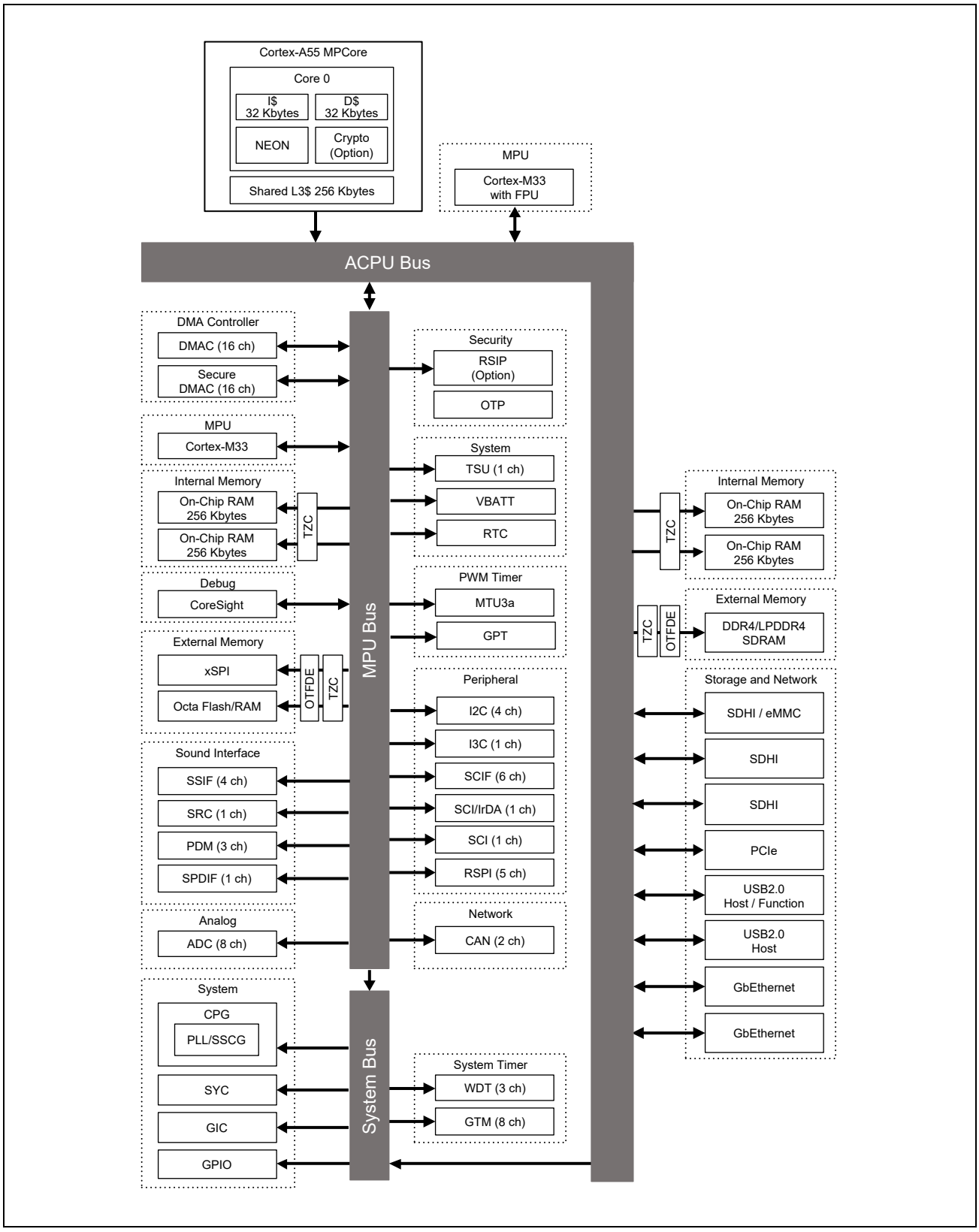


Figure 5.1 Configuration of LSI Internal Bus

5.2 Area Maps

5.2.1 Overall Address Space

Figure 5.2 shows the overall address space of this LSI and **Table 5.1** shows the detailed address space.

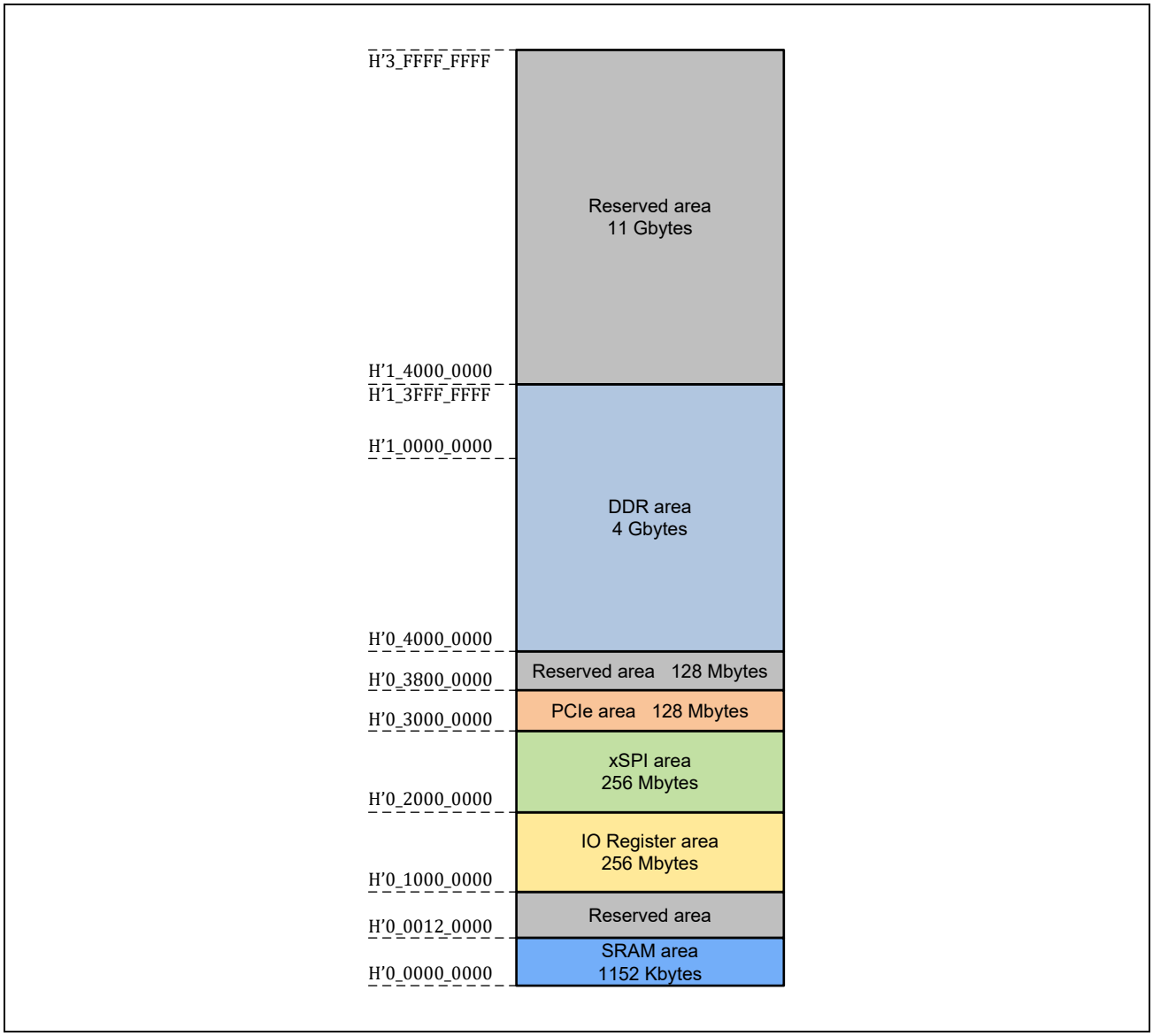


Figure 5.2 Overall Address Space

Some bus master units of this LSI require 34-bit address space access to access an address space of greater than 4 Gbytes in the whole 16-Gbyte address space of this LSI. For details, see **Section 5.4.2.1, 34-Bit Address Space Access**.

The Cortex-M33 address space for this LSI differs from the overall address space. It is the same as the address space of the default memory map of Cortex-M33. For details, see **Section 5.2.2, Cortex-M33/Cortex-M33_FPU Address Space**.

Table 5.1 Detailed Address Space (1/4)

Start Address	End Address	Size	Space	Remarks
H'1_4000_0000	H'3_FFFF_FFFF	11 Gbytes	Reserved	*1
H'0_4000_0000	H'1_3FFF_FFFF	4 Gbytes	DDR (Memory)	
H'0_3800_0000	H'0_3FFF_FFFF	128 Mbytes	Reserved	*1
H'0_3000_0000	H'0_37FF_FFFF	128 Mbytes	PCIe	
H'0_2000_0000	H'0_2FFF_FFFF	256 Mbytes	xSPI / Octa Flash / Octa RAM (Memory)	*2-1
H'0_1330_0000	H'0_1FFF_FFFF	205 Mbytes	Reserved	*1
H'0_12C0_0000	H'0_132F_FFFF	7 Mbytes	Reserved	*1
H'0_12A0_0000	H'0_12BF_FFFF	2 Mbyte	Reserved	*1
H'0_1280_3000	H'0_129F_FFFF	2036 Kbytes	Reserved	*1
H'0_1280_2C00	H'0_1280_2FFF	1 Kbyte	GTM ch7	
H'0_1280_2800	H'0_1280_2BFF	1 Kbyte	GTM ch6	
H'0_1280_2400	H'0_1280_27FF	1 Kbyte	GTM ch5	
H'0_1280_2000	H'0_1280_23FF	1 Kbyte	GTM ch4	
H'0_1280_1C00	H'0_1280_1FFF	1 Kbyte	GTM ch3	
H'0_1280_1800	H'0_1280_1BFF	1 Kbyte	GTM ch2	
H'0_1280_1400	H'0_1280_17FF	1 Kbyte	GTM ch1	
H'0_1280_1000	H'0_1280_13FF	1 Kbyte	GTM ch0	
H'0_1280_0C00	H'0_1280_0FFF	1 Kbyte	Reserved	*1
H'0_1280_0800	H'0_1280_0BFF	1 Kbyte	WDT CA55 core 0	
H'0_1280_0400	H'0_1280_07FF	1 Kbyte	WDT CM33	
H'0_1280_0000	H'0_1280_03FF	1 Kbyte	WDT CM33_FPU	
H'0_1270_0000	H'0_127F_FFFF	1 Mbyte	Reserved	*1
H'0_1250_0000	H'0_126F_FFFF	2 Mbytes	Reserved	*1
H'0_1240_0000	H'0_124F_FFFF	1 Mbytes	GIC	
H'0_11F0_0000	H'0_123F_FFFF	5 Mbyte	Reserved	*1
H'0_11E5_0000	H'0_11EF_FFFF	704 Kbytes	Reserved	*1
H'0_11E4_0000	H'0_11E4_FFFF	64 Kbytes	PCIe	
H'0_11E3_0000	H'0_11E3_FFFF	64 Kbytes	USB ch1 Host	
H'0_11E2_0000	H'0_11E2_FFFF	64 Kbytes	USB ch0 (OTG-Func)	
H'0_11E1_0000	H'0_11E1_FFFF	64 Kbytes	USB ch0 (OTG-Host)	
H'0_11E0_0000	H'0_11E0_FFFF	64 Kbytes	USBPHY Control	
H'0_11D0_0000	H'0_11DF_FFFF	1 Mbyte	Reserved	*1
H'0_11C5_0000	H'0_11CF_FFFF	704 Kbytes	Reserved	*1
H'0_11C4_0000	H'0_11C4_FFFF	64 Kbytes	Ether ch1	
H'0_11C3_0000	H'0_11C3_FFFF	64 Kbytes	Ether ch0	
H'0_11C2_0000	H'0_11C2_FFFF	64 Kbytes	SD ch2	
H'0_11C1_0000	H'0_11C1_FFFF	64 Kbytes	SD ch1	
H'0_11C0_0000	H'0_11C0_FFFF	64 Kbytes	SD ch0	
H'0_11B0_0000	H'0_11BF_FFFF	1 Mbyte	Reserved	*1
H'0_11A0_0000	H'0_11AF_FFFF	1 Mbyte	Reserved	*1
H'0_1190_0000	H'0_119F_FFFF	1 Mbyte	Reserved	*1
H'0_1187_0000	H'0_118F_FFFF	576 Kbytes	Reserved	*1
H'0_1186_0000	H'0_1186_FFFF	64 Kbytes	OTP	
H'0_1185_0000	H'0_1185_FFFF	64 Kbytes	Reserved	*1
H'0_1184_0000	H'0_1184_FFFF	64 Kbytes	Reserved	*1

Table 5.1 Detailed Address Space (2/4)

Start Address	End Address	Size	Space	Remarks
H'0_1183_0000	H'0_1183_FFFF	64 Kbytes	Non-Secure DMAC (DMAC_NS) APB	
H'0_1182_0000	H'0_1182_FFFF	64 Kbytes	Non-Secure DMAC (DMAC_NS) AXI	
H'0_1181_0000	H'0_1181_FFFF	64 Kbytes	Secure DMAC (DMAC_S) APB	
H'0_1180_0000	H'0_1180_FFFF	64 Kbytes	Secure DMAC (DMAC_S) AXI	
H'0_1170_0000	H'0_117F_FFFF	1 Mbyte	Reserved	*1
H'0_1160_0000	H'0_116F_FFFF	1 Mbyte	DDR (Control Reg)	
H'0_1140_0000	H'0_115F_FFFF	2 Mbytes	DDR (PHY)	
H'0_1130_0000	H'0_113F_FFFF	1 Mbyte	Reserved	*1
H'0_112E_0000	H'0_112F_FFFF	128 Kbytes	Reserved	*1
H'0_112D_0000	H'0_112D_FFFF	64 Kbytes	Reserved	*1
H'0_112C_0000	H'0_112C_FFFF	64 Kbytes	Reserved	*1
H'0_112B_0000	H'0_112B_FFFF	64 Kbytes	SRAM MCPU1 (Reg)	
H'0_112A_0000	H'0_112A_FFFF	64 Kbytes	SRAM MCPU0 (Reg)	
H'0_1129_0000	H'0_1129_FFFF	64 Kbytes	Reserved	*1
H'0_1128_0000	H'0_1128_FFFF	64 Kbytes	SRAM ACPU1 (Reg)	
H'0_1127_0000	H'0_1127_FFFF	64 Kbytes	SRAM ACPU0 (Reg)	
H'0_1126_0000	H'0_1126_FFFF	64 Kbytes	TZC (DDR)	
H'0_1125_0000	H'0_1125_FFFF	64 Kbytes	TZC (xSPI)	
H'0_1124_0000	H'0_1124_FFFF	64 Kbytes	TZC (SRAM MCPU1)	
H'0_1123_0000	H'0_1123_FFFF	64 Kbytes	TZC (SRAM MCPU0)	
H'0_1122_0000	H'0_1122_FFFF	64 Kbytes	Reserved	*1
H'0_1121_0000	H'0_1121_FFFF	64 Kbytes	TZC (SRAM ACPU1)	
H'0_1120_0000	H'0_1120_FFFF	64 Kbytes	TZC (SRAM ACPU0)	
H'0_1110_0000	H'0_111F_FFFF	1 Mbyte	Reserved	*1
H'0_1108_0000	H'0_110F_FFFF	512 Kbytes	Reserved	*1
H'0_1107_0000	H'0_1107_FFFF	64 Kbytes	IM33_FPU (Interrupt controller)	
H'0_1106_0000	H'0_1106_FFFF	64 Kbytes	IM33 (Interrupt controller)	
H'0_1105_0000	H'0_1105_FFFF	64 Kbytes	IA55 (Interrupt controller)	
H'0_1104_0000	H'0_1104_FFFF	64 Kbytes	Reserved	*1
H'0_1103_0000	H'0_1103_FFFF	64 Kbytes	GPIO	
H'0_1102_0000	H'0_1102_FFFF	64 Kbytes	SYSC	
H'0_1101_0000	H'0_1101_FFFF	64 Kbytes	CPG	
H'0_1100_0000	H'0_1100_FFFF	64 Kbytes	SYC	
H'0_10C0_0000	H'0_10FF_FFFF	4 Mbytes	CST (CoreSight)	
H'0_10B0_0000	H'0_10BF_FFFF	1 Mbyte	Reserved	*1
H'0_1080_0000	H'0_10AF_FFFF	3 Mbytes	Reserved	*1
H'0_1070_0000	H'0_107F_FFFF	1 Mbyte	Reserved	*1
H'0_1060_0000	H'0_106F_FFFF	1 Mbyte	Reserved	*1
H'0_1041_0000	H'0_105F_FFFF	1984 Kbytes	Reserved	*1
H'0_1040_0000	H'0_1040_FFFF	64 Kbytes	MHU	
H'0_1030_0000	H'0_103F_FFFF	1 Mbyte	Reserved	*1
H'0_1020_0000	H'0_102F_FFFF	1 Mbyte	Reserved	*1
H'0_100E_0000	H'0_101F_FFFF	1152 Kbytes	Reserved	*1
H'0_100C_0000	H'0_100D_FFFF	128 Kbytes	CANFD	
H'0_110B_0000	H'0_110B_FFFF	64 Kbytes	Reserved	*1

Table 5.1 Detailed Address Space (3/4)

Start Address	End Address	Size	Space	Remarks
H'0_100A_C000	H'0_100A_FFFF	16 Kbytes	Reserved	*1
H'0_100A_B400	H'0_100A_BFFF	3 Kbytes	Reserved	*1
H'0_100A_B000	H'0_100A_B3FF	1 Kbyte	RSPI ch4	
H'0_100A_AC00	H'0_100A_AFFF	1 Kbyte	RSPI ch3	
H'0_100A_A800	H'0_100A_ABFF	1 Kbyte	RSPI ch2	
H'0_100A_A400	H'0_100A_A7FF	1 Kbyte	RSPI ch1	
H'0_100A_A000	H'0_100A_A3FF	1 Kbyte	RSPI ch0	
H'0_100A_9000	H'0_100A_9FFF	4 Kbytes	SPDIF	
H'0_100A_8C00	H'0_100A_8FFF	1 Kbyte	SSIF ch3	
H'0_100A_8800	H'0_100A_8BFF	1 Kbyte	SSIF ch2	
H'0_100A_8400	H'0_100A_87FF	1 Kbyte	SSIF ch1	
H'0_100A_8000	H'0_100A_83FF	1 Kbyte	SSIF ch0	
H'0_100A_7000	H'0_100A_7FFF	4 Kbytes	SRC (Reg)	
H'0_100A_0000	H'0_100A_6FFF	28 Kbytes	SRC (Memory)	
H'0_1009_1000	H'0_1009_FFFF	60 Kbytes	Reserved	*1
H'0_1009_0C00	H'0_1009_0FFF	1 Kbyte	I2C ch3	
H'0_1009_0800	H'0_1009_0BFF	1 Kbyte	I2C ch2	
H'0_1009_0400	H'0_1009_07FF	1 Kbyte	I2C ch1	
H'0_1009_0000	H'0_1009_03FF	1 Kbyte	I2C ch0	
H'0_1008_0000	H'0_1008_FFFF	64 Kbytes	Octa Flash/RAM (Reg) / Reserved	*1,*2-2
H'0_1007_0000	H'0_1007_FFFF	64 Kbytes	xSPI (Write Buf) / Reserved	*1,*2-3
H'0_1006_0000	H'0_1006_FFFF	64 Kbytes	xSPI (Reg) / Reserved	*1,*2-4
H'0_1005_D000	H'0_1005_FFFF	12 Kbytes	Reserved	*1
H'0_1005_C000	H'0_1005_CFFF	4 Kbytes	VBATT	
H'0_1005_B000	H'0_1005_BFFF	4 Kbytes	I3C	
H'0_1005_A000	H'0_1005_AFFF	4 Kbytes	PDM	
H'0_1005_9000	H'0_1005_9FFF	4 Kbytes	TSU	
H'0_1005_8000	H'0_1005_8FFF	4 Kbytes	ADC	
H'0_1005_0000	H'0_1005_7FFF	32 Kbytes	Reserved	*1
H'0_1004_F000	H'0_1004_FFFF	4 Kbytes	Reserved	*1
H'0_1004_EC00	H'0_1004_EFFF	1 Kbyte	RTC	
H'0_1004_E800	H'0_1004_EBFF	1 Kbyte	Reserved	*1
H'0_1004_E400	H'0_1004_E7FF	1 Kbyte	GPT (ELC)	
H'0_1004_E000	H'0_1004_E3FF	1 Kbyte	SCIF ch5	
H'0_1004_D800	H'0_1004_DFFF	2 Kbytes	Reserved	*1
H'0_1004_D400	H'0_1004_D7FF	1 Kbyte	SCI ch1	
H'0_1004_D000	H'0_1004_D3FF	1 Kbyte	SCI ch0	
H'0_1004_CC00	H'0_1004_CFFF	1 Kbyte	IrDA (SCI)	
H'0_1004_C800	H'0_1004_CBFF	1 Kbyte	SCIF ch4	
H'0_1004_C400	H'0_1004_C7FF	1 Kbyte	SCIF ch3	
H'0_1004_C000	H'0_1004_C3FF	1 Kbyte	SCIF ch2	
H'0_1004_BC00	H'0_1004_BFFF	1 Kbyte	SCIF ch1	
H'0_1004_B800	H'0_1004_BBFF	1 Kbyte	SCIF ch0	
H'0_1004_9C00	H'0_1004_B7FF	7 Kbytes	Reserved	*1
H'0_1004_9800	H'0_1004_9BFF	1 Kbyte	POE3	

Table 5.1 Detailed Address Space (4/4)

Start Address	End Address	Size	Space	Remarks
H'0_1004_9400	H'0_1004_97FF	1 Kbyte	POEGD	
H'0_1004_9000	H'0_1004_93FF	1 Kbyte	POEGC	
H'0_1004_8C00	H'0_1004_8FFF	1 Kbyte	POEGB	
H'0_1004_8800	H'0_1004_8BFF	1 Kbyte	POEGA	
H'0_1004_8000	H'0_1004_87FF	2 Kbytes	GPT	
H'0_1004_0000	H'0_1004_7FFF	32 Kbytes	Reserved	*1
H'0_1000_0000	H'0_1003_FFFF	256 Kbytes	MTU3a	
H'0_0012_2000	H'0_0FFF_FFFF	260984 Kbytes	Reserved	*1
H'0_0012_0000	H'0_1012_1FFF	8 Kbytes	Reserved	*1
H'0_000E_0000	H'0_0011_FFFF	256 Kbytes	SRAM ACPU1 (Memory)	
H'0_000A_0000	H'0_000D_FFFF	256 Kbytes	SRAM ACPU0 (Memory)	
H'0_0006_0000	H'0_0009_FFFF	256 Kbytes	SRAM MCPU1 (Memory)	
H'0_0002_0000	H'0_0005_FFFF	256 Kbytes	SRAM MCPU0 (Memory)	
H'0_0000_0000	H'0_0001_FFFF	128 Kbytes	Reserved	*1

Note 1. Access to the reserved areas is prohibited.
If access to the reserved area is attempted, incorrect operation may occur.

Note 2. This is the area where the address map is switched depending on the value of SEL_SPI_OCTA. For the address map switching control, refer to **Section 5.4.4, Address Map switching function.**
*2-1: SEL_SPI_OCTA = 0: xSPI (Mem) area / SEL_SPI_OCTA = 1: Octa Flash/RAM (Mem) area
*2-2: SEL_SPI_OCTA = 0: Reserved area / SEL_SPI_OCTA = 1: Octa Flash/RAM (Reg) area
*2-3: SEL_SPI_OCTA = 0: xSPI (Write Buf) area / SEL_SPI_OCTA = 1: Reserved area
*2-4: SEL_SPI_OCTA = 0: xSPI (Reg) area / SEL_SPI_OCTA = 1: Reserved area

5.2.2 Cortex-M33/Cortex-M33_FPU Address Space

Figure 5.3 shows the default memory map of Cortex-M33/Cortex-M33_FPU and an overview of the Cortex-M33/Cortex-M33_FPU address space for this LSI, and **Table 5.2** shows the detailed address space. For the concept of the secure and non-secure states, see **Section 3, System CPU Cortex-M33/Cortex-M33_FPU**.

In this LSI, addresses H'0000_0000 to H'1FFF_FFFF are used as the code (program) area and addresses H'2000_0000 to H'3FFF_FFFF are used as the data area in the 1-Gbyte SRAM area (H'0000_0000 to H'3FFF_FFFF).

In this LSI, the hardware automatically translates the addresses in the Cortex-M33/Cortex-M33_FPU address space to those in the overall address space of this LSI. Users can therefore write programs using the addresses in the address space shown in **Figure 5.3** and **Table 5.2**, without considering the overall address space.

NOTE

Non-Secure area and Secure area of Cortex-M33/Cortex-M33_FPU Address Space for this LSI in **Figure 5.3** and **Table 5.2** are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. Non-Secure area and Secure area shown in **Figure 5.3** and **Table 5.2** are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

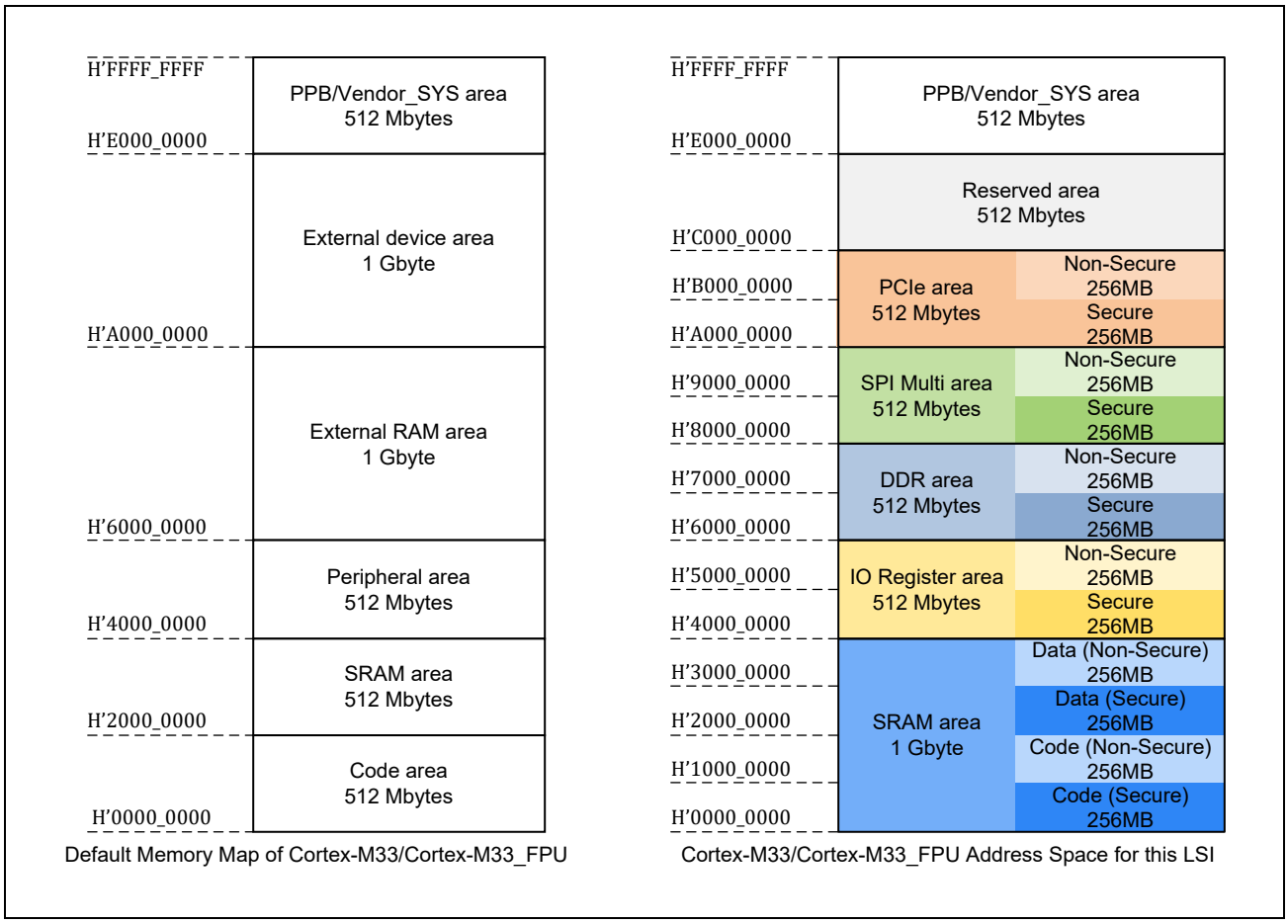


Figure 5.3 Overview of Cortex-M33/Cortex-M33_FPU Address Space

Table 5.2 Detailed Address Space of Cortex-M33/Cortex-M33_FPU (1/8)

Start Address	End Address	Size	Space	Remarks
H'E000_0000	H'FFFF_FFFF	512 Mbytes	PPB / Vendor_SYS	*5
H'C000_0000	H'DFFF_FFFF	512 Mbytes	Reserved	*1
H'B000_0000	H'BFFF_FFFF	256 Mbytes	PCIe (Non-Secure)	
H'A000_0000	H'AFFF_FFFF	256 Mbytes	PCIe (Secure)	
H'9000_0000	H'9FFF_FFFF	256 Mbytes	xSPI / Octa Flash / Octa RAM (Memory) (Secure)	*2-1
H'8000_0000	H'8FFF_FFFF	256 Mbytes	xSPI / Octa Flash / Octa RAM (Memory) (Non-Secure)	*2-1
H'7000_0000	H'7FFF_FFFF	256 Mbytes	DDR (Memory) (Secure)	
H'6000_0000	H'6FFF_FFFF	256 Mbytes	DDR (Memory) (Non-Secure)	
H'5330_0000	H'5FFF_FFFF	205 Mbytes	Reserved	*1
H'52C0_0000	H'532F_FFFF	7 Mbytes	Reserved	*1
H'52A0_0000	H'52BF_FFFF	2 Mbyte	Reserved	*1
H'5280_3000	H'529F_FFFF	2036 Kbytes	Reserved	*1
H'5280_2C00	H'5280_2FFF	1 Kbyte	GTM ch7 (Non-Secure)	
H'5280_2800	H'5280_2BFF	1 Kbyte	GTM ch6 (Non-Secure)	
H'5280_2400	H'5280_27FF	1 Kbyte	GTM ch5 (Non-Secure)	
H'5280_2000	H'5280_23FF	1 Kbyte	GTM ch4 (Non-Secure)	
H'5280_1C00	H'5280_1FFF	1 Kbyte	GTM ch3 (Non-Secure)	
H'5280_1800	H'5280_1BFF	1 Kbyte	GTM ch2 (Non-Secure)	
H'5280_1400	H'5280_17FF	1 Kbyte	GTM ch1 (Non-Secure)	
H'5280_1000	H'5280_13FF	1 Kbyte	GTM ch0 (Non-Secure)	
H'5280_0C00	H'5280_0FFF	1 Kbyte	Reserved	*1
H'5280_0800	H'5280_0BFF	1 Kbyte	WDT CA55 core 0 (Non-Secure)	
H'5280_0400	H'5280_07FF	1 Kbyte	WDT CM33 (Non-Secure)	
H'5280_0000	H'5280_03FF	1 Kbyte	WDT CM33_FPU (Non-Secure)	
H'5270_0000	H'527F_FFFF	1 Mbyte	Reserved	*1
H'5250_0000	H'526F_FFFF	2 Mbytes	Reserved	*1
H'5240_0000	H'524F_FFFF	1 Mbytes	GIC (Non-Secure)	
H'51F0_0000	H'523F_FFFF	5 Mbyte	Reserved	*1
H'51E5_0000	H'51EF_FFFF	704 Kbytes	Reserved	*1
H'51E4_0000	H'51E4_FFFF	64 Kbytes	PCIe (Non-Secure)	
H'51E3_0000	H'51E3_FFFF	64 Kbytes	USB ch1 Host (Non-Secure)	
H'51E2_0000	H'51E2_FFFF	64 Kbytes	USB ch0 (OTG-Func) (Non-Secure)	
H'51E1_0000	H'51E1_FFFF	64 Kbytes	USB ch0 (OTG-Host) (Non-Secure)	
H'51E0_0000	H'51E0_FFFF	64 Kbytes	USBPHY Control (Non-Secure)	
H'51D0_0000	H'51DF_FFFF	1 Mbyte	Reserved	*1
H'51C5_0000	H'51CF_FFFF	704 Kbytes	Reserved	*1
H'51C4_0000	H'51C4_FFFF	64 Kbytes	Ether ch1 (Non-Secure)	
H'51C3_0000	H'51C3_FFFF	64 Kbytes	Ether ch0 (Non-Secure)	
H'51C2_0000	H'51C2_FFFF	64 Kbytes	SD ch2 (Non-Secure)	
H'51C1_0000	H'51C1_FFFF	64 Kbytes	SD ch1 (Non-Secure)	
H'51C0_0000	H'51C0_FFFF	64 Kbytes	SD ch0 (Non-Secure)	
H'51B0_0000	H'51BF_FFFF	1 Mbyte	Reserved	*1
H'51A0_0000	H'51AF_FFFF	1 Mbyte	Reserved	*1
H'5190_0000	H'519F_FFFF	1 Mbyte	Reserved	*1

Table 5.2 Detailed Address Space of Cortex-M33/Cortex-M33_FPU (2/8)

Start Address	End Address	Size	Space	Remarks
H'5187_0000	H'518F_FFFF	576 Kbytes	Reserved	*1
H'5186_0000	H'5186_FFFF	64 Kbytes	OTP (Non-Secure)	
H'5185_0000	H'5185_FFFF	64 Kbytes	Reserved	*1
H'5184_0000	H'5184_FFFF	64 Kbytes	Reserved	*1
H'5183_0000	H'5183_FFFF	64 Kbytes	Non-Secure DMAC (DMAC_NS) APB (Non-Secure)	
H'5182_0000	H'5182_FFFF	64 Kbytes	Non-Secure DMAC (DMAC_NS) AXI (Non-Secure)	
H'5181_0000	H'5181_FFFF	64 Kbytes	Secure DMAC (DMAC_S) APB (Non-Secure)	
H'5180_0000	H'580_FFFF	64 Kbytes	Secure DMAC (DMAC_S) AXI (Non-Secure)	
H'5170_0000	H'517F_FFFF	1 Mbyte	Reserved	*1
H'5160_0000	H'516F_FFFF	1 Mbyte	DDR (Control Reg) (Non-Secure)	
H'5140_0000	H'515F_FFFF	2 Mbytes	DDR (PHY) (Non-Secure)	
H'5130_0000	H'513F_FFFF	1 Mbyte	Reserved	*1
H'512E_0000	H'512F_FFFF	128 Kbytes	Reserved	*1
H'512D_0000	H'512D_FFFF	64 Kbytes	Reserved	*1
H'512C_0000	H'512C_FFFF	64 Kbytes	Reserved	*1
H'512B_0000	H'512B_FFFF	64 Kbytes	SRAM MCPU1 (Reg) (Non-Secure)	
H'512A_0000	H'512A_FFFF	64 Kbytes	SRAM MCPU0 (Reg) (Non-Secure)	
H'5129_0000	H'5129_FFFF	64 Kbytes	Reserved	*1
H'5128_0000	H'5128_FFFF	64 Kbytes	SRAM ACPU1 (Reg) (Non-Secure)	
H'5127_0000	H'5127_FFFF	64 Kbytes	SRAM ACPU0 (Reg) (Non-Secure)	
H'5126_0000	H'5126_FFFF	64 Kbytes	TZC (DDR) (Non-Secure)	
H'5125_0000	H'5125_FFFF	64 Kbytes	TZC (xSPI) (Non-Secure)	
H'5124_0000	H'5124_FFFF	64 Kbytes	TZC (SRAM MCPU1) (Non-Secure)	
H'5123_0000	H'5123_FFFF	64 Kbytes	TZC (SRAM MCPU0) (Non-Secure)	
H'5122_0000	H'5122_FFFF	64 Kbytes	Reserved	*1
H'5121_0000	H'5121_FFFF	64 Kbytes	TZC (SRAM ACPU1) (Non-Secure)	
H'5120_0000	H'5120_FFFF	64 Kbytes	TZC (SRAM ACPU0) (Non-Secure)	
H'5110_0000	H'511F_FFFF	1 Mbyte	Reserved	*1
H'5108_0000	H'510F_FFFF	512 Kbytes	Reserved	*1
H'5107_0000	H'5107_FFFF	64 Kbytes	IM33_FPU (Interrupt controller) (Non-Secure)	
H'5106_0000	H'5106_FFFF	64 Kbytes	IM33 (Interrupt controller) (Non-Secure)	
H'5105_0000	H'5105_FFFF	64 Kbytes	IA55 (Interrupt controller) (Non-Secure)	
H'5104_0000	H'5104_FFFF	64 Kbytes	Reserved	*1
H'5103_0000	H'5103_FFFF	64 Kbytes	GPIO (Non-Secure)	
H'5102_0000	H'5102_FFFF	64 Kbytes	SYSC (Non-Secure)	
H'5101_0000	H'5101_FFFF	64 Kbytes	CPG (Non-Secure)	
H'5100_0000	H'5100_FFFF	64 Kbytes	SYC (Non-Secure)	
H'50C0_0000	H'50FF_FFFF	4 Mbytes	CST (CoreSight) (Non-Secure)	
H'50B0_0000	H'50BF_FFFF	1 Mbyte	Reserved	*1
H'5080_0000	H'50AF_FFFF	3 Mbytes	Reserved	*1
H'5070_0000	H'507F_FFFF	1 Mbyte	Reserved	*1
H'5060_0000	H'506F_FFFF	1 Mbyte	Reserved	*1

Table 5.2 Detailed Address Space of Cortex-M33/Cortex-M33_FPU (3/8)

Start Address	End Address	Size	Space	Remarks
H'5041_0000	H'505F_FFFF	1984 Kbytes	Reserved	*1
H'5040_0000	H'5040_FFFF	64 Kbytes	MHU (Non-Secure)	
H'5030_0000	H'503F_FFFF	1 Mbyte	Reserved	*1
H'5020_0000	H'502F_FFFF	1 Mbyte	Reserved	*1
H'500E_0000	H'501F_FFFF	1152 Kbytes	Reserved	*1
H'500C_0000	H'500D_FFFF	128 Kbytes	CANFD (Non-Secure)	
H'510B_0000	H'510B_FFFF	64 Kbytes	Reserved	*1
H'500A_C000	H'500A_FFFF	16 Kbytes	Reserved	*1
H'500A_B400	H'500A_BFFF	3 Kbytes	Reserved	*1
H'500A_B000	H'500A_B3FF	1 Kbyte	RSPI ch4 (Non-Secure)	
H'500A_AC00	H'500A_AFFF	1 Kbyte	RSPI ch3 (Non-Secure)	
H'500A_A800	H'500A_ABFF	1 Kbyte	RSPI ch2 (Non-Secure)	
H'500A_A400	H'500A_A7FF	1 Kbyte	RSPI ch1 (Non-Secure)	
H'500A_A000	H'500A_A3FF	1 Kbyte	RSPI ch0 (Non-Secure)	
H'500A_9000	H'500A_9FFF	4 Kbytes	SPDIF (Non-Secure)	
H'500A_8C00	H'500A_8FFF	1 Kbyte	SSIF ch3 (Non-Secure)	
H'500A_8800	H'500A_8BFF	1 Kbyte	SSIF ch2 (Non-Secure)	
H'500A_8400	H'500A_87FF	1 Kbyte	SSIF ch1 (Non-Secure)	
H'500A_8000	H'500A_83FF	1 Kbyte	SSIF ch0 (Non-Secure)	
H'500A_7000	H'500A_7FFF	4 Kbytes	SRC (Reg) (Non-Secure)	
H'500A_0000	H'500A_6FFF	28 Kbytes	SRC (Memory) (Non-Secure)	
H'5009_1000	H'5009_FFFF	60 Kbytes	Reserved	*1
H'5009_0C00	H'5009_0FFF	1 Kbyte	I2C ch3 (Non-Secure)	
H'5009_0800	H'5009_0BFF	1 Kbyte	I2C ch2 (Non-Secure)	
H'5009_0400	H'5009_07FF	1 Kbyte	I2C ch1 (Non-Secure)	
H'5009_0000	H'5009_03FF	1 Kbyte	I2C ch0 (Non-Secure)	
H'5008_0000	H'5008_FFFF	64 Kbytes	Octa Flash/RAM (Reg) / Reserved (Non-Secure)	*1,*2-2
H'5007_0000	H'5007_FFFF	64 Kbytes	xSPI (Write Buf) / Reserved (Non-Secure)	*1,*2-3
H'5006_0000	H'5006_FFFF	64 Kbytes	xSPI (Reg) / Reserved (Non-Secure)	*1,*2-4
H'5005_D000	H'5005_FFFF	12 Kbytes	Reserved	*1
H'5005_C000	H'5005_CFFF	4 Kbytes	VBATT (Non-Secure)	
H'5005_B000	H'5005_BFFF	4 Kbytes	I3C (Non-Secure)	
H'5005_A000	H'5005_AFFF	4 Kbytes	PDM (Non-Secure)	
H'5005_9000	H'5005_9FFF	4 Kbytes	TSU (Non-Secure)	
H'5005_8000	H'5005_8FFF	4 Kbytes	ADC (Non-Secure)	
H'5005_0000	H'5005_7FFF	32 Kbytes	Reserved	*1
H'5004_F000	H'5004_FFFF	4 Kbytes	Reserved	*1
H'5004_EC00	H'5004_EFFF	1 Kbyte	RTC (Non-Secure)	
H'5004_E800	H'5004_EBFF	1 Kbyte	Reserved	*1
H'5004_E400	H'5004_E7FF	1 Kbyte	GPT (ELC) (Non-Secure)	
H'5004_E000	H'5004_E3FF	1 Kbyte	SCIF ch5 (Non-Secure)	
H'5004_D800	H'5004_DFFF	2 Kbytes	Reserved	*1
H'5004_D400	H'5004_D7FF	1 Kbyte	SCI ch1 (Non-Secure)	
H'5004_D000	H'5004_D3FF	1 Kbyte	SCI ch0 (Non-Secure)	
H'5004_CC00	H'5004_CFFF	1 Kbyte	IrDA (SCI) (Non-Secure)	

Table 5.2 Detailed Address Space of Cortex-M33/Cortex-M33_FPU (4/8)

Start Address	End Address	Size	Space	Remarks
H'5004_C800	H'5004_CBFF	1 Kbyte	SCIF ch4 (Non-Secure)	
H'5004_C400	H'5004_C7FF	1 Kbyte	SCIF ch3 (Non-Secure)	
H'5004_C000	H'5004_C3FF	1 Kbyte	SCIF ch2 (Non-Secure)	
H'5004_BC00	H'5004_BFFF	1 Kbyte	SCIF ch1 (Non-Secure)	
H'5004_B800	H'5004_BBFF	1 Kbyte	SCIF ch0 (Non-Secure)	
H'5004_9C00	H'5004_B7FF	7 Kbytes	Reserved	*1
H'5004_9800	H'5004_9BFF	1 Kbyte	POE3 (Non-Secure)	
H'5004_9400	H'5004_97FF	1 Kbyte	POEGD (Non-Secure)	
H'5004_9000	H'5004_93FF	1 Kbyte	POEGC (Non-Secure)	
H'5004_8C00	H'5004_8FFF	1 Kbyte	POEGB (Non-Secure)	
H'5004_8800	H'5004_8BFF	1 Kbyte	POEGA (Non-Secure)	
H'5004_8000	H'5004_87FF	2 Kbytes	GPT (Non-Secure)	
H'5004_0000	H'5004_7FFF	32 Kbytes	Reserved	*1
H'5000_0000	H'5003_FFFF	256 Kbytes	MTU3a (Non-Secure)	
H'4330_0000	H'4FFF_FFFF	205 Mbytes	Reserved	*1
H'42C0_0000	H'432F_FFFF	7 Mbytes	Reserved	*1
H'42A0_0000	H'42BF_FFFF	2 Mbyte	Reserved	*1
H'4280_3000	H'429F_FFFF	2036 Kbytes	Reserved	*1
H'4280_2C00	H'4280_2FFF	1 Kbyte	GTM ch7 (Secure)	
H'4280_2800	H'4280_2BFF	1 Kbyte	GTM ch6 (Secure)	
H'4280_2400	H'4280_27FF	1 Kbyte	GTM ch5 (Secure)	
H'4280_2000	H'4280_23FF	1 Kbyte	GTM ch4 (Secure)	
H'4280_1C00	H'4280_1FFF	1 Kbyte	GTM ch3 (Secure)	
H'4280_1800	H'4280_1BFF	1 Kbyte	GTM ch2 (Secure)	
H'4280_1400	H'4280_17FF	1 Kbyte	GTM ch1 (Secure)	
H'4280_1000	H'4280_13FF	1 Kbyte	GTM ch0 (Secure)	
H'4280_0C00	H'4280_0FFF	1 Kbyte	Reserved	*1
H'4280_0800	H'4280_0BFF	1 Kbyte	WDT CA55 core 0 (Secure)	
H'4280_0400	H'4280_07FF	1 Kbyte	WDT CM33 (Secure)	
H'4280_0000	H'4280_03FF	1 Kbyte	WDT CM33_FPU (Secure)	
H'4270_0000	H'427F_FFFF	1 Mbyte	Reserved	*1
H'4250_0000	H'426F_FFFF	2 Mbytes	Reserved	*1
H'4240_0000	H'424F_FFFF	1 Mbytes	GIC (Secure)	
H'41F0_0000	H'423F_FFFF	5 Mbyte	Reserved	*1
H'41E5_0000	H'41EF_FFFF	704 Kbytes	Reserved	*1
H'41E4_0000	H'41E4_FFFF	64 Kbytes	PCIe (Secure)	
H'41E3_0000	H'41E3_FFFF	64 Kbytes	USB ch1 Host (Secure)	
H'41E2_0000	H'41E2_FFFF	64 Kbytes	USB ch0 (OTG-Func) (Secure)	
H'41E1_0000	H'41E1_FFFF	64 Kbytes	USB ch0 (OTG-Host) (Secure)	
H'41E0_0000	H'41E0_FFFF	64 Kbytes	USBPHY Control (Secure)	
H'41D0_0000	H'41DF_FFFF	1 Mbyte	Reserved	*1
H'41C5_0000	H'41CF_FFFF	704 Kbytes	Reserved	*1
H'41C4_0000	H'41C4_FFFF	64 Kbytes	Ether ch1 (Secure)	
H'41C3_0000	H'41C3_FFFF	64 Kbytes	Ether ch0 (Secure)	
H'41C2_0000	H'41C2_FFFF	64 Kbytes	SD ch2 (Secure)	

Table 5.2 Detailed Address Space of Cortex-M33/Cortex-M33_FPU (5/8)

Start Address	End Address	Size	Space	Remarks
H'41C1_0000	H'41C1_FFFF	64 Kbytes	SD ch1 (Secure)	
H'41C0_0000	H'41C0_FFFF	64 Kbytes	SD ch0 (Secure)	
H'41B0_0000	H'41BF_FFFF	1 Mbyte	Reserved	*1
H'41A0_0000	H'41AF_FFFF	1 Mbyte	Reserved	*1
H'4190_0000	H'419F_FFFF	1 Mbyte	Reserved	*1
H'4187_0000	H'418F_FFFF	576 Kbytes	Reserved	*1
H'4186_0000	H'4186_FFFF	64 Kbytes	OTP (Secure)	
H'4185_0000	H'4185_FFFF	64 Kbytes	Reserved	*1
H'4184_0000	H'4184_FFFF	64 Kbytes	Reserved	*1
H'4183_0000	H'4183_FFFF	64 Kbytes	Non-Secure DMAC (DMAC_NS) APB (Secure)	
H'4182_0000	H'4182_FFFF	64 Kbytes	Non-Secure DMAC (DMAC_NS) AXI (Secure)	
H'4181_0000	H'4181_FFFF	64 Kbytes	Secure DMAC (DMAC_S) APB (Secure)	
H'4180_0000	H'480_FFFF	64 Kbytes	Secure DMAC (DMAC_S) AXI (Secure)	
H'4170_0000	H'417F_FFFF	1 Mbyte	Reserved	*1
H'4160_0000	H'416F_FFFF	1 Mbyte	DDR (Control Reg) (Secure)	
H'4140_0000	H'415F_FFFF	2 Mbytes	DDR (PHY) (Secure)	
H'4130_0000	H'413F_FFFF	1 Mbyte	Reserved	*1
H'412E_0000	H'412F_FFFF	128 Kbytes	Reserved	*1
H'412D_0000	H'412D_FFFF	64 Kbytes	Reserved	*1
H'412C_0000	H'412C_FFFF	64 Kbytes	Reserved	*1
H'412B_0000	H'412B_FFFF	64 Kbytes	SRAM MCPU1 (Reg) (Secure)	
H'412A_0000	H'412A_FFFF	64 Kbytes	SRAM MCPU0 (Reg) (Secure)	
H'4129_0000	H'4129_FFFF	64 Kbytes	Reserved	*1
H'4128_0000	H'4128_FFFF	64 Kbytes	SRAM ACPU1 (Reg) (Secure)	
H'4127_0000	H'4127_FFFF	64 Kbytes	SRAM ACPU0 (Reg) (Secure)	
H'4126_0000	H'4126_FFFF	64 Kbytes	TZC (DDR) (Secure)	
H'4125_0000	H'4125_FFFF	64 Kbytes	TZC (xSPI) (Secure)	
H'4124_0000	H'4124_FFFF	64 Kbytes	TZC (SRAM MCPU1) (Secure)	
H'4123_0000	H'4123_FFFF	64 Kbytes	TZC (SRAM MCPU0) (Secure)	
H'4122_0000	H'4122_FFFF	64 Kbytes	Reserved	*1
H'4121_0000	H'4121_FFFF	64 Kbytes	TZC (SRAM ACPU1) (Secure)	
H'4120_0000	H'4120_FFFF	64 Kbytes	TZC (SRAM ACPU0) (Secure)	
H'4110_0000	H'411F_FFFF	1 Mbyte	Reserved	*1
H'4108_0000	H'410F_FFFF	512 Kbytes	Reserved	*1
H'4107_0000	H'4107_FFFF	64 Kbytes	IM33_FPU (Interrupt controller) (Secure)	
H'4106_0000	H'4106_FFFF	64 Kbytes	IM33 (Interrupt controller) (Secure)	
H'4105_0000	H'4105_FFFF	64 Kbytes	IA55 (Interrupt controller) (Secure)	
H'4104_0000	H'4104_FFFF	64 Kbytes	Reserved	*1
H'4103_0000	H'4103_FFFF	64 Kbytes	GPIO (Secure)	
H'4102_0000	H'4102_FFFF	64 Kbytes	SYSC (Secure)	
H'4101_0000	H'4101_FFFF	64 Kbytes	CPG (Secure)	
H'4100_0000	H'4100_FFFF	64 Kbytes	SYC (Secure)	
H'40C0_0000	H'40FF_FFFF	4 Mbytes	CST (CoreSight) (Secure)	
H'40B0_0000	H'40BF_FFFF	1 Mbyte	Reserved	*1

Table 5.2 Detailed Address Space of Cortex-M33/Cortex-M33_FPU (6/8)

Start Address	End Address	Size	Space	Remarks
H'4080_0000	H'40AF_FFFF	3 Mbytes	Reserved	*1
H'4070_0000	H'407F_FFFF	1 Mbyte	Reserved	*1
H'4060_0000	H'406F_FFFF	1 Mbyte	Reserved	*1
H'4041_0000	H'405F_FFFF	1984 Kbytes	Reserved	*1
H'4040_0000	H'4040_FFFF	64 Kbytes	MHU (Secure)	
H'4030_0000	H'403F_FFFF	1 Mbyte	Reserved	*1
H'4020_0000	H'402F_FFFF	1 Mbyte	Reserved	*1
H'400E_0000	H'401F_FFFF	1152 Kbytes	Reserved	*1
H'400C_0000	H'400D_FFFF	128 Kbytes	CANFD (Secure)	
H'410B_0000	H'410B_FFFF	64 Kbytes	Reserved	*1
H'400A_C000	H'400A_FFFF	16 Kbytes	Reserved	*1
H'400A_B400	H'400A_BFFF	3 Kbytes	Reserved	*1
H'400A_B000	H'400A_B3FF	1 Kbyte	RSPI ch4 (Secure)	
H'400A_AC00	H'400A_AFFF	1 Kbyte	RSPI ch3 (Secure)	
H'400A_A800	H'400A_ABFF	1 Kbyte	RSPI ch2 (Secure)	
H'400A_A400	H'400A_A7FF	1 Kbyte	RSPI ch1 (Secure)	
H'400A_A000	H'400A_A3FF	1 Kbyte	RSPI ch0 (Secure)	
H'400A_9000	H'400A_9FFF	4 Kbytes	SPDIF (Secure)	
H'400A_8C00	H'400A_8FFF	1 Kbyte	SSIF ch3 (Secure)	
H'400A_8800	H'400A_8BFF	1 Kbyte	SSIF ch2 (Secure)	
H'400A_8400	H'400A_87FF	1 Kbyte	SSIF ch1 (Secure)	
H'400A_8000	H'400A_83FF	1 Kbyte	SSIF ch0 (Secure)	
H'400A_7000	H'400A_7FFF	4 Kbytes	SRC (Reg) (Secure)	
H'400A_0000	H'400A_6FFF	28 Kbytes	SRC (Memory) (Secure)	
H'4009_1000	H'4009_FFFF	60 Kbytes	Reserved	*1
H'4009_0C00	H'4009_0FFF	1 Kbyte	I2C ch3 (Secure)	
H'4009_0800	H'4009_0BFF	1 Kbyte	I2C ch2 (Secure)	
H'4009_0400	H'4009_07FF	1 Kbyte	I2C ch1 (Secure)	
H'4009_0000	H'4009_03FF	1 Kbyte	I2C ch0 (Secure)	
H'4008_0000	H'4008_FFFF	64 Kbytes	Octa Flash/RAM (Reg) / Reserved (Secure)	*1,*2-2
H'4007_0000	H'4007_FFFF	64 Kbytes	xSPI (Write Buf) / Reserved (Secure)	*1,*2-3
H'4006_0000	H'4006_FFFF	64 Kbytes	xSPI (Reg) / Reserved (Secure)	*1,*2-4
H'4005_D000	H'4005_FFFF	12 Kbytes	Reserved	*1
H'4005_C000	H'4005_CFFF	4 Kbytes	VBATT (Secure)	
H'4005_B000	H'4005_BFFF	4 Kbytes	I3C (Secure)	
H'4005_A000	H'4005_AFFF	4 Kbytes	PDM (Secure)	
H'4005_9000	H'4005_9FFF	4 Kbytes	TSU (Secure)	
H'4005_8000	H'4005_8FFF	4 Kbytes	ADC (Secure)	
H'4005_0000	H'4005_7FFF	32 Kbytes	Reserved	*1
H'4004_F000	H'4004_FFFF	4 Kbytes	Reserved	*1
H'4004_EC00	H'4004_EFFF	1 Kbyte	RTC (Secure)	
H'4004_E800	H'4004_EBFF	1 Kbyte	Reserved	*1
H'4004_E400	H'4004_E7FF	1 Kbyte	GPT (ELC) (Secure)	
H'4004_E000	H'4004_E3FF	1 Kbyte	SCIF ch5 (Secure)	

Table 5.2 Detailed Address Space of Cortex-M33/Cortex-M33_FPU (7/8)

Start Address	End Address	Size	Space	Remarks
H'4004_D800	H'4004_DFFF	2 Kbytes	Reserved	*1
H'4004_D400	H'4004_D7FF	1 Kbyte	SCI ch1 (Secure)	
H'4004_D000	H'4004_D3FF	1 Kbyte	SCI ch0 (Secure)	
H'4004_CC00	H'4004_CFFF	1 Kbyte	IrDA (SCI) (Secure)	
H'4004_C800	H'4004_CBFF	1 Kbyte	SCIF ch4 (Secure)	
H'4004_C400	H'4004_C7FF	1 Kbyte	SCIF ch3 (Secure)	
H'4004_C000	H'4004_C3FF	1 Kbyte	SCIF ch2 (Secure)	
H'4004_BC00	H'4004_BFFF	1 Kbyte	SCIF ch1 (Secure)	
H'4004_B800	H'4004_BBFF	1 Kbyte	SCIF ch0 (Secure)	
H'4004_9C00	H'4004_B7FF	7 Kbytes	Reserved	*1
H'4004_9800	H'4004_9BFF	1 Kbyte	POE3 (Secure)	
H'4004_9400	H'4004_97FF	1 Kbyte	POEGD (Secure)	
H'4004_9000	H'4004_93FF	1 Kbyte	POEGC (Secure)	
H'4004_8C00	H'4004_8FFF	1 Kbyte	POEGB (Secure)	
H'4004_8800	H'4004_8BFF	1 Kbyte	POEGA (Secure)	
H'4004_8000	H'4004_87FF	2 Kbytes	GPT (Secure)	
H'4004_0000	H'4004_7FFF	32 Kbytes	Reserved	*1
H'4000_0000	H'4003_FFFF	256 Kbytes	MTU3a (Secure)	
H'3012_2000	H'3FFF_FFFF	260984 Kbytes	Reserved	*1
H'3012_0000	H'3012_1FFF	8 Kbytes	Reserved	*1
H'300E_0000	H'3011_FFFF	256 Kbytes	SRAM ACPU1 (Memory) (Data, Non-Secure)	*3
H'300A_0000	H'300D_FFFF	256 Kbytes	SRAM ACPU0 (Memory) (Data, Non-Secure)	*3
H'3006_0000	H'3009_FFFF	256 Kbytes	SRAM MCPU1 (Memory) (Data, Non-Secure)	*4
H'3002_0000	H'3005_FFFF	256 Kbytes	SRAM MCPU0 (Memory) (Data, Non-Secure)	*4
H'3000_0000	H'3001_FFFF	128 Kbytes	Reserved	*1
H'2012_2000	H'2FFF_FFFF	260984 Kbytes	Reserved	*1
H'2012_0000	H'2012_1FFF	8 Kbytes	Reserved	*1
H'200E_0000	H'2011_FFFF	256 Kbytes	SRAM ACPU1 (Memory) (Data, Secure)	*3
H'200A_0000	H'200D_FFFF	256 Kbytes	SRAM ACPU0 (Memory) (Data, Secure)	*3
H'2006_0000	H'2009_FFFF	256 Kbytes	SRAM MCPU1 (Memory) (Data, Secure)	*4
H'2002_0000	H'2005_FFFF	256 Kbytes	SRAM MCPU0 (Memory) (Data, Secure)	*4
H'2000_0000	H'2001_FFFF	128 Kbytes	Reserved	*1
H'1012_2000	H'1FFF_FFFF	260984 Kbytes	Reserved	*1
H'1012_0000	H'1012_1FFF	8 Kbytes	Reserved	*1
H'100E_0000	H'1011_FFFF	256 Kbytes	SRAM ACPU1 (Memory) (Code, Non-Secure)	*3
H'100A_0000	H'100D_FFFF	256 Kbytes	SRAM ACPU0 (Memory) (Code, Non-Secure)	*3
H'1006_0000	H'1009_FFFF	256 Kbytes	SRAM MCPU1 (Memory) (Code, Non-Secure)	*4
H'1002_0000	H'1005_FFFF	256 Kbytes	SRAM MCPU0 (Memory) (Code, Non-Secure)	*4
H'1000_0000	H'1001_FFFF	128 Kbytes	Reserved	*1

Table 5.2 Detailed Address Space of Cortex-M33/Cortex-M33_FPU (8/8)

Start Address	End Address	Size	Space	Remarks
H'0012_2000	H'0FFF_FFFF	260984 Kbytes	Reserved	*1
H'0012_0000	H'0012_1FFF	8 Kbytes	Reserved	*1
H'000E_0000	H'0011_FFFF	256 Kbytes	SRAM ACPU1 (Memory) (Code, Secure)	*3
H'000A_0000	H'000D_FFFF	256 Kbytes	SRAM ACPU0 (Memory) (Code, Secure)	*3
H'0006_0000	H'0009_FFFF	256 Kbytes	SRAM MCPU1 (Memory) (Code, Secure)	*4
H'0002_0000	H'0005_FFFF	256 Kbytes	SRAM MCPU0 (Memory) (Code, Secure)	*4
H'0000_0000	H'0001_FFFF	128 Kbytes	Reserved	*1

Note 1. Access to the reserved areas is prohibited.
If access to the reserved area is attempted, incorrect operation may occur.

Note 2. This is the area where the address map is switched depending on the value of SEL_SPI_OCTA. For the address map switching control, refer to **Section 5.4.4, Address Map switching function.**
*2-1: SEL_SPI_OCTA = 0: xSPI (Mem) area / SEL_SPI_OCTA = 1: Octa Flash/RAM (Mem) area
*2-2: SEL_SPI_OCTA = 0: Reserved area / SEL_SPI_OCTA = 1: Octa Flash/RAM (Reg) area
*2-3: SEL_SPI_OCTA = 0: xSPI (Write Buf) area / SEL_SPI_OCTA = 1: Reserved area
*2-4: SEL_SPI_OCTA = 0: xSPI (Reg) area / SEL_SPI_OCTA = 1: Reserved area

Note 3. The same SRAM ACPU0 (256 Kbyes) space is assigned to H'000A_0000 to H'000D_FFFF, H'100A_0000 to H'100D_FFFF, H'200A_0000 to H'200D_FFFF, H'300A_0000 to H'300D_FFFF.
The same SRAM ACPU1 (256 Kbyes) space is assigned to H'000E_0000 to H'0011_FFFF, H'100E_0000 to H'1011_FFFF, H'200E_0000 to H'2011_FFFF, H'300E_0000 to H'3011_FFFF.

Note 4. The same SRAM MCPU0 (256Kbyes) space is assigned to H'0002_0000 to H'0005_FFFF, H'1002_0000 to H'1005_FFFF, H'2002_0000 to H'2005_FFFF, H'3002_0000 to H'3005_FFFF.
The same SRAM MCPU1 (256 Kbyes) space is assigned to H'0006_0000 to H'0009_FFFF, H'1006_0000 to H'1009_FFFF, H'2006_0000 to H'2009_FFFF, H'3006_0000 to H'3009_FFFF.

Note 5. For details on the PPB/Vendor_SYS area, refer to “Arm Cortex-M33 Processor Technical Reference Manual”.

5.3 Accessible Areas

In the bus system of this LSI, each bus master unit can only access the areas that are used for register access or data transfer. **Table 5.3** shows the areas that can be accessed from each master.

The register area of TZC can be accessed only when Secure (AxPROT[1] = 0).

Table 5.3 Accessible Areas (1/2)

Slave Unit	Master Unit										
	Cortex-A55	DMAC_S	DMAC_NS	Cortex-M33	Cortex-M33_FPU	CoreSight [AXI-AP]	CoreSight [ETR]	SDHI (ch0, ch1, ch2)	Gether (ch0, ch1)	USB2.0 (ch0, ch1)	PCIe
DDR (Memory)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MHU	✓	✓	✓	✓	✓	✓	×	×	×	×	×
SRAM ACPU1,0 (Memory)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SRAM MCPU1,0 (Memory)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
xSPI (Memory)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
xSPI (Reg)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
Octa Flash/RAM (Memory)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Octa Flash/RAM (Reg)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
I2C (ch0 to ch3)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
CANFD	✓	✓	✓	✓	✓	✓	×	×	×	×	×
ADC	✓	✓	✓	✓	✓	✓	×	×	×	×	×
TSU	✓	✓	✓	✓	✓	✓	×	×	×	×	×
POEGA	✓	✓	✓	✓	✓	✓	×	×	×	×	×
POEGB	✓	✓	✓	✓	✓	✓	×	×	×	×	×
POEGC	✓	✓	✓	✓	✓	✓	×	×	×	×	×
POEGE	✓	✓	✓	✓	✓	✓	×	×	×	×	×
GPT	✓	✓	✓	✓	✓	✓	×	×	×	×	×
GPT_ELC	✓	✓	✓	✓	✓	✓	×	×	×	×	×
SRC (Memory/Reg)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
SPDIF	✓	✓	✓	✓	✓	✓	×	×	×	×	×
PDM	✓	✓	✓	✓	✓	✓	×	×	×	×	×
I3C	✓	✓	✓	✓	✓	✓	×	×	×	×	×
POE3	✓	✓	✓	✓	✓	✓	×	×	×	×	×
MTU3a	✓	✓	✓	✓	✓	✓	×	×	×	×	×
SSIF (ch0 to ch3)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
RSPI (ch0 to ch4)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
SCIF (ch0 to ch5)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
IrDA (SCI)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
SCI (ch0, ch1)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
WDT (ch0, ch1, ch2)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
GTM (ch0 to ch7)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
OTP	✓	✓	✓	✓	✓	✓	×	×	×	×	×
DMAC_S	✓	✓	✓	✓	✓	✓	×	×	×	×	×

Table 5.3 Accessible Areas (2/2)

Slave Unit	Master Unit										
	Cortex-A55	DMAC_S	DMAC_NS	Cortex-M33	Cortex-M33_FPU	CoreSight [AXI-AP]	CoreSight [ETR]	SDHI (ch0, ch1, ch2)	Gether (ch0, ch1)	USB2.0 (ch0, ch1)	PCIe
DMAC_NS	✓	✓	✓	✓	✓	✓	×	×	×	×	×
SDHI (ch0, ch1, ch2)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
Ether (ch0, ch1)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
USBPHY Control	✓	✓	✓	✓	✓	✓	×	×	×	×	×
USB ch0 (OTG-Host)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
USB ch0 (OTG-Func)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
USB ch1 Host	✓	✓	✓	✓	✓	✓	×	×	×	×	×
PCIe (Memory)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
PCIe (Reg)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
CoreSight	✓	✓	✓	✓	✓	✓	×	×	×	×	×
SYSC	✓	✓	✓	✓	✓	✓	×	×	×	×	×
SRAM ACPU1,0 (Reg)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
SRAM MCPU1,0 (Reg)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
TZC (SRAM ACPU1,0)	✓	✓	×*1	✓	✓	✓	×	×	×	×	×
TZC (SRAM MCPU1,0)	✓	✓	×*1	✓	✓	✓	×	×	×	×	×
TZC (xSPI)	✓	✓	×*1	✓	✓	✓	×	×	×	×	×
TZC (DDR)	✓	✓	×*1	✓	✓	✓	×	×	×	×	×
SYC	✓	✓	✓	✓	✓	✓	×	×	×	×	×
GPIO	✓	✓	✓	✓	✓	✓	×	×	×	×	×
CPG	✓	✓	✓	✓	✓	✓	×	×	×	×	×
IA55 (Interrupt controller)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
IM33 (Interrupt controller)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
IM33_FPU (Interrupt controller)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
GIC	✓	✓	✓	✓	✓	✓	×	×	×	×	×
DDR (Control Reg)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
DDR (PHY)	✓	✓	✓	✓	✓	✓	×	×	×	×	×
VBATT	✓	✓	✓	✓	✓	✓	×	×	×	×	×
RTC	✓	✓	✓	✓	✓	✓	×	×	×	×	×

Note 1. Access to TZC-400 registers is prohibited because DMAC_NS is always fixed to the Non-secure state and unprivileged mode by control signals from the SYSC.

Remarks: ✓: Accessible path, ×: Prohibited path

Note: Usage of paths marked with × in the table is prohibited.
If access to such a path is attempted, incorrect operation may occur.

5.4 Bus System Control

5.4.1 Security Control

5.4.1.1 Re-Setting the Security Attribute Output from Bus Masters

This facility is for re-setting the security attributes of bus transactions that are by bus master units. Specifically, use the master access control register (SYS_MSTACCCTLn: n = 0, 1, 2, 6) to re-set the security attributes. For details on the SYS_MSTACCCTLn register, see **Section 6.3, Register Descriptions**.

NOTE

The bus transaction signals ARPROT[1:0] and AWPROT[1:0] indicate the security attributes, which can be used to protect against illicit transactions.

- ARPROT[0] and AWPROT[0] being 0 or 1 respectively indicate non-privileged access or privileged access.
- ARPROT[1] and AWPROT[1] being 0 or 1 respectively indicate secure access or non-secure access.

The ARPROT and AWPROT signals are prescribed in as part of the AXI protocol. For details, see the AMBA AXI and ACE Protocol Specifications published by Arm Ltd.

Use the SYS_MSTACCCTLn register to make the following settings.

- Selection of security attribute source

  Selection of whether to use the unchanged values of ARPROT[1:0] or AWPROT[1:0] of the ARPROT[2:0] or AWPROT[2:0] signals, which are output from the bus master units, or to re-set ARPROT[1:0] or AWPROT[1:0] according to the settings of the SYS_MSTACCCTL register

- Value to be re-set for ARPROT[1:0] or AWPROT[1:0]

Table 5.4 shows the relation between the registers and bits for use in re-setting the security attributes and the target masters for control.

Table 5.4 List of Register Bits for Use in Re-setting of Security Attributes

Target Master for Control*1	Control Register*2	Write Access Control Bits*3,*4			Read Access Control Bits*3,*5		
		Bit for use in re-setting the security attributes immediately below		Bit for use in selecting the security attribute source	Bit for use in re-setting the security attributes immediately below		Bit for use in selecting the security attribute source
		AWPROT[0]	AWPROT[1]		ARPROT[0]	ARPROT[1]	
Secure DMAC	SYS_MST ACCCTL0	Bit 0: DMAC0_AWPU	Bit 1: DMAC0_AWNS	Bit 3: DMAC0_AWSEL	Bit 4: DMAC0_ARPU	Bit 5: DMAC0_ARNS	Bit 7: DMAC0_ARSEL
Non-Secure DMAC*6	SYS_MST ACCCTL0	Bit 8: DMAC1_AWPU	Bit 9: DMAC1_AWNS	Bit 11: DMAC1_AWSEL	Bit 12: DMAC1_ARPU	Bit 13: DMAC1_ARNS	Bit 15: DMAC1_ARSEL
SDHI/eMMC ch0	SYS_MST ACCCTL1	Bit 0: SDHI0_AWPU	Bit 1: SDHI0_AWNS	Bit 3: SDHI0_AWSEL	Bit 4: SDHI0_ARPU	Bit 5: SDHI0_ARNS	Bit 7: SDHI0_ARSEL
SDHI ch1	SYS_MST ACCCTL1	Bit 8: SDHI1_AWPU	Bit 9: SDHI1_AWNS	Bit 11: SDHI1_AWSEL	Bit 12: SDHI1_ARPU	Bit 13: SDHI1_ARNS	Bit 15: SDHI1_ARSEL
SDHI ch2	SYS_MST ACCCTL6	Bit 0: SDHI2_AWPU	Bit 1: SDHI2_AWNS	Bit 3: SDHI2_AWSEL	Bit 4: SDHI2_ARPU	Bit 5: SDHI2_ARNS	Bit 7: SDHI2_ARSEL
GbEthernet ch0	SYS_MST ACCCTL1	Bit 16: GEther0_AWPU	bit17: GEther0_AWNS	Bit 19: GEther0_AWSEL	Bit 20: GEther0_ARPU	Bit 21: GEther0_ARNS	Bit 23: GEther0_ARSEL
GbEthernet ch1	SYS_MST ACCCTL1	Bit 24: GEther1_AWPU	Bit 25: GEther1_AWNS	Bit 27: GEther1_AWSEL	Bit 28: GEther1_ARPU	Bit 29: GEther1_ARNS	Bit 31: GEther1_ARSEL
USB2.0 ch0 Host*7	SYS_MST ACCCTL2	Bit 0: USB2_0H_AWPU	Bit 1: USB2_0H_AWNS	Bit 3: USB2_0H_AWSEL	Bit 4: USB2_0H_ARPU	Bit 5: USB2_0H_ARNS	Bit 7: USB2_0H_ARSEL
USB2.0 ch0 Function*7	SYS_MST ACCCTL2	Bit 8: USB2_0D_AWPU	Bit 9: USB2_0D_AWNS	Bit 11: USB2_0D_AWSEL	Bit 12: USB2_0D_ARPU	Bit 13: USB2_0D_ARNS	Bit 15: USB2_0D_ARSEL
USB2.0 ch1 Host*7	SYS_MST ACCCTL2	Bit 16: USB2_1H_AWPU	Bit 17: USB2_1H_AWNS	Bit 19: USB2_1H_AWSEL	Bit 20: USB2_1H_ARPU	Bit 21: USB2_1H_ARNS	Bit 23: USB2_1H_ARSEL
PCIe	SYS_MST ACCCTL6	Bit 8: PCIE_AWPU	Bit 9: PCIE_AWNS	Bit 11: PCIE_AWSEL	Bit 12: PCIE_ARPU	Bit 13: PCIE_ARNS	Bit 15: PCIE_ARSEL

Note 1. Re-setting of security attributes for the Cortex-A55, Cortex-M33, and CoreSight bus masters is not possible.

Note 2. See **Section 6.3, Register Descriptions** for details on the registers.

Note 3. See **Section 6.3, Register Descriptions** for details on the bits.

Note 4. The values set in AWPU and AWNS are effective when the setting of the corresponding AWSEL bit is 1. For details, see the specifications of the registers in **Section 6.3, Register Descriptions**.

Note 5. The values set in ARPU and ARNS are effective when the setting of the corresponding ARSEL bit is 1. For details, see the specifications of the registers in **Section 6.3, Register Descriptions**.

Note 6. AWPU and ARPU, AWNS and ARNS, and AWSEL and ARSEL for use in control of the non-secure DMAC are respectively fixed to 1, 1, 0, 0, and 1, 1. Accordingly, access to the non-secure DMAC is always non-secure and non-privileged.

Note 7. When the setting of AWSEL for use in control of the USB2.0 module is 0, the values in AWPROT[0] and ARPROT[0] are the same as the value of HPROT[1] output from the USB2.0 module, and the values of AWPROT[1] and ARPROT[1] are always 1. For control of HPROT[1] in the USB2.0 module, see **Section 32A, USB 2.0 Host Module**. The HPROT signal is prescribed in the HB-Lite protocol. For details, see the AMBA 3 AHB-Lite Protocol Specification published by Arm Ltd.

5.4.1.2 Determining the Security Levels of Bus Slaves

This facility is for enabling or disabling access to a bus slave unit by comparing the security attribute that is input to the bus slave unit in a bus transaction with the security setting of the given slave. The security level can be set in one of the following ways. The applicable way depends on the slave.

- Setting by using the SL[1:0] bits in the slave control register (SYS_SLVACCCTLn: n = 0 to 12, 14, 16 to 18, 20 to 22)
- Setting by using the nsaid_wr_en[31:16] and nsaid_rd_en[15:0] bits in the region ID access register (REGION_ID_ACCESS_<n>: n = 0 to 2) of the TrustZone address space controller (TZC). For details on the TZC, see **Section 13, TrustZone Address Space Controller (TZC)**.

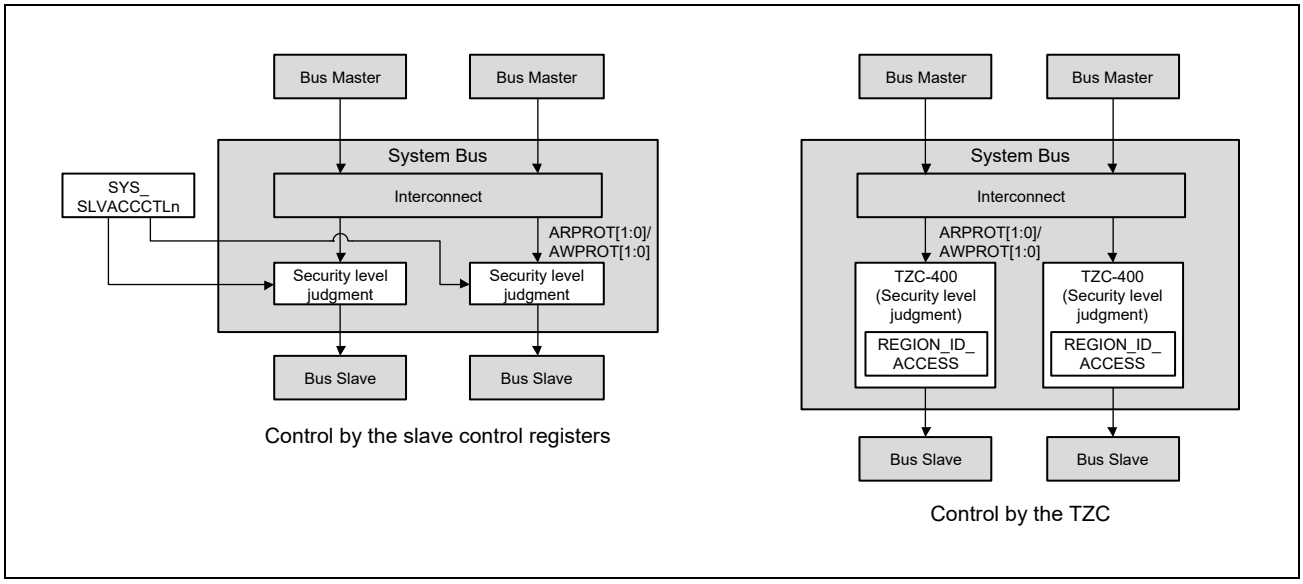


Figure 5.4 Security level determination ways

NOTE

The bus transaction signals ARPROT[1:0] and AWPROT[1:0] indicate the security attributes, which can be used to protect against illicit transactions.

- ARPROT[0] and AWPROT[0] being 0 or 1 respectively indicate non-privileged access or privileged access.
- ARPROT[1] and AWPROT[1] being 0 or 1 respectively indicate secure access or non-secure access.

The ARPROT and AWPROT signals are prescribed in as part of the AXI protocol. For details, see the AMBA AXI and ACE Protocol Specifications published by Arm Ltd.

Table 5.5 shows the correspondence between the settings of the slave control register and the security levels for input transactions. **Table 5.6** shows the correspondence between the settings of the region ID access register of the TZC and the security levels for input bus transactions.

If comparison indicates that access is disabled, a bus error is generated in response to the corresponding attempted bus transaction. Stopping of a slave unit is detected by reference to the module stop state (MSTOP) setting in the CPG. Attempted access to a slave unit in the module stop state leads to a bus error.

Table 5.5 Security Levels Set by Using the Slave Control Register

Setting of SL[1:0]	Security Attribute for Input Bus Transactions			
	Non-privileged Non-secure	Privileged Non-secure	Non-privileged Secure	Privileged Secure
00	Access allowed	Access allowed	Access allowed	Access allowed
01	Access not allowed	Access allowed	Access allowed	Access allowed
10	Access not allowed	Access not allowed	Access allowed	Access allowed
11	Access not allowed	Access not allowed	Access not allowed	Access allowed

Table 5.6 Security Levels Set by Using the TZC

Setting of nsaid_wr_en[31:16] and nsaid_rd_en[15:0]	Security Attribute for Input Bus Transactions			
	Non-privileged Non-secure	Privileged Non-secure	Non-privileged Secure	Privileged Secure
H'000F	Access allowed	Access allowed	Access allowed	Access allowed
H'000B	Access not allowed	Access allowed	Access allowed	Access allowed
H'0003	Access not allowed	Access not allowed	Access allowed	Access allowed
H'0002	Access not allowed	Access not allowed	Access not allowed	Access allowed

Table 5.7 lists the bus slaves for which the security levels are determined by using the slave control registers (SYS_SLVACCCTLn: n = 0 to 12, 14, 16 to 18, 20 to 22). **Table 5.8** lists the bus slaves for which security levels are determined by using the region ID access registers (REGION_ID_ACCESS_<n>) of the TZC.

Table 5.7 Bus Slaves for which the Security Levels are Determined by Using the Slave Control Registers (1/3)

Target Bus Slave for Control*1	Control Register*2	SL[1:0] Allocation Bits*3
SRAM ACPU0 (Reg)	SYS_SLVACCCTL0	Bits [1:0]
SRAM ACPU1 (Reg)	SYS_SLVACCCTL0	Bits [3:2]
SRAM MCPU0 (Reg)	SYS_SLVACCCTL0	Bits [5:4]
SRAM MCPU1 (Reg)	SYS_SLVACCCTL0	Bits [7:6]
TZC (SRAM ACPU0) (Reg)	SYS_SLVACCCTL2	Bits [1:0]
TZC (SRAM ACPU1) (Reg)	SYS_SLVACCCTL2	Bits [3:2]
TZC (SRAM MCPU0) (Reg)	SYS_SLVACCCTL2	Bits [5:4]
TZC (SRAM MCPU1) (Reg)	SYS_SLVACCCTL2	Bits [7:6]
TZC (xSPI) (Reg)	SYS_SLVACCCTL2	Bits [11:10]
TZC (DDR) (Reg)	SYS_SLVACCCTL2	Bits [13:12]
CoreSight	SYS_SLVACCCTL3	Bits [1:0]
CPG	SYS_SLVACCCTL3	Bits [3:2]
SYSC	SYS_SLVACCCTL3	Bits [5:4]
SYC	SYS_SLVACCCTL3	Bits [7:6]
GIC	SYS_SLVACCCTL3	Bits [9:8]
IA55/IM33/IM33_FPU	SYS_SLVACCCTL3	Bits [11:10]
GPIO	SYS_SLVACCCTL3	Bits [13:12]
MHU	SYS_SLVACCCTL3	Bits [15:14]
Secure DMAC	SYS_SLVACCCTL4	Bits [1:0]
Non-Secure DMAC	SYS_SLVACCCTL4	Bits [3:2]
GTM (channel 0)	SYS_SLVACCCTL4	Bits [5:4]

Table 5.7 Bus Slaves for which the Security Levels are Determined by Using the Slave Control Registers (2/3)

Target Bus Slave for Control*1	Control Register*2	SL[1:0] Allocation Bits*3
GTM (channel 1)	SYS_SLVACCCTL4	Bits [7:6]
GTM (channel 2)	SYS_SLVACCCTL4	Bits [9:8]
GTM (channel 3)	SYS_SLVACCCTL4	Bits [11:10]
GTM (channel 4)	SYS_SLVACCCTL4	Bits [13:12]
GTM (channel 5)	SYS_SLVACCCTL4	Bits [15:14]
GTM (channel 6)	SYS_SLVACCCTL4	Bits [17:16]
GTM (channel 7)	SYS_SLVACCCTL4	Bits [19:18]
WDT (channel 0)	SYS_SLVACCCTL4	Bits [21:20]
WDT (channel 1)	SYS_SLVACCCTL4	Bits [23:22]
WDT (channel 2)	SYS_SLVACCCTL4	Bits [25:24]
RTC	SYS_SLVACCCTL4	Bits [29:28]
MTU3a	SYS_SLVACCCTL5	Bits [1:0]
POE3	SYS_SLVACCCTL5	Bits [3:2]
GPT	SYS_SLVACCCTL5	Bits [5:4]
POEG	SYS_SLVACCCTL5	Bits [7:6]
DDR4/LPDDR4 Controller (Reg)	SYS_SLVACCCTL5	Bits [9:8]
xSPI	SYS_SLVACCCTL5	Bits [11:10]
Octa Flash/RAM	SYS_SLVACCCTL5	Bits [13:12]
USBPHY Control	SYS_SLVACCCTL6	Bits [1:0]
USB2.0 (channel 0)	SYS_SLVACCCTL6	Bits [3:2]
USB2.0 (channel 1)	SYS_SLVACCCTL6	Bits [5:4]
SDHI/eMMC (channel 0)	SYS_SLVACCCTL6	Bits [7:6]
SHDI (channel 1)	SYS_SLVACCCTL6	Bits [9:8]
SHDI (channel 2)	SYS_SLVACCCTL6	Bits [11:10]
GbEthernet (channel 0)	SYS_SLVACCCTL6	Bits [13:12]
GbEthernet (channel 1)	SYS_SLVACCCTL6	Bits [15:14]
PCIe	SYS_SLVACCCTL6	Bits [17:16]
I2C (channel 0)	SYS_SLVACCCTL7	Bits [1:0]
I2C (channel 1)	SYS_SLVACCCTL7	Bits [3:2]
I2C (channel 2)	SYS_SLVACCCTL7	Bits [5:4]
I2C (channel 3)	SYS_SLVACCCTL7	Bits [7:6]
I3C	SYS_SLVACCCTL7	Bits [9:8]
CANFD	SYS_SLVACCCTL7	Bits [11:10]
RSPI (channel 0)	SYS_SLVACCCTL7	Bits [13:12]
RSPI (channel 1)	SYS_SLVACCCTL7	Bits [15:14]
RSPI (channel 2)	SYS_SLVACCCTL7	Bits [17:16]
RSPI (channel 3)	SYS_SLVACCCTL7	Bits [19:18]
RSPI (channel 4)	SYS_SLVACCCTL7	Bits [21:20]
SCIF (channel 0)	SYS_SLVACCCTL8	Bits [1:0]
SCIF (channel 1)	SYS_SLVACCCTL8	Bits [3:2]
SCIF (channel 2)	SYS_SLVACCCTL8	Bits [5:4]
SCIF (channel 3)	SYS_SLVACCCTL8	Bits [7:6]
SCIF (channel 4)	SYS_SLVACCCTL8	Bits [9:8]
SCIF (channel 5)	SYS_SLVACCCTL8	Bits [11:10]
SCI (channel 0)	SYS_SLVACCCTL8	Bits [13:12]

Table 5.7 Bus Slaves for which the Security Levels are Determined by Using the Slave Control Registers (3/3)

Target Bus Slave for Control*1	Control Register*2	SL[1:0] Allocation Bits*3
SCI (channel 1)	SYS_SLVACCCTL8	Bits [15:14]
SCI (channel 0) (IrDA)	SYS_SLVACCCTL8	Bits [17:16]
SSIF (channel 0)	SYS_SLVACCCTL9	Bits [1:0]
SSIF (channel 1)	SYS_SLVACCCTL9	Bits [3:2]
SSIF (channel 2)	SYS_SLVACCCTL9	Bits [5:4]
SSIF (channel 3)	SYS_SLVACCCTL9	Bits [7:6]
SRC	SYS_SLVACCCTL9	Bits [9:8]
SPDIF	SYS_SLVACCCTL9	Bits [11:10]
PDM	SYS_SLVACCCTL9	Bits [13:12]
ADC	SYS_SLVACCCTL10	Bits [1:0]
TSU	SYS_SLVACCCTL10	Bits [3:2]
OTP	SYS_SLVACCCTL11	Bits [3:2]
VBATT	SYS_SLVACCCTL11	Bits [11:10]
Registers within the SYSC to control the Cortex-A55*4	SYS_SLVACCCTL12	Bits [1:0]
Registers within the SYSC to control the Cortex-M33*4	SYS_SLVACCCTL12	Bits [3:2]
Registers within the SYSC to control the Cortex-M33_FPU*4	SYS_SLVACCCTL12	Bits [5:4]
Register within the SYSC to control the LSI*4	SYS_SLVACCCTL14	Bits [1:0]
Registers within the SYSC to control the AOF*4	SYS_SLVACCCTL16	Bits [1:0]
Registers within the SYSC to control the Low Power Mode*4	SYS_SLVACCCTL17	Bits [1:0]
General-purpose registers within the SYSC*4	SYS_SLVACCCTL18	Bits [1:0]
IPCONT registers within the SYSC	SYS_SLVACCCTL20	Bits [1:0]

Note 1. The security levels for the following bus slaves incorporated in this product are determined by using TZC-400. For details on the registers for controlling the security levels of these bus slaves, see **Table 5.8, Bus Slaves for which Security Levels are Determined by Using the Region ID Access Registers of the TZC**.
— DDR4 and LPDDR4
— Memory areas of the SRAM ACU and SRAM MCPU
— Memory area, write buffer area, and control register area of SPI Multi I/O

Note 2. See **Section 6.3, Register Descriptions** for details on the registers.

Note 3. See **Section 6.3, Register Descriptions** for details on the bits.

Note 4. The SYSC has registers to control these functions. Setting of the security levels for each of the functions is possible. For details on the SYSC registers, see **Section 6.3, Register Descriptions**.

Table 5.8 Bus Slaves for which Security Levels are Determined by Using the Region ID Access Registers of the TZC

Target Bus Slave for Control	Security Control Unit (locations of the control registers)	Control Register*1	Control Bits*2
DDR	TZC (DDR)	REGION_ID_ACCESS_<n> (n: 0,1,2)*3	nsaid_wr_en[31:16], nsaid_rd_en[15:0]
xSPI, Octa Flash/RAM	TZC (SPI Multi I/O)	REGION_ID_ACCESS_0	nsaid_wr_en[31:16], nsaid_rd_en[15:0]
SRAM MCPU0	TZC (SRAM MCPU0)	REGION_ID_ACCESS_0	nsaid_wr_en[31:16], nsaid_rd_en[15:0]
SRAM MCPU1	TZC (SRAM MCPU1)	REGION_ID_ACCESS_0	nsaid_wr_en[31:16], nsaid_rd_en[15:0]
SRAM ACPU0	TZC (SRAM ACPU0)	REGION_ID_ACCESS_0	nsaid_wr_en[31:16], nsaid_rd_en[15:0]
SRAM ACPU1	TZC (SRAM ACPU1)	REGION_ID_ACCESS_0	nsaid_wr_en[31:16], nsaid_rd_en[15:0]

Note 1. For details on the related registers, see the Technical Reference Manual of CoreLink TZC-400 TrustZone Address Space Controller published by Arm Ltd.

Note 2. For details on the related bits, see the Technical Reference Manual of CoreLink TZC-400 TrustZone Address Space Controller published by Arm Ltd.

Note 3. REGION_ID_ACCESS_0 is for setting the security level for access to the DDR memory by using port 0 of the MEMC.
REGION_ID_ACCESS_1 is for setting the security level for access to the DDR memory by using port 1 of the MEMC.
REGION_ID_ACCESS_2 is for setting the security level for access to the DDR memory by using port 2 of the MEMC.
See **Section 13.1.2, Block Diagram** and **Figure 13.1** for the port number of the MEMC for use in access to the DDR memory from the bus masters.

5.4.2 Address Translation

5.4.2.1 34-Bit Address Space Access

This LSI has bus master units that can handle up to a 34-bit address space and bus master units that can only handle up to a 32-bit address space (actual size of 4 Gbytes). This function enables a bus master unit that can only handle up to a 32-bit address space (actual size of 4 Gbytes) to access an address space of greater than 4 Gbytes. The target bus master units are as follows.

- SDHI/eMMC
- GEthernet
- USB2.0
- DMAC

This function translates bus transaction addresses that are output from a bus master unit in 1-Gbyte units according to the settings of the address offset registers (SYS_AOFn: n = 0 to 6), as shown in the figure below. For details on the SYS_AOFn register, see **Section 6.3, Register Descriptions** in **Section 6, System Controller (SYSC)**.

- In the SYS_AOF registers, set bits [33:30] of the remapping destination address in each 1-Gbyte space corresponding to bits [31:30] of the address that is output from the bus master unit.
- Select the 4-bit value of the SYS_AOF register corresponding to the upper two bits (bits [31:30]) of the 32-bit address that is output from the bus master unit.
- In the selected 4-bit value, use the lower two bits to overwrite bits [31:30] of the original address that was output from the bus master unit, and add the upper two bits as bits [33:32] to the original address.

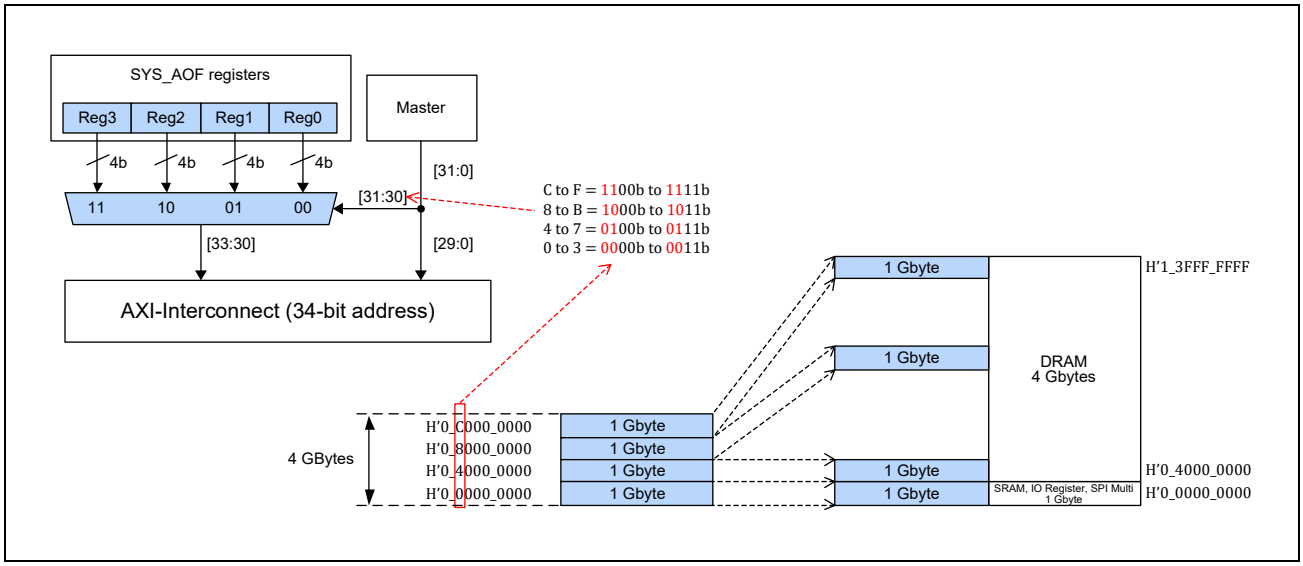


Figure 5.5 Address Space Extension (Overview)

CAUTION

When accessing DRAM from these bus master units, accesses that exceed a 4-Gbyte boundary are prohibited.

5.4.3 Bus Error Interrupt Generation

This LSI generates a bus error interrupt when any of the following conditions is met.

- A bus error is generated when access is disabled by the security setting.
- A bus error is generated when access to a slave unit in the module stop state (MSTOP) is attempted.
- A bus error is generated by a slave unit.

This function is supported for all bus master units included in this LSI.

Bus error interrupts are controlled (retained, cleared, and gathered into a single signal) by the interrupt synchronization circuits (IA55, IM33 and IM33_FPU). For details, including the types of the bus errors, see **Section 8, Interrupt Controller**.

NOTE

This function cannot distinguish between a bus error generated when access is disabled by the security setting and a bus error generated due to other causes.

In addition to the above bus error, an ERRINT interrupt is generated in any of the following cases.

- When the response to a write transaction with the bufferable attribute from the Cortex-M33/Cortex-M33_FPU Code AHB interface is ERROR.
- When the response to a write transaction with the bufferable attribute from the Cortex-M33/Cortex-M33_FPU System AHB interface is ERROR.

5.4.4 Address Map switching function

5.4.4.1 Address Remap Control

This function switches the system address map using the selection signal SEL_SPI_OCTA controlled by the SYS_IPCONT_SEL_SPI_OCTA register of the system control register (SYS Reg). The address map switching target is the Mem/Write Buf/Reg area of xSPI and the Mem/Reg area of Octa Flash/RAM.

As shown in **Figure 5.6**, when SEL_SPI_OCTA = 0, H'1006_0000 to H'1006_FFFF are xSPI (Reg), H'1007_0000 to H'1007_FFFF are xSPI (Write Buf), and H'2000_0000 to H'2FFF_FFFF are xSPI (Mem), respectively. It will be placed and H'1008_0000 to H'1008_FFFF will be Reserved. On the other hand, when SEL_SPI_OCTA = 1, H'1008_0000 to H'1008_FFFF is placed as Octa Flash/RAM (Reg), H'2000_0000 to H'2FFF_FFFF is placed as Octa Flash/RAM (Mem), and H'1006_0000 to H'1007_FFFF is Reserved.

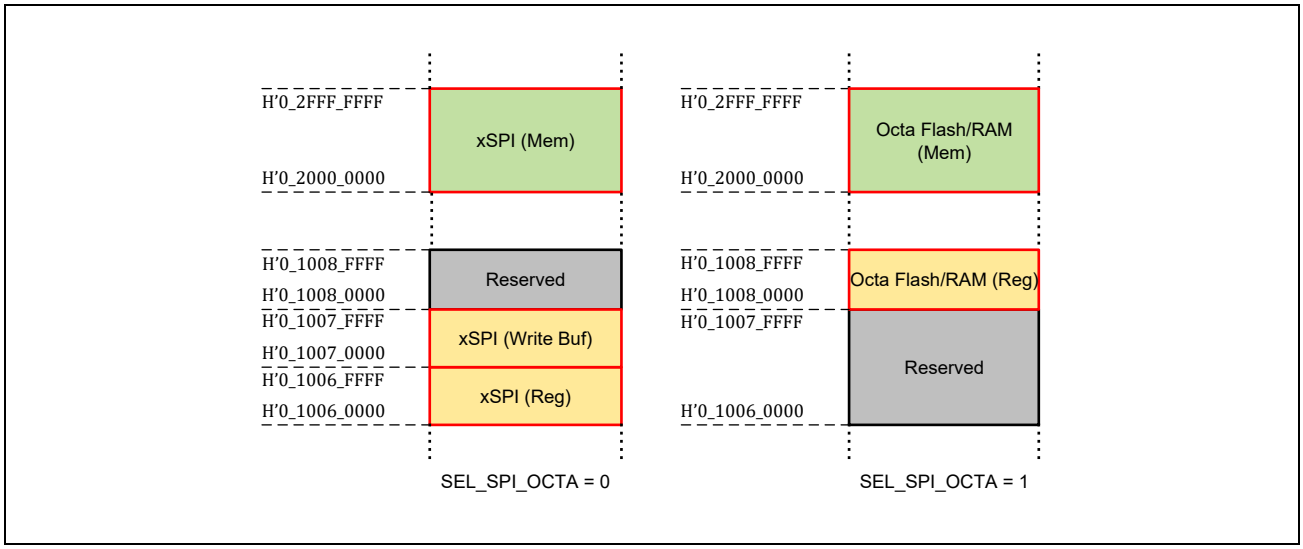


Figure 5.6 Address Remap Function (Overview)

NOTE

Change the settings of the SYS_IPCONT_SEL_SPI_OCTA register when bus access to xSPI and Octa Flash/RAM has not occurred. Also, make sure that access does not occur in the area before address map switching after changing the settings, such as accessing the xSPI area after switching SEL_SPI_OCTA from 0 to 1.

5.4.4.2 MSTOP Control Function

Module stop control is performed for each of the xSPI and Octa Flash/RAM areas for address map switching using one MSTOP control signal (MSTOP_MHSPI, MSTOP_MXOCTA). For module stop control, see **Section 5.4.1.2, Determining the Security Levels of Bus Slaves**.

If a bus access occurs in the area selected by the SEL_SPI_OCTA signal (0: xSPI, 1: Octa Flash/RAM) while the MSTOP signal is asserted, the security level determination function returns a DECERR response. If an access occurs to an area not selected by the SEL_SPI_OCTA signal, the security level determination function does not control it even if the MSTOP signal is asserted.

Table 5.9 MSTOP Control Function (Overview)

Setting			Accessibility to Each Area				
SEL_ SPI_OCTA	MSTP_ MHSPI	MSTOP_ MXOCTA	xSPI Mem	xSPI Write Buffer	xSPI Register	Octa Flash/RAM Mem	Octa Flash/RAM Register
0	0	0	Available			*2	Unavailable (DECERR response)
0	0	1					
0	1	0	Unavailable (DECERR response)				
0	1	1					
1	0	0	*1	Unavailable (DECERR response)		Available	
1	0	1				Unavailable (DECERR response)	
1	1	0				Available	
1	1	1				Unavailable (DECERR response)	

Note 1. Octa Flash/RAM is accessed by the address map switching function.

Note 2. xSPI is accessed by the address map switching function.

6. System Controller (SYSC)

6.1 Overview

6.1.1 Features

SYSC is a unit that performs system control of this LSI and has the following functions.

- Product information, external terminal state capture function
  - Management of product information of this LSI, and external terminal status
- Security control function
  - Re-setting the security attribute of transaction from each bus master
  - Judgement the security level of each bus slave (access management)
- 34-bit address space access function (see **Section 5.4.2.1, 34-Bit Address Space Access** for details)
  - If the bus master address is 32bit, do a 2bit extension to access the 34bit address space
- Low power consumption control (see **Section 41, Low Power Consumption** for details)
  - Control the transition of this LSI to each mode of low power consumption
  - Setting cancellation conditions and confirming factors at the time of cancellation
- WDT stop control
  - Control WDT stop from CoreSight
  - Stop control of specified channel of WDT from CPU

6.1.2 Block Diagram of SYSC

The block diagram of SYSC is shown in **Figure 6.1**.

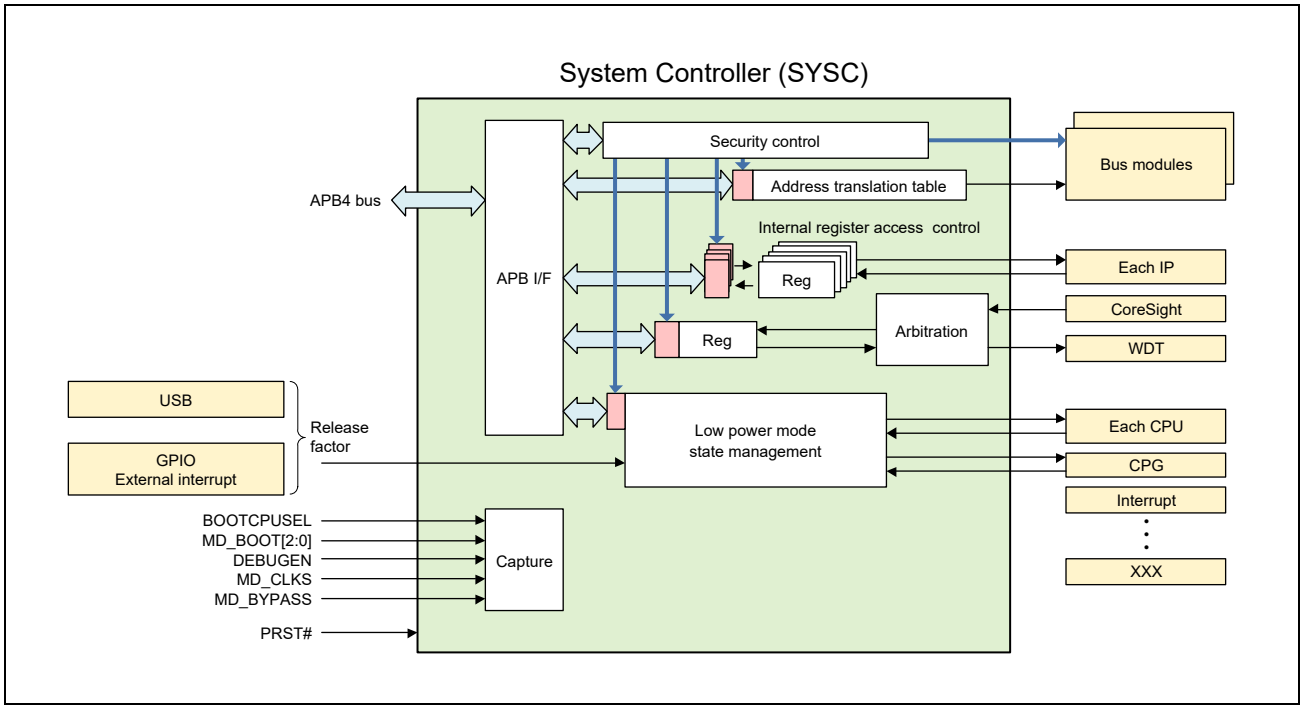


Figure 6.1 SYSC Block Diagram

6.2 Register Configuration

Table 6.1 shows the register configuration of SYSC. The address of the SYSC register is represented by the offset address from the base address. SYSC base address is as follows:

SYSC base address: H'0_1102_0000 (Overall Address Space)
SYSC base address: H'5102_0000 (Cortex-M33 Address Space Non-Secure)
SYSC base address: H'4102_0000 (Cortex-M33 Address Space Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 6.1 SYSC Register Configuration (1/3)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
Master Access Control Register0	SYS_MSTACCCTL0	R/W	H'00AA_AA00	H'0000	32
Master Access Control Register1	SYS_MSTACCCTL1	R/W	H'AAAA_AAAA	H'0004	32
Master Access Control Register2	SYS_MSTACCCTL2	R/W	H'00AA_AAAA	H'0008	32
Master Access Control Register6	SYS_MSTACCCTL6	R/W	H'0000_AAAA	H'0018	32
Slave Access Control Register0	SYS_SLVACCCTL0	R/W	H'0000_0000	H'0100	32
Slave Access Control Register2	SYS_SLVACCCTL2	R/W	H'0000_28AA	H'0108	32
Slave Access Control Register3	SYS_SLVACCCTL3	R/W	H'0000_0030	H'010C	32
Slave Access Control Register4	SYS_SLVACCCTL4	R/W	H'2000_0002	H'0110	32
Slave Access Control Register5	SYS_SLVACCCTL5	R/W	H'0000_0000	H'0114	32
Slave Access Control Register6	SYS_SLVACCCTL6	R/W	H'0000_0000	H'0118	32
Slave Access Control Register7	SYS_SLVACCCTL7	R/W	H'0000_0000	H'011C	32
Slave Access Control Register8	SYS_SLVACCCTL8	R/W	H'0000_0000	H'0120	32
Slave Access Control Register9	SYS_SLVACCCTL9	R/W	H'0000_0000	H'0124	32
Slave Access Control Register10	SYS_SLVACCCTL10	R/W	H'0000_0000	H'0128	32
Slave Access Control Register11	SYS_SLVACCCTL11	R/W	H'0000_02A2	H'012C	32
Slave Access Control Register12	SYS_SLVACCCTL12	R/W	H'0000_0000	H'0130	32
Slave Access Control Register14	SYS_SLVACCCTL14	R/W	H'0000_0000	H'0138	32
Slave Access Control Register16	SYS_SLVACCCTL16	R/W	H'0000_0000	H'0140	32
Slave Access Control Register17	SYS_SLVACCCTL17	R/W	H'0000_0000	H'0144	32
Slave Access Control Register18	SYS_SLVACCCTL18	R/W	H'0000_0000	H'0148	32
Slave Access Control Register20	SYS_SLVACCCTL20	R/W	H'0000_0000	H'0150	32
ECCRAM0 ECC Setting Register	SYS_RAM0_ECC	R/W	H'0000_0000	H'0200	32
ECCRAM0 Access Control Register	SYS_RAM0_EN	R/W	H'0000_0003	H'0204	32
ECCRAM1 ECC Setting Register	SYS_RAM1_ECC	R/W	H'0000_0000	H'0210	32
ECCRAM1 Access Control Register	SYS_RAM1_EN	R/W	H'0000_0003	H'0214	32
ECCRAM2 ECC Setting Register	SYS_RAM2_ECC	R/W	H'0000_0000	H'0220	32
ECCRAM2 Access Control Register	SYS_RAM2_EN	R/W	H'0000_0003	H'0224	32
ECCRAM3 ECC Setting Register	SYS_RAM3_ECC	R/W	H'0000_0000	H'0230	32
ECCRAM3 Access Control Register	SYS_RAM3_EN	R/W	H'0000_0003	H'0234	32
WDT0 Control Register	SYS_WDT0_CTRL	R/W	H'0001_0000	H'0250	32
WDT1 Control Register	SYS_WDT1_CTRL	R/W	H'0001_0000	H'0260	32

Table 6.1 SYSC Register Configuration (2/3)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
WDT2 Control Register	SYS_WDT2_CTRL	R/W	H'0001_0000	H'0270	32
DDR MCREG control Register	SYS_DDR_MCAR_CTRL	R/W	H'0000_0000	H'0304	32
XSPI Start Address for slave0 Register	SYS_XSPI_MAP_STAADD_CS0	R/W	H'2000_0000	H'0348	32
XSPI End Address for slave0 Register	SYS_XSPI_MAP_ENDADD_CS0	R/W	H'2FFF_FFFF	H'034C	32
XSPI Start Address for slave1 Register	SYS_XSPI_MAP_STAADD_CS1	R/W	H'0000_0000	H'0350	32
XSPI End Address for slave1 Register	SYS_XSPI_MAP_ENDADD_CS1	R/W	H'0000_0000	H'0354	32
GEther0 Config Register	SYS_GETH0_CFG	R	H'0000_0000	H'0380	32
GEther1 Config Register	SYS_GETH1_CFG	R	H'0000_0000	H'0390	32
PCIe Config Register	SYS_PCIE_CFG	R/W	H'0000_01FF	H'03A0	32
PCIe Monitor Register	SYS_PCIE_MON	R	H'0000_0000	H'03A4	32
PCIe Error Monitor Register	SYS_PCIE_ERR_MON	R	H'0000_0000	H'03AC	32
PCIe PHY Register	SYS_PCIE_PHY	R/W	H'0000_0001	H'03B4	32
I2C0 Config Register	SYS_I2C0_CFG	R/W	H'0000_0000	H'0400	32
I2C1 Config Register	SYS_I2C1_CFG	R/W	H'0000_0000	H'0410	32
I2C2 Config Register	SYS_I2C2_CFG	R/W	H'0000_0000	H'0420	32
I2C3 Config Register	SYS_I2C3_CFG	R/W	H'0000_0000	H'0430	32
I3C Config Register	SYS_I3C_CFG	R/W	H'0000_0000	H'0440	32
CA55 Core0 Reset Vector Address Low Configuration Register	SYS_CA55_CFG_RVAL0	R/W	H'0000_0000	H'0818	32
CA55 Core0 Reset Vector Address High Configuration Register	SYS_CA55_CFG_RVAH0	R/W	H'0000_0000	H'081C	32
CM33 Config Register0	SYS_CM33_CFG0	R/W	H'0001_312C	H'0844	32
CM33 Config Register1	SYS_CM33_CFG1	R/W	H'0001_312C	H'0848	32
CM33 Config Register2	SYS_CM33_CFG2	R/W	H'0000_0000	H'084C	32
CM33 Config Register3	SYS_CM33_CFG3	R/W	H'1006_0000	H'0850	32
CM33 Lock Register	SYS_CM33_LOCK	R/W	H'0000_0000	H'0854	32
CM33_FPU Config Register0	SYS_CM33FPU CFG0	R/W	H'0001_312C	H'0884	32
CM33_FPU Config Register1	SYS_CM33FPU CFG1	R/W	H'0001_312C	H'0888	32
CM33_FPU Config Register2	SYS_CM33FPU CFG2	R/W	H'0007_0000	H'088C	32
CM33_FPU Config Register3	SYS_CM33FPU CFG3	R/W	H'1008_0000	H'0890	32
CM33_FPU Lock Register	SYS_CM33FPU LOCK	R/W	H'0000_0000	H'0894	32
LSI Mode Signal Register	SYS_LSI_MODE	R	H'000x_xxxx	H'0A00	32
LSI Device ID Register	SYS_LSI_DEVID	R	H'x85E_0447	H'0A04	32
LSI Product Register	SYS_LSI_PRR	R	H'0xxx_xxxx	H'0A08	32
Address Offset Register0	SYS_AOF0	R/W	H'3210_3210	H'0C00	32
Address Offset Register1	SYS_AOF1	R/W	H'3210_3210	H'0C04	32
Address Offset Register2	SYS_AOF2	R/W	H'3210_3210	H'0C08	32
Address Offset Register3	SYS_AOF3	R/W	H'0000_3210	H'0C0C	32
Address Offset Register6	SYS_AOF6	R/W	H'3210_3210	H'0C18	32
Address Offset Register9	SYS_AOF9	R/W	H'0000_3210	H'0C24	32
Lowpower Sequence Control Register1	SYS_LP_CTL1	R/W	H'0000_0000	H'0D04	32
Lowpower Sequence Control Register2	SYS_LP_CTL2	R/W	H'0000_0000	H'0D08	32

Table 6.1 SYSC Register Configuration (3/3)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
Lowpower Sequence Control Register5	SYS_LP_CTL5	R/W	H'0000_0000	H'0D14	32
Lowpower Sequence Control Register6	SYS_LP_CTL6	R/W	H'0000_0000	H'0D18	32
Lowpower Sequence Control Register7	SYS_LP_CTL7	R/W	H'0000_0000	H'0D1C	32
Lowpower Sequence CM33 Control Register0	SYS_LP_CM33CTL0	R	H'0x0x_0xxx	H'0D24	32
CA55 Clock Control Register1	SYS_LP_CA55CK_CTL1	R	H'0000_0x0x	H'0D38	32
CA55 Clock Control Register2	SYS_LP_CA55CK_CTL2	R/W	H'0000_0F06	H'0D3C	32
CA55 Clock Control Register3	SYS_LP_CA55CK_CTL3	R	H'0x0x_0x0x	H'0D40	32
Lowpower Sequence CM33FPU Control Register0	SYS_LP_CM33FPUCTL0	R	H'0x0x_0xxx	H'0D54	32
Isolation Cell Control Register	SYS_PD_ISO_CTRL	R/W	H'0000_0000	H'0D60	32
DDRPHY Control Register	PWRDN_DDRPHY_CTRL	R/W	H'0000_0200	H'0D68	32
Isolation Region IO buffer SE18 Control Register	ISO_IOBUF_SE18_CTRL	R/W	H'0000_0001	H'0D6C	32
USB PWRRDY Register	SYS_USB_PWRRDY	R/W	H'0000_0000	H'0D70	32
PCIe RST_RSM_B terminal Control Register	SYS_PCIE_RST_RSM_B	R/W	H'0000_0000	H'0D74	32
General Register0	SYS_GPREG_0	R/W	H'0000_0000	H'0E00	32
General Register1	SYS_GPREG_1	R/W	H'0000_0000	H'0E04	32
General Register2	SYS_GPREG_2	R/W	H'0000_0000	H'0E08	32
General Register3	SYS_GPREG_3	R/W	H'0000_0000	H'0E0C	32
xSPI/Octa controller Select Register	SYS_IPCONT_SEL_SPI_OCTA	R/W	H'0000_0000	H'0E20	32
Cortex-M33 Address Space Definition Register	SYS_IPCONT_IDAUZERONS	R/W	H'0000_0000	H'0E24	32
Cortex-M33_FPU Address Space Definition Register	SYS_IPCONT_IDAUZERONS_FPU	R/W	H'0000_0000	H'0E28	32

6.3 Register Descriptions

6.3.1 Master Access Control Register 0 (SYS_MSTACCCTL0)

This register sets the secure attribute and privilege attribute of transactions from DMAC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMAC1_ARSEL	—	DMAC1_ARNS	DMAC1_ARPU	DMAC1_AWSEL	—	DMAC1_AWNS	DMAC1_AWPU	DMAC0_ARSEL	—	DMAC0_ARNS	DMAC0_ARPU	DMAC0_AWSEL	—	DMAC0_AWNS	DMAC0_AWPU
Initial Value	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	—	1	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
22	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21 to 19	—	101	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
18	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17, 16	—	10	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
15	DMAC1_ARSEL	1	R	Non-Secure DMAC read access secure attribute source selection The value of this register is always 1. The read value is always 1. For access from Non-Secure DMAC, ARPROT [1: 0] is always 10b.
14	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	DMAC1_ARNS	1	R	Non-Secure DMAC read access secure attribute 1: Non- Secure The value of this bit is always 1. The written value is ignored.
12	DMAC1_ARPU	0	R	Non-Secure DMAC read access privilege attribute 0: Non-privileged access The value of this bit is always 0. The written value is ignored.
11	DMAC1_AWSEL	1	R	Non-Secure DMAC write access security attribute source selection The value of this register is always 1. The read value is always 1 For access from Non-Secure DMAC, AWPROT [1: 0] is always 10b.
10	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
9	DMAC1_AWNS	1	R	Non-Secure DMAC write access secure attribute 1:Non- Secure The value of this bit is always 1. The written value is ignored.
8	DMAC1_AWPU	0	R	Non-Secure DMAC write access privilege attribute 0: Non-privileged access The value of this bit is always 0. The written value is ignored.
7	DMAC0_ARSEL	0	R/W	Secure DMAC read access security attribute source selection*1 0: ARPROT[1: 0] selects the value from Secure DMAC 1: DMAC0_ARPU is selected for ARPROT [0], and DMAC0_ARNS is selected for ARPROT [1].
6	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	DMAC0_ARNS	0	R/W	Secure DMAC read access secure attribute*1 0: Secure 1: Non-secure
4	DMAC0_ARPU	0	R/W	Secure DMAC read access privilege attribute*1 0: Non-privileged access 1: Privileged access
3	DMAC0_AWSEL	0	R/W	Secure DMAC write access security attribute source selection*1 0: AWPROT[1: 0] selects the value from Secure DMAC 1: DMAC0_AWPU is selected for AWPROT [0], and DMAC0_AWNS is selected for AWPROT [1].
2	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	DMAC0_AWNS	0	R/W	Secure DMAC write access secure attribute*1 0: Secure 1: Non-secure
0	DMAC0_AWPU	0	R/W	Secure DMAC write access privilege attribute*1 0: Non-privileged access 1: Privileged access

Note 1. This register setting is prohibited changing the setting while the target master is operating. Change it before the master initiates access.

6.3.2 Master Access Control Register 1 (SYS_MSTACCCTL1)

This register sets the secure attribute and privilege attribute of transactions from GbEthernet and SDHI/eMMC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GEther1_ARSEL	—	GEther1_ARNS	GEther1_ARPU	GEther1_AWSEL	—	GEther1_AWNS	GEther1_AWPU	GEther0_ARSEL	—	GEther0_ARNS	GEther0_ARPU	GEther0_AWSEL	—	GEther0_AWNS	GEther0_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDHI1_ARSEL	—	SDHI1_ARNS	SDHI1_ARPU	SDHI1_AWSEL	—	SDHI1_AWNS	SDHI1_AWPU	SDHI0_ARSEL	—	SDHI0_ARNS	SDHI0_ARPU	SDHI0_AWSEL	—	SDHI0_AWNS	SDHI0_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	GEther1_ARSEL	1	R/W	GbEthernet (channel 1) read access security attribute sourceselection *1 0: ARPROT[1:0] is always 10b 1: GEther1_ARPU is selected for ARPROT [0], and GEther1_ARNS is selected for ARPROT[1].
30	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29	GEther1_ARNS	1	R/W	GbEthernet (channel 1) read access secure attribute*1 0: Secure 1: Non-secure
28	GEther1_ARPU	0	R/W	GbEthernet (channel 1) read access privilege attribute*1 0: Non-privileged access 1: Privileged access
27	GEther1_AWSEL	1	R/W	GbEthernet (channel 1) write access security attribute source selection*1 0: AWPROT[1:0] is always 10b 1: GEther1_AWPU is selected for AWPROT [0], and GEther1_AWNS is selected for AWPROT[1].
26	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	GEther1_AWNS	1	R/W	GbEthernet (channel 1) write access secure attribute*1 0: Secure 1: Non-secure
24	GEther1_AWPU	0	R/W	GbEthernet (channel 1) write access privilege attribute*1 0: Non-privileged access 1: Privileged access
23	GEther0_ARSEL	1	R/W	GbEthernet (channel 0) read access security attribute source selection*1 0: ARPROT[1:0] is always 10b 1: GEther0_ARPU is selected for ARPROT [0], and GEther0_ARNS is selected for ARPROT[1].
22	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	GEther0_ARNS	1	R/W	GbEthernet (channel 0) read access secure attribute*1 0: Secure 1: Non-secure
20	GEther0_ARPU	0	R/W	GbEthernet (channel 0) read access privilege attribute*1 0: Non-privileged access 1: Privileged access

Bit	Bit Name	Initial Value	R/W	Description
19	GEther0_AWSEL	1	R/W	GbEthernet (channel 0) write access security attribute source selection*1 0: AWPROT[1:0] is always 10b 1: GEther0_AWPU is selected for AWPROT [0], and GEther0_AWNS is selected for AWPROT [1].
18	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	GEther0_AWNS	1	R/W	GbEthernet (channel 0) write access secure attribute*1 0: Secure 1: Non-secure
16	GEther0_AWPU	0	R/W	GbEthernet (channel 0) write access privilege attribute*1 0: Non-privileged access 1: Privileged access
15	SDHI1_ARSEL	1	R/W	SDHI (channel 1) read access security attributed source selection*1 0: ARPROT[1:0] is always 10b 1: SDHI1_ARPU is selected for ARPROT [0], and SDHI1_ARNS is selected for ARPROT [1].
14	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	SDHI1_ARNS	1	R/W	SDHI (channel 1) read access secure attribute*1 0: Secure 1: Non-secure
12	SDHI1_ARPU	0	R/W	SDHI (channel 1) read access privilege attribute*1 0: Non-privileged access 1: Privileged access
11	SDHI1_AWSEL	1	R/W	SDHI (channel 1) write access security attribute source selection*1 0: AWPROT[1:0] is always 10b 1: SDHI1_AWPU is selected for AWPROT [0], and SDHI1_AWNS is selected for AWPROT [1].
10	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	SDHI1_AWNS	1	R/W	SDHI (channel 1) write access secure attribute*1 0: Secure 1: Non-secure
8	SDHI1_AWPU	0	R/W	SDHI (channel 1) write access privilege attribute*1 0: Non-privileged access 1: Privileged access
7	SDHI0_ARSEL	1	R/W	SDHI/eMMC (channel 0) read access security attributed source selection *1 0: ARPROT[1:0] is always 10b 1: SDHI0_ARPU is selected for ARPROT [0], and SDHI0_ARNS is selected for ARPROT [1].
6	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	SDHI0_ARNS	1	R/W	SDHI/eMMC (channel 0) read access secure attribute*1 0: Secure 1: Non-secure
4	SDHI0_ARPU	0	R/W	SDHI/eMMC (channel 0) read access privilege attribute*1 0: Non-privileged access 1: Privileged access
3	SDHI0_AWSEL	1	R/W	SDHI/eMMC (channel 0) write access security attribute source selection*1 0: AWPROT[1:0] is always 10b 1: SDHI0_AWPU is selected for AWPROT [0], and SDHI0_AWNS is selected for AWPROT [1].

Bit	Bit Name	Initial Value	R/W	Description
2	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	SDHI0_AWNS	1	R/W	SDHI/eMMC (channel 0) write access secure attribute*1 0: Secure 1: Non-secure
0	SDHI0_AWPU	0	R/W	SDHI/eMMC (channel 0) write access privilege attribute*1 0: Non-privileged access 1: Privileged access

Note 1. This register setting is prohibited changing the setting while the target master is operating. Change it before the master initiates access.

6.3.3 Master Access Control Register 2 (SYS_MSTACCCTL2)

This register sets the secure attribute and privilege attribute of transactions from USB2.0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	USB2_1H_ARSEL	—	USB2_1H_ARNS	USB2_1H_ARPU	USB2_1H_AWSEL	—	USB2_1H_AWNS	USB2_1H_AWPU
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	USB2_0D_ARSEL	—	USB2_0D_ARNS	USB2_0D_ARPU	USB2_0D_AWSEL	—	USB2_0D_AWNS	USB2_0D_AWPU	USB2_0H_ARSEL	—	USB2_0H_ARNS	USB2_0H_ARPU	USB2_0H_AWSEL	—	USB2_0H_AWNS	USB2_0H_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	USB2_1H_ARSEL	1	R/W	USB2.0 (channel 1, host) read access security attribute source selection*1 0: ARPROT[0] selects the HPROT[1] value from USB2.0 channel 1, Host, and ARPROT[1] is always 1. 1: USB2_1H_ARPU is selected for ARPROT[0], and USB2_1H_ARNS is selected for ARPROT[1].
22	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	USB2_1H_ARNS	1	R/W	USB2.0 (channel 1, host) read access secure attribute*1 0: Secure 1: Non-secure
20	USB2_1H_ARPU	0	R/W	USB2.0 (channel 1, host) read access privilege attribute*1 0: Non-privileged access 1: Privileged access
19	USB2_1H_AWSEL	1	R/W	USB2.0 (channel 1, host) write access security attribute source selection*1 0: AWPROT[0] selects the HPROT[1] value from USB2.0 channel 1, Host, and AWPROT [1] is always 1 1: USB2_1H_AWPU is selected for AWPROT[0], and USB2_1H_AWNS is selected for AWPROT[1].
18	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	USB2_1H_AWNS	1	R/W	USB2.0 (channel 1, host) write access secure attribute*1 0: Secure 1: Non-secure
16	USB2_1H_AWPU	0	R/W	USB2.0 (channel 1, host) write access privilege attribute*1 0: Non-privileged access 1: Privileged access
15	USB2_0D_ARSEL	0	R/W	USB2.0 (channel 0, function) read access security attribute source selection*1 0: ARPROT[0] selects the HPROT[1] value from USB2.0 channel 0, Function, and ARPROT[1] is always 1. 1: USB2_0D_ARPU is selected for ARPROT[0], and USB2_0D_ARNS is selected for ARPROT[1].
14	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
13	USB2_0D_ARNS	1	R/W	USB2.0 (channel 0, function) read access secure attribute*1 0: Secure 1: Non-secure
12	USB2_0D_ARPU	0	R/W	USB2.0 (channel 0, function) read access privilege attribute*1 0: Non-privileged access 1: Privileged access
11	USB2_0D_AWSEL	1	R/W	USB2.0 (channel 0, function) write access security attribute source selection*1 0: AWPROT[0] selects the HPROT[1] value from USB2.0 channel 0, Function, and AWPROT[1] is always 1. 1: USB2_0D_AWPU is selected for AWPROT[0], and USB2_0D_AWNS is selected for AWPROT[1].
10	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	USB2_0D_AWNS	1	R/W	USB2.0 (channel 0, function) write access secure attribute*1 0: Secure 1: Non-secure
8	USB2_0D_AWPU	0	R/W	USB2.0 (channel 0, function) write access privilege attribute*1 0: Non-privileged access 1: Privileged access
7	USB2_0H_ARSEL	0	R/W	USB2.0 (channel 0, host) read access security attribute source selection*1 0: ARPROT[0] selects the HPROT[1] value from USB2.0 channel 0 Host, and ARPROT[1] is always 1 1: USB2_0H_ARPU is selected for ARPROT[0], and USB2_0H_ARNS is selected for ARPROT[1].
6	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	USB2_0H_ARNS	1	R/W	USB2.0 (channel 0, host) read access secure attribute*1 0: Secure 1: Non-secure
4	USB2_0H_ARPU	0	R/W	USB2.0 (channel 0, host) read access privilege attribute*1 0: Non-privileged access 1: Privileged access
3	USB2_0H_AWSEL	1	R/W	USB2.0 (channel 0, host) write access security attribute source selection*1 0: AWPROT[1] selects the HPROT[1] value from USB2.0 channel 0 Host, and AWPROT[0] is always 1. 1: USB2_0H_AWPU is selected for AWPROT[0], and USB2_0H_AWNS is selected for AWPROT[1].
2	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	USB2_0H_AWNS	1	R/W	USB2.0 (channel 0, host) write access secure attribute*1 0: Secure 1: Non-secure
0	USB2_0H_AWPU	0	R/W	USB2.0 (channel 0, host) write access privilege attribute*1 0: Non-privileged access 1: Privileged access

Note 1. This register setting is prohibited changing the setting while the target master is operating. Change it before the master initiates access.

6.3.4 Master Access Control Register 6 (SYS_MSTACCCTL6)

This register sets the secure attribute and privilege attribute of transactions from SDHI ch2 and PCIe.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCIE_ARSEL	—	PCIE_ARNS	PCIE_ARPU	PCIE_AWSEL	—	PCIE_AWNS	PCIE_AWPU	SDHI2_ARSEL	—	SDHI2_ARNS	SDHI2_ARPU	SDHI2_AWSEL	—	SDHI2_AWNS	SDHI2_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	H'0000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15	PCIE_ARSEL	0	R/W	PCIe read access security attribute source selection*1 0: ARPROT[1: 0] selects the value from PCIe. 1: PCIE_ARPU is selected for ARPROT[0], and PCIE_ARNS is selected for ARPROT[1].
14	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	PCIE_ARNS	1	R/W	PCIe read access secure attribute*1 0: Secure 1: Non-secure
12	PCIE_ARPU	0	R/W	PCIe read access privilege attribute*1 0: Non-privileged access 1: Privileged access
11	PCIE_AWSEL	1	R/W	PCIe write access security attribute source selection*1 0: AWPROT[1: 0] selects the value from PCIe. 1: PCIE_AWPU is selected for AWPROT[0], and PCIE_AWNS is selected for AWPROT[1].
10	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	PCIE_AWNS	1	R/W	PCIe write access secure attribute*1 0: Secure 1: Non-secure
8	PCIE_AWPU	0	R/W	PCIe write access privilege attribute*1 0: Non-privileged access 1: Privileged access
7	SDHI2_ARSEL	0	R/W	SDHI (channel 2) read access security attribute source selection*1 0: ARPROT[1:0] is always 10b 1: SDHI2_ARPU is selected for ARPROT[0], and SDHI2_ARNS is selected for ARPROT[1].
6	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	SDHI2_ARNS	1	R/W	SDHI (channel 2) read access secure attribute*1 0: Secure 1: Non-secure

Bit	Bit Name	Initial Value	R/W	Description
4	SDHI2_ARPU	0	R/W	SDHI (channel 2) read access privilege attribute*1 0: Non-privileged access 1: Privileged access
3	SDHI2_AWSEL	1	R/W	SDHI (channel 2) write access security attribute source selection*1 0: AWPROT[1:0] is always 10b 1: SDHI2_AWPU is selected for AWPROT[0], and SDHI2_AWNS is selected for AWPROT[1].
2	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	SDHI2_AWNS	1	R/W	SDHI (channel 2) write access secure attribute*1 0: Secure 1: Non-secure
0	SDHI2_AWPU	0	R/W	SDHI (channel 2) write access privilege attribute*1 0: Non-privileged access 1: Privileged access

Note 1. This register setting is prohibited changing the setting while the target master is operating. Change it before the master initiates access.

6.3.5 Slave Access Control Register 0 (SYS_SLVACCCTL0)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the SRAM ACPU0 (Reg), SRAM_ACPU1 (Reg), SRAM_MCPU0 (Reg) and SRAM MCPU1 (Reg).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SRAM3_SL[1:0]		SRAM2_SL[1:0]		SRAM1_SL[1:0]		SRAM0_SL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7, 6	SRAM3_SL [1:0]	00b	R/W	SRAM MCPU1 <Reg> slave access permission control Set the security level SL [1:0] for the slave SRAM MCPU1 <Reg>.
5, 4	SRAM2_SL [1:0]	00b	R/W	SRAM MCPU0 <Reg> slave access permission control Set the security level SL [1:0] for the slave SRAM MCPU0 <Reg>.
3, 2	SRAM1_SL [1:0]	00b	R/W	SRAM ACPU1 <Reg> slave access permission control Set the security level SL [1:0] for the slave SRAM ACPU1 <Reg>.
1, 0	SRAM0_SL [1:0]	00b	R/W	SRAM ACPU0 <Reg> slave access permission control Set the security level SL [1:0] for the slave SRAM ACPU0 <Reg>.

6.3.6 Slave Access Control Register 2 (SYS_SLVACCCTL2)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the TZC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TZC6_SL		TZC5_SL		—	—	TZC3_SL		TZC2_SL		TZC1_SL		TZC0_SL	
Initial Value	0	0	1	0	1	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13, 12	TZC6_SL[1:0]	10b	R/W	TZC (DDR) <Reg> slave access permission control Set the security level SL [1:0] for the slave TZC (DDR) <Reg>. These bits can only be set to 10b or 11b. Setting them to 00b or 01b is prohibited. Operation is not guaranteed when 00b or 01b is set.
11, 10	TZC5_SL[1:0]	10b	R/W	TZC (xSPI, Octa) <Reg> slave access permission control Set the security level SL [1:0] for the slave TZC (xSPI, Octa) <Reg>. These bits can only be set to 10b or 11b. Setting them to 00b or 01b is prohibited. Operation is not guaranteed when 00b or 01b is set.
9, 8	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7, 6	TZC3_SL[1:0]	10b	R/W	TZC (SRAM MCPU1) <Reg> slave access permission control Set the security level SL [1:0] for the slave TZC (SRAM MCPU1) <Reg>. These bits can only be set to 10b or 11b. Setting them to 00b or 01b is prohibited. Operation is not guaranteed when 00b or 01b is set.
5, 4	TZC2_SL[1:0]	10b	R/W	TZC (SRAM MCPU0) <Reg> slave access permission control Set the security level SL [1:0] for the slave TZC (SRAM MCPU0) <Reg>. These bits can only be set to 10b or 11b. Setting them to 00b or 01b is prohibited. Operation is not guaranteed when 00b or 01b is set.
3, 2	TZC1_SL[1:0]	10b	R/W	TZC (SRAM ACPU1) <Reg> slave access permission control Set the security level SL [1:0] for the slave TZC (SRAM ACPU1) <Reg>. These bits can only be set to 10b or 11b. Setting them to 00b or 01b is prohibited. Operation is not guaranteed when 00b or 01b is set.
1, 0	TZC0_SL[1:0]	10b	R/W	TZC (SRAM ACPU0) <Reg> slave access permission control Set the security level SL [1:0] for the slave TZC (SRAM ACPU0) <Reg>. These bits can only be set to 10b or 11b. Setting them to 00b or 01b is prohibited. Operation is not guaranteed when 00b or 01b is set.

6.3.7 Slave Access Control Register 3 (SYS_SLVACCCTL3)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the MHU, GPIO, IA55/IM33, GIC, SYC, SYSC, CPG and CoreSight.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MHU_SL		GPIO_SL		IA55_IM33_SL		GIC_SL		SYC_SL		SYSC_SL		CPG_SL		CST_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15, 14	MHU_SL[1:0]	00b	R/W	MHU slave access permission control Set the security level SL [1:0] for the slave MHU.
13, 12	GPIO_SL[1:0]	00b	R/W	GPIO slave access permission control Set the security level SL [1:0] for the slave GPIO.
11, 10	IA55_IM33_SL[1:0]	00b	R/W	IA55/IM33/IM33_FPU slave access permission control Set the security level SL [1:0] for the slave IA55/IM33.
9, 8	GIC_SL[1:0]	00b	R/W	GIC slave access permission control Set the security level SL [1:0] for the slave GIC.
7, 6	SYC_SL[1:0]	00b	R/W	SYC slave access permission control Set the security level SL [1:0] for the slave SYC.
5, 4	SYSC_SL [1:0]	11b	R	SYSC slave access permission control Reading the bits always returns 11b. Value cannot be written to these bits.
3, 2	CPG_SL[1:0]	00b	R/W	CPG slave access permission control Set the security level SL [1:0] for the slave CPG.
1, 0	CST_SL[1:0]	00b	R/W	CoreSight slave access permission control Set the security level SL [1:0] for the slave CoreSight.

6.3.8 Slave Access Control Register 4 (SYS_SLVACCCTL4)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the DMAC, GTM, WDT and RTC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	RTC_SL		—	—	WDT2_SL		WDT1_SL		WDT0_SL		OSTM7_SL		OSTM6_SL	
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTM5_SL		OSTM4_SL		OSTM3_SL		OSTM2_SL		OSTM1_SL		OSTM0_SL		DMAC1_SL		DMAC0_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29, 28	RTC_SL[1:0]	10b	R/W	RTC slave access permission control Set the security level SL [1:0] for the slave RTC.
27, 26	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
25, 24	WDT2_SL [1:0]	00b	R/W	WDT channel 2 slave access permission control Set the security level SL [1:0] for the slave WDT channel 2.
23, 22	WDT1_SL [1:0]	00b	R/W	WDT channel 1 slave access permission control Set the security level SL [1:0] for the slave WDT channel 1.
21, 20	WDT0_SL [1:0]	00b	R/W	WDT channel 0 slave access permission control Set the security level SL [1:0] for the slave WDT channel 0.
19, 18	OSTM7_SL [1:0]	00b	R/W	GTM channel 7 slave access permission control Set the security level SL [1:0] for the slave GTM channel 7.
17, 16	OSTM6_SL [1:0]	00b	R/W	GTM channel 6 slave access permission control Set the security level SL [1:0] for the slave GTM channel 6.
15, 14	OSTM5_SL [1:0]	00b	R/W	GTM channel 5 slave access permission control Set the security level SL [1:0] for the slave GTM channel 5.
13, 12	OSTM4_SL [1:0]	00b	R/W	GTM channel 4 slave access permission control Set the security level SL [1:0] for the slave GTM channel 4.
11, 10	OSTM3_SL [1:0]	00b	R/W	GTM channel 3 slave access permission control Set the security level SL [1:0] for the slave GTM channel 3.
9, 8	OSTM2_SL [1:0]	00b	R/W	GTM channel 2 slave access permission control Set the security level SL [1:0] for the slave GTM channel 2.
7, 6	OSTM1_SL [1:0]	00b	R/W	GTM channel 1 slave access permission control Set the security level SL [1:0] for the slave GTM channel 1.
5, 4	OSTM0_SL [1:0]	00b	R/W	GTM channel 0 slave access permission control Set the security level SL [1:0] for the slave GTM channel 0.
3, 2	DMAC1_SL [1:0]	00b	R/W	Non-Secure DNAC slave access permission control Set the security level SL [1:0] for the slave Non-Secure DMAC.
1, 0	DMAC0_SL [1:0]	10b	R/W	Secure DMAC slave access permission control Set the security level SL [1:0] for the slave Secure DMAC.

6.3.9 Slave Access Control Register 5 (SYS_SLVACCCTL5)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the MTU3a, POE3, GPT, POEG, DDR <Reg>, xSPI <Reg> and Octa Flash/RAM <Reg>.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	OCTA_SL		XSPI_SL		DDR_SL		POEG_SL		GPT_SL		POE3_SL		MTU3A_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13, 12	OCTA_SL [1:0]	00b	R/W	Octa Flash/RAM <Reg> slave access permission control Set the security level SL [1:0] for the slave Octa Flash/RAM <Reg>.
11, 10	XSPI_SL[1:0]	00b	R/W	xSPI <Reg> slave access permission control Set the security level SL [1:0] for the slave xSPI <Reg>.
9, 8	DDR_SL[1:0]	00b	R/W	DDR <Reg> slave access permission control Set the security level SL [1:0] for the slave DDR <Reg>.
7, 6	POEG_SL [1:0]	00b	R/W	POEG slave access permission control Set the security level SL [1:0] for the slave POEG.
5, 4	GPT_SL[1:0]	00b	R/W	GPT slave access permission control Set the security level SL [1:0] for the slave GPT.
3, 2	POE3_SL [1:0]	00b	R/W	POE3 slave access permission control Set the security level SL [1:0] for the slave POE3.
1, 0	MTU3A_SL [1:0]	00b	R/W	MTU3a slave access permission control Set the security level SL [1:0] for the slave MTU3a.

6.3.10 Slave Access Control Register 6 (SYS_SLVACCCTL6)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the USB2.0, SDHI/eMMC, GbEthernet and PCIe.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PCIE_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ETH1_SL		ETH0_SL		SDHI2_SL		SDHI1_SL		SDHI0_SL		USB21_SL		USB20_SL		USBT_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17, 16	PCIE_SL[1:0]	00b	R/W	PCIe slave access permission control Set the security level SL [1:0] for the slave PCIe.
15, 14	ETH1_SL[1:0]	00b	R/W	GbEthernet channel1 slave access permission control Set the security level SL [1:0] for the slave GbEthernet channel 1.
13, 12	ETH0_SL[1:0]	00b	R/W	GbEthernet channel 0 slave access permission control Set the security level SL [1:0] for the slave GbEthernet channel 0.
11, 10	SDHI2_SL [1:0]	00b	R/W	SDHI channel 2 slave access permission control Set the security level SL [1:0] for the slave SDHI channel 2.
9, 8	SDHI1_SL [1:0]	00b	R/W	SDHI channel 1 slave access permission control Set the security level SL [1:0] for the slave SDHI channel 1.
7, 6	SDHI0_SL [1:0]	00b	R/W	SDHI/eMMC channel 0 slave access permission control Set the security level SL [1:0] for the slave SDHI/eMMC channel 0.
5, 4	USB21_SL [1:0]	00b	R/W	USB2.0 channel 1 slave access permission control Set the security level SL [1:0] for the slave USB2.0 channel 1.
3, 2	USB20_SL [1:0]	00b	R/W	USB2.0 channel 0 slave access permission control Set the security level SL [1:0] for the slave USB2.0 channel 0. USB 2.0 channel 0 has a common security level for Host and Function.
1, 0	USBT_SL [1:0]	00b	R/W	USBPHY Control slave access permission control Set the security level SL [1:0] for the slave USBPHY Control.

6.3.11 Slave Access Control Register 7 (SYS_SLVACCCTL7)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the I2C, I3C, CANFD and RSPI.

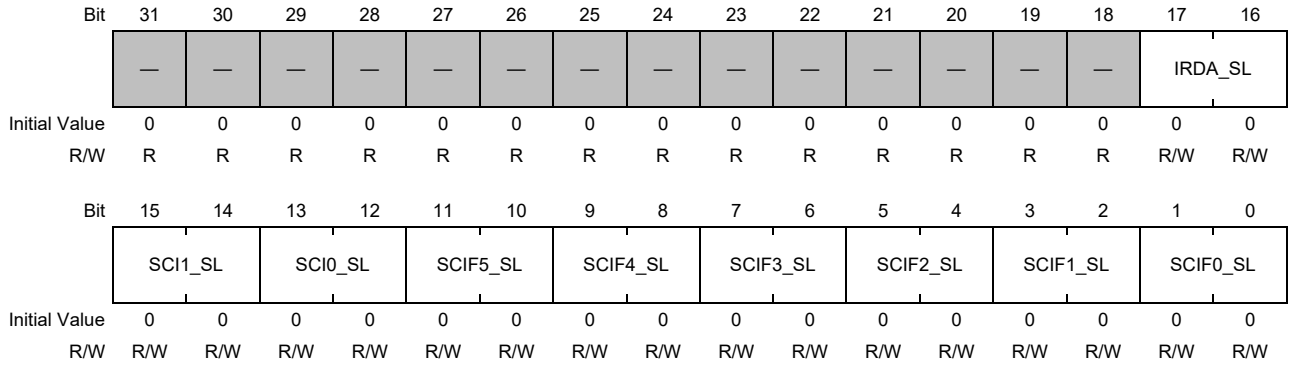
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	RSPI4_SL		RSPI3_SL		RSPI2_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSPI1_SL		RSPI0_SL		CANFD_SL		I3C_SL		I2C3_SL		I2C2_SL		I2C1_SL		I2C0_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21, 20	RSPI4_SL [1:0]	00b	R/W	RSPI channel 4 slave access permission control Set the security level SL [1:0] for the slave RSPI channel 4.
19, 18	RSPI3_SL [1:0]	00b	R/W	RSPI channel 3 slave access permission control Set the security level SL [1:0] for the slave RSPI channel 3.
17, 16	RSPI2_SL [1:0]	00b	R/W	RSPI channel 2 slave access permission control Set the security level SL [1:0] for the slave RSPI channel 2.
15, 14	RSPI1_SL [1:0]	00b	R/W	RSPI channel 1 slave access permission control Set the security level SL [1:0] for the slave RSPI channel 1.
13, 12	RSPI0_SL [1:0]	00b	R/W	RSPI channel 0 slave access permission control Set the security level SL [1:0] for the slave RSPI channel 0.
11, 10	CANFD_SL [1:0]	00b	R/W	CANFD slave access permission control Set the security level SL [1:0] for the slave CANFD.
9, 8	I3C_SL[1:0]	00b	R/W	I3C slave access permission control Set the security level SL [1:0] for the slave I3C.
7, 6	I2C3_SL[1:0]	00b	R/W	I2C channel 3 slave access permission control Set the security level SL [1:0] for the slave I2C channel 3.
5, 4	I2C2_SL[1:0]	00b	R/W	I2C channel 2 slave access permission control Set the security level SL [1:0] for the slave I2C channel 2.
3, 2	I2C1_SL[1:0]	00b	R/W	I2C channel 1 slave access permission control Set the security level SL [1:0] for the slave I2C channel 1.
1, 0	I2C0_SL[1:0]	00b	R/W	I2C channel 0 slave access permission control Set the security level SL [1:0] for the slave I2C channel 0.

6.3.12 Slave Access Control Register 8 (SYS_SLVACCCTL8)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the SCIF, SCI and IrDA.



Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17, 16	IRDA_SL[1:0]	00b	R/W	SCI channel 0 (IrDA) slave access permission control Set the security level SL [1:0] for the slave SCI channel 0 (IrDA).
15, 14	SCI1_SL1:0]	00b	R/W	SCI channel 1 slave access permission control Set the security level SL [1:0] for the slave SCI channel 1.
13, 12	SCI0_SL1:0]	00b	R/W	SCI channel 0 slave access permission control Set the security level SL [1:0] for the slave SCI channel 0.
11, 10	SCIF5_SL1:0]	00b	R/W	SCIF channel 5 slave access permission control Set the security level SL [1:0] for the slave SCIF channel 5.
9, 8	SCIF4_SL1:0]	00b	R/W	SCIF channel 4 slave access permission control Set the security level SL [1:0] for the slave SCIF channel 4.
7, 6	SCIF3_SL1:0]	00b	R/W	SCIF channel 3 slave access permission control Set the security level SL [1:0] for the slave SCIF channel 3.
5, 4	SCIF2_SL1:0]	00b	R/W	SCIF channel 2 slave access permission control Set the security level SL [1:0] for the slave SCIF channel 2.
3, 2	SCIF1_SL1:0]	00b	R/W	SCIF channel 1 slave access permission control Set the security level SL [1:0] for the slave SCIF channel 1.
1, 0	SCIF0_SL1:0]	00b	R/W	SCIF channel 0 slave access permission control Set the security level SL [1:0] for the slave SCIF channel 0.

6.3.13 Slave Access Control Register 9 (SYS_SLVACCCTL9)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the SSIF, SRC, SPDIF and PDM.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PDM_SL		SPDIF_SL		SRC_SL		SSIF3_SL		SSIF2_SL		SSIF1_SL		SSIF0_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13, 12	PDM_SL[1:0]	00b	R/W	PDM slave access permission control Set the security level SL [1:0] for the slave PDM. PDM has a common security level for channel 0 to channel 2.
11, 10	SPDIF_SL [1:0]	00b	R/W	SPDIF slave access permission control Set the security level SL [1:0] for the slave SPDIF.
9, 8	SRC_SL[1:0]	00b	R/W	SRC slave access permission control Set the security level SL [1:0] for the slave SRC.
7, 6	SSIF3_SL [1:0]	00b	R/W	SSIF channel 3 slave access permission control Set the security level SL [1:0] for the slave SSIF channel 3.
5, 4	SSIF2_SL [1:0]	00b	R/W	SSIF channel 2 slave access permission control Set the security level SL [1:0] for the slave SSIF channel 2.
3, 2	SSIF1_SL [1:0]	00b	R/W	SSIF channel 1 slave access permission control Set the security level SL [1:0] for the slave SSIF channel 1.
1, 0	SSIF0_SL [1:0]	00b	R/W	SSIF channel 0 slave access permission control Set the security level SL [1:0] for the slave SSIF channel 0.

6.3.14 Slave Access Control Register 10 (SYS_SLVACCCTL10)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the ADC and TSU.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	TSU_SL		ADC_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3, 2	TSU_SL[1:0]	00b	R/W	TSU slave access permission control Set the security level SL [1:0] for the slave TSU.
1, 0	ADC_SL[1:0]	00b	R/W	ADC slave access permission control Set the security level SL [1:0] for the slave ADC.

6.3.15 Slave Access Control Register 11 (SYS_SLVACCCTL11)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the OTP and VBATT.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VBATT_SL		—	—	—	—	—	—	OTP_SL		—	—
Initial Value	0	0	0	0	0	0	1	0	1	0	1	0	0	0	1	0
R/W	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11, 10	VBATT_SL [1:0]	00b	R/W	VBATT slave access permission control Set the security level SL [1:0] for the slave VBATT.
9 to 4	—	101010b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3, 2	OTP_SL[1:0]	00b	R/W	OTP slave access permission control Set the security level SL [1:0] for the slave OTP.
1, 0	—	10b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.16 Slave Access Control Register 12 (SYS_SLVACCCTL12)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the Cortex-A55 control register and Cortex-M33/Cortex-M33_FPU control register in SYSC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CM33FPU_SL		CM33_SL		CA55_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5, 4	CM33FPU_SL[1:0]	00b	R/W	SYSC <Cortex-M33_FPU Control Reg> slave access permission control Set the security level SL [1:0] for the Cortex-M33_FPU control register in SYSC.
3, 2	CM33_SL [1:0]	00b	R/W	SYSC <Cortex-M33 Control Reg> slave access permission control Set the security level SL [1:0] for the Cortex-M33 control register in SYSC.
1, 0	CA55_SL[1:0]	00b	R/W	SYSC <Cortex-A55 Control Reg> slave access permission control Set the security level SL [1:0] for the Cortex-A55 control register in SYSC.

6.3.17 Slave Access Control Register 14 (SYS_SLVACCCTL14)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the LSI control register in SYSC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LSI_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	LSI_SL[1:0]	00b	R/W	SYSC <LSI Control Reg> slave access permission control Set the security level SL [1:0] for the LSI control register in SYSC.

6.3.18 Slave Access Control Register 16 (SYS_SLVACCCTL16)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the AOF control register in SYSC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AOF_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	AOF_SL[1:0]	00b	R/W	SYSC <AOF Control Reg> slave access permission control Set the security level SL [1:0] for the AOF control register in SYSC.

6.3.19 Slave Access Control Register 17 (SYS_SLVACCCTL17)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the Low Power Mode control register in SYSC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LP_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	LP_SL[1:0]	00b	R/W	SYSC <Low Power Mode Control Reg> slave access permission control Set the security level SL [1:0] for the Low Power Mode control register in SYSC.

6.3.20 Slave Access Control Register 18 (SYS_SLVACCCTL18)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the general purpose register in SYSC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GPREG_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	GPREG_SL [1:0]	00b	R/W	SYSC <GPREG> slave access permission control Set the security level SL [1:0] for the General purpose register(GPREG) in SYSC.

6.3.21 Slave Access Control Register 20 (SYS_SLVACCCTL20)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the general purpose register in SYSC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IPCONT_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	IPCONT_SL [1:0]	00b	R/W	SYSC <IPCONT> slave access permission control Set the security level SL [1:0] for the General purpose register(IPCONT) in SYSC.

6.3.22 ECCRAM0 ECC Setting Register (SYS_RAM0_ECC)

This register is a register that enables and disables the ECC function of ECCRAM0 (SRAM ACPU0).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VECCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	VECCEN	0	R/W	Set enable or disable error correction for ECCRAM0 (SRAM ACPU0) 0: ECC function disabled 1: ECC function enabled

6.3.23 ECCRAM0 Access Control Register (SYS_RAM0_EN)

This register is a register that controls access to ECCRAM0 (SRAM ACPU0).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VLWEN	VCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	VLWEN	1	R/W	Set enable or disable writes to ECRAM0 (SRAM ACPU0) 0: Write disabled 1: Write enabled
0	VCEN	1	R/W	Set enable or disable access to ECRAM0 (SRAM ACPU0) 0: Access disabled 1: Access enabled

6.3.24 ECCRAM1 ECC Setting Register (SYS_RAM1_ECC)

This register is a register that enables and disables the ECC function of ECCRAM1 (SRAM ACPU1).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VECCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	VECCEN	0	R/W	Set enable or disable error correction for ECCRAM1 (SRAM ACPU1) 0: ECC function disabled 1: ECC function enabled

6.3.25 ECCRAM1 Access Control Register (SYS_RAM1_EN)

This register is the register that controls access to ECCRAM1 (SRAM ACPU1).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VLWEN	VCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	VLWEN	1	R/W	Set enable or disable writes to ECCRAM1 (SRAM ACPU1) 0: Write disabled 1: Write enabled
0	VCEN	1	R/W	Set enable or disable access to ECRAM1 (SRAM ACPU1) 0: Access disabled 1: Access enabled

6.3.26 ECCRAM2 ECC Setting Register (SYS_RAM2_ECC)

This register is a register that enables and disables the ECC function of ECCRAM2 (SRAM MCPU0).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VECCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	VECCEN	0	R/W	Set enable or disable error correction for ECCRAM2 (SRAM MCPU0) 0: ECC function disabled 1: ECC function enabled

6.3.27 ECCRAM2 Access Control Register (SYS_RAM2_EN)

This register is a register that controls access to ECCRAM2 (SRAM MCPU0).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VLWEN	VCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	VLWEN	1	R/W	Set enable or disable writes to ECRAM2 (SRAM MCPU0) 0: Write disabled 1: Write enabled
0	VCEN	1	R/W	Set enable or disable access to ECRAM2 (SRAM MCPU0) 0: Access disabled 1: Access enabled

6.3.28 ECCRAM3 ECC Setting Register (SYS_RAM3_ECC)

This register is a register that enables and disables the ECC function of ECCRAM3 (SRAM MCPU1).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VECCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	VECCEN	0	R/W	Set enable or disable error correction for ECCRAM3 (SRAM MCPU1) 0: ECC function disabled 1: ECC function enabled

6.3.29 ECCRAM3 Access Control Register (SYS_RAM3_EN)

This register is the register that controls access to ECCRAM3 (SRAM MCPU1).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VLWEN	VCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	VLWEN	1	R/W	Set enable or disable writes to ECCRAM3 (SRAM MCPU1) 0: Write disabled 1: Write enabled
0	VCEN	1	R/W	Set enable or disable access to ECRAM3 (SRAM MCPU1) 0: Access disabled 1: Access enabled

6.3.30 WDT0 Control Register (SYS_WDT0_CTRL)

This register is the register that set the count control of WDT0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTSTOPMASK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTSTOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	WDTSTOPMASK	1	R/W	Controls the stop function of WDT0 (for Cortex-A55 Core0) from CoreSight during debugging. 0: Stops WDT0 counting during debugging. 1: Not stop WDT0 counting during debugging.
15 to 1	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
0	WDTSTOP	0	R/W	Set whether the WDT0 (for Cortex-A55 Core0) counter is forced to stop 0: Continue counting 1: Stop counting and hold the counter value

6.3.31 WDT1 Control Register (SYS_WDT1_CTRL)

This register is the register that set the count control of WDT1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTSTOPMASK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTSTOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	WDTSTOPMASK	1	R/W	Controls the stop function of WDT1 (for Cortex-M33) from CoreSight during debugging. 0: Stops WDT1 counting during debugging. 1: Not stop WDT1 counting during debugging.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	WDTSTOP	0	R/W	Set whether the WDT1 (for Cortex-M33) counter is forced to stop 0: Continue counting 1: Stop counting and hold the counter value

6.3.32 WDT2 Control Register (SYS_WDT2_CTRL)

This register is the register that set the count control of WDT2.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTSTOPMASK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTSTOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	WDTSTOPMASK	1	R/W	Controls the stop function of WDT2 (for Cortex-M33_FPU) from CoreSight during debugging. 0: Stops WDT2 counting during debugging. 1: Not stop WDT2 counting during debugging.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	WDTSTOP	0	R/W	Set whether the WDT2 (for Cortex-M33_FPU) counter is forced to stop 0: Continue counting 1: Stop counting and hold the counter value

6.3.33 DDR Retention Mode Control Register (SYS_DDR_MCAR_CTRL)

This register is the register that control retention mode of DDR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MCAR_CTRL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	MCAR_CTRL	0	R/W	Retention Mode Control 0: Normal Operation 1: Retention Mode Entry Request
15 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	—	0	R	Reserved When read, 0 is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.34 XSPI CS0 Start Address Register (SYS XSPI_MAP_STAADD_CS0)

This register is the register that indicates the start address of xSPI CS0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MAP_STAADD_CS0															
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MAP_STAADD_CS0															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MAP_STAADD_CS0	H'2000_0000	R/W	Start Address for xSPI CS0

6.3.35 XSPI CS0 End Address Register (SYS XSPI_MAP_ENDADD_CS0)

This register is the register that indicates the end address of xSPI CS0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MAP_ENDADD_CS0															
Initial Value	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MAP_ENDADD_CS0															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MAP_ENDADD_CS0	H'2FFF_FFFF	R/W	End Address for xSPI CS0

6.3.36 XSPI CS1 Start Address Register (SYS XSPI_MAP_STAADD_CS1)

This register is the register that indicates the start address of xSPI CS1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MAP_STAADD_CS1															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MAP_STAADD_CS1															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MAP_STAADD_CS1	All 0	R/W	Start Address for xSPI CS1

6.3.37 XSPI CS1 End Address Register (SYS XSPI_MAP_ENDADD_CS1)

This register is the register that indicates the end address of xSPI CS1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MAP_ENDADD_CS1															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MAP_ENDADD_CS1															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MAP_ENDADD_CS1	All 0	R/W	End Address for xSPI CS1

6.3.38 GEther0 Config Register (SYS_GETH0_CFG)

This register is the register that indicates the state of Ether ch0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	FEC_GIGA_ENABLE	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	FEC_GIGA_ENABLE	0	R	Indicates the mode of speed for Ether ch0 0: 100M/1Mbps mod 1: 1Gbps mode
23 to 0	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.39 Gether1 Config Register (SYS_GETH1_CFG)

This register is the register that indicates the state of Ether ch1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	FEC_GIGA_ENABLE	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	FEC_GIGA_ENABLE	0	R	Indicates the mode of speed for Ether ch1 0: 100M/1Mbps mode 1: 1Gbps mode
23 to 0	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.40 PCIe Config Register (SYS_PCIE_CFG)

This register is the register that set configuration of PCIe.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ALLOW_ENTER_L1	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
8	ALLOW_ENTER_L1	1	R/W	Control transition to L1 state. 0: Suppress transition to L1 state 1: Allow transition to L1 state
7 to 0	—	All 1	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.41 PCIe Monitor Register (SYS_PCIE_MON)

This register is a register that shows PCIe power, clock and device status.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	D_STATE_OUT_F0		—	—	CLKL1PM_REQ	PMU_POWEROFF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5, 4	D_STATE_OUT_F0	00b	R	These bits show the value of PowerState bits of PCI_CFG_PMSC register. 00b: D0 01b: D1 10b: D2 11b: D3hot
3, 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLKL1PM_REQ	0	R	CLKL1PM clock request states 0: No need to supply 100MHz clock 1: Need to supply 100MHz clock
0	PMU_POWEROFF	0	R	Power Off information in L2 state. 0: Not POWEROFF state by transition to L2 state 1: POWEROFF state by transition to L2 state

6.3.42 PCIe Error Monitor Register (SYS_PCIE_ERR_MON)

This register is a register that shows PCIe error detection results.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	ERR_FATAL_DETECTED_F0	ERR_NONFATAL_DETECTED_F0	ERR_COR_DETECTED_F0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	ERR_FATAL_DETECTED_F0	0	R	Fatal error detection result 0: No error 1: Fatal error is detected
1	ERR_NONFATAL_DETECTED_F0	0	R	Non-Fatal error detection result 0: No error 1: Non-Fatal error is detected
0	ERR_COR_DETECTED_F0	0	R	Correctable error detection result 0: No error 1: Correctable error is detected

6.3.43 PCIe PHY Register (SYS_PCIE_PHY)

This register is a register that set the receiver termination setting of PCIe.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MODE_RXTERMINATION
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	MODE_RXTERMINATION	1	R/W	Set Receiver termination setting. 0: Hi-Z 1: 50 ohms to GND

6.3.44 I2C0 Config Register (SYS_I2C0_CFG)

This register is a register that set the bypass mode of the analog filter of I2C ch0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	af_bypass
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	af_bypass	0	R/W	Set whether to bypass the analog filter for noise removal for the input terminals (RIIC0_SDA, RIIC0_SCL) of I2C ch0. 0: Noise reduction is performed using an analog filter for the input signal. 1: The input signal is bypassed an analog filter and noise reduction is not performed.

6.3.45 I2C1 Config Register (SYS_I2C1_CFG)

This register is a register that set the bypass mode of the analog filter of I2C ch1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	af_bypass
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	af_bypass	0	R/W	Set whether to bypass the analog filter for noise removal for the input terminals (RIIC1_SDA, RIIC1_SCL) of I2C ch1. 0: Noise reduction is performed using an analog filter for the input signal. 1: The input signal is bypassed an analog filter and noise reduction is not performed.

6.3.46 I2C2 Config Register (SYS_I2C2_CFG)

This register is a register that set the bypass mode of the analog filter of I2C ch2.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	af_bypass
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	af_bypass	0	R/W	Set whether to bypass the analog filter for noise removal for the input terminals (RIIC2_SDA, RIIC2_SCL) of I2C ch2. 0: Noise reduction is performed using an analog filter for the input signal. 1: The input signal is bypassed an analog filter and noise reduction is not performed.

6.3.47 I2C3 Config Register (SYS_I2C3_CFG)

This register is a register that set the bypass mode of the analog filter of I2C ch3.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	af_bypass
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	af_bypass	0	R/W	Set whether to bypass the analog filter for noise removal for the input terminals (RIIC3_SDA, RIIC3_SCL) of I2C ch3. 0: Noise reduction is performed using an analog filter for the input signal. 1: The input signal is bypassed an analog filter and noise reduction is not performed.

6.3.48 I3C Config Register (SYS_I3C_CFG)

This register is a register that set the bypass mode of the analog filter of I3C.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	af_bypass
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	af_bypass	0	R/W	Set whether to bypass the analog filter for noise removal for the input terminals (I3C_SDA, I3C_SCL) of I3C. 0: Noise reduction is performed using an analog filter for the input signal. 1: The input signal is bypassed an analog filter and noise reduction is not performed.

6.3.49 CA55 Core0 Reset Vector Address Low Configuration Register (SYS_CA55_CFG_RVAL0)

This register is a register that set the reset vector address of Cortex-A55 core0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RVBARADDRL0[31:2]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RVBARADDRL0[31:2]														—	—
Initial Value	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	RVBARADDRL0[31:2]	H'0000_0A00	R/W	Represent bottom bit 31 to bit 2 of the reset vector address of Cortex-A55 Core0. When you change a vector address, you must set the address value and then release the reset (Cold reset/Cluster Warm reset/Core Warm reset). See **Section 2, System CPU Cortex-A55** for the sequence of each reset.
1, 0	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

6.3.50 CA55 Core0 Reset Vector Address High Configuration Register (SYS_CA55_CFG_RVAH0)

This register is a register that set the reset vector address of Cortex-A55 core0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RVBARADDRH0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	RVBARADDRH0[7:0]	H'00	R/W	Represent upper bit 39 to bit 32 of the reset vector address of Cortex-A55 Core0. When you change a vector address, you must set the address value and then release the reset (Cold reset/Cluster Warm reset/Core Warm reset). See **Section 2, System CPU Cortex-A55** for the sequence of each reset.

6.3.51 CM33 Config Register0 (SYS_CM33_CFG0)

This register is the register that set the secure Systick of Cortex-M33.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CONFIGSSYSTICK[25:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CONFIGSSYSTICK[25:0]															
Initial Value	0	0	1	1	0	0	0	1	0	0	1	0	1	1	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25 to 0	CONFIGSSYSTICK[25:0]	H'001_312C	R/W	Configuration register for secure Systick. The value must be determined before the cold reset of Cortex-M33 is released. See the Cortex-M33 chapter for more information on settings.

6.3.52 CM33 Config Register1 (SYS_CM33_CFG1)

This register is the register that set cortex-M33 non-secure Systick.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CONFIGNSSYSTICK[25:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CONFIGNSSYSTICK[25:0]															
Initial Value	0	0	1	1	0	0	0	1	0	0	1	0	1	1	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25 to 0	CONFIGNSSYSTICK[25:0]	H'001_312C	R/W	Configuration register for non-secure Systick. The value must be determined before the cold reset of Cortex-M33 is released. See the Cortex-M33 chapter for more information on settings.

6.3.53 CM33 Config Register2 (SYS_CM33_CFG2)

This register is the register that set the secure vector address of Cortex-M33.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INITSVTOR[31:7]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INITSVTOR[31:7]									—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	INITSVTOR[31:7]	H'000_0000	R/W	A register that indicates the secure vector address. The address value must be determined before the Cortex-M33 cold reset is released.
6 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

6.3.54 CM33 Config Register3 (SYS_CM33_CFG3)

This register is the register that set the non-secure vector address of Cortex-M33.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INITNSVTOR[31:7]															
Initial Value	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INITNSVTOR[31:7]									—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	INITNSVTOR[31:7]	H'020_0C00	R/W	A register that indicates the non-secure vector address. The address value must be determined before the Cortex-M33 cold reset is released.
6 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

6.3.55 CM33 Lock Register (SYS_CM33_LOCK)

This register is a register that set the vector-address change permission of Cortex-M33.

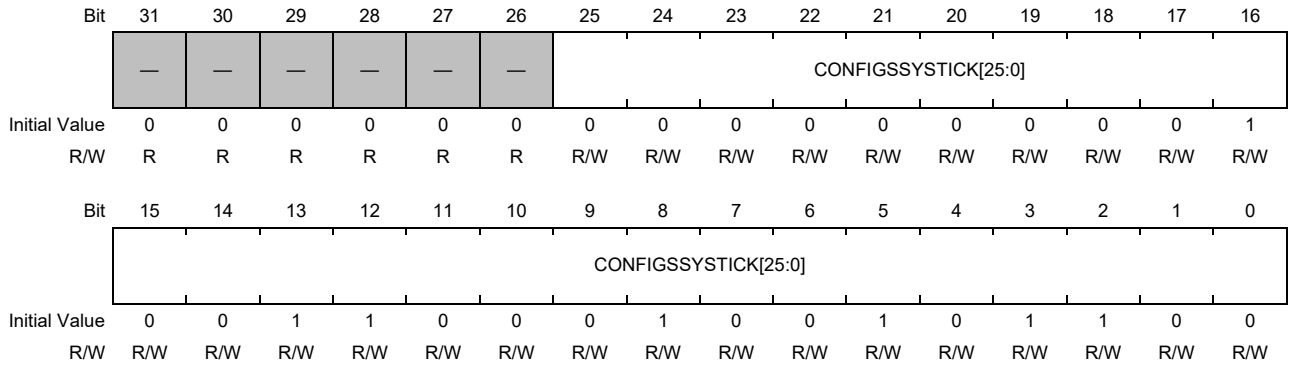
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LOCKNSVTOR	LOCKSVTAIRCR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	LOCKNSVTOR	0	R/W	Control the non-secure vector address change permission. 0: Changeable 1: Cannot be changed
0	LOCKSVTAIRCR	0	R/W	Control changing permission of the Secure Vector address, and the PRIS bit and the BFHFNMINS bit of the AIRCR register of Cortex-M33. 0: Changeable 1: Cannot be changed

6.3.56 CM33_FPU Config Register0 (SYS_CM33FPU_CFG0)

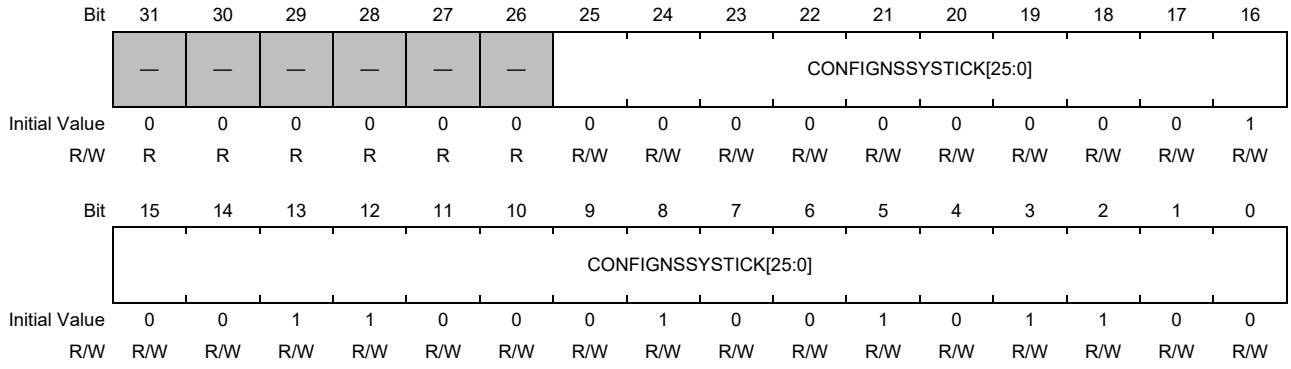
This register is the register that set the secure Systick of Cortex-M33_FPU.



Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25 to 0	CONFIGSSYSTICK[25:0]	H'001_312C	R/W	Configuration register for secure Systick. The value must be determined before the cold reset of Cortex-M33_FPU is released. See the Cortex-M33_FPU chapter for more information on settings.

6.3.57 CM33_FPU Config Register1 (SYS_CM33FPU_CFG1)

This register is the register that set cortex-M33_FPU non-secure Systick.



Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25 to 0	CONFIGNSSYSTICK[25:0]	H'001_312C	R/W	Configuration register for non-secure Systick. The value must be determined before the cold reset of Cortex-M33_FPU is released. See the Cortex-M33_FPU chapter for more information on settings.

6.3.58 CM33_FPU Config Register2 (SYS_CM33FPU_CFG2)

This register is the register that set the secure vector address of Cortex-M33_FPU.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INITSVTOR[31:7]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INITSVTOR[31:7]									—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	INITSVTOR [31:7]	H'000_0E 00	R/W	A register that indicates the secure vector address. The address value must be determined before the Cortex-M33_FPU cold reset is released.
6 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

6.3.59 CM33_FPU Config Register3 (SYS_CM33FPU_CFG3)

This register is the register that set the non-secure vector address of Cortex-M33_FPU.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INITNSVTOR[31:7]															
Initial Value	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INITNSVTOR[31:7]									—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	INITNSVTOR [31:7]	H'020_10 00	R/W	A register that indicates the non-secure vector address. The address value must be determined before the Cortex-M33_FPU cold reset is released.
6 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

6.3.60 CM33_FPU Lock Register (SYS_CM33FPU_LOCK)

This register is a register that set the vector-address change permission of Cortex-M33_FPU.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LOCKNSVTOR	LOCKSVTAIRCR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	LOCKNSVTOR	0	R/W	Control the non-secure vector address change permission. 0: Changeable 1: Cannot be changed
0	LOCKSVTAIRCR	0	R/W	Control changing permission of the Secure Vector address, and the PRIS bit and the BFHFNMINS bit of the AIRCR register of Cortex-M33_FPU. 0: Changeable 1: Cannot be changed

6.3.61 LSI Mode Signal Register (SYS_LSI_MODE)

This register indicates the state of the BOOTSELCPU terminal, the MD_BOOT terminal, the DEBUGEN terminal, the MD_CLKS terminal, and the MD_BYPASS terminal.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STAT_SEC_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	STAT_MD_BYPASS	STAT_MD_CLKS	—	—	STAT_DEBUGEN	—	—	STAT_MD_BOOT[2:0]			—	—	—	STAT_BOOTCPUSEL
Initial Value	0	0	—	—	0	0	—	—	0	—	—	—	0	0	0	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	STAT_SEC_EN	—	R	Indicates whether this LSI support the security function or not. 0: Not supported 1: Available
15, 14	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	STAT_MD_BYPASS	—	R	Indicates the terminal state of the external terminal MD_BYPASS. The value is updated at the rising edge of PRST#. If a reset occurs inside the LSI, this register is not updated.
12	STAT_MD_CLKS	—	R	Indicates the terminal state of the external terminal MD_CLKS. The value is updated at the rising edge of PRST#. If a reset occurs inside the LSI, this register is not updated.
11, 10	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	STAT_DEBUGEN	—	R	Indicates the terminal state of the external terminal DEBUGEN. The value is updated at the rising edge of PRST#. If a reset occurs inside the LSI, this register is not updated.
8	—	—	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6 to 4	STAT_MD_BOOT[2:0]	—	R	Indicates the terminal state of the external terminal MD_BOOT[2:0]. The value is updated at the rising edge of PRST#. If a reset occurs inside the LSI, this register is not updated.
3 to 1	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STAT_BOOTCPUSEL	—	R	Indicates the terminal state of the external terminal BOOTCPUSEL. The value is updated at the rising edge of PRST#. If a reset occurs inside the LSI, this register is not updated.

6.3.62 LSI Device ID Register (SYS_DEVID)

This register indicates the product specific device ID.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DEV_ID[31:28]				DEV_ID[27:0]											
Initial Value	—	—	—	—	1	0	0	0	0	1	0	1	1	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEV_ID[27:0]															
Initial Value	0	0	0	0	0	1	0	0	0	1	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	DEV_ID [31:28]	—	R	Indicates the product revision.
27 to 0	DEV_ID[27:0]	H'85E_04 47	R	Indicates the product specific fixed value.

6.3.63 LSI Product Register (SYS_LSI_PRR)

This register indicates information about product options.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	—	0	0	0	—	0	0	0	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	—	0	0	0	—	0	0	0	—	0	0	0	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
23 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
19 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
11 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.

6.3.64 SYS_AOF0

This register is an address offset register for accessing the 34-bit address space from SD ch0/ch1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OFS11_SXSDHI_1				OFS10_SXSDHI_1				OFS01_SXSDHI_1				OFS00_SXSDHI_1			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFS11_SXSDHI_0				OFS10_SXSDHI_0				OFS01_SXSDHI_0				OFS00_SXSDHI_0			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	OFS11_SXSDHI_1	0011b	R/W	Conversion value to address [33:30] when SD ch1 address [31:30] = 11*1
27 to 24	OFS10_SXSDHI_1	0010b	R/W	Conversion value to address [33:30] when SD ch1 address [31:30] = 10*1
23 to 20	OFS01_SXSDHI_1	0001b	R/W	Conversion value to address [33:30] when SD ch1 address [31:30] = 01*1
19 to 16	OFS00_SXSDHI_1	0000b	R/W	Conversion value to address [33:30] when SD ch1 address [31:30] = 00*1
15 to 12	OFS11_SXSDHI_0	0011b	R/W	Conversion value to address [33:30] when SD ch0 address [31:30] = 11*1
11 to 8	OFS10_SXSDHI_0	0010b	R/W	Conversion value to address [33:30] when SD ch0 address [31:30] = 10*1
7 to 4	OFS01_SXSDHI_0	0001b	R/W	Conversion value to address [33:30] when SD ch0 address [31:30] = 01*1
3 to 0	OFS00_SXSDHI_0	0000b	R/W	Conversion value to address [33:30] when SD ch0 address [31:30] = 00*1

Note 1. Master SD ch0/ch1 are the master of 32bit addresses (4 Gbytes space). Depending on the address value (in 1 GB) of the master address [31:30], you can access the address converted to the address space [33:30]. This gives you access to more than 4 Gbytes of space. This register setting is prohibited from changing the setting while the target master is in operation. Please change it while the access on the master side is stopped.

6.3.65 SYS_AOF1

This register is an address offset register for accessing the 34-bit address space from Ether ch0/ch1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OFS11_SXGIGE_1				OFS10_SXGIGE_1				OFS01_SXGIGE_1				OFS00_SXGIGE_1			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFS11_SXGIGE_0				OFS10_SXGIGE_0				OFS01_SXGIGE_0				OFS00_SXGIGE_0			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	OFS11_SXGIGE_1	0011b	R/W	Conversion value to address [33:30] when Ether ch1 address [31:30] = 11*1
27 to 24	OFS10_SXGIGE_1	0010b	R/W	Conversion value to address [33:30] when Ether ch1 address [31:30] = 10*1
23 to 20	OFS01_SXGIGE_1	0001b	R/W	Conversion value to address [33:30] when Ether ch1 address [31:30] = 01*1
19 to 16	OFS00_SXGIGE_1	0000b	R/W	Conversion value to address [33:30] when Ether ch1 address [31:30] = 00*1
15 to 12	OFS11_SXGIGE_0	0011b	R/W	Conversion value to address [33:30] when Ether ch0 address [31:30] = 11*1
11 to 8	OFS10_SXGIGE_0	0010b	R/W	Conversion value to address [33:30] when Ether ch0 address [31:30] = 10*1
7 to 4	OFS01_SXGIGE_0	0001b	R/W	Conversion value to address [33:30] when Ether ch0 address [31:30] = 01*1
3 to 0	OFS00_SXGIGE_0	0000b	R/W	Conversion value to address [33:30] when Ether ch0 address [31:30] = 00*1

Note 1. Master Ether ch0/ch1 are the master of 32bit addresses (4 Gbytes space). Depending on the address value (in 1 GB) of the master address [31:30], you can access the address converted to the address space [33:30]. This gives you access to more than 4 Gbytes of space. This register setting is prohibited from changing the setting while the target master is in operation. Please change it while the access on the master side is stopped.

6.3.66 SYS_AOF2

This register is an address offset register for accessing the 34-bit address space from USB2.0 ch0/ch1 Host.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OFS11_SXUSB2_1				OFS10_SXUSB2_1				OFS01_SXUSB2_1				OFS00_SXUSB2_1			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFS11_SXUSB2_0_H				OFS10_SXUSB2_0_H				OFS01_SXUSB2_0_H				OFS00_SXUSB2_0_H			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	OFS11_SXUSB2_1	0011b	R/W	Conversion value to address [33:30] when USB2.0 ch1 Host address [31:30] = 11*1
27 to 24	OFS10_SXUSB2_1	0010b	R/W	Conversion value to address [33:30] when USB2.0 ch1 Host address [31:30] = 10*1
23 to 20	OFS01_SXUSB2_1	0001b	R/W	Conversion value to address [33:30] when USB2.0 ch1 Host address [31:30] = 01*1
19 to 16	OFS00_SXUSB2_1	0000b	R/W	Conversion value to address [33:30] when USB2.0 ch1 Host address [31:30] = 00*1
15 to 12	OFS11_SXUSB2_0_H	0011b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Host address [31:30] = 11*1
11 to 8	OFS10_SXUSB2_0_H	0010b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Host address [31:30] = 10*1
7 to 4	OFS01_SXUSB2_0_H	0001b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Host address [31:30] = 01*1
3 to 0	OFS00_SXUSB2_0_H	0000b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Host address [31:30] = 00*1

Note 1. Master USB2.0 ch0/ch1 Host are the master of 32bit addresses (4 Gbytes space). Depending on the address value (in 1 GB) of the master address [31:30], you can access the address converted to the address space [33:30]. This gives you access to more than 4 Gbytes of space. This register setting is prohibited from changing the setting while the target master is in operation. Please change it while the access on the master side is stopped.

6.3.67 SYS_AOF3

This register is an address offset register for accessing the 34-bit address space from USB2.0 ch0 Function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFS11_SXUSB2_0_F				OFS10_SXUSB2_0_F				OFS01_SXUSB2_0_F				OFS00_SXUSB2_0_F			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 12	OFS11_SXUSB2_0_F	0011b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Function address [31:30] = 11*1
11 to 8	OFS10_SXUSB2_0_F	0010b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Function address [31:30] = 10*1
7 to 4	OFS01_SXUSB2_0_F	0001b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Function address [31:30] = 01*1
3 to 0	OFS00_SXUSB2_0_F	0000b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Function address [31:30] = 00*1

Note 1. Master USB2.0 ch0 Function is the master of 32bit addresses (4 Gbytes space). Depending on the address value (in 1 GB) of the master address [31:30], you can access the address converted to the address space [33:30]. This gives you access to more than 4 Gbytes of space. This register setting is prohibited from changing the setting while the target master is in operation. Please change it while the access on the master side is stopped.

6.3.68 SYS_AOF6

This register is an address offset register for accessing the 34-bit address space from Secure/Non-Secure DMAC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OFS11_SXDMAC_NS				OFS10_SXDMAC_NS				OFS01_SXDMAC_NS				OFS00_SXDMAC_NS			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFS11_SXDMAC_S				OFS10_SXDMAC_S				OFS01_SXDMAC_S				OFS00_SXDMAC_S			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	OFS11_SXDMAC_NS	0011b	R/W	Conversion value to address [33:30] when Non-Secure DMAC address [31:30] = 11*1
27 to 24	OFS10_SXDMAC_NS	0010b	R/W	Conversion value to address [33:30] when Non-Secure DMAC address [31:30] = 10*1
23 to 20	OFS01_SXDMAC_NS	0001b	R/W	Conversion value to address [33:30] when Non-Secure DMAC address [31:30] = 01*1
19 to 16	OFS00_SXDMAC_NS	0000b	R/W	Conversion value to address [33:30] when Non-Secure DMAC address [31:30] = 00*1
15 to 12	OFS11_SXDMAC_S	0011b	R/W	Conversion value to address [33:30] when Secure DMAC address [31:30] = 11*1
11 to 8	OFS10_SXDMAC_S	0010b	R/W	Conversion value to address [33:30] when Secure DMAC address [31:30] = 10*1
7 to 4	OFS01_SXDMAC_S	0001b	R/W	Conversion value to address [33:30] when Secure DMAC address [31:30] = 01*1
3 to 0	OFS00_SXDMAC_S	0000b	R/W	Conversion value to address [33:30] when Secure DMAC address [31:30] = 00*1

Note 1. Master Secure/Non-Secure DMAC are the master of 32bit addresses (4 Gbytes space). Depending on the address value (in 1 GB) of the master address [31:30], you can access the address converted to the address space [33:30]. This gives you access to more than 4 Gbytes of space. This register setting is prohibited from changing the setting while the target master is in operation. Please change it while the access on the master side is stopped.

6.3.69 SYS_AOF9

This register is an address offset register for accessing the 34-bit address space from SDHI ch2.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFS11_SXSDHI_2				OFS10_SXSDHI_2				OFS01_SXSDHI_2				OFS00_SXSDHI_2			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 12	OFS11_SXSDHI_2	0011b	R/W	Conversion value to address [33:30] when SD ch2 address [31:30] = 11*1
11 to 8	OFS10_SXSDHI_2	0010b	R/W	Conversion value to address [33:30] when SD ch2 address [31:30] = 10*1
7 to 4	OFS01_SXSDHI_2	0001b	R/W	Conversion value to address [33:30] when SD ch2 address [31:30] = 01*1
3 to 0	OFS00_SXSDHI_2	0000b	R/W	Conversion value to address [33:30] when SD ch2 address [31:30] = 00*1

Note 1. Master SD ch2 is the master of 32bit addresses (4 Gbytes space). Depending on the address value (in 1 GB) of the master address [31:30], you can access the address converted to the address space [33:30]. This gives you access to more than 4 Gbytes of space. This register setting is prohibited from changing the setting while the target master is in operation. Please change it while the access on the master side is stopped.

6.3.70 SYS_LP_CTL1

This register is a register that requests the transition to low power consumption mode and confirms the status.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CM33FPUSLEEP_ACK	CM33SLEEP_ACK	—	—	—	CA55SLEEP_ACK	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CM33FPUSLEEP_REQ	CM33SLEEP_REQ	—	—	—	CA55SLEEP_REQ	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29	CM33FPUSLEEP_ACK	0b	R/W	Cortex-M33_FPU Sleep Mode transition response 0: There is no response of transition to Cortex-M33_FPU Sleep Mode 1: There is a response of transition to Cortex-M33_FPU Sleep Mode
28	CM33SLEEP_ACK	0b	R/W	Cortex-M33 Sleep Mode transition response 0: There is no response of transition to Cortex-M33 Sleep Mode 1: There is a response of transition to Cortex-M33 Sleep Mode
27, 26	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
24	CA55SLEEP_ACK	0b	R/W	Cortex-A55 Sleep Mode transition response 0: There is no response of transition to Cortex-A55 Sleep Mode 1: There is a response of transition to Cortex-A55 Core0 Sleep Mode
23 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
16	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
15, 14	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	CM33FPUSLEEP_REQ	0b	R/W	Cortex-M33_FPU Sleep Mode transition request 0: There is no transition request to Cortex-M33_FPU Sleep Mode 1: There is a transition request to Cortex-M33_FPU Sleep Mode
12	CM33SLEEP_REQ	0b	R/W	Cortex-M33 Sleep Mode transition request 0: There is no transition request to Cortex-M33 Sleep Mode 1: There is a transition request to Cortex-M33 Sleep Mode
11, 10	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
9	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
8	CA55SLEEP_REQ	0b	R/W	Cortex-A55 Sleep Mode transition request 0: There is no transition request to Cortex-A55 Sleep Mode 1: There is a transition request to Cortex-A55 Core0 Sleep Mode
7 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.71 SYS_LP_CTL2

This register is set before and after Cortex-A55 Sleep Mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CA55_STBYCTL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CA55_STBY CTL	0b	R/W	Registers to be set before and after cortex-A55 Sleep Mode. 0: Cortex-A55 Sleep Mode operation complete. When using in other modes, be sure to 0. 1: Cortex-A55 Sleep Mode start

6.3.72 SYS_LP_CTL5

This register controls the low power mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CM33FPUSLEEP_F	CM33SLEEP_F	—	CA55SLEEP0_F	—	—	—	—	—	AMCLKQDENY_F	ASCLKQDENY_F	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R	R/W	R	R	R	R	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
19 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	CM33FPUSLEEP_F	0b	R/W	Cortex-M33_FPU Sleep Mode request flag 0: No Cortex-M33_FPU Sleep Mode request (clear with 0 write) 1: Cortex-M33_FPU Sleep Mode request (SYS_LPM_INT interrupt occurs)
10	CM33SLEEP_F	0b	R/W	Cortex-M33 Sleep Mode request flag 0: No Cortex-M33 Sleep Mode request (clear with 0 write) 1: Cortex-M33 Sleep Mode request (SYS_LPM_INT interrupt occurs)
9	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
8	CA55SLEEP0_F	0b	R/W	Cortex-A55 core0 Sleep Mode request flag 0: No Cortex-A55 core0 Sleep Mode request (clear with 0 write) 1: Cortex-A55 core0 Sleep Mode request (SYS_LPM_INT interrupt occurs)
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	AMCLKQDENY_F	0b	R/W	Cortex-A55 ACE Asynchronous Bridge Master Interface QDENY Factor AMCLKQDENY Flag 0: No Deny response (clear with 0 write) 1: Deny response (SYS_CA55_DENY interrupt occurs)

Bit	Bit Name	Initial Value	R/W	Description
1	ASCLKQDENY_F	0b	R/W	Cortex-A55 ACE Asynchronous Bridge Slave Interface QDENY Factor ASCLKQDENY Flag 0: No Deny response (clear with 0 write) 1: Deny response (SYS_CA55_DENY interrupt occurs)
0	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.73 SYS_LP_CTL6

This register controls the low power mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CM33FPUSLEEP_E	CM33SLEEP_E	—	CA55SLEEP0_E	—	—	—	—	—	AMCLKQDENY_E	ASCLKQDENY_E	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R	R/W	R	R	R	R	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	CM33FPUSLEEP_E	0b	R/W	Cortex-M33_FPU Sleep Mode Requirements Mask 0: Mask Cortex-M33_FPU Sleep Mode requirements factor 1: Don’t mask Cortex-M33_FPU Sleep Mode requirements factor
10	CM33SLEEP_E	0b	R/W	Cortex-M33 Sleep Mode Requirements Mask 0: Mask Cortex-M33 Sleep Mode requirements factor 1: Don’t mask Cortex-M33 Sleep Mode requirements factor
9	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
8	CA55SLEEP0_E	0b	R/W	Cortex-A55 core0 Sleep Mode Requirements Mask 0: Mask Cortex-A55 core0 Sleep Mode requirements factor 1: Don’t mask Cortex-A55 core0 Sleep Mode requirements factor
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
5	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
4	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	AMCLKQDENY_E	0b	R/W	Cortex-A55 ACE Asynchronous Bridge Master Interface QDENY Configuration Factor Mask 0: Masks the configuration factors AMCLKQDENY_F for cortex-A55 ACE asynchronous bridge master interface QDENY 1: Do not mask the configuration factor AMCLKQDENY_F for Cortex-A55 ACE Asynchronous Bridge Master Interface QDENY

Bit	Bit Name	Initial Value	R/W	Description
1	ASCLKQDENY_E	0b	R/W	Cortex-A55 ACE Asynchronous Bridge Slave Interface QDENY Configuration Factor Mask 0: Masks the configuration factors ASCLKQDENY_F for cortex-A55 ACE asynchronous bridge slave interface QDENY 1: Do not mask the configuration factor ASCLKQDENY_F for Cortex-A55 ACE Asynchronous bridge slave Interface QDENY
0	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.74 SYS_LP_CTL7

This register controls the low power mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IM33FPU_MASK	IM33_MASK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	IM33FPU_MASK	0b	R/W	IM33FPU to Cortex-M33_FPU Interrupt Mask Settings Set this bit before entering Cortex-M33_FPU Sleep Mode (Deep Sleep), Cortex-M33_FPU Warm Reset. The interrupt mask setting is automatically removed by the hardware. 0: Do not mask interrupts from IM33FPU to Cortex-M33_FPU. 1: Mask interrupts from IM33FPU to Cortex-M33_FPU.
0	IM33_MASK	0b	R/W	IM33 to Cortex-M33 Interrupt Mask Settings Set this bit before entering Cortex-M33 Sleep Mode (Deep Sleep), Cortex-M33 Warm Reset. The interrupt mask setting is automatically removed by the hardware. 0: Do not mask interrupts from IM33 to Cortex-M33. 1: Mask interrupts from IM33 to Cortex-M33.

6.3.75 SYS_LP_CM33CTL0

This register controls the low power mode of Cortex-M33.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	—	0	0	0	0	0	0	0	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SYSRESETREQ	—	—	—	—	SLEEPDEEP	—	—	—	SLEEPMODE
Initial Value	0	0	0	0	0	0	—	—	0	0	0	—	0	0	0	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
24	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
16	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	SYSRESETREQ	—	R	Warm reset request from Cortex-M33 0: No request 1: Requested
8	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
7 to 5	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	SLEEPDEEP	—	R	Cortex-M33 SLEEPDEEP status 0: Cortex-M33 is not in the Sleep Mode 1: Cortex-M33 is either in the Sleep Mode For more information, see **Section 3, System CPU Cortex-M33/Cortex-M33_FPU**.
3 to 1	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SLEEPMODE	—	R	Cortex-M33 Sleep mode status 0: Not in the Sleep Mode. 1: In the Sleep Mode Asserted in multiple modes. For more information, see **Section 3, System CPU Cortex-M33/Cortex-M33_FPU**.

6.3.76 SYS_LP_CA55CK_CTL1

This register controls the low power mode of Cortex-A55.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PDBGCLKQACTIVE	GICCLKQACTIVE	ATCLKQACTIVE	PCLKQACTIVE	—	—	—	—	—	AMCLKQACTIVE	ASCLKQACTIVE	—
Initial Value	0	0	0	0	—	—	—	—	0	0	0	0	0	—	—	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	PDBGCLKQACTIVE	—	R	Represents the status of a QACTIVE signal from the PDBGCLK domain of Cortex-A55 cluster.*1
10	GICCLKQACTIVE	—	R	Represents the status of a QACTIVE signal from the GICCLK domain of Cortex-A55 cluster.*1
9	ATCLKQACTIVE	—	R	Represents the status of a QACTIVE signal from the ATCLK domain of Cortex-A55 cluster.*1
8	PCLKQACTIVE	—	R	Represents the status of a QACTIVE signal from the PCLK domain of Cortex-A55 cluster.*1
7 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	AMCLKQACTIVE	—	R	Represents the state of the QACTIVE signal from cortex-A55 ACE asynchronous bridge Master Interface.*1
1	ASCLKQACTIVE	—	R	Represents the state of the QACTIVE signal from the Cortex-A55 ACE asynchronous bridge Slave Interface.*1
0	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.

Note 1. The state of this register is used to determine whether subsequent Q-channel controls will start clock stop operations. For more information about QACTIVE signal specifications, see AMBA® Low Power Interface ARM® Q-Channel and P-Channel Interfaces.

6.3.77 SYS_LP_CA55CK_CTL2

This register controls the low power mode of Cortex-A55.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PDBGCLKQREQn	GICCLKQREQn	ATCLKQREQn	PCLKQREQn	—	—	—	—	—	AMCLKQREQn	ASCLKQREQn	—
Initial Value	0	0	0	0	1	1	1	1	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	PDBGCLKQREQn	1b	R/W	Set the level of the QREQn signal to the PDBGCLK domain of the Cortex-A55 cluster. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with PDBGCLKQACCEPTn and PDBGCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
10	GICCLKQREQn	1b	R/W	Set the level of the QREQn signal to the GICCLK domain of the Cortex-A55 cluster. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with GICCLKQACCEPTn and GICCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
9	ATCLKQREQn	1b	R/W	Set the level of the QREQn signal to the ATCLK domain of the Cortex-A55 cluster. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with ATCLKQACCEPTn and ATCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
8	PCLKQREQn	1b	R/W	Set the level of the QREQn signal to the PCLK domain of the Cortex-A55 cluster. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with PCLKQACCEPTn and PCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
7 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	AMCLKQREQn	1b	R/W	Set the level of the QREQn signal to the ACE asynchronous bridge Master Interface on cortex-A55. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with AMCLKQACCEPTn and AMCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.

Bit	Bit Name	Initial Value	R/W	Description
1	ASCLKQREQn	1b	R/W	Set the level of the QREQn signal to the ACE asynchronous bridge Slave Interface on cortex-A55. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with ASCLKQACCEPTn and ASCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
0	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.78 SYS_LP_CA55CK_CTL3

This register controls the low power mode of Cortex-A55.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	PDBGCLKQDENY	GICCLKQDENY	ATCLKQDENY	PCLKQDENY	—	—	—	—	—	AMCLKQDENY	ASCLKQDENY	—
Initial Value	0	0	0	0	—	—	—	—	0	0	0	0	0	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PDBGCLKQACCEPTn	GICCLKQACCEPTn	ATCLKQACCEPTn	PCLKQACCEPTn	—	—	—	—	—	AMCLKQACCEPTn	ASCLKQACCEPTn	CA55_COREINSTRRUN[0]
Initial Value	0	0	0	0	—	—	—	—	0	0	0	0	0	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	0000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27	PDBGCLKQDENY	—	R	Represents the status of the QDENY signal from the PDBGCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with PDBGCLKQREQn and PDBGCLKQACCEPTn. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
26	GICCLKQDENY	—	R	Represents the status of the QDENY signal from the GICCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with GICCLKQREQn and GICCLKQACCEPTn. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
25	ATCLKQDENY	—	R	Represents the status of the QDENY signal from the ATCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with ATCLKQREQn and ATCLKQACCEPTn. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
24	PCLKQDENY	—	R	Represents the status of the QDENY signal from the PCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with PCLKQREQn and PCLKQACCEPTn. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
23 to 19	—	00000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	AMCLKQDENY	—	R	Represents the status of the QDENY signal from the ACE asynchronous bridge Master Interface on cortex-A55. This register represents the state of the Q-channel in combination with AMCLKQREQn and AMCLKQACCEPTn. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.

Bit	Bit Name	Initial Value	R/W	Description
17	ASCLKQDENY	—	R	Represents the status of the QDENY signal from the ACE asynchronous bridge Slave Interface on cortex-A55. This register represents the state of the Q-channel in combination with ASCLKQREQn and ASCLKQACCEPTn. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
16	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
15 to 12	—	0000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	PDBGCLKQACCEPTn	—	R	Represents the state of the QACCEPTn signal from the PDBGCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with PDBGCLKQREQn and PDBGCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
10	GICCLKQACCEPTn	—	R	Represents the state of the QACCEPTn signal from the GICCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with GICCLKQREQn and GICCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
9	ATCLKQACCEPTn	—	R	Represents the state of the QACCEPTn signal from the ATCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with ATCLKQREQn and ATCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
8	PCLKQACCEPTn	—	R	Represents the state of the QACCEPTn signal from the PCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with PCLKQREQn and PCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
7 to 3	—	00000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	AMCLKQACCEPTn	—	R	Represents the state of the QACCEPTn signal from the ACE asynchronous bridge Master Interface on cortex-A55. This register represents the state of the Q-channel in combination with AMCLKQREQn and AMCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
1	ASCLKQACCEPTn	—	R	Represents the state of the QACCEPTn signal from the ACE asynchronous bridge Slave Interface on cortex-A55. This register represents the state of the Q-channel in combination with ASCLKQREQn and ASCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
0	CA55_COREINSTRRUN[0]	—	R	Indicates that Cortex-A55 Core0 has transitioned to the WFI/WFE state. 0: Transitioned or OFF state 1: Normal operation

6.3.79 SYS_LP_CM33FPUCTL0

This register controls the low power mode of Cortex-M33_FPU.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	—	0	0	0	0	0	0	0	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SYSRESETREQ	—	—	—	—	SLEEPDEEP	—	—	—	SLEEPMODE
Initial Value	0	0	0	0	0	0	—	—	0	0	0	—	0	0	0	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
24	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
16	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	SYSRESETREQ	—	R	Warm reset request from Cortex-M33_FPU 0: No request 1: Requested
8	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
7 to 5	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	SLEEPDEEP	—	R	Cortex-M33_FPU SLEEPDEEP status 0: Cortex-M33_FPU is not in the Sleep Mode 1: Cortex-M33_FPU is either in the Sleep Mode For more information, see **Section 3, System CPU Cortex-M33/Cortex-M33_FPU**.
3 to 1	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SLEEPMODE	—	R	Cortex-M33_FPU Sleep mode status 0: Not in the Sleep Mode. 1: In the Sleep Mode Asserted in multiple modes. For more information, see **Section 3, System CPU Cortex-M33/Cortex-M33_FPU**.

6.3.80 Isolation Cell Control Register (SYS_PD_ISO_CTRL)

This register is isolation cell in the PD_ISOVCC region control register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PD_ISOVCC_ISOEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PD_ISOVCC_ISOEN	0b	R/W	Controls isolation cell in the PD_ISOVCC region 0: Isolation Cell OFF (PD_ISOVCC power supply) 1: Isolation Cell ON (PD_ISOVCC power off)

6.3.81 DDRPHY Control Register (PWRDN_DDRPHY_CTRL)

This register is DDR PHY control register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DDRPHY_CTRL3	DDRPJHY_CTRL3_EN	—	—	DDRPHY_CTRL2	DDRPHY_CTRL2_EN	—	—	DDRPHY_CTRL1	DDRPHY_CTRL1_EN
Initial Value	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	DDRPHY_CTRL3	1b	R/W	DDRPHY retention mode control3. Set this bit to 1 when DDRPHY Retention mode. 0: Normal operation 1: DDRPHY Retention mode
8	DDRPHY_CTRL3_EN	0b	R/W	DDRPHY retention mode control3 enable. When this bit =1, command of DDRPHY_CTRL3 is enable. 0: Command of DDRPHY_CTRL3 is not enable. 1: Command of DDRPHY_CTRL3 is enable.
7, 6	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	DDRPHY_CTRL2	0b	R/W	DDRPHY retention mode control2. Set this bit to 0 when DDRPHY Retention mode. 0: DDRPHY Retention mode 1: Normal operation
4	DDRPHY_CTRL2_EN	0b	R/W	DDRPHY retention mode control2 enable. When this bit =1, command of DDRPHY_CTRL2 is enable. 0: Command of DDRPHY_CTRL2 is not enable. 1: Command of DDRPHY_CTRL2 is enable.
3, 2	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	DDRPHY_CTRL1	0b	R/W	DDRPHY retention mode control1. Set this bit to 0 when DDRPHY Retention mode. 0: DDRPHY Retention mode 1: Normal operation
0	DDRPHY_CTRL1_EN	0b	R/W	DDRPHY retention mode control1 enable. When this bit =1, command of DDRPHY_CTRL1 is enable. 0: Command of DDRPHY_CTRL1 is not enable. 1: Command of DDRPHY_CTRL1 is enable.

6.3.82 Isolation Region IO Buffer SE18 Control Register (ISO_IOBUF_SE18_CTRL)

This register is SE18 of IO buffer in the isolation region control register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ISO_IOBUF_SE18
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	ISO_IOBUF_SE18	1b	R/W	Controls SE18 of IO buffer in the isolation region 0: Controls SE18=0 1: Controls SE18=1

6.3.83 USB PWRRDY Register (SYS_USB_PWRRDY)

This register is PWERRDY terminal of USB control register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PWRRDY_N
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PWRRDY_N	0b	R/W	Controls PWRRDY terminal of USB 0: PWRRDY 1: PWRRDY down When turning off the USB region power, set this bit to 1. When turning on the USB region power, set this bit to 0

6.3.84 PCIe RST_RSM_B Terminal Control Register (SYS_PCIE_RST_RSM_B)

This register is RST_RSM_B terminal of PCIe control register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PCIE_RST_RSM_B
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PCIE_RST_RSM_B	0b	R/W	Controls RST_RSM_B terminal of PCIe 0: RST_RSM_B=0 1: RST_RSM_B=1 Set RST_RSM_B=1 after PCIe power is applied. When the power in the PCIe region is turned off, set RST_RSM_B=0 before turning off the power supply.

6.3.85 General Register0 (SYS_GPREG_0)

This register is a general-purpose register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GPREG0[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPREG0[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GPREG0[31:0]	H'0000_0000	R/W	General-purpose register 0 It is a 32-bit register. It is possible to write any value, read the written value.

6.3.86 General Register1 (SYS_GPREG_1)

This register is a general-purpose register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GPREG1[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPREG1[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GPREG1 [31:0]	H'0000_0000	R/W	General-purpose register 1 It is a 32-bit register. It is possible to write any value, read the written value.

6.3.87 General Register2 (SYS_GPREG_2)

This register is a general-purpose register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GPREG2[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPREG2[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GPREG2 [31:0]	H'0000_0000	R/W	General-purpose register 2 It is a 32-bit register. It is possible to write any value, read the written value.

6.3.88 General Register3 (SYS_GPREG_3)

This register is a general-purpose register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GPREG3[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPREG3[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GPREG3 [31:0]	H'0000_0000	R/W	General-purpose register 3 It is a 32-bit register. It is possible to write any value, read the written value.

6.3.89 xSPI/Octa Controller Select Register (SYS_IPCONT_SEL_SPI_OCTA)

This register is xSPI/Octa Controller select register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SEL_SPI_OCTA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SEL_SPI_OCTA	0b	R/W	xSPI/Octa Controller select 0: xSPI 1: Octa Controller

6.3.90 Cortex-M33 Address Space Definition Register (SYS_IPCONT_ IDAUZERONS)

This register is Cortex-M33 Address Space definition register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IDAUZERONS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	IDAUZERONS	0b	R/W	Non-Secure area and Secure area definition of Cortex-M33 Address Space 0: Secure (lower), Non-Secure (upper) 1: Secure (upper), Non-Secure (lower)

6.3.91 Cortex-M33_FPU Address Space Definition Register (SYS_IPCONT_ IDAUZERONS_FPU)

This register is Cortex-M33_FPU Address Space definition register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IDAUZERONS_FPU
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	IDAUZERONS_FPU	0b	R/W	Non-Secure area and Secure area definition of Cortex-M33_FPU Address Space 0: Secure (lower), Non-Secure (upper) 1: Secure (upper), Non-Secure (lower)

6.4 Operation

6.4.1 External Terminal State Capture Function

Captures the state of the following five external terminals at the rising edge of PRST# terminal and holds them into the SYS_LSI_MODE register. The value is not updated by a reset that occurs inside the LSI.

- BOOTCPUSEL
- MD_BOOT[2:0]
- DEBUGEN
- MD_CLKS
- MD_BYPASS

6.4.2 WDT Stop Control Function

By writing 1 to the WDTSTOP bit of the SYS_WDTn_CTRL register (n: 0, 1, 2), the following WDT corresponding to n is forcibly stopped and the count value is retained.

Also, by writing 1 to the WDTSTOPMASK bit of the SYS_WDTn_CTRL register (n: 0, 1, 2), the following WDT count corresponding to n can be continued even during debugging. If 0 is written, the WDT count will stop during debugging.

n	Stop WDT
0	WDT0 (for Cortex-A55 core0)
1	WDT1 (for Cortex-M33)
2	WDT2 (for Cortex-M33_FPU)

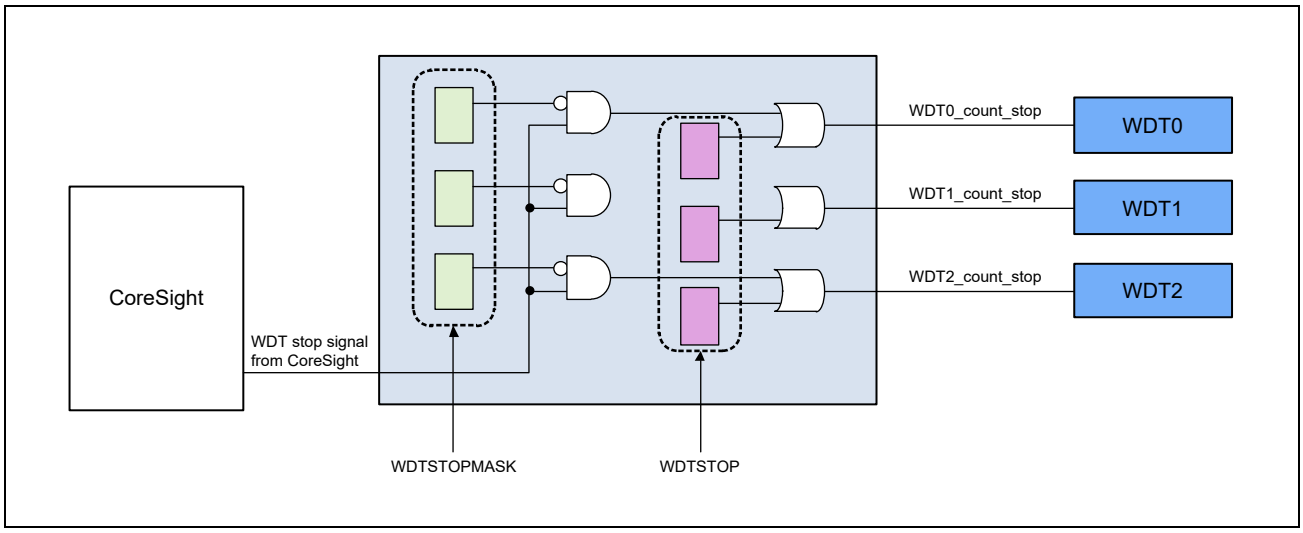


Figure 6.2 Overview of WDT Stop Function

7. Clock Pulse Generator (CPG)

This chapter describes the functions of the clock pulse generator module (CPG).

7.1 Features

7.1.1 Overview

Table 7.1 provides a list of CPG features.

Table 7.1 List of Functions

Features	Specification
PLL control	• SSCG control, PLL multiplication factor setting, and ON/OFF control are performed by register setting. • This LSI is equipped with 5 PLLs. The roles of each are as follows: – PLL1 (SSCG-PLL: Default SSCG ON): Cortex-A55 only – PLL2 (SSCG-PLL: Default SSCG OFF): For SYSTEM-BUS & Module (no SSCG) – PLL3 (SSCG-PLL: Default SSCG ON): For SYSTEM-BUS & Module (SSCG Yes) – PLL4 (SSCG-PLL: Default SSCG ON): DDR-SDRAM Only – PLL6 (SSCG-PLL: Default SSCG OFF): Cortex-M33 / Gigabit Ethernet Interface • Initial value of SSCG for each PLL (PLL3 is fixed): – Fmod: 30.0 (PLL1), 30.3 kHz (PLL3, PLL4 and PLL6) – Spread: Down spread – Modulation Depth: −2.00% (PLL1 and PLL6), −1.00% (PLL4)
Clock generation and control	The clock to be supplied to each module is generated from the external clock input, crystal oscillator input/output (XIN/XOUT), or PLL output clock. • Selection of the peri-ratio by register setting • Clock path selection by clock selector by register setting • Clock supply ON/OFF control is performed by register setting. (Module Standby Mode control) • Cortex-M33 Sleep Mode control • Software Standby Mode control
Reset generation and control	A reset is generated from the reset factor in (). The types of reset are as follows. • System reset (external pin) • Software reset (system reboot, reset control of each module) • Watchdog reset (WDT module) • Cortex-A55 Warm reset control (software control) • Cortex-M33 Warm reset control (software control) • ON/OFF control for each unit clock (software control)
Monitor	• PLL clock monitor – The function to monitor the operation of the output clocks from PLL1 to PLL4 and PLL6 • Clock monitor for each module – Function to monitor that the clock monitor is operating • Reset monitor – Function to monitor the reset status of each module

7.1.2 CPG Configuration

Figure 7.1 shows the CPG configuration.

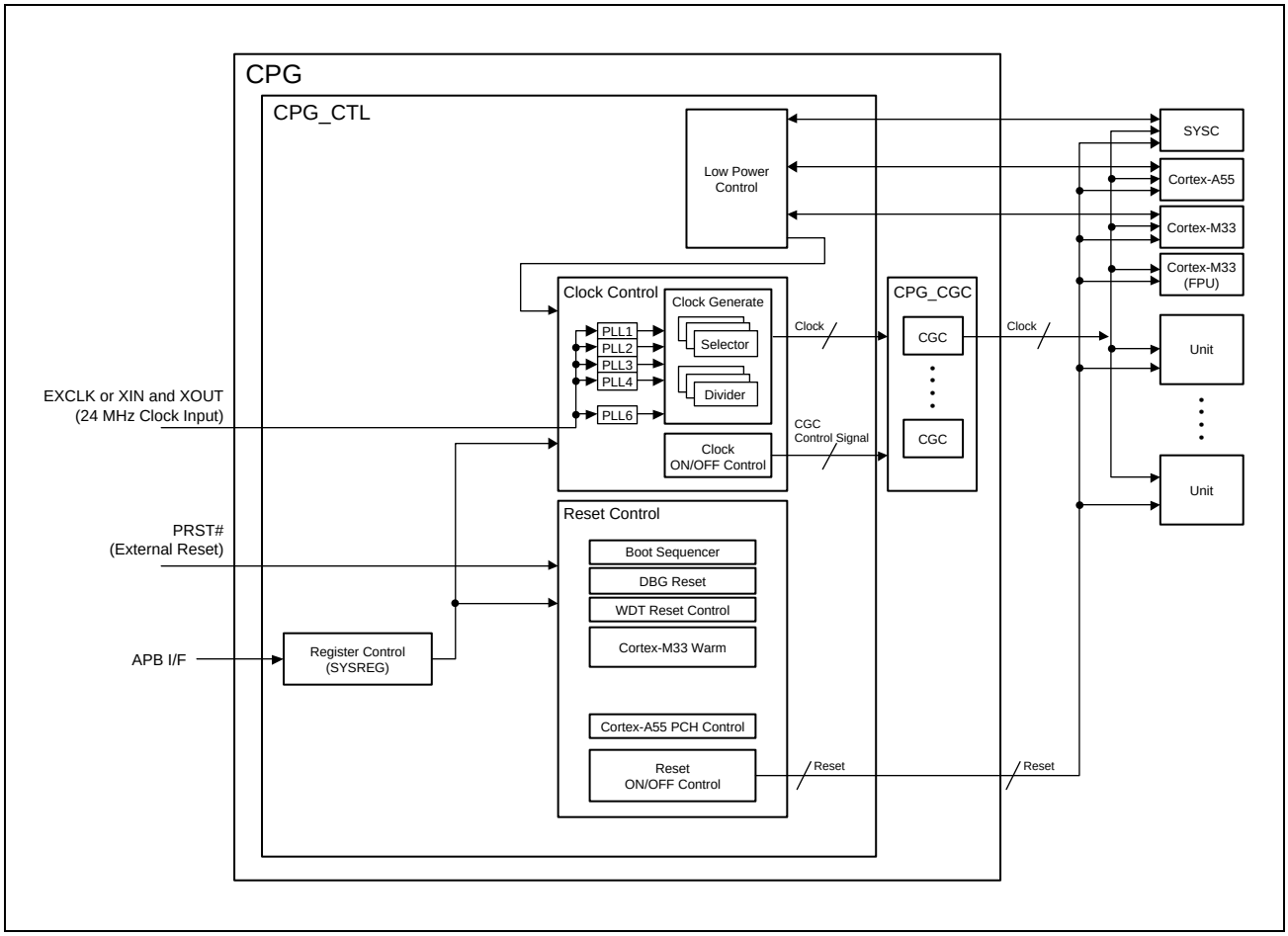


Figure 7.1 CPG Configuration

7.1.3 Pin Description

The table below shows a list of CPG pins.

Table 7.2 Pin List

Name	Pin Name	IO	Functions
Mode control pin	MD_BOOT0	I	Set the operating mode. Do not change during PRST# pin asserts and after negating until the operating mode is confirmed.
	MD_BOOT1	I	
	MD_BOOT2	I	
	MD_CLKS	I	Set ON / OFF of SSCG circuit operation of PLL3. Do not change during PRST# terminal asserts and after negating until the operating mode is confirmed.
OSC Clock Switching Pin	MD_BYPASS	I	This pin is used to switch between the crystal oscillator and the external clock. 0: crystal oscillator 1: external clock (bypass) Do not change during PRST# pin asserts and after negating until the operating mode is confirmed.
Crystal oscillator input/output pin	XIN	I	This is the pin to connect the crystal oscillator.
	XOUT	O	
Power on reset input pin	PRST#	I	When this pin becomes low level, it will be in the power on reset state.

7.1.4 Clock

7.1.4.1 Clock System Diagram

Figure 7.2 and **Figure 7.3** show the clock system diagram.

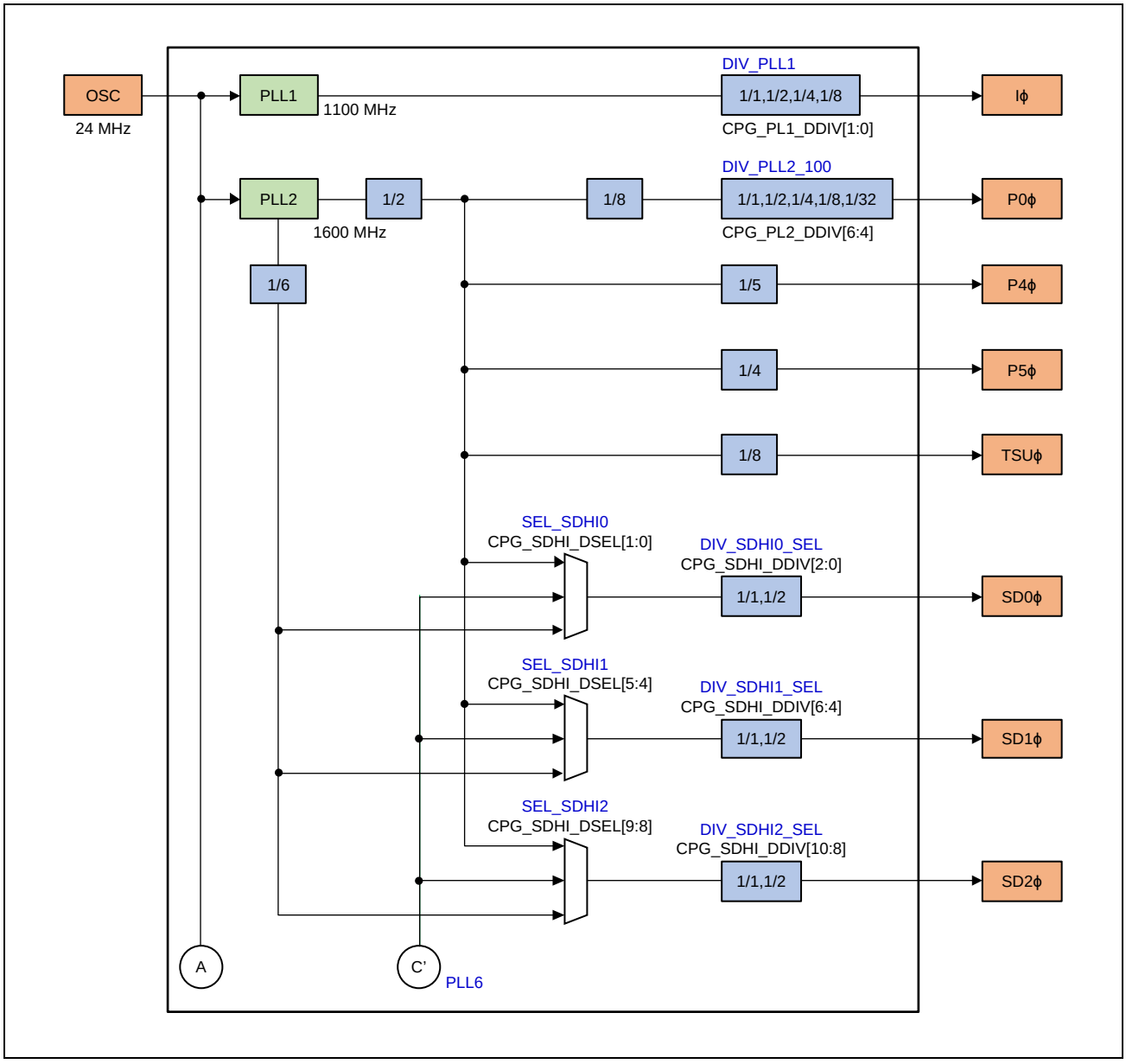


Figure 7.2 Clock System Diagram (1)

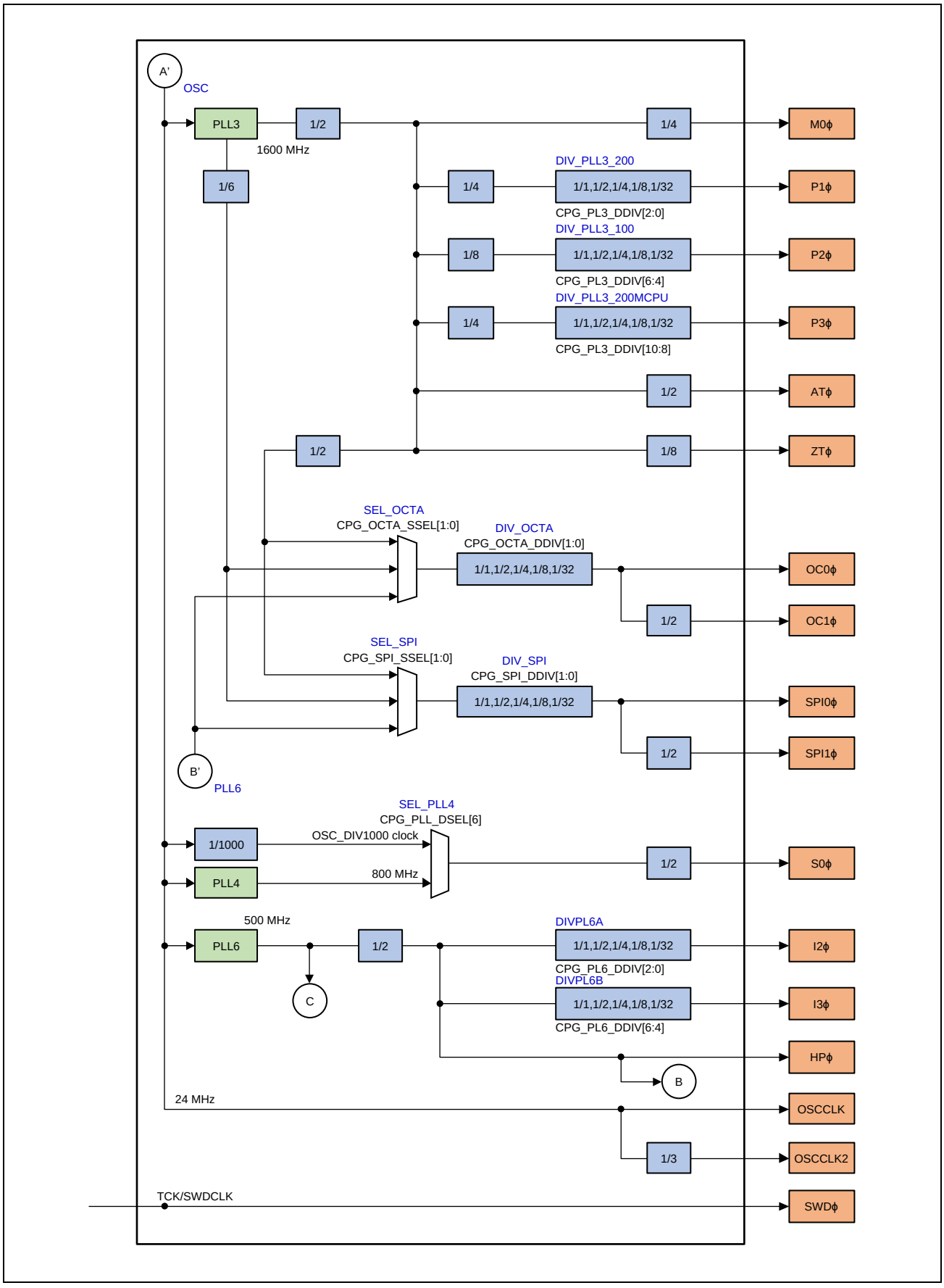


Figure 7.3 Clock System Diagram (2)

7.1.4.2 Clock List

Please refer to another file for the clock list.

7.2 Register Description

7.2.1 Register Attributes

The following table shows the register attributes described in Register Details.

Table 7.3 Register Attributes

Register Attributes	Description
RW	A register that can be read and written.
RW0	A register that can read and 0b write. 1b Write is invalid.
RW1	A register that can read and 1b write. 0b Write is invalid.
R0W	It is a register that can be written with 0b Read at all times.
R1W	It is a register that can be written with 1b Read at all times.
R	A register that can read. Write is invalid.
R0W0	It is a register that can always read 0b and write 0b. 1b Write is invalid.
R0W1	It is a register that can always read 0b and write 1b. 0b Write is invalid.
R1W0	It is a register that can always read 1b and write 0b. 1b Write is invalid.
R1W1	It is a register that can always read 1b and write 1b. 0b Write is invalid.
RCW0	A register that can be cleared by Read and 0b Write is possible. 1b Write is invalid.
RCW1	A register that can be cleared by Read and 1b Write is possible. 0b Write is invalid.

Note: This LSI does not guarantee the access result to the Reserved bit area and the undefined bit area.
At the same time, the Reserved bit area and undefined area may have values other than the initial value “0”. We do not guarantee the result of changing these values.

7.2.2 Register Classification

The table below shows the general classification of the register space in this module.

The address of the CPG register is represented by an offset address from the CPG-based address.

Base Address: H'0_1101_0000 (Cortex-A55 Address Space)
Base Address: H'4101_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
Base Address: H'5101_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 7.4 Register Address Space

Classification	Offset Address
SSCG PLL Control Registers (PLL1, 2, 3, 4, 6)	H'000 - H'0FF
OTP Monitor Registers	H'100 - H'17F
Reserved	H'180 - H'1FF
Frequency Dynamic Change Control Registers	H'200 - H'3FF
Frequency Static Change Control Registers	H'400 - H'4FF
Clock Control Registers	H'500 - H'67F
Clock Monitor Registers	H'680 - H'7FF
Reset Control Registers	H'800 - H'97F
Reset Monitor Registers	H'980 - H'AFF
Other Registers	H'B00 - H'FFF

7.2.3 Register Configuration

Table 7.5 Register List (1/7)

Register Name	Abbreviation	R/W	Initial Value*1	Address*2	Access Size
PLL1 (SSCG) Standby Control Register	CPG_PLL1_STBY	RW	H'0000_0005	H'000	32
PLL4 (SSCG) Standby Control Register	CPG_PLL4_STBY	RW	H'0000_0004	H'030	32
PLL6 (SSCG) Standby Control Register	CPG_PLL6_STBY	RW	H'0000_0001	H'050	32
PLL1 (SSCG) Output Clock Setting Register 1	CPG_PLL1_CLK1	RW	H'0458_E001	H'004	32
PLL4 (SSCG) Output Clock Setting Register 1	CPG_PLL4_CLK1	RW	H'0498_E000	H'034	32
PLL6 (SSCG) Output Clock Setting Register 1	CPG_PLL6_CLK1	RW	H'048F_8000	H'054	32
PLL1 (SSCG) Output Clock Setting Register 2	CPG_PLL1_CLK2	RW	H'0000_0023	H'008	32
PLL4 (SSCG) Output Clock Setting Register 2	CPG_PLL4_CLK2	RW	H'0000_0002	H'038	32
PLL6 (SSCG) Output Clock Setting Register 2	CPG_PLL6_CLK2	RW	H'0000_0003	H'058	32
PLL1 (SSCG) Monitor Register	CPG_PLL1_MON	R	H'0000_0011	H'00C	32
PLL4 (SSCG) Monitor Register	CPG_PLL4_MON	R	H'0000_0000	H'03C	32
PLL6 (SSCG) Monitor Register	CPG_PLL6_MON	R	H'0000_0011	H'05C	32
PLL1_SEL_SETTING Register	CPG_PLL1_SETTING	RW	H'0000_0001	H'100	32
CPG_OTPPLL0_MON Register	CPG_OTPPLL0_MON	R	H'xxxx_xxxx	H'104	32
CPG_OTPPLL1_MON Register	CPG_OTPPLL1_MON	R	H'xxxx_xxxx	H'108	32
CPG_OTPPLL2_MON Register	CPG_OTPPLL2_MON	R	H'xxxx_xxxx	H'10C	32
PLL2(SSCG) Monitor Register	CPG_PLL2_MON	R	H'0000_0011	H'01C	32
PLL3(SSCG) Monitor Register	CPG_PLL3_MON	R	H'0000_0011	H'02C	32
Division Ratio Setting (PLL1) Register	CPG_PL1_DDIV	RW	H'0000_000x	H'200	32
Division Ratio Setting (PLL2) Register	CPG_PL2_DDIV	RW	H'0000_0000	H'204	32
Division Ratio Setting (PLL3) Register	CPG_PL3_DDIV	RW	H'0000_0000	H'208	32
Division Ratio Setting (PLL6) Register	CPG_PL6_DDIV	RW	H'0000_0000	H'214	32
Division Ratio Setting (SDHI) Register	CPG_SDHI_DDIV	RW	H'0000_0111	H'218	32
Division Ratio Setting (OCTA) Register	CPG_OCTA_DDIV	RW	H'0000_0003	H'21C	32
Division Ratio Setting (SPI) Register	CPG_SPI_DDIV	RW	H'0000_0003	H'220	32
Source Clock Setting (PLL) Register	CPG_PLL_DSEL	RW	H'0000_0415	H'240	32
Source Clock Setting (SDHI) Register	CPG_SDHI_DSEL	RW	H'0000_0333	H'244	32
Clock Status Monitor Register	CPG_CLKDIVSTATUS	R	H'0000_0000	H'280	32
Clock Status (Selector Setting) Monitor Register	CPG_CLKSELSTATUS	R	H'0000_0000	H'284	32
Source Clock Setting Register	CPG_OCTA_SSEL	RW	H'0000_0002	H'400	32
Source Clock Setting Register	CPG_SPI_SSEL	RW	H'0000_0002	H'404	32
Clock Control Register Cortex-A55	CPG_CLKON_CA55	RW	H'0000_003F	H'500	32
Clock Control Register Cortex-M33	CPG_CLKON_CM33	RW	H'0000_0000	H'504	32
Clock Control Register ACPU	CPG_CLKON_SRAM_ACPU	RW	H'0000_0007	H'508	32
Clock Control Register MCPU	CPG_CLKON_SRAM_MCPU	RW	H'0000_0003	H'50C	32
Clock Control Register GIC600	CPG_CLKON_GIC600	RW	H'0000_0001	H'514	32
Clock Control Register IA55	CPG_CLKON_IA55	RW	H'0000_0003	H'518	32
Clock Control Register IM33	CPG_CLKON_IM33	RW	H'0000_0303	H'51C	32
Clock Control Register MHU	CPG_CLKON_MHU	RW	H'0000_0000	H'520	32
Clock Control Register CST	CPG_CLKON_CST	RW	H'0000_3FFF	H'524	32
Clock Control Register SYC	CPG_CLKON_SYC	RW	H'0000_0001	H'528	32
Clock Control Register DMAC_REG	CPG_CLKON_DMAC_REG	RW	H'0000_0000	H'52C	32

Table 7.5 Register List (2/7)

Register Name	Abbreviation	R/W	Initial Value*1	Address*2	Access Size
Clock Control Register GTM	CPG_CLKON_GTM	RW	H'0000_0000	H'534	32
Clock Control Register MTU	CPG_CLKON_MTU	RW	H'0000_0000	H'538	32
Clock Control Register POE3	CPG_CLKON_POE3	RW	H'0000_0000	H'53C	32
Clock Control Register GPT	CPG_CLKON_GPT	RW	H'0000_0000	H'540	32
Clock Control Register POEG	CPG_CLKON_POEG	RW	H'0000_0000	H'544	32
Clock Control Register WDT	CPG_CLKON_WDT	RW	H'0000_000F	H'548	32
Clock Control Register DDR	CPG_CLKON_DDR	RW	H'0000_0000	H'54C	32
Clock Control Register SPI	CPG_CLKON_SPI	RW	H'0000_0000	H'550	32
Clock Control Register SDHI	CPG_CLKON_SDHI	RW	H'0000_0000	H'554	32
Clock Control Register SSI	CPG_CLKON_SSI	RW	H'0000_0000	H'570	32
Clock Control Register SRC	CPG_CLKON_SRC	RW	H'0000_0000	H'574	32
Clock Control Register USB	CPG_CLKON_USB	RW	H'0000_0000	H'578	32
Clock Control Register ETH	CPG_CLKON_ETH	RW	H'0000_0000	H'57C	32
Clock Control Register I2C	CPG_CLKON_I2C	RW	H'0000_0000	H'580	32
Clock Control Register SCIF	CPG_CLKON_SCIF	RW	H'0000_0000	H'584	32
Clock Control Register SCI	CPG_CLKON_SCI	RW	H'0000_0000	H'588	32
Clock Control Register IRDA	CPG_CLKON_IRDA	RW	H'0000_0000	H'58C	32
Clock Control Register RSPI	CPG_CLKON_RSPI	RW	H'0000_0000	H'590	32
Clock Control Register CANFD	CPG_CLKON_CANFD	RW	H'0000_0000	H'594	32
Clock Control Register GPIO	CPG_CLKON_GPIO	RW	H'0000_0001	H'598	32
Clock Control Register ADC	CPG_CLKON_ADC	RW	H'0000_0000	H'5A8	32
Clock Control Register TSU	CPG_CLKON_TSU	RW	H'0000_0000	H'5AC	32
Clock Control Register AXI_ACPU_BUS	CPG_CLKON_AXI_ACPU_BUS	RW	H'0000_007F	H'5B4	32
Clock Control Register AXI_MCPU_BUS	CPG_CLKON_AXI_MCPU_BUS	RW	H'0000_DDFF	H'5B8	32
Clock Control Register AXI_COM_BUS	CPG_CLKON_AXI_COM_BUS	RW	H'0000_0303	H'5BC	32
Clock Control Register PERI_COM	CPG_CLKON_PERI_COM	RW	H'0000_0303	H'5C4	32
Clock Control Register REG1_BUS	CPG_CLKON_REG1_BUS	RW	H'0000_0003	H'5C8	32
Clock Control Register REG0_BUS	CPG_CLKON_REG0_BUS	RW	H'0000_000F	H'5CC	32
Clock Control Register PERI_CPU	CPG_CLKON_PERI_CPU	RW	H'0000_00FF	H'5D0	32
Clock Control Register PERI_DDR	CPG_CLKON_PERI_DDR	RW	H'0000_0001	H'5D8	32
Clock Control Register AXI_TZCDDR	CPG_CLKON_AXI_TZCDDR	RW	H'0000_000F	H'5DC	32
Clock Control Register OCTA	CPG_CLKON_OCTA	RW	H'0000_0000	H'5F4	32
Clock Control Register OTFDE_DDR	CPG_CLKON_OTFDE_DDR	RW	H'0000_0000	H'5F8	32
Clock Control Register OTFDE_SPI	CPG_CLKON_OTFDE_SPI	RW	H'0000_0000	H'600	32
Clock Control Register PDM	CPG_CLKON_PDM	RW	H'0000_0000	H'604	32
Clock Control Register PCI	CPG_CLKON_PCI	RW	H'0000_0000	H'608	32
Clock Control Register SPDIF	CPG_CLKON_SPDIF	RW	H'0000_0000	H'60C	32
Clock Control Register I3C	CPG_CLKON_I3C	RW	H'0000_0000	H'610	32
Clock Control Register VBAT	CPG_CLKON_VBAT	RW	H'0000_0000	H'614	32
Clock Monitor Register Cortex-A55	CPG_CLKMON_CA55	R	H'0000_003F	H'680	32
Clock Monitor Register Cortex-M33	CPG_CLKMON_CM33	R	H'0000_0000	H'684	32
Clock Monitor Register SRAM_ACPU	CPG_CLKMON_SRAM_ACPU	R	H'0000_0007	H'688	32

Table 7.5 Register List (3/7)

Register Name	Abbreviation	R/W	Initial Value*1	Address*2	Access Size
Clock Monitor Register SRAM_MCPU	CPG_CLKMON_SRAM_MCPU	R	H'0000_0003	H'68C	32
Clock Monitor Register GIC600	CPG_CLKMON_GIC600	R	H'0000_0001	H'694	32
Clock Monitor Register IA55	CPG_CLKMON_IA55	R	H'0000_0003	H'698	32
Clock Monitor Register IM33	CPG_CLKMON_IM33	R	H'0000_0303	H'69C	32
Clock Monitor Register MHU	CPG_CLKMON_MHU	R	H'0000_0000	H'6A0	32
Clock Monitor Register CST	CPG_CLKMON_CST	R	H'0000_3FFF	H'6A4	32
Clock Monitor Register SYC	CPG_CLKMON_SYC	R	H'0000_0001	H'6A8	32
Clock Monitor Register DMAC_REG	CPG_CLKMON_DMAC_REG	R	H'0000_0000	H'6AC	32
Clock Monitor Register GTM	CPG_CLKMON_GTM	R	H'0000_0000	H'6B4	32
Clock Monitor Register MTU	CPG_CLKMON_MTU	R	H'0000_0000	H'6B8	32
Clock Monitor Register POE3	CPG_CLKMON_POE3	R	H'0000_0000	H'6BC	32
Clock Monitor Register GPT	CPG_CLKMON_GPT	R	H'0000_0000	H'6C0	32
Clock Monitor Register POEG	CPG_CLKMON_POEG	R	H'0000_0000	H'6C4	32
Clock Monitor Register WDT	CPG_CLKMON_WDT	R	H'0000_000F	H'6C8	32
Clock Monitor Register DDR	CPG_CLKMON_DDR	R	H'0000_0000	H'6CC	32
Clock Monitor Register SPI	CPG_CLKMON_SPI	R	H'0000_0000	H'6D0	32
Clock Monitor Register SDHI	CPG_CLKMON_SDHI	R	H'0000_0000	H'6D4	32
Clock Monitor Register SSI	CPG_CLKMON_SSI	R	H'0000_0000	H'6F0	32
Clock Monitor Register SRC	CPG_CLKMON_SRC	R	H'0000_0000	H'6F4	32
Clock Monitor Register USB	CPG_CLKMON_USB	R	H'0000_0000	H'6F8	32
Clock Monitor Register ETH	CPG_CLKMON_ETH	R	H'0000_0000	H'6FC	32
Clock Monitor Register I2C	CPG_CLKMON_I2C	R	H'0000_0000	H'700	32
Clock Monitor Register SCIF	CPG_CLKMON_SCIF	R	H'0000_0000	H'704	32
Clock Monitor Register SCI	CPG_CLKMON_SCI	R	H'0000_0000	H'708	32
Clock Monitor Register IRDA	CPG_CLKMON_IRDA	R	H'0000_0000	H'70C	32
Clock Monitor Register RSPI	CPG_CLKMON_RSPI	R	H'0000_0000	H'710	32
Clock Monitor Register CANFD	CPG_CLKMON_CANFD	R	H'0000_0000	H'714	32
Clock Monitor Register GPIO	CPG_CLKMON_GPIO	R	H'0000_0001	H'718	32
Clock Monitor Register ADC	CPG_CLKMON_ADC	R	H'0000_0000	H'728	32
Clock Monitor Register TSU	CPG_CLKMON_TSU	R	H'0000_0000	H'72C	32
Clock Monitor Register AXI_ACPU_BUS	CPG_CLKMON_AXI_ACPU_BUS	R	H'0000_007F	H'734	32
Clock Monitor Register AXI_MCPU_BUS	CPG_CLKMON_AXI_MCPU_BUS	R	H'0000_DDFF	H'738	32
Clock Monitor Register AXI_COM_BUS	CPG_CLKMON_AXI_COM_BUS	R	H'0000_0303	H'73C	32
Clock Monitor Register PERI_COM	CPG_CLKMON_PERI_COM	R	H'0000_0303	H'744	32
Clock Monitor Register REG1_BUS	CPG_CLKMON_REG1_BUS	R	H'0000_0003	H'748	32
Clock Monitor Register REG0_BUS	CPG_CLKMON_REG0_BUS	R	H'0000_000F	H'74C	32
Clock Monitor Register PERI_CPU	CPG_CLKMON_PERI_CPU	R	H'0000_00FF	H'750	32
Clock Monitor Register PERI_DDR	CPG_CLKMON_PERI_DDR	R	H'0000_0001	H'758	32
Clock Monitor Register AXI_TZCDDR	CPG_CLKMON_AXI_TZCDDR	R	H'0000_000F	H'75C	32
Clock Monitor Register OCTA	CPG_CLKMON_OCTA	R	H'0000_0000	H'774	32

Table 7.5 Register List (4/7)

Register Name	Abbreviation	R/W	Initial Value*1	Address*2	Access Size
Clock Monitor Register OTFDE_DDR	CPG_CLKMON_OTFDE_DDR	R	H'0000_0000	H'778	32
Clock Monitor Register OTFDE_SPI	CPG_CLKMON_OTFDE_SPI	R	H'0000_0000	H'780	32
Clock Monitor Register PDM	CPG_CLKMON_PDM	R	H'0000_0000	H'784	32
Clock Monitor Register PCI	CPG_CLKMON_PCI	R	H'0000_0000	H'788	32
Clock Monitor Register SPDIF	CPG_CLKMON_SPDIF	R	H'0000_0000	H'78C	32
Clock Monitor Register I3C	CPG_CLKMON_I3C	R	H'0000_0000	H'790	32
Clock Monitor Register VBAT	CPG_CLKMON_VBAT	R	H'0000_0000	H'794	32
Reset Control Register Cortex-A55	CPG_RST_CA55	RW	H'0000_1FF5	H'800	32
Reset Control Register Cortex-M33	CPG_RST_CM33	RW	H'0000_0000	H'804	32
Reset Control Register SRAM_ACPU	CPG_RST_SRAM_ACPU	RW	H'0000_0007	H'808	32
Reset Control Register SRAM_MCPU	CPG_RST_SRAM_MCPU	RW	H'0000_0003	H'80C	32
Reset Control Register GIC600	CPG_RST_GIC600	RW	H'0000_0003	H'814	32
Reset Control Register IA55	CPG_RST_IA55	RW	H'0000_0001	H'818	32
Reset Control Register IM33	CPG_RST_IM33	RW	H'0000_0101	H'81C	32
Reset Control Register MHU	CPG_RST_MHU	RW	H'0000_0000	H'820	32
Reset Control Register SYC	CPG_RST_SYC	RW	H'0000_0001	H'828	32
Reset Control Register DMAC	CPG_RST_DMAC	RW	H'0000_0000	H'82C	32
Reset Control Register GTM	CPG_RST_GTM	RW	H'0000_0000	H'834	32
Reset Control Register MTU	CPG_RST_MTU	RW	H'0000_0000	H'838	32
Reset Control Register POE3	CPG_RST_POE3	RW	H'0000_0000	H'83C	32
Reset Control Register GPT	CPG_RST_GPT	RW	H'0000_0000	H'840	32
Reset Control Register POEG	CPG_RST_POEG	RW	H'0000_0000	H'844	32
Reset Control Register WDT	CPG_RST_WDT	RW	H'0000_0003	H'848	32
Reset Control Register DDR	CPG_RST_DDR	RW	H'0000_0000	H'84C	32
Reset Control Register SPI	CPG_RST_SPI	RW	H'0000_0000	H'850	32
Reset Control Register SDHI	CPG_RST_SDHI	RW	H'0000_0000	H'854	32
Reset Control Register SSIF	CPG_RST_SSIF	RW	H'0000_0000	H'870	32
Reset Control Register SRC	CPG_RST_SRC	RW	H'0000_0000	H'874	32
Reset Control Register USB	CPG_RST_USB	RW	H'0000_0000	H'878	32
Reset Control Register ETH	CPG_RST_ETH	RW	H'0000_0000	H'87C	32
Reset Control Register I2C	CPG_RST_I2C	RW	H'0000_0000	H'880	32
Reset Control Register SCIF	CPG_RST_SCIF	RW	H'0000_0000	H'884	32
Reset Control Register SCI	CPG_RST_SCI	RW	H'0000_0000	H'888	32
Reset Control Register IRDA	CPG_RST_IRDA	RW	H'0000_0000	H'88C	32
Reset Control Register RSPI	CPG_RST_RSPI	RW	H'0000_0000	H'890	32
Reset Control Register CANFD	CPG_RST_CANFD	RW	H'0000_0000	H'894	32
Reset Control Register GPIO	CPG_RST_GPIO	RW	H'0000_0007	H'898	32
Reset Control Register ADC	CPG_RST_ADC	RW	H'0000_0000	H'8A8	32
Reset Control Register TSU	CPG_RST_TSU	RW	H'0000_0000	H'8AC	32
Reset Control Register AXI_ACPU_BUS	CPG_RST_AXI_ACPU_BUS	RW	H'0000_0001	H'8B4	32
Reset Control Register AXI_MCPU_BUS	CPG_RST_AXI_MCPU_BUS	RW	H'0000_0001	H'8B8	32
Reset Control Register AXI_COM_BUS	CPG_RST_AXI_COM_BUS	RW	H'0000_0003	H'8BC	32
Reset Control Register PERI_COM	CPG_RST_PERI_COM	RW	H'0000_0003	H'8C4	32
Reset Control Register REG1_BUS	CPG_RST_REG1_BUS	RW	H'0000_0001	H'8C8	32

Table 7.5 Register List (5/7)

Register Name	Abbreviation	R/W	Initial Value*1	Address*2	Access Size
Reset Control Register REG0_BUS	CPG_RST_REG0_BUS	RW	H'0000_0001	H'8CC	32
Reset Control Register PERI_CPU	CPG_RST_PERI_CPU	RW	H'0000_0003	H'8D0	32
Reset Control Register PERI_DDR	CPG_RST_PERI_DDR	RW	H'0000_0001	H'8D8	32
Reset Control Register AXI_TZCDDR	CPG_RST_AXI_TZCDDR	RW	H'0000_000F	H'8DC	32
Reset Control Register OCTA	CPG_RST_OCTA	RW	H'0000_0000	H'8F4	32
Reset Control Register OTFDE_DDR	CPG_RST_OTFDE_DDR	RW	H'0000_0000	H'8F8	32
Reset Control Register OTFDE_SPI	CPG_RST_OTFDE_SPI	RW	H'0000_0000	H'900	32
Reset Control Register PDM	CPG_RST_PDM	RW	H'0000_0000	H'904	32
Reset Control Register PCI	CPG_RST_PCI	RW	H'0000_0000	H'908	32
Reset Control Register SPDIF	CPG_RST_SPDIF	RW	H'0000_0000	H'90C	32
Reset Control Register I3C	CPG_RST_I3C	RW	H'0000_0000	H'910	32
Reset Control Register VBAT	CPG_RST_VBAT	RW	H'0000_0000	H'914	32
Reset Monitor Register Cortex-CA55	CPG_RSTMON_CA55	R	H'0000_000A	H'980	32
Reset Monitor Register Cortex-M33	CPG_RSTMON_CM33	R	H'0000_0707	H'984	32
Reset Monitor Register SRAM_ACPU	CPG_RSTMON_SRAM_ACPU	R	H'0000_0000	H'988	32
Reset Monitor Register SRAM_MCPU	CPG_RSTMON_SRAM_MCPU	R	H'0000_0000	H'98C	32
Reset Monitor Register GIC600	CPG_RSTMON_GIC600	R	H'0000_0000	H'994	32
Reset Monitor Register IA55	CPG_RSTMON_IA55	R	H'0000_0000	H'998	32
Reset Monitor Register IM33	CPG_RSTMON_IM33	R	H'0000_0000	H'99C	32
Reset Monitor Register MHU	CPG_RSTMON_MHU	R	H'0000_0001	H'9A0	32
Reset Monitor Register SYC	CPG_RSTMON_SYC	R	H'0000_0000	H'9A8	32
Reset Monitor Register DMAC	CPG_RSTMON_DMAC	R	H'0000_0003	H'9AC	32
Reset Monitor Register GTM	CPG_RSTMON_GTM	R	H'0000_00FF	H'9B4	32
Reset Monitor Register MTU	CPG_RSTMON_MTU	R	H'0000_0001	H'9B8	32
Reset Monitor Register POE3	CPG_RSTMON_POE3	R	H'0000_0001	H'9BC	32
Reset Monitor Register GPT	CPG_RSTMON_GPT	R	H'0000_0001	H'9C0	32
Reset Monitor Register POEG	CPG_RSTMON_POEG	R	H'0000_000F	H'9C4	32
Reset Monitor Register WDT	CPG_RSTMON_WDT	R	H'0000_000C	H'9C8	32
Reset Monitor Register DDR	CPG_RSTMON_DDR	R	H'0000_01FF	H'9CC	32
Reset Monitor Register SPI	CPG_RSTMON_SPI	R	H'0000_0003	H'9D0	32
Reset Monitor Register SDHI	CPG_RSTMON_SDHI	R	H'0000_0007	H'9D4	32
Reset Monitor Register SSIF	CPG_RSTMON_SSIF	R	H'0000_000F	H'9F0	32
Reset Monitor Register SRC	CPG_RSTMON_SRC	R	H'0000_0001	H'9F4	32
Reset Monitor Register USB	CPG_RSTMON_USB	R	H'0000_000F	H'9F8	32
Reset Monitor Register ETH	CPG_RSTMON_ETH	R	H'0000_0003	H'9FC	32
Reset Monitor Register I2C	CPG_RSTMON_I2C	R	H'0000_000F	H'A00	32
Reset Monitor Register SCIF	CPG_RSTMON_SCIF	R	H'0000_003F	H'A04	32
Reset Monitor Register SCI	CPG_RSTMON_SCI	R	H'0000_0003	H'A08	32
Reset Monitor Register IRDA	CPG_RSTMON_IRDA	R	H'0000_0001	H'A0C	32
Reset Monitor Register RSPI	CPG_RSTMON_RSPI	R	H'0000_001F	H'A10	32
Reset Monitor Register CANFD	CPG_RSTMON_CANFD	R	H'0000_0003	H'A14	32
Reset Monitor Register GPIO	CPG_RSTMON_GPIO	R	H'0000_0000	H'A18	32
Reset Monitor Register ADC	CPG_RSTMON_ADC	R	H'0000_0003	H'A28	32

Table 7.5 Register List (6/7)

Register Name	Abbreviation	R/W	Initial Value*1	Address*2	Access Size
Reset Monitor Register TSU	CPG_RSTMON_TSU	R	H'0000_0003	H'A2C	32
Reset Monitor Register AXI_ACPU_BUS	CPG_RSTMON_AXI_ACPU_BUS	R	H'0000_0000	H'A34	32
Reset Monitor Register AXI_MCPU_BUS	CPG_RSTMON_AXI_MCPU_BUS	R	H'0000_0000	H'A38	32
Reset Monitor Register AXI_COM_BUS	CPG_RSTMON_AXI_COM_BUS	R	H'0000_0000	H'A3C	32
Reset Monitor Register PERI_COM	CPG_RSTMON_PERI_COM	R	H'0000_0000	H'A44	32
Reset Monitor Register REG1_BUS	CPG_RSTMON_REG1_BUS	R	H'0000_0000	H'A48	32
Reset Monitor Register REG0_BUS	CPG_RSTMON_REG0_BUS	R	H'0000_0000	H'A4C	32
Reset Monitor Register PERI_CPU	CPG_RSTMON_PERI_CPU	R	H'0000_0000	H'A50	32
Reset Monitor Register PERI_DDR	CPG_RSTMON_PERI_DDR	R	H'0000_0000	H'A58	32
Reset Monitor Register AXI_TZCDDR	CPG_RSTMON_AXI_TZCDDR	R	H'0000_0010	H'A5C	32
Reset Monitor Register OCTA	CPG_RSTMON_OCTA	R	H'0000_0001	H'A74	32
Reset Monitor Register OTFDE_DDR	CPG_RSTMON_OTFDE_DDR	R	H'0000_0003	H'A78	32
Reset Monitor Register OTFDE_SPI	CPG_RSTMON_OTFDE_SPI	R	H'0000_0003	H'A80	32
Reset Monitor Register PDM	CPG_RSTMON_PDM	R	H'0000_0007	H'A84	32
Reset Monitor Register PCI	CPG_RSTMON_PCI	R	H'0000_007F	H'A88	32
Reset Monitor Register SPDIF	CPG_RSTMON_SPDIF	R	H'0000_0001	H'A8C	32
Reset Monitor Register I3C	CPG_RSTMON_I3C	R	H'0000_0003	H'A90	32
Reset Monitor Register VBAT	CPG_RSTMON_VBAT	R	H'0000_0003	H'A94	32
WDT Overflow System Reset Register	CPG_WDTOVF_RST	RW	H'0000_0000	H'B10	32
WDT Reset Selector Register	CPG_WDTRST_SEL	RW	H'0000_0088	H'B14	32
Cortex-A55 Cluster Power Status Monitor Register	CPG_CLUSTER_PCHMON	R	H'0000_000x	H'B30	32
Cortex-A55 Cluster Power Status Control Register	CPG_CLUSTER_PCHCTL	RW	H'0048_0000	H'B34	32
Cortex-A55 Core0 Power Status Monitor Register	CPG_CORE0_PCHMON	R	H'0000_000x	H'B38	32
Cortex-A55 Core0 Power Status Control Register	CPG_CORE0_PCHCTL	RW	H'0008_0000	H'B3C	32
MSTOP Register ACPU	CPG_BUS_ACPU_MSTOP	RW	H'0000_0000	H'B60	32
MSTOP Register MCPU1	CPG_BUS_MCPU1_MSTOP	RW	H'0000_0000	H'B64	32
MSTOP Register MCPU2	CPG_BUS_MCPU2_MSTOP	RW	H'0000_0000	H'B68	32
MSTOP Register PERI_COM	CPG_BUS_PERI_COM_MSTOP	RW	H'0000_0000	H'B6C	32
MSTOP Register PERI_CPU	CPG_BUS_PERI_CPU_MSTOP	RW	H'0000_0000	H'B70	32
MSTOP Register PERI_DDR	CPG_BUS_PERI_DDR_MSTOP	RW	H'0000_0000	H'B74	32
MSTOP Register REG0	CPG_BUS_REG0_MSTOP	RW	H'0000_0000	H'B7C	32
MSTOP Register REG1	CPG_BUS_REG1_MSTOP	RW	H'0000_0000	H'B80	32
MSTOP Register TZCDDR	CPG_BUS_TZCDDR_MSTOP	RW	H'0000_0000	H'B84	32
MSTOP Register MHU	CPG_MHU_MSTOP	RW	H'0000_0000	H'B88	32
MSTOP Register MCPU3	CPG_BUS_MCPU3_MSTOP	RW	H'0000_0000	H'B90	32
MSTOP Register PERI_CPU2	CPG_BUS_PERI_CPU2_MSTOP	RW	H'0000_0000	H'B94	32
Power Down IP Register 1	CPG_PWRDN_IP1	RW	H'0000_0000	H'BB0	32

Table 7.5 Register List (7/7)

Register Name	Abbreviation	R/W	Initial Value*1	Address*2	Access Size
Power Down IP Register 2	CPG_PWRDN_IP2	RW	H'0000_0000	H'BB4	32
Power Down MSTOP Register	CPG_PWRDN_MSTOP	RW	H'0000_0000	H'BC0	32
Power Down CLKON Register	CPG_PWRDN_CLKON	RW	H'0000_0000	H'BC4	32
Power Down RST Register	CPG_PWRDN_RST	RW	H'0000_0000	H'BC8	32
Return Func 1 Register	CPG_RET_FUNC1	RW	H'0000_0001	H'BD0	32
Return Func 2 Register	CPG_RET_FUNC2	RW	H'0000_0007	H'BD4	32
Return Func 3 Register	CPG_RET_FUNC3	RW	H'0000_000x	H'BD8	32
Other Function Register 2	CPG_OTHERFUNC2_REG	RW	H'0000_0000	H'BEC	32

Note 1. **Initial value**: The reset is released according to the sequence of the system reset, and the initial value is the one immediately after the last reset release in the sequence.

Note 2. **Address**: The offset address from the base address.

7.2.4 Register Descriptions

7.2.4.1 PLLn (SSCG) Standby Control Register (CPG_PLLn_STBY) (n = 1, 4 or 6)

This register is used to control the power-saving mode and standby state and enable or disable the SSCG.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	SSCG_MODE_WEN	—	SSCG_EN_WEN	—	RESETB_WEN
Initial Value n = 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W1	R	R0W1	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SSCG_MODE	—	SSCG_EN	—	RESETB
Initial Value n = 1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
Initial Value n = 4	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Initial Value n = 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	R	RW	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	SSCG_MODE_WEN	0b	R0W1	Flag for enabling the writing to the SSCG_MODE bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	SSCG_EN_WEN	0b	R0W1	Flag for enabling the writing to the SSCG_EN bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	RESETB_WEN	0b	R0W1	Flag for enabling the writing to the RESETB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	SSCG_MODE	0b	RW	SSCG center spread or down spread setting (see the mode setting table below.*1) When writing to these bits, set the SSCG_MODE_WEN bit to 1 at the same time. 0: Down spread 1: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	SSCG_EN	n = 1, 4: 1b n = 6: 0b	RW	SSCG enable or disable setting (see the mode setting table below.*1) When writing to this bit, set the SSCG_EN_WEN bit to 1 at the same time. 0: SSCG is disabled. 1: SSCG is enabled.
1	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RESETB	n = 1, 6: 1b n = 4: 0b	RW	PLL reset setting (see the mode setting table below.*1) When writing to this bit, set the RESETB_WEN bit to 1 at the same time. 0: Reset state 1: Active state (PLL is released from the reset state.)

Note 1. Mode Setting Table

Mode		Bit Name		
		RESETB	SSCG_EN	SSCG_MODE[1:0]
Reset state		0	—	—
SSCG disabled		1	0	—
SSCG enabled	Down spread		1	0b
	Setting prohibited			1b

7.2.4.2 PLLn (SSCG) Output Clock Setting Register 1 (CPG_PLLn_CLK1) (n = 1, 4 or 6)

This register is used to specify the frequency division values k, m, and p for SSCG PLLn (n = 1, 4 or 6).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	DIV_P			DIV_M				DIV_NI					
Initial Value n = 1	0	0	0	0	0	1	0	0	0	1	0	1	1	0	0	0
Initial Value n = 4	0	0	0	0	0	1	0	0	1	0	0	1	1	0	0	0
Initial Value n = 6	0	0	0	0	0	1	0	0	1	0	0	0	1	1	1	1
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DIV_NI			DIV_NF												RANGE SEL
Initial Value n = 1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1
Initial Value n = 4	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved
28 to 26	DIV_P	n = 1: n = 4: n = 6: 001b	RW	Divider P is a bit for configuration.*3 You can set the value of 1, 2, 4, 8, 16 (decimal). 111 to 100: Division Ratio “pr” = 16 011: Division Ratio “pr” = 8 010: Division Ratio “pr” = 4 001: Division Ratio “pr” = 2 000: Division Ratio “pr” = 1
25 to 22	DIV_M	n = 1: 0001b n = 4: n = 6: 0010b	RW	Pre-Divider is a bit for configuration.*3 It can be set within the range of values of 1 to 12 (decimal) and within the range that satisfies the constraints of Note 1 and Note 2. 1111 to 1100: Setting prohibited 1011: Division Value “mr” = 12 1010: Division Value “mr” = 11 ⋮ 0001: Division Value “mr” = 2 0000: Division Value “mr” = 1
21 to 13	DIV_NI	n = 1: n = 4: 0_1100_0111b n = 6: 0_0111_1100b	RW	Main-Divider is a bit for configuration.*3 It can be set within the range of values of 56 to 375 (decimal) and within the range that satisfies the constraints of Note 1 and Note 2. 1_1111_1111 to 1_0111_0111: Setting prohibited 1_0111_0110: Division Value “nir” = 375 1_0111_0101: Division Value “nir” = 374 ⋮ 0_0011_1000: Division Value “nir” = 57 0_0011_0111: Division Value “nir” = 56 0_0011_0110 to 0_0000_0000: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
12 to 1	DIV_NF	All 0	RW	Delta-Signa Modulator(DSM) is a bit for configuration.*3 It can be set within the range of values of 0 to 4095 (decimal) and within the range that satisfies the constraint of Note 1. 1111_1111_1111: Division Value “nfr” = 4095 1111_1111_1110: Division Value “nfr” = 4094 ⋮ 0000_0000_0001: Division Value “nfr” = 1 0000_0000_0000: Division Value “nfr” = 0
0	RANGESEL	n = 1: 1b n = 4: n = 6: 0b	RW	This register is used to configure the Delta-Signa Modulator (DSM).*3 It can be set within the range of values of 0 to 4095 (decimal) and within the range that satisfies the constraint of Note 1. 0: 900 MHz ≤ DCO ≤ 2000 MHz 1: 2000 MHz ≤ DCO ≤ 3000 MHz

Note 1. The PLL oscillation frequency (fdco) and output frequency (fout) can be calculated by the following equations.

- Equations (fstd = Input frequency)

nr = nir + nfr / 4096

fdco = (nr × fstd) / mr

fout = (nr × fstd) / (mr×pr)

- Restrictions

8 MHz ≤	*fstd*	≤ 192 MHz
8 MHz ≤	*fstd / mr*	≤ 16 MHz
56.25 ≤	*nr*	≤ 375
0.293 ≤	*nr / (mr × pr)*	≤ 375
900 MHz ≤	*fdco*	≤ 2,000 MHz @RANGESEL = 0
2,000 MHz ≤	*fdco*	≤ 3,000 MHz @RANGESEL = 1

Note 2. Specify DIV_P, DIV_M, DIV_NI, DIV_NF, and RANGESEL within the allowable ranges of fdco and fout for each PLL shown below.

SSCG PLL	fout Min.	Default fout Setting	fout Max.
PLL1	1,100 MHz	1,100 MHz	1,100 MHz
PLL4	400 MHz	800 MHz	800 MHz
PLL6	500 MHz	500 MHz	500 MHz

Note 3. The PLL should be reset after the PLL settings are modified. For details, refer to **Section 7.4.4, Procedures for PLL Setup**.

7.2.4.3 PLLn (SSCG) Output Clock Setting Register 2 (CPG_PLLn_CLK2) (n = 1, 4 or 6)

This register is used to specify the frequency division value s and SSCG modulation values mfr and mrr for SSCG PLLn (n = 1, 4 or 6).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value n = 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SELMFREQ					SELMPERCENT		
Initial Value n = 1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Initial Value n = 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
Initial Value n = 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 3	SELMFREQ	n = 1: 0_0100b n = 4: n = 6: All 0	RW	This is a bit for setting the modulation frequency. If you change the settings, you will need to reset the PLL. For details, refer to **Section 7.4.4, Procedures for PLL Setup**. Specify a value from 0 to 31 (decimal).*1 1_1111: SELMFREQ = 31 1_1110: SELMFREQ = 30 ⋮ 0_0001: SELMFREQ = 1 0_0000: SELMFREQ = 0
2 to 0	SELMPERCENT	n = 1: n = 6: 11b n = 4: 10b	RW	This is a bit for setting the modulation rate. If you change the settings, you will need to reset the PLL. For details, refer to **Section 7.4.4, Procedures for PLL Setup**. Specify a value from 0 to 4 (decimal).*1 111 to 101: Setting prohibited 100: “SELMPERCENT” = 4 ⋮ 001: “SELMPERCENT” = 1 000: “SELMPERCENT” = 0

Note 1. The modulation frequency (MF) and modulation ratio (pk-pk) (MR) can be calculated by the following equations.

- Equations

  MF = PFD input frequency / (Divided ratio × 4)

  Note: PFD input frequency (fpfd): Input frequency (fstd)/mr

- Modulation frequency

SELMFREQ [4:0]	Divided ratio	PFD input frequency (fpfd) [MHz]							
		8	9	10	11	12	13	14	15
00000	66	30.30	34.09	37.88	41.67	45.45	49.24	53.03	56.82
00001	73	27.40	30.82	34.25	37.67	41.10	44.52	47.95	51.37
00010	81	24.69	27.78	30.86	33.95	37.04	40.12	43.21	46.30
00011	90	22.22	25.00	27.78	30.56	33.33	36.11	38.89	41.67
00100	100	20.00	22.50	25.00	27.50	30.00	32.50	35.00	37.50
00101	111	18.02	20.27	22.52	24.77	27.03	29.28	31.53	33.78
00110	122	16.39	18.44	20.49	22.54	24.59	26.64	28.69	30.74
00111	133	15.04	16.92	18.80	20.68	22.56	24.44	26.32	28.20
01000	145	13.79	15.52	17.24	18.97	20.69	22.41	24.14	25.86
01001	159	12.58	14.15	15.72	17.30	18.87	20.44	22.01	23.58
01010	175	11.43	12.86	14.29	15.71	17.14	18.57	20.00	21.43
01011	193	10.36	11.66	12.95	14.25	15.54	16.84	18.13	19.43
01100	213	9.39	10.56	11.74	12.91	14.08	15.26	16.43	17.61
01101	236	8.47	9.53	10.59	11.65	12.71	13.77	14.83	15.89
01110	261	7.66	8.62	9.58	10.54	11.49	12.45	13.41	14.37
01111	290	6.90	7.76	8.62	9.48	10.34	11.21	12.07	12.93
10000	322	6.21	6.99	7.76	8.54	9.32	10.09	10.87	11.65
10001	357	5.60	6.30	7.00	7.70	8.40	9.10	9.80	10.50
10010	396	5.05	5.68	6.31	6.94	7.58	8.21	8.84	9.47
10011	440	4.55	5.11	5.68	6.25	6.82	7.39	7.95	8.52
10100	488	4.10	4.61	5.12	5.64	6.15	6.66	7.17	7.68
10101	542	3.69	4.15	4.61	5.07	5.54	6.00	6.46	6.92
10110	600	3.33	3.75	4.17	4.58	5.00	5.42	5.83	6.25
10111	666	3.00	3.38	3.75	4.13	4.50	4.88	5.26	5.63
11000	750	2.67	3.00	3.33	3.67	4.00	4.33	4.67	5.00
11001	840	2.38	2.68	2.98	3.27	3.57	3.87	4.17	4.46
11010	920	2.17	2.45	2.72	2.99	3.26	3.53	3.80	4.08
11011	1000	2.00	2.25	2.50	2.75	3.00	3.25	3.50	3.75
11100	1166	1.72	1.93	2.14	2.36	2.57	2.79	3.00	3.22
11101	1333	1.50	1.69	1.88	2.06	2.25	2.44	2.63	2.81
11110	1500	1.33	1.50	1.67	1.83	2.00	2.17	2.33	2.50
11111	1666	1.20	1.35	1.50	1.65	1.80	1.95	2.10	2.25

- Modulation ratio

SELMPERCENT [2:0]	down spread
000	−0.25%
001	−0.5%
010	−1.0%
011	−2.0%
100	−3.0%
101	Invalid
110	Invalid
111	Invalid

Important:

For the procedures for setting up the PLL, see **Section 7.4.4, Procedures for PLL Setup**.

Initial Values:

The initial values of the registers for PLL1, PLL4 and PLL6 and output clocks are shown below.

- Initial values of registers

	CPG_PLLn_STBY (n = 1, 4 or 6)		
	SSCG_MODE	SSCG_EN	RESETB
PLL	Bit 4	Bit 2	Bit 0
PLL1	H'0 (Down spread)	1 (SSCG is enabled)	1 (Released from the reset state)
PLL4	H'0 (Down spread)	1 (SSCG is enabled)	1 (Reset state)
PLL6	H'0 (Down spread)	0 (SSCG is disabled)	1 (Reset state)

	CPG_PLLn_CLK1 (n = 1, 4 or 6)					CPG_PLLn_CLK2 (n = 1, 4 or 6)	
	DIV_P	DIV_M	DIV_NI	DIV_NF	RANGESEL	SELMFREQ	SELMPERCENT
PLL	Bits 28 to 26	Bits 25 to 22	Bits 21 to 13	Bits 12 to 1	Bit 0	Bits 7 to 3	Bits 2 to 0
PLL1	H'1 (pr = 2)	H'1 (mr = 2)	H'0C7 (nir = 200)	H'000 (nfr = 0)	1	H'04	H'3
PLL4	H'1 (pr = 2)	H'2 (mr = 3)	H'0C7 (nir = 200)	H'000 (nfr = 0)	0	H'00	H'2
PLL6	H'1 (pr = 2)	H'2 (mr = 3)	H'07C (nir = 125)	H'000 (nfr = 0)	0	H'00	H'3

- Output clocks (initial values)

PLL	PLL Operating Mode	Frequency (fout) [MHz]	SSCG	Fmod [kHz]	Spread	Modulation Depth [%]
PLL1	Active state	1,100	Enabled	30.0	Down spread	−2.00
PLL4	Reset state	800	Enabled	30.3	Down spread	−1.00
PLL6	Active state	500	Disabled	30.3	Down spread	−2.00

7.2.4.4 PLLn (SSCG) Monitor Register (CPG_PLLn_MON) (n = 1, 2, 3, 4 or 6)

This register is used to monitor the state of the SSCG PLLn (n = 1, 2, 3, 4 or 6).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value n = 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PLLn_LOCK	—	—	—	PLLn_RESETB
Initial Value n = 1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
Initial Value n = 2	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
Initial Value n = 3	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
Initial Value n = 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 6	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	PLLn_LOCK	n = 1: 1b n = 2: 1b n = 3: 1b n = 4: 0b n = 6: 1b	R	SSCG PLL lock state monitoring 0: PLL is not locked. 1: PLL is locked.
3 to 1	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PLLn_RESETB	n = 1: 1b n = 2: 1b n = 3: 1b n = 4: 0b n = 6: 1b	R	SSCG PLL operating mode monitoring 1: Normal mode 0: Reset state (standby mode)

7.2.4.5 PLL1_SEL_SETTING Register (CPG_PLL1_SETTING)

This register is used to specify the frequency of PLL1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SEL_PLL1_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SEL_PLL1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	SEL_PLL1_WEN	0b	R0W1	Flag for enabling the writing to the SEL_PLL1 bit. This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SEL_PLL1	1b	RW	The PLL multiplication factor is specified. 0: PLL1 operates according to the settings in the PLL1 SYSREG of the CPG. 1: When bit 0 of OTP_OTPPLL0 is 1, PLL1 operates according to the settings of OTP_OTPPLL0 to OTP_OTPPLL2. When bit 0 of OTP_OTPPLL0 is 0, PLL1 operates according to the settings in the PLL1 SYSREG of the CPG.

7.2.4.6 CPG_OTPPLL0_MON Register (CPG_OTPPLL0_MON)

This register indicates the state of the OTP_OTPPLL0[31:0] pins of the CPG.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OTP31_MON	OTP30_MON	OTP29_MON	OTP28_MON	OTP27_MON	OTP26_MON	OTP25_MON	OTP24_MON	OTP23_MON	OTP22_MON	OTP21_MON	OTP20_MON	OTP19_MON	OTP18_MON	OTP17_MON	OTP16_MON
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OTP15_MON	OTP14_MON	OTP13_MON	OTP12_MON	OTP11_MON	OTP10_MON	OTP9_MON	OTP8_MON	OTP7_MON	OTP6_MON	OTP5_MON	OTP4_MON	OTP3_MON	OTP2_MON	OTP1_MON	OTP0_MON
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	OTPn_MON [n = 31 to 0]	—	R	These bits indicate the state of the OTP_OTPPLL0[31:0] pins of the CPG. *Note*: The values specified in the OTP are set in these bits as the initial values.

7.2.4.7 CPG_OTPPLL1_MON Register (CPG_OTPPLL1_MON)

This register indicates the state of the OTP_OTPPLL1[31:0] pins of the CPG.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OTP31_MON	OTP30_MON	OTP29_MON	OTP28_MON	OTP27_MON	OTP26_MON	OTP25_MON	OTP24_MON	OTP23_MON	OTP22_MON	OTP21_MON	OTP20_MON	OTP19_MON	OTP18_MON	OTP17_MON	OTP16_MON
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OTP15_MON	OTP14_MON	OTP13_MON	OTP12_MON	OTP11_MON	OTP10_MON	OTP9_MON	OTP8_MON	OTP7_MON	OTP6_MON	OTP5_MON	OTP4_MON	OTP3_MON	OTP2_MON	OTP1_MON	OTP0_MON
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	OTPn_MON [n = 31 to 0]	—	R	These bits indicate the state of the OTP_OTPPLL1[31:0] pins of the CPG. *Note*: The values specified in the OTP are set in these bits as the initial values.

7.2.4.8 CPG_OTPPLL2_MON Register (CPG_OTPPLL2_MON)

This register indicates the state of the OTP_OTPPLL2[31:0] pins of the CPG.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OTP31_MON	OTP30_MON	OTP29_MON	OTP28_MON	OTP27_MON	OTP26_MON	OTP25_MON	OTP24_MON	OTP23_MON	OTP22_MON	OTP21_MON	OTP20_MON	OTP19_MON	OTP18_MON	OTP17_MON	OTP16_MON
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OTP15_MON	OTP14_MON	OTP13_MON	OTP12_MON	OTP11_MON	OTP10_MON	OTP9_MON	OTP8_MON	OTP7_MON	OTP6_MON	OTP5_MON	OTP4_MON	OTP3_MON	OTP2_MON	OTP1_MON	OTP0_MON
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	OTPn_MON [n = 31 to 0]	—	R	These bits indicate the state of the OTP_OTPPLL2[31:0] pins of the CPG. *Note*: The values specified in the OTP are set in these bits as the initial values.

7.2.4.9 Division Ratio Setting (PLL1) Register (CPG_PL1_DDIV)

This register is used to set the division ratio of the clock for the Cortex-A55.

The PLL1 (1,100 MHz) output is used as the source clock.

The setting of this register can be dynamically modified while the clock is operating.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DIV_PLL1SET_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DIVPL1_SET	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	These bits are always read as 0. Writing to these bits is ignored.
16	DIV_PLL1SET_WEN	0b	R0W1	Flag for enabling the writing to the DIVPL1_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	These bits are always read as 0. Writing to these bits is ignored.
1, 0	DIVPL1_SET	—	RW	Iϕ (DIV_PLL1) division ratio setting 00b: 1/1 (1,100 MHz) 01b: 1/2 (550 MHz) 10b: 1/4 (275 MHz) 11b: 1/8 (137.5 MHz)

7.2.4.10 Division Ratio Setting (PLL2) Register (CPG_PL2_DDIV)

This register is used to set the division ratios of the clocks for the system bus and the IP modules that cannot use SSCG clocks.

The PLL2 (up to 1,600 MHz) output is used as the source clock.

The setting of this register can be dynamically modified while the clock is operating.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	DIV_PLL2_B_WEN	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W1	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	DIVPL2B_SET			—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	R	R	R	R

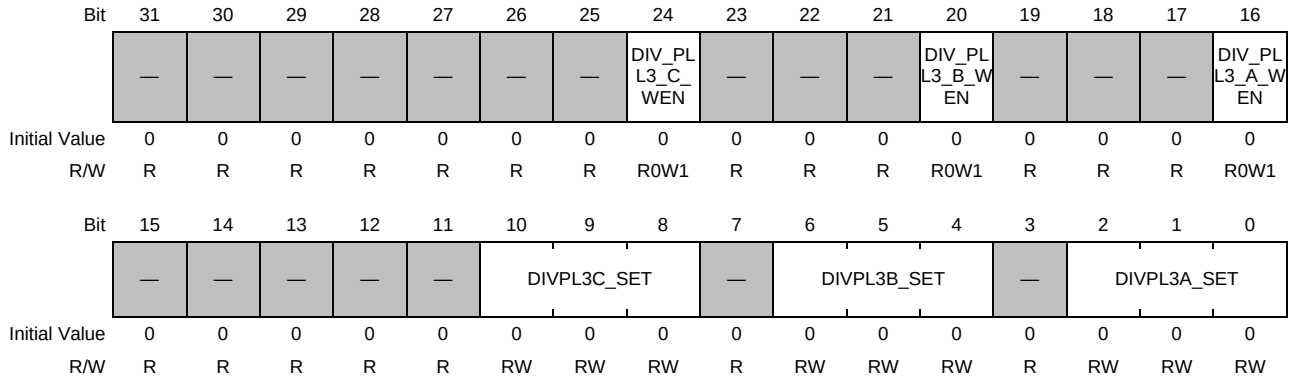
Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	DIV_PLL2_B_WEN	0b	R0W1	Flag for enabling the writing to the DIVPL2B_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19 to 17	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
15 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6 to 4	DIVPL2B_SET	000b	RW	P0ϕ (DIV_PLL2_100) division ratio setting 000b: 1/1 (100 MHz) 001b: 1/2 (50 MHz) 010b: 1/4 (25 MHz) 011b: 1/8 (12.5 MHz) 100b: 1/32 (3.125 MHz) Others: Setting prohibited
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	—	000b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

7.2.4.11 Division Ratio Setting (PLL3) Register (CPG_PL3_DDIV)

This register is used to set the division ratios of the clocks for the system bus and the IP modules that can use SSCG clocks.

The clock obtained by dividing the output of PLL3 is used as the source clock.

The setting of this register can be dynamically modified while the clock is operating.



Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	DIV_PLL3_C_WEN	0b	R0W1	Flag for enabling the writing to the DIVPL3C_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23 to 21	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	DIV_PLL3_B_WEN	0b	R0W1	Flag for enabling the writing to the DIVPL3B_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19 to 17	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	DIV_PLL3_A_WEN	0b	R0W1	Flag for enabling the writing to the DIVPL3A_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10 to 8	DIVPL3C_SET	000b	RW	P3ϕ (DIV_PLL3_200MCPU) division ratio setting 000b: 1/1 (200 MHz) 001b: 1/2 (100 MHz) 010b: 1/4 (50 MHz) 011b: 1/8 (25 MHz) 100b: 1/32 (6.25 MHz) Others: Setting prohibited
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
6 to 4	DIVPL3B_SET	000b	RW	P2ϕ(DIV_PLL3_100)division ratio setting 000b: 1/1 (100 MHz) 001b: 1/2 (50 MHz) 010b: 1/4 (25 MHz) 011b: 1/8 (12.5 MHz) 100b: 1/32 (3.125 MHz) Others: Setting prohibited
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	DIVPL3A_SET	000b	RW	P1ϕ(DIV_PLL3_200) division ratio setting 000b: 1/1 (200 MHz) 001b: 1/2 (100 MHz) 010b: 1/4 (50 MHz) 011b: 1/8 (25 MHz) 100b: 1/32 (6.25 MHz) Others: Setting prohibited

7.2.4.12 Division Ratio Setting (PLL6) Register (CPG_PL6_DDIV)

This register is used to set the division ratio of the clock for the Cortex-M33.

1/2 of the PLL6 (500 MHz) output is used as the source clock.

The setting of this register can be dynamically modified while the clock is operating.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	DIV_PLL6_B_WEN	—	—	—	DIV_PLL6_A_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W1	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	DIVPL6B_SET			—	DIVPL6A_SET		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	DIV_PLL6_B_WEN	0b	R0W1	Flag for enabling the writing to the DIVPL6B_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19 to 17	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	DIV_PLL6_A_WEN	0b	R0W1	Flag for enabling the writing to the DIVPL6A_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6 to 4	DIVPL6B_SET	000b	RW	I3Φ division ratio setting. 000b: 1/1 (250 MHz) 001b: 1/2 (125 MHz) 010b: 1/4 (62.5 MHz) 011b: 1/8 (31.25 MHz) 100b: 1/32 (7.8125 MHz) Others: Setting prohibited
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	DIVPL6A_SET	000b	RW	I2Φ division ratio setting. 000b: 1/1 (250 MHz) 001b: 1/2 (125 MHz) 010b: 1/4 (62.5 MHz) 011b: 1/8 (31.25 MHz) 100b: 1/32 (7.8125 MHz) Others: Setting prohibited

7.2.4.13 Division Ratio Setting (SDHI) Register (CPG_SDHI_DDIV)

This register is used to set the division ratio of the clock for the SDHI.

SEL_SDHIn (n = 0, 1 and 2) output is used as the source clock.

The setting of this register can be dynamically modified while the clock is operating.

If 800 MHz is selected in the CPG_SDHI_DSEL, 1/1 setting is prohibited.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	DIVSDHI2_WEN	—	—	—	DIVSDHI1_WEN	—	—	—	DIVSDHI0_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R0W1	R	R	R	R0W1	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DIVSDHI2_SET	—	—	—	DIVSDHI1_SET	—	—	—	DIVSDHI0_SET
Initial Value	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1
R/W	R	R	R	R	R	R	R	RW	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	DIVSDHI2_WEN	0b	R0W1	Flag for enabling the writing to the DIVSDHI2_SET bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23 to 21	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	DIVSDHI1_WEN	0b	R0W1	Flag for enabling the writing to the DIVSDHI1_SET bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19 to 17	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	DIVSDHI0_WEN	0b	R0W1	Flag for enabling the writing to the DIVSDHI0_SET bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	DIVSDHI2_SET	1b	RW	DIV_SDHI2 division ratio setting. 0b: 1/1 (If 800 MHz is selected in the CPG_SDHI_DSEL, setting is prohibited.) 1b: 1/2
7 to 5	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	DIVSDHI1_SET	1b	RW	DIV_SDHI1 division ratio setting. 0b: 1/1 (If 800 MHz is selected in the CPG_SDHI_DSEL, setting is prohibited.) 1b: 1/2
3 to 1	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	DIVSDHI0_SET	1b	RW	DIV_SDHI0 division ratio setting. 0b: 1/1 (If 800 MHz is selected in the CPG_SDHI_DSEL, setting is prohibited.) 1b: 1/2

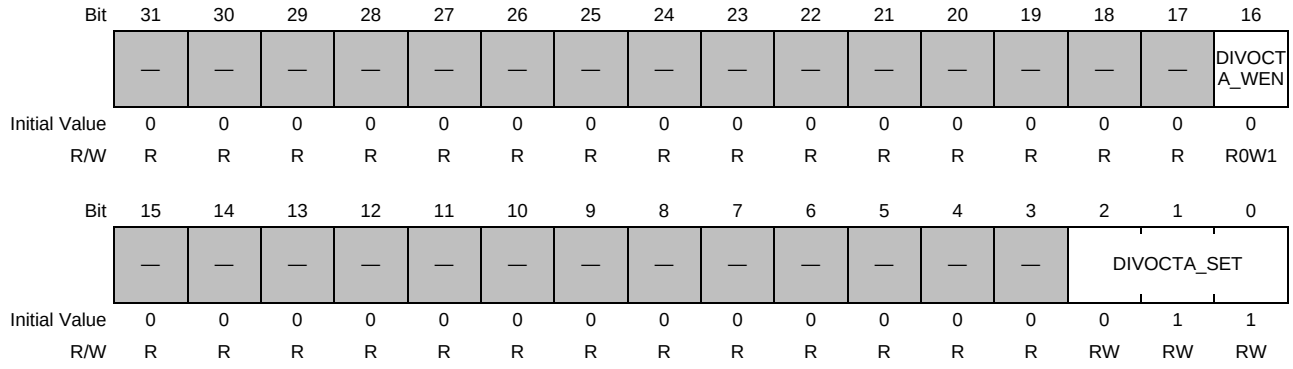
7.2.4.14 Division Ratio Setting (OCTA) Register (CPG_OCTA_DDIV)

This register is used to set the division ratio of the clock for the OCTA.

SEL_OCTA output is used as the source clock.

The setting of this register can be dynamically modified while the clock is operating.

If 400 MHz is selected in the CPG_OCTA_SSEL, 1/1 setting is prohibited.



Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	These bits are always read as 0. Writing to these bits is ignored.
16	DIVOCTA_WEN	0b	R0W1	Flag for enabling the writing to the DIVOCTA_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	These bits are always read as 0. Writing to these bits is ignored.
2 to 0	DIVOCTA_SET	011b	RW	DIV_OCTA (OC0ϕ, OC1ϕ) division ratio setting. 000b: 1/1 (If 400 MHz is selected in the CPG_OCTA_SSEL, setting is prohibited.) 001b: 1/2 010b: 1/4 011b: 1/8 100b: 1/32 Others: Setting prohibited

7.2.4.15 Division Ratio Setting (SPI) Register (CPG_SPI_DDIV)

This register is used to set the division ratio of the clock for the SPI.

SEL_SPI output is used as the source clock.

The setting of this register can be dynamically modified while the clock is operating.

If 400 MHz is selected in the CPG_SPI_SSEL, 1/1 setting is prohibited.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DIVSPI_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DIVSPI_SET		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	These bits are always read as 0. Writing to these bits is ignored.
16	DIVSPI_WEN	0b	R0W1	Flag for enabling the writing to the DIVSPI_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	These bits are always read as 0. Writing to these bits is ignored.
2 to 0	DIVSPI_SET	011b	RW	DIV_SPI (SPI0ϕ, SPI1ϕ) division ratio setting. 000b: 1/1 (If 400 MHz is selected in the CPG_SPI_SSEL, setting is prohibited.) 001b: 1/2 010b: 1/4 011b: 1/8 100b: 1/32 Others: Setting prohibited

7.2.4.16 Source Clock Setting (PLL) Register (CPG_PLL_DSEL)

This register is used to switch the clocks for the PLL.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	SELPL4_WEN	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R0W1	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SELPL4_SET	—	—	—	—	—	—
Initial Value	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	1
R/W	R	R	R	R	R	R	R	R	R	RW	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
22	SEL PL4_WEN	0b	R0W1	Flag for enabling the writing to the SELPL4_SET bit This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21 to 7	—	H'0008	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
6	SELPL4_SET	0b	RW	SEL_PLL4 clock switching setting. 0b: OSC 1/1000th of a divide-by-clock 1b: PLL4 Clock
5 to 0	—	H'15	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

7.2.4.17 Source Clock Setting (SDHI) Register (CPG_SDHI_DSEL)

This register is used to switch the clocks for the SDHI.

The setting of this register can be dynamically modified while the clock is operating.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	SEL_SDHI2_WEN	—	—	—	SEL_SDHI1_WEN	—	—	—	SEL_SDHI0_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R0W1	R	R	R	R0W1	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SEL_SDHI2_SET		—	—	SEL_SDHI1_SET		—	—	SEL_SDHI0_SET	
Initial Value	0	0	0	0	0	0	1	1	0	0	1	1	0	0	1	1
R/W	R	R	R	R	R	R	RW	RW	R	R	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	SEL_SDHI2_WEN	0b	R0W1	Flag for enabling the writing to the SEL_SDHI2_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23 to 21	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	SEL_SDHI1_WEN	0b	R0W1	Flag for enabling the writing to the SEL_SDHI1_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19 to 17	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	SEL_SDHI0_WEN	0b	R0W1	Flag for enabling the writing to the SEL_SDHI0_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9, 8	SEL_SDHI2_SET	11b	RW	SEL_SDHI2 clock switching 00b: 800 MHz (Be sure to set it to 1/2 in the CPG_SDHI_DDIV.) 01b: Setting prohibited 10b: 500 MHz 11b: 266 MHz
7, 6	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5, 4	SEL_SDHI1_SET	11b	RW	SEL_SDHI1 clock switching 00b: 800 MHz (Be sure to set it to 1/2 in the CPG_SDHI_DDIV) 01b: Setting prohibited 10b: 500 MHz 11b: 266 MHz
3, 2	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	SEL_SDHI0_SET	11b	RW	SEL_SDHI0 clock switching 00b: 800 MHz (Be sure to set it to 1/2 in the CPG_SDHI_DDIV) 01b: Setting prohibited 10b: 500 MHz 11b: 266 MHz

Note: When 1 is written to the SEL_SDHI2_WEN, SEL_SDHI1_WEN, or SEL_SDHI0_WEN bit placed in the upper 16 bits of this register, the clock switching control begins even if the settings in the lower bits are not changed. Check the status monitor register and wait until the switching is completed. The clock will temporarily stop at the timing of switching.

7.2.4.18 Clock Status Monitor Register (CPG_CLKDIVSTATUS)

This register is used to monitor whether clock switching is completed in the dynamically modifiable frequency dividers and selectors.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	DIV SPI_STS	DIVOCTA_STS	—	DIVSDHI2_STS	DIVSDHI1_STS	DIVSDHI0_STS	—	—	DIVPL6B_STS	DIVPL6A_STS	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DIVPL3C_STS	DIVPL3B_STS	DIVPL3A_STS	—	—	DIVPL2B_STS	—	—	—	—	DIVPL1_STS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00b	R	Reserved When read, the initial value is read. The written value will be ignored.
29	DIVSPI_STS	0b	R	DIV_SPI clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
28	DIVOCTA_STS	0b	R	DIV_OCTA clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
27	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
26	DIVSDHI2_STS	0b	R	DIV_SDHI2 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
25	DIVSDHI1_STS	0b	R	DIV_SDHI1 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
24	DIVSDHI0_STS	0b	R	DIV_SDHI20 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
23, 22	—	00b	R	Reserved When read, the initial value is read. The written value will be ignored.
21	DIVPL6B_STS	0b	R	DIV_PLL6_CM33_FPU clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
20	DIVPL6A_STS	0b	R	DIV_PLL6_CM33 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
19 to 11	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
10	DIVPL3C_STS	0b	R	DIV_PLL3_200MCPU clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
9	DIVPL3B_STS	0b	R	DIV_PLL3_100 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).

Bit	Bit Name	Initial Value	R/W	Description
8	DIVPL3A_STS	0b	R	DIV_PLL3_200 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
7, 6	—	00b	R	Reserved When read, the initial value is read. The written value will be ignored.
5	DIVPL2B_STS	0b	R	DIV_PLL2_100 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
4	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3 to 1	—	000b	R	Reserved When read, the initial value is read. The written value will be ignored.
0	DIVPL1_STS	0b	R	DIVPL1 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).

7.2.4.19 Clock Status (Selector Setting) Monitor Register (CPG_CLKSELSTATUS)

This is a dynamically modifiable selector clock switching completion status monitor register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	SELSDHI2_STS	SELSDHI1_STS	SELSDHI0_STS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SELPL6_STS	—	—	—	SELPL4_STS	—	SELPL3_STS	—	SELPL2_STS	—	SELPL1_STS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
18	SELSDHI2_STS	0b	R	SEL_SDHI2 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
17	SELSDHI1_STS	0b	R	SEL_SDHI1 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
16	SELSDHI0_STS	0b	R	SEL_SDHI0 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
15 to 11	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
10	SELPL6_STS	0b	R	SEL_PLL6 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
9 to 7	—	000b	R	Reserved When read, the initial value is read. The written value will be ignored.
6	SELPL4_STS	0b	R	SEL_PLL4 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
5	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
4	SELPL3_STS	0b	R	SEL_PLL3 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
3	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
2	SELPL2_STS	0b	R	SEL_PLL2 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
1	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	SEL PL1_STS	0b	R	SEL_PLL1 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).

7.2.4.20 Source Clock Setting Register (CPG_OCTA_SSEL)

This is the OCTA source clock switching register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SELOCTA_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SELOCTA_SET	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	These bits are always read as 0. Writing to these bits is ignored.
16	SELOCTA_WEN	0b	R0W1	Flag for enabling the writing to the SELOCTA_SET bit This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	These bits are always read as 0. Writing to these bits is ignored.
1, 0	SELOCTA_SET	10b	RW	Setting the SEL_OCTA Source Clock 00b: 400 MHz (Be sure to keep it to 1/2 or less in the CPG_OCTA_DDIV) 01b: Setting prohibited 10b: 266 MHz 11b: 250 MHz

7.2.4.21 Source Clock Setting Register (CPG_SPI_SSEL)

This is the SPI source clock switching register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SELSPI_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SELSPI_SET	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	These bits are always read as 0. Writing to these bits is ignored.
16	SELSPI_WEN	0b	R0W1	Flag for enabling the writing to the SELSPI_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	These bits are always read as 0. Writing to these bits is ignored.
1, 0	SELSPI_SET	10b	RW	Setting the SEL_SPI Source Clock 00b: 400 MHz (Be sure to keep it to 1/2 or less in the CPG_SPI_DDIV) 01b: Setting prohibited 10b: 266 MHz 11b: 250 MHz

7.2.4.22 Clock Control Register Cortex-A55 (CPG_CLKON_CA55)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	CLK5_ON	1b	RW	The CA55_TSCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
4	CLK4_ON	1b	RW	The CA55_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
3	CLK3_ON	1b	RW	The CA55_GICCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	1b	RW	The CA55_ATCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	1b	RW	The CA55_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	The CA55_SCLK/PERICLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.23 Clock Control Register Cortex-M33 (CPG_CLKON_CM33)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CLK9_ONWEN	CLK8_ONWEN	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R0W1	R0W1	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLK9_ON	CLK8_ON	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	CLK9_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	CLK8_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	CLK9_ON	0b	RW	The CM33_FPU_TSCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
8	CLK8_ON	0b	RW	The CM33_FPU_CLKIN clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The CM33_TSCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The CM33_CLKIN clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.24 Clock Control Register ACPU (CPG_CLKON_SRAM_ACPU)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	CLK2_ON	1b	RW	The SRAM_ACPU_ACLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	1b	RW	The SRAM_ACPU_ACLK1 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	The SRAM_ACPU_ACLK0 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.25 Clock Control Register MCPU (CPG_CLKON_SRAM_MCPU)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	1b	RW	The SRAM_MCPU_ACLK1 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	The SRAM_MCPU_ACLK0 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.26 Clock Control Register GIC600 (CPG_CLKON_GIC600)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	1b	RW	The GIC600_GICCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.27 Clock Control Register IA55 (CPG_CLKON_IA55)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	1b	RW	The IA55_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	The IA55_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.28 Clock Control Register IM33 (CPG_CLKON_IM33)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CLK9_ONWEN	CLK8_ONWEN	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R0W1	R0W1	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLK9_ON	CLK8_ON	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	CLK9_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	CLK8_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	CLK9_ON	1b	RW	The IM33_FPU_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
8	CLK8_ON	1b	RW	The IM33_FPU_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	1b	RW	The IM33_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	The IM33_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.29 Clock Control Register MHU (CPG_CLKON_MHU)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The MHU_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.30 Clock Control Register CST (CPG_CLKON_CST)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CLK13_ONWEN	CLK12_ONWEN	CLK11_ONWEN	CLK10_ONWEN	CLK9_ONWEN	CLK8_ONWEN	CLK7_ONWEN	CLK6_ONWEN	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CLK13_ON	CLK12_ON	CLK11_ON	CLK10_ON	CLK9_ON	CLK8_ON	CLK7_ON	CLK6_ON	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29	CLK13_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK13_ON bit (bit 13) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
28	CLK12_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK12_ON bit (bit 12) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
27	CLK11_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK11_ON bit (bit 11) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
26	CLK10_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	CLK9_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	CLK8_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	CLK7_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	CLK6_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15, 14	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	CLK13_ON	1b	RW	The CST_ATB_CM33_FPU_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
12	CLK12_ON	1b	RW	The CST_AHB_CM33_FPU_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
11	CLK11_ON	1b	RW	The CST_APB_CM33_FPU_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
10	CLK10_ON	1b	RW	The CST_AXI_ETR_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
9	CLK9_ON	1b	RW	The CST_AXI_SB_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
8	CLK8_ON	1b	RW	The CST_ATB_CM33_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
7	CLK7_ON	1b	RW	The CST_ATB_CA55_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
6	CLK6_ON	1b	RW	The CST_AHB_ATH_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
5	CLK5_ON	1b	RW	The CST_AHB_CM33_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
4	CLK4_ON	1b	RW	The CST_APB_CA55_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	1b	RW	The CST_APB_CM33_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	1b	RW	The CST_APB_SB_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	1b	RW	The CST_TS_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	The CST_CS_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.31 Clock Control Register SYC (CPG_CLKON_SYC)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	1b	RW	The SYC_CNT_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.32 Clock Control Register DMAC_REG (CPG_CLKON_DMAC_REG)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The DMAC_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The DMAC_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.33 Clock Control Register GTM (CPG_CLKON_GTM)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CLK7_ONWEN	CLK6_ONWEN	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CLK7_ON	CLK6_ON	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	CLK7_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	CLK6_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
7	CLK7_ON	0b	RW	The OSTM7_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
6	CLK6_ON	0b	RW	The OSTM6_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
5	CLK5_ON	0b	RW	The OSTM5_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
4	CLK4_ON	0b	RW	The OSTM4_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	0b	RW	The OSTM3_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The OSTM2_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The OSTM1_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The OSTM0_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.34 Clock Control Register MTU (CPG_CLKON_MTU)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The MTU_X_MCK_MTU3 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.35 Clock Control Register POE3 (CPG_CLKON_POE3)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The POE3_CLKM_POE clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.36 Clock Control Register GPT (CPG_CLKON_GPT)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The GPT_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.37 Clock Control Register POEG (CPG_CLKON_POEG)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_ON	0b	RW	The POEG_D_CLKP is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	ThePOEG_C_CLKP is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The POEG_B_CLKP clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The POEG_A_CLKP clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.38 Clock Control Register WDT (CPG_CLKON_WDT)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23, 22	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7, 6	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
5	CLK5_ON	0b	RW	The WDT2_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
4	CLK4_ON	0b	RW	The WDT2_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	1b	RW	The WDT1_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	1b	RW	The WDT1_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	1b	RW	The WDT0_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	The WDT0_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.39 Clock Control Register DDR (CPG_CLKON_DDR)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ONWEN	CLK0_2NWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_ON	0b	RW	The DDR_AXI1_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The DDR_AXI0_ACLK is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The DDR_REG_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The DDR_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.40 Clock Control Register SPI (CPG_CLKON_SPI)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_ON	0b	RW	The SPI_CLKX2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The SPI_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The SPI_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The SPI_HCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.41 Clock Control Register SDHI (CPG_CLKON_SDHI)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	CLK11_ONWEN	CLK10_ONWEN	CLK9_ONWEN	CLK8_ONWEN	CLK7_ONWEN	CLK6_ONWEN	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CLK11_ON	CLK10_ON	CLK9_ON	CLK8_ON	CLK7_ON	CLK6_ON	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27	CLK11_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK11_ON bit (bit 11) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
26	CLK10_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	CLK9_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	CLK8_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	CLK7_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	CLK6_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	CLK11_ON	0b	RW	The SDHI2_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
10	CLK10_ON	0b	RW	The SDHI2_CLK_HS clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
9	CLK9_ON	0b	RW	The SDHI2_IMCLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
8	CLK8_ON	0b	RW	The SDHI2_IMCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
7	CLK7_ON	0b	RW	The SDHI1_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
6	CLK6_ON	0b	RW	The SDHI1_CLK_HS clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
5	CLK5_ON	0b	RW	The SDHI1_IMCLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
4	CLK4_ON	0b	RW	The SDHI1_IMCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	0b	RW	The SDHI0_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The SDHI0_CLK_HS clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The SDHI0_IMCLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
0	CLK0_ON	0b	RW	The SDHI0_IMCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.42 Clock Control Register SSI (CPG_CLKON_SSI)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CLK7_ONWEN	CLK6_ONWEN	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CLK7_ON	CLK6_ON	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	CLK7_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	CLK6_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
7	CLK7_ON	0b	RW	The SSI3_PCLK_SFR clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
6	CLK6_ON	0b	RW	The SSI3_PCLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
5	CLK5_ON	0b	RW	The SSI2_PCLK_SFR clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
4	CLK4_ON	0b	RW	The SSI2_PCLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	0b	RW	The SSI1_PCLK_SFR clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The SSI1_PCLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The SSI0_PCLK_SFR clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The SSI0_PCLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.43 Clock Control Register SRC (CPG_CLKON_SRC)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The SRC_CLKP clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.44 Clock Control Register USB (CPG_CLKON_USB)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_ON	0b	RW	The USB_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The USB_U2P_EXR_CPUCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The USB_U2H1_HCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The USB_U2H0_HCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.45 Clock Control Register ETH (CPG_CLKON_ETH)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CLK9_ONWEN	CLK8_ONWEN	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R0W1	R0W1	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLK9_ON	CLK8_ON	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	CLK9_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	CLK8_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	CLK9_ON	0b	RW	The ETH1_REFCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
8	CLK8_ON	0b	RW	The ETH0_REFCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The ETH1_CLK_AXI/CHI clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The ETH0_CLK_AXI/CHI clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.46 Clock Control Register I2C (CPG_CLKON_I2C)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_ON	0b	RW	The I2C3_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The I2C2_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The I2C1_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The I2C0_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.47 Clock Control Register SCIF (CPG_CLKON_SCIF)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	CLK5_ON	0b	RW	The SCIF5_CLK_PCK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
4	CLK4_ON	0b	RW	The SCIF4_CLK_PCK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
3	CLK3_ON	0b	RW	The SCIF3_CLK_PCK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The SCIF2_CLK_PCK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The SCIF1_CLK_PCK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The SCIF0_CLK_PCK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.48 Clock Control Register SCI (CPG_CLKON_SCI)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The SCI1_CLKP clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The SCI0_CLKP clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.49 Clock Control Register IRDA (CPG_CLKON_IRDA)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The IRDA_CLKP clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.50 Clock Control Register RSPI (CPG_CLKON_RSPI)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	CLK4_ON	0b	RW	The RSPI4_CLKB clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	0b	RW	The RSPI3_CLKB clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The RSPI2_CLKB clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The RSPI1_CLKB clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
0	CLK0_ON	0b	RW	The RSPI0_CLKB clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.51 Clock Control Register CANFD (CPG_CLKON_CANFD)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The CANFD_CLK_RAM clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The CANFD_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.52 Clock Control Register GPIO (CPG_CLKON_GPIO)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	1b	RW	The GPIO_HCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.53 Clock Control Register ADC (CPG_CLKON_ADC)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The ADC_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The ADC_ADCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.54 Clock Control Register TSU (CPG_CLKON_TSU)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	CLK0_ON	0b	RW	The TSU_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.55 Clock Control Register AXI_ACPU_BUS (CPG_CLKON_AXI_ACPU_BUS)

This register is used to supply or stop clocks for individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	CLK6_ONWEN	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CLK6_ON	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	CLK6_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	CLK6_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
5	CLK5_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
4	CLK4_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.

7.2.4.56 Clock Control Register AXI_MCPU_BUS (CPG_CLKON_AXI_MCPU_BUS)

This register is used to supply or stop clocks for individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CLK15_ONWEN	CLK14_ONWEN	—	CLK12_ONWEN	CLK11_ONWEN	CLK10_ONWEN	—	CLK8_ONWEN	CL76_ONWEN	CLK6_ONWEN	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W1	R0W1	R	R0W1	R0W1	R0W1	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLK15_ON	CLK14_ON	—	CLK12_ON	CLK11_ON	CLK10_ON	—	CLK8_ON	CLK7_ON	CLK0_ON	CLK0_ON	CLK0_ON	CLK0_ON	CLK0_ON	CLK0_ON	CLK0_ON
Initial Value	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1
R/W	RW	RW	R	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	CLK15_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK15_ON bit (bit 15) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
30	CLK14_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK14_ON bit (bit 14) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
29	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
28	CLK12_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK12_ON bit (bit 12) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
27	CLK11_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK11_ON bit (bit 11) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
26	CLK10_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
24	CLK8_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	CLK7_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
22	CLK6_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15	CLK15_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
14	CLK14_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
13	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
12	CLK12_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
11	CLK11_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
10	CLK10_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
9	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
8	CLK8_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
7	CLK7_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
6	CLK6_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
5	CLK5_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
4	CLK4_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.

7.2.4.57 Clock Control Register AXI_COM_BUS (CPG_CLKON_AXI_COM_BUS)

This register is used to supply or stop clocks for individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CLK9_ONWEN	CLK8_ONWEN	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R0W1	R0W1	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLK9_ON	CLK8_ON	—	—	—	—	—	—	CLK0_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	CLK9_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	CLK8_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	CLK9_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
8	CLK8_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.

7.2.4.58 Clock Control Register PERI_COM (CPG_CLKON_PERI_COM)

This register is used to supply or stop clocks for individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CLK9_ONWEN	CLK8_ONWEN	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R0W1	R0W1	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLK9_ON	CLK8_ON	—	—	—	—	—	—	CLK0_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	CLK9_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	CLK8_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	CLK9_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
8	CLK8_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.

7.2.4.59 Clock Control Register REG1_BUS (CPG_CLKON_REG1_BUS)

This register is used to supply or stop clocks for individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.

7.2.4.60 Clock Control Register REG0_BUS (CPG_CLKON_REG0_BUS)

This register is used to supply or stop clocks for individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ON	CLK2_ON	CLK0_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.

7.2.4.61 Clock Control Register PERI_CPU (CPG_CLKON_PERI_CPU)

This register is used to supply or stop clocks for individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CLK7_ONWEN	CLK6_ONWEN	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CLK7_ON	CLK6_ON	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	CLK7_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	CLK6_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	CLK7_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
6	CLK6_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
5	CLK5_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
4	CLK4_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.

7.2.4.62 Clock Control Register PERI_DDR (CPG_CLKON_PERI_DDR)

This register is used to supply or stop clocks for individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	1b	RW	0: Clock is stopped. 1: Clock is supplied.

7.2.4.63 Clock Control Register AXI_TZCDDR (CPG_CLKON_AXI_TZCDDR)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20 to 17	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3 to 1	—	111b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	CLK0_ON	1b	RW	The BUS_TZCDDR_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.64 Clock Control Register OCTA (CPG_CLKON_OCTA)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The OCTA_MCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The OCTA_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.65 Clock Control Register OTFDE_DDR (CPG_CLKON_OTFDE_DDR)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The OTFDE_DDR_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The OTFDE_DDR_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.66 Clock Control Register OTFDE_SPI (CPG_CLKON_OTFDE_SPI)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The OTFDE_SPI_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The OTFDE_SPI_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.67 Clock Control Register PDM (CPG_CLKON_PDM)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21 to 18	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5 to 2	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	CLK1_ON	0b	RW	The PDM_CCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The PDM_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.68 Clock Control Register PCI (CPG_CLKON_PCI)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The PCI_CLKL1PM clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The PCI_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.69 Clock Control Register SPDIF (CPG_CLKON_SPDIF)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The SPDIF_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.70 Clock Control Register I3C (CPG_CLKON_I3C)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The I3C_TCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The I3C_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.71 Clock Control Register VBAT (CPG_CLKON_VBAT)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	CLK0_ON	0b	RW	The VBAT_BCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.72 Clock Monitor Register Cortex-A55 (CPG_CLKMON_CA55)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLK5_MON	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	CLK5_MON	1b	R	The state of the CA55_TSCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	1b	R	The state of the CA55_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	1b	R	The state of the CA55_GICCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	1b	R	The state of the CA55_ATCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	1b	R	The state of the CA55_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	The state of the CA55_SCLK/PERICLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.73 Clock Monitor Register Cortex-M33 (CPG_CLKMON_CM33)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLK9_MON	CLK8_MON	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	CLK9_MON	0b	R	The state of the CM33_FPU_TSCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
8	CLK8_MON	0b	R	The state of the CM33_FPU_CLKIN clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the CM33_TSCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the CM33_CLKIN clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.74 Clock Monitor Register SRAM_ACPU (CPG_CLKMON_SRAM_ACPU)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	CLK2_MON	1b	R	The state of the SRAM_ACPU_ACLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	1b	R	The state of the SRAM_ACPU_ACLK1 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	The state of the SRAM_ACPU_ACLK0 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.75 Clock Monitor Register SRAM_MCPU (CPG_CLKMON_SRAM_MCPU)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	1b	R	The state of the SRAM_MCPU_ACLK1 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	The state of the SRAM_MCPU_ACLK0 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.76 Clock Monitor Register GIC600 (CPG_CLKMON_GIC600)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	1b	R	The state of the GIC600_GICCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.77 Clock Monitor Register IA55 (CPG_CLKMON_IA55)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	1b	R	The state of the IA55_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	The state of the IA55_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.78 Clock Monitor Register IM33 (CPG_CLKMON_IM33)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLK9_MON	CLK8_MON	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	CLK9_MON	1b	R	The state of the IM33_FPU_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
8	CLK8_MON	1b	R	The state of the IM33_FPU_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	1b	R	The state of the IM33_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	The state of the IM33_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.79 Clock Monitor Register MHU (CPG_CLKMON_MHU)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the MHU_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.80 Clock Monitor Register CST (CPG_CLKMON_CST)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CLK13_MON	CLK12_MON	CLK11_MON	CLK10_MON	CLK9_MON	CLK8_MON	CLK7_MON	CLK6_MON	CLK5_MON	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	CLK13_MON	1b	R	The state of the CST_ATB_CM33_FPU_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
12	CLK12_MON	1b	R	The state of the CST_AHB_CM33_FPU_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
11	CLK11_MON	1b	R	The state of the CST_APB_CM33_FPU_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
10	CLK10_MON	1b	R	The state of the CST_AXI_ETR_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
9	CLK9_MON	1b	R	The state of the CST_AXI_SB_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
8	CLK8_MON	1b	R	The state of the CST_ATB_CM33_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
7	CLK7_MON	1b	R	The state of the CST_ATB_CA55_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
6	CLK6_MON	1b	R	The state of the CST_AHB_ATH_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
5	CLK5_MON	1b	R	The state of the CST_AHB_CM33_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	1b	R	The state of the CST_APB_CA55_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	1b	R	The state of the CST_APB_CM33_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
2	CLK2_MON	1b	R	The state of the CST_APB_SB_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	1b	R	The state of the CST_TS_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	The state of the CST_CS_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

Note: No CGC is provided for CST_SWCLKTCK.

7.2.4.81 Clock Monitor Register SYC (CPG_CLKMON_SYC)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	1b	R	The state of the SYC_CNT_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.82 Clock Monitor Register DMAC_REG (CPG_CLKMON_DMAC_REG)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the DMAC_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the DMAC_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.83 Clock Monitor Register GTM (CPG_CLKMON_GTM)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CLK7_MON	CLK6_MON	CLK5_MON	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	CLK7_MON	0b	R	The state of the OSTM7_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
6	CLK6_MON	0b	R	The state of the OSTM6_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
5	CLK5_MON	0b	R	The state of the OSTM5_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	0b	R	The state of the OSTM4_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	0b	R	The state of the OSTM3_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the OSTM2_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the OSTM1_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the OSTM0_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.84 Clock Monitor Register MTU (CPG_CLKMON_MTU)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the MTU_X_MCK_MTU3 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.85 Clock Monitor Register POE3 (CPG_CLKMON_POE3)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the POE3_CLKM_POE clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.86 Clock Monitor Register GPT (CPG_CLKMON_GPT)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the GPT_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.87 Clock Monitor Register POEG (CPG_CLKMON_POEG)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_MON	0b	R	The state of the POEG_D_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the POEG_C_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the POEG_B_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the POEG_A_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.88 Clock Monitor Register WDT (CPG_CLKMON_WDT)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLK5_MON	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
7, 6	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
5	CLK5_MON	0b	R	The state of the WDT2_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	0b	R	The state of the WDT2_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	1b	R	The state of the WDT1_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	1b	R	The state of the WDT1_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	1b	R	The state of the WDT0_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	The state of the WDT0_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.89 Clock Monitor Register DDR (CPG_CLKMON_DDR)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_MON	0b	R	The state of the DDR_AXI1_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the DDR_AXI0_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the DDR_REG_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the DDR_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.90 Clock Monitor Register SPI (CPG_CLKMON_SPI)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_MON	0b	R	The state of the SPI_CLKX2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the SPI_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the SPI_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the SPI_HCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.91 Clock Monitor Register SDHI (CPG_CLKMON_SDHI)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CLK11_MON	CLK10_MON	CLK9_MON	CLK8_MON	CLK7_MON	CLK6_MON	CLK5_MON	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	CLK11_MON	0b	R	The state of the SDHI2_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
10	CLK10_MON	0b	R	The state of the SDHI2_CLK_HS clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
9	CLK9_MON	0b	R	The state of the SDHI2_IMCLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
8	CLK8_MON	0b	R	The state of the SDHI2_IMCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
7	CLK7_MON	0b	R	The state of the SDHI1_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
6	CLK6_MON	0b	R	The state of the SDHI1_CLK_HS clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
5	CLK5_MON	0b	R	The state of the SDHI1_IMCLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	0b	R	The state of the SDHI1_IMCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	0b	R	The state of the SDHI0_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the SDHI0_CLK_HS clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the SDHI0_IMCLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
0	CLK0_MON	0b	R	The state of the SDHI0_IMCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.92 Clock Monitor Register SSI (CPG_CLKMON_SSI)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CLK7_MON	CLK6_MON	CLK5_MON	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	CLK7_MON	0b	R	The state of the SSI3_PCLK_SFR clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
6	CLK6_MON	0b	R	The state of the SSI3_PCLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
5	CLK5_MON	0b	R	The state of the SSI2_PCLK_SFR clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	0b	R	The state of the SSI2_PCLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	0b	R	The state of the SSI1_PCLK_SFR clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the SSI1_PCLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the SSI0_PCLK_SFR clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the SSI0_PCLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.93 Clock Monitor Register SRC (CPG_CLKMON_SRC)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the SRC_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.94 Clock Monitor Register USB (CPG_CLKMON_USB)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_MON	0b	R	The state of the USB_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the USB_U2P_EXR_CPUCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the USB_U2H1_HCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the USB_U2H0_HCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.95 Clock Monitor Register ETH (CPG_CLKMON_ETH)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLK9_MON	CLK8_MON	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	CLK9_MON	0b	R	The state of the ETH1_REFCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
8	CLK8_MON	0b	R	The state of the ETH0_REFCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the ETH1_CLK_AXI/CHI clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the ETH0_CLK_AXI/CHI clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

Note: No CGC is provided for ETH1_REFCLK and ETH0_REFCLK.

7.2.4.96 Clock Monitor Register I2C (CPG_CLKMON_I2C)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_MON	0b	R	The state of the I2C3_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the I2C2_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the I2C1_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the I2C0_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.97 Clock Monitor Register SCIF (CPG_CLKMON_SCIF)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLK5_MON	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	CLK5_MON	0b	R	The state of the SCIF5_CLK_PCK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	0b	R	The state of the SCIF4_CLK_PCK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	0b	R	The state of the SCIF3_CLK_PCK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the SCIF2_CLK_PCK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the SCIF1_CLK_PCK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the SCIF0_CLK_PCK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.98 Clock Monitor Register SCI (CPG_CLKMON_SCI)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the SCI1_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the SCI0_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.99 Clock Monitor Register IRDA (CPG_CLKMON_IRDA)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the IRDA_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.100 Clock Monitor Register RSPI (CPG_CLKMON_RSPI)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	CLK4_MON	0b	R	The state of the RSPI4_CLKB clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	0b	R	The state of the RSPI3_CLKB clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the RSPI2_CLKB clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the RSPI1_CLKB clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the RSPI0_CLKB clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.101 Clock Monitor Register CANFD (CPG_CLKMON_CANFD)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the CANFD_CLK_RAM clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the CANFD_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.102 Clock Monitor Register GPIO (CPG_CLKMON_GPIO)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	1b	R	The state of the GPIO_HCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.103 Clock Monitor Register ADC (CPG_CLKMON_ADC)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the ADC_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the ADC_ADCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.104 Clock Monitor Register TSU (CPG_CLKMON_TSU)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	CLK0_MON	0b	R	The state of the TSU_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.105 Clock Monitor Register AXI_ACPU_BUS (CPG_CLKMON_AXI_ACPU_BUS)

This register is used to monitor the state of the clocks supplied to individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CLK6_MON	CLK5_MON	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	CLK6_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
5	CLK5_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.

7.2.4.106 Clock Monitor Register AXI_MCPU_BUS (CPG_CLKMON_AXI_MCPU_BUS)

This register is used to monitor the state of the clocks supplied to individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLK15_MON	CLK14_MON	—	CLK12_MON	CLK11_MON	CLK10_MON	—	CLK8_MON	CLK7_MON	CLK6_MON	CLK5_MON	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15	CLK15_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
14	CLK14_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
13	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	CLK12_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
11	CLK11_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
10	CLK10_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
9	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	CLK8_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
7	CLK7_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
6	CLK6_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
5	CLK5_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
0	CLK0_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.

7.2.4.107 Clock Monitor Register AXI_COM_BUS (CPG_CLKMON_AXI_COM_BUS)

This register is used to monitor the state of the clocks supplied to individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLK9_MON	CLK8_MON	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	CLK9_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
8	CLK8_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.

7.2.4.108 Clock Monitor Register PERI_COM (CPG_CLKMON_PERI_COM)

This register is used to monitor the state of the clocks supplied to individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLK9_MON	CLK8_MON	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	CLK9_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
8	CLK8_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.

7.2.4.109 Clock Monitor Register REG1_BUS (CPG_CLKMON_REG1_BUS)

This register is used to monitor the state of the clocks supplied to individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.

7.2.4.110 Clock Monitor Register REG0_BUS (CPG_CLKMON_REG0_BUS)

This register is used to monitor the state of the clocks supplied to individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.

7.2.4.111 Clock Monitor Register PERI_CPU (CPG_CLMKON_PERI_CPU)

This register is used to monitor the state of the clocks supplied to individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CLK7_MON	CLK6_MON	CLK5_MON	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	CLK7_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
6	CLK6_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
5	CLK5_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.

7.2.4.112 Clock Monitor Register PERI_DDR (CPG_CLKMON_PERI_DDR)

This register is used to monitor the state of the clocks supplied to individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	1b	R	0: Clock is not supplied. 1: Clock is supplied.

7.2.4.113 Clock Monitor Register AXI_TZCDDR (CPG_CLKMON_AXI_TZCDDR)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3 to 1	—	111b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	CLK0_MON	1b	R	The state of the BUS_TZCDDR_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.114 Clock Monitor Register OCTA (CPG_CLKMON_OCTA)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the OCTA_MCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the OCTA_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.115 Clock Monitor Register OTFDE_DDR (CPG_CLKMON_OTFDE_DDR)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the OTFDE_DDR_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the OTFDE_DDR_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.116 Clock Monitor Register OTFDE_SPI (CPG_CLKMON_OTFDE_SPI)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the OTFDE_SPI_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the OTFDE_SPI_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.117 Clock Monitor Register PDM (CPG_CLKMON_PDM)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the PDM_CCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the PDM_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.118 Clock Monitor Register PCI (CPG_CLKMON_PCI)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the PCI_CLKL1PM clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the PCI_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.119 Clock Monitor Register SPDIF (CPG_CLKMON_SPDIF)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the SPDIF_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.120 Clock Monitor Register I3C (CPG_CLKMON_I3C)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the I3C_TCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the I3C_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.121 Clock Monitor Register VBAT (CPG_CLKMON_VBAT)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	CLK0_MON	0b	R	The state of the VBAT_BCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.122 Reset Control Register Cortex-A55 (CPG_RST_CA55)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	UNIT12_RSTWEN	UNIT11_RSTWEN	UNIT10_RSTWEN	UNIT9_RSTWEN	UNIT8_RSTWEN	UNIT7_RSTWEN	UNIT6_RSTWEN	UNIT5_RSTWEN	UNIT4_RSTWEN	—	UNIT2_RSTWEN	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R	R0W1	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	UNIT12_RSTB	UNIT11_RSTB	UNIT10_RSTB	UNIT9_RSTB	UNIT8_RSTB	UNIT7_RSTB	UNIT6_RSTB	UNIT5_RSTB	UNIT4_RSTB	—	UNIT2_RSTB	—	UNIT0_RSTB
Initial Value	0	0	0	1	1	1	1	1	1	1	1	1	0	1	0	1
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	UNIT12_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT12_RSTB bit (bit 12) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
27	UNIT11_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT11_RSTB bit (bit 11) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
26	UNIT10_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT10_RSTB bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	UNIT9_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT9_RSTB bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	UNIT8_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT8_RSTB bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	UNIT7_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT7_RSTB bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	UNIT6_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT6_RSTB bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	UNIT5_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT5_RSTB bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
20	UNIT4_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT4_RSTB bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	UNIT12_RSTB	1b	RW	The CA55_RST12 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
11	UNIT11_RSTB	1b	RW	The CA55_RST11 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
10	UNIT10_RSTB	1b	RW	The CA55_RST10 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
9	UNIT9_RSTB	1b	RW	The CA55_RST9 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
8	UNIT8_RSTB	1b	RW	The CA55_RST8 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
7	UNIT7_RSTB	1b	RW	The CA55_RST7 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
6	UNIT6_RSTB	1b	RW	The CA55_RST6reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
5	UNIT5_RSTB	1b	RW	The CA55_RST5 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
4	UNIT4_RSTB	1b	RW	The CA55_RST4 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
3	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

Bit	Bit Name	Initial Value	R/W	Description
2	UNIT2_RSTB	1b	RW	The CA55_RST3_0 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	UNIT0_RSTB	1b	RW	The CA55_RST1_0 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.123 Reset Control Register Cortex-M33 (CPG_RST_CM33)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	UNIT10_RSTWEN	UNIT9_RSTWEN	UNIT8_RSTWEN	—	—	—	—	—	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R0W1	R0W1	R0W1	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	UNIT10_RSTB	UNIT9_RSTB	UNIT8_RSTB	—	—	—	—	—	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	RW	RW	RW	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	UNIT10_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT10_RSTB bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	UNIT9_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT9_RSTB bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	UNIT8_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT8_RSTB bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	UNIT10_RSTB	0b	RW	The CM33_FPU_MISCRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
9	UNIT9_RSTB	0b	RW	The CM33_FPU_NSYSRESET reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

Bit	Bit Name	Initial Value	R/W	Description
8	UNIT8_RSTB	0b	RW	The CM33_FPU_NPORESET reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
7 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	UNIT2_RSTB	0b	RW	The CM33_MISCRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The CM33_NSYSRESET reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The CM33_NPORESET reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.124 Reset Control Register SRAM_ACPU (CPG_RST_SRAM_ACPU)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	UNIT2_RSTB	1b	RW	The SRAM_ACPU_ARESETN2 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	1b	RW	The SRAM_ACPU_ARESETN1 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	1b	RW	The SRAM_ACPU_ARESETN0 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.125 Reset Control Register SRAM_MCPU (CPG_RST_SRAM_MCPU)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	1b	RW	The SRAM_MCPU_ARESETN1 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	1b	RW	The SRAM_MCPU_ARESETN0 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.126 Reset Control Register GIC600 (CPG_RST_GIC600)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	1b	RW	The GIC600_DBG_GICRESET_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	1b	RW	The GIC600_GICRESET_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.127 Reset Control Register IA55 (CPG_RST_IA55)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	1b	RW	The IA55_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.128 Reset Control Register IM33 (CPG_RST_IM33)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	UNIT8_RSTWEN	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R0W1	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	UNIT8_RSTB	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	UNIT8_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT8_RSTB bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	UNIT8_RSTB	1b	RW	The IM33_FPU_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
7 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	1b	RW	The IM33_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.129 Reset Control Register MHU (CPG_RST_MHU)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The MHU_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.130 Reset Control Register DMAC (CPG_RST_DMAC)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The DMAC_RST_ASYNC reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The DMAC_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.131 Reset Control Register SYC (CPG_RST_SYC)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	1b	RW	The SYC_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.132 Reset Control Register GTM (CPG_RST_GTM)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	UNIT7_RSTWEN	UNIT6_RSTWEN	UNIT5_RSTWEN	UNIT4_RSTWEN	UNIT3_RSTWEN	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	UNIT7_RSTB	UNIT6_RSTB	UNIT5_RSTB	UNIT4_RSTB	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	UNIT7_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT7_RSTB bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	UNIT6_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT6_RSTB bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	UNIT5_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT5_RSTB bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	UNIT4_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT4_RSTB bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	UNIT3_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
7	UNIT7_RSTB	0b	RW	The OSTM7_PRESETZ reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
6	UNIT6_RSTB	0b	RW	The OSTM6_PRESETZ reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
5	UNIT5_RSTB	0b	RW	The OSTM5_PRESETZ reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
4	UNIT4_RSTB	0b	RW	The OSTM4_PRESETZ reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
3	UNIT3_RSTB	0b	RW	The OSTM3_PRESETZ reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The OSTM2_PRESETZ reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The OSTM1_PRESETZ reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The OSTM0_PRESETZ reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.133 Reset Control Register MTU (CPG_RST_MTU)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The MTU_X_PRESET_MTU3 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.134 Reset Control Register POE3 (CPG_RST_POE3)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The POE3_RST_M_REG reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.135 Reset Control Register GPT (CPG_RST_GPT)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The GPT_RST_C reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.136 Reset Control Register POEG (CPG_RST_POEG)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTWEN	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	UNIT3_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	UNIT3_RSTB	0b	RW	The POEG_D_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The POEG_C_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The POEG_B_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The POEG_A_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.137 Reset Control Register WDT (CPG_RST_WDT)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2	UNIT2_RSTB	0b	RW	The WDT2_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	1b	RW	The WDT1_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	1b	RW	The WDT0_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.138 Reset Control Register DDR (CPG_RST_DDR)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	UNIT8_RSTWEN	UNIT7_RSTWEN	UNIT6_RSTWEN	—	—	UNIT3_RSTWEN	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	UNIT8_RSTB	UNIT7_RSTB	UNIT6_RSTB	—	—	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	UNIT8_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT8_RSTB bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	UNIT7_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT7_RSTB bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	UNIT6_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT6_RSTB bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21, 20	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
19	UNIT3_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
8	UNIT8_RSTB	0b	RW	The DDR_PWROKIN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
7	UNIT7_RSTB	0b	RW	The DDR_RESET reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
6	UNIT6_RSTB	0b	RW	The DDR_REG_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
5, 4	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	UNIT3_RSTB	0b	RW	The DDR_AXI1_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The DDR_AXI0_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The DDR_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The DDR_RESET_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.139 Reset Control Register SPI (CPG_RST_SPI)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The SPI_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The SPI_HRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.140 Reset Control Register SDHI (CPG_RST_SDHI)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	UNIT2_RSTB	0b	RW	The SDHI2_IXRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The SDHI1_IXRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The SDHI0_IXRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.141 Reset Control Register SSIF (CPG_RST_SSIF)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTWEN	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	UNIT3_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	UNIT3_RSTB	0b	RW	The SSI3_RST_M2_REG reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The SSI2_RST_M2_REG reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The SSI1_RST_M2_REG reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The SSI0_RST_M2_REG reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.142 Reset Control Register SRC (CPG_RST_SRC)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The SRC_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.143 Reset Control Register USB (CPG_RST_USB)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTWEN	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	UNIT3_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	UNIT3_RSTB	0b	RW	The USB_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The USB_U2P_EXL_SYSRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The USB_U2H1_HRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The USB_U2H0_HRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.144 Reset Control Register ETH (CPG_RST_ETH)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The ETH1_RST_HW_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The ETH0_RST_HW_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.145 Reset Control Register I2C (CPG_RST_I2C)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTWEN	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	UNIT3_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	UNIT3_RSTB	0b	RW	The I2C3_MRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The I2C2_MRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The I2C1_MRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The I2C0_MRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.146 Reset Control Register SCIF (CPG_RST_SCIF)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	UNIT5_RSTWEN	UNIT4_RSTWEN	UNIT3_RSTWEN	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	UNIT5_RSTB	UNIT4_RSTB	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	UNIT5_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT5_RSTB bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	UNIT4_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT4_RSTB bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	UNIT3_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	UNIT5_RSTB	0b	RW	The SCIF5_RST_SYSTEM_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
4	UNIT4_RSTB	0b	RW	The SCIF4_RST_SYSTEM_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

Bit	Bit Name	Initial Value	R/W	Description
3	UNIT3_RSTB	0b	RW	The SCIF3_RST_SYSTEM_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The SCIF2_RST_SYSTEM_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The SCIF1_RST_SYSTEM_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The SCIF0_RST_SYSTEM_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.147 Reset Control Register SCI (CPG_RST_SCI)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The SCI1_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The SCI0_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.148 Reset Control Register IRDA (CPG_RST_IRDA)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The IRDA_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.149 Reset Control Register RSPI (CPG_RST_RSPI)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	UNIT4_RSTWEN	UNIT3_RSTWEN	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	UNIT4_RSTB	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	UNIT4_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT4_RSTB bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	UNIT3_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	UNIT4_RSTB	0b	RW	The RSPI4_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
3	UNIT3_RSTB	0b	RW	The RSPI3_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The RSPI2_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The RSPI1_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

Bit	Bit Name	Initial Value	R/W	Description
0	UNIT0_RSTB	0b	RW	The RSPI0_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.150 Reset Control Register CANFD (CPG_RST_CANFD)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The CANFD_RSTC_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The CANFD_RSTP_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.151 Reset Control Register GPIO (CPG_RST_GPIO)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	UNIT2_RSTB	1b	RW	The GPIO_SPARE_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	1b	RW	The GPIO_PORT_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	1b	RW	The GPIO_RSTN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.152 Reset Control Register ADC (CPG_RST_ADC)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The ADC_ADRST_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The ADC_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.153 Reset Control Register TSU (CPG_RST_TSU)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	UNIT0_RSTB	0b	RW	The TSU_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.154 Reset Control Register AXI_ACPU_BUS (CPG_RST_AXI_ACPU_BUS)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	1b	RW	0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.155 Reset Control Register AXI_MCPU_BUS (CPG_RST_AXI_MCPU_BUS)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	1b	RW	0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.156 Reset Control Register AXI_COM_BUS (CPG_RST_AXI_COM_BUS)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	1b	RW	0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	1b	RW	0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.157 Reset Control Register PERI_COM (CPG_RST_PERI_COM)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	1b	RW	0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	1b	RW	0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.158 Reset Control Register REG1_BUS (CPG_RST_REG1_BUS)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	1b	RW	0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.159 Reset Control Register REG0_BUS (CPG_RST_REG0_BUS)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	1b	RW	0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.160 Reset Control Register PERI_CPU (CPG_RST_PERI_CPU)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	1b	RW	0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	1b	RW	0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.161 Reset Control Register PERI_DDR (CPG_RST_PERI_DDR)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	1b	RW	0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.162 Reset Control Register AXI_TZCDDR (CPG_RST_AXI_TZCDDR)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTWEN	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
19	UNIT3_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	UNIT3_RSTB	1b	RW	The BUS_TZCDDR_ARESET2N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	1b	RW	The BUS_TZCDDR_ARESET1N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	1b	RW	The BUS_TZCDDR_ARESET0N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

Bit	Bit Name	Initial Value	R/W	Description
0	UNIT0_RSTB	1b	RW	The BUS_TZCDDR_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.163 Reset Control Register OCTA (CPG_RST_OCTA)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The OCTA_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.164 Reset Control Register OTFDE_DDR (CPG_RST_OTFDE_DDR)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The OTFDE_DDR_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The OTFDE_DDR_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.165 Reset Control Register OTFDE_SPI (CPG_RST_OTFDE_SPI)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The OTFDE_SPI_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The OTFDE_SPI_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.166 Reset Control Register PDM (CPG_RST_PDM)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18, 17	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2, 1	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	UNIT0_RSTB	0b	RW	The PDM0_PRESETNT reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.167 Reset Control Register PCI (CPG_RST_PCI)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	UNIT6_RSTWEN	UNIT5_RSTWEN	UNIT4_RSTWEN	UNIT3_RSTWEN	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	UNIT6_RSTB	UNIT5_RSTB	UNIT4_RSTB	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	UNIT6_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT6_RSTB bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	UNIT5_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT5_RSTB bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	UNIT4_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT4_RSTB bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	UNIT3_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	UNIT6_RSTB	0b	RW	The PCI_RST_LOAD_B reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

Bit	Bit Name	Initial Value	R/W	Description
5	UNIT5_RSTB	0b	RW	The PCI_RST_CFG_B reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
4	UNIT4_RSTB	0b	RW	The PCI_RST_RSM_B reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
3	UNIT3_RSTB	0b	RW	The PCI_RST_PS_B reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The PCI_RST_GP_B reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The PCI_RST_B reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The PCI_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.168 Reset Control Register SPDIF (CPG_RST_SPDIF)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The SPDIF_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.169 Reset Control Register I3C (CPG_RST_I3C)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The I3C_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The I3C_TRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.170 Reset Control Register VBAT (CPG_RST_VBAT)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	UNIT0_RSTB	0b	RW	The VBAT_BRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.171 Reset Monitor Register Cortex-CA55 (CPG_RSTMON_CA55)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	RST12_MON	RST11_MON	RST10_MON	RST9_MON	RST8_MON	RST7_MON	RST6_MON	RST5_MON	RST4_MON	—	RST2_MON	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	RST12_MON	0b	R	The state of the CA55_RST12 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
11	RST11_MON	0b	R	The state of the CA55_RST11 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
10	RST10_MON	0b	R	The state of the CA55_RST10 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
9	RST9_MON	0b	R	The state of the CA55_RST9 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
8	RST8_MON	0b	R	The state of the CA55_RST8 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
7	RST7_MON	0b	R	The state of the CA55_RST7 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
6	RST6_MON	0b	R	The state of the CA55_RST6 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
5	RST5_MON	0b	R	The state of the CA55_RST5 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
4	RST4_MON	0b	R	The state of the CA55_RST4 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
3	—	1b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2	RST2_MON	0b	R	The state of the CA55_RST3_0 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

Bit	Bit Name	Initial Value	R/W	Description
1	—	1b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	RST0_MON	0b	R	The state of the CA55_RST1_0 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.172 Reset Monitor Register Cortex-M33 (CPG_RSTMON_CM33)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RST10_MON	RST9_MON	RST8_MON	—	—	—	—	—	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	RST10_MON	1b	R	The state of the CM33_FPU_MISCRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
9	RST9_MON	1b	R	The state of the CM33_FPU_NSYSRESET reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
8	RST8_MON	1b	R	The state of the CM33_FPU_NPORESET reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
7 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	RST2_MON	1b	R	The state of the CM33_MISCRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the CM33_NSYSRESET reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the CM33_NPORESET reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.173 Reset Monitor Register SRAM_ACPU (CPG_RSTMON_SRAM_ACPU)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	0b	R	The state of the SRAM_ACPU_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.174 Reset Monitor Register SRAM_MCPU (CPG_RSTMON_SRAM_MCPU)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	0b	R	The state of the SRAM_MCPU_ARESETN1 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	0b	R	The state of the SRAM_MCPU_ARESETN0 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.175 Reset Monitor Register GIC600 (CPG_RSTMON_GIC600)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	0b	R	The state of the GIC600_DBG_GICRESET_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	0b	R	The state of the GIC600_GICRESET_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.176 Reset Monitor Register IA55 (CPG_RSTMON_IA55)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	0b	R	The state of the IA55_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.177 Reset Monitor Register IM33 (CPG_RSTMON_IM33)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	0b	R	The state of the IM33_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.178 Reset Monitor Register MHU (CPG_RSTMON_MHU)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the MHU_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.179 Reset Monitor Register DMAC (CPG_RSTMON_DMAC)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the DMAC_RST_ASYNC reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the DMAC_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.180 Reset Monitor Register SYC (CPG_RSTMON_SYC)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	0b	R	The state of the SYC_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.181 Reset Monitor Register GTM (CPG_RSTMON_GTM)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RST7_MON	RST2_MON	RST2_MON	RST2_MON	RST2_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	RST7_MON	1b	R	The state of the OSTM7_PRESETZ reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
6	RST6_MON	1b	R	The state of the OSTM6_PRESETZ reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
5	RST5_MON	1b	R	The state of the OSTM5_PRESETZ reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
4	RST4_MON	1b	R	The state of the OSTM4_PRESETZ reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
3	RST3_MON	1b	R	The state of the OSTM3_PRESETZ reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the OSTM2_PRESETZ reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the OSTM1_PRESETZ reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the OSTM0_PRESETZ reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.182 Reset Monitor Register MTU (CPG_RSTMON_MTU)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the MTU_X_PRESET_MTU3 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.183 Reset Monitor Register POE3 (CPG_RSTMON_POE3)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the POE3_RST_M_REG reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.184 Reset Monitor Register GPT (CPG_RSTMON_GPT)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the GPT_RST_C reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.185 Reset Monitor Register POEG (CPG_RSTMON_POEG)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	RST3_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	RST3_MON	1b	R	The state of the POEG_D_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the POEG_C_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the POEG_B_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the POEG_A_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.186 Reset Monitor Register WDT (CPG_RSTMON_WDT)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	—	1b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2	RST2_MON	1b	R	The state of the WDT2_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	0b	R	The state of the WDT1_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	0b	R	The state of the WDT0_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.187 Reset Monitor Register DDR (CPG_RSTMON_DDR)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	RST8_MON	RST7_MON	RST6_MON	—	—	RST3_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	RST8_MON	1b	R	The state of the DDR_PWROKIN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
7	RST7_MON	1b	R	The state of the DDR_RESET reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
6	RST6_MON	1b	R	The state of the DDR_REG_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
5, 4	—	11b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	RST3_MON	1b	R	The state of the DDR_AXI1_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the DDR_AXI0_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the DDR_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the DDR_RESET_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.188 Reset Monitor Register SPI (CPG_RSTMON_SPI)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the SPI_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the SPI_HRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.189 Reset Monitor Register SDHI (CPG_RSTMON_SDHI)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	RST2_MON	1b	R	The state of the SDHI2_IXRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the SDHI1_IXRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the SDHI0_IXRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.190 Reset Monitor Register SSIF (CPG_RSTMON_SSIF)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	RST3_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	RST3_MON	1b	R	The state of the SSI3_RST_M2_REG reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the SSI2_RST_M2_REG reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the SSI1_RST_M2_REG reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the SSI0_RST_M2_REG reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.191 Reset Monitor Register SRC (CPG_RSTMON_SRC)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the SRC_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.192 Reset Monitor Register USB (CPG_RSTMON_USB)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	RST3_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	RST3_MON	1b	R	The state of the USB_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the USB_U2P_EXL_SYSRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the USB_U2H1_HRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the USB_U2H0_HRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.193 Reset Monitor Register ETH (CPG_RSTMON_ETH)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the ETH1_RST_HW_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the ETH0_RST_HW_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.194 Reset Monitor Register I2C (CPG_RSTMON_I2C)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	RST3_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	RST3_MON	1b	R	The state of the I2C3_MRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the I2C2_MRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the I2C1_MRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the I2C0_MRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.195 Reset Monitor Register SCIF (CPG_RSTMON_SCIF)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	RST5_MON	RST4_MON	RST3_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	RST5_MON	1b	R	The state of the SCIF5_RST_SYSTEM_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
4	RST4_MON	1b	R	The state of the SCIF4_RST_SYSTEM_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
3	RST3_MON	1b	R	The state of the SCIF3_RST_SYSTEM_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the SCIF2_RST_SYSTEM_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the SCIF1_RST_SYSTEM_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the SCIF0_RST_SYSTEM_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.196 Reset Monitor Register SCI (CPG_RSTMON_SCI)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the SCI1_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the SCI0_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.197 Reset Monitor Register IRDA (CPG_RSTMON_IRDA)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the IRDA_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.198 Reset Monitor Register RSPI (CPG_RSTMON_RSPI)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	RST4_MON	RST3_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	RST4_MON	1b	R	The state of the RSPI4_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
3	RST3_MON	1b	R	The state of the RSPI3_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the RSPI2_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the RSPI1_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the RSPI0_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.199 Reset Monitor Register CANFD (CPG_RSTMON_CANFD)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the CANFD_RSTC_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the CANFD_RSTP_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.200 Reset Monitor Register GPIO (CPG_RSTMON_GPIO)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	RST2_MON	0b	R	The state of the GPIO_SPARE_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	0b	R	The state of the GPIO_PORT_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	0b	R	The state of the GPIO_RSTN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.201 Reset Monitor Register ADC (CPG_RSTMON_ADC)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the ADC_ADRST_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the ADC_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.202 Reset Monitor Register TSU (CPG_RSTMON_TSU)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	—	1b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	RST0_MON	1b	R	The state of the TSU_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.203 Reset Monitor Register AXI_ACPU_BUS (CPG_RSTMON_AXI_ACPU_BUS)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	0b	R	0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.204 Reset Monitor Register AXI_MCPU_BUS (CPG_RSTMON_AXI_MCPU_BUS)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	0b	R	0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.205 Reset Monitor Register AXI_COM_BUS (CPG_RSTMON_AXI_COM_BUS)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	0b	R	0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	0b	R	0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.206 Reset Monitor Register PERI_COM (CPG_RSTMON_PERI_COM)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	0b	R	0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	0b	R	0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.207 Reset Monitor Register REG1_BUS (CPG_RSTMON_REG1_BUS)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	0b	R	0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.208 Reset Monitor Register REG0_BUS (CPG_RSTMON_REG0_BUS)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	0b	R	0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.209 Reset Monitor Register PERI_CPU (CPG_RSTMON_PERI_CPU)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	0b	R	0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	0b	R	0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.210 eset Monitor Register PERI_DDR (CPG_RSTMON_PERI_DDR)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	0b	R	0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.211 Reset Monitor Register AXI_TZCDDR (CPG_RSTMON_AXI_TZCDDR)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	RST3_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	1b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	RST3_MON	0b	R	The state of the BUS_TZCDDR_ARESET2N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	0b	R	The state of the BUS_TZCDDR_ARESET1N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	0b	R	The state of the BUS_TZCDDR_ARESET0N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	0b	R	The state of the BUS_TZCDDR_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.212 Reset Monitor Register OCTA (CPG_RSTMON_OCTA)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the OCTA_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.213 Reset Monitor Register OTFDE_DDR (CPG_RSTMON_OTFDE_DDR)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the OTFDE_DDR_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the OTFDE_DDR_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.214 Reset Monitor Register OTFDE_SPI (CPG_RSTMON_OTFDE_SPI)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the OTFDE_SPI_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the OTFDE_SPI_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.215 Reset Monitor Register PDM (CPG_RSTMON_PDM)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2, 1	—	11b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	RST0_MON	1b	R	The state of the PDM0_PRESETNT reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.216 Reset Monitor Register PCI (CPG_RSTMON_PCI)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RST6_MON	RST5_MON	RST4_MON	RST3_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	RST6_MON	1b	R	The state of the PCI_RST_LOAD_B reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
5	RST5_MON	1b	R	The state of the PCI_RST_CFG_B reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
4	RST4_MON	1b	R	The state of the PCI_RST_RSM_B reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
3	RST3_MON	1b	R	The state of the PCI_RST_PS_B reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the PCI_RST_GP_B reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the PCI_RST_B reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the PCI_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.217 Reset Monitor Register SPDIF (CPG_RSTMON_SPDIF)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the SPDIF_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.218 Reset Monitor Register I3C (CPG_RSTMON_I3C)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the I3C_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the I3C_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.219 Reset Monitor Register VBAT (CPG_RSTMON_VBAT)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	—	1b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	RST0_MON	1b	R	The state of the VBAT_BRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.220 WDT Overflow System Reset Register (CPG_WDTOVF_RST)

After the release from a system reset applied by the WDT reset circuit in response to a reset request such as a WDT overflow, this register can be used to identify the WDT channel that generated the reset request. If a system reset is not applied according to the WDTRSTSEL register setting, the WDT is not reset and the source of the reset request can be identified by checking the interrupt from the WDT and the status register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTOVF2_WEN	WDTOVF1_WEN	WDTOVF0_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTOVF2	WDTOVF1	WDTOVF0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
18	WDTOVF2_WEN	0b	R0W1	Flag for enabling the writing to the WDTOVF2 bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	WDTOVF1_WEN	0b	R0W1	Flag for enabling the writing to the WDTOVF1 bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	WDTOVF0_WEN	0b	R0W1	Flag for enabling the writing to the WDTOVF0 bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2	WDTOVF2	0b	RW	The system reset generated by WDT channel 2 for the Cortex-M33_FPU is indicated. 0: System reset has not been generated by WDT channel 2 for the Cortex-M33_FPU. 1: System reset has been generated by WDT channel 2 for the Cortex-M33_FPU. *Note*: This bit is cleared by writing 1.
1	WDTOVF1	0b	RW	The system reset generated by WDT channel 1 for the Cortex-M33 is indicated. 0: System reset has not been generated by WDT channel 2 for the Cortex-M33. 1: System reset has been generated by WDT channel 2 for the Cortex-M33. *Note*: This bit is cleared by writing 1.

Bit	Bit Name	Initial Value	R/W	Description
0	WDTOVF0	0b	RW	The system reset generated WDT channel 0 for Cortex-A55 Core 0 is indicated. 0: System reset has not been generated by WDT channel 0 for Cortex-A55 Core 0. 1: System reset has been generated by WDT channel 0 for Cortex-A55 Core 0. *Note*: This bit is cleared by writing 1.

7.2.4.221 WDT Reset Selector Register (CPG_WDTRST_SEL)

This register is used to mask reset requests from the WDT.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	WDTRSTSEL10_WEN	WDTRSTSEL9_WEN	WDTRSTSEL8_WEN	—	WDTRSTSEL6_WEN	WDTRSTSEL5_WEN	WDTRSTSEL4_WEN	—	WDTRSTSEL2_WEN	WDTRSTSEL1_WEN	WDTRSTSEL0_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R0W1	R0W1	R0W1	R	R0W1	R0W1	R0W1	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	WDTRSTSEL10	WDTRSTSEL9	WDTRSTSEL8	—	WDTRSTSEL6	WDTRSTSEL5	WDTRSTSEL4	—	WDTRSTSEL2	WDTRSTSEL1	WDTRSTSEL0
Initial Value	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
R/W	R	R	R	R	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	WDTRSTSEL10_WEN	0b	R0W1	Flag for enabling the writing to the WDTRSTSEL10 bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	WDTRSTSEL9_WEN	0b	R0W1	Flag for enabling the writing to the WDTRSTSEL9 bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	WDTRSTSEL8_WEN	0b	R0W1	Flag for enabling the writing to the WDTRSTSEL8 bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
22	WDTRSTSEL6_WEN	0b	R0W1	Flag for enabling the writing to the WDTRSTSEL6 bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	WDTRSTSEL5_WEN	0b	R0W1	Flag for enabling the writing to the WDTRSTSEL5 bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	WDTRSTSEL4_WEN	0b	R0W1	Flag for enabling the writing to the WDTRSTSEL4 bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
18	WDTRSTSEL2_WEN	0b	R0W1	Flag for enabling the writing to the WDTRSTSEL2 bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
17	WDTRSTSEL1_WEN	0b	R0W1	Flag for enabling the writing to the WDTRSTSEL1 bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	WDTRSTSEL0_WEN	0b	R0W1	Flag for enabling the writing to the WDTRSTSEL0 bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	WDTRSTSEL10	0b	RW	Whether to mask the reset request by WDT channel 2 is specified. 0: Cold reset of the Cortex-M33_FPU is masked. 1: Cold reset of the Cortex-M33_FPU is enabled.
9	WDTRSTSEL9	0b	RW	Whether to mask the reset request by WDT channel 1 is specified. 0: Cold reset of the Cortex-M33 is masked. 1: Cold reset of the Cortex-M33 is enabled.
8	WDTRSTSEL8	0b	RW	Whether to mask the reset request by WDT channel 0 is specified. 0: Cold reset of the Cortex-A55 is masked. 1: Cold reset of the Cortex-A55 is enabled.
7	—	1b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
6	WDTRSTSEL6	0b	RW	Whether to mask the assertion of WDTOVFOUT due to a request from WDT channel 2 for the Cortex-M33_FPU is specified. 0: Assertion of WDTOVFOUT is masked. 1: Assertion of WDTOVFOUT is enabled.
5	WDTRSTSEL5	0b	RW	Whether to mask the assertion of WDTOVFOUT due to a request from WDT channel 1 for the Cortex-M33 is specified. 0: Assertion of WDTOVFOUT is masked. 1: Assertion of WDTOVFOUT is enabled.
4	WDTRSTSEL4	0b	RW	Whether to mask the assertion of WDTOVFOUT due to a request from WDT channel 0 for Cortex-A55 Core 0 is specified. 0: Assertion of WDTOVFOUT is masked. 1: Assertion of WDTOVFOUT is enabled.
3	—	1b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2	WDTRSTSEL2	0b	RW	Whether to mask WDTSYSREST[2] due to a request from WDT channel 2 for the Cortex-M33_FPU is specified. 0: System reset is masked. 1: System reset is enabled.
1	WDTRSTSEL1	0b	RW	Whether to mask WDTSYSREST[1] due to a request from WDT channel 1 for the Cortex-M33 is specified. 0: System reset is masked. 1: System reset is enabled.
0	WDTRSTSEL0	0b	RW	Whether to mask WDTSYSREST[0] due to a request from WDT channel 0 for Cortex-A55 Core 0 is specified. 0: System reset is masked. 1: System reset is enabled.

Note: Do not make the settings WDTRSTSEL3 = 1 and WDTRSTSEL7 = 0. If these settings are attempted, a system reset initializes the WDTRSTSEL7 bit to 1 (initial value) and WDTOVFOUT is unexpectedly asserted.

7.2.4.222 Cortex-A55 Cluster Power Status Monitor Register (CPG_CLUSTER_PCHMON)

A handshake through the P-Channel is necessary before a warm reset in the Cortex-A55.

This register is monitored by the CPU for software control of the handshake.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PDENY_MON	PACCEPT_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	PDENY_MON	—	R	Signal indicating the denial of power mode transition in the cluster
0	PACCEPT_MON	—	R	Signal indicating the acceptance of power mode transition in the cluster

7.2.4.223 Cortex-A55 Cluster Power Status Control Register (CPG_CLUSTER_PCHCTL)

A handshake through the P-Channel is necessary before a warm reset in the Cortex-A55.

This register is used for software control of the handshake.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	PSTATE0_SET						
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PREQ_SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22 to 16	PSTATE0_SET	100_1000b	RW	Destination power mode of the cluster 100_1000b: ON 000_0000b: OFF
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PREQ_SET	0b	RW	Signal for indicating a request of power mode transition in the cluster

7.2.4.224 Cortex-A55 Core 0 Power Status Monitor Register (CPG_CORE0_PCHMON)

A handshake through the P-Channel is necessary before a warm reset in the Cortex-A55.

This register is monitored by the CPU for software control of the handshake.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PDENY0_MON	PACCEPT0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	PDENY0_MON	—	R	Signal indicating the denial of power mode transition in the core 0
0	PACCEPT0_MON	—	R	Signal indicating the acceptance of power mode transition in the core 0

7.2.4.225 Cortex-A55 Core 0 Power Status Control Register (CPG_CORE0_PCHCTL)

A handshake through the P-Channel is necessary before a warm reset in the Cortex-A55.

This register is used for software control of the handshake.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	PSTATE0_SET					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PREQ0_SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21 to 16	PSTATE0_SET	00_1000b	RW	Destination power mode of the core 0 00_1000b: ON 00_0001b: OFF (Emulated) 00_0000b: OFF
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PREQ0_SET	0b	RW	Signal for indicating a request of power mode transition in the core 0

7.2.4.226 MSTOP Register ACPU (CPG_BUS_ACPU_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	MSTOP0_ON	0b	RW	The state of BUS_ACPU_MSTOP_MXSRAM_A operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.227 MSTOP Register MCPU1 (CPG_BUS_MCPU1_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSTOP15_ON_WEN	MSTOP14_ON_WEN	MSTOP13_ON_WEN	MSTOP12_ON_WEN	MSTOP11_ON_WEN	MSTOP10_ON_WEN	MSTOP9_ON_WEN	MSTOP8_ON_WEN	MSTOP7_ON_WEN	MSTOP6_ON_WEN	MSTOP5_ON_WEN	MSTOP4_ON_WEN	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSTOP15_ON	MSTOP14_ON	MSTOP13_ON	MSTOP12_ON	MSTOP11_ON	MSTOP10_ON	MSTOP9_ON	MSTOP8_ON	MSTOP7_ON	MSTOP6_ON	MSTOP5_ON	MSTOP4_ON	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	MSTOP15_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP15_ON bit (bit 15) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
30	MSTOP14_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP14_ON bit (bit 14) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
29	MSTOP13_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP13_ON bit (bit 13) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
28	MSTOP12_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP12_ON bit (bit 12) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
27	MSTOP11_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP11_ON bit (bit 11) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
26	MSTOP10_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	MSTOP9_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	MSTOP8_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	MSTOP7_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	MSTOP5_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	MSTOP4_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15	MSTOP15_ON	0b	RW	The state of BUS_MCPU_MSTOP13_MHRSPI_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
14	MSTOP14_ON	0b	RW	The state of BUS_MCPU_MSTOP12_MHRSPI_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
13	MSTOP13_ON	0b	RW	The state of BUS_MCPU_MSTOP11_MHSSIF_3 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
12	MSTOP12_ON	0b	RW	The state of BUS_MCPU_MSTOP10_MHSSIF_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
11	MSTOP11_ON	0b	RW	The state of BUS_MCPU_MSTOP9_MHSSIF_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
10	MSTOP10_ON	0b	RW	The state of BUS_MCPU_MSTOP8_MHSSIF_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
9	MSTOP9_ON	0b	RW	The state of BUS_MCPU_MSTOP7_MHPOE3 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
8	MSTOP8_ON	0b	RW	The state of BUS_MCPU_MSTOP6_MHPOEGD operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

Bit	Bit Name	Initial Value	R/W	Description
7	MSTOP7_ON	0b	RW	The state of BUS_MCPU_MSTOP5_MHPOEGC operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
6	MSTOP6_ON	0b	RW	The state of BUS_MCPU_MSTOP4_MHPOEGB operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
5	MSTOP5_ON	0b	RW	The state of BUS_MCPU_MSTOP3_MHPOEGA operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
4	MSTOP4_ON	0b	RW	The state of BUS_MCPU_MSTOP2_MHGPT operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
3	MSTOP3_ON	0b	RW	The state of BUS_MCPU_MSTOP1_MHSRC operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_MCPU_MSTOP0_MHMTU3A operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_MCPU_MSTOP_MHSPI operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_MCPU_MSTOP_MXSRAM_M operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.228 MSTOP Register MCPU2 (CPG_BUS_MCPU2_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSTOP15_ON_WEN	MSTOP14_ON_WEN	MSTOP13_ON_WEN	MSTOP12_ON_WEN	MSTOP11_ON_WEN	MSTOP10_ON_WEN	MSTOP9_ON_WEN	MSTOP8_ON_WEN	MSTOP7_ON_WEN	MSTOP6_ON_WEN	MSTOP5_ON_WEN	MSTOP4_ON_WEN	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSTOP15_ON	MSTOP14_ON	MSTOP13_ON	MSTOP12_ON	MSTOP11_ON	MSTOP10_ON	MSTOP9_ON	MSTOP8_ON	MSTOP7_ON	MSTOP6_ON	MSTOP5_ON	MSTOP4_ON	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	MSTOP15_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP15_ON bit (bit 15) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
30	MSTOP14_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP14_ON bit (bit 14) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
29	MSTOP13_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP13_ON bit (bit 13) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
28	MSTOP12_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP12_ON bit (bit 12) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
27	MSTOP11_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP11_ON bit (bit 11) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
26	MSTOP10_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	MSTOP9_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	MSTOP8_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	MSTOP7_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	MSTOP5_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	MSTOP4_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15	MSTOP15_ON	0b	RW	The state of BUS_MCPU_MSTOP29_MPTSU operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
14	MSTOP14_ON	0b	RW	The state of BUS_MCPU_MSTOP28_MPADC operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
13	MSTOP13_ON	0b	RW	The state of BUS_MCPU_MSTOP27_MPI2C_3 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
12	MSTOP12_ON	0b	RW	The state of BUS_MCPU_MSTOP26_MPI2C_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
11	MSTOP11_ON	0b	RW	The state of BUS_MCPU_MSTOP25_MPI2C_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
10	MSTOP10_ON	0b	RW	The state of BUS_MCPU_MSTOP24_MPI2C_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
9	MSTOP9_ON	0b	RW	The state of BUS_MCPU_MSTOP23_MPCANFD operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
8	MSTOP8_ON	0b	RW	The state of BUS_MCPU_MSTOP22_MHSCI_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

Bit	Bit Name	Initial Value	R/W	Description
7	MSTOP7_ON	0b	RW	The state of BUS_MCPU_MSTOP21_MHSCI_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
6	MSTOP6_ON	0b	RW	The state of BUS_MCPU_MSTOP20_MHIRDA operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
5	MSTOP5_ON	0b	RW	The state of BUS_MCPU_MSTOP19_MHSCIF_4 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
4	MSTOP4_ON	0b	RW	The state of BUS_MCPU_MSTOP18_MHSCIF_3 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
3	MSTOP3_ON	0b	RW	The state of BUS_MCPU_MSTOP17_MHSCIF_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_MCPU_MSTOP16_MHSCIF_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_MCPU_MSTOP15_MHSCIF_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_MCPU_MSTOP14_MHRSPI_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.229 MSTOP Register PERI_COM (CPG_BUS_PERI_COM_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	MSTOP11_ON_WEN	MSTOP10_ON_WEN	MSTOP9_ON_WEN	MSTOP8_ON_WEN	MSTOP7_ON_WEN	MSTOP6_ON_WEN	MSTOP5_ON_WEN	MSTOP4_ON_WEN	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MSTOP11_ON	MSTOP10_ON	MSTOP9_ON	MSTOP8_ON	MSTOP7_ON	MSTOP6_ON	MSTOP5_ON	MSTOP4_ON	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27	MSTOP11_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP11_ON bit (bit 11) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
26	MSTOP10_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	MSTOP9_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	MSTOP8_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	MSTOP7_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	MSTOP5_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	MSTOP4_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	MSTOP11_ON	0b	RW	The state of BUS_PERI_COM_MSTOP_MXSDHI_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
10	MSTOP10_ON	0b	RW	The state of BUS_PERI_COM_MSTOP_MXPCI operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
9	MSTOP9_ON	0b	RW	The state of BUS_PERI_COM_MSTOP_MXCOM1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
8	MSTOP8_ON	0b	RW	The state of BUS_PERI_COM_MSTOP8_MXCOM operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
7	MSTOP7_ON	0b	RW	The state of BUS_PERI_COM_MSTOP7_MHUSB2_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
6	MSTOP6_ON	0b	RW	The state of BUS_PERI_COM_MSTOP6_MHUSB2_0_F operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
5	MSTOP5_ON	0b	RW	The state of BUS_PERI_COM_MSTOP5_MHUSB2_0_H operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
4	MSTOP4_ON	0b	RW	The state of BUS_PERI_COM_MSTOP4_MPUSBT operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
3	MSTOP3_ON	0b	RW	The state of BUS_PERI_COM_MSTOP3_MPGIGE_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_PERI_COM_MSTOP2_MPGIGE_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_PERI_COM_MSTOP1_MXSDHI_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

Bit	Bit Name	Initial Value	R/W	Description
0	MSTOP0_ON	0b	RW	The state of BUS_PERI_COM_MSTOP0_MXSDHI_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.230 MSTOP Register PERI_CPU (CPG_BUS_PERI_CPU_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSTOP15_ON_WEN	MSTOP14_ON_WEN	MSTOP13_ON_WEN	MSTOP12_ON_WEN	MSTOP11_ON_WEN	MSTOP10_ON_WEN	MSTOP9_ON_WEN	MSTOP8_ON_WEN	MSTOP7_ON_WEN	MSTOP6_ON_WEN	—	MSTOP4_ON_WEN	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSTOP15_ON	MSTOP14_ON	MSTOP13_ON	MSTOP12_ON	MSTOP11_ON	MSTOP10_ON	MSTOP9_ON	MSTOP8_ON	MSTOP7_ON	MSTOP6_ON	—	MSTOP4_ON	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	MSTOP15_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP15_ON bit (bit 15) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
30	MSTOP14_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP14_ON bit (bit 14) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
29	MSTOP13_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP13_ON bit (bit 13) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
28	MSTOP12_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP12_ON bit (bit 12) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
27	MSTOP11_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP11_ON bit (bit 11) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
26	MSTOP10_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	MSTOP9_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	MSTOP8_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	MSTOP7_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	MSTOP4_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15	MSTOP15_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP15_MXREG0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
14	MSTOP14_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP14_MPIM33 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
13	MSTOP13_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP13_MPIA55 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
12	MSTOP12_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP12_MPSRAM_M operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
11	MSTOP11_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP11_MPSRAM_A operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
10	MSTOP10_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP10_MPTZC_3 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
9	MSTOP9_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP9_MPTZC_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
8	MSTOP8_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP8_MPTZC_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

Bit	Bit Name	Initial Value	R/W	Description
7	MSTOP7_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP7_MPTZC_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
6	MSTOP6_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP6_MHGPIO operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
5	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	MSTOP4_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP4_MPCPG operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
3	MSTOP3_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP3_MPSYC operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP2_MPCST operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP1_MXACPU operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP0_MXMCPU operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.231 MSTOP Register PERI_DDR (CPG_BUS_PERI_DDR_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	MSTOP1_ON	0b	RW	The state of BUS_PERI_DDR_MSTOP1_MXMEMC_REG operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_PERI_DDR_MSTOP0_MPPHY operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.232 MSTOP Register REG0 (CPG_BUS_REG0_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	MSTOP11_ON_WEN	MSTOP10_ON_WEN	MSTOP9_ON_WEN	MSTOP8_ON_WEN	MSTOP7_ON_WEN	MSTOP6_ON_WEN	MSTOP5_ON_WEN	MSTOP4_ON_WEN	—	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MSTOP11_ON	MSTOP10_ON	MSTOP9_ON	MSTOP8_ON	MSTOP7_ON	MSTOP6_ON	MSTOP5_ON	MSTOP4_ON	—	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27	MSTOP11_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP11_ON bit (bit 11) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
26	MSTOP10_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	MSTOP9_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	MSTOP8_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	MSTOP7_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	MSTOP5_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	MSTOP4_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
19	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	MSTOP11_ON	0b	RW	The state of BUS_REG0_MSTOP_MPOSTM_7 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
10	MSTOP10_ON	0b	RW	The state of BUS_REG0_MSTOP_MPOSTM_6 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
9	MSTOP9_ON	0b	RW	The state of BUS_REG0_MSTOP_MPOSTM_5 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
8	MSTOP8_ON	0b	RW	The state of BUS_REG0_MSTOP_MPOSTM_4 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
7	MSTOP7_ON	0b	RW	The state of BUS_REG0_MSTOP_MPOSTM_3 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
6	MSTOP6_ON	0b	RW	The state of BUS_REG0_MSTOP_MPOSTM_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
5	MSTOP5_ON	0b	RW	The state of BUS_REG0_MSTOP_MPOSTM_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
4	MSTOP4_ON	0b	RW	The state of BUS_REG0_MSTOP_MPOSTM_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
3	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2	MSTOP2_ON	0b	RW	The state of BUS_REG0_MSTOP_MPWDT_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_REG0_MSTOP_MPWDT_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

Bit	Bit Name	Initial Value	R/W	Description
0	MSTOP0_ON	0b	RW	The state of BUS_REG0_MSTOP_MPWDT_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.233 MSTOP Register REG1 (CPG_BUS_REG1_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	MSTOP6_ON_WEN	MSTOP5_ON_WEN	—	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	MSTOP6_ON	MSTOP5_ON	—	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	MSTOP5_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
7	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
6	MSTOP6_ON	0b	RW	The state of BUS_REG1_MSTOP_MPRSIPG_OTP operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
5	MSTOP5_ON	0b	RW	The state of BUS_REG1_MSTOP_MHRSIPG operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
4	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	MSTOP3_ON	0b	RW	The state of BUS_REG1_MSTOP_MPDMAC_NS operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_REG1_MSTOP_MXDMAC_NS operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_REG1_MSTOP_MPDMAC_S operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_REG1_MSTOP_MXDMAC_S operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.234 MSTOP Register TZCDDR (CPG_BUS_TZCDDR_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2	MSTOP2_ON	0b	RW	The state of BUS_TZCDDR_MSTOP2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_TZCDDR_MSTOP1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_TZCDDR_MSTOP0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.235 MSTOP Register MHU (CPG_MHU_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MHU_MSTOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	R	R	R	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	MHU_MSTOP	0b	RW	The state of MHU_MSTOP operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.236 MSTOP Register MCPU3 (CPG_BUS_MCPU3_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	MSTOP10_ON_WEN	MSTOP9_ON_WEN	MSTOP8_ON_WEN	MSTOP7_ON_WEN	MSTOP6_ON_WEN	MSTOP5_ON_WEN	MSTOP4_ON_WEN	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	MSTOP10_ON	MSTOP9_ON	MSTOP8_ON	MSTOP7_ON	MSTOP6_ON	MSTOP5_ON	MSTOP4_ON	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	MSTOP10_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	MSTOP9_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	MSTOP8_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	MSTOP7_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	MSTOP5_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	MSTOP4_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	MSTOP10_ON	0b	RW	The state of BUS_MCPU_MSTOP_MPI3C operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
9	MSTOP9_ON	0b	RW	The state of BUS_MCPU_MSTOP_MPPDM operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
8	MSTOP8_ON	0b	RW	The state of BUS_MCPU_MSTOP_MPVBATT operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
7	MSTOP7_ON	0b	RW	The state of BUS_MCPU_MSTOP_MHRTC operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
6	MSTOP6_ON	0b	RW	The state of BUS_MCPU_MSTOP_MHSPDIF operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
5	MSTOP5_ON	0b	RW	The state of BUS_MCPU_MSTOP_MHGPT_ELC operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
4	MSTOP4_ON	0b	RW	The state of BUS_MCPU_MSTOP_MHSCIF_5 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
3	MSTOP3_ON	0b	RW	The state of BUS_MCPU_MSTOP_MHRSPI_4 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_MCPU_MSTOP_MHRSPI_3 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_MCPU_MSTOP_MXSRAM_M1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_MCPU_MSTOP_MXOCTA operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.237 MSTOP Register BUS_PERI_CPU2 (CPG_BUS_PERI_CPU2_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	MSTOP9_ON_WEN	MSTOP8_ON_WEN	MSTOP7_ON_WEN	MSTOP6_ON_WEN	MSTOP5_ON_WEN	MSTOP4_ON_WEN	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	MSTOP9_ON	MSTOP8_ON	MSTOP7_ON	MSTOP6_ON	MSTOP5_ON	MSTOP4_ON	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	MSTOP9_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	MSTOP8_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	MSTOP7_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	MSTOP5_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	MSTOP4_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	MSTOP9_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP_MPSRAM_M1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
8	MSTOP8_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP_MPSRAM_A2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
7	MSTOP7_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP_MPSRAM_A1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
6	MSTOP6_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP_MPTZC_6 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
5	MSTOP5_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP_MPTZC_5 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
4	MSTOP4_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP_MPTZC_4 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
3	MSTOP3_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP_MXOTFED_XSPI operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP_MXOTFED_DDR operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP_MPIM33_FPU operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP_MHGPIO_ISO operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.238 Power Down IP Register 1 (CPG_PWRDN_IP1)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PWRDN15_ON_WEN	PWRDN14_ON_WEN	PWRDN13_ON_WEN	PWRDN12_ON_WEN	PWRDN11_ON_WEN	PWRDN10_ON_WEN	PWRDN9_ON_WEN	PWRDN8_ON_WEN	—	PWRDN6_ON_WEN	PWRDN5_ON_WEN	PWRDN4_ON_WEN	PWRDN3_ON_WEN	PWRDN2_ON_WEN	PWRDN1_ON_WEN	PWRDN0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PWRDN15_ON	PWRDN14_ON	PWRDN13_ON	PWRDN12_ON	PWRDN11_ON	PWRDN10_ON	PWRDN9_ON	PWRDN8_ON	—	PWRDN6_ON	PWRDN5_ON	PWRDN4_ON	PWRDN3_ON	PWRDN2_ON	PWRDN1_ON	PWRDN0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	PWRDN15_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN15_ON bit (bit 15) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
30	PWRDN14_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN14_ON bit (bit 14) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
29	PWRDN13_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN13_ON bit (bit 13) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
28	PWRDN12_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN12_ON bit (bit 12) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
27	PWRDN11_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN11_ON bit (bit 11) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
26	PWRDN10_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	PWRDN9_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	PWRDN8_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
22	PWRDN6_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	PWRDN5_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	PWRDN4_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	PWRDN3_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	PWRDN2_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	PWRDN1_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	PWRDN0_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15	PWRDN15_ON	0b	RW	Controls power down for SDHI2 in Power Down mode. 0: Not subject to power down 1: Subject to power down
14	PWRDN14_ON	0b	RW	Controls power down for SDHI1 in Power Down mode. 0: Not subject to power down 1: Subject to power down
13	PWRDN13_ON	0b	RW	Controls power down for SDHI0 in Power Down mode. 0: Not subject to power down 1: Subject to power down
12	PWRDN12_ON	0b	RW	Controls power down for GBE1 in Power Down mode. 0: Not subject to power down 1: Subject to power down
11	PWRDN11_ON	0b	RW	Controls power down for GBE0 in Power Down mode. 0: Not subject to power down 1: Subject to power down
10	PWRDN10_ON	0b	RW	Controls power down for USB_APB in Power Down mode. 0: Not subject to power down 1: Subject to power down
9	PWRDN9_ON	0b	RW	Controls power down for USB0 in Power Down mode. 0: Not subject to power down 1: Subject to power down
8	PWRDN8_ON	0b	RW	Controls power down for USB1 in Power Down mode. 0: Not subject to power down 1: Subject to power down

Bit	Bit Name	Initial Value	R/W	Description
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	PWRDN6_ON	0b	RW	Controls power down for WDT(Ch2) in Power Down mode. 0: Not subject to power down 1: Subject to power down
5	PWRDN5_ON	0b	RW	Controls power down for WDT(Ch0) in Power Down mode. 0: Not subject to power down 1: Subject to power down
4	PWRDN4_ON	0b	RW	Controls power down for IM33_FPU in Power Down mode. 0: Not subject to power down 1: Subject to power down
3	PWRDN3_ON	0b	RW	Controls power down for IA55 in Power Down mode. 0: Not subject to power down 1: Subject to power down
2	PWRDN2_ON	0b	RW	Controls power down for GIC in Power Down mode. 0: Not subject to power down 1: Subject to power down
1	PWRDN1_ON	0b	RW	Controls power down for MHU in Power Down mode. 0: Not subject to power down 1: Subject to power down
0	PWRDN0_ON	0b	RW	Controls power down for SRAM_ACPU in Power Down mode. 0: Not subject to power down 1: Subject to power down

7.2.4.239 Power Down IP Register 2 (CPG_PWRDN_IP2)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	PWRDN4_ON_WEN	PWRDN3_ON_WEN	PWRDN2_ON_WEN	PWRDN1_ON_WEN	PWRDN0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PWRDN4_ON	PWRDN3_ON	PWRDN2_ON	PWRDN1_ON	PWRDN0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	PWRDN4_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	PWRDN3_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	PWRDN2_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	PWRDN1_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	PWRDN0_ON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	PWRDN4_ON	0b	RW	Controls power down for RSIPG in Power Down mode. 0: Not subject to power down 1: Subject to power down
3	PWRDN3_ON	0b	RW	Controls power down for OTFDE_SPI in Power Down mode. 0: Not subject to power down 1: Subject to power down
2	PWRDN2_ON	0b	RW	Controls power down for OTFDE_DDR in Power Down mode. 0: Not subject to power down 1: Subject to power down
1	PWRDN1_ON	0b	RW	Controls power down for AXI_TZCDDR in Power Down mode. 0: Not subject to power down 1: Subject to power down

Bit	Bit Name	Initial Value	R/W	Description
0	PWRDN0_ON	0b	RW	Controls power down for DDR in Power Down mode. 0: Not subject to power down 1: Subject to power down

7.2.4.240 Power Down MSTOP Register (CPG_PWRDN_MSTOP)

It works in conjunction with the Power Down IP register to control the standstill state of the unit in Power Down mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PWRDN_MSTOP_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PWRDN_MSTOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	PWRDN_MSTOP_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN_MSTOP bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PWRDN_MSTOP	0b	RW	Indicates the operation status of the BUS in Power Down mode. 0: Power ON state (MSTOP = 0) 1: Power off (MSTOP = 1)

7.2.4.241 Power Down CLKON Register (CPG_PWRDN_CLKON)

In conjunction with the Power Down IP register, it controls the ON/OFF of each unit clock in Power Down mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PWRDN_CLKON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PWRDN_CLKON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	PWRDN_CLKON_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN_CLKON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PWRDN_CLKON	0b	RW	Controls the operation of the unit clock in Power Down mode. 0: Power ON state (CLK = ON) 1: Power off (CLK = OFF)

7.2.4.242 Power Down RST Register (CPG_PWRDN_RST)

It works in conjunction with the Power Down IP register to control the reset of each unit in Power Down mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PWRDN_RST_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PWRDN_RST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	PWRDN_RST_WEN	0b	R0W1	Flag for enabling the writing to the PWRDN_RST bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PWRDN_RST	0b	RW	Controls the reset pin in Power Down mode. 0: Power ON state (reset release) 1: Power off (reset state)

7.2.4.243 Return Func 1 Register (CPG_RET_FUNC1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RSTB_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	RSTB_WEN	0b	R0W1	Flag for enabling the writing to the RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RSTB	1b	RW	0: Reset ON 1: Reset OFF

7.2.4.244 Return Func 2 Register (CPG_RET_FUNC2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	FUNC_EN		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	FUNC_EN	111b	RW	Make the following settings for each bit. 0: Func OFF 1: Func ON

7.2.4.245 Return Func 3 Register (CPG_RET_FUNC3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STATUS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STATUS	—	R	0: Not completed 1: Completed

7.2.4.246 Other Function Register 2 (CPG_OTHERFUNC2_REG)

This register is used for miscellaneous functions.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RES15_ON_WEN	RES14_ON_WEN	RES13_ON_WEN	RES12_ON_WEN	RES11_ON_WEN	RES10_ON_WEN	RES9_ON_WEN	RES8_ON_WEN	RES7_ON_WEN	RES6_ON_WEN	RES5_ON_WEN	RES4_ON_WEN	RES3_ON_WEN	RES2_ON_WEN	RES1_ON_WEN	RES0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RES15_SET	RES14_SET	RES13_SET	RES12_SET	RES11_SET	RES10_SET	RES9_SET	RES8_SET	RES7_SET	RES6_SET	RES5_SET	RES4_SET	RES3_SET	RES2_SET	RES1_SET	RES0_SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	RESn_ON_WEN [n = 15 to 1]	All 0	R0W1	Flags for enabling the writing to the RESn_SET bits. These bits are always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	RES0_ON_WEN	0b	R0W1	Flag for enabling the writing to the RES0_SET bit (bit 0) (DDR_RST_N). This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	RESn_SET [n = 15 to 1]	All 0	RW	No function is assigned to these bits. When the flags for enabling the writing are set to 1, these bits can be written to. The written value is read from these bits.
0	RES0_SET	0b	RW	The DDR_RST_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

Note: The change to the setting in this register is immediately reflected in the reset signal. Therefore, no register is provided to monitor the result of the change.

7.3 Functions of the CPG

This section describes the functions of the CPG.

7.3.1 Clock Control

The CPG has the following five types of clock control functions.

- PLL control, setting, and monitoring functions
- Clock frequency dividers and selectors
- Fixed frequency dividers
- Clock enabling function

7.3.1.1 PLL Control, Setting, and Monitoring Functions

This section describes the functions for controlling, setting, and monitoring the PLL (PLL1 to PLL4 and PLL6).

The PLL is controlled through the registers in the CPG.

The PLL control module counts the PLL lock wait time to generate the PLL lock signal and it is used to control the PLL output clocks by the CGC so that invalid clocks are not output before the PLL is locked.

The reset state and lock state of the PLL can be monitored through registers in the CPG.

The following shows the PLL control, setting, and monitoring functions.

- PLL operating mode control
- Output clock setting
- PLL reset and lock state monitoring

These functions are described in detail in the following sections.

(1) PLL Operating Mode Control

The CPG registers are used to specify the SSCG PLL operating mode and make SSCG settings.

The normal mode or standby mode can be selected as the operating mode.

The following can be set up through CPG registers.

- Operating mode setting (normal mode or standby mode)
- SSCG enable or disable setting (valid in the normal mode)

The following table shows the registers to be used.

For details of the settings, refer to the functional description of the target register.

Table 7.6 Registers for Controlling the PLL Operating Mode

Register Name	Abbreviation	Function
PLLn (SSCG) standby control registers	CPG_PLLn_STBY (n = 1, 4 or 6)	Standby mode setting (reset state) and SSCG enable or disable setting

(2) Output Clock Setting

The CPG registers are used to specify the clocks output from the PLL.

The following values can be set up.

- Output clock frequency setting
- SSCG modulation value setting

Table 7.7 Registers for Setting PLL Output Clocks

Register Name	Abbreviation	Function
PLLn (SSCG) output clock setting register 1	CPG_PLLn_CLK1	Frequency setting
PLLn (SSCG) output clock setting register 2	CPG_PLLn_CLK2	Frequency setting and SSCG modulation value setting

Note: n = 1, 4 or 6

(3) PLL State Monitoring

The reset state and lock state of the PLL can be monitored through CPG registers.

The following table shows the registers to be used.

Table 7.8 Registers for Monitoring the PLL State

Register Name	Abbreviation	Function
PLLn monitor registers	CPG_PLLn_MON (n = 1, 2, 3, 4 or 6)	PLL state monitoring

(4) SSCG Switching in PLL3

In PLL3, the SSCG function is enabled or disabled through the MD_CLKS signal from the SYSC.

The state of the MD_CLKS signal is determined when the LSI is released from the system reset (PRST#) state and the signal state does not change after the system is activated.

The SSCG enabled or disable state in PLL3 cannot be switched through register settings.

MD_CLKS = 0: SSCG is disabled.

MD_CLKS = 1: SSCG is enabled.

7.3.1.2 Clock Frequency Dividers and Selectors

The CPG uses several frequency dividers and selectors to provide appropriate clocks that meet the specifications of individual modules. The frequency dividers and selectors are classified into the dynamic switching type that can be switched without generating a glitch and the static switching type that should be switched after the clock for the target module is stopped because a glitch is generated if this type of divider or selector is switched while the clock is supplied.

The following frequency dividers and selectors are provided.

- Fixed frequency dividers
- Dynamic switching selectors
- Dynamic switching variable-frequency dividers
  (with a duty ratio of 50:50 when divided by an even number)
- Static switching selectors
- Static switching variable-frequency dividers
  (with a duty ratio of 50:50 when divided by an even number)

A specific procedure should be used to switch each frequency divider or selector. Refer to **Section 7.4.5, Procedure for Switching the Division Ratio of the Dynamic Switching Frequency Dividers**, **Section 7.4.6, Procedure for Switching Clocks by the Dynamic Switching Frequency Selectors**, and **Section 7.4.7, Procedure for Switching Clocks by the Static Switching Selectors**.

7.3.1.3 Fixed Frequency Dividers

A fixed frequency divider uses a fixed division value. The CPG provides 1/2, 1/4, 1/8, 1/10, 1/16, and 1/32 dividers, and all fixed dividers generate clocks with a duty ratio of 50:50.

7.3.1.4 PLL Clock Monitoring

This function monitors whether the PLL1 to PLL4 and PLL6 output clocks are operating.

The frequency in the CPG is divided to obtain the frequency that can be monitored through the external pin of this LSI and then output to the upper layer.

The following table shows the instance names and clock frequencies of the monitor points.

Table 7.9 Instance Names and Clock Frequencies of the Monitor Points

PLL Name	Instance Name of Monitor Point	Maximum Frequency (MHz)	Divider Value	Output Clock Frequency (MHz)
PLL1	CPG_CTL/PLL1/CLKOUT	1,100	64	17.1875
PLL2	CPG_CTL/PLL2/CLKOUT	1,600	64	25
PLL3	CPG_CTL/PLL3/CLKOUT	1,600	64	25
PLL4	CPG_CTL/PLL4/CLKOUT	800	64	12.5
PLL6	CPG_CTL/PLL6/CLKOUT	500	64	7.8125

7.3.2 Clock Generation and Control Functions

The following types of reset are provided.

Table 7.10 Reset Types

No.	Type	Method	Source	Range of Reset
1	System reset	Hardware reset	Reset by PRST#	All
2		WDT system reset	WDT overflow, etc.	All (The WDTOVF register is excluded.)
3	Module reset	WDT system reset	WDT overflow, etc.	CPU corresponding to each WDT
4		Software reset	Control through CPG registers	individual on-chip modules

7.3.2.1 Range of Reset Application

The following shows the application (target) range of the reset generated by each reset source.

(1) Hardware sources (PRST#, TRST#, and DEBUGEN)

(2) WDT system reset sources

Table 7.11 Range of Reset Application

No.	Operating Mode DEBUGEN	Reset Source: Debug Reset TRST#	Reset Source: System Reset PRST#	Reset Source: WDT Reset	CST-IF: CST_DAP ntrst	CST-IF: CST_DAP npotrst	Range of Reset Application: Debug related (Excludes CST_DAP) Cortex-A55	Range of Reset Application: Debug related (Excludes CST_DAP) Other than Cortex-A55	Range of Reset Application: WDTOVF Register *1	Range of Reset Application: All Unit Other than *1,*2,*3	Description of Reset
							Reset signal for each target range generated by the reset generation circuit				
							Cortex-A55 Debug Reset (rst_CA55DBGn)	CST Reset (rst_DEBUGn)	Reset for WDTOVF Register (rst_WDTOVFn)	System Reset (rst_SYSTEMn)	
1	0	0	0	x	Reset	Reset	Reset	Reset	Reset	Reset	Hardware system reset
2	0	0	1	0	Reset	Reset	Reset*4	Reset	Released	Reset*3	WDT system reset
3	0	0	1	1	Reset	Reset	Released	Reset	Released	Released	Normal operating mode
4	0	1	0	x	—	—	—	—	—	—	DEBUG = 0 & TRST# = 1 IS PROHIBITED
5	0	1	1	0	—	—	—	—	—	—	DEBUG = 0 & TRST# = 1 IS PROHIBITED
6	0	1	1	1	—	—	—	—	—	—	DEBUG = 0 & TRST# = 1 IS PROHIBITED
7	1	0	0	x	Reset	Reset	Reset	Reset	Reset	Reset	Hardware system reset
8	1	0	1	0	Reset	Reset*3	Reset*4	Reset*3	Released	Reset*3	WDT system reset
9	1	0	1	1	Reset	Released	Released	Released	Released	Released	Debug mode (with DAP reset)
10	1	1	0	x	Reset	Reset	Reset	Reset	Reset	Reset	Debug mode, Hardware system reset
11	1	1	1	0	Reset*3	Reset*3	Reset*4	Reset*3	Released	Reset*3	Debug mode, WDT system reset
12	1	1	1	1	Released	Released	Released	Released	Released	Released	Debug mode (No DAP Reset)

Note 1. Debug-related circuits: CoreSight/Cortex-A55 debug-related circuits (Cortex-M33 is not applicable)

Note 2. Reset when the system reset is enabled by the WDTRSTSEL register.

Note 3. Reset when the Cortex-A55 reset is enabled by the WDTRSTSEL register.

7.3.2.2 System Reset (External Pin)

A system reset is applied when the reset signal from the external PRST# pin is asserted and it initializes the entire LSI.

7.3.2.3 WDT System Reset

This LSI has three WDT channels and the CPG issues a reset request when any of the following sources is generated.

WDT Channel	Target of Monitoring	Function
Channel 0	Cortex-A55 Core 0	Reset due to a parity error in Cortex-A55 Core 0
Channel 1	Cortex-M33	Reset due to the LOCK signal in Cortex-M33
Channel 2	Cortex-M33_FPU	Reset due to the LOCK signal in Cortex-M33_FPU

(1) WDT System Reset Circuit

The reset requests from three WDT channels can be assigned to the desired signals shown below by using the masking function specified through the settings of the WDTRSTSEL register implemented in the CPG.

If a bit in the WDTRSTSEL register is set to 1, the corresponding signal is asserted when the WDT issues a reset request.

Target of Monitoring	Function
WDT system reset	The reset request signal WDTRST# is sent to the reset generation circuit to apply a system reset.
WDTOVF_PERROUT	The WDTOVF_PERR signal is asserted to notify the external devices of the application of a system reset.
Cortex-A55 cold reset	A cold reset request is issued when either WDT channel 0 detects an error.
Cortex-M33 cold reset	A cold reset request is issued when either WDT channel 1 or 2 detects an error.

Bit	Bit Name	Initial Value	R/W	Reset Source Module	Function
0	WDTRSTSEL0	0	R/W	WDT (channel 0) Cortex-A55 Core 0	When the corresponding WDT issues a reset request due to an overflow, etc. 0: WDT system reset is masked. 1: WDT system reset is applied.
1	WDTRSTSEL1	0	R/W	WDT (channel 1) Cortex-M33	
2	WDTRSTSEL2	0	R/W	WDT (channel 2) Cortex-M33_FPU	
3	Reserved	1	R	—	—
4	WDTRSTSEL4	0	R/W	WDT (channel 0) Cortex-A55 Core 0	When the corresponding WDT issues a reset request due to an overflow, etc. 0: Assertion of WDTOVFOUT is masked. 1: WDTOVFOUT is asserted.
5	WDTRSTSEL5	0	R/W	WDT (channel 1) Cortex-M33	
6	WDTRSTSEL6	0	R/W	WDT (channel 2) Cortex-M33_FPU	
7	Reserved	1	R	—	—
8	WDTRSTSEL8	0	R/W	WDT (channel 0) Cortex-A55 Core 0	When WDT channel 0 issues a reset request*1,*2 0: Cold reset of the Cortex-A55 is masked. 1: Cold reset of the Cortex-A55 is applied.
9	WDTRSTSEL9	0	R/W	WDT (channel 1) Cortex-M33	When WDT channel 1 issues a reset request*3 0: Mask paired Cortex-M33_FPU Cold-Reset. 1: Cold reset of the Cortex-M33 is applied.
10	WDTRSTSEL10	0	R/W	WDT (channel 2) Cortex-M33_FPU	When WDT channel 2 issues a reset request*3 0: Mask paired Cortex-M33 Cold-Reset. 1: Cold reset of the Cortex-M33_FPU is applied.

Note 1. When an error occurs in Core 0 in the Cortex-A55, a cold reset is applied to the entire Cortex-A55.

Note 2. For the timing of the cold reset in the Cortex-A55, refer to **Section 7.3.2.3(4), Timing of Cortex-A55 Cold Reset due to the WDT Source**.

Note 3. For the timing of the cold reset in the Cortex-M33, refer to **Section 7.3.2.3(5), Timing of Cortex-M33 Cold Reset due to the WDT Source**.

(2) WDTOVF_PERROUT Signal

This is an external signal that notifies the system that the WDT has applied a WDT system reset to this LSI.

If another WDT system reset is requested before the output of this signal pulse is completed, it is ignored.

The GPIO settings of the WDTOVF_PERROUT# pin such as the driving capability are reset by the GPIO_RST_WDTOVFN signal for the CPG_WDTOVFRST[3:0] register bits.

The following shows the sequence of reset by the WDT.

(1) The WDT channel selected by the WDTRSTSEL[6:4] bits generates a reset request.

(2) The CPG applies a WDT system reset.

(3) 64 cycles of the 24-MHz clock are waited.
* This is the period until the necessary settings are reflected in the entire LSI.

(4) A low pulse is output from the WDTOVF_PERROUT pin for 64 cycles of the 24-MHz clock.

(5) 64 cycles of the 24-MHz clock are waited.

(6) The WDT system reset signal is deasserted and the boot sequence is performed.

(3) WDT Overflow System Reset Register (CPG_WDTOVF_RST)

After the release from the WDT system reset applied by the WDT system reset circuit in response to a reset request such as a WDT overflow, this register is used to identify the WDT channel that generated the reset request.

This register is not reset by the WDT system reset generated by the WDT.

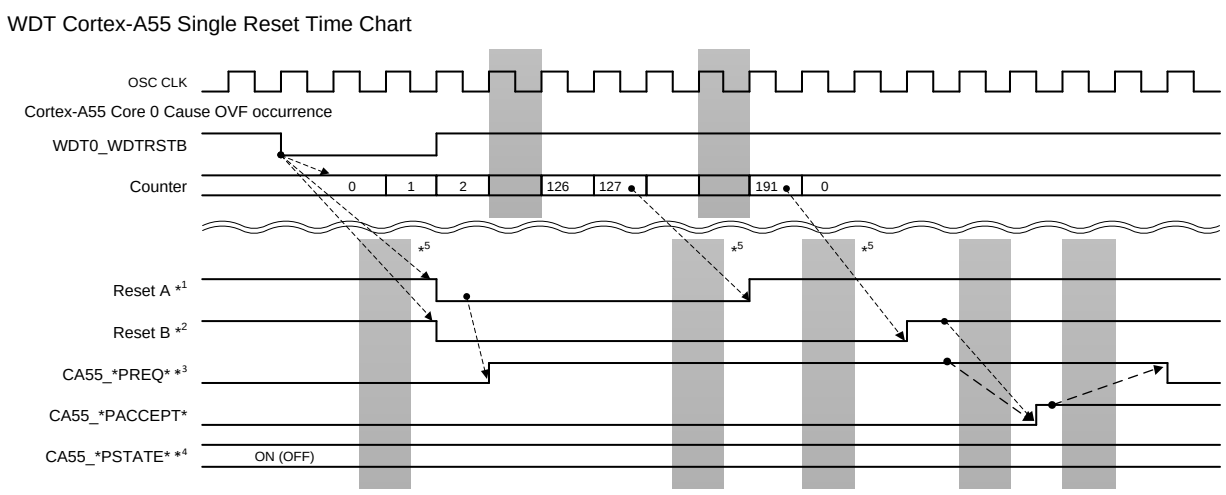
If a WDT system reset is not applied according to the WDTRSTSEL register setting, the WDT is not reset and the source of the reset request can be identified by checking the interrupt from the WDT and the status register in the WDT.

Table 7.12 WDT Overflow System Reset Register (CPG_WDTOVF_RST)

Bit	Bit Name	Initial Value	R/W	Reset Source Module	Function
0	WDTOVF0	0	R/W	WDT (channel 0) Cortex-A55 Core 0	When a WDT channel has applied a WDT system reset, the corresponding bit is set to 1. These bits are cleared by writing 1.
1	WDTOVF1	0	R/W	WDT (channel 1) Cortex-M33	
2	WDTOVF2	0	R/W	WDT (channel 2) Cortex-M33_FPU	
15 to 3	Reserved	0	R	—	—
16	WDTOVF0_WEN	0	R0W1	—	Writing to the WDTOVF0 bit (bit 0) is masked. This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	WDTOVF1_WEN	0	R0W1	—	Writing to the WDTOVF1 bit (bit 1) is masked. This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	WDTOVF2_WEN	0	R0W1	—	Writing to the WDTOVF2 bit (bit 2) is masked. This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
31 to 19	Reserved	0	R	—	—

(4) Timing of Cortex-A55 Cold Reset due to the WDT Source

The following figure shows the timing of the cold reset for the Cortex-A55 alone due to the WDT source.



Note 1. CA55_RST10/CA55_RST11/CA55_RST12

Note 2. CA55_RST0/CA55_RST3/CA55_RST4/CA55_RST5/CA55_RST6/CA55_RST7/CA55_RST8/CA55_RST9

Note 3. The SYSREG setting must be set to ‘0’.

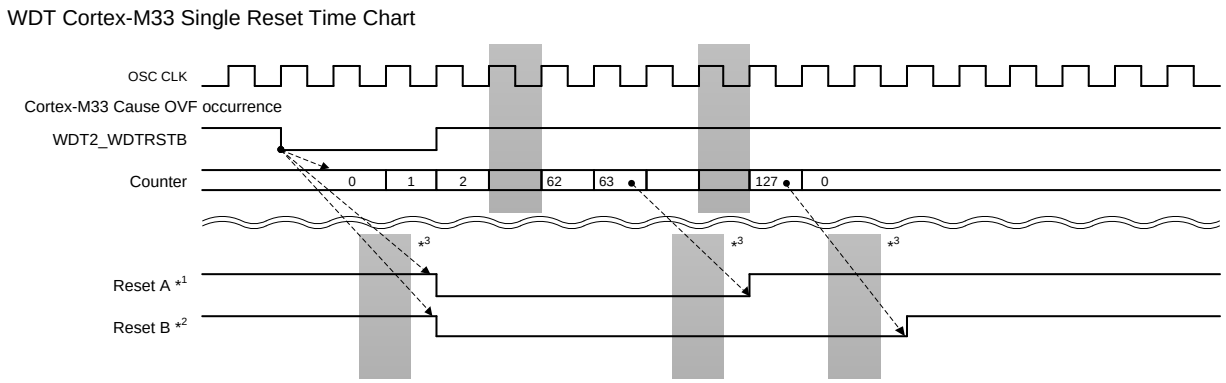
Note 4. The SYSREG setting is ‘ON’ for Core 0/Cluster.

Note 5. Cortex-A55 reset generation has a certain time lag (approximately 35CLK@24MHz).

Figure 7.4 Timing of Cold Reset for the Cortex-A55 Alone due to the WDT Source

(5) Timing of Cortex-M33 Cold Reset due to the WDT Source

The following figure shows the timing of the cold reset for the Cortex-M33 alone due to the WDT source.For Cortex-M33_FPU operation, change the signal name etc. from CM33_* to CM33_FPU_*.



Note 1. CM33_MISCRESETN

Note 2. CA55_NPORESET/CM33_NSYSRESET

Note 3. Cortex-M33 reset generation has a certain time lag (approximately 35CLK@24MHz)

Figure 7.5 Timing of Cold Reset for the Cortex-M33 Alone due to the WDT Source

7.3.2.4 Cortex-A55 Reset

The following four types of reset corresponding to the target ranges of reset are provided for the Cortex-A55.

When the P-Channel circuit receives a request for a warm reset ((2), (3) below), the P-Channel control is required before the application of a reset to the Cortex-A55.

(1) Cold reset
(2) Cluster warm reset
(3) Core 0 Warm reset

When a request to reset the Cortex-A55 alone is generated in the WDT, an error has probably occurred in the Cortex-A55, and the application of a warm reset may be difficult because the warm reset requires a handshake through the P-Channel. Therefore, a cold reset is applied to the Cortex-A55 instead of a warm reset.

- A warm reset for the Cortex-A55 is controlled by software executed in the Cortex-M33.
- After a cold reset (system reset) (normal), Cortex-A55 Core 0 is automatically activated by hardware.
- Refer to **Section 7.3.2.3(4), Timing of Cortex-A55 Cold Reset due to the WDT Source** for the operation of a cold reset (reset of a single core requested from the WDT). Reset control from CPG is performed by hardware.

(1) P-Channel Control for Cortex-A55 Reset

A handshake through the P-Channel is necessary before the application of a warm reset to the Cortex-A55. The registers in the CPG are monitored by the CPU for software control of the handshake. Three types of P-Channel are provided corresponding to the warm reset types. The CPG incorporates the registers (shown in the next page) for the following P-Channel signals and performs a handshake triggered by the PREQ signal. After the handshake, the reset signals for the target of the warm reset should be asserted and then negated by software to apply a warm reset.

Only the negation of the PREQ signal at the release of a cold reset (a system reset or a cold reset from the WDT) is controlled by the CPG hardware.

NOTES

1. The LSI that does not incorporate the Cortex-M33 does not support the warm reset of Cortex-A55.
2. When controlling the P-Channel by software, mask the cold reset of a single core of the Cortex-A55 from the WDT so that the cold reset is not applied. The system reset from the WDT does not need to be masked (the WDT system reset takes priority over the warm reset).
3. When enabling the cold reset of a single core of the Cortex-A55 from the WDT, make the “ON” setting through the PSTATE signal in the corresponding channel so that the PREQ signal is automatically controlled after the release from the reset state.

(a) P-Channel Control Registers in the Cortex-A55

The Cortex-A55 provides a P-Channel for power control for each of the regions — that is, the cluster region including the L3 and SCU, Core 0 region.

The P-Channel is driven in the PERIPHCLK domain, and the CPG contains the registers for controlling and monitoring the P-Channel signals other than the PACTIVE signal.

The CPG performs synchronization for the PACCEPT and PDENY signals.

For the PREQ and PSTATE signals, the Cortex-A55 has a synchronization circuit; the CPG does not perform synchronization.

For the P-Channel control registers, refer to following sections.

Section 7.2.4.222, Cortex-A55 Cluster Power Status Monitor Register (CPG_CLUSTER_PCHMON)

(b) P-Channel Handshake Control

The following describes the operation of the handshake through the P-Channel in the Cortex-A55.

Cold Reset (System Reset)

A handshake is necessary for a transition to the “ON” state.

PREQ = 1 and PSTATE = ON in the initial state. After the release from the reset state, a handshake will be performed automatically and Core 0 will be activated.

The CPG automatically clears PREQ for Core 0 to 0 after the release from the reset state.

Cold Reset (Cold Reset of the Cortex-A55 Alone Requested from the WDT)

A handshake is necessary for a transition to the “ON” state.

When the cold reset of the Cortex-A55 alone requested from the WDT is enabled, set PREQ to 0 and PSTATE to “ON” in the register in advance. After the release from the reset state, a handshake is automatically executed to activate Core 0.

The CPG automatically clears PREQ for Core 0 to 0 after the release from the reset state.

Warm Reset

Although the CPG has P-Channel control registers, it does not automatically control the P-Channel when a warm reset is requested; the user should control the P-Channel by software.

Figure 7.6 shows a flowchart and **Figure 7.7** shows a timing chart of the handshake for a warm reset through the Cortex-A55 P-Channel.

To execute the handshake, the user should manipulate registers in the CPG by software.

- For a single-core Cortex-A55:

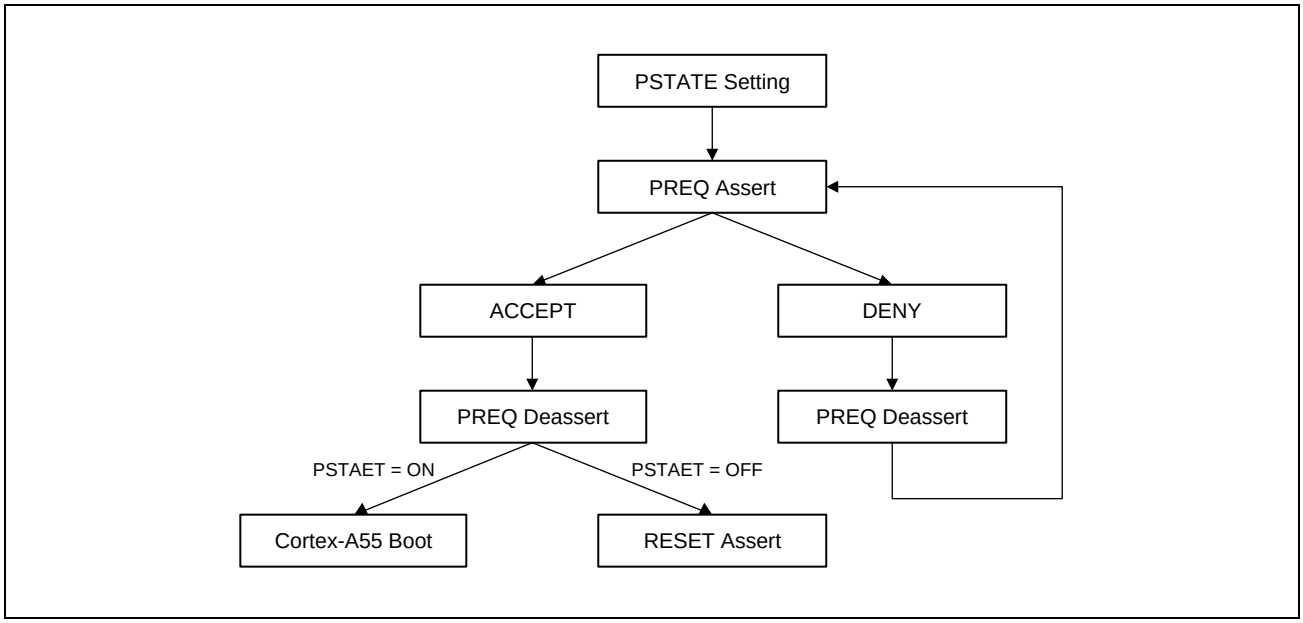


Figure 7.6 Flowchart of a Handshake through the Cortex-A55 P-Channel

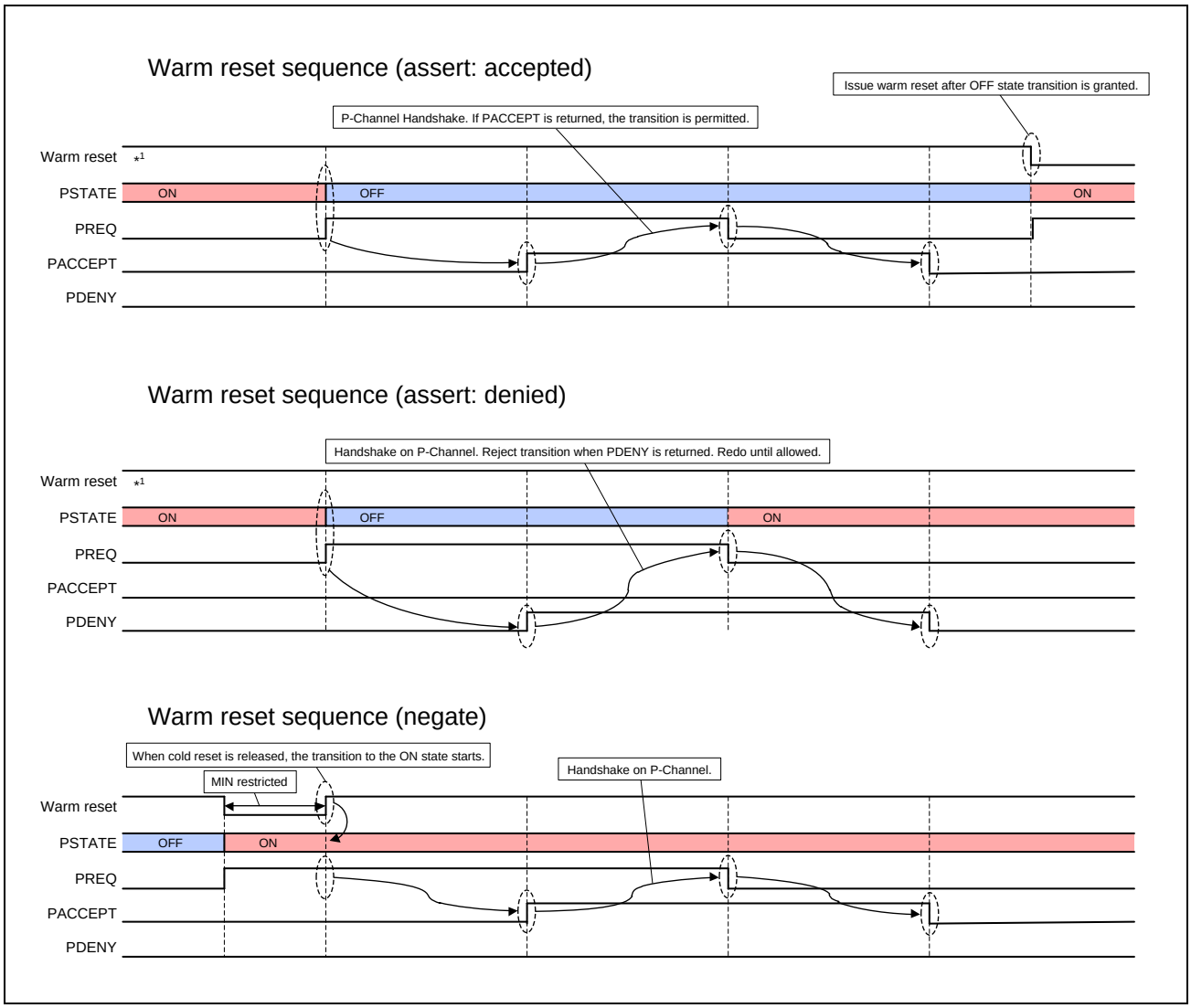
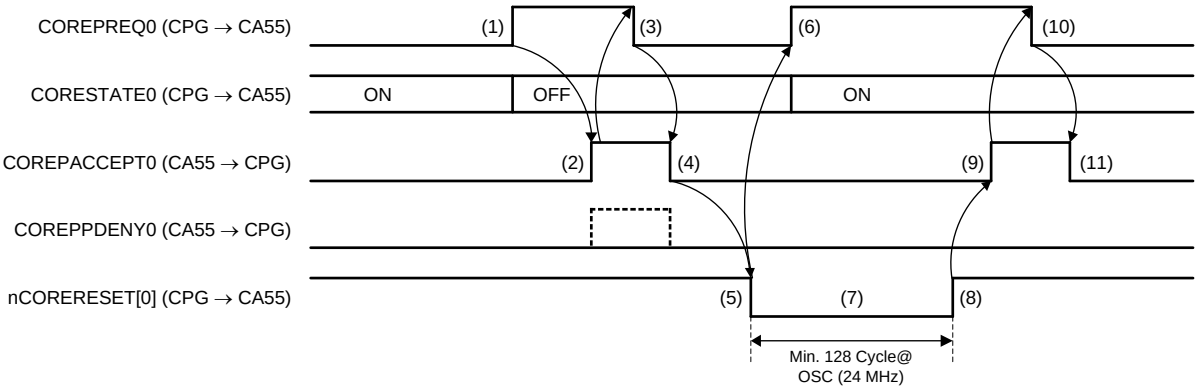


Figure 7.7 Timing Chart of a Handshake through the Cortex-A55 P-Channel (Warm Reset)

(c) Cortex-A55 Warm Reset Sequence

The following describes the warm reset sequence through the P-Channel.

Figure 7.8 shows a waveform when a warm reset of Cortex-A55 Core 0 is requested and accepted. When the request is denied, steps 1 to 4 are repeated until the request is accepted. The same sequence is used for the warm reset of the cluster.



(1) The Cortex-M33 asserts COREPREQ0 according to CPG register settings (PSTATE = OFF).

(2) The Cortex-A55 asserts COREPACCEPT0.

(3) The Cortex-M33 deasserts COREPREQ0 according to CPG register settings.

(4) The Cortex-A55 deasserts COREPACCEPT0.

(5) The Cortex-M33 asserts the target reset signal (nCORERESET[0]) assigned to the warm reset of Cortex-A55 Core 0 according to CPG register settings.

(6) The Cortex-M33 asserts COREPREQ0 according to CPG register settings (PSTATE = ON).

(7) The Cortex-M33 waits for a certain period after the assertion of the reset signal.
18 cycles of the Cortex-A55 clock or 128 cycles at 24 MHz

(8) The Cortex-M33 deasserts the target reset signal (nCORERESET[0]) assigned to the warm reset of Cortex-A55 Core 0 according to CPG register settings.

(9) The Cortex-A55 asserts COREPACCEPT0.

(10) The Cortex-M33 deasserts COREPREQ0 according to CPG register settings.

(11) The Cortex-A55 deasserts COREPACCEPT0.

Figure 7.8 Cortex-A55 Warm Reset Sequence

(2) Operation of the Cortex-A55 Reset Control Circuit

The reset control circuit asserts the reset signal assigned to the reset request from the reset generator or software.

Reset Request Source	Reset Signal Assertion	Reset Signal Deassertion
System reset (including the reset from the WDT)	The CPG asserts the reset signal assigned to the cold reset.	The reset sequencer (boot sequencer) in the CPG releases the reset terminal assigned to the cold reset from the reset state. (The software reset register bit that corresponds to the reset terminal assigned to the cold reset indicates the reset released state in the initial state.)
Cold reset (software reset)*1	The Cortex-M33 asserts the reset signal assigned to the cold reset.	The Cortex-M33 deasserts the asserted signal so that the timing described in **Section 7.3.2.4(3), Timing of Cortex-A55 Reset**.
Cold reset (WDT)	The CPG asserts the reset signal assigned to the cold reset.	The CPG deasserts the asserted signal so that the timing described in **Section 7.3.2.3(4), Timing of Cortex-A55 Cold Reset due to the WDT Source**, is satisfied.
Cluster warm reset	The Cortex-M33 asserts the reset signal assigned to the cluster warm reset.	The Cortex-M33 deasserts the asserted signal after a period of 128 cycles at 24 MHz has elapsed.
Core 0 warm reset	The Cortex-M33 asserts the reset signal assigned to the Core 0 warm reset.	The Cortex-M33 deasserts the asserted signal after a period of 128 cycles at 24 MHz has elapsed.
Software reset*1	A software reset is applied by asserting the reset signal according to the data written to the register by software.	The CPG does not automatically deassert the reset signal. The reset signal retains the asserted state until the next access to the registers.

Note 1. The target reset signal should be controlled by software.

(3) Timing of Cortex-A55 Reset

There is a restriction on the timing of the release of the Cortex-A55 from the reset state.

When a reset is controlled by software, the restriction should be satisfied by software.

Hardware control is done for the system reset (including the reset from the WDT source) and the Cortex-A55 cold reset from the WDT.

The reset signals are classified into two groups:

Group (A): nCPUPORESET, nCORERESET, nSPORESET, nSRESET, nPRESET, nATRESET, nGICRESET, and nPERIPHRESET

Group (B): nARESET, nMISCRESET, and nPDBGRESET

To satisfy the restriction, the CPG deasserts group (B) after a period of 64 cycles at 24 MHz has elapsed since the deassertion of group (A).

7.3.2.5 Cortex-M33 Reset

Two types of reset corresponding to the target ranges of reset are provided for the Cortex-M33.

When a request to reset the Cortex-M33 alone is generated in the WDT, an error has probably occurred in the Cortex-M33, and the application of a warm reset may be difficult because the warm reset requires a handshake through the P-Channel. Therefore, a cold reset is applied to the Cortex-M33 instead of a warm reset.

- For the warm reset, refer to **Section 7.3.2.5(1), Handshake Control for Cortex-M33 Reset**, and **Section 7.3.2.5(2), Cortex-M33 Reset Sequence**. The reset control from the CPG is performed by hardware.
- To release the Cortex-M33 from the cold reset (system reset) state, refer to **Section 7.3.2.5(4), Timing of Cortex-M33 Reset**, and satisfy the described restriction. The reset control from the CPG needs software operation.
- The reset control from the CPG is performed by hardware.

MISCRESETn is a reset signal for the sub-system (SS) layer. This signal is used for a cold reset.
(For details, refer to the restriction on the timing of the Cortex-M33 cold reset.)

(1) Handshake Control for Cortex-M33 Reset

Like the warm reset for the Cortex-A55, the Cortex-M33 requires a handshake process to control the warm reset. After SYSRESETREQ is set to 1 in the Cortex-M33 and the Cortex-M33 enters the WFI state, the CPG executes a handshake with the Cortex-M33 to apply a warm reset to the Cortex-M33.

(2) Cortex-M33 Reset Sequence

For a warm reset for the Cortex-A55, the reset request signal nSYSRESET is issued as a result of the handshake between the CPG and Cortex-M33 and the signal pulse is generated for 128 cycles of the 24-MHz clock after the assertion of the signal.

The nSYSRESET signal is asserted under the conditions of SYSRESETREQ = high and SLEEPING = high.

(3) Operation of the Cortex-M33 Reset Control Circuit

The reset control circuit asserts the reset signal assigned to the reset request from the reset generator or software.

Reset Request Source	Reset Signal Assertion	Reset Signal Deassertion
System reset (including the reset from the WDT)	The software reset register is set to the initial value (initial value = cold reset).	The reset state is retained until the Cortex-A55 releases it.
Cold reset (software reset)*1	The Cortex-A55 asserts the following signals assigned to the cold reset. nPORESET nSYSRESET MISCRESTn	The Cortex-A55 deasserts MISCRESETn after a period of 64 cycles at 24 MHz has elapsed since the assertion of MISCRESETn. nSYSRESET and nPORESET are deasserted after a period of 64 cycles at 24 MHz has elapsed since the deassertion of MISCRESETn.
Cold reset (WDT)	The CPG asserts the following signals assigned to the cold reset. nPORESET nSYSRESET MISCRESTn	The CPG deasserts MISCRESETn after a period of 64 cycles at 24 MHz has elapsed since the assertion of MISCRESETn. nSYSRESET and nPORESET are deasserted after a period of 128 cycles at 24 MHz has elapsed since the deassertion of MISCRESETn. (See **Figure 7.5, Timing of Cold Reset for the Cortex-M33 Alone due to the WDT Source**.)
Warm reset	The CPG asserts the nSYSRESET signal assigned to the Cortex-M33 warm reset according to the procedure of Cortex-M33 warm reset.	The CPG deasserts the asserted signal after a period of 128 cycles at 24 MHz has elapsed.
Software reset*1	A software reset is applied by asserting the reset signal according to the data written to the register by software.	The reset signal is not deasserted. The reset signal retains the asserted state until the next access to the registers.

Note 1. The target reset signal should be controlled by software.

(4) Timing of Cortex-M33 Reset

There is a restriction on the timing of the release of the Cortex-M33 from the reset state.

When a reset is controlled by software, the restriction should be satisfied by software.

Hardware control is done for the Cortex-M33 cold reset from the WDT.

The reset signals are classified into two groups:

Group (A): MISCRESETn

Group (B): nPORESET and nSYSRESET

To satisfy the restriction, the CPG deasserts group (B) after a period of 64 cycles at 24 MHz has elapsed since the deassertion of group (A).

7.3.2.6 Module Reset Application Control

The application of a reset is controlled for individual modules.

In addition to a reset applied to individual modules according to register settings by software, some modules are reset in a certain reset mode.

For register setting specifications, refer to the descriptions of the reset control register *** (CPG_RST_***).

For a clock that is controlled by the corresponding CGC and is synchronized with a reset signal, the processing for stopping the clock is performed at the release of the reset state.

7.4 Operating Procedures

This section describes the procedures for operating the CPG.

7.4.1 Procedures for Supplying and Stopping Module Clocks

Use the following procedures to switch between the supply and stop of the target clock for a module.

* Sample procedures for the SRAM_MCPU clock (SRAM_MCPU_ACLK0) are described here.

Supplying a Clock:

1. Setting the register for supplying the SRAM_MCPU clock (SRAM_MCPU_ACLK0)

   Set up the following bits in the clock control register SRAM_MCPU (CPG_CLKON_SRAM_MCPU).
   — Setting for supplying the clock: Bit 0 (CLK0_ON) = 1 and bit 16 (CLK0_ONWEN) = 1

2. Confirming that the output of the SRAM_MCPU clock (SRAM_MCPU_ACLK0) has started

   Read the following bit in the clock monitor register SRAM_MCPU (CPG_CLKMON_SRAM_MCPU) to confirm that the output of the clock has started.
   — Clock state: Bit 0 (UNIT0_CLK_MON), 1: Clock is supplied. (0: Clock is stopped.)

Stopping a Clock:

1. Setting the register for stopping the SRAM_MCPU clock (SRAM_MCPU_ACLK0)

   Set up the following bits in the clock control register SRAM_MCPU (CPG_CLKON_SRAM_MCPU).
   — Setting for stopping the clock: Bit 0 (CLK0_ON) = 0 and bit 16 (CLK0_ONWEN) = 1

2. Confirming that the output of the SRAM_MCPU clock (SRAM_MCPU_ACLK0) has been stopped

   Read the following bit in the clock monitor register SRAM_MCPU (CPG_CLKMON_SRAM_MCPU) to confirm that the output of the clock has been stopped.
   — Clock state: Bit 0 (UNIT0_CLK_MON), 0: Clock is stopped. (1: Clock is supplied.)

The following registers are used in the above procedures.

— 7.2.4.x Clock control registers ***** (CPG_CLKON_*****)

— 7.2.4.x Clock monitor registers ***** (CPG_CLKMON_*****)

For the procedures for supplying and stopping a clock signal, refer also to the section on the target module. If specific procedures for the module are described, use them.

7.4.2 Procedures for Supplying and Stopping Reset Signals

Use the following procedures to switch between the supply (reset state) and stop (released from the reset state) of the target reset signal for a module.

* Sample procedures for the SRAM_MCPU reset (SRAM_MCPU_ARESETN0) are described here.

Supplying a Reset Signal:

1. Setting the register for supplying the SRAM_MCPU reset signal (SRAM_MCPU_ARESETN0)

   Set up the following bits in the reset control register SRAM_MCPU (CPG_RST_SRAM_MCPU).
   — Setting for supplying the reset signal: Bit 0 (UNIT0_RSTB) = 1 and bit 16 (UNIT0_RST WEN) = 1

Stopping a Reset Signal:

1. Setting the register for stopping the SRAM_MCPU reset signal (SRAM_MCPU_ARESETN0)

   Set up the following bits in the reset control register SRAM_MCPU (CPG_RST_SRAM_MCPU).

   - Setting for stopping the reset signal: Bit 0 (UNIT0_RSTB) = 0 and bit 16 (UNIT0_RST WEN) = 1

2. Confirming that the output of the SRAM_MCPU reset signal (SRAM_MCPU_ARESETN0) has been stopped

   Read the following bit in the reset monitor register SRAM_MCPU (CPG_RSTMON_SRAM_MCPU) to confirm that the output of the reset signal has been stopped.
   — Reset state: Bit 0 (UNIT0_RST_MON)
   0: Reset signal is stopped (released from the reset state)
   1: Reset signal is output.

The following registers are used in the above procedures.

— 7.2.4.x Reset control registers ***** (CPG_RST_*****)

— 7.2.4.x Reset monitor registers ***** (CPG_RSTMON_*****)

For the procedures for supplying and stopping a reset signal, refer also to the section on the target module. If specific procedures for the module are described, use them.

7.4.3 Procedure for Activating Modules

Use the following procedure to activate an inactive module after a system reset.

For the specific procedure for activating the target module, refer to the section on the module.

1. Set up the clock control register for the clock signal connected to the target module to start the supply of the clock. Note that the PLL for the clock should be started before the clock if the PLL is stopped.
2. Read the clock monitor register to confirm that the supply of the target clock has started.
3. Set up the reset control register for the reset signal connected to the target module to release the module from the reset state.
4. Read the reset monitor register to conform that the module has been released from the reset state.

7.4.4 Procedures for PLL Setup

The following shows the procedures for PLL setup.

7.4.4.1 Procedure for Setting PLL Normal Mode (Changing the Output Clock and SSCG Mode)

Use the following procedure to shift the PLL from the standby mode to the normal mode (change the output clock and SSCG mode).

[n = 1, 4 or 6: PLL1, PLL4 or PLL6]

1) Confirming the PLL operating mode (confirming that the PLL is in the standby mode)

- Check the following bit in the PLLn (SSCG) monitor register (CPG_PLLn_MON).
- Shift to the standby mode: Confirm that bit 0 (PLLn_RESETB) = 0 (standby mode).
- Check the following bit in the PLLn (SSCG) monitor register (CPG_PLLn_MON).
- Unlock of PLL: Confirm that bit 4 (PLLn_LOCK) = 0 (PLL is not locked).

Wait until the above conditions are satisfied.

2) Setting the output clock

- Modify the following bits in PLLn (SSCG) output clock setting register 1 (CPG_PLLn_CLK1).
- For details of the settings, refer to **Section 7.2.4, Register Descriptions**.
- Setting of output clock (desired setting): bit0(RANGESEL), bits 12 to 1 (DIV_NF_11 to DIV_NF_0), bits 21 to 13 (DIV_NI_8 to DIV_NI_0), bits 25 to 22 (DIV_M_3 to DIV_M_0), and bits 28 to 26 (DIV_P_2 to DIV_P_0)

3) Setting the SSCG modulation value

- Modify the following bits in PLLn (SSCG) output clock setting register 2 (CPG_PLLn_CLK2).
- For details of the settings, refer to **Section 7.2.4, Register Descriptions**.
- Setting of SSCG modulation value (desired setting): Bits 2 to 0 (MRR_2 to MRR_0) and bits 7 to 3 (MFR_4 to MFR_0)

4) Setting the PLL normal mode

- Set up the following bits in the PLLn (SSCG) standby control register (CPG_PLLn_STBY).
- Setting of operating mode: Bit 0 (RESETB) = 1 (normal mode)
- Setting of SSCG mode: Bit 2 (SSCG_EN) = Desired setting

5) Confirming the shift to the PLL normal mode and the stable output of the clock

- Check the following bit in the PLLn (SSCG) monitor register (CPG_PLLn_MON).
- Shift to standby mode: Confirm that bit 0 (PLLn_RESETB) = 1 (normal mode).
- Check the following bit in the PLLn (SSCG) monitor register (CPG_PLLn_MON).
- Lock of PLL: Confirm that bit 4 (PLLn_LOCK) = 1 (PLL is locked).

Wait until the above conditions are satisfied.

NOTE

If there is no change in a parameter in step (2) or (3), the corresponding register does not need to be modified.

7.4.4.2 Procedure for Setting PLL Standby Mode

Use the following procedure to shift the PLL from the normal standby mode to the standby mode.

[n = 1, 4 or 6: PLL1, PLL4 or PLL6]

1) Setting the PLL standby mode

- Set the following bits of the PLLn (SSCG) standby control register (CPG_PLLn_STBY).
- Operation mode setting: bit0 (RESETB) = 0 (standby mode) (other bits are “0”).

2) Confirming the shift to the PLL standby mode

- Check the following bits of the PLLn (SSCG) monitor register (CPG_PLLn_MON).
- Standby mode transition confirmation: bit0 (PLLn_RESETB) = 0 (standby mode).
- Check the following bits of the PLLn (SSCG) monitor register (CPG_PLLn_MON).
- Confirmation of PLL lock release: bit4 (PLLn_LOCK) = 0 (PLL is not locked).

Wait until the above conditions are satisfied.

7.4.4.3 PLLs with Restrictions

PLL1 to PLL3, and PLL6 cannot be switched to desired clock settings by the user because these PLLs output clocks for the CPU and bus operation.

PLL1 outputs clocks dedicated to the Cortex-A55. To modify its settings after booting, the registers should be manipulated from the Cortex-M33.

The PLL2 and PLL3 settings cannot be modified after booting because these PLLs output system clocks for bus operation.

PLL6 is the operating clock for Cortex-M33, so if you want to change the settings after booting, you must do so on the Cortex-A55.

PLL4 can be set up, started, or stopped by the user at the desired timing under the allowable conditions of use for the DDR.

7.4.5 Procedure for Switching the Division Ratio of the Dynamic Switching Frequency Dividers

Use the following procedure to set the division ratio of the dynamic switching frequency dividers.

* A sample procedure for DIV_PLL1 is described here.

1) Confirming the DIV_PLL1 state (checking that it is not busy)

- Check the following bit in the clock status monitor register (CPG_CLKDIVSTATUS).
- DIV_PLL1 state: Confirm that bit 0 (DIVPL1_STS)) = 0 (switching is completed).

Wait until the above condition is satisfied.

2) Setting the DIV_PLL1 division ratio

- Set up the following bits in the division ratio setting (PLL1) register (CPG_PL1_DDIV).
- Setting of division ratio: Bits 1 and 0 (DIVPL1_SET) = desired value and bit 16 (DIV_PLL1SET_WEN) = 1

3) Confirming the DIV_PLL1 state (checking that it is not busy)

- Check the following bit in the clock status monitor register (CPG_CLKDIVSTATUS).
- DIV_PLL1 state: Confirm that bit 0 (DIVPL1_STS) = 0 (switching is completed).

Wait until the above condition is satisfied.

The following registers are used in the above procedure.

— Division ratio setting (PLL1) register (CPG_PL1_DDIV)

— Division ratio setting (PLL2) register (CPG_PL2_DDIV)

— Division ratio setting (PLL3) register (CPG_PL3_DDIV)

— Division ratio setting (PLL6) register (CPG_PL6_DDIV)

— Division ratio setting (SDHI) register (CPG_SDHI_DDIV)

— Division ratio setting (OCTA) register (CPG_OCTA_DDIV)

— Division ratio setting (SPI) register (CPG_SPI_DDIV)

The following status monitor register is used.

— Clock status monitor register (CPG_CLKDIVSTATUS)

When 1 is written to a xx_WEN bit placed in the upper 16 bits of a division ratio setting register, the clock switching control begins even if the settings in the lower bits are not changed. Check the status monitor register (CPG_CLKDIVSTATUS) and wait until the switching is completed. The clock will temporarily stop at the timing of switching.

7.4.6 Procedure for Switching Clocks by the Dynamic Switching Frequency Selectors

Use the following procedure to switch clocks by the dynamic switching frequency selectors.

* A sample procedure for SEL_PLL4 is described here.

1) Confirming the SEL_PLL4 state (checking that it is not busy)

- Check the following bit in the clock status monitor register (CPG_CLKDIVSTATUS).
- SEL_PLL4 state: Confirm that bit 6 (SELPLL4_STS) = 0 (switching is completed).

Wait until the above condition is satisfied.

2) Setting the switching of the clock output from SEL_PLL4

- Set up the following bits in the source clock setting (PLL) register (CPG_PLL_DSEL).
- Setting of clocks switching (switch from PLL4 to OSC_DIV1000): Bits 31 to 0 = H’0040_0000
  (switch from OSC_DIV1000 to PLL4): Bits 31 to 0 = H’0040_0040

Confirming the SEL_PLL4 state (checking that it is not busy)

- Check the following bit in the clock status monitor register (CPG_CLKDIVSTATUS).
- SEL_PLL4 state: Confirm that bit 6 (SELPLL4_STS) = 0 (switching is completed).

Wait until the above condition is satisfied.

The following registers are used in the above procedure.

— Source clock setting (PLL) register (CPG_PLL_DSEL)

— Source clock setting (SDHI) register (CPG_SDHI_DSEL)

When 1 is written to a xx_WEN bit placed in the upper 16 bits of a source clock setting register, the clock switching control begins even if the settings in the lower bits are not changed. Check the status monitor register (CPG_CLKDIVSTATUS) and wait until the switching is completed. The clock will temporarily stop at the timing of switching.

7.4.7 Procedure for Switching Clocks by the Static Switching Selectors

Use the following procedure to set the division ratio of the static switching frequency dividers and to switch clocks by the static switching frequency selectors.

When a source clock is active: Stop the clock. → Make settings for switching clocks. → Start the clock.

When a source clock is stopped: Make settings for switching.

The following registers are used in the above procedure.

— Source clock setting (OCTA) register (CPG_OCTA_SSEL)*1

— Source clock setting (SPI) register (CPG_SPI_SSEL)

Note 1. When switching SEL_OCTA in CPG_OCTA_SSEL, please note the following points.

1) Do not access OCTA during switching.
2) If a TZC_XSPI_INT interrupt is occurring, allow the interrupt to complete before stopping the clock on the CPG_CLKON_AXI_MCPU_BUS.
3) Before switching, stop the following clocks.
   - CPG_CLKON_OCTA (bit1, bit0)
   - CPG_CLKON_AXI_MCPU_BUS (bit11)
4) CPG_CLKON_OCTA and CPG_CLKON_AXI_MCPU_BUS are not in the order to stop the clock.

7.4.8 Boot Sequence (Normal Operation)

After the reset sequence ends, perform the following operations if necessary.

- Start PLLs.
- Set up clock selectors.
- Set up clock division ratios.
- Supply clocks to modules.
- Release modules from the reset state.

7.4.9 Boot Sequence (Debug Mode)

After the reset sequence (debug mode) ends, perform the following operations if necessary.

- Start PLLs.
- Set up clock selectors.
- Set up clock division ratios.
- Supply clocks to modules.
- Release modules from the reset state.

7.4.10 General Operating Procedures

The following describes general operating procedures.

7.4.10.1 Procedure for Writing to Registers

The registers in the CPG have write-enable flags in the upper 16 bits and normal data fields in the lower 16 bits except for the registers shown below. Therefore, only the desired bit can be modified without read-modify-write operation.

To modify the value of a bit, write 1 to the write-enable flag having the target bit number + 16 and write a desired value to the target bit.

(Examples: To modify bit 0 to 1b, write H'0001_0001. To modify it to 0b, write H'0001_0000. When H'0000_xxxx is written, the value of bits 15 to 0 does not change.)

NOTE

In some registers, a single write-enable flag controls multiple data bits. Refer to the description of each register.

Registers that have data fields in the upper bits instead of write-enable flags:

— PLLn (SSCG) output clock setting register 1 (CPG_PLLn_CLK1): (n = 1, 4 or 6)

— PLLn (SSCG) output clock setting register 2 (CPG_PLLn_CLK2): (n = 1, 4 or 6)

— Cortex-A55 cluster power status control register (CPG_CLUSTER_PCHCTL)

— Cortex-A55 Core 0 power status control register (CPG_CORE0_PCHCTL)

— Return Func 2 Register (CPG_RET_FUNC2)

7.4.11 Switching the Source Clock Setting (PLL) Register (CPG_PLL_DSEL bit6)

When switching the source clock setting (PLL) register (CPG_ PLL_DSEL bit6), wait at least 800 µs after Cortex-A55 boots during the boot sequence.

8. Interrupt Controller

The interrupt controller decides the priority of interrupt sources and controls interrupt requests to the CPU.

The interrupt controller registers set the order of priority of each interrupt, allowing the user to process interrupt requests according to the user-set priority.

8.1 Features

■ Equipped with Arm® CoreLink™ GIC-600 Generic Interrupt Controller*1 for Arm Cortex-A55.

- 32 levels of priority.
- SGI (Software Generated Interrupt): 16 interrupts
- PPI (Private Peripheral Interrupt): 16 interrupts
- SPI (Shared Peripheral Interrupt): 480 interrupts
- LPI (Locality-specific Peripheral Interrupt): Not supported

■ Nested Vectored Interrupt Controller (NVIC)*2 built in Arm Cortex-M33.

- 256 levels of priority
- 480 interrupts

■ External Interrupts

- NMI, IRQ0-7, TINT0-31
- Noise filter function.*3

■ Internal Interrupts

- On-chip peripheral interrupts
- On-chip Bus error interrupts
- On-chip RAM ECC error interrupts

Note 1. Refer to Arm® CoreLink™ GIC-600 Generic Interrupt Controller Revision: r1p6 Technical Reference Manual.
Note 2. Refer to Arm® Cortex®-M33 Processor Revision: r0p4 Technical Reference Manual.
Note 3. Refer to **Section 45, General Purpose Input Output Port (GPIO)**.

8.1.1 External Signal Pins

The following table shows external signal pins for Interrupt Controller.

Table 8.1 External Signal Pins

Name	Width	I/O	Description
NMI	1	I	NMI (Non Maskable Interrupt) pin. Refer to **Section 8.7.1**.
IRQ0 – IRQ7	8	I	IRQ (Interrupt Request) pins. GPIO pins can be used as IRQ pins. Refer to **Section 8.7.2**.
P0_0 – P18_5	82	I	GPIO pins. GPIO pins can be used as external interrupt input pins (GPIOINT0-81). Assign only 32 pins in GPIOINT0-81 to TINT0-31. Refer to **Section 8.7.3**.

8.1.2 Block Diagram

The block diagram of Interrupt Controller is as follows.

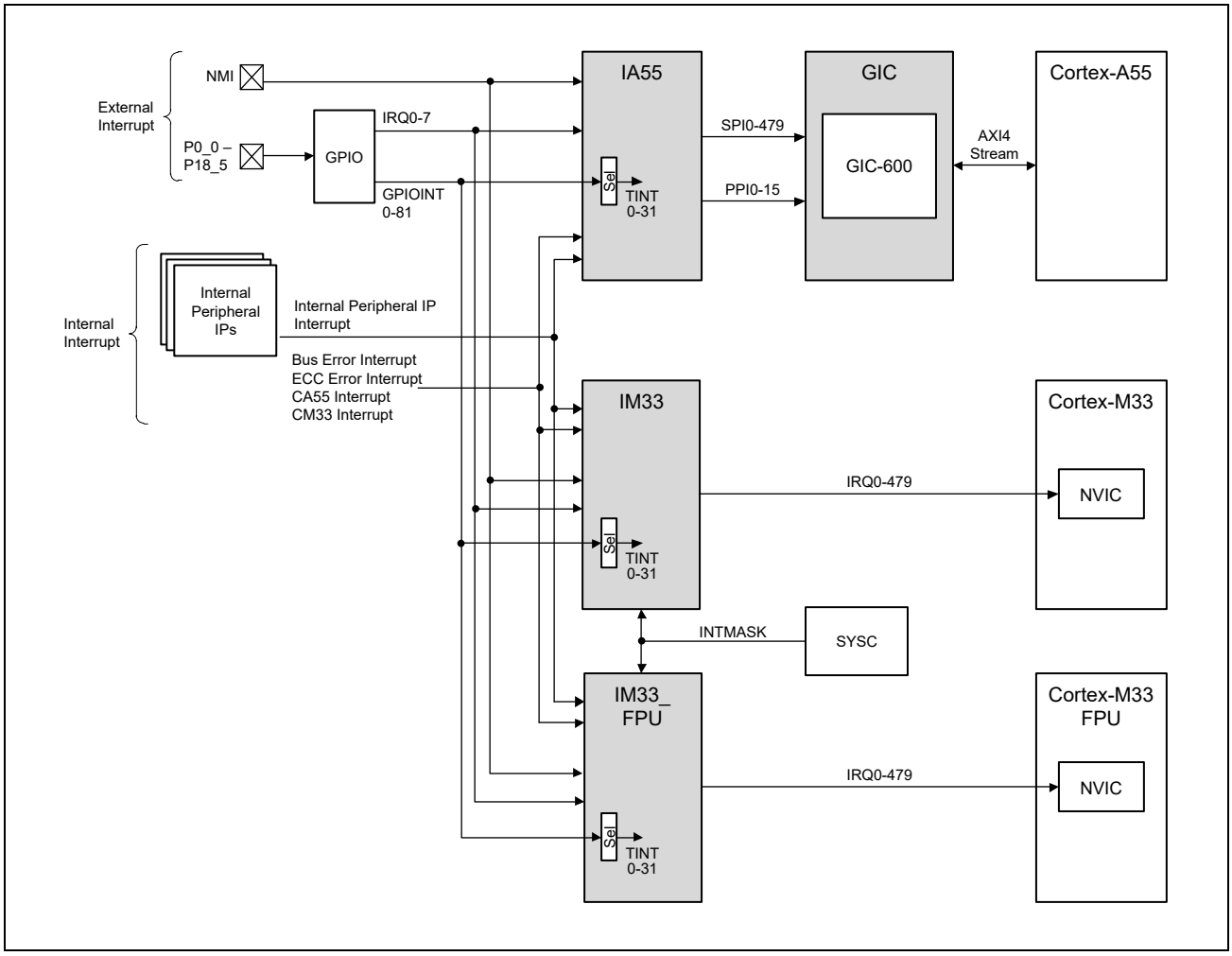


Figure 8.1 Interrupt Controller Block Diagram

GIC

GIC is equipped with Arm® CoreLink™ GIC-600 Generic Interrupt Controller for Cortex-A55.

IA55

IA55 performs various interrupt controls including synchronization for the external interrupts of NMI, IRQ, and GPIOINT and the interrupts of the built-in peripheral interrupts output by each IP. And it notifies the interrupt to the GIC.

- Select 32 TINT from 82 GPIOINT.
- Integration of bus error interrupts from system bus.
- Integration of ECC error interrupts from On-chip RAM.
- Indicate interrupt status. (NMI, IRQ, TINT, integrated bus error interrupt and integrated ECC error interrupt)
- Setting of interrupt detection method. (NMI, IRQ and TINT)

IM33/IM33_FPU

IM33/IM33_FPU performs various interrupt controls including synchronization for the external interrupts of NMI, IRQ and GPIOINT and internal peripheral interrupts output by each IP. And it notifies the interrupt to the built-in interrupt controller (NVIC) for Cortex-M33/Cortex-M33_FPU.

- Select 32 TINT from 82 GPIOINT.
- Integration of bus error interrupts from system bus.
- Integration of ECC error interrupts from On-chip RAM.
- Indicate interrupt status. (NMI, IRQ, TINT, integrated bus error interrupt and integrated ECC error interrupt)
- Set interrupt detection method. (NMI, IRQ and TINT)
- All interrupts can be masked by setting of SYSC Register. (IM33_MASK/IM33FPU_MASK bit of SYS_LP_CLT7 register.)
  Refer to **Section 41, Low Power Mode**.

8.2 Interrupt Mapping

The following table indicates interrupt mapping for Cortex-A55 and Cortex-M33/Cortex-M33_FPU.

Refer to **Section 14, Direct Memory Access Controller** with the interrupts through DMA Controller, and **Section 8.8.1**.

Table 8.2 Interrupt Mapping (1/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
GIC (Software Interrupt)	SGI ID0	0	SGI 0	—	—	—	
	SGI ID1	1	SGI 1	—	—	—	
	SGI ID2	2	SGI 2	—	—	—	
	SGI ID3	3	SGI 3	—	—	—	
	SGI ID4	4	SGI 4	—	—	—	
	SGI ID5	5	SGI 5	—	—	—	
	SGI ID6	6	SGI 6	—	—	—	
	SGI ID7	7	SGI 7	—	—	—	
	SGI ID8	8	SGI 8	—	—	—	
	SGI ID9	9	SGI 9	—	—	—	
	SGI ID10	10	SGI 10	—	—	—	
	SGI ID11	11	SGI 11	—	—	—	
	SGI ID12	12	SGI 12	—	—	—	
	SGI ID13	13	SGI 13	—	—	—	
	SGI ID14	14	SGI 14	—	—	—	
	SGI ID15	15	SGI 15	—	—	—	
Reserved	—	16	PPI 0	—	—	—	
Reserved	—	17	PPI 1	—	—	—	
Reserved	—	18	PPI 2	—	—	—	
Reserved	—	19	PPI 3	—	—	—	
Reserved	—	20	PPI 4	—	—	—	
Reserved	—	21	PPI 5	—	—	—	
Cortex-A55	nCOMIRQ	22	PPI 6	—	—	Level	
	nPMUIRQ	23	PPI 7	—	—	Level	
	CTIIRQ	24	PPI 8	—	—	Level	
	nVCPUMNTIRQ	25	PPI 9	—	—	Level	
	nCNTHPIRQ	26	PPI 10	—	—	Level	
	nCNTVIRQ	27	PPI 11	—	—	Level	
	nCNTHVIRQ	28	PPI 12	—	—	Level	
	nCNTPSIRQ	29	PPI 13	—	—	Level	
	nCNTPNSIRQ	30	PPI 14	—	—	Level	
Reserved	—	31	PPI 15	—	—	—	

Table 8.2 Interrupt Mapping (2/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
NMI pin	NMI	32	SPI 0	IRQ 0	IRQ 0	Level	*4
IRQ pins	IRQ0	33	SPI 1	IRQ 1	IRQ 1	Level	*5
	IRQ1	34	SPI 2	IRQ 2	IRQ 2	Level	*5
	IRQ2	35	SPI 3	IRQ 3	IRQ 3	Level	*5
	IRQ3	36	SPI 4	IRQ 4	IRQ 4	Level	*5
	IRQ4	37	SPI 5	IRQ 5	IRQ 5	Level	*5
	IRQ5	38	SPI 6	IRQ 6	IRQ 6	Level	*5
	IRQ6	39	SPI 7	IRQ 7	IRQ 7	Level	*5
	IRQ7	40	SPI 8	IRQ 8	IRQ 8	Level	*5
Cortex-A55	nERRIRQ[1]	41	SPI 9	IRQ 9	IRQ 9	Level	
	nERRIRQ[0]	42	SPI 10	IRQ 10	IRQ 10	Level	
	nFAULTIRQ[1]	43	SPI 11	IRQ 11	IRQ 11	Level	
	nFAULTIRQ[0]	44	SPI 12	IRQ 12	IRQ 12	Level	
	nCOMMIRQ[0]	45	SPI 13	IRQ 13	IRQ 13	Level	
	nPMUIRQ[0]	46	SPI 14	IRQ 14	IRQ 14	Level	
	nCLUSTERPMUIRQ	47	SPI 15	IRQ 15	IRQ 15	Level	
Cortex-M33	CTIIRQ[1]	48	SPI 16	IRQ 16	IRQ 16	Level	
	CTIIRQ[0]	49	SPI 17	IRQ 17	IRQ 17	Level	
Cortex-M33_FPU	CTIIRQ[1]	50	SPI 18	IRQ 18	IRQ 18	Level	
	CTIIRQ[0]	51	SPI 19	IRQ 19	IRQ 19	Level	
	FPIXC	52	SPI 20	IRQ 20	IRQ 20	Level	
	FPIDC	53	SPI 21	IRQ 21	IRQ 21	Level	
	FPOFC	54	SPI 22	IRQ 22	IRQ 22	Level	
	FPUFC	55	SPI 23	IRQ 23	IRQ 23	Level	
	FPDZC	56	SPI 24	IRQ 24	IRQ 24	Level	
	FPIOC	57	SPI 25	IRQ 25	IRQ 25	Level	
GIC	fault_int	58	SPI 26	IRQ 26	IRQ 26	Level	
	err_int	59	SPI 27	IRQ 27	IRQ 27	Level	
	pmu_int	60	SPI 28	IRQ 28	IRQ 28	Level	
System Bus	BUS_ERR_INT	61	SPI 29	IRQ 29	IRQ 29	Level	*7
Reserved	—	62	SPI 30	IRQ 30	IRQ 30	—	
ECCRAM	EC7TIE1	63	SPI 31	IRQ 31	IRQ 31	Level	*8
	EC7TIE2	64	SPI 32	IRQ 32	IRQ 32	Level	*8
	EC7TIOVF	65	SPI 33	IRQ 33	IRQ 33	Level	*8
DDR/LPDDR4	controller_int	66	SPI 34	IRQ 34	IRQ 34	Level	
	dwc_ddrphy_int_n	67	SPI 35	IRQ 35	IRQ 35	Level	
	sref_error	68	SPI 36	IRQ 36	IRQ 36	Level	
xSPI	int_spi_level	69	SPI 37	IRQ 37	IRQ 37	Level	
	int_spi_err_level	70	SPI 38	IRQ 38	IRQ 38	Level	

Table 8.2 Interrupt Mapping (3/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
SYSC	SYS_LPM_INT	71	SPI 39	IRQ 39	IRQ 39	Level	
	SYS_CA55STBYDONE_INT	72	SPI 40	IRQ 40	IRQ 40	Level	
	SYS_CM33STBYR_INT	73	SPI 41	IRQ 41	IRQ 41	Level	
	SYS_CA55_DENY	74	SPI 42	IRQ 42	IRQ 42	Level	
VBATTB	TAMPDI	75	SPI 43	IRQ 43	IRQ 43	Level	
GTM (ch0)	OSTM0TINT	76	SPI 44	IRQ 44	IRQ 44	Edge	
GTM (ch1)	OSTM1TINT	77	SPI 45	IRQ 45	IRQ 45	Edge	
GTM (ch2)	OSTM2TINT	78	SPI 46	IRQ 46	IRQ 46	Edge	
GTM (ch3)	OSTM3TINT	79	SPI 47	IRQ 47	IRQ 47	Edge	
GTM (ch4)	OSTM4TINT	80	SPI 48	IRQ 48	IRQ 48	Edge	
GTM (ch5)	OSTM5TINT	81	SPI 49	IRQ 49	IRQ 49	Edge	
GTM (ch6)	OSTM6TINT	82	SPI 50	IRQ 50	IRQ 50	Edge	
GTM (ch7)	OSTM7TINT	83	SPI 51	IRQ 51	IRQ 51	Edge	
WDT for CA55 (core0)	PERROUT_A0	84	SPI 52	IRQ 52	IRQ 52	Level	
WDT for CA55 (core0)	WDTINT _A0	85	SPI 53	IRQ 53	IRQ 53	Level	
WDT for CM33	PERROUT_M0	86	SPI 54	IRQ 54	IRQ 54	Level	
WDT for CM33	WDTINT_M0	87	SPI 55	IRQ 55	IRQ 55	Level	
WDT for CM33_FPU	PERROUT_M1	88	SPI 56	IRQ 56	IRQ 56	Level	
WDT for CM33_FPU	WDTINT _M1	89	SPI 57	IRQ 57	IRQ 57	Level	
MHU for CA55	rsp_ch0_ns	90	SPI 58	—	—	Level	
	rsp_ch1_ns	91	SPI 59	—	—	Level	
	msg_ch2_ns	92	SPI 60	—	—	Level	
	msg_ch4_ns	93	SPI 61	—	—	Level	
	rsp_ch0_s	94	SPI 62	—	—	Level	
	rsp_ch1_s	95	SPI 63	—	—	Level	
	msg_ch2_s	96	SPI 64	—	—	Level	
	msg_ch4_s	97	SPI 65	—	—	Level	
	swint_ch2_ns	98	SPI 66	—	—	Level	
	swint_ch4_ns	99	SPI 67	—	—	Level	
MHU for CM33	msg_ch1_ns	—	—	IRQ 58	—	Level	
	msg_ch3_ns	—	—	IRQ 59	—	Level	
	rsp_ch4_ns	—	—	IRQ 60	—	Level	
	rsp_ch5_ns	—	—	IRQ 61	—	Level	
	msg_ch1_s	—	—	IRQ 62	—	Level	
	msg_ch3_s	—	—	IRQ 63	—	Level	
	rsp_ch4_s	—	—	IRQ 64	—	Level	
	rsp_ch5_s	—	—	IRQ 65	—	Level	
	swint_ch0_ns	—	—	IRQ 66	—	Level	
	swint_ch1_ns	—	—	IRQ 67	—	Level	

Table 8.2 Interrupt Mapping (4/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
MHU for CM33_FPU	msg_ch0_ns	—	—	—	IRQ 58	Level	
	rsp_ch2_ns	—	—	—	IRQ 59	Level	
	rsp_ch3_ns	—	—	—	IRQ 60	Level	
	msg_ch5_ns	—	—	—	IRQ 61	Level	
	msg_ch0_s	—	—	—	IRQ 62	Level	
	rsp_ch2_s	—	—	—	IRQ 63	Level	
	rsp_ch3_s	—	—	—	IRQ 64	Level	
	msg_ch5_s	—	—	—	IRQ 65	Level	
	swint_ch3_ns	—	—	—	IRQ 66	Level	
	swint_ch5_ns	—	—	—	IRQ 67	Level	
GbE (ch0)	pif_int_n_ch0	100	SPI 68	IRQ 68	IRQ 68	Level	
	int_fil_n_ch0	101	SPI 69	IRQ 69	IRQ 69	Level	
	int_arp_ns_n_ch0	102	SPI 70	IRQ 70	IRQ 70	Level	
GbE (ch1)	pif_int_n_ch1	103	SPI 71	IRQ 71	IRQ 71	Level	
	int_fil_n_ch1	104	SPI 72	IRQ 72	IRQ 72	Level	
	int_arp_ns_n_ch1	105	SPI 73	IRQ 73	IRQ 73	Level	
USB2.0 (Host ch0)	U2H0_INT	106	SPI 74	IRQ 74	IRQ 74	Level	
	U2H0_OHCI_INT	107	SPI 75	IRQ 75	IRQ 75	Level	
	U2H0_EHCI_INT	108	SPI 76	IRQ 76	IRQ 76	Level	
	U2H0_WAKEON_INT	109	SPI 77	IRQ 77	IRQ 77	Level	
	U2H0_OBINT	110	SPI 78	IRQ 78	IRQ 78	Level	
USB2.0 (Host ch1)	U2H1_INT	111	SPI 79	IRQ 79	IRQ 79	Level	
	U2H1_OHCI_INT	112	SPI 80	IRQ 80	IRQ 80	Level	
	U2H1_EHCI_INT	113	SPI 81	IRQ 81	IRQ 81	Level	
	U2H1_WAKEON_INT	114	SPI 82	IRQ 82	IRQ 82	Level	
	U2H1_OBINT	115	SPI 83	IRQ 83	IRQ 83	Level	
USB2.0 (Function)	U2P_INT_DMAERR	116	SPI 84	IRQ 84	IRQ 84	Level	
	U2P_IXL_INT	117	SPI 85	IRQ 85	IRQ 85	Edge	
	U2P_INT_DMA[0]	118	SPI 86	IRQ 86	IRQ 86	Level	
	U2P_INT_DMA[1]	119	SPI 87	IRQ 87	IRQ 87	Level	
SDHI/eMMC (ch0)	OXMNIRQ0	120	SPI 88	IRQ 88	IRQ 88	Level	
	OXASIOIRQ0	121	SPI 89	IRQ 89	IRQ 89	Level	
SDHI/eMMC (ch1)	OXMNIRQ1	122	SPI 90	IRQ 90	IRQ 90	Level	
	OXASIOIRQ1	123	SPI 91	IRQ 91	IRQ 91	Level	
SDHI/eMMC (ch2)	OXMNIRQ2	124	SPI 92	IRQ 92	IRQ 92	Level	
	OXASIOIRQ2	125	SPI 93	IRQ 93	IRQ 93	Level	

Table 8.2 Interrupt Mapping (5/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
DMAC (Secure)	DAMERR_S	126	SPI 94	IRQ 94	IRQ 94	Edge	
	DMAINT0_S	127	SPI 95	IRQ 95	IRQ 95	Edge	
	DMAINT1_S	128	SPI 96	IRQ 96	IRQ 96	Edge	
	DMAINT2_S	129	SPI 97	IRQ 97	IRQ 97	Edge	
	DMAINT3_S	130	SPI 98	IRQ 98	IRQ 98	Edge	
	DMAINT4_S	131	SPI 99	IRQ 99	IRQ 99	Edge	
	DMAINT5_S	132	SPI 100	IRQ 100	IRQ 100	Edge	
	DMAINT6_S	133	SPI 101	IRQ 101	IRQ 101	Edge	
	DMAINT7_S	134	SPI 102	IRQ 102	IRQ 102	Edge	
	DMAINT8_S	135	SPI 103	IRQ 103	IRQ 103	Edge	
	DMAINT9_S	136	SPI 104	IRQ 104	IRQ 104	Edge	
	DMAINT10_S	137	SPI 105	IRQ 105	IRQ 105	Edge	
	DMAINT11_S	138	SPI 106	IRQ 106	IRQ 106	Edge	
	DMAINT12_S	139	SPI 107	IRQ 107	IRQ 107	Edge	
	DMAINT13_S	140	SPI 108	IRQ 108	IRQ 108	Edge	
	DMAINT14_S	141	SPI 109	IRQ 109	IRQ 109	Edge	
	DMAINT15_S	142	SPI 110	IRQ 110	IRQ 110	Edge	
DMAC (Non-Secure)	DMAERR_NS	143	SPI 111	IRQ 111	IRQ 111	Edge	
	DMAINT0_NS	144	SPI 112	IRQ 112	IRQ 112	Edge	
	DMAINT1_NS	145	SPI 113	IRQ 113	IRQ 113	Edge	
	DMAINT2_NS	146	SPI 114	IRQ 114	IRQ 114	Edge	
	DMAINT3_NS	147	SPI 115	IRQ 115	IRQ 115	Edge	
	DMAINT4_NS	148	SPI 116	IRQ 116	IRQ 116	Edge	
	DMAINT5_NS	149	SPI 117	IRQ 117	IRQ 117	Edge	
	DMAINT6_NS	150	SPI 118	IRQ 118	IRQ 118	Edge	
	DMAINT7_NS	151	SPI 119	IRQ 119	IRQ 119	Edge	
	DMAINT8_NS	152	SPI 120	IRQ 120	IRQ 120	Edge	
	DMAINT9_NS	153	SPI 121	IRQ 121	IRQ 121	Edge	
	DMAINT10_NS	154	SPI 122	IRQ 122	IRQ 122	Edge	
	DMAINT11_NS	155	SPI 123	IRQ 123	IRQ 123	Edge	
	DMAINT12_NS	156	SPI 124	IRQ 124	IRQ 124	Edge	
	DMAINT13_NS	157	SPI 125	IRQ 125	IRQ 125	Edge	
	DMAINT14_NS	158	SPI 126	IRQ 126	IRQ 126	Edge	
	DMAINT15_NS	159	SPI 127	IRQ 127	IRQ 127	Edge	

Table 8.2 Interrupt Mapping (6/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
GPT (Ch0)	CCMPA0	160	SPI 128	IRQ 128	IRQ 128	Edge	
	CCMPB0	161	SPI 129	IRQ 129	IRQ 129	Edge	
	CMPC0	162	SPI 130	IRQ 130	IRQ 130	Edge	
	CMPD0	163	SPI 131	IRQ 131	IRQ 131	Edge	
	CMPE0	164	SPI 132	IRQ 132	IRQ 132	Edge	*9
	CMPF0	165	SPI 133	IRQ 133	IRQ 133	Edge	*9
	ADTRGA0	166	SPI 134	IRQ 134	IRQ 134	Edge	*9
	ADTRGB0	167	SPI 135	IRQ 135	IRQ 135	Edge	*9
	OVF0	168	SPI 136	IRQ 136	IRQ 136	Edge	*9
	UNF0	169	SPI 137	IRQ 137	IRQ 137	Edge	*9
Reserved	—	170	SPI 138	IRQ 138	IRQ 138	—	
Reserved	—	171	SPI 139	IRQ 139	IRQ 139	—	
Reserved	—	172	SPI 140	IRQ 140	IRQ 140	—	
GPT (Ch1)	CCMPA1	173	SPI 141	IRQ 141	IRQ 141	Edge	
	CCMPB1	174	SPI 142	IRQ 142	IRQ 142	Edge	
	CMPC1	175	SPI 143	IRQ 143	IRQ 143	Edge	
	CMPD1	176	SPI 144	IRQ 144	IRQ 144	Edge	
	CMPE1	177	SPI 145	IRQ 145	IRQ 145	Edge	*9
	CMPF1	178	SPI 146	IRQ 146	IRQ 146	Edge	*9
	ADTRGA1	179	SPI 147	IRQ 147	IRQ 147	Edge	*9
	ADTRGB1	180	SPI 148	IRQ 148	IRQ 148	Edge	*9
	OVF1	181	SPI 149	IRQ 149	IRQ 149	Edge	*9
	UNF1	182	SPI 150	IRQ 150	IRQ 150	Edge	*9
Reserved	—	183	SPI 151	IRQ 151	IRQ 151	—	
Reserved	—	184	SPI 152	IRQ 152	IRQ 152	—	
Reserved	—	185	SPI 153	IRQ 153	IRQ 153	—	
GPT (ch2)	CCMPA2	186	SPI 154	IRQ 154	IRQ 154	Edge	
	CCMPB2	187	SPI 155	IRQ 155	IRQ 155	Edge	
	CMPC2	188	SPI 156	IRQ 156	IRQ 156	Edge	
	CMPD2	189	SPI 157	IRQ 157	IRQ 157	Edge	
	CMPE2	190	SPI 158	IRQ 158	IRQ 158	Edge	*9
	CMPF2	191	SPI 159	IRQ 159	IRQ 159	Edge	*9
	ADTRGA2	192	SPI 160	IRQ 160	IRQ 160	Edge	*9
	ADTRGB2	193	SPI 161	IRQ 161	IRQ 161	Edge	*9
	OVF2	194	SPI 162	IRQ 162	IRQ 162	Edge	*9
	UNF2	195	SPI 163	IRQ 163	IRQ 163	Edge	*9
Reserved	—	196	SPI 164	IRQ 164	IRQ 164	—	
Reserved	—	197	SPI 165	IRQ 165	IRQ 165	—	
Reserved	—	198	SPI 166	IRQ 166	IRQ 166	—	

Table 8.2 Interrupt Mapping (7/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
GTP (ch3)	CCMPA3	199	SPI 167	IRQ 167	IRQ 167	Edge	
	CCMPB3	200	SPI 168	IRQ 168	IRQ 168	Edge	
	CMPC3	201	SPI 169	IRQ 169	IRQ 169	Edge	
	CMPD3	202	SPI 170	IRQ 170	IRQ 170	Edge	
	CMPE3	203	SPI 171	IRQ 171	IRQ 171	Edge	*9
	CMPF3	204	SPI 172	IRQ 172	IRQ 172	Edge	*9
	ADTRGA3	205	SPI 173	IRQ 173	IRQ 173	Edge	*9
	ADTRGB3	206	SPI 174	IRQ 174	IRQ 174	Edge	*9
	OVF3	207	SPI 175	IRQ 175	IRQ 175	Edge	*9
	UNF3	208	SPI 176	IRQ 176	IRQ 176	Edge	*9
Reserved	—	209	SPI 177	IRQ 177	IRQ 177	—	
Reserved	—	210	SPI 178	IRQ 178	IRQ 178	—	
Reserved	—	211	SPI 179	IRQ 179	IRQ 179	—	
GPT (ch4)	CCMPA4	212	SPI 180	IRQ 180	IRQ 180	Edge	
	CCMPB4	213	SPI 181	IRQ 181	IRQ 181	Edge	
	CMPC4	214	SPI 182	IRQ 182	IRQ 182	Edge	
	CMPD4	215	SPI 183	IRQ 183	IRQ 183	Edge	
	CMPE4	216	SPI 184	IRQ 184	IRQ 184	Edge	*9
	CMPF4	217	SPI 185	IRQ 185	IRQ 185	Edge	*9
	ADTRGA4	218	SPI 186	IRQ 186	IRQ 186	Edge	*9
	ADTRGB4	219	SPI 187	IRQ 187	IRQ 187	Edge	*9
	OVF4	220	SPI 188	IRQ 188	IRQ 188	Edge	*9
	UNF4	221	SPI 189	IRQ 189	IRQ 189	Edge	*9
Reserved	—	222	SPI 190	IRQ 190	IRQ 190	—	
Reserved	—	223	SPI 191	IRQ 191	IRQ 191	—	
Reserved	—	224	SPI 192	IRQ 192	IRQ 192	—	
GPT (ch5)	CCMPA5	225	SPI 193	IRQ 193	IRQ 193	Edge	
	CCMPB5	226	SPI 194	IRQ 194	IRQ 194	Edge	
	CMPC5	227	SPI 195	IRQ 195	IRQ 195	Edge	
	CMPD5	228	SPI 196	IRQ 196	IRQ 196	Edge	
	CMPE5	229	SPI 197	IRQ 197	IRQ 197	Edge	*9
	CMPF5	230	SPI 198	IRQ 198	IRQ 198	Edge	*9
	ADTRGA5	231	SPI 199	IRQ 199	IRQ 199	Edge	*9
	ADTRGB5	232	SPI 200	IRQ 200	IRQ 200	Edge	*9
	OVF5	233	SPI 201	IRQ 201	IRQ 201	Edge	*9
	UNF5	234	SPI 202	IRQ 202	IRQ 202	Edge	*9
Reserved	—	235	SPI 203	IRQ 203	IRQ 203	—	
Reserved	—	236	SPI 204	IRQ 204	IRQ 204	—	
Reserved	—	237	SPI 205	IRQ 205	IRQ 205	—	

Table 8.2 Interrupt Mapping (8/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
GPT (ch6)	CCMPA6	238	SPI 206	IRQ 206	IRQ 206	Edge	
	CCMPB6	239	SPI 207	IRQ 207	IRQ 207	Edge	
	CMPC6	240	SPI 208	IRQ 208	IRQ 208	Edge	
	CMPD6	241	SPI 209	IRQ 209	IRQ 209	Edge	
	CMPE6	242	SPI 210	IRQ 210	IRQ 210	Edge	*9
	CMPF6	243	SPI 211	IRQ 211	IRQ 211	Edge	*9
	ADTRGA6	244	SPI 212	IRQ 212	IRQ 212	Edge	*9
	ADTRGB6	245	SPI 213	IRQ 213	IRQ 213	Edge	*9
	OVF6	246	SPI 214	IRQ 214	IRQ 214	Edge	*9
	UNF6	247	SPI 215	IRQ 215	IRQ 215	Edge	*9
Reserved	—	248	SPI 216	IRQ 216	IRQ 216	—	
Reserved	—	249	SPI 217	IRQ 217	IRQ 217	—	
Reserved	—	250	SPI 218	IRQ 218	IRQ 218	—	
GPT (ch7)	CCMPA7	251	SPI 219	IRQ 219	IRQ 219	Edge	
	CCMPB7	252	SPI 220	IRQ 220	IRQ 220	Edge	
	CMPC7	253	SPI 221	IRQ 221	IRQ 221	Edge	
	CMPD7	254	SPI 222	IRQ 222	IRQ 222	Edge	
	CMPE7	255	SPI 223	IRQ 223	IRQ 223	Edge	*9
	CMPF7	256	SPI 224	IRQ 224	IRQ 224	Edge	*9
	ADTRGA7	257	SPI 225	IRQ 225	IRQ 225	Edge	
	ADTRGB7	258	SPI 226	IRQ 226	IRQ 226	Edge	
	OVF6	259	SPI 227	IRQ 227	IRQ 227	Edge	
	UNF7	260	SPI 228	IRQ 228	IRQ 228	Edge	
Reserved	—	261	SPI 229	IRQ 229	IRQ 229	—	
Reserved	—	262	SPI 230	IRQ 230	IRQ 230	—	
Reserved	—	263	SPI 231	IRQ 231	IRQ 231	—	
POE3	OEI1	264	SPI 232	IRQ 232	IRQ 232	Level	
	OEI2	265	SPI 233	IRQ 233	IRQ 233	Level	
	OEI3	266	SPI 234	IRQ 234	IRQ 234	Level	
	OEI4	267	SPI 235	IRQ 235	IRQ 235	Level	
POEGA	GROUP0	268	SPI 236	IRQ 236	IRQ 236	Level	
POEGB	GROUP1	269	SPI 237	IRQ 237	IRQ 237	Level	
POEGC	GROUP2	270	SPI 238	IRQ 238	IRQ 238	Level	
POEGD	GROUP3	271	SPI 239	IRQ 239	IRQ 239	Level	
SSIF (ch0)	INT_ssif_int_req_0	272	SPI 240	IRQ 240	IRQ 240	Level	
	INT_ssif_dma_rx_0	273	SPI 241	IRQ 241	IRQ 241	Edge	
	INT_ssif_dma_tx_0	274	SPI 242	IRQ 242	IRQ 242	Edge	
SSIF (ch1)	INT_ssif_int_req_1	275	SPI 243	IRQ 243	IRQ 243	Level	
	INT_ssif_dma_rx_1	276	SPI 244	IRQ 244	IRQ 244	Edge	
	INT_ssif_dma_tx_1	277	SPI 245	IRQ 245	IRQ 245	Edge	

Table 8.2 Interrupt Mapping (9/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
SSIF (ch2)	INT_ssif_int_req_2	278	SPI 246	IRQ 246	IRQ 246	Level	
	INT_ssif_dma_rx_2	279	SPI 247	IRQ 247	IRQ 247	Edge	
	INT_ssif_dma_tx_2	280	SPI 248	IRQ 248	IRQ 248	Edge	
SSIF (ch3)	INT_ssif_int_req_3	281	SPI 249	IRQ 249	IRQ 249	Level	
	INT_ssif_dma_rx_3	282	SPI 250	IRQ 250	IRQ 250	Edge	
	INT_ssif_dma_tx_3	283	SPI 251	IRQ 251	IRQ 251	Edge	
SRC	SRC_IDEI	284	SPI 252	IRQ 252	IRQ 252	Edge	
	SRC_ODFI	285	SPI 253	IRQ 253	IRQ 253	Edge	
	SRC_CEF	286	SPI 254	IRQ 254	IRQ 254	Level	
	SRC_UDF	287	SPI 255	IRQ 255	IRQ 255	Level	
	SRC_OVF	288	SPI 256	IRQ 256	IRQ 256	Level	
I2C (ch0)	INTRIICTEI0	289	SPI 257	IRQ 257	IRQ 257	Level	
	INTRIICNAKI0	290	SPI 258	IRQ 258	IRQ 258	Level	
	INTRIICSPI0	291	SPI 259	IRQ 259	IRQ 259	Level	
	INTRIICSTI0	292	SPI 260	IRQ 260	IRQ 260	Level	
	INTRIICALI0	293	SPI 261	IRQ 261	IRQ 261	Level	
	INTRIICTMOI0	294	SPI 262	IRQ 262	IRQ 262	Level	
	INTRIICRI0	295	SPI 263	IRQ 263	IRQ 263	Edge	
	INTRIICTI0	296	SPI 264	IRQ 264	IRQ 264	Edge	
I2C (ch1)	INTRIICTEI0	297	SPI 265	IRQ 265	IRQ 265	Level	
	INTRIICNAKI1	298	SPI 266	IRQ 266	IRQ 266	Level	
	INTRIICSPI1	299	SPI 267	IRQ 267	IRQ 267	Level	
	INTRIICSTI1	300	SPI 268	IRQ 268	IRQ 268	Level	
	INTRIICALI0	301	SPI 269	IRQ 269	IRQ 269	Level	
	INTRIICTMOI1	302	SPI 270	IRQ 270	IRQ 270	Level	
	INTRIICRI1	303	SPI 271	IRQ 271	IRQ 271	Edge	
	INTRIICTI1	304	SPI 272	IRQ 272	IRQ 272	Edge	
I2C (ch2)	INTRIICTEI2	305	SPI 273	IRQ 273	IRQ 273	Level	
	INTRIICNAKI2	306	SPI 274	IRQ 274	IRQ 274	Level	
	INTRIICSPI12	307	SPI 275	IRQ 275	IRQ 275	Level	
	INTRIICSTI2	308	SPI 276	IRQ 276	IRQ 276	Level	
	INTRIICALI2	309	SPI 277	IRQ 277	IRQ 277	Level	
	INTRIICTMOI2	310	SPI 278	IRQ 278	IRQ 278	Level	
	INTRIICRI2	311	SPI 279	IRQ 279	IRQ 279	Edge	
	INTRIICTI2	312	SPI 280	IRQ 280	IRQ 280	Edge	

Table 8.2 Interrupt Mapping (10/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
I2C (ch3)	INTRIICTEI3	313	SPI 281	IRQ 281	IRQ 281	Level	
	INTRIICNAKI3	314	SPI 282	IRQ 282	IRQ 282	Level	
	INTRIICSPI13	315	SPI 283	IRQ 283	IRQ 283	Level	
	INTRIICSTI3	316	SPI 284	IRQ 284	IRQ 284	Level	
	INTRIICALI2	317	SPI 285	IRQ 285	IRQ 285	Level	
	INTRIICTMOI3	318	SPI 286	IRQ 286	IRQ 286	Level	
	INTRIICRI3	319	SPI 287	IRQ 287	IRQ 287	Edge	
	INTRIICTI3	320	SPI 288	IRQ 288	IRQ 288	Edge	
I3C	INT_ri3c_ierr_n	321	SPI 289	IRQ 289	IRQ 289	Level	
	INT_ri3c_terr_n	322	SPI 290	IRQ 290	IRQ 290	Level	
Reserved	—	323	SPI 291	IRQ 291	IRQ 291	—	
Reserved	—	324	SPI 292	IRQ 292	IRQ 292	—	
I3C	INT_ri3c_abort_n	325	SPI 293	IRQ 293	IRQ 293	Level	
	INT_ri3c_resp_n	326	SPI 294	IRQ 294	IRQ 294	Edge	
	INT_ri3c_cmd_n	327	SPI 295	IRQ 295	IRQ 295	Edge	
	INT_ri3c_ibi_n	328	SPI 296	IRQ 296	IRQ 296	Edge	
	INT_ri3c_rx_n	329	SPI 297	IRQ 297	IRQ 297	Edge	
	INT_ri3c_tx_n	330	SPI 298	IRQ 298	IRQ 298	Edge	
	INT_ri3c_rcv_n	331	SPI 299	IRQ 299	IRQ 299	Edge	
Reserved	—	332	SPI 300	IRQ 300	IRQ 300	—	
Reserved	—	333	SPI 301	IRQ 301	IRQ 301	—	
Reserved	—	334	SPI 302	IRQ 302	IRQ 302	—	
Reserved	—	335	SPI 303	IRQ 303	IRQ 303	—	
I3C	INT_ri3c_st_n	336	SPI 304	IRQ 304	IRQ 304	Level	
	INT_ri3c_sp_n	337	SPI 305	IRQ 305	IRQ 305	Level	
	INT_ri3c_exit_n	338	SPI 306	IRQ 306	IRQ 306	Level	
	INT_ri3c_tend_n	339	SPI 307	IRQ 307	IRQ 307	Level	
	INT_ri3c_nack_n	340	SPI 308	IRQ 308	IRQ 308	Level	
	INT_ri3c_al_n	341	SPI 309	IRQ 309	IRQ 309	Level	
	INT_ri3c_tmo_n	342	SPI 310	IRQ 310	IRQ 310	Level	
	INT_ri3c_wu_n	343	SPI 311	IRQ 311	IRQ 311	Level	
ADC	INTAD	344	SPI 312	IRQ 312	IRQ 312	Edge	
Reserved	—	345	SPI 313	IRQ 313	IRQ 313	—	
Reserved	—	346	SPI 314	IRQ 314	IRQ 314	—	
RTC	intreq_rtc_alm_n	347	SPI 315	IRQ 315	IRQ 315	Level	
	intreq_rtc_prd_n	348	SPI 316	IRQ 316	IRQ 316	Level	
	intreq_rtc_cup_n	349	SPI 317	IRQ 317	IRQ 317	Level	
Reserved	—	350	SPI 318	IRQ 318	IRQ 318	—	
Reserved	—	351	SPI 319	IRQ 319	IRQ 319	—	

Table 8.2 Interrupt Mapping (11/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
SCIFA (ch0)	ERI0	352	SPI 320	IRQ 320	IRQ 320	Level	
	BRI0	353	SPI 321	IRQ 321	IRQ 321	Level	
	RXI0	354	SPI 322	IRQ 322	IRQ 322	Level	
	TXI0	355	SPI 323	IRQ 323	IRQ 323	Level	
	TEI0_DRI0	356	SPI 324	IRQ 324	IRQ 324	Level	
SCIFA (ch1)	ERI1	357	SPI 325	IRQ 325	IRQ 325	Level	
	BRI1	358	SPI 326	IRQ 326	IRQ 326	Level	
	RXI1	359	SPI 327	IRQ 327	IRQ 327	Level	
	TXI1	360	SPI 328	IRQ 328	IRQ 328	Level	
	TEI1_DRI1	361	SPI 329	IRQ 329	IRQ 329	Level	
SCIFA (ch2)	ERI2	362	SPI 330	IRQ 330	IRQ 330	Level	
	BRI2	363	SPI 331	IRQ 331	IRQ 331	Level	
	RXI2	364	SPI 332	IRQ 332	IRQ 332	Level	
	TXI2	365	SPI 333	IRQ 333	IRQ 333	Level	
	TEI2_DRI2	366	SPI 334	IRQ 334	IRQ 334	Level	
SCIFA (ch3)	ERI3	367	SPI 335	IRQ 335	IRQ 335	Level	
	BRI3	368	SPI 336	IRQ 336	IRQ 336	Level	
	RXI3	369	SPI 337	IRQ 337	IRQ 337	Level	
	TXI3	370	SPI 338	IRQ 338	IRQ 338	Level	
	TEI3_DRI3	371	SPI 339	IRQ 339	IRQ 339	Level	
SCIFA (ch4)	ERI4	372	SPI 340	IRQ 340	IRQ 340	Level	
	BRI4	373	SPI 341	IRQ 341	IRQ 341	Level	
	RXI4	374	SPI 342	IRQ 342	IRQ 342	Level	
	TXI4	375	SPI 343	IRQ 343	IRQ 343	Level	
	TEI4_DRI4	376	SPI 344	IRQ 344	IRQ 344	Level	
SCIFA (ch5)	ERI5	377	SPI 345	IRQ 345	IRQ 345	Level	
	BRI5	378	SPI 346	IRQ 346	IRQ 346	Level	
	RXI5	379	SPI 347	IRQ 347	IRQ 347	Level	
	TXI5	380	SPI 348	IRQ 348	IRQ 348	Level	
	TEI5_DRI5	381	SPI 349	IRQ 349	IRQ 349	Level	
SCIg (ch0)	ERI0	382	SPI 350	IRQ 350	IRQ 350	Level	
	RXI0	383	SPI 351	IRQ 351	IRQ 351	Edge	
	TXI0	384	SPI 352	IRQ 352	IRQ 352	Edge	
	TEI0	385	SPI 353	IRQ 353	IRQ 353	Level	
SCIg (ch1)	ERI1	386	SPI 354	IRQ 354	IRQ 354	Level	
	RXI1	387	SPI 355	IRQ 355	IRQ 355	Edge	
	TXI1	388	SPI 356	IRQ 356	IRQ 356	Edge	
	TEI1	389	SPI 357	IRQ 357	IRQ 357	Level	
RSPI (ch0)	SPEI0	390	SPI 358	IRQ 358	IRQ 358	Level	
	SPRI0	391	SPI 359	IRQ 359	IRQ 359	Level	
	SPTI0	392	SPI 360	IRQ 360	IRQ 360	Level	

Table 8.2 Interrupt Mapping (12/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
RSPI (ch1)	SPEI1	393	SPI 361	IRQ 361	IRQ 361	Level	
	SPRI1	394	SPI 362	IRQ 362	IRQ 362	Level	
	SPTI1	395	SPI 363	IRQ 363	IRQ 363	Level	
RSPI (ch2)	SPEI2	396	SPI 364	IRQ 364	IRQ 364	Level	
	SPRI2	397	SPI 365	IRQ 365	IRQ 365	Level	
	SPTI2	398	SPI 366	IRQ 366	IRQ 366	Level	
RSPI (ch3)	SPEI3	399	SPI 367	IRQ 367	IRQ 367	Level	
	SPRI3	400	SPI 368	IRQ 368	IRQ 368	Level	
	SPTI3	401	SPI 369	IRQ 369	IRQ 369	Level	
RSPI (ch4)	SPEI4	402	SPI 370	IRQ 370	IRQ 370	Level	
	SPRI4	403	SPI 371	IRQ 371	IRQ 371	Level	
	SPTI4	404	SPI 372	IRQ 372	IRQ 372	Level	
CANFD	INTRCANGERR	405	SPI 373	IRQ 373	IRQ 373	Level	
	INTRCANGRECC	406	SPI 374	IRQ 374	IRQ 374	Level	
	INTRCAN0REC	407	SPI 375	IRQ 375	IRQ 375	Level	
	INTRCAN1REC	408	SPI 376	IRQ 376	IRQ 376	Level	
	INTRCAN0ERR	409	SPI 377	IRQ 377	IRQ 377	Level	
	INTRCAN1ERR	410	SPI 378	IRQ 378	IRQ 378	Level	
	INTRCAN0TRX	411	SPI 379	IRQ 379	IRQ 379	Level	
	INTRCAN1TRX	412	SPI 380	IRQ 380	IRQ 380	Level	
Reserved	—	413	SPI 381	IRQ 381	IRQ 381	—	
Reserved	—	414	SPI 382	IRQ 382	IRQ 382	—	
Reserved	—	415	SPI 383	IRQ 383	IRQ 383	—	
Reserved	—	416	SPI 384	IRQ 384	IRQ 384	—	
Reserved	—	417	SPI 385	IRQ 385	IRQ 385	—	
Reserved	—	418	SPI 386	IRQ 386	IRQ 386	—	
SPDIF	intreq_spdif_n	419	SPI 387	IRQ 387	IRQ 387	Level	
PDM	INT_PDM_ERR0	420	SPI 388	IRQ 388	IRQ 388	Level	
	INT_PDM_ERR1	421	SPI 389	IRQ 389	IRQ 389	Level	
	INT_PDM_ERR2	422	SPI 390	IRQ 390	IRQ 390	Level	
	INT_PDM_DAT0	423	SPI 391	IRQ 391	IRQ 391	Level	
	INT_PDM_DAT1	424	SPI 392	IRQ 392	IRQ 392	Level	
	INT_PDM_DAT2	425	SPI 393	IRQ 393	IRQ 393	Level	
	INT_PDM_SDET	426	SPI 394	IRQ 394	IRQ 394	Level	

Table 8.2 Interrupt Mapping (13/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
PCIe	INT_SERR	427	SPI 395	IRQ 395	IRQ 395	Level	
	INT_SERR_COR	428	SPI 396	IRQ 396	IRQ 396	Level	
	INT_SERR_NONFATAL	429	SPI 397	IRQ 397	IRQ 397	Level	
	INT_SERR_FATAL	430	SPI 398	IRQ 398	IRQ 398	Level	
	AXI_ERR_INT	431	SPI 399	IRQ 399	IRQ 399	Level	
	INTA_RC	432	SPI 400	IRQ 400	IRQ 400	Level	
	INTB_RC	433	SPI 401	IRQ 401	IRQ 401	Level	
	INTC_RC	434	SPI 402	IRQ 402	IRQ 402	Level	
	INTD_RC	435	SPI 403	IRQ 403	IRQ 403	Level	
	INTMSI_RC	436	SPI 404	IRQ 404	IRQ 404	Level	
	INT_LINK_BANDWIDTH	437	SPI 405	IRQ 405	IRQ 405	Level	
	INT_PM_PME	438	SPI 406	IRQ 406	IRQ 406	Level	
	DMA_INT	439	SPI 407	IRQ 407	IRQ 407	Level	
	PCIE_EVT_INT	440	SPI 408	IRQ 408	IRQ 408	Level	
	MSG_INT	441	SPI 409	IRQ 409	IRQ 409	Level	
	INT_ALL	442	SPI 410	IRQ 410	IRQ 410	Level	
Reserved	—	443	SPI 411	IRQ 411	IRQ 411	—	
Reserved	—	444	SPI 412	IRQ 412	IRQ 412	—	
Reserved	—	445	SPI 413	IRQ 413	IRQ 413	—	
Reserved	—	446	SPI 414	IRQ 414	IRQ 414	—	
Reserved	—	447	SPI 415	IRQ 415	IRQ 415	—	
Reserved	—	448	SPI 416	IRQ 416	IRQ 416	—	
Reserved	—	449	SPI 417	IRQ 417	IRQ 417	—	
Reserved	—	450	SPI 418	IRQ 418	IRQ 418	—	
Reserved	—	451	SPI 419	IRQ 419	IRQ 419	—	
Reserved	—	452	SPI 420	IRQ 420	IRQ 420	—	
Reserved	—	453	SPI 421	IRQ 421	IRQ 421	—	
Reserved	—	454	SPI 422	IRQ 422	IRQ 422	—	
Reserved	—	455	SPI 423	IRQ 423	IRQ 423	—	
Reserved	—	456	SPI 424	IRQ 424	IRQ 424	—	
Reserved	—	457	SPI 425	IRQ 425	IRQ 425	—	
Reserved	—	458	SPI 426	IRQ 426	IRQ 426	—	
Reserved	—	459	SPI 427	IRQ 427	IRQ 427	—	
Reserved	—	460	SPI 428	IRQ 428	IRQ 428	—	

Table 8.2 Interrupt Mapping (14/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
GPIO Interrupt	TINT0	461	SPI 429	IRQ 429	IRQ 429	Level	*6
	TINT1	462	SPI 430	IRQ 430	IRQ 430	Level	*6
	TINT2	463	SPI 431	IRQ 431	IRQ 431	Level	*6
	TINT3	464	SPI 432	IRQ 432	IRQ 432	Level	*6
	TINT4	465	SPI 433	IRQ 433	IRQ 433	Level	*6
	TINT5	466	SPI 434	IRQ 434	IRQ 434	Level	*6
	TINT6	467	SPI 435	IRQ 435	IRQ 435	Level	*6
	TINT7	468	SPI 436	IRQ 436	IRQ 436	Level	*6
	TINT8	469	SPI 437	IRQ 437	IRQ 437	Level	*6
	TINT10	471	SPI 439	IRQ 439	IRQ 439	Level	*6
	TINT11	472	SPI 440	IRQ 440	IRQ 440	Level	*6
	TINT12	473	SPI 441	IRQ 441	IRQ 441	Level	*6
	TINT13	474	SPI 442	IRQ 442	IRQ 442	Level	*6
	TINT14	475	SPI 443	IRQ 443	IRQ 443	Level	*6
	TINT15	476	SPI 444	IRQ 444	IRQ 444	Level	*6
	TINT16	477	SPI 445	IRQ 445	IRQ 445	Level	*6
	TINT17	478	SPI 446	IRQ 446	IRQ 446	Level	*6
	TINT18	479	SPI 447	IRQ 447	IRQ 447	Level	*6
	TINT19	480	SPI 448	IRQ 448	IRQ 448	Level	*6
	TINT20	481	SPI 449	IRQ 449	IRQ 449	Level	*6
	TINT21	482	SPI 450	IRQ 450	IRQ 450	Level	*6
	TINT22	483	SPI 451	IRQ 451	IRQ 451	Level	*6
	TINT23	484	SPI 452	IRQ 452	IRQ 452	Level	*6
	TINT24	485	SPI 453	IRQ 453	IRQ 453	Level	*6
	TINT25	486	SPI 454	IRQ 454	IRQ 454	Level	*6
	TINT26	487	SPI 455	IRQ 455	IRQ 455	Level	*6
	TINT27	488	SPI 456	IRQ 456	IRQ 456	Level	*6
	TINT28	489	SPI 457	IRQ 457	IRQ 457	Level	*6
	TINT29	490	SPI 458	IRQ 458	IRQ 458	Level	*6
	TINT30	491	SPI 459	IRQ 459	IRQ 459	Level	*6
	TINT31	492	SPI 460	IRQ 460	IRQ 460	Level	*6
Reserved	—	493	SPI 461	IRQ 461	IRQ 461	—	
Reserved	—	494	SPI 462	IRQ 462	IRQ 462	—	
Reserved	—	495	SPI 463	IRQ 463	IRQ 463	—	
Reserved	—	496	SPI 464	IRQ 464	IRQ 464	—	
Reserved	—	497	SPI 465	IRQ 465	IRQ 465	—	
Reserved	—	498	SPI 466	IRQ 466	IRQ 466	—	
Reserved	—	499	SPI 467	IRQ 467	IRQ 467	—	
Reserved	—	500	SPI 468	IRQ 468	IRQ 468	—	
Reserved	—	501	SPI 469	IRQ 469	IRQ 469	—	
Reserved	—	502	SPI 470	IRQ 470	IRQ 470	—	
Reserved	—	503	SPI 471	IRQ 471	IRQ 471	—	
Reserved	—	504	SPI 472	IRQ 472	IRQ 472	—	

Table 8.2 Interrupt Mapping (15/15)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Cortex-M33_FPU	Interrupt Type*2	Note
		Interrupt ID	SGI, PPI, SPI No.	IRQ No.*1	IRQ No.*1		
Reserved	—	505	SPI 473	IRQ 473	IRQ 473	—	
Reserved	—	506	SPI 474	IRQ 474	IRQ 474	—	
Reserved	—	507	SPI 475	IRQ 475	IRQ 475	—	
Reserved	—	508	SPI 476	IRQ 476	IRQ 476	—	
Reserved	—	509	SPI 477	IRQ 477	IRQ 477	—	
Reserved	—	510	SPI 478	IRQ 478	IRQ 478	—	
Reserved	—	511	SPI 479	IRQ 479	IRQ 479	—	

Note 1. Exception handlers of Cortex-M33/Cortex-M33_FPU has Interrupt Service Routines (ISRs), Fault handler and System handler.
IRQ0-479 is treated as Interrupt Service Routines (ISRs).
Refer to “Arm Cortex-M33 Processor Technical Reference Manual” and “Arm Cortex-M33 Processor User Guide” for details and targets of Fault handler and System handler.

Note 2. Set GICD_ICFGR<n> of the interrupt controller for Cortex-A55 (GIC-600) according to the description of Interrupt Type for each Interrupt ID.
Refer to “ARM CoreLink GIC-600 Generic Interrupt Controller Technical Reference Manual” for the specification of GIC-600 registers.
Equivalent settings are not required for the interrupt controller for Cortex-M33/Cortex-M33_FPU (Cortex-M33/Cortex-M33_FPU built-in NVIC).

Note 3. Refer to the source unit section for the specification of each interrupt source.

Note 4. NMI is not treated as NMI exception.
NMI interrupt setting can be controlled by the registers in IA55 and IM33//IM33_FPU. Refer to **Section 8.7.1** for details.

Note 5. IRQ0-7 interrupt settings can be controlled by the registers in IA55 and IM33/IM33_FPU. Refer to **Section 8.7.2** for details.

Note 6. TINT0-31 interrupt settings can be controlled by the registers in IA55 and IM33/IM33_FPU. Refer to **Section 8.7.3** for details.

Note 7. BUS_ERR_INT is the interrupt integrated multiple interrupts into one in IA55 and IM33.
The interrupt settings can be controlled by the registers in IA55 and IM33/IM33_FPU. Refer to **Section 8.7.4.1** for details.

Note 8. Each EC7TIE1, TC7TIE2 and EC7TIOVF is the interrupt integrated multiple interrupts into one in IA55 and IM33/IM33_FPU.
The interrupt settings can be controlled by the registers in IA55 and IM33/IM33_FPU. Refer to **Section 8.7.4.2** for details.

Note 9. GPT and MTU3a interrupts are selected by the registers in IA55 and IM33/IM33_FPU. Refer to **Section 8.7.5** for details.

8.3 GIC-600 and NVIC Register Configuration

■ GIC-600 Register

Refer to Arm® CoreLink™ GIC-600 Generic Interrupt Controller Revision: r1p6 Technical Reference Manual.
ITS and LPI are not supported.

Base Address: H'0_1240_0000 (Cortex-A55 Address Space)

Base Address: H'5240_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)
Base Address: H'4240_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

■ NVIC Register

Refer to Arm® Cortex®-M33 Processor Revision: r0p4 Technical Reference Manual.

8.4 GIC-600 and NVIC Register Descriptions

■ GIC-600 Register

Refer to Arm® CoreLink™ GIC-600 Generic Interrupt Controller Revision: r1p6 Technical Reference Manual for detail.

■ NVIC Register

Refer to Arm® Cortex®-M33 Processor Revision: r0p4 Technical Reference Manual for detail.

8.5 IA55 and IM33/IM33_FPU Register Configuration

■ IA55

Base Address: H'0_1105_0000 (Cortex-A55Address Space)

Base Address: H'5105_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)
Base Address: H'4105_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)

■ IM33

Base Address: H'0_1106_0000 (Cortex-A55 Address Space)

Base Address: H'5106_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)
Base Address: H'4106_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)

■ IM33_FPU

Base Address: H'0_1107_0000 (Cortex-A55 Address Space)

Base Address: H'5107_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)
Base Address: H'4107_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

The following table shows IA55 and IM33/IM33_FPU Register list.
Prohibit to write undefined area.

Table 8.3 IA55 and IM33/IM33_FPU Register List

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
NMI Status Control Register	NSCR	R/W	H'0000_0000	H'0000	32
NMI Interrupt Type Selection Register	NITSR	R/W	H'0000_0000	H'0004	32
IRQ Status Control Register	ISCR	R/W	H'0000_0000	H'0010	32
IRQ Interrupt Type Selection Register	IITSR	R/W	H'0000_0000	H'0014	32
TINT Status Control Register	TSCR	R/W	H'0000_0000	H'0020	32
TINT Interrupt Type Selection Register0	TITSR0	R/W	H'0000_0000	H'0024	32
TINT Interrupt Type Selection Register1	TITSR1	R/W	H'0000_0000	H'0028	32
TINT Source Selection Register0	TSSR0	R/W	H'0000_0000	H'0030	32
TINT Source Selection Register1	TSSR1	R/W	H'0000_0000	H'0034	32
TINT Source Selection Register2	TSSR2	R/W	H'0000_0000	H'0038	32
TINT Source Selection Register3	TSSR3	R/W	H'0000_0000	H'003C	32
TINT Source Selection Register4	TSSR4	R/W	H'0000_0000	H'0040	32
TINT Source Selection Register5	TSSR5	R/W	H'0000_0000	H'0044	32
TINT Source Selection Register6	TSSR6	R/W	H'0000_0000	H'0048	32
TINT Source Selection Register7	TSSR7	R/W	H'0000_0000	H'004C	32
Bus Error Interrupt Status Control Register0	BEISR0	R/W	H'0000_0000	H'0050	32
Bus Error Interrupt Status Control Register1	BEISR1	R/W	H'0000_0000	H'0054	32
ECCRAM Error Interrupt Status Control Register0	EREISR0	R/W	H'0000_0000	H'0060	32
ECCRAM Error Interrupt Status Control Register1	EREISR1	R/W	H'0000_0000	H'0064	32
ECCRAM Error Interrupt Status Control Register2	EREISR2	R/W	H'0000_0000	H'0068	32
ECCRAM Error Interrupt Status Control Register3	EREISR3	R/W	H'0000_0000	H'006C	32
ECCRAM Error Interrupt Status Control Register4	EREISR4	R/W	H'0000_0000	H'0070	32
ECCRAM Error Interrupt Status Control Register5	EREISR5	R/W	H'0000_0000	H'0074	32
Interrupt Selection (GPT/MTU3a) Register0	INTPMSEL0	R/W	H'0000_0000	H'0080	32
Interrupt Selection (GPT/MTU3a) Register1	INTPMSEL1	R/W	H'0000_0000	H'0084	32

8.6 IA55 and IM33/IM33_FPU Register Descriptions

8.6.1 NMI Status Control Register (NSCR)

This register shows NMI status.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NSMON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NSTAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	NSMON	0	R	This bit indicates NMI pin signal level. [Read operation] 0: NMI pin is Low-level. 1: NMI pin is High-level. [Write operation] Invalid to write.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	NSTAT	0	R/W	This bit indicates NMI interrupt status. NMI interrupt status can be cleared writing 0 in case NSTAT is 1. [Read operation] 0: NMI interrupt is not detected. 1: NMI interrupt is detected. [Write operation] • In case NSTAT is 1 0: NMI interrupt status is cleared. 1: Invalid to write. • In case NSTAT is 0 Invalid to write.

8.6.2 NMI Interrupt Type Selection Register (NITSR)

This register selects NMI detecting method.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NTSEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	NTSEL	0	R/W	This bit is used to select NMI interrupt detection method. 0: Falling-edge detection. 1: Rising-edge detection.

8.6.3 IRQ Status Control Register (ISCR)

This register shows IRQ status.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ISTAT7	ISTAT6	ISTAT5	ISTAT4	ISTAT3	ISTAT2	ISTAT1	ISTAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
n	ISTATn	0	R/W	This bit indicates IRQn interrupt status. When the interrupt detection method is set to an edge type detection in IITSR, IRQn interrupt status can be cleared writing 0 in case ISTATn is 1. [Read operation] 0: IRQn interrupt is not detected. 1: IRQn interrupt is detected. [Write operation] When “Falling-edge detection”, “Rising-edge detection” or “Falling/Rising-edge detection” is set in ITSR: • In case ISTAT is 1 0: IRQn interrupt detection status is cleared. 1: Invalid to write. • In case ISTAT is 0 Invalid to write. When “Low-level detection” is set in IITSR: Invalid to write.

Note: n = 7 to 0

8.6.4 IRQ Interrupt Type Selection Register (IITSR)

This register selects IRQ detecting method.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IITSEL7		IITSEL6		IITSEL5		IITSEL4		IITSEL3		IITSEL2		IITSEL1		IITSEL0	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2n+1 to 2n	IITSEL	All 0	R/W	This bit is used to select IRQn interrupt detection method. 00: Low-level detection. 01: Falling-edge detection. 10: Rising-edge detection. 11: Falling/Rising-edge detection.

Note: n = 7 to 0

8.6.5 TINT Status Control Register (TSCR)

This register shows TINT status.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSTAT 31	TSTAT 30	TSTAT 29	TSTAT 28	TSTAT 27	TSTAT 26	TSTAT 25	TSTAT 24	TSTAT 23	TSTAT 22	TSTAT 21	TSTAT 20	TSTAT 19	TSTAT 18	TSTAT 17	TSTAT 16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSTAT 15	TSTAT 14	TSTAT 13	TSTAT 12	TSTAT 11	TSTAT 10	TSTAT9	TSTAT8	TSTAT7	TSTAT6	TSTAT5	TSTAT4	TSTAT3	TSTAT2	TSTAT1	TSTAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
n	TSTATn	All 0	R/W	This bit indicates TINTn interrupt status. When the interrupt detection method is set to edge type detection in TITSR0/1, TINTn interrupt status can be cleared writing 0 in case TSTATn is 1. [Read operation] 0: TINTn interrupt is not detected. 1: TINTn interrupt is detected. [Write operation] When “Rising-edge detection” or “Falling-edge detection” is set in TITSR0/1: • In case TSTAT is 1 0: TINTn interrupt detection status is cleared. 1: Invalid to write. • In case TSTAT is 0 Invalid to write. When “Low-level detection” or ”High-Level detection” is set in TITSR0/1: Invalid to write.

Note: n = 31 to 0

8.6.6 TINT Interrupt Type Selection Register0 (TITSR0)

This register selects detecting method of TINT15 to TINT0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TITSEL15		TITSEL14		TITSEL13		TITSEL12		TITSEL11		TITSEL10		TITSEL9		TITSEL8	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TITSEL7		TITSEL6		TITSEL5		TITSEL4		TITSEL3		TITSEL2		TITSEL1		TITSEL0	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
2n+1 to 2n	TITSELn	All 0	R/W	This bit is used to select TINTn interrupt detection method. 00: Rising-edge detection. 01: Falling-edge detection. 10: High-level detection. 11: Low-level detection.

Note: n = 15 to 0

8.6.7 TINT Interrupt Type Selection Register1 (TITSR1)

This register selects detecting method of TINT31 to TINT16.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TITSEL31		TITSEL30		TITSEL29		TITSEL28		TITSEL27		TITSEL26		TITSEL25		TITSEL24	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TITSEL23		TITSEL22		TITSEL21		TITSEL20		TITSEL19		TITSEL18		TITSEL17		TITSEL16	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
2n+1 to 2n	TITSEL n+16	All 0	R/W	This bit is used to select TINTn+16 interrupt detection method. 00: Rising-edge detection. 01: Falling-edge detection. 10: High-level detection. 11: Low-level detection.

Note: n = 15 to 0

8.6.8 TINT Source Selection Register0 (TSSR0)

This register selects the interrupt source of TINT3 to TINT0.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN3	TSSEL3							TIEN2	TSSEL2						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN1	TSSEL1							TIEN0	TSSEL0						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
8n+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8n+6 to 8n	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1010001: TINTn = GPIOINT81 1010010: Cannot be set. ~ 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 3 to 0

8.6.9 TINT Source Selection Register1 (TSSR1)

This register selects the interrupt source of TINT7 to TINT4.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN7	TSSEL7							TIEN6	TSSEL6						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN5	TSSEL5							TIEN4	TSSEL4						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
8(n-4)+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8(n-4)+6 to 8(n-4)	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1010001: TINTn = GPIOINT81 1010010: Cannot be set. ~ 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 7 to 4

8.6.10 TINT Source Selection Register2 (TSSR2)

This register selects the interrupt source of TINT11 to TINT8.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN11	TSSEL11							TIEN10	TSSEL10						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN9	TSSEL9							TIEN8	TSSEL8						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

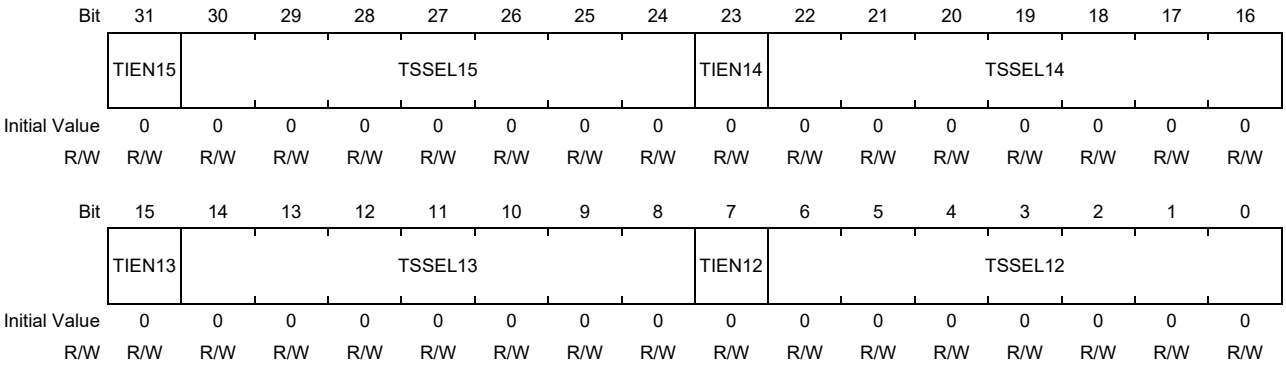
Bit	Bit Name	Initial Value	R/W	Description
8(n-8)+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8(n-8)+6 to 8(n-8)	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1010001: TINTn = GPIOINT81 1010010: Cannot be set. ~ 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 11 to 8

8.6.11 TINT Source Selection Register3 (TSSR3)

This register selects the interrupt source of TINT15 to TINT12.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.



Bit	Bit Name	Initial Value	R/W	Description
8(n-12)+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8(n-12)+6 to 8(n-12)	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1010001: TINTn = GPIOINT81 1010010: Cannot be set. ~ 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 15 to 12

8.6.12 TINT Source Selection Register4 (TSSR4)

This register selects the interrupt source of TINT19 to TINT16.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN19	TSSEL19							TIEN18	TSSEL18						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN17	TSSEL17							TIEN16	TSSEL16						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
8(n-16)+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8(n-16)+6 to 8(n-16)	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1010001: TINTn = GPIOINT81 1010010: Cannot be set. ~ 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 19 to 16

8.6.13 TINT Source Selection Register5 (TSSR5)

This register selects the interrupt source of TINT23 to TINT20.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN23	TSSEL23							TIEN22	TSSEL22						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN21	TSSEL21							TIEN20	TSSEL20						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
8(n-20)+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8(n-20)+6 to 8(n-20)	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1010001: TINTn = GPIOINT81 1010010: Cannot be set. ~ 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 23 to 20

8.6.14 TINT Source Selection Register6 (TSSR6)

This register selects the interrupt source of TINT27 to TINT24.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN27	TSSEL27							TIEN26	TSSEL26						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN25	TSSEL25							TIEN24	TSSEL24						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
8(n-24)+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8(n-24)+6 to 8(n-24)	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1010001: TINTn = GPIOINT81 1010010: Cannot be set. ~ 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 27 to 24

8.6.15 TINT Source Selection Register7 (TSSR7)

This register selects the interrupt source of TINT31 to TINT28.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN31	TSSEL31							TIEN30	TSSEL30						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN29	TSSEL29							TIEN28	TSSEL28						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
8(n-28)+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8(n-28)+6 to 8(n-28)	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1010001: TINTn = GPIOINT81 1010010: Cannot be set. ~ 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 31 to 28

8.6.16 Bus Error Interrupt Status Control Register0 (BEISR0)

This register shows Bus error status.

Refer to LSI Internal Bus section and **Section 8.7.4.1** for bus error interrupt cause.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BESTAT31	BESTAT30	BESTAT29	BESTAT28	BESTAT27	BESTAT26	BESTAT25	BESTAT24	BESTAT23	BESTAT22	BESTAT21	BESTAT20	BESTAT19	BESTAT18	BESTAT17	BESTAT16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BESTAT15	BESTAT14	BESTAT13	TSTAT12	BESTAT11	BESTAT10	BESTAT9	BESTAT8	BESTAT7	BESTAT6	BESTAT5	BESTAT4	BESTAT3	BESTAT2	BESTAT1	BESTAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
n	BESTATn	All 0	R/W	This bit indicates BUSERR_INTn interrupt status. BUSERR_INTn interrupt status can be cleared writing 0 in case BESTAT is 1. [Read operation] 0: BUSERR_INTn is not detected. 1: BUSERR_INTn is detected. [Write operation] • In case BESTATn is 1 0: BUSERR_INTn interrupt status is cleared. 1: Invalid to write. • In case BESTATn is 0 Invalid to write.

Note: n = 31 to 0

8.6.17 Bus Error Interrupt Status Control Register1 (BEISR1)

This register shows Bus error status.

Refer to LSI Internal Bus section and **Section 8.7.4.1** for bus error interrupt cause.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BESTAT46	BESTAT45	BESTAT44	BESTAT43	BESTAT42	BESTAT41	BESTAT40	BESTAT39	BESTAT38	BESTAT37	BESTAT36	BESTAT35	BESTAT34	BESTAT33	BESTAT32
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
n-32	BESTATn	All 0	R/W	This bit indicates BUSERR_INTn interrupt status. BUSERR_INTn interrupt status can be cleared writing 0 in case BESTAT is 1. [Read operation] 0: BUSERR_INTn is not detected. 1: BUSERR_INTn is detected. [Write operation] • In case BESTATn is 1 0: BUSERR_INTn interrupt status is cleared. 1: Invalid to write. • In case BESTATn is 0 Invalid to write.

Note: n = 46 to 32

8.6.18 ECCRAM Error Interrupt Status Control Register0 (EREISR0)

This register shows ECC 1bit Error status of ECCRAMA0 and ECCRAMA1.

Refer to **Section 8.7.4.2** for ECCRAM error interrupt cause.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	A11E STAT7	A11 ESTAT6	A11E STAT5	A11E STAT4	A11E STAT3	A11E STAT2	A11E STAT1	A11E STAT0	A01E STAT7	A01E STAT6	A01E STAT5	A01E STAT4	A01E STAT3	A01E STAT2	A01E STAT1	A01E STAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
n+8	A11ESTATn	All 0	R/W	This bit indicates ECCRAMA1_1En interrupt status. ECCRAMA1_1En interrupt status can be cleared writing 0 in case A11ESTATn is 1. [Read operation] 0: ECCRAMA1_1En is not detected. 1: ECCRAMA1_1En is detected. Write operation: • In case A11ESTATn is 1 0: ECCRAMA1_1En interrupt status is cleared. 1: Invalid to write. • In case A11ESTATn is 0 Invalid to write.
n	A01ESTATn	All 0	R/W	This bit indicates ECCRAMA0_1En interrupt status. ECCRAMA0_1En interrupt status can be cleared writing 0 in case A01ESTATn is 1. [Read operation] 0: ECCRAMA0_1En is not detected. 1: ECCRAMA0_1En is detected. [Write operation] • In case A01ESTATn is 1 0: ECCRAMA0_1En interrupt status is cleared. 1: Invalid to write. • In case A01ESTATn is 0 Invalid to write.

Note: n = 7 to 0

8.6.19 ECCRAM Error Interrupt Status Control Register1 (EREISR1)

This register shows ECC 1bit Error status of ECCRAMM0 and ECCSRAMM1.

Refer to **Section 8.7.4.2** for ECCRAM error interrupt cause.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	M11E STAT7	M11E STAT6	M11E STAT5	M11E STAT4	M11E STAT3	M11E STAT2	M11E STAT1	M11E STAT0	M01E STAT7	M01E STAT6	M01E STAT5	M01E STAT4	M01E STAT3	M01E STAT2	M01E STAT1	M01E STAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
n+8	M11ESTATn	All 0	R/W	This bit indicates ECCRAMM1_1En interrupt status. ECCRAMM1_1En interrupt status can be cleared writing 0 in case M11ESTATn is 1. [Read operation] 0: ECCRAMM1_1En is not detected. 1: ECCRAMM1_1En is detected. Write operation: • In case M11ESTATn is 1 0: ECCRAMM1_1En interrupt status is cleared. 1: Invalid to write. • In case M11ESTATn is 0 Invalid to write.
n	M01ESTATn	All 0	R/W	This bit indicates ECCRAMM0_1En interrupt status. ECCRAMM0_1En interrupt status can be cleared writing 0 in case M01ESTATn is 1. [Read operation] 0: ECCRAMM0_1En is not detected. 1: ECCRAMM0_1En is detected. [Write operation] • In case M01ESTATn is 1 0: ECCRAMM0_1En interrupt status is cleared. 1: Invalid to write. • In case M01ESTATn is 0 Invalid to write.

Note: n = 7 to 0

8.6.20 ECCRAM Error Interrupt Status Control Register0 (EREISR2)

This register shows ECC 2bit Error status of ECCRAMA0 and ECCRAMA1.

Refer to **Section 8.7.4.2** for ECCRAM error interrupt cause.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	A12E STAT7	A12E STAT6	A12E STAT5	A12E STAT4	A11E STAT3	A12E STAT2	A12E STAT1	A12E STAT0	A02E STAT7	A02E STAT6	A02E STAT5	A02E STAT4	A02E STAT3	A02E STAT2	A02E STAT1	A02E STAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
n+8	A12ESTATn	All 0	R/W	This bit indicates ECCRAMA1_2En interrupt status. ECCRAMA1_2En interrupt status can be cleared writing 0 in case A12ESTATn is 1. [Read operation] 0: ECCRAMA1_2En is not detected. 1: ECCRAMA1_2En is detected. Write operation: • In case A12ESTATn is 1 0: ECCRAMA1_12n interrupt status is cleared. 1: Invalid to write. • In case A12ESTATn is 0 Invalid to write.
n	A02ESTATn	All 0	R/W	This bit indicates ECCRAMA0_2En interrupt status. ECCRAMA0_2En interrupt status can be cleared writing 0 in case A02ESTATn is 1. [Read operation] 0: ECCRAMA0_2En is not detected. 1: ECCRAMA0_2En is detected. [Write operation] • In case A02ESTATn is 1 0: ECCRAMA0_2En interrupt status is cleared. 1: Invalid to write. • In case A02ESTATn is 0 Invalid to write.

Note: n = 7 to 0

8.6.21 ECCRAM Error Interrupt Status Control Register1 (EREISR3)

This register shows ECC 2bit Error status of ECCRAMM0 and ECCSRAMM1.

Refer to **Section 8.7.4.2** for ECCRAM error interrupt cause.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	M12E STAT7	M12E STAT6	M12E STAT5	M12E STAT4	M12E STAT3	M12E STAT2	M12E STAT1	M12E STAT0	M02E STAT7	M02E STAT6	M02E STAT5	M02E STAT4	M02E STAT3	M02E STAT2	M02E STAT1	M02E STAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
n+8	M12ESTATn	All 0	R/W	This bit indicates ECCRAMM1_2En interrupt status. ECCRAMM1_2En interrupt status can be cleared writing 0 in case M12ESTATn is 1. [Read operation] 0: ECCRAMM1_2En is not detected. 1: ECCRAMM1_2En is detected. Write operation: • In case M12ESTATn is 1 0: ECCRAMM1_2En interrupt status is cleared. 1: Invalid to write. • In case M12ESTATn is 0 Invalid to write.
n	M02ESTATn	All 0	R/W	This bit indicates ECCRAMM0_2En interrupt status. ECCRAMM0_2En interrupt status can be cleared writing 0 in case M02ESTATn is 1. [Read operation] 0: ECCRAMM0_2En is not detected. 1: ECCRAMM0_2En is detected. [Write operation] • In case M02ESTATn is 1 0: ECCRAMM0_2En interrupt status is cleared. 1: Invalid to write. • In case M02ESTATn is 0 Invalid to write.

Note: n = 7 to 0

8.6.22 ECCRAM Error Interrupt Status Control Register0 (EREISR4)

This register shows ECC Overflow Error status of ECCRAMA0 and ECCRAMA1.

Refer to **Section 8.7.4.2** for ECCRAM error interrupt cause.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	A1OF STAT7	A1OF STAT6	A1OF STAT5	A1OF STAT4	A1OF STAT3	A1OF STAT2	A1OF STAT1	A1OF STAT0	A0OF STAT7	A0OF STAT6	A0OF STAT5	A0OF STAT4	A0OF STAT3	A0OF STAT2	A0OF STAT1	A0OF STAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
n+8	A1OFSTATn	All 0	R/W	This bit indicates ECCRAMA1_OFn interrupt status. ECCRAMA1_OFn interrupt status can be cleared writing 0 in case A1OFSTATn is 1. [Read operation] 0: ECCRAMA1_OFn is not detected. 1: ECCRAMA1_OFn is detected. Write operation: • In case A1OFSTATn is 1 0: ECCRAMA1_OFn interrupt status is cleared. 1: Invalid to write. • In case A1OFSTATn is 0 Invalid to write.
n	A0OFSTATn	All 0	R/W	This bit indicates ECCRAMA0_OFn interrupt status. ECCRAMA0_OFn interrupt status can be cleared writing 0 in case A0OFSTATn is 1. [Read operation] 0: ECCRAMA0_OFn is not detected. 1: ECCRAMA0_OFn is detected. [Write operation] • In case A0OFSTATn is 1 0: ECCRAMA0_OFn interrupt status is cleared. 1: Invalid to write. • In case A0OFSTATn is 0 Invalid to write.

Note: n = 7 to 0

8.6.23 ECCRAM Error Interrupt Status Control Register1 (EREISR5)

This register shows ECC Overflow Error status of ECCRAMM0 and ECCSRAMM1.

Refer to **Section 8.7.4.2** for ECCRAM error interrupt cause.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	M1OF STAT7	M1OF STAT6	M1OF STAT5	M1OF STAT4	M1OF STAT3	M1OF STAT2	M1OF STAT1	M1OF STAT0	M0OF STAT7	M0OF STAT6	M0OF STAT5	M0OF STAT4	M0OF STAT3	M0OF STAT2	M0OF STAT1	M0OF STAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
n+8	M1OFSTATn	All 0	R/W	This bit indicates ECCRAMM1_OFn interrupt status. ECCRAMM1_OFn interrupt status can be cleared writing 0 in case M1OFSTATn is 1. [Read operation] 0: ECCRAMM1_OFn is not detected. 1: ECCRAMM1_OFn is detected. Write operation: • In case M1OFSTATn is 1 0: ECCRAMM1_OFn interrupt status is cleared. 1: Invalid to write. • In case M1OFSTATn is 0 Invalid to write.
n	M0OFSTATn	All 0	R/W	This bit indicates ECCRAMM0_OFn interrupt status. ECCRAMM0_OFn interrupt status can be cleared writing 0 in case M0OFSTATn is 1. [Read operation] 0: ECCRAMM0_OFn is not detected. 1: ECCRAMM0_OFn is detected. [Write operation] • In case M0OFSTATn is 1 0: ECCRAMM0_OFn interrupt status is cleared. 1: Invalid to write. • In case M0OFSTATn is 0 Invalid to write.

Note: n = 7 to 0

8.6.24 Interrupt Selection (GPT/MTU3a) Register0 (INTPMSEL0)

This register selects the GPT interrupts and MTU3a interrupts.

Refer to **Section 8.7.5** for the interrupt mapping.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PMINTSEL31	PMINTSEL30	PMINTSEL29	PMINTSEL28	PMINTSEL27	PMINTSEL26	PMINTSEL25	PMINTSEL24	PMINTSEL23	PMINTSEL22	PMINTSEL21	PMINTSEL20	PMINTSEL19	PMINTSEL18	PMINTSEL17	PMINTSEL16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMINTSEL15	PMINTSEL14	PMINTSEL13	PMINTSEL12	PMINTSEL11	PMINTSEL10	PMINTSEL9	PMINTSEL8	PMINTSEL7	PMINTSEL6	PMINTSEL5	PMINTSEL4	PMINTSEL3	PMINTSEL2	PMINTSEL1	PMINTSEL0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
n	PMINTSELn	All 0	R/W	This bit selects GPT interrupt and MTU3a interrupt. 0: GPT interrupt 1: MTU3a interrupt

Note: n = 31 to 0

8.6.25 Interrupt Selection (GPT/MTU3a) Register1 (INTPMSEL1)

This register selects the GPT interrupts and MTU3a interrupts.

Refer to **Section 8.7.5** for the interrupt mapping.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PMINTSEL43	PMINTSEL42	PMINTSEL41	PMINTSEL40	PMINTSEL39	PMINTSEL38	PMINTSEL37	PMINTSEL36	PMINTSEL35	PMINTSEL34	PMINTSEL33	PMINTSEL32
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
n	PMINTSEL n+32	All 0	R/W	This bit selects GPT interrupt and MTU3a interrupt. 0: GPT interrupt 1: MTU3a interrupt

Note: n = 11 to 0

8.7 Operation

8.7.1 NMI Interrupt

NMI interrupt is the highest priority interrupt by default even though it can be masked.

NMI is not treated as NMI exception.

[Setting]

- A noise filter function of NMI pin can be enabled by GPIO register setting.
  Refer to **Section 45.4.1, Operation for GPIO Function** for detail.
- NMI pin input is detected by edge. A detection method is set by NTSEL bit of NMI Interrupt Type Selection Register (NITSR) in IA55/IM33/IM33_FPU. It can be selected “Rising-edge detection” or “Falling-edge detection”.

[Status Control]

- NMI interrupt request can be confirmed by reading NSTAT bit of NMI Status Control Register (NSCR) in IA55/IM33/IM33_FPU.
  The interrupt request can be cleared by writing 0 to NSTAT bit.
- A signal level (High or Low) of NMI pin can be confirmed by reading NSMON bit of NMI Status Control Register (NSCR) in IA55/IM33/IM33_FPU.

8.7.2 IRQ Interrupt

IRQ interrupt is the interrupt from IRQ0-7 input pins.

[Setting]

- Using IRQ pins as interrupt must be mapped GPIO pins onto IRQ pins by GPIO register setting.
  Refer to **Section 45.4.2, Operation for Peripheral Function** for detail.
- A noise filter function of IRQ pins can be enabled by GPIO register setting.
  Refer to **Section 45.4.1, Operation for GPIO Function** for detail.
- IRQ pin inputs are detected by a level or an edge. A detection method is set by ITSEL bit of IRQ Interrupt Type Selection Register (IITSR) in IA55/IM33/IM33_FPU. It can be selected “Low-level detection”, “Falling-edge detection”, “Rising-edge detection” or “Falling/Rising-edge detection”.

[Status Control]

- IRQ interrupt requests can be confirmed by reading ISTATn bit of IRQ Status Control Register (ISCR) in IA55/IM33/IM33_FPU
- In case selected a level detection in IITSR, the interrupt requests are cleared when the interrupt sources de-assert the interrupts.
- In case selected an edge detection in IITSR, the interrupt requests can be cleared by writing 0 to ISTATn bit.

8.7.3 GPIO Interrupt (TINT)

GPIO interrupt is the interrupt using GPIO pins as external interrupt input pins.

[Setting]

- When using GPIO pins as interrupts, assign GPIO pins as external interrupt input pins (GPIOINT0-81) by GPIO register setting.
  - Refer to **Section 45.4.1, Operation for GPIO Function** for details.
  - Refer to **Table 8.4** for mapping GPIO pins onto GPIOINT0-81.
- The noise filter function of GPIO pins assigned to TINT can be enabled by GPIO register setting.
  - Refer to **Section 45.4.1, Operation for GPIO Function** for details.
- Assign any 32 interrupt sources to TINT from GPIOINT0-81 by TINT Source Selection Register (TSSR0-7) in IA55/IM33/IM33_FPU.

  Note: Even if the TINT is assigned by TSSR0-7, no interrupt request is generated unless it is assigned to an external pin interrupt by GPIO registers.
- TINT inputs are detected by a level or an edge. A detection method is set by TTSEL bit of TINT Interrupt Type Selection Register (TITSR0/1) in IA55/IM33/IM33_FPU. It can be selected “Rising-edge detection”, “Falling-edge detection”, “High-level detection” or “Low-level detection”.

[Status Control]

- TINT interrupt requests can be confirmed by reading TSTATn bit of TINT Status Control Register (TSCR) in IA55/IM33/IM33_FPU.
- In case selected a level detection in TITSR0/1, the interrupt requests are cleared when the interrupt sources de-assert the interrupts.
- In case selected an edge detection in TITSR0/1, the interrupt requests can be cleared by writing 0 to TSTATn bit.

Table 8.4 GPIOINT Mapping

No.	GPIO Pin	GPIOINT
1	P0_0	GPIOINT0
2	P0_1	GPIOINT1
3	P0_2	GPIOINT2
4	P0_3	GPIOINT3
5	P1_0	GPIOINT4
6	P1_1	GPIOINT5
7	P1_2	GPIOINT6
8	P1_3	GPIOINT7
9	P1_4	GPIOINT8
10	P2_0	GPIOINT9
11	P2_1	GPIOINT10
12	P2_2	GPIOINT11
13	P2_3	GPIOINT12
14	P3_0	GPIOINT13
15	P3_1	GPIOINT14
16	P3_2	GPIOINT15
17	P3_3	GPIOINT16
18	P4_0	GPIOINT17
19	P4_1	GPIOINT18
20	P4_2	GPIOINT19
21	P4_3	GPIOINT20
22	P4_4	GPIOINT21
23	P4_5	GPIOINT22
24	P5_0	GPIOINT23
25	P5_1	GPIOINT24
26	P5_2	GPIOINT25
27	P5_3	GPIOINT26
28	P5_4	GPIOINT27
29	P6_0	GPIOINT28
30	P6_1	GPIOINT29
31	P6_2	GPIOINT30
32	P6_3	GPIOINT31
33	P6_4	GPIOINT32
34	P7_0	GPIOINT33
35	P7_1	GPIOINT34
36	P7_2	GPIOINT35
37	P7_3	GPIOINT36
38	P7_4	GPIOINT37
39	P8_0	GPIOINT38
40	P8_1	GPIOINT39
41	P8_2	GPIOINT40
42	P8_3	GPIOINT41
43	P8_4	GPIOINT42
44	P9_0	GPIOINT43
45	P9_1	GPIOINT44
46	P9_2	GPIOINT45
47	P9_3	GPIOINT46
48	P10_0	GPIOINT47
49	P10_1	GPIOINT48
50	P10_2	GPIOINT49
51	P10_3	GPIOINT50
52	P10_4	GPIOINT51
53	P11_0	GPIOINT52
54	P11_1	GPIOINT53
55	P11_2	GPIOINT54
56	P11_3	GPIOINT55
57	P12_0	GPIOINT56
58	P12_1	GPIOINT57
59	P13_0	GPIOINT58
60	P13_1	GPIOINT59
61	P13_2	GPIOINT60
62	P13_3	GPIOINT61
63	P13_4	GPIOINT62
64	P14_0	GPIOINT63
65	P14_1	GPIOINT64
66	P14_2	GPIOINT65
67	P15_0	GPIOINT66
68	P15_1	GPIOINT67
69	P15_2	GPIOINT68
70	P15_3	GPIOINT69
71	P16_0	GPIOINT70
72	P16_1	GPIOINT71
73	P17_0	GPIOINT72
74	P17_1	GPIOINT73
75	P17_2	GPIOINT74
76	P17_3	GPIOINT75
77	P18_0	GPIOINT76
78	P18_1	GPIOINT77
79	P18_2	GPIOINT78
80	P18_3	GPIOINT79
81	P18_4	GPIOINT80
82	P18_5	GPIOINT81

8.7.4 Internal Interrupt

8.7.4.1 Bus Error Interrupt

Bus error interrupts (BUSERR_INT0-46) generated by system bus are integrated into one interrupt (BUS_ERR_INT) by IA55/IM33/IM33_FPU and the integrated interrupt is notified to CPU.

[Status Control]

- Bus error interrupt requests can be confirmed by reading BUS Error Interrupt Status Control Register (BEISR0/1) in IA55/IM33/IM33_FPU.
  - Refer to **Table 8.5** and **Table 8.6** for mapping BESTAT0-46 onto bus error interrupt causes.
- The interrupt requests can be cleared by writing 0 to BESTAT0-46 bit.

Table 8.5 BEISR0 Bus Error Mapping

BEISR0	Interrupt	Description
BESTAT0	BUSERR_INT0	Bus Write Error Interrupt for CA55
BESTAT1	BUSERR_INT1	Bus Read Error Interrupt for CA55
BESTAT2	BUSERR_INT2	Bus Write Error Interrupt for CM33_FPU_S
BESTAT3	BUSERR_INT3	Bus Read Error Interrupt for CM33_FPU_S
BESTAT4	BUSERR_INT4	Bus Write Error Interrupt for CM33_FPU_C
BESTAT5	BUSERR_INT5	Bus Read Error Interrupt for CM33_FPU_C
BESTAT6	BUSERR_INT6	AHB5toAXI3 Bridge interrupt
BESTAT7	BUSERR_INT7	AHB5toAXI3 Bridge interrupt
BESTAT8	BUSERR_INT8	TZC-400 interrupt (TZC_SRAM_ACPU0_INT)
BESTAT9	BUSERR_INT9	TZC-400 interrupt (TZC_SRAM_ACPU1_INT)
BESTAT10	BUSERR_INT10	Reserved
BESTAT11	BUSERR_INT11	Bus Write Error Interrupt for SDHI Ch0
BESTAT12	BUSERR_INT12	Bus Read Error Interrupt for SDHI Ch0
BESTAT13	BUSERR_INT13	Bus Write Error Interrupt for SDHI Ch1
BESTAT14	BUSERR_INT14	Bus Read Error Interrupt for SDHI Ch1
BESTAT15	BUSERR_INT15	Bus Write Error Interrupt for Gether Ch0
BESTAT16	BUSERR_INT16	Bus Read Error Interrupt for Gether Ch0
BESTAT17	BUSERR_INT17	Bus Write Error Interrupt for Gether Ch1
BESTAT18	BUSERR_INT18	Bus Read Error Interrupt for Gether Ch1
BESTAT19	BUSERR_INT19	Bus Write Error Interrupt for USB Ch0 Host
BESTAT20	BUSERR_INT20	Bus Read Error Interrupt for USB Ch0 Host
BESTAT21	BUSERR_INT21	Bus Write Error Interrupt for USB Ch0 Function
BESTAT22	BUSERR_INT22	Bus Read Error Interrupt for USB Ch0 Function
BESTAT23	BUSERR_INT23	Bus Write Error Interrupt for USB Ch1 Host
BESTAT24	BUSERR_INT24	Bus Read Error Interrupt for USB Ch1 Host
BESTAT25	BUSERR_INT25	Bus Write Error Interrupt for SDHI Ch2
BESTAT26	BUSERR_INT26	Bus Read Error Interrupt for SDHI Ch2
BESTAT27	BUSERR_INT27	Bus Write Error Interrupt for PCIe
BESTAT28	BUSERR_INT28	Bus Read Error Interrupt for PCIe
BESTAT29	BUSERR_INT29	Bus Write Error Interrupt for CST_ETR
BESTAT30	BUSERR_INT30	Bus Read Error Interrupt for CST_ETR
BESTAT31	BUSERR_INT31	Bus Write Error Interrupt for CST_AP

Table 8.6 BEISR1 Bus Error Mapping

BEISR1	Interrupt	Description
BESTAT32	BUSERR_INT32	Bus Read Error Interrupt for CST_AP
BESTAT33	BUSERR_INT33	Bus Write Error Interrupt for CM33_S
BESTAT34	BUSERR_INT34	Bus Read Error Interrupt for CM33_S
BESTAT35	BUSERR_INT35	Bus Write Error Interrupt for CM33_C
BESTAT36	BUSERR_INT36	Bus Read Error Interrupt for CM33_C
BESTAT37	BUSERR_INT37	Bus Write Error Interrupt for DMAC_S
BESTAT38	BUSERR_INT38	Bus Read Error Interrupt for DMAC_S
BESTAT39	BUSERR_INT39	Bus Write Error Interrupt for DMAC_NS
BESTAT40	BUSERR_INT40	Bus Read Error Interrupt for DMAC_NS
BESTAT41	BUSERR_INT41	TZC-400 interrupt (TZC_SRAM_MCPU0__INT)
BESTAT42	BUSERR_INT42	TZC-400 interrupt (TZC_SRAM_MCPU1_INT)
BESTAT43	BUSERR_INT43	TZC-400 interrupt (TZC_XSPI_INT)
BESTAT44	BUSERR_INT44	AHB5toAXI3 Bridge interrupt
BESTAT45	BUSERR_INT45	AHB5toAXI3 Bridge interrupt
BESTAT46	BUSERR_INT46	TZC-400 interrupt (TZC_DDR_INT)

8.7.4.2 ECCRAM Error Interrupt

ECCRAM error interrupts generated by On-Chip RAM (ECCRAMA0, ECCRAMA1, ECCRAMM0, ECCRAMM1) are integrated with IA55/IM33/IM33_FPU for each interrupt cause and notified to the CPU.

Each On-Chip RAM outputs eight 1-bit ECC error interrupts, eight 2-bit ECC error interrupts and eight overflow error interrupts. These interrupts are integrated per interrupt cause.

- Integrated 1bit error interrupt: EC7TIE1
- Integrated 2bit error interrupt: EC7TIE2
- Integrated overflow error interrupt: EC7TIOVF

[Status Control]

- ECCRAM error interrupt requests can be confirmed by reading ECCRAM Error Interrupt Status Control Register (EREISR0 to EREISR7) in IA55/IM33/IM33_FPU.
  - Refer to **Table 8.7** for mapping A01ESTAT0-7, A11ESTAT0-7, M01ESTAT0-7 and M11ESTAT0-7 onto ECCRAM error interrupt causes.
  - Refer to **Table 8.8** for mapping A02ESTAT0-7, A12ESTAT0-7, M0E2STAT0-7 and M2E2STAT0-7 onto ECCRAM error interrupt causes.
  - Refer to **Table 8.9** for mapping A0OFSTAT0-7, A1OFSTAT0-7, M0OFSTAT0-7 and M1OFSTAT0-7 onto ECCRAM error interrupt causes.
- The interrupt requests can be cleared by writing 0 to corresponding bit of the EREISR0 to EREISR7.

Table 8.7 ECC 1bit Error Mapping

Register Name	Register Status Bit	Interrupt	Description	Integrated Interrupt
EREISR0	A01ESTAT0	ECCRAMA0_1E0	ECC 1bit Error interrupt bit[31:0]	EC7TIE1
	A01ESTAT1	ECCRAMA0_1E1	ECC 1bit Error interrupt bit[63:32]	
	A01ESTAT2	ECCRAMA0_1E2	ECC 1bit Error interrupt bit[95:64]	
	A01ESTAT3	ECCRAMA0_1E3	ECC 1bit Error interrupt bit[127:96]	
	A01ESTAT4	ECCRAMA0_1E4	ECC 1bit Error interrupt bit[159:128]	
	A01ESTAT5	ECCRAMA0_1E5	ECC 1bit Error interrupt bit[191:160]	
	A01ESTAT6	ECCRAMA0_1E6	ECC 1bit Error interrupt bit[223:192]	
	A01ESTAT7	ECCRAMA0_1E7	ECC 1bit Error interrupt bit[255:224]	
	A11ESTAT0	ECCRAMA1_1E0	ECC 1bit Error interrupt bit[31:0]	
	A11ESTAT1	ECCRAMA1_1E1	ECC 1bit Error interrupt bit[63:32]	
	A11ESTAT2	ECCRAMA1_1E2	ECC 1bit Error interrupt bit[95:64]	
	A11ESTAT3	ECCRAMA1_1E3	ECC 1bit Error interrupt bit[127:96]	
	A11ESTAT4	ECCRAMA1_1E4	ECC 1bit Error interrupt bit[159:128]	
	A11ESTAT5	ECCRAMA1_1E5	ECC 1bit Error interrupt bit[191:160]	
	A11ESTAT6	ECCRAMA1_1E6	ECC 1bit Error interrupt bit[223:192]	
	A11ESTAT7	ECCRAMA1_1E7	ECC 1bit Error interrupt bit[255:224]	
EREISR1	M01ESTAT0	ECCRAMM0_1E0	ECC 1bit Error interrupt bit[31:0]	
	M01ESTAT1	ECCRAMM0_1E1	ECC 1bit Error interrupt bit[63:32]	
	M01ESTAT2	ECCRAMM0_1E2	ECC 1bit Error interrupt bit[95:64]	
	M01ESTAT3	ECCRAMM0_1E3	ECC 1bit Error interrupt bit[127:96]	
	M01ESTAT4	ECCRAMM0_1E4	ECC 1bit Error interrupt bit[159:128]	
	M01ESTAT5	ECCRAMM0_1E5	ECC 1bit Error interrupt bit[191:160]	
	M01ESTAT6	ECCRAMM0_1E6	ECC 1bit Error interrupt bit[223:192]	
	M01ESTAT7	ECCRAMM0_1E7	ECC 1bit Error interrupt bit[255:224]	
	M11ESTAT0	ECCRAMM1_1E0	ECC 1bit Error interrupt bit[31:0]	
	M11ESTAT1	ECCRAMM1_1E1	ECC 1bit Error interrupt bit[63:32]	
	M11ESTAT2	ECCRAMM1_1E2	ECC 1bit Error interrupt bit[95:64]	
	M11ESTAT3	ECCRAMM1_1E3	ECC 1bit Error interrupt bit[127:96]	
	M11ESTAT4	ECCRAMM1_1E4	ECC 1bit Error interrupt bit[159:128]	
	M11ESTAT5	ECCRAMM1_1E5	ECC 1bit Error interrupt bit[191:160]	
	M11ESTAT6	ECCRAMM1_1E6	ECC 1bit Error interrupt bit[223:192]	
	M11ESTAT7	ECCRAMM1_1E7	ECC 1bit Error interrupt bit[255:224]	

Table 8.8 ECC 2bit Error Mapping

Register Name	Register Status Bit	Interrupt	Description	Integrated Interrupt
EREISR2	A02ESTAT0	ECCRAMA0_2E0	ECC 2bit Error interrupt bit[31:0]	EC7TIE2
	A02ESTAT1	ECCRAMA0_2E1	ECC 2bit Error interrupt bit[63:32]	
	A02ESTAT2	ECCRAMA0_2E2	ECC 2bit Error interrupt bit[95:64]	
	A02ESTAT3	ECCRAMA0_2E3	ECC 2bit Error interrupt bit[127:96]	
	A02ESTAT4	ECCRAMA0_2E4	ECC 2bit Error interrupt bit[159:128]	
	A02ESTAT5	ECCRAMA0_2E5	ECC 2bit Error interrupt bit[191:160]	
	A02ESTAT6	ECCRAMA0_2E6	ECC 2bit Error interrupt bit[223:192]	
	A02ESTAT7	ECCRAMA0_2E7	ECC 2bit Error interrupt bit[255:224]	
	A12ESTAT0	ECCRAMA1_2E0	ECC 2bit Error interrupt bit[31:0]	
	A12ESTAT1	ECCRAMA1_2E1	ECC 2bit Error interrupt bit[63:32]	
	A12ESTAT2	ECCRAMA1_2E2	ECC 2bit Error interrupt bit[95:64]	
	A12ESTAT3	ECCRAMA1_2E3	ECC 2bit Error interrupt bit[127:96]	
	A12ESTAT4	ECCRAMA1_2E4	ECC 2bit Error interrupt bit[159:128]	
	A12ESTAT5	ECCRAMA1_2E5	ECC 2bit Error interrupt bit[191:160]	
	A12ESTAT6	ECCRAMA1_2E6	ECC 2bit Error interrupt bit[223:192]	
	A12ESTAT7	ECCRAMA1_2E7	ECC 2bit Error interrupt bit[255:224]	
EREISR3	M02ESTAT0	ECCRAMM0_2E0	ECC 2bit Error interrupt bit[31:0]	
	M02ESTAT1	ECCRAMM0_2E1	ECC 2bit Error interrupt bit[63:32]	
	M02ESTAT2	ECCRAMM0_2E2	ECC 2bit Error interrupt bit[95:64]	
	M02ESTAT3	ECCRAMM0_2E3	ECC 2bit Error interrupt bit[127:96]	
	M02ESTAT4	ECCRAMM0_2E4	ECC 2bit Error interrupt bit[159:128]	
	M02ESTAT5	ECCRAMM0_2E5	ECC 2bit Error interrupt bit[191:160]	
	M02ESTAT6	ECCRAMM0_2E6	ECC 2bit Error interrupt bit[223:192]	
	M02ESTAT7	ECCRAMM0_2E7	ECC 2bit Error interrupt bit[255:224]	
	M12ESTAT0	ECCRAMM1_2E0	ECC 2bit Error interrupt bit[31:0]	
	M12ESTAT1	ECCRAMM1_2E1	ECC 2bit Error interrupt bit[63:32]	
	M12ESTAT2	ECCRAMM1_2E2	ECC 2bit Error interrupt bit[95:64]	
	M12ESTAT3	ECCRAMM1_2E3	ECC 2bit Error interrupt bit[127:96]	
	M12ESTAT4	ECCRAMM1_2E4	ECC 2bit Error interrupt bit[159:128]	
	M12ESTAT5	ECCRAMM1_2E5	ECC 2bit Error interrupt bit[191:160]	
	M12ESTAT6	ECCRAMM1_2E6	ECC 2bit Error interrupt bit[223:192]	
	M12ESTAT7	ECCRAMM1_2E7	ECC 2bit Error interrupt bit[255:224]	

Table 8.9 ECC Overflow Error Mapping

Register Name	Register Status Bit	Interrupt	Description	Integrated Interrupt
EREISR4	A0OFSTAT0	ECCRAMA0_OF0	ECC Overflow Error interrupt bit[31:0]	EC7TIOVF
	A0OFSTAT1	ECCRAMA0_OF1	ECC Overflow Error interrupt bit[63:32]	
	A0OFSTAT2	ECCRAMA0_OF2	ECC Overflow Error interrupt bit[95:64]	
	A0OFSTAT3	ECCRAMA0_OF3	ECC Overflow Error interrupt bit[127:96]	
	A0OFSTAT4	ECCRAMA0_OF4	ECC Overflow Error interrupt bit[159:128]	
	A0OFSTAT5	ECCRAMA0_OF5	ECC Overflow Error interrupt bit[191:160]	
	A0OFSTAT6	ECCRAMA0_OF6	ECC Overflow Error interrupt bit[223:192]	
	A0OFSTAT7	ECCRAMA0_OF7	ECC Overflow Error interrupt bit[255:224]	
	A1OFSTAT0	ECCRAMA1_OF0	ECC Overflow Error interrupt bit[31:0]	
	A1OFSTAT1	ECCRAMA1_OF1	ECC Overflow Error interrupt bit[63:32]	
	A1OFSTAT2	ECCRAMA1_OF2	ECC Overflow Error interrupt bit[95:64]	
	A1OFSTAT3	ECCRAMA1_OF3	ECC Overflow Error interrupt bit[127:96]	
	A1OFSTAT4	ECCRAMA1_OF4	ECC Overflow Error interrupt bit[159:128]	
	A1OFSTAT5	ECCRAMA1_OF5	ECC Overflow Error interrupt bit[191:160]	
	A1OFSTAT6	ECCRAMA1_OF6	ECC Overflow Error interrupt bit[223:192]	
	A1OFSTAT7	ECCRAMA1_OF7	ECC Overflow Error interrupt bit[255:224]	
EREISR5	M0OFSTAT0	ECCRAMM0_OF0	ECC Overflow Error interrupt bit[31:0]	
	M0OFSTAT1	ECCRAMM0_OF1	ECC Overflow Error interrupt bit[63:32]	
	M0OFSTAT2	ECCRAMM0_OF2	ECC Overflow Error interrupt bit[95:64]	
	M0OFSTAT3	ECCRAMM0_OF3	ECC Overflow Error interrupt bit[127:96]	
	M0OFSTAT4	ECCRAMM0_OF4	ECC Overflow Error interrupt bit[159:128]	
	M0OFSTAT5	ECCRAMM0_OF5	ECC Overflow Error interrupt bit[191:160]	
	M0OFSTAT6	ECCRAMM0_OF6	ECC Overflow Error interrupt bit[223:192]	
	M0OFSTAT7	ECCRAMM0_OF7	ECC Overflow Error interrupt bit[255:224]	
	M1OFSTAT0	ECCRAMM1_OF0	ECC Overflow Error interrupt bit[31:0]	
	M1OFSTAT1	ECCRAMM1_OF1	ECC Overflow Error interrupt bit[63:32]	
	M1OFSTAT2	ECCRAMM1_OF2	ECC Overflow Error interrupt bit[95:64]	
	M1OFSTAT3	ECCRAMM1_OF3	ECC Overflow Error interrupt bit[127:96]	
	M1OFSTAT4	ECCRAMM1_OF4	ECC Overflow Error interrupt bit[159:128]	
	M1OFSTAT5	ECCRAMM1_OF5	ECC Overflow Error interrupt bit[191:160]	
	M1OFSTAT6	ECCRAMM1_OF6	ECC Overflow Error interrupt bit[223:192]	
	M1OFSTAT7	ECCRAMM1_OF7	ECC Overflow Error interrupt bit[255:224]	

8.7.5 GPT/MTU3a Interrupt Selection

GPT interrupt and MTU3a Interrupt are selected by the INTPMSEL0/1 register.

Refer to the following tables.

Table 8.10 Selection of GPT Interrupt and MTU3a Interrupt by the INTPMSEL0 Register

Interrupt Source	Cause of Interrupt	Interrupt Source	Cause of Interrupt	Selection Register	SPI / IRQ No.
GPT (ch0)	CMPE0	MTU3a (ch0)	TGIA0	INTPMSEL0[0]	132
	CMPF0		TGIB0	INTPMSEL0[1]	133
	ADTRGA0		TGIC0	INTPMSEL0[2]	134
	ADTRGB0		TGID0	INTPMSEL0[3]	135
	OVF0		TCIV0	INTPMSEL0[4]	136
	UNF0		TGIE0	INTPMSEL0[5]	137
GPT (ch1)	CMPE1		TGIF0	INTPMSEL0[6]	145
	CMPF1	MTU3a (ch1)	TGIA1	INTPMSEL0[7]	146
	ADTRGA1		TGIB1	INTPMSEL0[8]	147
	ADTRGB1		TCIV1	INTPMSEL0[9]	148
	OVF1		TCIU1	INTPMSEL0[10]	149
	UNF1	MTU3a (ch2)	TGIA2	INTPMSEL0[11]	150
GPT (ch2)	CMPE2		TGIB2	INTPMSEL0[12]	158
	CMPF2		TCIV2	INTPMSEL0[13]	159
	ADTRGA2		TCIU2	INTPMSEL0[14]	160
	ADTRGB2	MTU3a (ch3)	TGIA3	INTPMSEL0[15]	161
	OVF2		TGIB3	INTPMSEL0[16]	162
	UNF2		TGIC3	INTPMSEL0[17]	163
GPT (ch3)	CMPE3		TGID3	INTPMSEL0[18]	171
	CMPF3		TCIV3	INTPMSEL0[19]	172
	ADTRGA3	MTU3a (ch4)	TGIA4	INTPMSEL0[20]	173
	ADTRGB3		TGIB4	INTPMSEL0[21]	174
	OVF3		TGIC4	INTPMSEL0[22]	175
	UNF3		TGID4	INTPMSEL0[23]	176
GPT (ch4)	CMPE4		TCIV4	INTPMSEL0[24]	184
	CMPF4	MTU3a (ch5)	TGIU5	INTPMSEL0[25]	185
	ADTRGA4		TGIV5	INTPMSEL0[26]	186
	ADTRGB4		TGIW5	INTPMSEL0[27]	187
	OVF4	MTU3a (ch6)	TGIA6	INTPMSEL0[28]	188
	UNF4		TGIB6	INTPMSEL0[29]	189
GPT (ch5)	CMPE5		TGIC6	INTPMSEL0[30]	197
	CMPF5		TGID6	INTPMSEL0[31]	198

Table 8.11 Selection of GPT Interrupt and MTU3a Interrupt by the INTPMSEL1 Register

Interrupt Source	Cause of Interrupt	Interrupt Source	Cause of Interrupt	Selection Register	SPI / IRQ No.
GPT (ch5)	ADTRGA5	MTU3a (ch6)	TCIV6	INTPMSEL1[0]	199
	ADTRGB5	MTU3a (ch7)	TGIA7	INTPMSEL1[1]	200
	OVF5		TGIB7	INTPMSEL1[2]	201
	UNF5		TGIC7	INTPMSEL1[3]	202
GPT (ch6)	CMPE6		TGID7	INTPMSEL1[4]	210
	CMPF6		TCIV7	INTPMSEL1[5]	211
	ADTRGA6	MTU3a (ch8)	TGIA8	INTPMSEL1[6]	212
	ADTRGB6		TGIB8	INTPMSEL1[7]	213
	OVF6		TGIC8	INTPMSEL1[8]	214
	UNF6		TGID8	INTPMSEL1[9]	215
GPT (ch7)	CMPE7		TCIV8	INTPMSEL1[10]	223
	CMPF7		TCIU8	INTPMSEL1[11]	224

8.8 Usage Note

8.8.1 Precaution when use the peripheral modules which can initiate DMA Controller

Some on-chip peripheral modules use the same signal both for an interrupt request and for a DMA transfer request. Refer to **Section 14.4, DMA Extension Resource Selectors 0/0S to 7/7S** for detail.

If such a module is selected by a DMARSn/nS register, the signal works as a DMA transfer request signal and interrupt requests to the interrupt controller are masked.

To enable the interrupt, clear the setting of DMARSn/nS (set all MID[7:0] and RID[1:0] to 0).
In addition, need following register settings even if DMA controller is not used.

- Set CPG_CLKON_DMAC_REG register to supply a clock for DMA Controller.
  Refer to **Section 7.2.4, Clock Control Register DMAC_REG** for register detail.
- Set CPG_RST_DMAC register to release a reset for DMA Controller.
  Refer to **Section 7.2.4, Reset Control Register DMAC** for register detail.

8.8.2 Clear Timing of Interrupt Cause

Clear the interrupt cause flag in the interrupt exception service routine. It takes time from clearing the interrupt cause flag until the interrupt cause to the CPU is actually removed. Therefore, to prevent the interrupt cause that should have been cleared from being accidentally accepted again, the interrupt cause flag is read after clearing, and then the return instruction is executed.

8.8.3 Precaution when Changing Interrupt Settings

To change the NMI, IRQ, TINT interrupt settings, mask the interrupts and change the settings.

- When changing the noise filter settings.
- When switching the GPIO pins to IRQ or GPIOINT.
- When changing the source of TINT.
- When changing the interrupt detection method.

When changing the NMI, IRQ, TINT interrupt detection method to the edge type, perform the process of clearing the interrupt status after changing the setting.

- NMI: Write 0 to the NSTAT bit of NSCR.
- IRQ: Write 0 to the ISTATn bit of ISCR.
- TINT: Write 0 to the TSTATn bit of TSCR.

9. DDR4/LPDDR4 SDRAM Memory Controller (MEMC)

DDR4/LPDDR4 SDRAM Memory Controller (MEMC) is External Bus Controller for DDR4/LPDDR4 SDRAM (DDR). This block supports up to DDR4/LPDDR4-1600 SDRAM. Interface bus width is 16 bits. In line ECC can be supported.

For setup and configuration to this block, please use and refer to the linux software package of this LSI including DDR setup codes.

9.1 Features

This unit consists of MC (Memory controller) and PHY. It supports the following specifications.

- DDR4/LPDDR4 (JEDEC STANDARD JESD79-4A/JESD209-4)

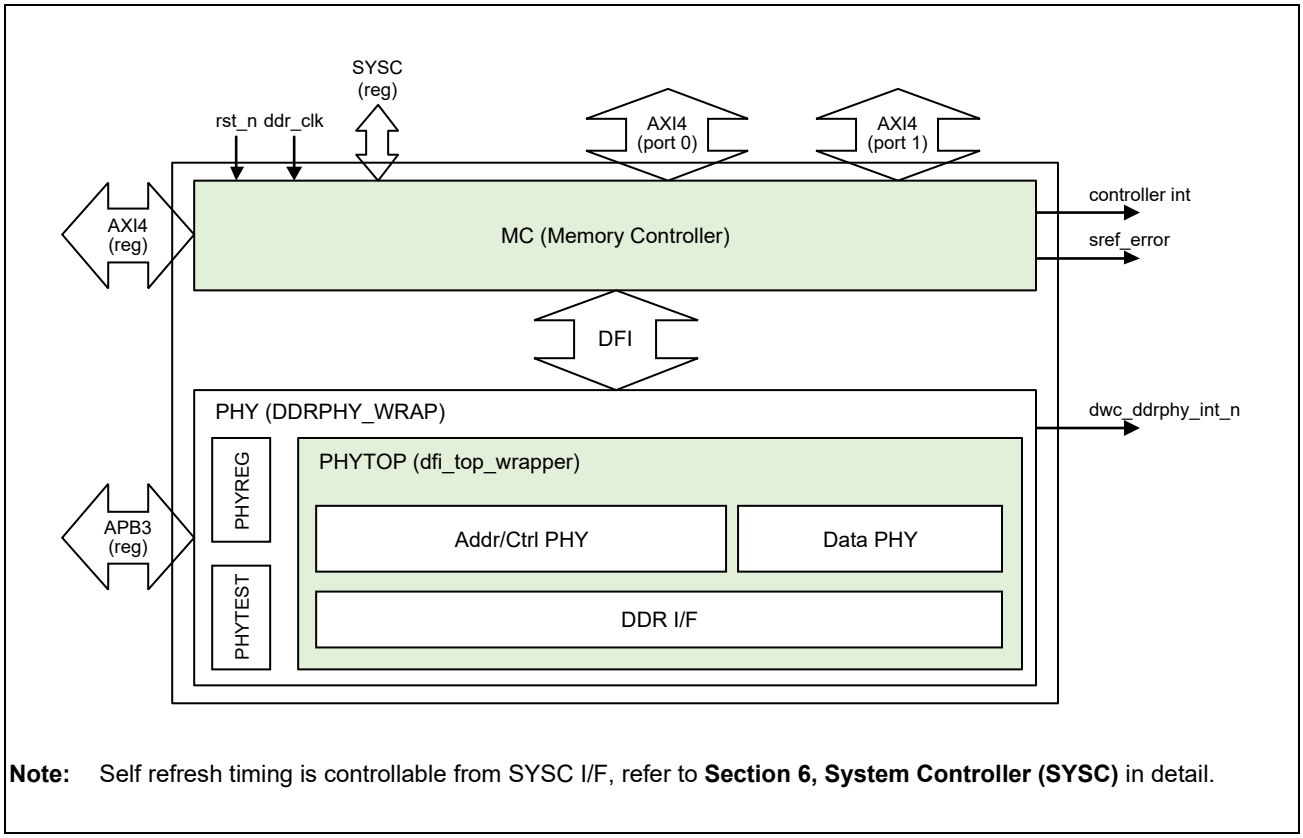
Figure 9.1 shows features of this block.

Table 9.1 DDR4/LPDDR4 SDRAM Memory Controller (MEMC) Features

Feature	Description
DRAM IF	• DDR4/LPDDR4: 1600 Mbps (800 MHz) • Width: 16 bits (Full (16 bits) or Half (8 bits) data width mode) • Rank: DDR4: 1rank, LPDDR4: 1rank, 2rank • Density: DDR4: up to 4 Gbytes (x16bit width/1pcs) • Density: LPDDR4: up to 1 Gbytes (x16bit width/1pcs)
MC (Memory controller)	• Fully pipelined command, read and write data interfaces to the controller. • Advanced bank look-ahead features for high memory throughput. • A programmable register interface to control memory parameters and protocols including auto pre-charge. • Full initialization of memory on controller reset. • ECC function for single bit and double bit error reporting, single bit error correction, and programmable removal of ECC storage.
PHY	• Low standby and DDR_VDDQ signaling power down to 1.2 V (DDR4)/1.1 V (LPDDR4) operation • PHY independent, firmware-based training using an embedded calibration processor • Support for a SW controllable DQ bit and AC bit swizzling
Low power	• Multiple low power states (Self-Refresh, Retention) • Self-Refresh Entry/Exit request • Automatic or Software interface

9.2 Block Diagram

Figure 9.1 shows the diagrams of this unit.



Note: Self refresh timing is controllable from SYSC I/F, refer to **Section 6, System Controller (SYSC)** in detail.

Figure 9.1 Block Diagram

This unit has the following restrictions.

[DDR4]

- Not support the followings;
  - Burst Chop mode
  - Interleaving
  - Gear-down Mode
  - Fine Granularity Refresh Mode except Normal mode (Fixed 1x)
  - CAL Mode
  - CRC
  - CA Parity
- ODT is only allowed in DLL on mode.

[LPDDR4]

- Not supported followings;
  - Interleaving
  - Gear-down Mode
  - Fine Granularity Refresh Mode except Normal mode (Fixed 1x)
  - CRC
  - CA Parity

9.3 Interface

9.3.1 Power, Ground

Table 9.2 shows the power and ground pins of this unit.

Table 9.2 Power, Ground

Name	I/O	Description
DDR_VDDQ	Supply	Power supply for I/O
DDR_VAA	Supply	Power Supply for PLL
VDD	Supply	Power supply for DDR Core
VSS	Ground	Ground for DDR Core

9.3.2 External Pins

Table 9.3 shows the external pins for DDR4/LPDDR4 interface.

Table 9.3 External Pins List (1/2)

Pin Name	I/O	Description	Default Function	
			DDR4	LPDDR4
BP_ZN	O	calibration external reference resistor	BP_ZN	BP_ZN
BP_VREF	IO	External Vref monitor and Analog test.	BP_VREF	BP_VREF
BP_MEMRESET#	O	DRAM reset	RESET_N	RESET_N
BP_A0	IO	DRAM Command/Address	CKE0	CKE
BP_A2	IO	DRAM Command/Address	CS0_n	CS0
BP_A3	IO	DRAM Command/Address	C0	CS1
BP_A4	IO	DRAM Command/Address	BG0	CLK_T
BP_A5	IO	DRAM Command/Address	BG1	CLK_C
BP_A6	IO	DRAM Command/Address	ACT_N	—
BP_A7	IO	DRAM Command/Address	A9	—
BP_A8	IO	DRAM Command/Address	A12	CA0
BP_A9	IO	DRAM Command/Address	A11	CA1
BP_A10	IO	DRAM Command/Address	A7	CA2
BP_A11	IO	DRAM Command/Address	A8	CA3
BP_A12	IO	DRAM Command/Address	A6	CA4
BP_A13	IO	DRAM Command/Address	A5	—
BP_A14	IO	DRAM Command/Address	A4	—
BP_A15	IO	DRAM Command/Address	A3	—
BP_A16	IO	DRAM Command/Address	CLK0_T	—
BP_A17	IO	DRAM Command/Address	CLK0_C	—
BP_A24	IO	DRAM Command/Address	A2	—
BP_A25	IO	DRAM Command/Address	A1	—
BP_A26	IO	DRAM Command/Address	BA1	—
BP_A28	IO	DRAM Command/Address	A13	—
BP_A29	IO	DRAM Command/Address	BA0	—
BP_A30	IO	DRAM Command/Address	A10	—

Table 9.3 External Pins List (2/2)

Pin Name	I/O	Description	Default Function	
			DDR4	LPDDR4
BP_A31	IO	DRAM Command/Address	A0	—
BP_A33	IO	DRAM Command/Address	CAS_N	—
BP_A34	IO	DRAM Command/Address	WE_N	—
BP_A35	IO	DRAM Command/Address	RAS_N	—
BP_A36	IO	DRAM Command/Address	ODT0	—
BP_D0	IO	DRAM Data	DQ0	DQ0
BP_D1	IO	DRAM Data	DQ1	DQ1
BP_D2	IO	DRAM Data	DQ2	DQ2
BP_D3	IO	DRAM Data	DQ3	DQ3
BP_D4	IO	DRAM Data	DQ4	DQ4
BP_D5	IO	DRAM Data	DQ5	DQ5
BP_D6	IO	DRAM Data	DQ6	DQ6
BP_D7	IO	DRAM Data	DQ7	DQ7
BP_D8	IO	DRAM Data	DM0/DBI0	DM0/DBI0
BP_D9	IO	DRAM Data	DQS_T0	DQS_T0
BP_D10	IO	DRAM Data	DQS_C0	DQS_C0
BP_D12	IO	DRAM Data	DQ8	DQ8
BP_D13	IO	DRAM Data	DQ9	DQ9
BP_D14	IO	DRAM Data	DQ10	DQ10
BP_D15	IO	DRAM Data	DQ11	DQ11
BP_D16	IO	DRAM Data	DQ12	DQ12
BP_D17	IO	DRAM Data	DQ13	DQ13
BP_D18	IO	DRAM Data	DQ14	DQ14
BP_D19	IO	DRAM Data	DQ15	DQ15
BP_D20	IO	DRAM Data	DM1/DBI1	DM1/DBI1
BP_D21	IO	DRAM Data	DQS_T1	DQS_T1
BP_D22	IO	DRAM Data	DQS_C1	DQS_C1

9.3.3 Interrupt

This unit has the interrupt pin. The interrupt of this unit is level signal.

Table 9.4 Interrupt Interface

Port Name	I/O	Description	Active Level	Level/Pulse
controller_int	O	Interrupt signal from the controller. This is a level-sensitive signal which will be asserted when the controller detects any interrupt conditions.	High	Level
sref_error	O	SREF control error.	High	Level
dwc_ddrphy_int_n	O	Interrupt signal from the DDR PHY.	Low	Level

9.4 Usage Note

9.4.1 Power On

The power on sequence for the DDR_VDDQ and VDD domains is such that either the 2 voltages should be ramped up together or VDD should be ramped up prior to DDR_VDDQ in order to prevent voltage stress in the DDR IOs.

10. On-chip RAM

This LSI has an on-chip RAM for work area. These memory units can be used to store instructions or data. The operation and write access to the on-chip RAM can be enabled or disabled through the RAM enable bit ((VCEN in SYS_RAMn_EN register) (n = 0-3) and RAM write enable bit (VLWEN in SYS_RAMn_EN register) (n = 0-3).

For details on the register, refer to the **Section 6.3, Register Descriptions** in **Section 6, System Controller (SYSC)**.

10.1 Features

■ Memory size

1 Mbyte (SRAM ACPU: 512 Kbytes (256KB × 2), SRAM MCPU: 512 Kbytes (256KB × 2))

■ Ports

On-chip RAM has one read and write port and is connected to the AXI bus.

■ Method of arbitration

When the same port of the on-chip RAM is accessed from different masters simultaneously, the AXI bus performs arbitration in round-robin mode.

■ Number of access cycles

The number of cycles for access to read or write is one cycle of SRAM_ACPU_ACLK (P1ϕ).

The number of cycles for access to read or write is one cycle of SRAM_MCPU_ACLK (P3ϕ).

11. Error Correcting Code (ECC) for On-Chip RAM

11.1 Overview

This LSI chip has four 256-Kbyte areas of on-chip RAM for use as the working area.

Each of on-chip RAM incorporates an ECC function, which allows the detection and correction of 1-bit errors and the detection of 2-bit errors. When a 1-bit or 2-bit error occurs, the address where the error occurred can be stored in an ECC error address register.

The data bus width is 256 bits. This is divided into eight channels as shown in **Figure 11.1**, with detection handled in each 32-bit channel.

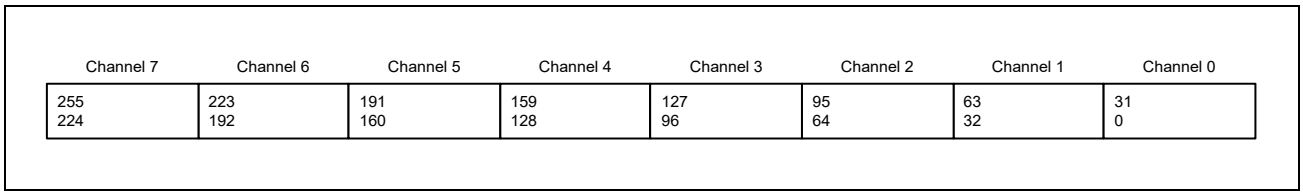


Figure 11.1 Division for ECC Processing

11.1.1 Features

- Error detection and correction
  - Detection and correction of 1-bit errors and detection of 2-bit errors are possible.
  - An error involving 3 or more bits cannot be correctly handled and may result in erroneous detection and correction.
- Control of error detection and correction
  - Error detection and correction, and correction of 1-bit errors can respectively be enabled or disabled.
- Capture of the addresses where errors occurred
  - When a 1-bit or 2-bit error is detected, the address where the error was encountered is stored in an ECC error address register. Each channel has eight ECC error address registers.
- Interrupt request
  - An interrupt request can be generated on the detection of a 1-bit error (selectable as enabled or disabled).
  - An interrupt request can be generated on the detection of a 2-bit error (selectable as enabled or disabled).
  - An overflow interrupt is generated from each channel on detection of an error when all ECC error address registers contain values.

11.2 Register Configuration

Table 11.1 and **Table 11.2** respectively show the base register and base address. The addresses of the ECC registers are represented by offsets from the base address.

Table 11.1 Base Register

Base Register Name	On-Chip RAM	Register Abbreviation
<REG_base>	SRAM ACPU0	ECCRAM0
	SRAM ACPU1	ECCRAM1
	SRAM MCPU0	ECCRAM2
	SRAM MCPU1	ECCRAM3

Table 11.2 Base Register

Base Register Name	On-Chip RAM	Register Abbreviation
<ADR_base>	SRAM ACPU0	H'0_1127_0000 (Overall Address Space) H'4127_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure) H'5127_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)
	SRAM ACPU1	H'0_1128_0000 (Overall Address Space) H'4128_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure) H'5128_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)
	SRAM MCPU0	H'0_112A_0000 (Overall Address Space) H'412A_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure) H'512A_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)
	SRAM MCPU1	H'0_112B_0000 (Overall Address Space) H'412B_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure) H'512B_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Note: Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 11.3 lists the ECC registers.

Do not attempt access to the on-chip RAM areas while the registers listed in **Table 11.3** are being set.

Table 11.3 List of the ECC Registers (1/2)

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
0	ECC control register_0	<REG_base>CTL_0	R/W	H'0000_0010	<ADR_base> + H'0000	32
	ECC error address register 0_0	<REG_base>EAD0_0	R	H'0000_0000	<ADR_base> + H'0010	32
	ECC error address register 1_0	<REG_base>EAD1_0	R	H'0000_0000	<ADR_base> + H'0014	32
	ECC error address register 2_0	<REG_base>EAD2_0	R	H'0000_0000	<ADR_base> + H'0018	32
	ECC error address register 3_0	<REG_base>EAD3_0	R	H'0000_0000	<ADR_base> + H'001C	32
	ECC error address register 4_0	<REG_base>EAD4_0	R	H'0000_0000	<ADR_base> + H'0020	32
	ECC error address register 5_0	<REG_base>EAD5_0	R	H'0000_0000	<ADR_base> + H'0024	32
	ECC error address register 6_0	<REG_base>EAD6_0	R	H'0000_0000	<ADR_base> + H'0028	32
	ECC error address register 7_0	<REG_base>EAD7_0	R	H'0000_0000	<ADR_base> + H'002C	32
1	ECC control register_1	<REG_base>CTL_1	R/W	H'0000_0010	<ADR_base> + H'0040	32
	ECC error address register 0_1	<REG_base>EAD0_1	R	H'0000_0000	<ADR_base> + H'0050	32
	ECC error address register 1_1	<REG_base>EAD1_1	R	H'0000_0000	<ADR_base> + H'0054	32
	ECC error address register 2_1	<REG_base>EAD2_1	R	H'0000_0000	<ADR_base> + H'0058	32
	ECC error address register 3_1	<REG_base>EAD3_1	R	H'0000_0000	<ADR_base> + H'005C	32
	ECC error address register 4_1	<REG_base>EAD4_1	R	H'0000_0000	<ADR_base> + H'0060	32
	ECC error address register 5_1	<REG_base>EAD5_1	R	H'0000_0000	<ADR_base> + H'0064	32
	ECC error address register 6_1	<REG_base>EAD6_1	R	H'0000_0000	<ADR_base> + H'0068	32
	ECC error address register 7_1	<REG_base>EAD7_1	R	H'0000_0000	<ADR_base> + H'006C	32
2	ECC control register_2	<REG_base>CTL_2	R/W	H'0000_0010	<ADR_base> + H'0080	32
	ECC error address register 0_2	<REG_base>EAD0_2	R	H'0000_0000	<ADR_base> + H'0090	32
	ECC error address register 1_2	<REG_base>EAD1_2	R	H'0000_0000	<ADR_base> + H'0094	32
	ECC error address register 2_2	<REG_base>EAD2_2	R	H'0000_0000	<ADR_base> + H'0098	32
	ECC error address register 3_2	<REG_base>EAD3_2	R	H'0000_0000	<ADR_base> + H'009C	32
	ECC error address register 4_2	<REG_base>EAD4_2	R	H'0000_0000	<ADR_base> + H'00A0	32
	ECC error address register 5_2	<REG_base>EAD5_2	R	H'0000_0000	<ADR_base> + H'00A4	32
	ECC error address register 6_2	<REG_base>EAD6_2	R	H'0000_0000	<ADR_base> + H'00A8	32
	ECC error address register 7_2	<REG_base>EAD7_2	R	H'0000_0000	<ADR_base> + H'00AC	32
3	ECC control register_3	<REG_base>CTL_3	R/W	H'0000_0010	<ADR_base> + H'00C0	32
	ECC error address register 0_3	<REG_base>EAD0_3	R	H'0000_0000	<ADR_base> + H'00D0	32
	ECC error address register 1_3	<REG_base>EAD1_3	R	H'0000_0000	<ADR_base> + H'00D4	32
	ECC error address register 2_3	<REG_base>EAD2_3	R	H'0000_0000	<ADR_base> + H'00D8	32
	ECC error address register 3_3	<REG_base>EAD3_3	R	H'0000_0000	<ADR_base> + H'00DC	32
	ECC error address register 4_3	<REG_base>EAD4_3	R	H'0000_0000	<ADR_base> + H'00E0	32
	ECC error address register 5_3	<REG_base>EAD5_3	R	H'0000_0000	<ADR_base> + H'00E4	32
	ECC error address register 6_3	<REG_base>EAD6_3	R	H'0000_0000	<ADR_base> + H'00E8	32
	ECC error address register 7_3	<REG_base>EAD7_3	R	H'0000_0000	<ADR_base> + H'00EC	32

Table 11.3 List of the ECC Registers (2/2)

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
4	ECC control register_4	<REG_base>CTL_4	R/W	H'0000_0010	<ADR_base> + H'0100	32
	ECC error address register 0_4	<REG_base>EAD0_4	R	H'0000_0000	<ADR_base> + H'0110	32
	ECC error address register 1_4	<REG_base>EAD1_4	R	H'0000_0000	<ADR_base> + H'0114	32
	ECC error address register 2_4	<REG_base>EAD2_4	R	H'0000_0000	<ADR_base> + H'0118	32
	ECC error address register 3_4	<REG_base>EAD3_4	R	H'0000_0000	<ADR_base> + H'011C	32
	ECC error address register 4_4	<REG_base>EAD4_4	R	H'0000_0000	<ADR_base> + H'0120	32
	ECC error address register 5_4	<REG_base>EAD5_4	R	H'0000_0000	<ADR_base> + H'0124	32
	ECC error address register 6_4	<REG_base>EAD6_4	R	H'0000_0000	<ADR_base> + H'0128	32
	ECC error address register 7_4	<REG_base>EAD7_4	R	H'0000_0000	<ADR_base> + H'012C	32
5	ECC control register_5	<REG_base>CTL_5	R/W	H'0000_0010	<ADR_base> + H'0140	32
	ECC error address register 0_5	<REG_base>EAD0_5	R	H'0000_0000	<ADR_base> + H'0150	32
	ECC error address register 1_5	<REG_base>EAD1_5	R	H'0000_0000	<ADR_base> + H'0154	32
	ECC error address register 2_5	<REG_base>EAD2_5	R	H'0000_0000	<ADR_base> + H'0158	32
	ECC error address register 3_5	<REG_base>EAD3_5	R	H'0000_0000	<ADR_base> + H'015C	32
	ECC error address register 4_5	<REG_base>EAD4_5	R	H'0000_0000	<ADR_base> + H'0160	32
	ECC error address register 5_5	<REG_base>EAD5_5	R	H'0000_0000	<ADR_base> + H'0164	32
	ECC error address register 6_5	<REG_base>EAD6_5	R	H'0000_0000	<ADR_base> + H'0168	32
	ECC error address register 7_5	<REG_base>EAD7_5	R	H'0000_0000	<ADR_base> + H'016C	32
6	ECC control register_6	<REG_base>CTL_6	R/W	H'0000_0010	<ADR_base> + H'0180	32
	ECC error address register 0_6	<REG_base>EAD0_6	R	H'0000_0000	<ADR_base> + H'0190	32
	ECC error address register 1_6	<REG_base>EAD1_6	R	H'0000_0000	<ADR_base> + H'0194	32
	ECC error address register 2_6	<REG_base>EAD2_6	R	H'0000_0000	<ADR_base> + H'0198	32
	ECC error address register 3_6	<REG_base>EAD3_6	R	H'0000_0000	<ADR_base> + H'019C	32
	ECC error address register 4_6	<REG_base>EAD4_6	R	H'0000_0000	<ADR_base> + H'01A0	32
	ECC error address register 5_6	<REG_base>EAD5_6	R	H'0000_0000	<ADR_base> + H'01A4	32
	ECC error address register 6_6	<REG_base>EAD6_6	R	H'0000_0000	<ADR_base> + H'01A8	32
	ECC error address register 7_6	<REG_base>EAD7_6	R	H'0000_0000	<ADR_base> + H'01AC	32
7	ECC control register_7	<REG_base>CTL_7	R/W	H'0000_0010	<ADR_base> + H'01C0	32
	ECC error address register 0_7	<REG_base>EAD0_7	R	H'0000_0000	<ADR_base> + H'01D0	32
	ECC error address register 1_7	<REG_base>EAD1_7	R	H'0000_0000	<ADR_base> + H'01D4	32
	ECC error address register 2_7	<REG_base>EAD2_7	R	H'0000_0000	<ADR_base> + H'01D8	32
	ECC error address register 3_7	<REG_base>EAD3_7	R	H'0000_0000	<ADR_base> + H'01DC	32
	ECC error address register 4_7	<REG_base>EAD4_7	R	H'0000_0000	<ADR_base> + H'01E0	32
	ECC error address register 5_7	<REG_base>EAD5_7	R	H'0000_0000	<ADR_base> + H'01E4	32
	ECC error address register 6_7	<REG_base>EAD6_7	R	H'0000_0000	<ADR_base> + H'01E8	32
	ECC error address register 7_7	<REG_base>EAD7_7	R	H'0000_0000	<ADR_base> + H'01EC	32

11.3 Register Descriptions

11.3.1 ECC Control Registers_n (<REG_base>CTL_n) (n = 0 to 7)

Each ECC control register_n is used to control the ECC function in the corresponding channel n shown in **Figure 11.1**.

After writing the initial value to the entire usage area of the on-chip RAM, set the given ECERVF bits to 1 to enable error detection. Be sure to write 01b to the corresponding EMCA[1:0] bits to enable writing to an ECERVF bit.

For writing the initial value to the on-chip RAM areas, refer to **Section 11.5, Initializing the Detection-Usage Areas of the On-Chip RAM**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECDEDF7	ECSEDF7	ECDEDF6	ECSEDF6	ECDEDF5	ECSEDF5	ECDEDF4	ECSEDF4	ECDEDF3	ECSEDF3	ECDEDF2	ECSEDF2	ECDEDF1	ECSEDF1	ECDEDF0	ECSEDF0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EMCA[1:0]		—	—	ECOVFF	ECER2C	ECER1C	—	—	ECERVF	EC1ECP	EC2EDIC	EC1EDIC	ECER2F	ECER1F	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R	R	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	ECDEDF7	0	R	ECC 2-Bit Error Detection Flags ECDEDFm (m = 0 to 7) These bits indicate the occurrence of 2-bit errors in the data output from the RAM while error detection is enabled. The addresses of the errors will be stored in the <REG_base>EADm_n registers.
30	ECSEDF7	0	R	
29	ECDEDF6	0	R	
28	ECSEDF6	0	R	
27	ECDEDF5	0	R	ECC 1-Bit Error Detection Flags ECSEDFm (m = 0 to 7) These bits indicate the occurrence of 1-bit errors in the data output from the RAM while error detection is enabled. The addresses of the errors will be stored in the <REG_base>EADm_n registers.
26	ECSEDF5	0	R	
25	ECDEDF4	0	R	
24	ECSEDF4	0	R	
23	ECDEDF3	0	R	The ECDEDFm and ECSEDFm bits are cleared to 0 under any of the following conditions. 1. Power-on reset 2. Writing 1 to the corresponding ECER2C bit 3. Setting the ECC error detection enable bit to “disabled” (ECERVF = 0) 4. Reset by WDT overflow
22	ECSEDF3	0	R	
21	ECDEDF2	0	R	
20	ECSEDF2	0	R	
19	ECDEDF1	0	R	
18	ECSEDF1	0	R	
17	ECDEDF0	0	R	
16	ECSEDF0	0	R	
15, 14	EMCA[1:0]	00b	R/W	Enable Writing to ECERVF 01b: Enabled Others: Disabled These bits are always read as 00b.
13, 12	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

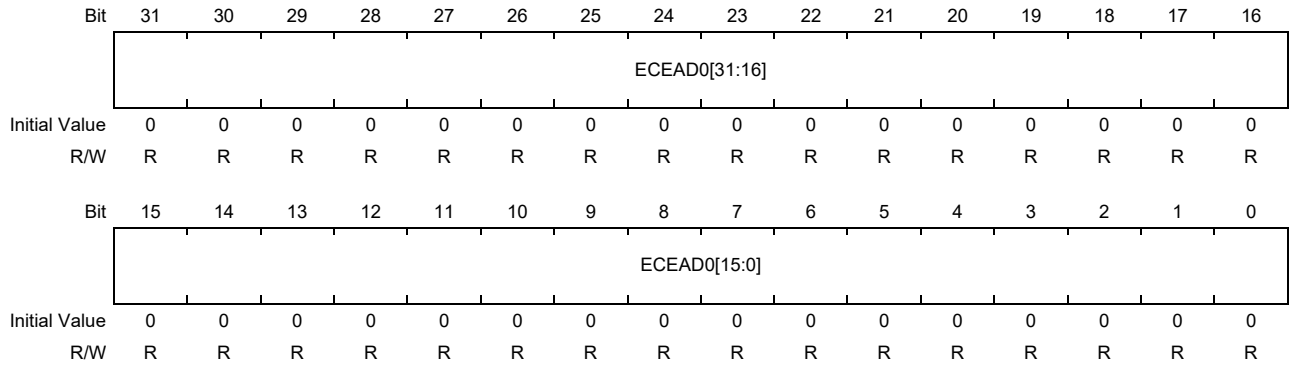
Bit	Bit Name	Initial Value	R/W	Description
11	ECOVFF	0	R	ECC Overflow Detection Flag This bit is set to 1 and an overflow interrupt is output on detection of an error when the ECC error address registers (<REG_base>EADm_n (m = 0 to 7)) all contain data. An overflow interrupt is output even if the error is detected while the setting of this bit is already 1. This bit is cleared to 0 under any of the following conditions. 1. Power-on reset 2. Writing 1 to the corresponding ECER2C bit or ECER1C bit 3. Setting the ECC error detection enable bit to “disabled” (ECERVF = 0)
10	ECER2C	0	R/W	2-Bit Error Detection Flag Clear 0: The internal state does not change. 1: The corresponding 2-bit error detection flag ECER2F is cleared. This bit is always read as 0.
9	ECER1C	0	R/W	1-Bit Error Detection Flag Clear 0: The internal state does not change. 1: The corresponding 1-bit error detection flag ECER1F is cleared. This bit is always read as 0.
8, 7	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
6	ECERVF	0	R/W	ECC Error Detection Enable 0: Disables error detection. 1: Enables error detection. 1-bit error correction (EC1ECP) and interrupt control (EC2EDIC and EC1EDIC) are only possible when error detection is enabled. *Note*: To write to this bit, the corresponding EMCA[1:0] bits must be set to 01b.
5	EC1ECP	0	R/W	1-Bit Error Correction Enable 0: The error is corrected when a 1-bit error is detected. 1: The error is not corrected when a 1-bit error is detected.
4	EC2EDIC	1	R/W	2-Bit Error Detection Interrupt Control 0: An interrupt is not generated when a 2-bit error is detected. 1: An interrupt is generated when a 2-bit error is detected. The interrupt is only generated when the corresponding 2-bit error detection flag (ECER2F) changes from 0 to 1.
3	EC1EDIC	0	R/W	1-Bit Error Detection Interrupt Control 0: An interrupt is not generated when a 1-bit error is detected. 1: An interrupt is generated when a 1-bit error is detected. The interrupt is only generated when the corresponding 1-bit error detection flag (ECER1F) changes from 0 to 1.
2	ECER2F	0	R	2-Bit Error Detection Flag 0: A 2-bit error has not occurred. 1: A 2-bit error has occurred. This bit is cleared to 0 under any of the following conditions. 1. Power-on reset 2. Writing 1 to the corresponding ECER2C bit 3. Setting the ECC error detection enable bit to "disabled" (ECERVF = 0)
1	ECER1F	0	R	1-Bit Error Detection Flag 0: A 1-bit error has not occurred. 1: A 1-bit error has occurred. This bit is cleared to 0 under any of the following conditions. 1. Power-on reset 2. Writing 1 to the corresponding ECER1C bit 3. Setting the ECC error detection enable bit to "disabled" (ECERVF = 0)

Bit	Bit Name	Initial Value	R/W	Description
0	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

11.3.2 ECC Error Address Registers m_n (<REG_base>EADm_n) (m = 0 to 7) (n = 0 to 7)

When a 1- or 2-bit error occurs, up to eight error addresses are sequentially stored in the registers from <REG_base>EAD0_n to <REG_base>EAD7_n.

When a 2-bit error is detected while only 1-bit error addresses were stored up to <REG_base>EAD7_n, <REG_base>EAD7_n is overwritten by the address of the 2-bit error. Otherwise, it will not be overwritten. For details, refer to **Section 11.6.4, ECC Error Address Register Overwrite Conditions**.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ECEAD0 [31:0]	All 0	R	These bits hold the address of the (m + 1)th error to have been detected. These bits are cleared to 0 under the following condition: clearing ECER2F or ECER1F in the <REG_base>CTL_n register.

11.4 Initializing the ECC Function

Initialize the ECC by following the procedure below to enable the ECC function.

1. Set the VECCEN bit in the ECCRAM setting register corresponding to the RAM area for which the ECC function is to be enabled to 1.
2. Write the initial value to the entire usage area of the RAM for which the ECC function is to be enabled.
3. Set the ECERVF bits in the ECC control registers_n to 1 to enable error detection. Then, set the respective bits to control 1-bit error correction and interrupts.

For notes on the settings of registers of this module, refer to **Section 11.6.1, Notes on Setting the Registers of This Module**.

11.5 Initializing the Detection-Usage Areas of the On-Chip RAM

Before using the ECC function, write the initial value to the entire usage area of the on-chip RAM for which the ECC function is to be enabled. The initial value can be any value. Since the detection of ECC errors is handled in 32-bit units in each channel, the initial value should also be written in 32-bit units.

Enabling error detection without initialization may cause an ECC error due to unintended reading.

Examples of Operations that May Lead to Errors

1. When writing is to proceed in 8- or 16-bit units, reading the 32-bit value from the given address and then writing the data
2. If the locations are not initialized, an ECC error may occur when the 32-bit value is read. For details on writing in 8- or 16-bit units, refer to **Section 11.6.2, 8- or 16-Bit Access**.
3. When the CPU cache is enabled, if a cache miss occurs in writing, the cache being refilled by a 1-line unit
   At this time if data are not initialized, an ECC error may occur because data are read in 1-line (32-byte) units.

11.6 Usage Notes

11.6.1 Notes on Setting the Registers of This Module

Ensure that none of the bus masters proceeds with access to the on-chip RAM areas while the registers listed in **Table 11.3** are being set up.

11.6.2 8- or 16-Bit Access

The ECC function for the on-chip RAM areas handles error detection in 32-bit units. Accordingly, when access is to be in 8- or 16-bit units, the operations proceed as follows.

(1) 8- or 16-bit reading

(a) The 32 bits of data are read and the ECC is used to determine whether or not an error is present.

(b) The 8 or 16 bits of data are then returned to the bus master that issued the read request.

(2) 8- or 16-bit writing

(a) The 32 bits of data from the address that contains the location for writing are read and the ECC is used to determine whether or not an error is present.

(b) The 32 bits of data that have been read are then overwritten with the 8- or 16-bit data.

(c) The ECC value for the overwritten 32-bit data in (b) is added by writing it to the RAM.

11.6.3 Numbers of Cycles for Reading from or Writing to a RAM Area with the ECC Function Enabled

The numbers of cycles for reading from or writing to an on-chip RAM area with the ECC function enabled are as follows.

(1) Reading

The number of cycles for reading in 8- or 16-bit and 32-bit units when the ECC function is enabled is one cycle of SRAM_ACPU_ACLK or SRAM_MCPU_ACLK longer than when the ECC function is disabled.

(2) Writing

The number of cycles for writing in 8- or 16-bit units when the ECC function is enabled is two cycles of SRAM_ACPU_ACLK or SRAM_MCPU_ACLK longer than when the ECC function is disabled. The number of cycles for writing in 32-bit units when the ECC function is enabled is the same as when the ECC function is disabled.

11.6.4 ECC Error Address Register Overwrite Conditions

If a new ECC error occurs with the ECC error address stored up to <REG_base> EAD7_n, <REG_base> EAD7_n may or may not be overwritten. The conditions to be overwritten are shown below.

Error Address Storage Status Up to <REG_base> EAD7_n	New ECC Error	<REG_base>EAD7_n Overwrite	Overflow Interrupt
Only 1-bit error	1-bit error	Not overwritten	Occur
Only 1-bit error	2-bit error	Overwritten	Occur
Only 2-bit error	1-bit error	Not overwritten	Occur
Only 2-bit error	2-bit error	Not overwritten	Occur
1-bit error and 2-bit error	1-bit error	Not overwritten	Occur
1-bit error and 2-bit error	2-bit error	Not overwritten	Occur

12. Message Handling Unit (MHU)

MHU is a function for message communication between Cortex-A55(CA55) cores and Cortex-M33(CM33/CM33_FPU). Message communication is done by shared RAM (On-chip RAM) for passing message and response between CPUs and the function (MHU) for notifying when messages and responses are stored in the memory.

12.1 Features

Table 12.1 shows MHU feature summary.

Table 12.1 Feature Summary

Feature	Description
Interrupt by Message and response	Generate interrupt to notify each Cortex-A55, Cortex-M33_FPU or Cortex-M33 when the message/response is stored in shared RAM (On-chip RAM).
Software interrupt control	Control registers for interrupt setting, clearing, and status confirmation
Event routing function	Event routing function is available between Cortex-A55, Cortex-M33_FPU and Cortex-M33. The Cluster that sends and receives event signals has both EVENTI / EVNETO interface, and the router responds to the EVENTI handshake and initiates the EVENTO handshake with other than the module that sent the event.

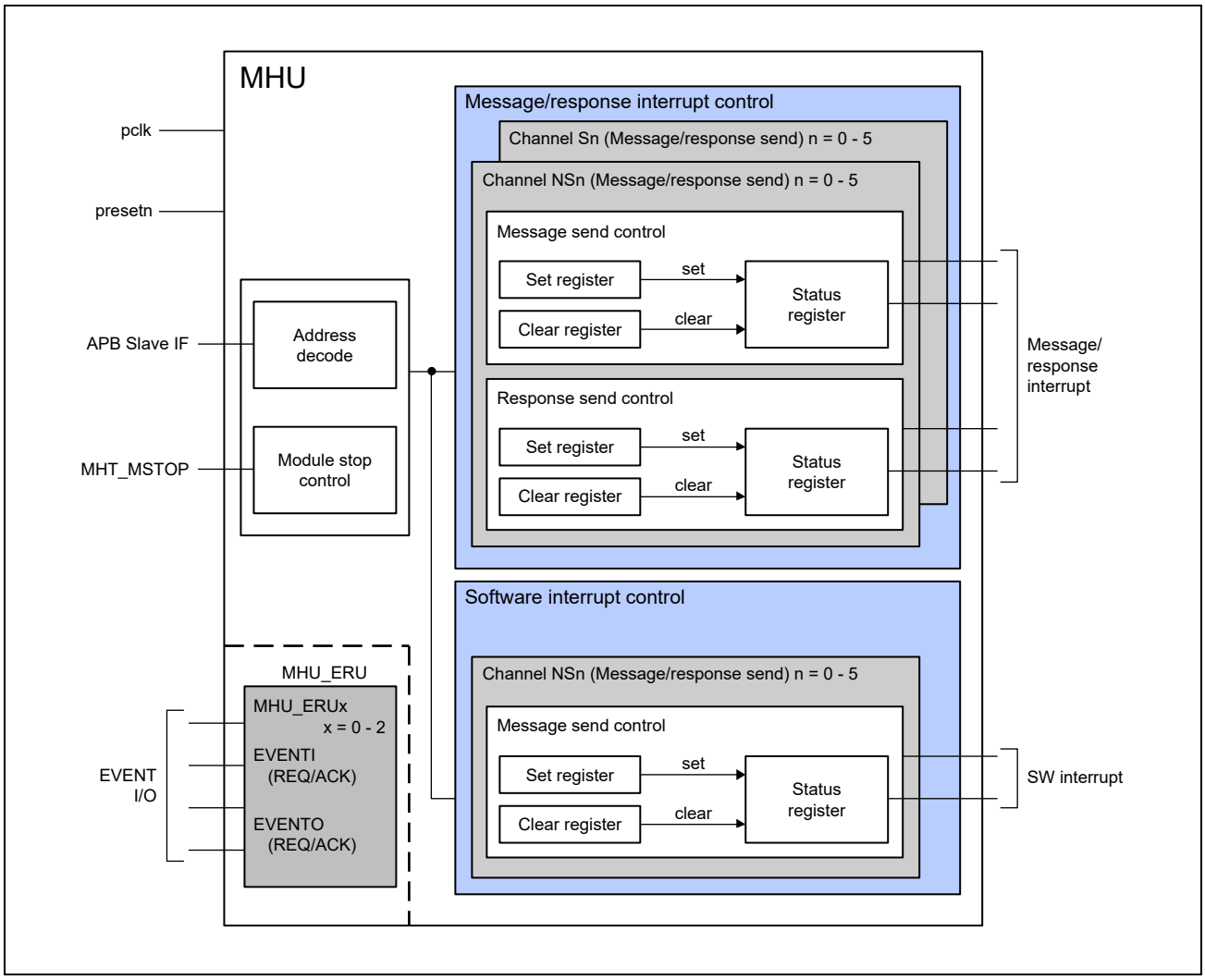


Figure 12.1 MHU Block Diagram

Message/response interrupt control registers

Registers that control message/response interrupts.

The message/response interrupt control register controls interrupts with the set register, clear register, and status register. An interrupt is asserted by writing “1” to the set register. The interrupt is negated by writing “1” to the clear register. The status can be checked by reading the status register. With these as one set, the message transmission processing register

One channel is composed of a pair of data transmission processing register and response transmission processing register, and a total of 12 channels are mounted. The breakdown of 12 channels is as follows.

Shows MHU feature summary.

Non-secure interrupt control register (Channel NS0 to 5):

Channel NS0 (CA55: Message transmission/CM33_FPU: Response transmission)

Channel NS1 (CA55: Message transmission/CM33: Response transmission)

Channel NS2 (CM33_FPU: Message transmission/CA55: Response transmission)

Channel NS3 (CM33_FPU: Message transmission/CM33: Response transmission)

Channel NS4 (CM33: Message transmission/CA55: Response transmission)

Channel NS5 (CM33: Message transmission/CM33_FPU: Response transmission)

Secure interrupt control register (Channel S0 to 5):

Channel S0 (CA55: Message transmission/CM33_FPU: Response transmission)

Channel S1 (CA55: Message transmission/CM33 : Response transmission)

Channel S2 (CM33_FPU: Message transmission/CA55: Response transmission)

Channel S3 (CM33_FPU: Message transmission/CM33: Response transmission)

Channel S4 (CM33: Message transmission/CA55: Response transmission)

Channel S5 (CM33: Message transmission/CM33_FPU: Response transmission)

Software interrupt control register

Registers that control software interrupts.

The software interrupt control register controls interrupts using the set register, clear register, and status register. An interrupt is asserted by writing “1” to the set register. The interrupt is negated by writing “1” to the clear register. You can also check the interrupt status by reading the status register. With these as one set, the message transmission processing register and response.

One channel is composed of a pair of transmission processing registers, and a total of 6 channels are mounted.

Module stop control

This circuit refers to the module stop control (MHU_MSTOP) in CPG_MHU_MSTOP register on Clock Pulse Generator (CPG) and returns an error response if an access occurs while the module is stopped.

Event routing control circuit

Figure 12.2 figure shows event routing control circuit. In detail, refer to **Section 12.5.2, Event Routing Function**.

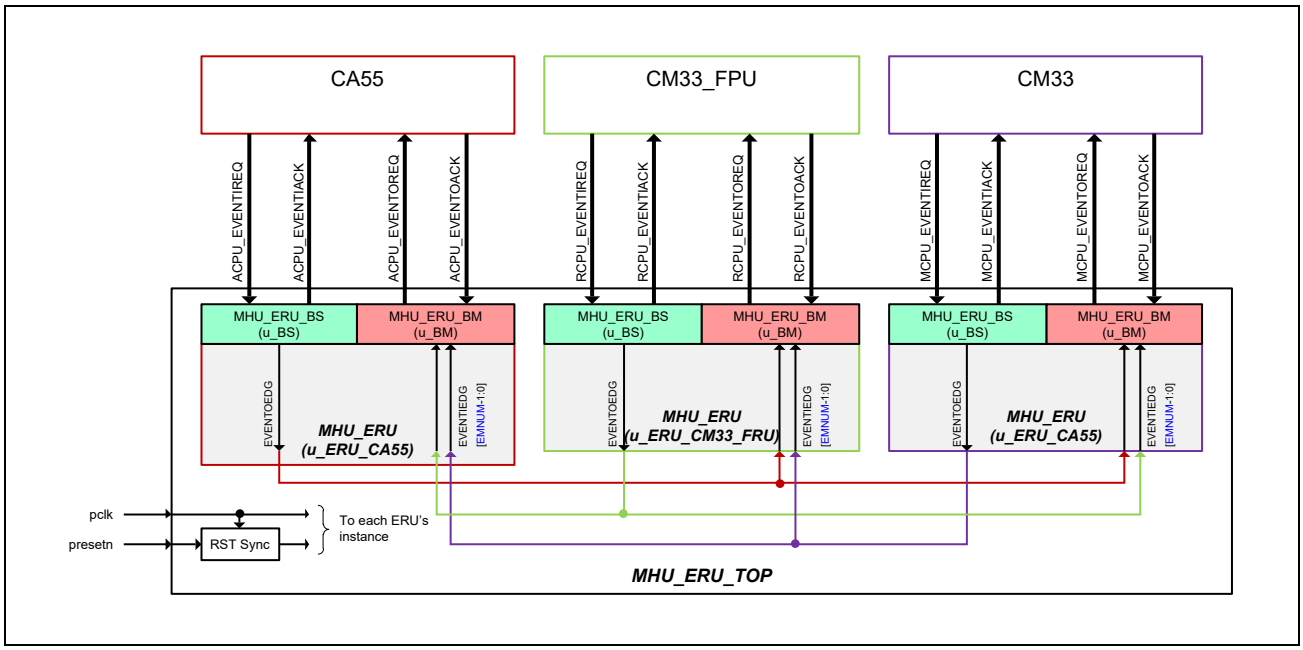


Figure 12.2 Event Routing Unit Circuit Structure

Table 12.2 shows Event Routing functional terminals.

Table 12.2 Event Routing Terminal

Terminal Name	Direction	Function	Pulse/Level	Active Level
CA55_EVENTOREQ	OUT	Event output request for CA55	Pulse	High
CA55_EVENTOACK	IN	Event output acknowledge for CA55	Level	High
CA55_EVENTIREQ	IN	Event input request for CA55	Level	High
CA55_EVENTIACK	OUT	Event input acknowledge for CA55	Pulse	High
CM33_EVENTOREQ	OUT	Event output request for CM33	Pulse	High
CM33_EVENTOACK	IN	Event output acknowledge for CM33	Level	High
CM33_EVENTIREQ	IN	Event input request for CM33	Level	High
CM33_EVENTIACK	OUT	Event input acknowledge for CM33	Pulse	High
CM33_FPU_EVENTOREQ	OUT	Event output request for CM33_FPU	Pulse	High
CM33_FPU_EVENTOACK	IN	Event output acknowledge for CM33_FPU	Level	High
CM33_FPU_EVENTIREQ	IN	Event input request for CM33_FPU	Level	High
CM33_FPU_EVENTIACK	OUT	Event input acknowledge for CM33_FPU	Pulse	High

12.2 Interrupt

Table 12.3 shows interrupt output.

Table 12.3 Interrupt Type

Output	Description	Pulse/Level	Active Level
msg_ch0_ns	Non-secure message transmission interrupt (CA55 -> CM33_FPU)	Level	HIGH
rsp_ch0_ns	Non-secure response transmission interrupt (CM33_FPU -> CA55)	Level	HIGH
msg_ch1_ns	Non-secure message transmission interrupt (CA55 -> CM33)	Level	HIGH
rsp_ch1_ns	Non-secure response transmission interrupt (CM33 -> CA55)	Level	HIGH
msg_ch2_ns	Non-secure message transmission interrupt (CM33_FPU -> CA55)	Level	HIGH
rsp_ch2_ns	Non-secure response transmission interrupt (CA55 -> CM33_FPU)	Level	HIGH
msg_ch3_ns	Non-secure message transmission interrupt (CM33_FPU -> CM33)	Level	HIGH
rsp_ch3_ns	Non-secure response transmission interrupt (CM33 -> CM33_FPU)	Level	HIGH
msg_ch4_ns	Non-secure message transmission interrupt (CM33 -> CA55)	Level	HIGH
rsp_ch4_ns	Non-secure response transmission interrupt (CA55 -> CM33)	Level	HIGH
msg_ch5_ns	Non-secure message transmission interrupt (CM33 -> CM33_FPU)	Level	HIGH
rsp_ch5_ns	Non-secure response transmission interrupt (CM33_FPU -> CM33)	Level	HIGH
msg_ch0_s	Non-secure message transmission interrupt (CA55 -> CM33_FPU)	Level	HIGH
rsp_ch0_s	Non-secure response transmission interrupt (CM33_FPU -> CA55)	Level	HIGH
msg_ch1_s	Secure message transmission interrupt (CA55 -> CM33)	Level	HIGH
rsp_ch1_s	Secure response transmission interrupt (CM33 -> CA55)	Level	HIGH
msg_ch2_s	Non-secure message transmission interrupt (CM33_FPU -> CA55)	Level	HIGH
rsp_ch2_s	Non-secure response transmission interrupt (CA55 -> CM33_FPU)	Level	HIGH
msg_ch3_s	Non-secure message transmission interrupt (CM33_FPU -> CM33)	Level	HIGH
rsp_ch3_s	Non-secure response transmission interrupt (CM33 -> CM33_FPU)	Level	HIGH
msg_ch4_s	Secure message transmission interrupt (CM33 -> CA55)	Level	HIGH
rsp_ch4_s	Secure response transmission interrupt (CA55 -> CM33)	Level	HIGH
msg_ch5_s	Non-secure message transmission interrupt (CM33 -> CM33_FPU)	Level	HIGH
rsp_ch5_s	Non-secure response transmission interrupt (CM33_FPU -> CM33)	Level	HIGH
swint_ch0_ns	software interrupt 0 (CA55 -> CM33)	Level	HIGH
swint_ch1_ns	software interrupt 1 (CM33_FPU -> CM33)	Level	HIGH
swint_ch2_ns	software interrupt 2 (CM33 -> CA55)	Level	HIGH
swint_ch3_ns	software interrupt 3 (CM33 -> CM33_FPU)	Level	HIGH
swint_ch4_ns	software interrupt 4 (CM33_FPU -> CA55)	Level	HIGH
swint_ch5_ns	software interrupt 5 (CA55 -> CM33_FPU)	Level	HIGH

12.3 Register Configuration

Table 12.4 shows address mapping for MHU. MHU has 4KB address space.

Table 12.4 MHU Address Map

Offset Address (CA55 MHU Base address: H'0_1040_0000) (CM33/CM33_FPU Secure MHU Base address: H'4040_0000) (CM33/CM33_FPU Non-secure MHU Base address: H'5040_0000)	Description
H'0000 – H'07FF	Non-secure message/response interrupt control register
H'0800 – H'0FFF	Non-secure software interrupt control register
H'1000 – H'17FF	Secure message/response interrupt control register
H'1800 – H'1FFF	Reserved
H'2000 – H'FFFF	Reserved

Note: Prohibit to access Reserved area.

Note: Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 12.5 shows register attributes

Table 12.5 Register Attributes

Attribute	Description
RW	Read&Write register
RW0	Read&Write0 register. Invalid Write 1.
RW1	Read&Write1 register. Invalid Write 0.
R0W	Read0&Write register. Always Read 0.
R1W	Read1&Write register. Always Read 1.
R	Read Only
R0W0	Read0&Write0 register. Always Read 0. Invalid Write 1.
R0W1	Read0&Write1 register. Always Read 0. Invalid Write 0.
R1W0	Read1&Write0 register. Always Read 1. Invalid Write 1.
R1W1	Read1&Write1 register. Always Read 1. Invalid Write 0.
RCW0	Read by clear. Write0 register. Invalid Write 1.
RCW1	Read by clear. Write1 register. Invalid Write 0.

Note: No guaranteed function for access to Reserved area, Debug and undefined area.

Table 12.6 shows registers

Table 12.6 List of Registers (1/4)

Offset Address	Register Name	Abbreviation	Initial value	Access Size
Non-secure message/response interrupt				
Non-secure message transmission interrupt register (CA55 -> CM33_FPU) channel 0				
H'0000	message transmission interrupt Status register	MSG_INT_STS0_NS	H'0000_0000	32 bits
H'0004	message transmission interrupt Set register	MSG_INT_SET0_NS	H'0000_0000	32 bits
H'0008	message transmission interrupt Clear register	MSG_INT_CLR0_NS	H'0000_0000	32 bits
H'000C	Reserved	—	—	—
Non-secure response transmission interrupt register (CM33_FPU -> CA55) channel 0				
H'0010	response transmission interrupt Status register	RSP_INT_STS0_NS	H'0000_0000	32 bits
H'0014	response transmission interrupt Set register	RSP_INT_SET0_NS	H'0000_0000	32 bits
H'0018	response transmission interrupt Clear register	RSP_INT_CLR0_NS	H'0000_0000	32 bits
H'001C	Reserved	—	—	—
Non-secure message transmission interrupt register (CA55 -> CM33) channel 1				
H'0020	message transmission interrupt Status register	MSG_INT_STS1_NS	H'0000_0000	32 bits
H'0024	message transmission interrupt Set register	MSG_INT_SET1_NS	H'0000_0000	32 bits
H'0028	message transmission interrupt Clear register	MSG_INT_CLR1_NS	H'0000_0000	32 bits
H'002C	Reserved	—	—	—
Non-secure response transmission interrupt register (CM33 -> CA55) channel 1				
H'0030	response transmission interrupt Status register	RSP_INT_STS1_NS	H'0000_0000	32 bits
H'0034	response transmission interrupt Set register	RSP_INT_SET1_NS	H'0000_0000	32 bits
H'0038	response transmission interrupt Clear register	RSP_INT_CLR1_NS	H'0000_0000	32 bits
H'003C	Reserved	—	—	—
Non-secure message transmission interrupt register (CM33_FPU -> CA55) channel 2				
H'0040	message transmission interrupt Status register	MSG_INT_STS2_NS	H'0000_0000	32 bits
H'0044	message transmission interrupt Set register	MSG_INT_SET2_NS	H'0000_0000	32 bits
H'0048	message transmission interrupt Clear register	MSG_INT_CLR2_NS	H'0000_0000	32 bits
H'004C	Reserved	—	—	—
Non-secure response transmission interrupt register (CA55 -> CM33_FPU) channel 2				
H'0050	response transmission interrupt Status register	RSP_INT_STS2_NS	H'0000_0000	32 bits
H'0054	response transmission interrupt Set register	RSP_INT_SET2_NS	H'0000_0000	32 bits
H'0058	response transmission interrupt Clear register	RSP_INT_CLR2_NS	H'0000_0000	32 bits
H'005C	Reserved	—	—	—
Non-secure message transmission interrupt register (CM33_FPU -> CM33) channel 3				
H'0060	message transmission interrupt Status register	MSG_INT_STS3_NS	H'0000_0000	32 bits
H'0064	message transmission interrupt Set register	MSG_INT_SET3_NS	H'0000_0000	32 bits
H'0068	message transmission interrupt Clear register	MSG_INT_CLR3_NS	H'0000_0000	32 bits
H'006C	Reserved	—	—	—
Non-secure response transmission interrupt register (CM33 -> CM33_FPU) channel 3				
H'0070	response transmission interrupt Status register	RSP_INT_STS3_NS	H'0000_0000	32 bits
H'0074	response transmission interrupt Set register	RSP_INT_SET3_NS	H'0000_0000	32 bits
H'0078	response transmission interrupt Clear register	RSP_INT_CLR3_NS	H'0000_0000	32 bits
H'007C	Reserved	—	—	—

Table 12.6 List of Registers (2/4)

Offset Address	Register Name	Abbreviation	Initial value	Access Size
Non-secure message transmission interrupt register (CM33 -> CA55) channel 4				
H'0080	message transmission interrupt Status register	MSG_INT_STS4_NS	H'0000_0000	32 bits
H'0084	message transmission interrupt Set register	MSG_INT_SET4_NS	H'0000_0000	32 bits
H'0088	message transmission interrupt Clear register	MSG_INT_CLR4_NS	H'0000_0000	32 bits
H'008C	Reserved	—	—	—
Non-secure response transmission interrupt register (CA55 -> CM33) channel 4				
H'0090	response transmission interrupt Status register	RSP_INT_STS4_NS	H'0000_0000	32 bits
H'0094	response transmission interrupt Set register	RSP_INT_SET4_NS	H'0000_0000	32 bits
H'0098	response transmission interrupt Clear register	RSP_INT_CLR4_NS	H'0000_0000	32 bits
H'009C	Reserved	—	—	—
Non-secure message transmission interrupt register (CM33 -> CM33_FPU) channel 5				
H'00A0	message transmission interrupt Status register	MSG_INT_STS5_NS	H'0000_0000	32 bits
H'00A4	message transmission interrupt Set register	MSG_INT_SET5_NS	H'0000_0000	32 bits
H'00A8	message transmission interrupt Clear register	MSG_INT_CLR5_NS	H'0000_0000	32 bits
H'00AC	Reserved	—	—	—
Non-secure response transmission interrupt register (CM33_FPU -> CM33) channel 5				
H'00B0	response transmission interrupt Status register	RSP_INT_STS5_NS	H'0000_0000	32 bits
H'00B4	response transmission interrupt Set register	RSP_INT_SET5_NS	H'0000_0000	32 bits
H'00B8	response transmission interrupt Clear register	RSP_INT_CLR5_NS	H'0000_0000	32 bits
H'00BC	Reserved	—	—	—
Non-secure software interrupt				
Non-secure software interrupt register channel 0				
H'0800	software interrupt Status register	SW_INT_STS0_NS	H'0000_0000	32 bits
H'0804	software interrupt Set register	SW_INT_SET0_NS	H'0000_0000	32 bits
H'0808	software interrupt Clear register	SW_INT_CLR0_NS	H'0000_0000	32 bits
H'080C	Reserved	—	—	—
Non-secure software interrupt register channel 1				
H'0810	software interrupt Status register	SW_INT_STS1_NS	H'0000_0000	32 bits
H'0814	software interrupt Set register	SW_INT_SET1_NS	H'0000_0000	32 bits
H'0818	software interrupt Clear register	SW_INT_CLR1_NS	H'0000_0000	32 bits
H'081C	Reserved	—	—	—
Non-secure software interrupt register channel 2				
H'0820	software interrupt Status register	SW_INT_STS2_NS	H'0000_0000	32 bits
H'0824	software interrupt Set register	SW_INT_SET2_NS	H'0000_0000	32 bits
H'0828	software interrupt Clear register	SW_INT_CLR2_NS	H'0000_0000	32 bits
H'082C	Reserved	—	—	—
Non-secure software interrupt register channel 3				
H'0830	software interrupt Status register	SW_INT_STS3_NS	H'0000_0000	32 bits
H'0834	software interrupt Set register	SW_INT_SET3_NS	H'0000_0000	32 bits
H'0838	software interrupt Clear register	SW_INT_CLR3_NS	H'0000_0000	32 bits
H'083C	Reserved	—	—	—
Non-secure software interrupt register channel 4				
H'0840	software interrupt Status register	SW_INT_STS4_NS	H'0000_0000	32 bits
H'0844	software interrupt Set register	SW_INT_SET4_NS	H'0000_0000	32 bits
H'0848	software interrupt Clear register	SW_INT_CLR4_NS	H'0000_0000	32 bits

Table 12.6 List of Registers (3/4)

Offset Address	Register Name	Abbreviation	Initial value	Access Size
H'084C	Reserved	—	—	—
Non-secure software interrupt register channel 5				
H'0850	software interrupt Status register	SW_INT_STS5_NS	H'0000_0000	32 bits
H'0854	software interrupt Set register	SW_INT_SET5_NS	H'0000_0000	32 bits
H'0858	software interrupt Clear register	SW_INT_CLR5_NS	H'0000_0000	32 bits
H'085C	Reserved	—	—	—
Secure message/response interrupt				
Secure message transmission interrupt register (CA55 -> CM33_FPU) channel 0				
H'1000	message transmission interrupt Status register	MSG_INT_STS0_S	H'0000_0000	32 bits
H'1004	message transmission interrupt Set register	MSG_INT_SET0_S	H'0000_0000	32 bits
H'1008	message transmission interrupt Clear register	MSG_INT_CLR0_S	H'0000_0000	32 bits
H'100C	Reserved	—	—	—
Secure response transmission interrupt register (CM33_FPU -> CA55) channel 0				
H'1010	response transmission interrupt Status register	RSP_INT_STS0_S	H'0000_0000	32 bits
H'1014	response transmission interrupt Set register	RSP_INT_SET0_S	H'0000_0000	32 bits
H'1018	response transmission interrupt Clear register	RSP_INT_CLR0_S	H'0000_0000	32 bits
H'101C	Reserved	—	—	—
Secure message transmission interrupt register (CA55 -> CM33) channel 1				
H'1020	message transmission interrupt Status register	MSG_INT_STS1_S	H'0000_0000	32 bits
H'1024	message transmission interrupt Set register	MSG_INT_SET1_S	H'0000_0000	32 bits
H'1028	message transmission interrupt Clear register	MSG_INT_CLR1_S	H'0000_0000	32 bits
H'102C	Reserved	—	—	—
Secure response transmission interrupt register (CM33 -> CA55) channel 1				
H'1030	response transmission interrupt Status register	RSP_INT_STS1_S	H'0000_0000	32 bits
H'1034	response transmission interrupt Set register	RSP_INT_SET1_S	H'0000_0000	32 bits
H'1038	response transmission interrupt Clear register	RSP_INT_CLR1_S	H'0000_0000	32 bits
H'103C	Reserved	—	—	—
Secure message transmission interrupt register (CM33_FPU -> CA55) channel 2				
H'1040	message transmission interrupt Status register	MSG_INT_STS2_S	H'0000_0000	32 bits
H'1044	message transmission interrupt Set register	MSG_INT_SET2_S	H'0000_0000	32 bits
H'1048	message transmission interrupt Clear register	MSG_INT_CLR2_S	H'0000_0000	32 bits
H'104C	Reserved	—	—	—
Secure message transmission interrupt register (CA55 -> CM33_FPU) channel 2				
H'1050	response transmission interrupt Status register	RSP_INT_STS2_S	H'0000_0000	32 bits
H'1054	response transmission interrupt Set register	RSP_INT_SET2_S	H'0000_0000	32 bits
H'1058	response transmission interrupt Clear register	RSP_INT_CLR2_S	H'0000_0000	32 bits
H'105C	Reserved	—	—	—
Secure message transmission interrupt register (CM33_FPU -> CM33) channel 3				
H'1060	message transmission interrupt Status register	MSG_INT_STS3_S	H'0000_0000	32 bits
H'1064	message transmission interrupt Set register	MSG_INT_SET3_S	H'0000_0000	32 bits
H'1068	message transmission interrupt Clear register	MSG_INT_CLR3_S	H'0000_0000	32 bits
H'106C	Reserved	—	—	—
Secure message transmission interrupt register (CM33 -> CM33_FPU) channel 3				
H'1070	response transmission interrupt Status register	RSP_INT_STS3_S	H'0000_0000	32 bits
H'1074	response transmission interrupt Set register	RSP_INT_SET3_S	H'0000_0000	32 bits

Table 12.6 List of Registers (4/4)

Offset Address	Register Name	Abbreviation	Initial value	Access Size
H'1078	response transmission interrupt Clear register	RSP_INT_CLR3_S	H'0000_0000	32 bits
H'107C	Reserved	—	—	—
Secure message transmission interrupt register (CM33 -> CA55) channel 4				
H'1080	message transmission interrupt Status register	MSG_INT_STS4_S	H'0000_0000	32 bits
H'1084	message transmission interrupt Set register	MSG_INT_SET4_S	H'0000_0000	32 bits
H'1088	message transmission interrupt Clear register	MSG_INT_CLR4_S	H'0000_0000	32 bits
H'108C	Reserved	—	—	—
Secure response transmission interrupt register (CA55 -> CM33) channel 4				
H'1090	response transmission interrupt Status register	RSP_INT_STS4_S	H'0000_0000	32 bits
H'1094	response transmission interrupt Set register	RSP_INT_SET4_S	H'0000_0000	32 bits
H'1098	response transmission interrupt Clear register	RSP_INT_CLR4_S	H'0000_0000	32 bits
H'109C	Reserved	—	—	—
Secure message transmission interrupt register (CM33 -> CM33_FPU) channel 5				
H'10A0	message transmission interrupt Status register	MSG_INT_STS5_S	H'0000_0000	32 bits
H'10A4	message transmission interrupt Set register	MSG_INT_SET5_S	H'0000_0000	32 bits
H'10A8	message transmission interrupt Clear register	MSG_INT_CLR5_S	H'0000_0000	32 bits
H'10AC	Reserved	—	—	—
Secure response transmission interrupt register (CM33_FPU -> CM33) channel 5				
H'10B0	response transmission interrupt Status register	RSP_INT_STS5_S	H'0000_0000	32 bits
H'10B4	response transmission interrupt Set register	RSP_INT_SET5_S	H'0000_0000	32 bits
H'10B8	response transmission interrupt Clear register	RSP_INT_CLR5_S	H'0000_0000	32 bits
H'10BC	Reserved	—	—	—
H'10C0 ~ H'FFFF	Reserved			

Note: Access size must be 32bit. Prohibit to access to Reserved register.

12.4 Register Descriptions

12.4.1 Non-Secure Message Transmission Interrupt Status Register (CA55 -> CM33_FPU)

12.4.2 Non-Secure Message Transmission Interrupt Status Register (CA55 -> CM33)

12.4.3 Non-Secure Message Transmission Interrupt Status Register (CM33_FPU -> CA55)

12.4.4 Non-Secure Message Transmission Interrupt Status Register (CM33_FPU -> CM33)

12.4.5 Non-Secure Message Transmission Interrupt Status Register (CM33 -> CA55)

12.4.6 Non-Secure Message Transmission Interrupt Status Register (CM33 -> CM33_FPU)

Status register for Non-secure message transmission interrupt (Name: MSG_INT_STSn_NS <n=0 to 5>)

Access Size: 32 bits

Address(es): MSG_INT_STS0_NS: H'000
MSG_INT_STS1_NS: H'020
MSG_INT_STS2_NS: H'040
MSG_INT_STS3_NS: H'060
MSG_INT_STS4_NS: H'080
MSG_INT_STS5_NS: H'0A0

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STAT	0	R	Show status of message transmission interrupt. Read only. Write is invalid. 0: No message transmission interrupt (No interrupt being notified to the interrupt controller at the message destination) 1: Assert message transmission interrupt (There is an interrupt being notified to the interrupt controller at the message destination.)

12.4.7 Non-Secure Message Transmission Interrupt Set Register (CA55 -> CM33_FPU)

12.4.8 Non-Secure Message Transmission Interrupt Set Register (CA55 -> CM33)

12.4.9 Non-Secure Message Transmission Interrupt Set Register (CM33_FPU -> CA55)

12.4.10 Non-Secure Message Transmission Interrupt Set Register (CM33_FPU -> CM33)

12.4.11 Non-Secure Message Transmission Interrupt Set Register (CM33 -> CA55)

12.4.12 Non-Secure Message Transmission Interrupt Set Register (CM33 -> CM33_FPU)

Set register for Non-secure message transmission interrupt (Name: MSG_INT_SETn_NS <n=0 to 5>)

Access Size: 32 bits

Address(es): MSG_INT_SET0_NS: H'004
MSG_INT_SET1_NS: H'024
MSG_INT_SET2_NS: H'044
MSG_INT_SET3_NS: H'064
MSG_INT_SET4_NS: H'084
MSG_INT_SET5_NS: H'0A4

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SET	0	R0W1	Set message transmission interrupt. Write "1" only. Always read "0". 0: invalid 1: Issue interrupt to interrupt controller of the message destination.

12.4.13 Non-Secure Message Transmission Interrupt Clear Register (CA55 -> CM33_FPU)

12.4.14 Non-Secure Message Transmission Interrupt Clear Register (CA55 -> CM33)

12.4.15 Non-Secure Message Transmission Interrupt Clear Register (CM33_FPU -> CA55)

12.4.16 Non-Secure Message Transmission Interrupt Clear Register (CM33_FPU -> CM33)

12.4.17 Non-Secure Message Transmission Interrupt Clear Register (CM33 -> CA55)

12.4.18 Non-Secure Message Transmission Interrupt Clear Register (CM33 -> CM33_FPU)

Clear register of Non-secure message transmission interrupt (Name: MSG_INT_CLRn_NS <n=0 to 5>)

Access Size: 32 bits

Address(es): MSG_INT_CLR0_NS: H'008
MSG_INT_CLR1_NS: H'028
MSG_INT_CLR2_NS: H'048
MSG_INT_CLR3_NS: H'068
MSG_INT_CLR4_NS: H'088
MSG_INT_CLR5_NS: H'0A8

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLEAR	0	R0W1	Clear message transmission interrupt Write "1" only. Always read "0". 0: invalid 1: Clear interrupt to interrupt controller of the message destination.

12.4.19 Non-Secure Response Transmission Interrupt Status Register (CM33_FPU -> CA55)

12.4.20 Non-Secure Response Transmission Interrupt Status Register (CM33 -> CA55)

12.4.21 Non-Secure Response Transmission Interrupt Status Register (CA55 -> CM33_FPU)

12.4.22 Non-Secure Response Transmission Interrupt Status Register (CM33 -> CM33_FPU)

12.4.23 Non-Secure Response Transmission Interrupt Status Register (CA55 -> CM33)

12.4.24 Non-Secure Response Transmission Interrupt Status Register (CM33_FPU -> CM33)

Status register for Non-secure response transmission interrupt (Name: RSP_INT_STSn_NS <n=0 to 5>)

Access Size: 32 bits

Address(es): RSP_INT_STS0_NS: H'010
RSP_INT_STS1_NS: H'030
RSP_INT_STS2_NS: H'050
RSP_INT_STS3_NS: H'070
RSP_INT_STS4_NS: H'090
RSP_INT_STS5_NS: H'0B0

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STAT	0	R	Show status of response transmission interrupt. Read only. Write is invalid. 0: No response transmission interrupt (No interrupt being notified to the interrupt controller at the message destination) 1: Assert response transmission interrupt (There is an interrupt being notified to the interrupt controller at the message destination.)

12.4.25 Non-Secure Response Transmission Interrupt Set Register (CM33_FPU -> CA55)

12.4.26 Non-Secure Response Transmission Interrupt Set Register (CM33 -> CA55)

12.4.27 Non-Secure Response Transmission Interrupt Set Register (CA55 -> CM33_FPU)

12.4.28 Non-Secure Response Transmission Interrupt Set Register (CM33 -> CM33_FPU)

12.4.29 Non-Secure Response Transmission Interrupt Set Register (CA55 -> CM33)

12.4.30 Non-Secure Response Transmission Interrupt Set Register (CM33_FPU -> CM33)

Set register of Non-secure response transmission interrupt (Name: RSP_INT_SETn_NS <n=0 to 5>)

Access Size: 32 bits

Address(es): RSP_INT_SET0_NS: H'014
RSP_INT_SET1_NS: H'034
RSP_INT_SET2_NS: H'054
RSP_INT_SET3_NS: H'074
RSP_INT_SET4_NS: H'094
RSP_INT_SET5_NS: H'0B4

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SET	0	R0W1	Set response transmission interrupt. Write "1" only. Always read "0". 0: invalid 1: Issue interrupt to interrupt controller of the message destination.

12.4.31 Non-Secure Response Transmission Interrupt Clear Register (CM33_FPU -> CA55)

12.4.32 Non-Secure Response Transmission Interrupt Clear Register (CM33 -> CA55)

12.4.33 Non-Secure Response Transmission Interrupt Clear Register (CA55 -> CM33_FPU)

12.4.34 Non-Secure Response Transmission Interrupt Clear Register (CM33 -> CM33_FPU)

12.4.35 Non-Secure Response Transmission Interrupt Clear Register (CA55 -> CM33)

12.4.36 Non-Secure Response Transmission Interrupt Clear Register (CM33_FPU -> CM33)

Clear register of Non-secure response transmission interrupt (Name: RSP_INT_CLRn_NS <n=0 to 5>)

Access Size: 32bit

Address(es): RSP_INT_CLR0_NS: H'018
RSP_INT_CLR1_NS: H'038
RSP_INT_CLR2_NS: H'058
RSP_INT_CLR3_NS: H'078
RSP_INT_CLR4_NS: H'098
RSP_INT_CLR5_NS: H'0B8

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLEAR	0	R0W1	Clear response transmission interrupt Write "1" only. Always read "0". 0: invalid 1: Clear interrupt to interrupt controller of the message destination.

12.4.37 Non-Secure Software Interrupt Status Register (ch0)

12.4.38 Non-Secure Software Interrupt Status Register (ch1)

12.4.39 Non-Secure Software Interrupt Status Register (ch2)

12.4.40 Non-Secure Software Interrupt Status Register (ch3)

12.4.41 Non-Secure Software Interrupt Status Register (ch4)

12.4.42 Non-Secure Software Interrupt Status Register (ch5)

Status register of Non-secure software interrupt Status register (Name: SW_INT_STSn_NS <n=0 to 5>)

Access Size: 32 bits

Address(es): SW_INT_STS0_NS: H'0800
SW_INT_STS1_NS: H'0810
SW_INT_STS2_NS: H'0820
SW_INT_STS3_NS: H'0830
SW_INT_STS4_NS: H'0840
SW_INT_STS5_NS: H'0850

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STAT	0	R	Show status of Non-secure software interrupt. Write is invalid. 0: No interrupt (No interrupt being notified to the interrupt controller) 1: Software interrupt is asserted. (There is an interrupt being notified to the interrupt controller)

12.4.43 Non-Secure Software Interrupt Set Register (ch0)

12.4.44 Non-Secure Software Interrupt Set Register (ch1)

12.4.45 Non-Secure Software Interrupt Set Register (ch2)

12.4.46 Non-Secure Software Interrupt Set Register (ch3)

12.4.47 Non-Secure Software Interrupt Set Register (ch4)

12.4.48 Non-Secure Software Interrupt Set Register (ch5)

Set register of Non-secure software interrupt (Name: SW_INT_SETn_NS <n=0 to 5>)

Access Size: 32 bits

Address(es): SW_INT_SET0_NS: H'0804
SW_INT_SET1_NS: H'0814
SW_INT_SET2_NS: H'0824
SW_INT_SET3_NS: H'0834
SW_INT_SET4_NS: H'0844
SW_INT_SET5_NS: H'0854

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SET	0	R0W1	Set Non-secure software interrupt. Write "1" only. Always read "0". 0: invalid 1:Issue interrupt to interrupt controller

12.4.49 Non-Secure Software Interrupt Clear Register (ch0)

12.4.50 Non-Secure Software Interrupt Clear Register (ch1)

12.4.51 Non-Secure Software Interrupt Clear Register (ch2)

12.4.52 Non-Secure Software Interrupt Clear Register (ch3)

12.4.53 Non-Secure Software Interrupt Clear Register (ch4)

12.4.54 Non-Secure Software Interrupt Clear Register (ch5)

Clear register of Non-secure software interrupt Clear register(Name: SW_INT_CLRn_NS <n=0 to 5>)

Access Size: 32 bits

Address(es): SW_INT_CLR0_NS: H'0808
SW_INT_CLR1_NS: H'0818
SW_INT_CLR2_NS: H'0828
SW_INT_CLR3_NS: H'0838
SW_INT_CLR4_NS: H'0848
SW_INT_CLR5_NS: H'0858

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLEAR	0	R0W1	Clear Non-secure software interrupt. Write "1" only. Always read "0". 0: invalid 1:Clear interrupt to interrupt controller.

12.4.55 Secure Message Transmission Interrupt Status Register (CA55 -> CM33_FPU)

12.4.56 Secure Message Transmission Interrupt Status Register (CA55 -> CM33)

12.4.57 Secure Message Transmission Interrupt Status Register (CM33_FPU -> CA55)

12.4.58 Secure Message Transmission Interrupt Status Register (CM33_FPU -> CM33)

12.4.59 Secure Message Transmission Interrupt Status Register (CM33 -> CA55)

12.4.60 Secure Message Transmission Interrupt Status Register (CM33 -> CM33_FPU)

Status register of Secure message transmission interrupt (Name: MSG_INT_STSn_S <n=0 to 5>)

Access Size: 32 bits

Address(es): MSG_INT_STS0_S: H'1000
MSG_INT_STS1_S: H'1020
MSG_INT_STS2_S: H'1040
MSG_INT_STS3_S: H'1060
MSG_INT_STS4_S: H'1080
MSG_INT_STS5_S: H'10A0

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STAT	0	R	Show status of message transmission interrupt. Write is invalid. 0: No Message transmission interrupt (No interrupt being notified to the interrupt controller of the message destination) 1: message transmission interrupt is asserted. (There is an interrupt being notified to the interrupt controller of the message destination)

12.4.61 Secure Message Transmission Interrupt Set Register (CA55 -> CM33_FPU)

12.4.62 Secure Message Transmission Interrupt Set Register (CA55 -> CM33)

12.4.63 Secure Message Transmission Interrupt Set Register (CM33_FPU -> CA55)

12.4.64 Secure Message Transmission Interrupt Set Register (CM33_FPU -> CM33)

12.4.65 Secure Message Transmission Interrupt Set Register (CM33 -> CA55)

12.4.66 Secure Message Transmission Interrupt Set Register (CM33 -> CM33_FPU)

Set register of Secure message transmission interrupt (Name: MSG_INT_SETn_S <n=0 to 5>)

Access Size: 32 bits

Address(es): MSG_INT_SET0_S: H'1004
MSG_INT_SET1_S: H'1024
MSG_INT_SET2_S: H'1044
MSG_INT_SET3_S: H'1064
MSG_INT_SET4_S: H'1084
MSG_INT_SET5_S: H'10A4

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SET	0	R0W1	Set message transmission interrupt Write "1" only. Always read "0". 0: invalid 1: Issue interrupt to interrupt controller of the message destination.

12.4.67 Secure Message Transmission Interrupt Clear Register (CA55 -> CM33_FPU)

12.4.68 Secure Message Transmission Interrupt Clear Register (CA55 -> CM33)

12.4.69 Secure Message Transmission Interrupt Clear Register (CM33_FPU -> CA55)

12.4.70 Secure Message Transmission Interrupt Clear Register (CM33_FPU -> CM33)

12.4.71 Secure Message Transmission Interrupt Clear Register (CM33 -> CA55)

12.4.72 Secure Message Transmission Interrupt Clear Register (CM33 -> CM33_FPU)

Clear register of Secure message transmission interrupt (Name: MSG_INT_CLRn_S <n=0 to 5>)

Access Size: 32 bits

Address(es): MSG_INT_CLR0_S: H'1008
MSG_INT_CLR1_S: H'1028
MSG_INT_CLR2_S: H'1048
MSG_INT_CLR3_S: H'1068
MSG_INT_CLR4_S: H'1088
MSG_INT_CLR5_S: H'10A8

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLEAR	0	R0W1	Clear message transmission interrupt. Write "1" only. Always read "0". 0: invalid 1: Clear interrupt to interrupt controller of the message destination.

12.4.73 Secure Response Transmission Interrupt Status Register (CM33_FPU -> CA55)

12.4.74 Secure Response Transmission Interrupt Status Register (CM33 -> CA55)

12.4.75 Secure Response Transmission Interrupt Status Register (CA55 -> CM33_FPU)

12.4.76 Secure Response Transmission Interrupt Status Register (CM33 -> CM33_FPU)

12.4.77 Secure Response Transmission Interrupt Status Register (CA55 -> CM33)

12.4.78 Secure Response Transmission Interrupt Status Register (CM33_FPU -> CM33)

Status register of Secure response transmission interrupt (Name: RSP_INT_STSn_S <n=0 to 5>)

Access Size: 32 bits

Address(es): RSP_INT_STS0_S: H'1010
RSP_INT_STS1_S: H'1030
RSP_INT_STS2_S: H'1050
RSP_INT_STS3_S: H'1070
RSP_INT_STS4_S: H'1090
RSP_INT_STS5_S: H'10B0

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STAT	0	R	Show status of response transmission interrupt. Write is invalid. 0: No response transmission interrupt (No interrupt being notified to the interrupt controller of the message destination.) 1: response transmission interrupt is asserted. (There is an interrupt being notified to the interrupt controller of the message destination.)

12.4.79 Secure Response Transmission Interrupt Set Register (CM33_FPU -> CA55)

12.4.80 Secure Response Transmission Interrupt Set Register (CM33 -> CA55)

12.4.81 Secure Response Transmission Interrupt Set Register (CA55 -> CM33_FPU)

12.4.82 Secure Response Transmission Interrupt Set Register (CM33 -> CM33_FPU)

12.4.83 Secure Response Transmission Interrupt Set Register (CA55 -> CM33)

12.4.84 Secure Response Transmission Interrupt Set Register (CM33_FPU -> CM33)

Set register of Secure response transmission interrupt (Name: RSP_INT_SETn_NS <n=0 to 5>)

Access Size: 32 bits

Address(es): RSP_INT_SET0_S: H'1014
RSP_INT_SET1_S: H'1034
RSP_INT_SET2_S: H'1054
RSP_INT_SET3_S: H'1074
RSP_INT_SET4_S: H'1094
RSP_INT_SET5_S: H'10B4

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SET	0	R0W1	Set response transmission interrupt Write "1" only. Always read "0". 0: invalid 1: Issue interrupt to interrupt controller of the message destination.

12.4.85 Secure Response Transmission Interrupt Clear Register (CM33_FPU -> CA55)

12.4.86 Secure Response Transmission Interrupt Clear Register (CM33 -> CA55)

12.4.87 Secure Response Transmission Interrupt Clear Register (CA55 -> CM33_FPU)

12.4.88 Secure Response Transmission Interrupt Clear Register (CM33 -> CM33_FPU)

12.4.89 Secure Response Transmission Interrupt Clear Register (CA55 -> CM33)

12.4.90 Secure Response Transmission Interrupt Clear Register (CM33_FPU -> CM33)

Clear register of Secure response transmission interrupt Clear register (Name: RSP_INT_CLRn_S <n=0 to 5>)

Access Size: 32 bits

Address(es): RSP_INT_CLR0_S: H'1018
RSP_INT_CLR1_S: H'1038
RSP_INT_CLR2_S: H'1058
RSP_INT_CLR3_S: H'1078
RSP_INT_CLR4_S: H'1098
RSP_INT_CLR5_S: H'10B8

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLEAR	0	R0W1	Clear response transmission interrupt Write "1" only. Always read "0". 0: invalid 1: Clear interrupt to interrupt controller of the message destination.

12.5 Function Description

12.5.1 Inter CPU Communication

The hardware configuration is shown in **Figure 12.3**. Inter-CPU communication is established by sending a message from the message source to the message destination and sending a response from the message destination to the message source. The MHU uses an interrupt to notify that the message / response to the CPU has been written to Shared RAM(On-chip RAM). Interrupts are controlled by the Set register, Clear register, and Status register. Interrupt can be asserted by writing “1” to the Set register. Interrupt is negated by writing “1” to the Clear register, and check the interrupt status by reading the Status register. With these registers as one set, one channel is composed of a message transmission processing / response transmission processing pair. A total of 12 channels are installed so that bidirectional communication can be performed between Cortex-A55, Cortex-M33_FPU and Cortex-M33. **Table 12.7** shows the channel and message source / destination correspondence.

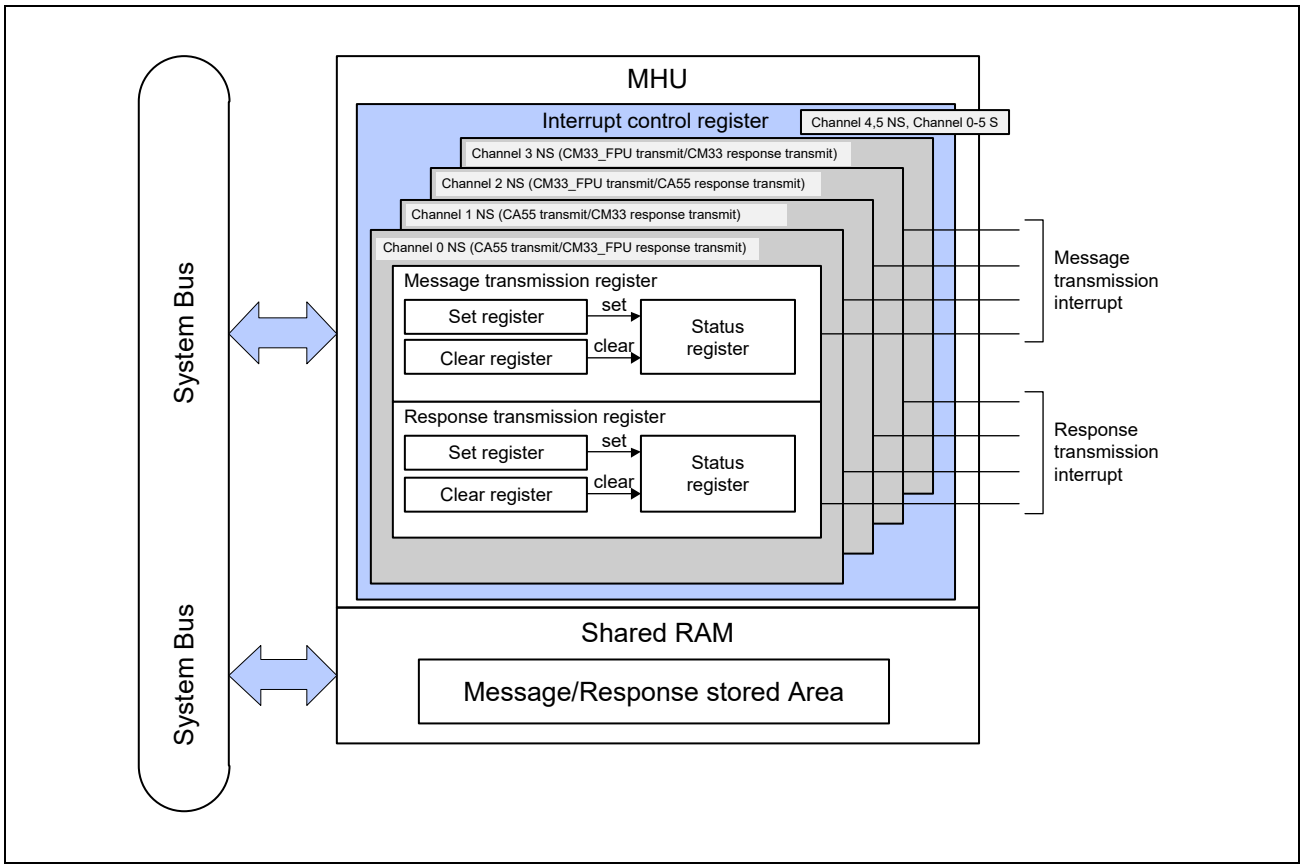


Figure 12.3 Hardware Configuration

Table 12.7 Channel and Message Destination/Source Reference

Channel	Source	Destination	Message Transmit Register	Response Transmit Register
channel 0 NS	CA55	CM33_FPU	MSG_INT_(SET/CLR/STS)0_NS	RSP_INT_(SET/CLR/STS)0_NS
channel 1 NS	CA55	CM33	MSG_INT_(SET/CLR/STS)1_NS	RSP_INT_(SET/CLR/STS)1_NS
channel 2 NS	CM33_FPU	CA55	MSG_INT_(SET/CLR/STS)2_NS	RSP_INT_(SET/CLR/STS)2_NS
channel 3 NS	CM33_FPU	CM33	MSG_INT_(SET/CLR/STS)3_NS	RSP_INT_(SET/CLR/STS)3_NS
channel 4 NS	CM33	CA55	MSG_INT_(SET/CLR/STS)4_NS	RSP_INT_(SET/CLR/STS)4_NS
channel 5 NS	CM33	CM33_FPU	MSG_INT_(SET/CLR/STS)5_NS	RSP_INT_(SET/CLR/STS)5_NS
channel 0 S	CA55	CM33_FPU	MSG_INT_(SET/CLR/STS)0_S	RSP_INT_(SET/CLR/STS)0_S
channel 1 S	CA55	CM33	MSG_INT_(SET/CLR/STS)1_S	RSP_INT_(SET/CLR/STS)1_S
channel 2 S	CM33_FPU	CA55	MSG_INT_(SET/CLR/STS)2_S	RSP_INT_(SET/CLR/STS)2_S
channel 3 S	CM33_FPU	CM33	MSG_INT_(SET/CLR/STS)3_S	RSP_INT_(SET/CLR/STS)3_S
channel 4 S	CM33	CA55	MSG_INT_(SET/CLR/STS)4_S	RSP_INT_(SET/CLR/STS)4_S
channel 5 S	CM33	CM33_FPU	MSG_INT_(SET/CLR/STS)5_S	RSP_INT_(SET/CLR/STS)5_S

12.5.2 Event Routing Function

(1) 4-Phase handshake

Figure 12.4 shows each EVENT I/O interface on the router meets the requirements for a REQ/ACK 4-phase handshake. EVENT I/O interface is considered to be in the Idle state when both REQ/ACK are Low, and in the Busy state during the rest of the period.

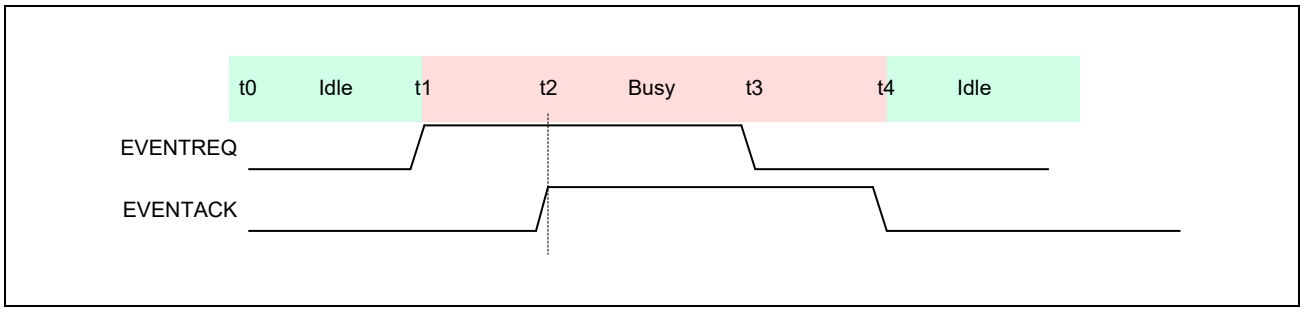


Figure 12.4 4-Phase Handshake

(2) Single Event handshake

Figure 12.5 is the simplest single event example. A pair of red arrows indicates the start and end of a single REQ/ACK handshake. One EM (Event Master) sends an event to one or more ESs (Event Slave) through an EB (Event Bridge), that is a MHU_ERU. Even if there are multiple ESs, the handshake is performed with independent interfaces.

There are no restrictions other than the time between each step being as short as possible.

There is no restriction on the order of completion of the handshake on different interfaces. For example, t2 can be executed after t3.

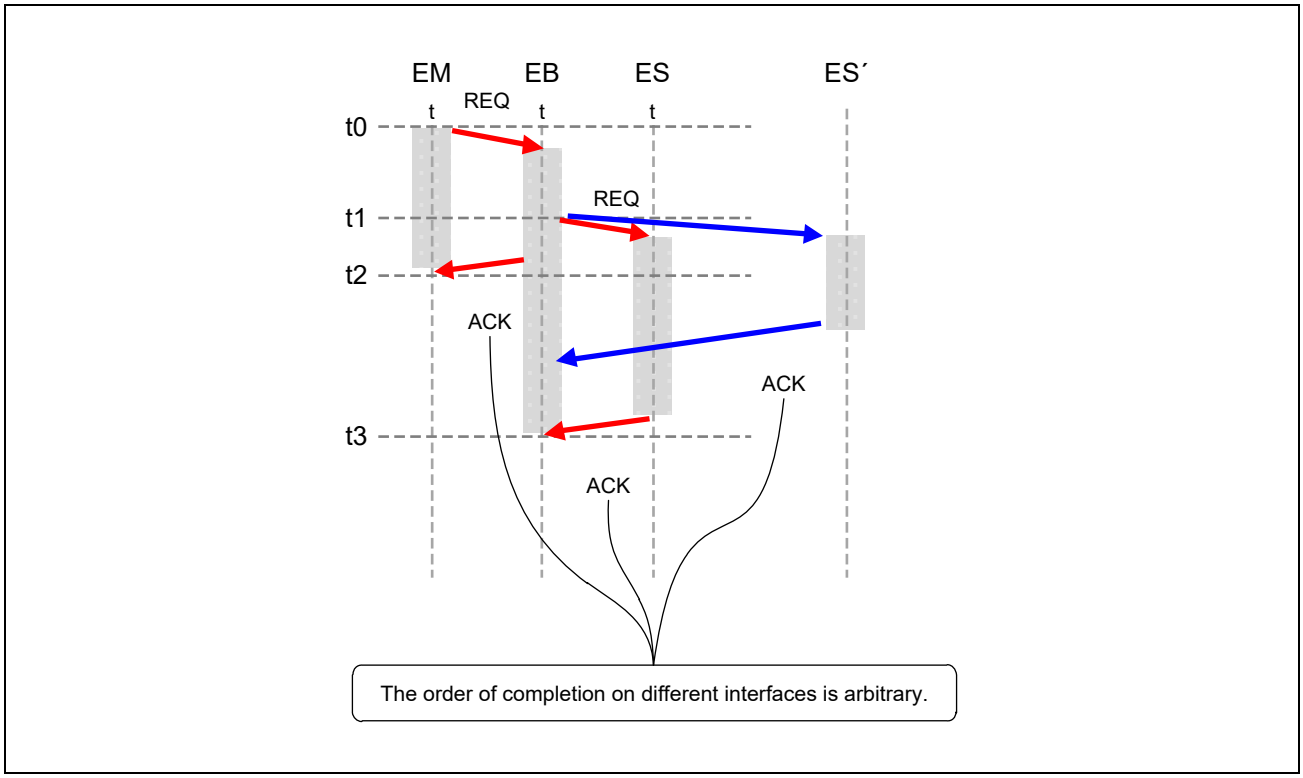


Figure 12.5 Single Event Handshake

(3) Multiple Event handshake (Event Hold)

If the EVENTO interface is busy and the EVENTI interface receives an event, the EVENTI interface completes the handshake without waiting for the EVENTO interface handshake to complete. EVENTO interface holds the next event until the first event completes.

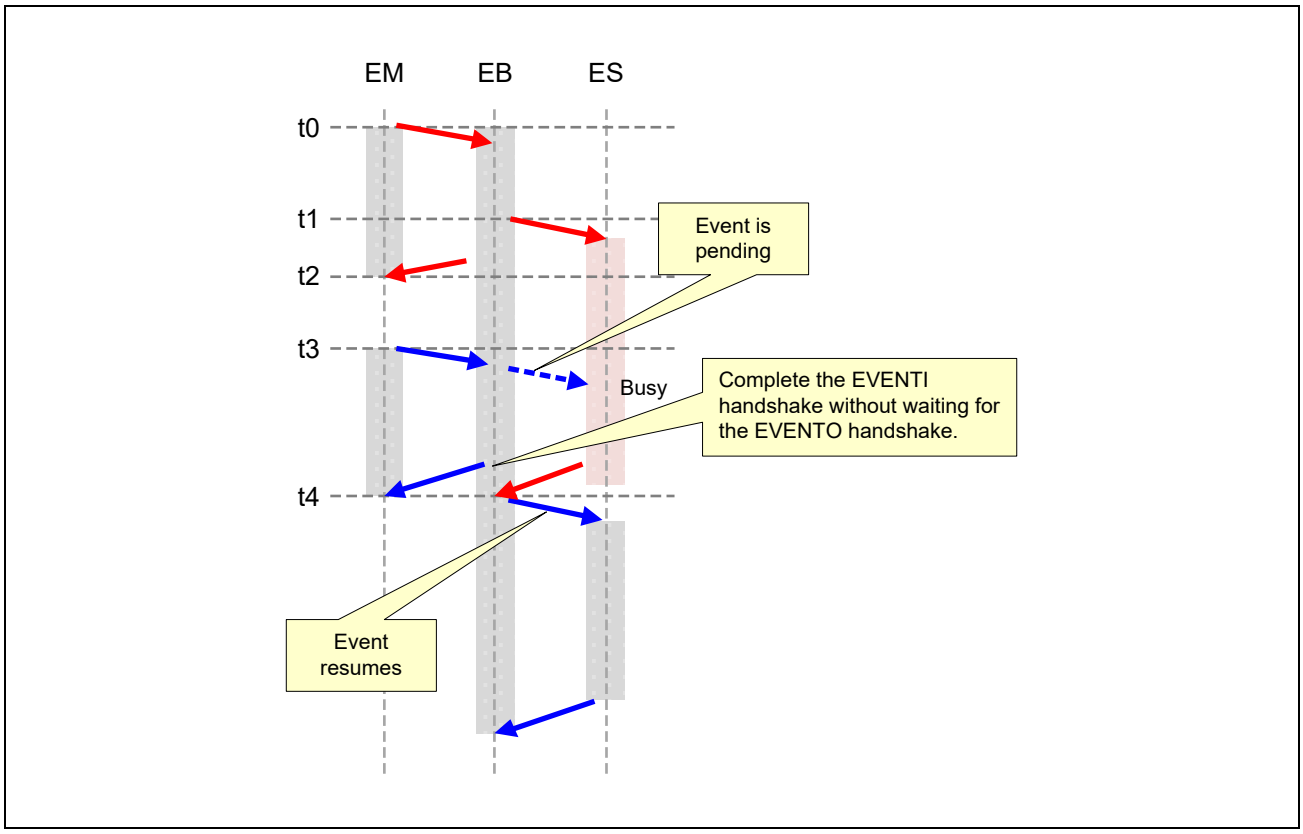


Figure 12.6 Multiple Event Handshake (Event Hold)

(4) Multiple Event (Event Integration)

If the router has multiple pending events, unified them when the EVENTO interface goes into the Idle state.

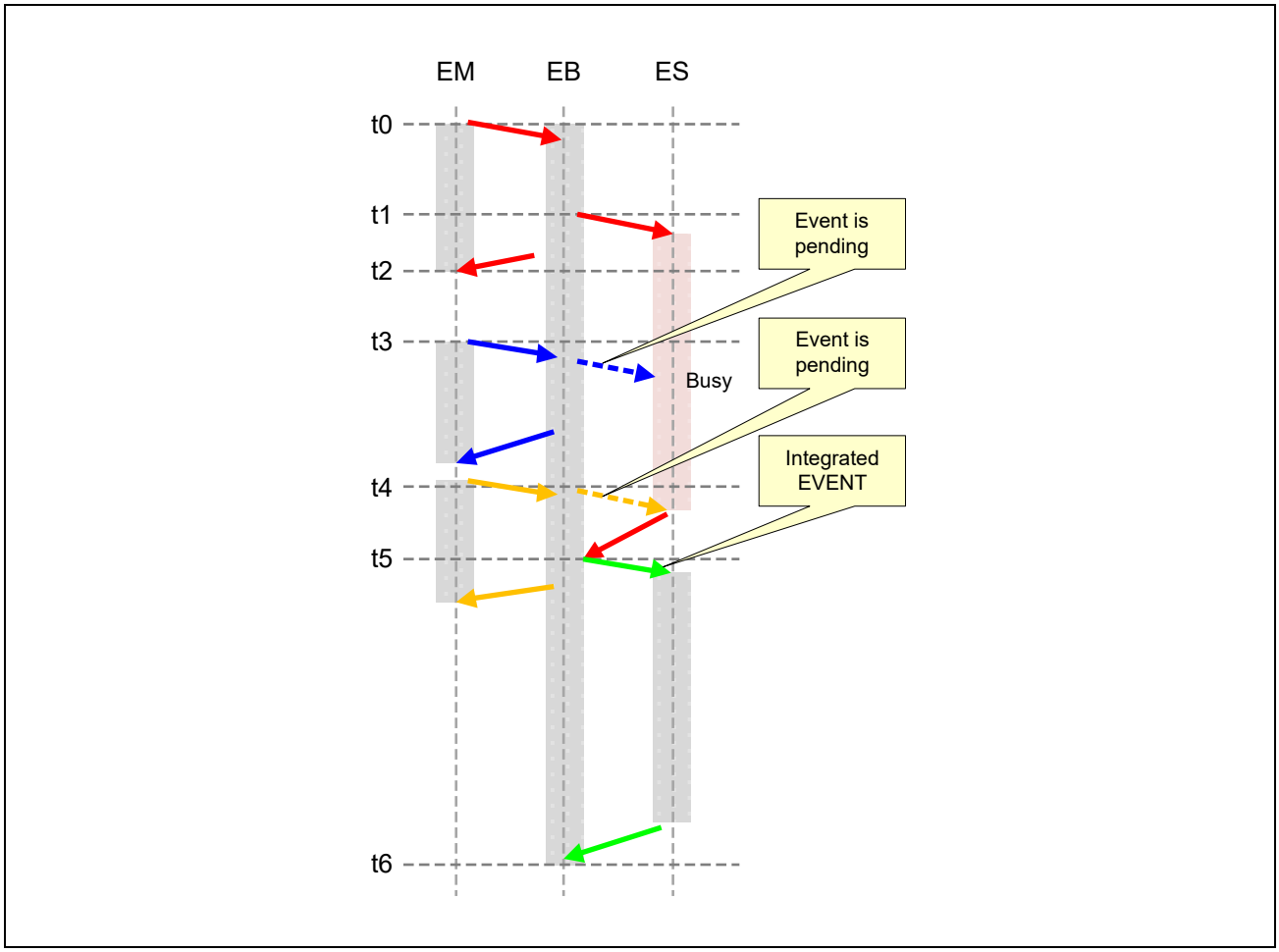


Figure 12.7 Multiple Event Handshake (EVENT Integration)

12.6 Operation Sequence

12.6.1 Message Transmit Sequence

Show Message transmit sequence from Cortex-M33/Cortex-M33_FPU to Cortex-A55 in below and **Figure 12.8**.

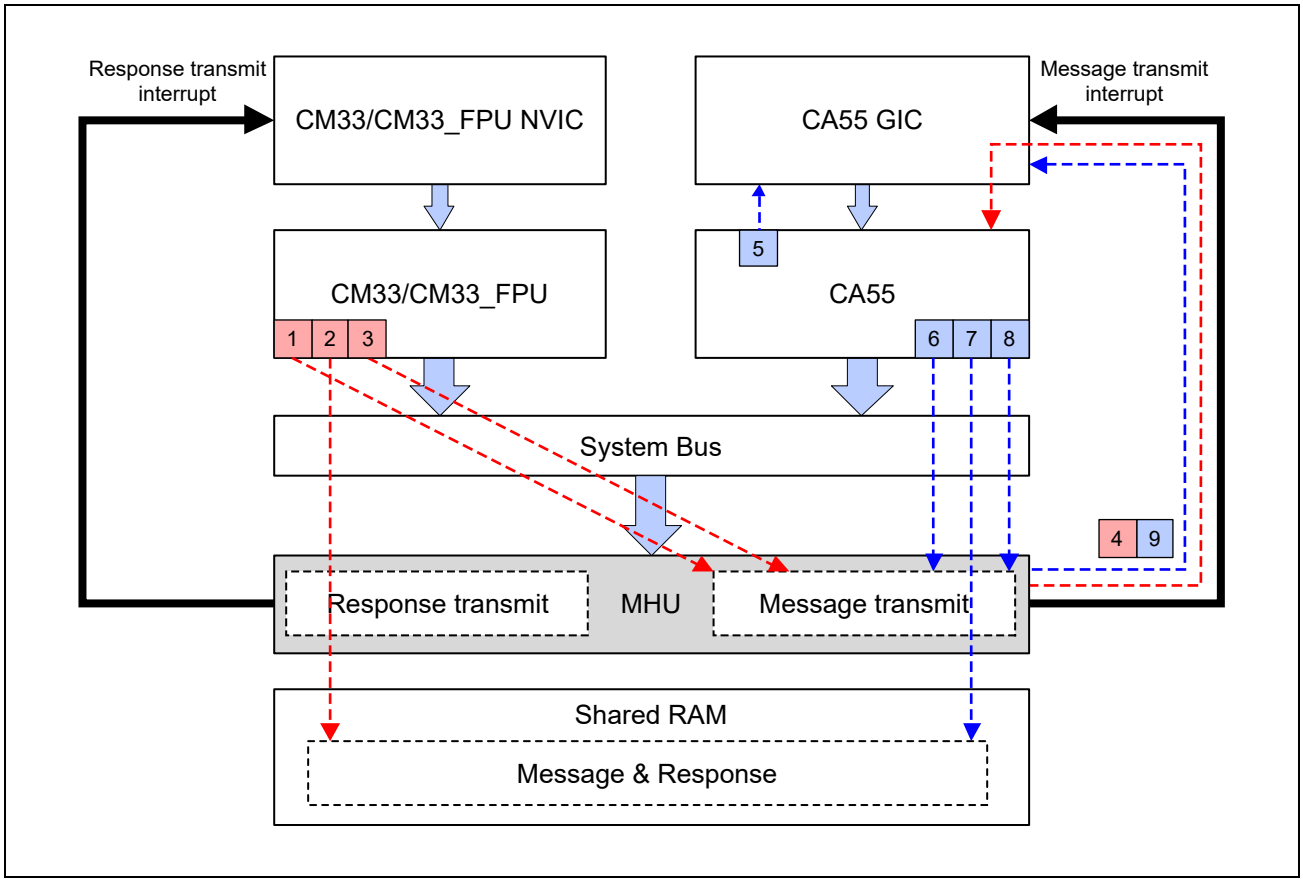


Figure 12.8 Message Transmit Sequence (CM33/CM33_FPU -> CA55)

(1) Cortex-M33/Cortex-M33_FPU: Check “0” for bit0 in register: MSG_INT_STS4/MSG_INT_STS2. (Confirm cleared status and move to (2))

(2) Cortex-M33/Cortex-M33_FPU: Write message to Shared RAM

(3) Cortex-M33/Cortex-M33_FPU: Write “1” to bit 0 in register: MSG_INT_SET4/MSG_INT_SET2

(4) MHU channel 4/2: Assert interrupt: int_msg_ch4/ch2 for Cortex-A55

(5) Cortex-A55: Check interrupt reason (Detect message transmit)

(6) Cortex-A55: Check “1” for bit 0 in register: MSG_INT_STS4/MSG_INT_STS2

(7) Cortex-A55: Read message from shared RAM

(8) Cortex-A55: Write “1” to bit0 in register: MSG_INT_CLR4/MSG_INT_CLR2

(9) MHU channel 4/2: Negate interrupt: int_msg_ch4/ch2 to Cortex-A55

12.6.2 Response Transmit Sequence

Show Response transmit sequence from Cortex-A55 to Cortex-M33/Cortex-M33_FPU in below and **Figure 12.9**.

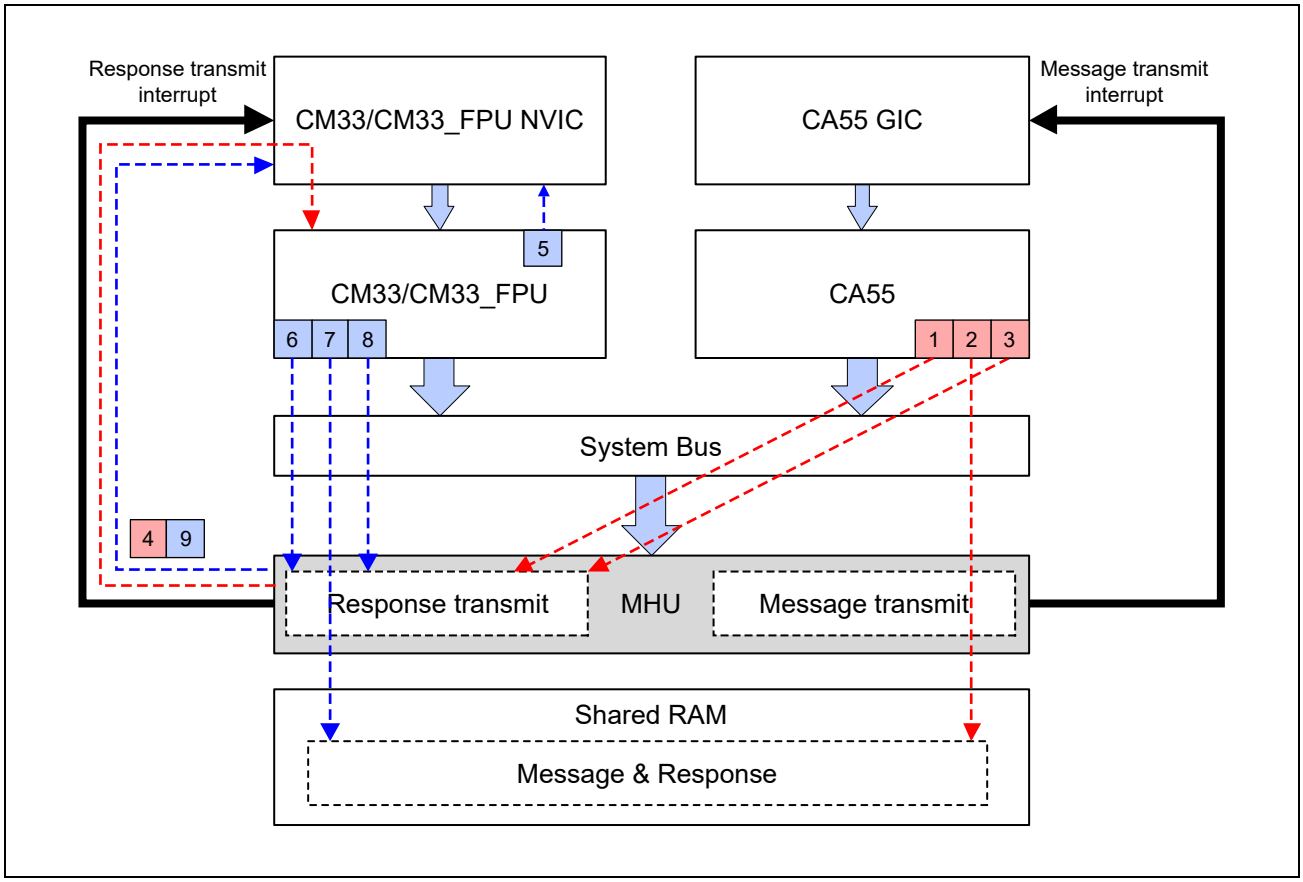


Figure 12.9 Response Transmit Sequence (CA55 -> CM33/CM33_FPU)

(1) Cortex-A55:	Check “0” for bit0 in register: RSP_INT_STS4/RSP_INT_STS2. (Confirm cleared status and move to (2))
(2) Cortex-A55:	Write Response to Shared RAM
(3) Cortex-A55:	Write “1” to bit 0 in register: RSP_INT_SET4/RSP_INT_SET2
(4) MHU channel 4/2:	Assert interrupt: int_rsp_ch4/ch2 for Cortex-M33/Cortex-M33_FPU
(5) Cortex-M33/Cortex-M33_FPU:	Check interrupt reason (Detect response transmit)
(6) Cortex-M33/Cortex-M33_FPU:	Check “1” for bit 0 in register: RSP_INT_STS4/RSP_INT_STS2
(7) Cortex-M33/Cortex-M33_FPU:	Read response from shared RAM
(8) Cortex-M33/Cortex-M33_FPU:	Write “1” to bit0 in register: RSP_INT_CLR4/RSP_INT_CLR2
(9) MHU channel 4/2:	Negate interrupt:int_rsp_ch4/ch2 to Cortex-M33/Cortex-M33_FPU

13. TrustZone Address Space Controller (TZC)

13.1 Overview

This LSI has six TrustZone Address Space Controller (TZC) to realize memory access in a safe area. It performs security checks on transactions to memory or peripherals. Transactions must meet security requirements to access memory or peripherals. This is an IP (“CoreLink™ TrustZone Address Space Controller TZC-400”) provided by Arm, for details on the functions of TZC-400, see the relevant Technical Reference Manual.

13.1.1 Features

The TZC works between TrustZone system ACE-Lite masters and ACE-Lite slaves to filter bus access from master to slave. It performs filtering based on the security requirements specified in the address space.

Each TZC is installed on the interface of the following modules of this LSI.

- Internal RAM for ACPU (SRAM: 256KB × 2)
- Internal RAM for MCPU (SRAM: 256 KB × 2)
- xSPI/Octa Memory Controller
- DDR4/LPDDR4 SDRAM Memory Controller

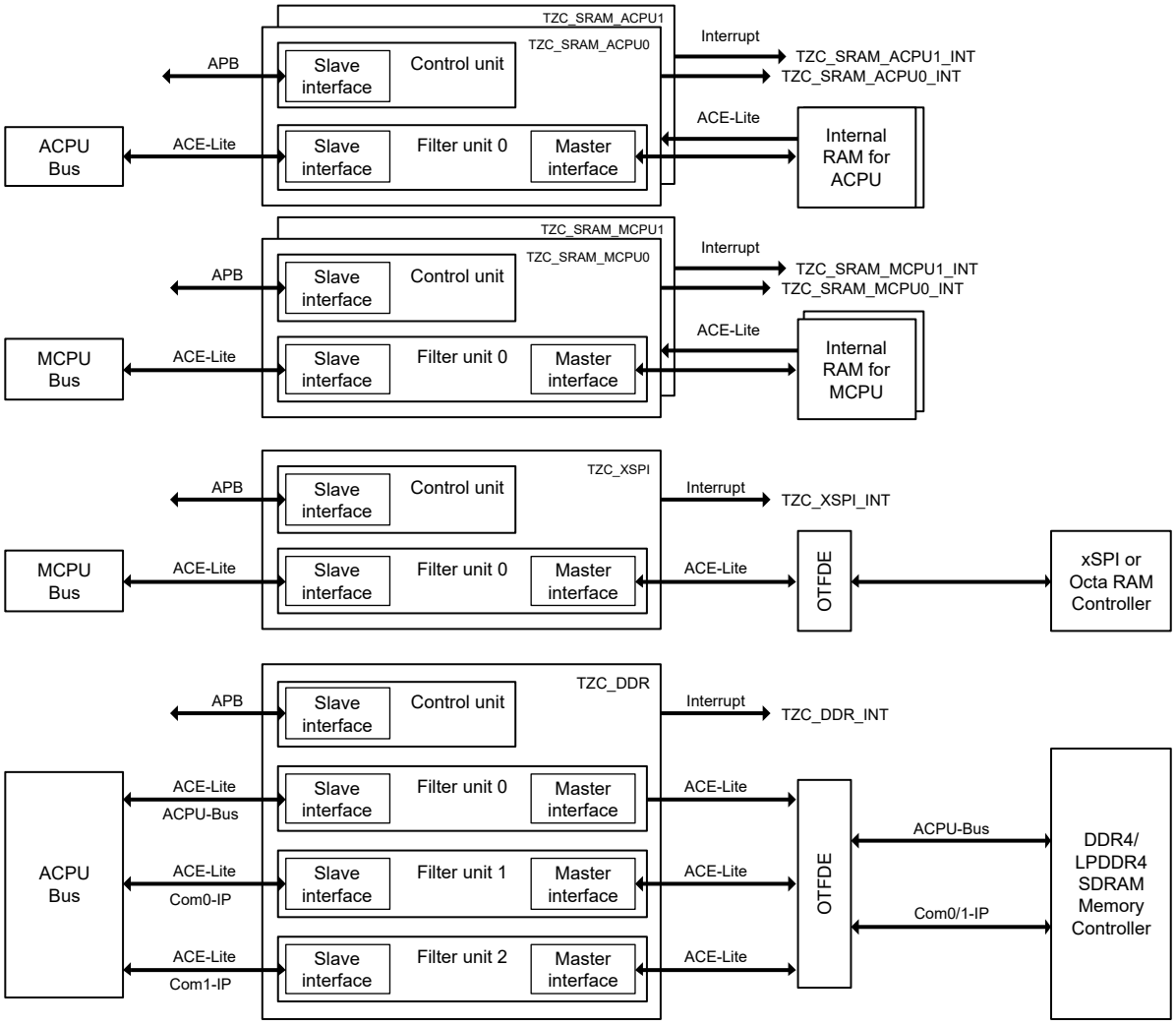
For details on the internal bus, please refer to **Section 5, LSI Internal Bus**.

The TZC provides the following key features:

- The ability to define up to eight address regions in the area map.
- A default base region to cover all remaining portions of the address map.
- Software programmable security access permissions for each address region through an APB interface. This includes the default base region, Region 0.
- Filter units only allow data transfer between an ACE-Lite master and an ACE-Lite slave if the security status of the ACE-Lite transaction and its identity match the security settings of the memory region it addresses.
- Common region configuration register settings shared between multiple filter units.
- Normal path for accesses with a much higher outstanding access support.
- Identity-based filtering of Non-secure accesses.
- Reporting and interrupt signaling that is configurable from software to manage failed permission checks. You can program the TZC to assert Interrupt (TZC_SRAM_ACPU0_INT, TZC_SRAM_ACPU1_INT, TZC_SRAM_MCPU0_INT, TZC_SRAM_MCPU1_INT, TZC_XSPI_INT, TZC_DDR_INT) when an access fails its security check.
- Gate keeper to allow or block accesses to each filter unit.
- Support for 256 outstanding transactions on the normal paths.

13.1.2 Block Diagram

Figure 13.1 shows a block diagram.



Note: OTFDE: On-The-Fly Decryption/Encryption

Figure 13.1 Block Diagram

Filter units perform security checks. Each filter unit has an ACE-Lite slave interface and an ACE-Lite master interface. All filter units operate from one set of shared region configuration registers. This ensures consistency across all filter units.

The number of filter units of TZC installed on each interface is as follows.

- Internal RAM for ACPU (256 KB): 2 filter unit 0, 1
- Internal RAM for MCPU (256 KB): 2 filter unit 0, 1
- xSPI/Octa RAM Controller: 1 filter unit 0
- DDR4/LPDDR4 SDRAM Memory Controller: 3 filter units
  - filter unit 0: Access from the ACPU bus passes
  - filter unit 1: Access from the Communication 0 IP (SDHI, GbE) passes
  - filter unit 2: Access from the Communication 1 IP (USB, PCIe) passes

13.1.3 External Pins

In TZC, there are no external pins.

13.2 Register Configuration

The base address of the TZC associated with each module is as follows:

- TZC_SRAM_ACPU0: Internal RAM for ACPU0 (256KB)

  H'0_1120_0000 (Cortex-A55 Address Space)
  H'4120_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
  H'5120_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

- TZC_SRAM_ACPU1: Internal RAM for ACPU1 (256KB)

  H'0_1121_0000 (Cortex-A55 Address Space)
  H'4121_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
  H'5121_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

- TZC_SRAM_MCPU0: Internal RAM for MCPU0 (256KB)

  H'0_1123_0000 (Cortex-A55 Address Space)
  H'4123_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
  H'5123_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

- TZC_SRAM_MCPU1: Internal RAM for MCPU1 (256KB)

  H'0_1124_0000 (Cortex-A55 Address Space)
  H'4124_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
  H'5124_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

- TZC_XSPI: xSPI/Octa RAM Memory Controller

  H'0_1125_0000 (Cortex-A55 Address Space)
  H'4125_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
  H'5125_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

- TZC_DDR: DDR4/LPDDR4 SDRAM Memory Controller

  H'0_1126_0000 (Cortex-A55 Address Space)
  H'4126_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
  H'5126_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

For details on the functions of TZC-400, see the relevant Technical Reference Manual. (ARM® CoreLink™ TZC-400 TrustZone® Address Space Controller Technical Reference Manual, Revision: r0p1)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

13.3 Register Descriptions

For details on the functions of TZC-400, see the relevant Technical Reference Manual. (ARM® CoreLink™ TZC-400 TrustZone® Address Space Controller Technical Reference Manual, Revision: r0p1)

TZC has registers to specify the address area where access is controlled. Please refer Overall Address Space (**Table 5.1, Detailed Address Space**) for the addresses specified in these registers.

13.4 Operation

For details on the functions of TZC-400, see the relevant Technical Reference Manual. (ARM® CoreLink™ TZC-400 TrustZone® Address Space Controller Technical Reference Manual, Revision: r0p1)

13.5 Usage Note

For details on the functions of TZC-400, see the relevant Technical Reference Manual. (ARM® CoreLink™ TZC-400 TrustZone® Address Space Controller Technical Reference Manual, Revision: r0p1)

14. Direct Memory Access Controller

The direct memory access controller can be used in place of the CPU to perform high-speed transfers between external memory, on-chip memory, memory-mapped external devices, and on-chip peripheral modules.

This module has controllers that handles secure and non-secure access. Do not assign the same DMA transfer request to secure access and non- secure access, respectively.

14.1 Features

- Number of channels selectable: 16 channels (CH0 to CH15).
- 4-Gbyte address space (according to the architecture)
- Transfer data size: Byte, 2 bytes, 4 bytes, 8 bytes, 16 bytes, 32 bytes, 64 bytes, and 128 bytes
- Maximum transfer count: $2^{32} - 1$ bytes
- Address mode: Dual address mode
- Transfer requests: Can be selected from the two types of on-chip peripheral module request, and auto request (software trigger)
- Transfer mode: Single transfer mode and block transfer mode are selectable.
- Priority: The channel priority levels within channels 0 to 7 and within channels 8 to 15 are selectable between fixed mode and round-robin mode (the channel priority level between the group of channels 0 to 7 and the group of channels 8 to 15 is round-robin mode).
- Interrupt request: An interrupt request can be sent to the CPU on completion of data transfer (DMA transfer end interrupt per channel) or on occurrence of a transfer error (DMA error interrupt).
- The DMA registers have a continuous execution function that allows the next DMA transfer to be executed continuously by making settings for the next DMA transfer during execution of the current DMA transfer. This continuous execution function can be enabled or disabled independently in each channel.
- Link mode: In this mode, the setting data (descriptor data) located in the memory by the CPU is automatically retrieved by the DMAC, and DMA transfer is performed according to those values.
- Buffer sweep: If an ongoing DMA transfer is forced to end, the data already retrieved into the buffer can be output before DMA transfer ends.
- Interval: A specific DMA transfer interval can be specified to adjust the bus occupancy.

14.2 Input/Output Pins

Requests from external pins are not supported.

14.3 Register Configuration

The register configuration is shown in the figure below.

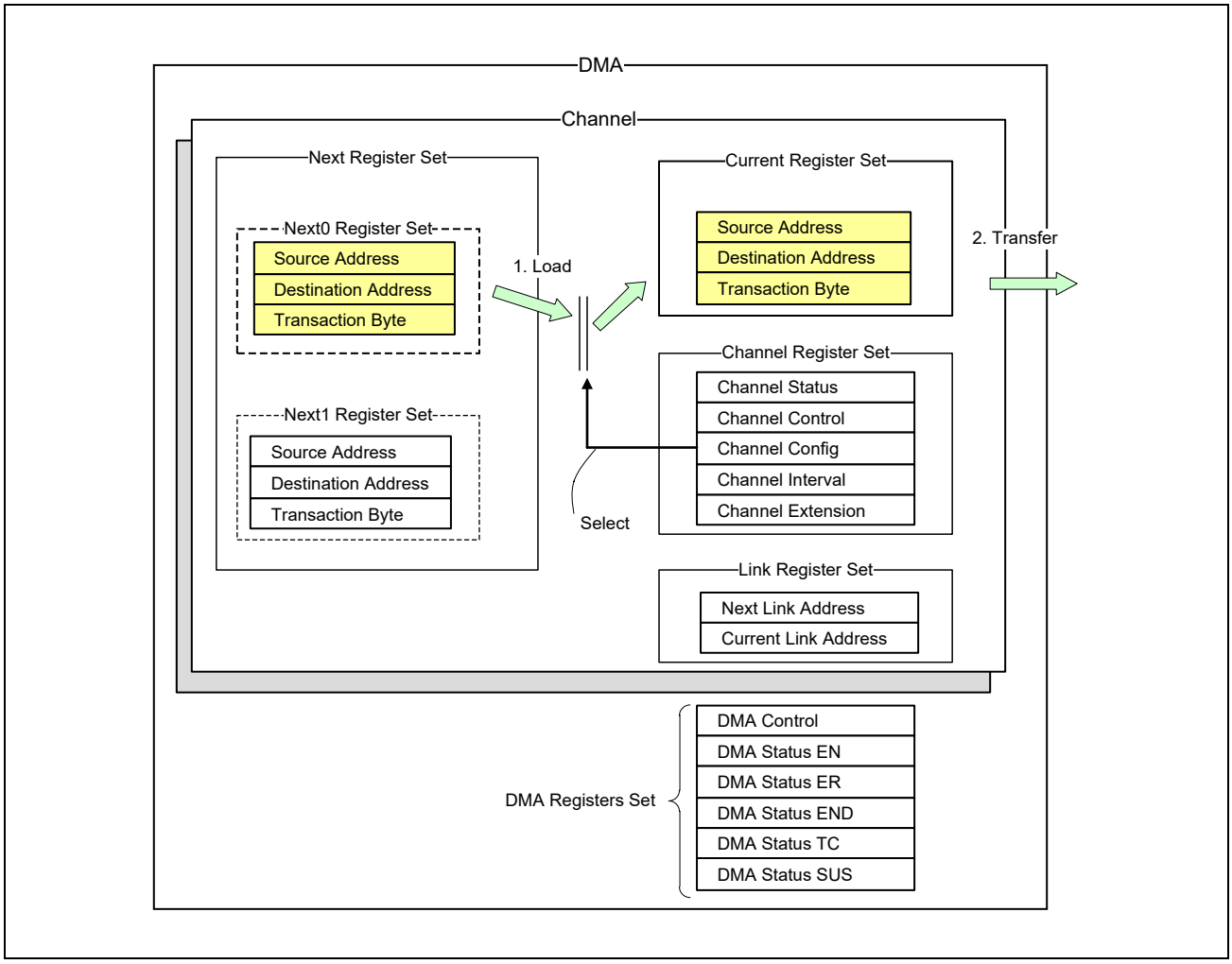


Figure 14.1 Register Configuration

(a) Next Register Set

This register set is used to set the source address, destination address, and transfer byte count of the DMA transaction to be executed next.

It consists of the Next0 register set and the Next1 register set.

In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.

These registers set values are loaded to the Current Register Set and used for DMA transfer.

(b) Current Register Set

This register set indicates the source address, destination address, and transfer byte count of the currently executed DMA transaction.

The values are loaded from the Next0/1 register set (register mode) or from the descriptor read data (link mode). The user cannot write directly to this register set.

The register set is automatically updated each time a DMA transaction is executed.

(c) Channel Register Set

This register set is used to make the DMA transfer settings.
The settings to be made with this register set include channel status indication, channel control, DMA transaction setting, and DMA transaction interval.

(d) Link Register Set

This register set consists of a register that sets the address of the descriptor to be loaded next in link mode (Next Link Address Register) and a register that indicates the address of the currently executed descriptor (Current Link Address Register).

The Current Link Address Register is automatically updated when a descriptor is read. The user cannot write directly to this register set.

(e) DMA Register Set

This register set consists of a register that controls DMA as a whole and registers that indicate the status of the corresponding channels. It enables channel priority control as well as the monitoring of the channel status (EN, ER, END, TC, and SUS).

(f) Extended Resource Selector Register Set

This register set is used to select the on-chip peripheral module to perform DMA transfer request.

14.4 Register Descriptions

Table 14.1 lists the register configuration. There are eleven control registers and five status registers for each channel, and twelve common control registers are used by all channels. In addition, there is one extension resource selector per two channels.

The notation for the registers of each channel is as follows.

- Next 0 Source Address Register of channel 0 for secure access: Next 0 Source Address Register 0S (abbreviation: N0SA_0S)
- Next 0 Source Address Register of channel 0 for non-secure access: Next 0 Source Address Register 0 (abbreviation: N0SA_0)

(1) AXI I/F

Base Address Name	Base Address	
<Base_S0>	H'0_1180_0000	(Cortex-A55 Address Space)
	H'4180_0000	(Cortex-M33/Cortex-M33_FPU Address Space Secure)
	H'5180_0000	(Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)
<Base_NS0>	H'0_1182_0000	(Cortex-A55 Address Space)
	H'4182_0000	(Cortex-M33/Cortex-M33_FPU Address Space Secure)
	H'5182_0000	(Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Note: Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 14.1 Register Configuration (1/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
0	Next0 Source Address Register 0/0S	N0SA_0	N0SA_0S	RW	H'0000_0000	<Base_NS0> +H'0000	<Base_S0>+ H'0000	32
	Next0 Destination Address Register 0/0S	N0DA_0	N0DA_0S	RW	H'0000_0000	<Base_NS0> +H'0004	<Base_S0>+ H'0004	32
	Next0 Transaction Byte Register 0/0S	N0TB_0	N0TB_0S	RW	H'0000_0000	<Base_NS0> +H'0008	<Base_S0>+ H'0008	32
	Next1 Source Address Register 0/0S	N1SA_0	N1SA_0S	RW	H'0000_0000	<Base_NS0> +H'000C	<Base_S0>+ H'000C	32
	Next1 Destination Address Register 0/0S	N1DA_0	N1DA_0S	RW	H'0000_0000	<Base_NS0> +H'0010	<Base_S0>+ H'0010	32
	Next1 Transaction Byte Register 0/0S	N1TB_0	N1TB_0S	RW	H'0000_0000	<Base_NS0> +H'0014	<Base_S0>+ H'0014	32
	Current Source Address Register 0/0S	CRSA_0	CRSA_0S	R	H'0000_0000	<Base_NS0> +H'0018	<Base_S0>+ H'0018	32
	Current Destination Address Register 0/0S	CRDA_0	CRDA_0S	R	H'0000_0000	<Base_NS0> +H'001C	<Base_S0>+ H'001C	32
	Current Transaction Byte Register 0/0S	CRTB_0	CRTB_0S	R	H'0000_0000	<Base_NS0> +H'0020	<Base_S0>+ H'0020	32
	Channel Status Register 0/0S	CHSTAT_0	CHSTAT_0S	R	H'0000_0000	<Base_NS0> +H'0024	<Base_S0>+ H'0024	32
	Channel Control Register 0/0S	CHCTRL_0	CHCTRL_0S	RW	H'0000_0000	<Base_NS0> +H'0028	<Base_S0>+ H'0028	32
	Channel Configuration Register 0/0S	CHCFG_0	CHCFG_0S	RW	H'0000_0000	<Base_NS0> +H'002C	<Base_S0>+ H'002C	32

Table 14.1 Register Configuration (2/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
0	Channel Interval Register 0/0S	CHITVL_0	CHITVL_0S	RW	H'0000_0000	<Base_NS0> +H'0030	<Base_S0>+ H'0030	32
	Channel Extension Register 0/0S	CHEXT_0	CHEXT_0S	RW	H'0000_0000	<Base_NS0> +H'0034	<Base_S0>+ H'0034	32
	Next Link Address Register 0/0S	NXLA_0	NXLA_0S	RW	H'0000_0000	<Base_NS0> +H'0038	<Base_S0>+ H'0038	32
	Current Link Address Register 0/0S	CRLA_0	CRLA_0S	R	H'0000_0000	<Base_NS0> +H'003C	<Base_S0>+ H'003C	32
1	Next0 Source Address Register 1/1S	N0SA_1	N0SA_1S	RW	H'0000_0000	<Base_NS0> +H'0040	<Base_S0>+ H'0040	32
	Next0 Destination Address Register 1/1S	N0DA_1	N0DA_1S	RW	H'0000_0000	<Base_NS0> +H'0044	<Base_S0>+ H'0044	32
	Next0 Transaction Byte Register 1/1S	N0TB_1	N0TB_1S	RW	H'0000_0000	<Base_NS0> +H'0048	<Base_S0>+ H'0048	32
	Next1 Source Address Register 1/1S	N1SA_1	N1SA_1S	RW	H'0000_0000	<Base_NS0> +H'004C	<Base_S0>+ H'004C	32
	Next1 Destination Address Register 1/1S	N1DA_1	N1DA_1S	RW	H'0000_0000	<Base_NS0> +H'0050	<Base_S0>+ H'0050	32
	Next1 Transaction Byte Register 1/1S	N1TB_1	N1TB_1S	RW	H'0000_0000	<Base_NS0> +H'0054	<Base_S0>+ H'0054	32
	Current Source Address Register 1/1S	CRSA_1	CRSA_1S	R	H'0000_0000	<Base_NS0> +H'0058	<Base_S0>+ H'0058	32
	Current Destination Address Register 1/1S	CRDA_1	CRDA_1S	R	H'0000_0000	<Base_NS0> +H'005C	<Base_S0>+ H'005C	32
	Current Transaction Byte Register 1/1S	CRTB_1	CRTB_1S	R	H'0000_0000	<Base_NS0> +H'0060	<Base_S0>+ H'0060	32
	Channel Status Register 1/1S	CHSTAT_1	CHSTAT_1S	R	H'0000_0000	<Base_NS0> +H'0064	<Base_S0>+ H'0064	32
	Channel Control Register 1/1S	CHCTRL_1	CHCTRL_1S	RW	H'0000_0000	<Base_NS0> +H'0068	<Base_S0>+ H'0068	32
	Channel Configuration Register 1/1S	CHCFG_1	CHCFG_1S	RW	H'0000_0000	<Base_NS0> +H'006C	<Base_S0>+ H'006C	32
	Channel Interval Register 1/1S	CHITVL_1	CHITVL_1S	RW	H'0000_0000	<Base_NS0> +H'0070	<Base_S0>+ H'0070	32
	Channel Extension Register 1/1S	CHEXT_1	CHEXT_1S	RW	H'0000_0000	<Base_NS0> +H'0074	<Base_S0>+ H'0074	32
	Next Link Address Register 1/1S	NXLA_1	NXLA_1S	RW	H'0000_0000	<Base_NS0> +H'0078	<Base_S0>+ H'0078	32
	Current Link Address Register 1/1S	CRLA_1	CRLA_1S	R	H'0000_0000	<Base_NS0> +H'007C	<Base_S0>+ H'007C	32
2	Next0 Source Address Register 2/2S	N0SA_2	N0SA_2S	RW	H'0000_0000	<Base_NS0> +H'0080	<Base_S0>+ H'0080	32
	Next0 Destination Address Register 2/2S	N0DA_2	N0DA_2S	RW	H'0000_0000	<Base_NS0> +H'0084	<Base_S0>+ H'0084	32
	Next0 Transaction Byte Register 2/2S	N0TB_2	N0TB_2S	RW	H'0000_0000	<Base_NS0> +H'0088	<Base_S0>+ H'0088	32
	Next1 Source Address Register 2/2S	N1SA_2	N1SA_2S	RW	H'0000_0000	<Base_NS0> +H'008C	<Base_S0>+ H'008C	32
	Next1 Destination Address Register 2/2S	N1DA_2	N1DA_2S	RW	H'0000_0000	<Base_NS0> +H'0090	<Base_S0>+ H'0090	32
	Next1 Transaction Byte Register 2/2S	N1TB_2	N1TB_2S	RW	H'0000_0000	<Base_NS0> +H'0094	<Base_S0>+ H'0094	32

Table 14.1 Register Configuration (3/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
2	Current Source Address Register 2/2S	CRSA_2	CRSA_2S	R	H'0000_0000	<Base_NS0> +H'0098	<Base_S0>+ H'0098	32
	Current Destination Address Register 2/2S	CRDA_2	CRDA_2S	R	H'0000_0000	<Base_NS0> +H'009C	<Base_S0>+ H'009C	32
	Current Transaction Byte Register 2/2S	CRTB_2	CRTB_2S	R	H'0000_0000	<Base_NS0> +H'00A0	<Base_S0>+ H'00A0	32
	Channel Status Register 2/2S	CHSTAT_2	CHSTAT_2S	R	H'0000_0000	<Base_NS0> +H'00A4	<Base_S0>+ H'00A4	32
	Channel Control Register 2/2S	CHCTRL_2	CHCTRL_2S	RW	H'0000_0000	<Base_NS0> +H'00A8	<Base_S0>+ H'00A8	32
	Channel Configuration Register 2/2S	CHCFG_2	CHCFG_2S	RW	H'0000_0000	<Base_NS0> +H'00AC	<Base_S0>+ H'00AC	32
	Channel Interval Register 2/2S	CHITVL_2	CHITVL_2S	RW	H'0000_0000	<Base_NS0> +H'00B0	<Base_S0>+ H'00B0	32
	Channel Extension Register 2/2S	CHEXT_2	CHEXT_2S	RW	H'0000_0000	<Base_NS0> +H'00B4	<Base_S0>+ H'00B4	32
	Next Link Address Register 2/2S	NXLA_2	NXLA_2S	RW	H'0000_0000	<Base_NS0> +H'00B8	<Base_S0>+ H'00B8	32
	Current Link Address Register 2/2S	CRLA_2	CRLA_2S	R	H'0000_0000	<Base_NS0> +H'00BC	<Base_S0>+ H'00BC	32
3	Next0 Source Address Register 3/3S	N0SA_3	N0SA_3S	RW	H'0000_0000	<Base_NS0> +H'00C0	<Base_S0>+ H'00C0	32
	Next0 Destination Address Register 3/3S	N0DA_3	N0DA_3S	RW	H'0000_0000	<Base_NS0> +H'00C4	<Base_S0>+ H'00C4	32
	Next0 Transaction Byte Register 3/3S	N0TB_3	N0TB_3S	RW	H'0000_0000	<Base_NS0> +H'00C8	<Base_S0>+ H'00C8	32
	Next1 Source Address Register 3/3S	N1SA_3	N1SA_3S	RW	H'0000_0000	<Base_NS0> +H'00CC	<Base_S0>+ H'00CC	32
	Next1 Destination Address Register 3/3S	N1DA_3	N1DA_3S	RW	H'0000_0000	<Base_NS0> +H'00D0	<Base_S0>+ H'00D0	32
	Next1 Transaction Byte Register 3/3S	N1TB_3	N1TB_3S	RW	H'0000_0000	<Base_NS0> +H'00D4	<Base_S0>+ H'00D4	32
	Current Source Address Register 3/3S	CRSA_3	CRSA_3S	R	H'0000_0000	<Base_NS0> +H'00D8	<Base_S0>+ H'00D8	32
	Current Destination Address Register 3/3S	CRDA_3	CRDA_3S	R	H'0000_0000	<Base_NS0> +H'00DC	<Base_S0>+ H'00DC	32
	Current Transaction Byte Register 3/3S	CRTB_3	CRTB_3S	R	H'0000_0000	<Base_NS0> +H'00E0	<Base_S0>+ H'00E0	32
	Channel Status Register 3/3S	CHSTAT_3	CHSTAT_3S	R	H'0000_0000	<Base_NS0> +H'00E4	<Base_S0>+ H'00E4	32
	Channel Control Register 3/3S	CHCTRL_3	CHCTRL_3S	RW	H'0000_0000	<Base_NS0> +H'00E8	<Base_S0>+ H'00E8	32
	Channel Configuration Register 3/3S	CHCFG_3	CHCFG_3S	RW	H'0000_0000	<Base_NS0> +H'00EC	<Base_S0>+ H'00EC	32
	Channel Interval Register 3/3S	CHITVL_3	CHITVL_3S	RW	H'0000_0000	<Base_NS0> +H'00F0	<Base_S0>+ H'00F0	32
	Channel Extension Register 3/3S	CHEXT_3	CHEXT_3S	RW	H'0000_0000	<Base_NS0> +H'00F4	<Base_S0>+ H'00F4	32
	Next Link Address Register 3/3S	NXLA_3	NXLA_3S	RW	H'0000_0000	<Base_NS0> +H'00F8	<Base_S0>+ H'00F8	32
	Current Link Address Register 3/3S	CRLA_3	CRLA_3S	R	H'0000_0000	<Base_NS0> +H'00FC	<Base_S0>+ H'00FC	32

Table 14.1 Register Configuration (4/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
4	Next0 Source Address Register 4/4S	N0SA_4	N0SA_4S	RW	H'0000_0000	<Base_NS0> +H'0100	<Base_S0>+ H'0100	32
	Next0 Destination Address Register 4/4S	N0DA_4	N0DA_4S	RW	H'0000_0000	<Base_NS0> +H'0104	<Base_S0>+ H'0104	32
	Next0 Transaction Byte Register 4/4S	N0TB_4	N0TB_4S	RW	H'0000_0000	<Base_NS0> +H'0108	<Base_S0>+ H'0108	32
	Next1 Source Address Register 4/4S	N1SA_4	N1SA_4S	RW	H'0000_0000	<Base_NS0> +H'010C	<Base_S0>+ H'010C	32
	Next1 Destination Address Register 4/4S	N1DA_4	N1DA_4S	RW	H'0000_0000	<Base_NS0> +H'0110	<Base_S0>+ H'0110	32
	Next1 Transaction Byte Register 4/4S	N1TB_4	N1TB_4S	RW	H'0000_0000	<Base_NS0> +H'0114	<Base_S0>+ H'0114	32
	Current Source Address Register 4/4S	CRSA_4	CRSA_4S	R	H'0000_0000	<Base_NS0> +H'0118	<Base_S0>+ H'0118	32
	Current Destination Address Register 4/4S	CRDA_4	CRDA_4S	R	H'0000_0000	<Base_NS0> +H'011C	<Base_S0>+ H'011C	32
	Current Transaction Byte Register 4/4S	CRTB_4	CRTB_4S	R	H'0000_0000	<Base_NS0> +H'0120	<Base_S0>+ H'0120	32
	Channel Status Register 4/4S	CHSTAT_4	CHSTAT_4S	R	H'0000_0000	<Base_NS0> +H'0124	<Base_S0>+ H'0124	32
	Channel Control Register 4/4S	CHCTRL_4	CHCTRL_4S	RW	H'0000_0000	<Base_NS0> +H'0128	<Base_S0>+ H'0128	32
	Channel Configuration Register 4/4S	CHCFG_4	CHCFG_4S	RW	H'0000_0000	<Base_NS0> +H'012C	<Base_S0>+ H'012C	32
	Channel Interval Register 4/4S	CHITVL_4	CHITVL_4S	RW	H'0000_0000	<Base_NS0> +H'0130	<Base_S0>+ H'0130	32
	Channel Extension Register 4/4S	CHEXT_4	CHEXT_4S	RW	H'0000_0000	<Base_NS0> +H'0134	<Base_S0>+ H'0134	32
	Next Link Address Register 4/4S	NXLA_4	NXLA_4S	RW	H'0000_0000	<Base_NS0> +H'0138	<Base_S0>+ H'0138	32
	Current Link Address Register 4/4S	CRLA_4	CRLA_4S	R	H'0000_0000	<Base_NS0> +H'013C	<Base_S0>+ H'013C	32
5	Next0 Source Address Register 5/5S	N0SA_5	N0SA_5S	RW	H'0000_0000	<Base_NS0> +H'0140	<Base_S0>+ H'0140	32
	Next0 Destination Address Register 5/5S	N0DA_5	N0DA_5S	RW	H'0000_0000	<Base_NS0> +H'0144	<Base_S0>+ H'0144	32
	Next0 Transaction Byte Register 5/5S	N0TB_5	N0TB_5S	RW	H'0000_0000	<Base_NS0> +H'0148	<Base_S0>+ H'0148	32
	Next1 Source Address Register 5/5S	N1SA_5	N1SA_5S	RW	H'0000_0000	<Base_NS0> +H'014C	<Base_S0>+ H'014C	32
	Next1 Destination Address Register 5/5S	N1DA_5	N1DA_5S	RW	H'0000_0000	<Base_NS0> +H'0150	<Base_S0>+ H'0150	32
	Next1 Transaction Byte Register 5/5S	N1TB_5	N1TB_5S	RW	H'0000_0000	<Base_NS0> +H'0154	<Base_S0>+ H'0154	32
	Current Source Address Register 5/5S	CRSA_5	CRSA_5S	R	H'0000_0000	<Base_NS0> +H'0158	<Base_S0>+ H'0158	32
	Current Destination Address Register 5/5S	CRDA_5	CRDA_5S	R	H'0000_0000	<Base_NS0> +H'015C	<Base_S0>+ H'015C	32
	Current Transaction Byte Register 5/5S	CRTB_5	CRTB_5S	R	H'0000_0000	<Base_NS0> +H'0160	<Base_S0>+ H'0160	32
	Channel Status Register 5/5S	CHSTAT_5	CHSTAT_5S	R	H'0000_0000	<Base_NS0> +H'0164	<Base_S0>+ H'0164	32

Table 14.1 Register Configuration (5/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
5	Channel Control Register 5/5S	CHCTRL_5	CHCTRL_5S	RW	H'0000_0000	<Base_NS0> +H'0168	<Base_S0>+ H'0168	32
	Channel Configuration Register 5/5S	CHCFG_5	CHCFG_5S	RW	H'0000_0000	<Base_NS0> +H'016C	<Base_S0>+ H'016C	32
	Channel Interval Register 5/5S	CHITVL_5	CHITVL_5S	RW	H'0000_0000	<Base_NS0> +H'0170	<Base_S0>+ H'0170	32
	Channel Extension Register 5/5S	CHEXT_5	CHEXT_5S	RW	H'0000_0000	<Base_NS0> +H'0174	<Base_S0>+ H'0174	32
	Next Link Address Register 5/5S	NXLA_5	NXLA_5S	RW	H'0000_0000	<Base_NS0> +H'0178	<Base_S0>+ H'0178	32
	Current Link Address Register 5/5S	CRLA_5	CRLA_5S	R	H'0000_0000	<Base_NS0> +H'017C	<Base_S0>+ H'017C	32
6	Next0 Source Address Register 6/6S	N0SA_6	N0SA_6S	RW	H'0000_0000	<Base_NS0> +H'0180	<Base_S0>+ H'0180	32
	Next0 Destination Address Register 6/6S	N0DA_6	N0DA_6S	RW	H'0000_0000	<Base_NS0> +H'0184	<Base_S0>+ H'0184	32
	Next0 Transaction Byte Register 6/6S	N0TB_6	N0TB_6S	RW	H'0000_0000	<Base_NS0> +H'0188	<Base_S0>+ H'0188	32
	Next1 Source Address Register 6/6S	N1SA_6	N1SA_6S	RW	H'0000_0000	<Base_NS0> +H'018C	<Base_S0>+ H'018C	32
	Next1 Destination Address Register 6/6S	N1DA_6	N1DA_6S	RW	H'0000_0000	<Base_NS0> +H'0190	<Base_S0>+ H'0190	32
	Next1 Transaction Byte Register 6/6S	N1TB_6	N1TB_6S	RW	H'0000_0000	<Base_NS0> +H'0194	<Base_S0>+ H'0194	32
	Current Source Address Register 6/6S	CRSA_6	CRSA_6S	R	H'0000_0000	<Base_NS0> +H'0198	<Base_S0>+ H'0198	32
	Current Destination Address Register 6/6S	CRDA_6	CRDA_6S	R	H'0000_0000	<Base_NS0> +H'019C	<Base_S0>+ H'019C	32
	Current Transaction Byte Register 6/6S	CRTB_6	CRTB_6S	R	H'0000_0000	<Base_NS0> +H'01A0	<Base_S0>+ H'01A0	32
	Channel Status Register 6/6S	CHSTAT_6	CHSTAT_6S	R	H'0000_0000	<Base_NS0> +H'01A4	<Base_S0>+ H'01A4	32
	Channel Control Register 6/6S	CHCTRL_6	CHCTRL_6S	RW	H'0000_0000	<Base_NS0> +H'01A8	<Base_S0>+ H'01A8	32
	Channel Configuration Register 6/6S	CHCFG_6	CHCFG_6S	RW	H'0000_0000	<Base_NS0> +H'01AC	<Base_S0>+ H'01AC	32
	Channel Interval Register 6/6S	CHITVL_6	CHITVL_6S	RW	H'0000_0000	<Base_NS0> +H'01B0	<Base_S0>+ H'01B0	32
	Channel Extension Register 6/6S	CHEXT_6	CHEXT_6S	RW	H'0000_0000	<Base_NS0> +H'01B4	<Base_S0>+ H'01B4	32
	Next Link Address Register 6/6S	NXLA_6	NXLA_6S	RW	H'0000_0000	<Base_NS0> +H'01B8	<Base_S0>+ H'01B8	32
	Current Link Address Register 6/6S	CRLA_6	CRLA_6S	R	H'0000_0000	<Base_NS0> +H'01BC	<Base_S0>+ H'01BC	32

Table 14.1 Register Configuration (6/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
7	Next0 Source Address Register 7/7S	N0SA_7	N0SA_7S	RW	H'0000_0000	<Base_NS0> +H'01C0	<Base_S0>+ H'01C0	32
	Next0 Destination Address Register 7/7S	N0DA_7	N0DA_7S	RW	H'0000_0000	<Base_NS0> +H'01C4	<Base_S0>+ H'01C4	32
	Next0 Transaction Byte Register 7/7S	N0TB_7	N0TB_7S	RW	H'0000_0000	<Base_NS0> +H'01C8	<Base_S0>+ H'01C8	32
	Next1 Source Address Register 7/7S	N1SA_7	N1SA_7S	RW	H'0000_0000	<Base_NS0> +H'01CC	<Base_S0>+ H'01CC	32
	Next1 Destination Address Register 7/7S	N1DA_7	N1DA_7S	RW	H'0000_0000	<Base_NS0> +H'01D0	<Base_S0>+ H'01D0	32
	Next1 Transaction Byte Register 7/7S	N1TB_7	N1TB_7S	RW	H'0000_0000	<Base_NS0> +H'01D4	<Base_S0>+ H'01D4	32
	Current Source Address Register 7/7S	CRSA_7	CRSA_7S	R	H'0000_0000	<Base_NS0> +H'01D8	<Base_S0>+ H'01D8	32
	Current Destination Address Register 7/7S	CRDA_7	CRDA_7S	R	H'0000_0000	<Base_NS0> +H'01DC	<Base_S0>+ H'01DC	32
	Current Transaction Byte Register 7/7S	CRTB_7	CRTB_7S	R	H'0000_0000	<Base_NS0> +H'01E0	<Base_S0>+ H'01E0	32
	Channel Status Register 7/7S	CHSTAT_7	CHSTAT_7S	R	H'0000_0000	<Base_NS0> +H'01E4	<Base_S0>+ H'01E4	32
	Channel Control Register 7/7S	CHCTRL_7	CHCTRL_7S	RW	H'0000_0000	<Base_NS0> +H'01E8	<Base_S0>+ H'01E8	32
	Channel Configuration Register 7/7S	CHCFG_7	CHCFG_7S	RW	H'0000_0000	<Base_NS0> +H'01EC	<Base_S0>+ H'01EC	32
	Channel Interval Register 7/7S	CHITVL_7	CHITVL_7S	RW	H'0000_0000	<Base_NS0> +H'01F0	<Base_S0>+ H'01F0	32
	Channel Extension Register 7/7S	CHEXT_7	CHEXT_7S	RW	H'0000_0000	<Base_NS0> +H'01F4	<Base_S0>+ H'01F4	32
	Next Link Address Register 7/7S	NXLA_7	NXLA_7S	RW	H'0000_0000	<Base_NS0> +H'01F8	<Base_S0>+ H'01F8	32
	Current Link Address Register 7/7S	CRLA_7	CRLA_7S	R	H'0000_0000	<Base_NS0> +H'01FC	<Base_S0>+ H'01FC	32
Common for 0 to 7	DMA Control Registers 0-7/0-7S	DCTRL_0_7	DCTRL_0_7S	RW	H'0000_0000	<Base_NS0> +H'0300	<Base_S0>+ H'0300	32
	DMA Status EN Registers 0-7/0-7S	DSTAT_EN_0_7	DSTAT_EN_0_7S	R	H'0000_0000	<Base_NS0> +H'0310	<Base_S0>+ H'0310	32
	DMA Status ER Registers 0-7/0-7S	DSTAT_ER_0_7	DSTAT_ER_0_7S	R	H'0000_0000	<Base_NS0> +H'0314	<Base_S0>+ H'0314	32
	DMA Status END Registers 0-7/0-7S	DSTAT_END_0_7	DSTAT_END_0_7S	R	H'0000_0000	<Base_NS0> +H'0318	<Base_S0>+ H'0318	32
	DMA Status TC Registers 0-7/0-7S	DSTAT_TC_0_7	DSTAT_TC_0_7S	R	H'0000_0000	<Base_NS0> +H'031C	<Base_S0>+ H'031C	32
	DMA Status SUS Registers 0-7/0-7S	DSTAT_SUS_0_7	DSTAT_SUS_0_7S	R	H'0000_0000	<Base_NS0> +H'0320	<Base_S0>+ H'0320	32

Table 14.1 Register Configuration (7/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
8	Next0 Source Address Register 8/8S	N0SA_8	N0SA_8S	RW	H'0000_0000	<Base_NS0> +H'0400	<Base_S0>+ H'0400	32
	Next0 Destination Address Register 8/8S	N0DA_8	N0DA_8S	RW	H'0000_0000	<Base_NS0> +H'0404	<Base_S0>+ H'0404	32
	Next0 Transaction Byte Register 8/8S	N0TB_8	N0TB_8S	RW	H'0000_0000	<Base_NS0> +H'0408	<Base_S0>+ H'0408	32
	Next1 Source Address Register 8/8S	N1SA_8	N1SA_8S	RW	H'0000_0000	<Base_NS0> +H'040C	<Base_S0>+ H'040C	32
	Next1 Destination Address Register 8/8S	N1DA_8	N1DA_8S	RW	H'0000_0000	<Base_NS0> +H'0410	<Base_S0>+ H'0410	32
	Next1 Transaction Byte Register 8/8S	N1TB_8	N1TB_8S	RW	H'0000_0000	<Base_NS0> +H'0414	<Base_S0>+ H'0414	32
	Current Source Address Register 8/8S	CRSA_8	CRSA_8S	R	H'0000_0000	<Base_NS0> +H'0418	<Base_S0>+ H'0418	32
	Current Destination Address Register 8/8S	CRDA_8	CRDA_8S	R	H'0000_0000	<Base_NS0> +H'041C	<Base_S0>+ H'041C	32
	Current Transaction Byte Register 8/8S	CRTB_8	CRTB_8S	R	H'0000_0000	<Base_NS0> +H'0420	<Base_S0>+ H'0420	32
	Channel Status Register 8/8S	CHSTAT_8	HSTAT_8S	R	H'0000_0000	<Base_NS0> +H'0424	<Base_S0>+ H'0424	32
	Channel Control Register 8/8S	CHCTRL_8	CHCTRL_8S	RW	H'0000_0000	<Base_NS0> +H'0428	<Base_S0>+ H'0428	32
	Channel Configuration Register 8/8S	CHCFG_8	CHCFG_8S	RW	H'0000_0000	<Base_NS0> +H'042C	<Base_S0>+ H'042C	32
	Channel Interval Register 8/8S	CHITVL_8	CHITVL_8S	RW	H'0000_0000	<Base_NS0> +H'0430	<Base_S0>+ H'0430	32
	Channel Extension Register 8/8S	CHEXT_8	CHEXT_8S	RW	H'0000_0000	<Base_NS0> +H'0434	<Base_S0>+ H'0434	32
	Next Link Address Register 8/8S	NXLA_8	NXLA_8S	RW	H'0000_0000	<Base_NS0> +H'0438	<Base_S0>+ H'0438	32
	Current Link Address Register 8/8S	CRLA_8	CRLA_8S	R	H'0000_0000	<Base_NS0> +H'043C	<Base_S0>+ H'043C	32
9	Next0 Source Address Register 9/9S	N0SA_9	N0SA_9S	RW	H'0000_0000	<Base_NS0> +H'0440	<Base_S0>+ H'0440	32
	Next0 Destination Address Register 9/9S	N0DA_9	N0DA_9S	RW	H'0000_0000	<Base_NS0> +H'0444	<Base_S0>+ H'0444	32
	Next0 Transaction Byte Register 9/9S	N0TB_9	N0TB_9S	RW	H'0000_0000	<Base_NS0> +H'0448	<Base_S0>+ H'0448	32
	Next1 Source Address Register 9/9S	N1SA_9	N1SA_9S	RW	H'0000_0000	<Base_NS0> +H'044C	<Base_S0>+ H'044C	32
	Next1 Destination Address Register 9/9S	N1DA_9	N1DA_9S	RW	H'0000_0000	<Base_NS0> +H'0450	<Base_S0>+ H'0450	32
	Next1 Transaction Byte Register 9/9S	N1TB_9	N1TB_9S	RW	H'0000_0000	<Base_NS0> +H'0454	<Base_S0>+ H'0454	32
	Current Source Address Register 9/9S	CRSA_9	CRSA_9S	R	H'0000_0000	<Base_NS0> +H'0458	<Base_S0>+ H'0458	32
	Current Destination Address Register 9/9S	CRDA_9	CRDA_9S	R	H'0000_0000	<Base_NS0> +H'045C	<Base_S0>+ H'045C	32
	Current Transaction Byte Register 9/9S	CRTB_9	CRTB_9S	R	H'0000_0000	<Base_NS0> +H'0460	<Base_S0>+ H'0460	32
	Channel Status Register 9/9S	CHSTAT_9	CHSTAT_9S	R	H'0000_0000	<Base_NS0> +H'0464	<Base_S0>+ H'0464	32

Table 14.1 Register Configuration (8/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
9	Channel Control Register 9/9S	CHCTRL_9	CHCTRL_9S	RW	H'0000_0000	<Base_NS0> +H'0468	<Base_S0>+ H'0468	32
	Channel Configuration Register 9/9S	CHCFG_9	CHCFG_9S	RW	H'0000_0000	<Base_NS0> +H'046C	<Base_S0>+ H'046C	32
	Channel Interval Register 9/9S	CHITVL_9	CHITVL_9S	RW	H'0000_0000	<Base_NS0> +H'0470	<Base_S0>+ H'0470	32
	Channel Extension Register 9/9S	CHEXT_9	CHEXT_9S	RW	H'0000_0000	<Base_NS0> +H'0474	<Base_S0>+ H'0474	32
	Next Link Address Register 9/9S	NXLA_9	NXLA_9S	RW	H'0000_0000	<Base_NS0> +H'0478	<Base_S0>+ H'0478	32
	Current Link Address Register 9/9S	CRLA_9	CRLA_9S	R	H'0000_0000	<Base_NS0> +H'047C	<Base_S0>+ H'047C	32
10	Next0 Source Address Register 10/10S	N0SA_10	N0SA_10S	RW	H'0000_0000	<Base_NS0> +H'0480	<Base_S0>+ H'0480	32
	Next0 Destination Address Register 10/10S	N0DA_10	N0DA_10S	RW	H'0000_0000	<Base_NS0> +H'0484	<Base_S0>+ H'0484	32
	Next0 Transaction Byte Register 10/10S	N0TB_10	N0TB_10S	RW	H'0000_0000	<Base_NS0> +H'0488	<Base_S0>+ H'0488	32
	Next1 Source Address Register 10/10S	N1SA_10	N1SA_10S	RW	H'0000_0000	<Base_NS0> +H'048C	<Base_S0>+ H'048C	32
	Next1 Destination Address Register 10/10S	N1DA_10	N1DA_10S	RW	H'0000_0000	<Base_NS0> +H'0490	<Base_S0>+ H'0490	32
	Next1 Transaction Byte Register 10/10S	N1TB_10	N1TB_10S	RW	H'0000_0000	<Base_NS0> +H'0494	<Base_S0>+ H'0494	32
	Current Source Address Register 10/10S	CRSA_10	CRSA_10S	R	H'0000_0000	<Base_NS0> +H'0498	<Base_S0>+ H'0498	32
	Current Destination Address Register 10/10S	CRDA_10	CRDA_10S	R	H'0000_0000	<Base_NS0> +H'049C	<Base_S0>+ H'049C	32
	Current Transaction Byte Register 10/10S	CRTB_10	CRTB_10S	R	H'0000_0000	<Base_NS0> +H'04A0	<Base_S0>+ H'04A0	32
	Channel Status Register 10/10S	CHSTAT_10	CHSTAT_10S	R	H'0000_0000	<Base_NS0> +H'04A4	<Base_S0>+ H'04A4	32
	Channel Control Register 10/10S	CHCTRL_10	CHCTRL_10S	RW	H'0000_0000	<Base_NS0> +H'04A8	<Base_S0>+ H'04A8	32
	Channel Configuration Register 10/10S	CHCFG_10	CHCFG_10S	RW	H'0000_0000	<Base_NS0> +H'04AC	<Base_S0>+ H'04AC	32
	Channel Interval Register 10/10S	CHITVL_10	CHITVL_10S	RW	H'0000_0000	<Base_NS0> +H'04B0	<Base_S0>+ H'04B0	32
	Channel Extension Register 10/10S	CHEXT_10	CHEXT_10S	RW	H'0000_0000	<Base_NS0> +H'04B4	<Base_S0>+ H'04B4	32
	Next Link Address Register 10/10S	NXLA_10	NXLA_10S	RW	H'0000_0000	<Base_NS0> +H'04B8	<Base_S0>+ H'04B8	32
	Current Link Address Register 10/10S	CRLA_10	CRLA_10S	R	H'0000_0000	<Base_NS0> +H'04BC	<Base_S0>+ H'04BC	32

Table 14.1 Register Configuration (9/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
11	Next0 Source Address Register 11/11S	N0SA_11	N0SA_11S	RW	H'0000_0000	<Base_NS0> +H'04C0	<Base_S0>+ H'04C0	32
	Next0 Destination Address Register 11/11S	N0DA_11	N0DA_11S	RW	H'0000_0000	<Base_NS0> +H'04C4	<Base_S0>+ H'04C4	32
	Next0 Transaction Byte Register 11/11S	N0TB_11	N0TB_11S	RW	H'0000_0000	<Base_NS0> +H'04C8	<Base_S0>+ H'04C8	32
	Next1 Source Address Register 11/11S	N1SA_11	N1SA_11S	RW	H'0000_0000	<Base_NS0> +H'04CC	<Base_S0>+ H'04CC	32
	Next1 Destination Address Register 11/11S	N1DA_11	N1DA_11S	RW	H'0000_0000	<Base_NS0> +H'04D0	<Base_S0>+ H'04D0	32
	Next1 Transaction Byte Register 11/11S	N1TB_11	N1TB_11S	RW	H'0000_0000	<Base_NS0> +H'04D4	<Base_S0>+ H'04D4	32
	Current Source Address Register 11/11S	CRSA_11	CRSA_11S	R	H'0000_0000	<Base_NS0> +H'04D8	<Base_S0>+ H'04D8	32
	Current Destination Address Register 11/11S	CRDA_11	CRDA_11S	R	H'0000_0000	<Base_NS0> +H'04DC	<Base_S0>+ H'04DC	32
	Current Transaction Byte Register 11/11S	CRTB_11	CRTB_11S	R	H'0000_0000	<Base_NS0> +H'04E0	<Base_S0>+ H'04E0	32
	Channel Status Register 11/11S	CHSTAT_11	CHSTAT_11S	R	H'0000_0000	<Base_NS0> +H'04E4	<Base_S0>+ H'04E4	32
	Channel Control Register 11/11S	CHCTRL_11	CHCTRL_11S	RW	H'0000_0000	<Base_NS0> +H'04E8	<Base_S0>+ H'04E8	32
	Channel Configuration Register 11/11S	CHCFG_11	CHCFG_11S	RW	H'0000_0000	<Base_NS0> +H'04EC	<Base_S0>+ H'04EC	32
	Channel Interval Register 11/11S	CHITVL_11	CHITVL_11S	RW	H'0000_0000	<Base_NS0> +H'04F0	<Base_S0>+ H'04F0	32
	Channel Extension Register 11/11S	CHEXT_11	CHEXT_11S	RW	H'0000_0000	<Base_NS0> +H'04F4	<Base_S0>+ H'04F4	32
	Next Link Address Register 11/11S	NXLA_11	NXLA_11S	RW	H'0000_0000	<Base_NS0> +H'04F8	<Base_S0>+ H'04F8	32
	Current Link Address Register 11/11S	CRLA_11	CRLA_11S	R	H'0000_0000	<Base_NS0> +H'04FC	<Base_S0>+ H'04FC	32
12	Next0 Source Address Register 12/12S	N0SA_12	N0SA_12S	RW	H'0000_0000	<Base_NS0> +H'0500	<Base_S0>+ H'0500	32
	Next0 Destination Address Register 12/12S	N0DA_12	N0DA_12S	RW	H'0000_0000	<Base_NS0> +H'0504	<Base_S0>+ H'0504	32
	Next0 Transaction Byte Register 12/12S	N0TB_12	N0TB_12S	RW	H'0000_0000	<Base_NS0> +H'0508	<Base_S0>+ H'0508	32
	Next1 Source Address Register 12/12S	N1SA_12	N1SA_12S	RW	H'0000_0000	<Base_NS0> +H'050C	<Base_S0>+ H'050C	32
	Next1 Destination Address Register 12/12S	N1DA_12	N1DA_12S	RW	H'0000_0000	<Base_NS0> +H'0510	<Base_S0>+ H'0510	32
	Next1 Transaction Byte Register 12/12S	N1TB_12	N1TB_12S	RW	H'0000_0000	<Base_NS0> +H'0514	<Base_S0>+ H'0514	32
	Current Source Address Register 12/12S	CRSA_12	CRSA_12S	R	H'0000_0000	<Base_NS0> +H'0518	<Base_S0>+ H'0518	32
	Current Destination Address Register 12/12S	CRDA_12	CRDA_12S	R	H'0000_0000	<Base_NS0> +H'051C	<Base_S0>+ H'051C	32
	Current Transaction Byte Register 12/12S	CRTB_12	CRTB_12S	R	H'0000_0000	<Base_NS0> +H'0520	<Base_S0>+ H'0520	32
	Channel Status Register 12/12S	CHSTAT_12	CHSTAT_12S	R	H'0000_0000	<Base_NS0> +H'0524	<Base_S0>+ H'0524	32

Table 14.1 Register Configuration (10/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
12	Channel Control Register 12/12S	CHCTRL_12	CHCTRL_12S	RW	H'0000_0000	<Base_NS0> +H'0528	<Base_S0>+ H'0528	32
	Channel Configuration Register 12/12S	CHCFG_12	CHCFG_12S	RW	H'0000_0000	<Base_NS0> +H'052C	<Base_S0>+ H'052C	32
	Channel Interval Register 12/12S	CHITVL_12	CHITVL_12S	RW	H'0000_0000	<Base_NS0> +H'0530	<Base_S0>+ H'0530	32
	Channel Extension Register 12/12S	CHEXT_12	CHEXT_12S	RW	H'0000_0000	<Base_NS0> +H'0534	<Base_S0>+ H'0534	32
	Next Link Address Register 12/12S	NXLA_12	NXLA_12S	RW	H'0000_0000	<Base_NS0> +H'0538	<Base_S0>+ H'0538	32
	Current Link Address Register 12/12S	CRLA_12	CRLA_12S	R	H'0000_0000	<Base_NS0> +H'053C	<Base_S0>+ H'053C	32
13	Next0 Source Address Register 13/13S	N0SA_13	N0SA_13S	RW	H'0000_0000	<Base_NS0> +H'0540	<Base_S0>+ H'0540	32
	Next0 Destination Address Register 13/13S	N0DA_13	N0DA_13S	RW	H'0000_0000	<Base_NS0> +H'0544	<Base_S0>+ H'0544	32
	Next0 Transaction Byte Register 13/13S	N0TB_13	N0TB_13S	RW	H'0000_0000	<Base_NS0> +H'0548	<Base_S0>+ H'0548	32
	Next1 Source Address Register 13/13S	N1SA_13	N1SA_13S	RW	H'0000_0000	<Base_NS0> +H'054C	<Base_S0>+ H'054C	32
	Next1 Destination Address Register 13/13S	N1DA_13	N1DA_13S	RW	H'0000_0000	<Base_NS0> +H'0550	<Base_S0>+ H'0550	32
	Next1 Transaction Byte Register 13/13S	N1TB_13	N1TB_13S	RW	H'0000_0000	<Base_NS0> +H'0554	<Base_S0>+ H'0554	32
	Current Source Address Register 13/13S	CRSA_13	CRSA_13S	R	H'0000_0000	<Base_NS0> +H'0558	<Base_S0>+ H'0558	32
	Current Destination Address Register 13/13S	CRDA_13	CRDA_13S	R	H'0000_0000	<Base_NS0> +H'055C	<Base_S0>+ H'055C	32
	Current Transaction Byte Register 13/13S	CRTB_13	CRTB_13S	R	H'0000_0000	<Base_NS0> +H'0560	<Base_S0>+ H'0560	32
	Channel Status Register 13/13S	CHSTAT_13	CHSTAT_13S	R	H'0000_0000	<Base_NS0> +H'0564	<Base_S0>+ H'0564	32
	Channel Control Register 13/13S	CHCTRL_13	CHCTRL_13S	RW	H'0000_0000	<Base_NS0> +H'0568	<Base_S0>+ H'0568	32
	Channel Configuration Register 13/13S	CHCFG_13	CHCFG_13S	RW	H'0000_0000	<Base_NS0> +H'056C	<Base_S0>+ H'056C	32
	Channel Interval Register 13/13S	CHITVL_13	CHITVL_13S	RW	H'0000_0000	<Base_NS0> +H'0570	<Base_S0>+ H'0570	32
	Channel Extension Register 13/13S	CHEXT_13	CHEXT_13S	RW	H'0000_0000	<Base_NS0> +H'0574	<Base_S0>+ H'0574	32
	Next Link Address Register 13/13S	NXLA_13	NXLA_13S	RW	H'0000_0000	<Base_NS0> +H'0578	<Base_S0>+ H'0578	32
	Current Link Address Register 13/13S	CRLA_13	CRLA_13S	R	H'0000_0000	<Base_NS0> +H'057C	<Base_S0>+ H'057C	32

Table 14.1 Register Configuration (11/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
14	Next0 Source Address Register 14/14S	N0SA_14	N0SA_14S	RW	H'0000_0000	<Base_NS0> +H'0580	<Base_S0>+ H'0580	32
	Next0 Destination Address Register 14/14S	N0DA_14	N0DA_14S	RW	H'0000_0000	<Base_NS0> +H'0584	<Base_S0>+ H'0584	32
	Next0 Transaction Byte Register 14/14S	N0TB_14	N0TB_14S	RW	H'0000_0000	<Base_NS0> +H'0588	<Base_S0>+ H'0588	32
	Next1 Source Address Register 14/14S	N1SA_14	N1SA_14S	RW	H'0000_0000	<Base_NS0> +H'058C	<Base_S0>+ H'058C	32
	Next1 Destination Address Register 14/14S	N1DA_14	N1DA_14S	RW	H'0000_0000	<Base_NS0> +H'0590	<Base_S0>+ H'0590	32
	Next1 Transaction Byte Register 14/14S	N1TB_14	N1TB_14S	RW	H'0000_0000	<Base_NS0> +H'0594	<Base_S0>+ H'0594	32
	Current Source Address Register 14/14S	CRSA_14	CRSA_14S	R	H'0000_0000	<Base_NS0> +H'0598	<Base_S0>+ H'0598	32
	Current Destination Address Register 14/14S	CRDA_14	CRDA_14S	R	H'0000_0000	<Base_NS0> +H'059C	<Base_S0>+ H'059C	32
	Current Transaction Byte Register 14/14S	CRTB_14	CRTB_14S	R	H'0000_0000	<Base_NS0> +H'05A0	<Base_S0>+ H'05A0	32
	Channel Status Register 14/14S	CHSTAT_14	CHSTAT_14S	R	H'0000_0000	<Base_NS0> +H'05A4	<Base_S0>+ H'05A4	32
	Channel Control Register 14/14S	CHCTRL_14	CHCTRL_14S	RW	H'0000_0000	<Base_NS0> +H'05A8	<Base_S0>+ H'05A8	32
	Channel Configuration Register 14/14S	CHCFG_14	CHCFG_14S	RW	H'0000_0000	<Base_NS0> +H'05AC	<Base_S0>+ H'05AC	32
	Channel Interval Register 14/14S	CHITVL_14	CHITVL_14S	RW	H'0000_0000	<Base_NS0> +H'05B0	<Base_S0>+ H'05B0	32
	Channel Extension Register 14/14S	CHEXT_14	CHEXT_14S	RW	H'0000_0000	<Base_NS0> +H'05B4	<Base_S0>+ H'05B4	32
	Next Link Address Register 14/14S	NXLA_14	NXLA_14S	RW	H'0000_0000	<Base_NS0> +H'05B8	<Base_S0>+ H'05B8	32
	Current Link Address Register 14/14S	CRLA_14	CRLA_14S	R	H'0000_0000	<Base_NS0> +H'05BC	<Base_S0>+ H'05BC	32
15	Next0 Source Address Register 15/15S	N0SA_15	N0SA_15S	RW	H'0000_0000	<Base_NS0> +H'05C0	<Base_S0>+ H'05C0	32
	Next0 Destination Address Register 15/15S	N0DA_15	N0DA_15S	RW	H'0000_0000	<Base_NS0> +H'05C4	<Base_S0>+ H'05C4	32
	Next0 Transaction Byte Register 15/15S	N0TB_15	N0TB_15S	RW	H'0000_0000	<Base_NS0> +H'05C8	<Base_S0>+ H'05C8	32
	Next1 Source Address Register 15/15S	N1SA_15	N1SA_15S	RW	H'0000_0000	<Base_NS0> +H'05CC	<Base_S0>+ H'05CC	32
	Next1 Destination Address Register 15/15S	N1DA_15	N1DA_15S	RW	H'0000_0000	<Base_NS0> +H'05D0	<Base_S0>+ H'05D0	32
	Next1 Transaction Byte Register 15/15S	N1TB_15	N1TB_15S	RW	H'0000_0000	<Base_NS0> +H'05D4	<Base_S0>+ H'05D4	32
	Current Source Address Register 15/15S	CRSA_15	CRSA_15S	R	H'0000_0000	<Base_NS0> +H'05D8	<Base_S0>+ H'05D8	32
	Current Destination Address Register 15/15S	CRDA_15	CRDA_15S	R	H'0000_0000	<Base_NS0> +H'05DC	<Base_S0>+ H'05DC	32
	Current Transaction Byte Register 15/15S	CRTB_15	CRTB_15S	R	H'0000_0000	<Base_NS0> +H'05E0	<Base_S0>+ H'05E0	32
	Channel Status Register 15/15S	CHSTAT_15	CHSTAT_15S	R	H'0000_0000	<Base_NS0> +H'05E4	<Base_S0>+ H'05E4	32

Table 14.1 Register Configuration (12/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
15	Channel Control Register 15/15S	CHCTRL_15	CHCTRL_15S	RW	H'0000_0000	<Base_NS0> +H'05E8	<Base_S0>+ H'05E8	32
	Channel Configuration Register 15/15S	CHCFG_15	CHCFG_15S	RW	H'0000_0000	<Base_NS0> +H'05EC	<Base_S0>+ H'05EC	32
	Channel Interval Register 15/15S	CHITVL_15	CHITVL_15S	RW	H'0000_0000	<Base_NS0> +H'05F0	<Base_S0>+ H'05F0	32
	Channel Extension Register 15/15S	CHEXT_15	CHEXT_15S	RW	H'0000_0000	<Base_NS0> +H'05F4	<Base_S0>+ H'05F4	32
	Next Link Address Register 15/15S	NXLA_15	NXLA_15S	RW	H'0000_0000	<Base_NS0> +H'05F8	<Base_S0>+ H'05F8	32
	Current Link Address Register 15/15S	CRLA_15	CRLA_15S	R	H'0000_0000	<Base_NS0> +H'05FC	<Base_S0>+ H'05FC	32
Common for 8 to 15	DMA Control Registers 8-15/8-15S	DCTRL_8_15	DCTRL_8_15 S	RW	H'0000_0000	<Base_NS0> +H'0700	<Base_S0>+ H'0700	32
	DMA Status EN Registers 8-15/8-15S	DSTAT_EN_8 _15	DSTAT_EN_8 _15S	R	H'0000_0000	<Base_NS0> +H'0710	<Base_S0>+ H'0710	32
	DMA Status ER Registers 8-15/8-15S	DSTAT_ER_8 _15	DSTAT_ER_8 _15S	R	H'0000_0000	<Base_NS0> +H'0714	<Base_S0>+ H'0714	32
	DMA Status END Registers 8-15/8-15S	DSTAT_END _8_15	DSTAT_END _8_15S	R	H'0000_0000	<Base_NS0> +H'0718	<Base_S0>+ H'0718	32
	DMA Status TC Registers 8-15/8-15S	DSTAT_TC_8 _15	DSTAT_TC_8 _15S	R	H'0000_0000	<Base_NS0> +H'071C	<Base_S0>+ H'071C	32
	DMA Status SUS Registers 8-15/8-15S	DSTAT_SUS _8_15	DSTAT_SUS _8_15S	R	H'0000_0000	<Base_NS0> +H'0720	<Base_S0>+ H'0720	32

(2) APB I/F

Base Address Name	Base Address	
<Base_S1>	H'0_1181_0000	(Cortex-A55 Address Space)
	H'4181_0000	(Cortex-M33/Cortex-M33_FPU Address Space Secure)
	H'5181_0000	(Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)
<Base_NS1>	H'0_1183_0000	(Cortex-A55 Address Space)
	H'4183_0000	(Cortex-M33/Cortex-M33_FPU Address Space Secure)
	H'5183_0000	(Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Note: Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 14.2 Register Configuration

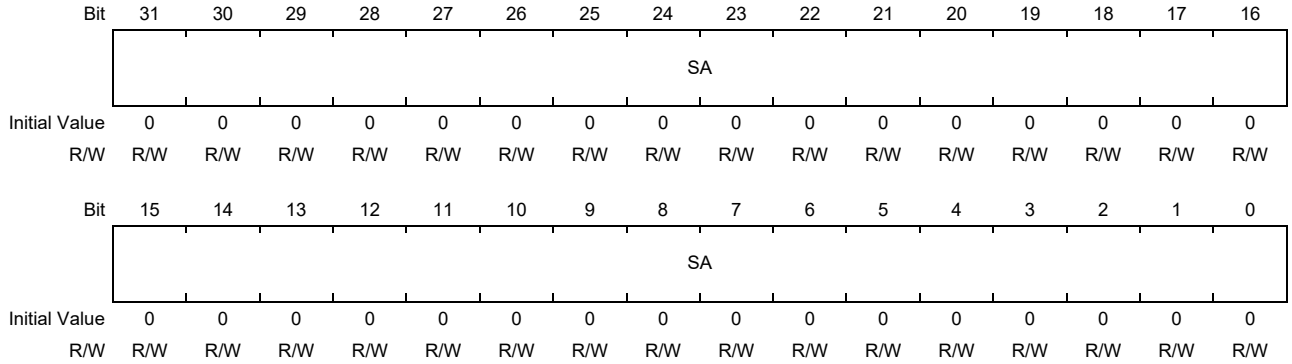
Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
0/1	DMA Extension Resource Selectors 0/0S	DMARS0	DMARS0S	RW	H'0000_0000	<Base_NS1> +H'0000	<Base_S1>+ H'0000	32
2/3	DMA Extension Resource Selectors 1/1S	DMARS1	DMARS1S	RW	H'0000_0000	<Base_NS1> +H'0004	<Base_S1>+ H'0004	32
4/5	DMA Extension Resource Selectors 2/2S	DMARS 2	DMARS 2S	RW	H'0000_0000	<Base_NS1> +H'0008	<Base_S1>+ H'0008	32
6/7	DMA Extension Resource Selectors 3/3S	DMARS 3	DMARS 3S	RW	H'0000_0000	<Base_NS1> +H'000C	<Base_S1>+ H'000C	32
8/9	DMA Extension Resource Selectors 4/4S	DMARS 4	DMARS 4S	RW	H'0000_0000	<Base_NS1> +H'0010	<Base_S1>+ H'0010	32
10/11	DMA Extension Resource Selectors 5/5S	DMARS 5	DMARS 5S	RW	H'0000_0000	<Base_NS1> +H'0014	<Base_S1>+ H'0014	32
12/13	DMA Extension Resource Selectors 6/6S	DMARS 6	DMARS 6S	RW	H'0000_0000	<Base_NS1> +H'0018	<Base_S1>+ H'0018	32
14/15	DMA Extension Resource Selectors 7/7S	DMARS 7	DMARS 7S	RW	H'0000_0000	<Base_NS1> +H'001C	<Base_S1>+ H'001C	32

14.4.1 Next Source Address Register n/nS (N0SA_n/nS, N1SA_n/nS)

This register sets the DMA transfer source address (32 bits) of DMA channel n (n = 0 to 15) which is to be executed next. N0SA_n/nS is for the Next0 Register Set, and N1SA_n/nS is for the Next1 Register Set.

In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.

These register set values are loaded to the Current Register Set and used for DMA transfer.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SA	All 0	R/W	Source Address Sets the start address of the DMA transfer source.

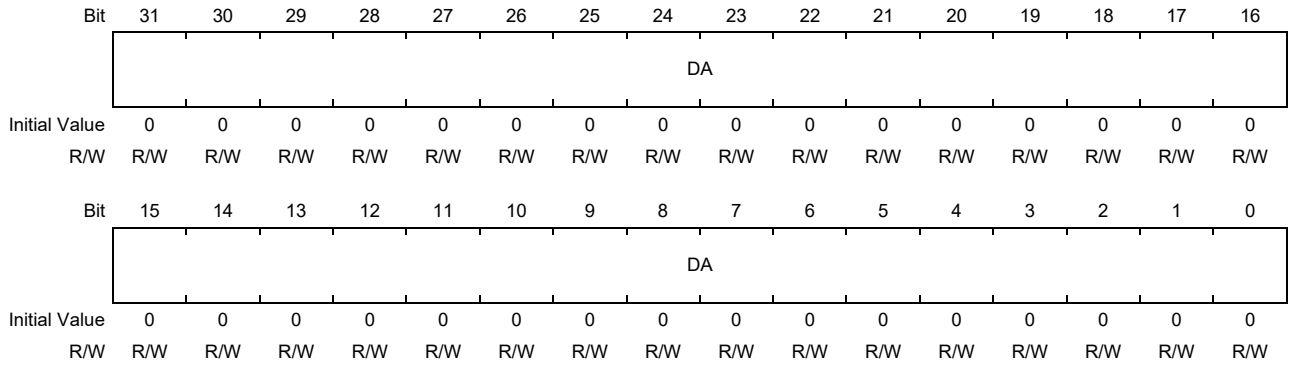
14.4.2 Next Destination Address Register n/nS (N0DA_n/nS, N1DA_n/nS)

This register sets the DMA transfer destination address (32 bits) of DMA channel n (n = 0 to 15) which is to be executed next.

N0DA_n/nS is for the Next0 Register Set, and N1DA_n/nS is for the Next1 Register Set.

In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.

These register set values are loaded to the Current Register Set and used for DMA transfer.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DA	All 0	R/W	Destination Address Sets the start address of the DMA transfer destination.

14.4.3 Next Transaction Byte Register n/nS (N0TB_n/nS, N1TB_n/nS)

This register sets the total transfer byte count (DMA transaction) of DMA channel (n = 0 to 15) which is to be executed next.

N0TB_n/nS is for the Next0 Register Set, and N1TB_n/nS is for the Next1 Register Set.

In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.

These register set values are loaded to the Current Register Set and used for DMA transfer.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TB															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TB															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TB	All 0	R/W	Transaction Byte Sets the total transfer byte count. *Note*: Do not start a DMA transaction with 0 set in this register.

14.4.4 Current Source Address Register n/nS (CRSA_n/nS)

This register indicates the DMA transfer source address of DMA channel n (n = 0 to 15).

The values are loaded from the Next0/1 register set in register mode or from the descriptor read data in link mode. This register cannot be written by software.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRSA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRSA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRSA	All 0	R	Current Source Address Register Indicates the read address of the next DMA transaction. The value automatically increments during the DMA transaction. (The value is fixed when 1 is set in SAD of the CHCFG_n/nS register.) The value increments when a read transfer starts. Read this register after DMA stops (0 is set in EN of the CHSTAT_n/nS register). (Any value obtained during the DMA operation should be handled as a reference value.)

14.4.5 Current Destination Address Register n/nS (CRDA_n/nS)

This register indicates the DMA transfer destination address of DMA channel n (n = 0 to 15).

The values are loaded from the Next0/1 register set in register mode or from the descriptor read data in link mode. This register cannot be written by software.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRDA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRDA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRDA	All 0	R	Current Destination Address Register Indicates the write address of the next DMA transaction. The value automatically increments during the DMA transaction. (The value is fixed when 1 is set in DAD of the CHCFG_n/nS register.) The value increments when a write transfer starts. Read this register after DMA stops (0 is set in EN of the CHSTAT_n/nS register). (Any value obtained during the DMA operation should be handled as a reference value.)

14.4.6 Current Transaction Byte Register n/nS (CRTB_n/nS)

This register indicates the total transfer byte count of DMA channel n (n = 0 to 15). The value of this register becomes 0 when the transaction ends.

The values are loaded from the Next0/1 register set in register mode or from the descriptor read data in link mode. This register cannot be written by software.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRTB															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRTB															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRTB	All 0	R	Current Transaction Byte Register Indicates the remaining transfer byte count of the currently executed DMA transaction. The value automatically decrements during the DMA transaction. The value decrements when a write transfer is completed. Read this register after DMA stops (0 is set in EN of the CHSTAT_n/nS register). (Any value obtained during the DMA operation should be handled as a reference value.)

14.4.7 Channel Status Register n/nS (CHSTAT_n/nS)

This register indicates the status of DMA channel n (n = 0 to 15).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INTMSK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MODE	DER	DW	DL	SR	TC	END	ER	SUS	TACT	RQST	EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
16	INTMSK	0	R	Indicates the temporary mask status of the DMA transfer end interrupt. 1: Masked temporarily 0: Unmasked temporarily Set condition(s): • When SETINTMSK (CHCTRL_n/nS) is set to 1 Reset condition(s): • When CLRINTMSK (CHCTRL_n/nS) is set to 1 • When SWRST (CHCTRL_n/nS) is set to 1
15 to 12	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
11	MODE	0	R	DMA Mode Indicates the DMA mode. It corresponds to the value set in the DMS bit of the CHCFG_n/nS register. 0: Register mode 1: Link mode
10	DER	0	R	Descriptor Error Indicates whether the link valid value of the read descriptor is invalid (LV = 0) (this is not dependent on the DIM level of the descriptor). If a descriptor error has occurred, the transfer is stopped but no DMA error interrupt occurs. 0: Descriptor Error not detected 1: Descriptor Error detected Set condition(s): • When the LV value loaded with the descriptor in link mode is 0 Reset condition(s): • When SWRST (CHCTRL_n/nS) is set to 1
9	DW	0	R	Descriptor WriteBack Indicates the descriptor writeback status. The bit maintains 1 if a bus error is received during descriptor writeback. 0: Operation other than writeback is being performed for the header in link mode. 1: (ER = 0) Writeback is being performed for the header in link mode. (ER = 1) A bus error occurs during writeback for the header in link mode. Set condition(s): • When header writeback in link mode starts Reset condition(s): • When header writeback in link mode ends with an OK response • When SWRST (CHCTRL_n/nS) is set to 1

Bit	Bit Name	Initial Value	R/W	Description
8	DL	0	R	Descriptor Load Indicates whether the descriptor is being loaded. The bit maintains 1 if a bus error is received during descriptor load. 0: Operation other than descriptor load 1: (ER = 0) Descriptor load is in progress in link mode. (ER = 1) A bus error occurs during descriptor load in link mode. Set condition(s): • When descriptor load in link mode starts Reset condition(s): • When descriptor load in link mode ends with an OK response • When SWRST (CHCTRL_n/nS) is set to 1
7	SR	0	R	Selected Register Set Indicates the register set currently selected in register mode. 0: Next0 Register Set 1: Next1 Register Set Set condition(s): • When RSEL (CHCFG_n/nS) is set to 1 Reset condition(s): • When RSEL (CHCFG_n/nS) is set to 0
6	TC	0	R	Terminal Count Indicates whether the DMA transaction is completed. 0: DMA transfer not completed 1: DMA transfer completed Set condition(s): • When data equivalent to the total transfer byte count set in the CRTB register has been transferred in register mode • When data equivalent to the total transfer byte count set in the CRTB register has been transferred in link mode, with 1 set in WBD of the descriptor header • When descriptor writeback is completed in link mode, with 0 set in WBD of the descriptor header Clear condition(s): • When the CLRTC (CHCTRL_n/nS) bit is set to 1 • When the SWRST (CHCTRL_n/nS) bit is set to 1
5	END	0	R	DMAEND Interrupted Indicates whether the DMA transaction is completed and whether the DMA transfer end interrupt has occurred. 0: DMA transfer not completed 1: DMA transfer completed Set condition(s): • When one of the set conditions for the TC bit is met and 0 is set in DEM of the CHCFG_n/nS register • When the descriptor is read in link mode and both LV of the header and DIM are set to 0 Clear condition(s): • When CLREND (CHCTRL_n/nS) is set to 1 • When SWRST (CHCTRL_n/nS) is set to 1

Bit	Bit Name	Initial Value	R/W	Description
4	ER	0	R	Error bit Indicates that a DMA error interrupt has occurred because an error response has been received from the transfer source or destination and a bus error has occurred during the DMA transfer. 0: No bus error has occurred 1: A DMA error interrupt has occurred due to a bus error Set condition(s): • When a bus error has occurred during a bus cycle Clear condition(s): • When SWRST (CHCTRL_n/nS) is set to 1
3	SUS	0	R	Suspend Indicates whether the channel is suspended. 0: Channel_n not suspended 1: Channel_n suspended Set condition(s): • When SETSUS (CHCTRL_n/nS) is set to 1 during a DMA transfer on Channel_n, creating a SUSPEND status internally Clear condition(s): • When CLRSUS (CHCTRL_n/nS) is set to 1 • When CLREN (CHCTRL_n/nS) is set to 1
2	TACT	0	R	Transaction Active Indicates whether the DMAC is active. This bit is intended to check that the channel is completely inactive. 0: DMA on Channel_n inactive 1: DMA on Channel_n active Set condition(s): • When a DMA transaction starts on Channel_n Clear condition(s): • When a DMA transaction is completed
1	RQST	0	R	Request Indicates whether a transfer request is being received. 0: DMA transfer request not being received 1: DMA transfer request being received Set condition(s): • When the STG bit (CHCTRL_n/nS) is set to 1 (auto request) • When a transfer request is received from the DMA request source set in the CHCFG_n/nS register Clear condition(s): • When SWRST (CHCTRL_n/nS) is set to 1 • When CLRRQ (CHCTRL_n/nS) is set to 1 • When a transfer is executed on the side specified by REQD (CHCFG_n/nS) in single transfer mode (TM = 0). • When all DMA transactions are completed in register mode (the transaction ends with REN set to 0) • When the DMA transfer of the last descriptor (LE = 1) is completed in link mode • When descriptor read stops (LV = 0) in link mode • When a bus error is received due to an error response

Bit	Bit Name	Initial Value	R/W	Description
0	EN	0	R	Enable Indicates whether the operation of DMA channel n is enabled or disabled. 0: Operation disabled 1: Operation enabled Set condition(s): • When SETEN (CHCTRL_n/nS) is set to 1 Clear condition(s): • When SWRST (CHCTRL_n/nS) is set to 1 • When CLREN (CHCTRL_n/nS) is set to 1 • When a bus error is received due to an error response during the transfer • When all DMA transactions are completed in register mode (the transaction ends with REN set to 0) • When the DMA transfer of the last descriptor (LE = 1) is completed in link mode (writeback when WBD is set to 0) • When descriptor read stops (LV = 0) in link mode

CAUTION

If the ER bit is set to 1 for any transfer, the whole transfer should be handled as invalid.

To suspend a DMA transaction, mask or clear the transfer request or clear the Enable bit (for the procedure, see **Section 14.7.8(2), Transfer Stop**).

If a transfer request from an on-chip peripheral module input is made concurrently with an auto request (by setting 1 in the STG bit) for the same one channel, the trigger source that takes effect cannot be identified. Make sure that only one of these transfer requests is used in the system.

When transfer is requested by an auto request, wait for the last requested DMA transfer to complete (use the Current Register or other data to check the status) before setting the STG bit for the next transfer request.

14.4.8 Channel Control Register n/nS (CHCTRL_n/nS)

This register controls the DMA transfer operation on DMA channel n (n = 0 to 15).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLRINTMSK	SETINTMSK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLRSUS	SETSUS	—	CLRTC	CLRENDD	CLRRQ	SWRST	STG	CLREN	SETEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
17	CLRINTMSK	0	R/W	When this bit is set to 1, the mask of the DMA transfer end interrupt is cleared. Also, the INTMSK bit of the CHSTAT_n/nS register is set to 0. If the mask is cleared when 1 is set in both LVINT of the DCTRL register and END of the CHSTAT_n/nS register, the DMA transfer end interrupt becomes active. (It does not become active when 0 is set in LVINT.) A read operation results in 0 being read. 1: Clears the mask set by SETINTMSK. 0: Does not affect the operation.
16	SETINTMSK	0	R/W	When this bit is set to 1, the DMA transfer end interrupt is temporarily masked. Also, the INTMSK bit of the CHSTAT_n/nS register is set to 1. A read operation results in 0 being read. 1: Masks the DMA transfer end interrupt. 0: Does not affect the operation.
15 to 10	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
9	CLRSUS	0	R/W	Clear Suspend Clears the suspend status. Setting this bit to 1 when 1 is set in SUS of the CHSTAT_n/nS register can clear the suspend status. An attempt to read this bit results in 0 being read. 1: Clears the suspend status of the current DMA transfer. 0: Does not affect the operation.
8	SETSUS	0	R/W	Set Suspend Suspends the current DMA transfer. Setting this bit to 1 when 1 is set in EN of the CHSTAT_n/nS register can suspend the current DMA transfer. An attempt to read this bit results in 0 being read. 1: Suspends the current DMA transfer. 0: Does not affect the operation.
7	—	0	R	Reserved. Set 0. A read operation results in 0 being read.
6	CLRTC	0	R/W	Clear TC bit Setting this bit to 1 can clear the TC bit of the CHSTAT_n/nS register. An attempt to read this bit results in 0 being read. 1: Clears the TC bit. 0: Does not affect the operation.

Bit	Bit Name	Initial Value	R/W	Description
5	CLREND	0	R/W	Clear End bit Setting this bit to 1 can clear the END bit of the CHSTAT_n/nS register. Also, the DMA transfer end interrupt is cleared. An attempt to read this bit results in 0 being read. 1: Clears the END bit. 0: Does not affect the operation.
4	CLRRQ	0	R/W	Clear Request bit Setting this bit to 1 can clear the RQST bit of the CHSTAT_n/nS register. An attempt to read this bit results in 0 being read. 1: Clears the RQST bit. 0: Does not affect the operation.
3	SWRST	0	R/W	Software Reset Setting this bit to 1 can clear the channel status register (CHSTAT_n/nS). When setting this bit to 1, make sure that both the EN bit and TACT bit are set to 0. An attempt to read this bit results in 0 being read. 1: Resets the channel status register. 0: Does not affect the operation.
2	STG	0	R/W	Software Trigger Setting this bit to 1 sets an auto request. If this bit is set at the same time the SWRST bit is set, the clear operation by the SWRST bit takes precedence. An attempt to read this bit results in 0 being read. 1: Sets a transfer request triggered by an auto request (sets 1 in the RQST bit). 0: Does not affect the operation.
1	CLREN	0	R/W	Clear Enable Setting this bit to 1 can clear the EN bit (for details, see **Section 14.7.8(2), Transfer Stop**). An attempt to read this bit results in 0 being read. 1: Stops the DMA transfer (clears the EN bit). 0: Does not affect the operation.
0	SETEN	0	R/W	Set Enable Enables a DMA transfer on DMA channel n. If this bit is set at the same time the SWRST bit is set, the clear operation by the SWRST bit takes precedence and the transfer does not start. An attempt to read this bit results in 0 being read. 1: Enables a DMA transfer (sets 1 in the EN bit). 0: Does not affect the operation.

14.4.9 Channel Configuration Register n/nS (CHCFG_n/nS)

This register controls the DMA transfer operation on DMA channel n (n = 0 to 15).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMS	REN	RSW	RSEL	SBE	—	—	DEM	—	TM	DAD	SAD	DDS[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDS[3:0]				—	AM[2:0]			—	LVL	HIEN	LOEN	REQD	SEL[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	DMS	0	R/W	DMA Mode Select Sets the DMA mode. 0: Register mode (initial value) 1: Link mode
30	REN	0	R/W	Register Set Enable After a DMA transaction is completed, DMA transfers are continued using the Next register set selected by RSEL. This bit is valid only in register mode. 0: Does not continue DMA transfers. 1: Continues DMA transfers. Set condition(s): • When 1 is written to this bit Clear condition(s): • When 0 is written to this bit • When a DMA transaction is completed, with REN set to 1
29	RSW	0	R/W	Register Select Switch Inverts RSEL automatically after a DMA transaction is completed. This bit is valid only in register mode. 0: Does not invert RSEL automatically after a DMA transaction (initial value). 1: Inverts RSEL automatically after a DMA transaction.
28	RSEL	0	R/W	Register Set Select Selects the Next register set to be executed next. This bit is valid only in register mode. When RSW is set to 1, this bit is inverted automatically when a DMA transaction is completed. 0: Executes the Next0 Register Set (initial value). 1: Executes the Next1 Register Set. Transition condition(s): • When a DMA transaction is completed, with RSW set to 1
27	SBE	0	R/W	Sweep Buffer Enable Selects whether to sweep (write) the data already read into the buffer and stop the DMA transfer if the Enable bit is cleared to 0 during a DMA transaction. The sweep mode is available only when REQD is set to 0. 0: Stops the DMA transfer without sweeping the buffer (initial value). 1: Stops the DMA transfer after sweeping the buffer.
26, 25	—	00	R	Reserved. Set 0. A read operation results in 0 being read.

Bit	Bit Name	Initial Value	R/W	Description
24	DEM	0	R/W	DMA Transfer End Interrupt Mask Masks the DMA transfer end interrupt for register mode transfer. If 1 is set in this bit when a DMA transfer end interrupt is output, the DMA transfer end interrupt signal is not asserted. In this case, DEM is cleared to 0 automatically. 0: Does not mask the DMA transfer end interrupt (initial value). 1: Masks the DMA transfer end interrupt. Clear condition(s): • When a DMA transaction is completed with DEM set to 1
23	—	0	R	Reserved. Set 0. A read operation results in 0 being read.
22	TM	0	R/W	Transfer Mode Sets the DMA transfer mode. 0: Single transfer mode (initial value) 1: Block transfer mode
21	DAD	0	R/W	Sets the destination address counting direction of DMA channel n. 0: Increment (initial value) 1: Fixed
20	SAD	0	R/W	Sets the source address counting direction of DMA channel n. 0: Increment (initial value) 1: Fixed
19 to 16	DDS[3:0]	0000	R/W	Destination Data Size Sets the DMA transfer size of the transfer destination.

Value	Size	Remark
0000	8 bits	Initial value
0001	16 bits	
0010	32 bits	
0011	64 bits	
0100	128 bits	
0101	256 bits	
0110	512 bits	
0111	1024 bits	
Other than the above	—	Setting prohibited

15 to 12	SDS[3:0]	0000	R/W	Source Data Size Sets the DMA transfer size of the transfer source.

Value	Size	Remark
0000	8 bits	Initial value
0001	16 bits	
0010	32 bits	
0011	64 bits	
0100	128 bits	
0101	256 bits	
0110	512 bits	
0111	1024 bits	
Other than the above	—	Setting prohibited

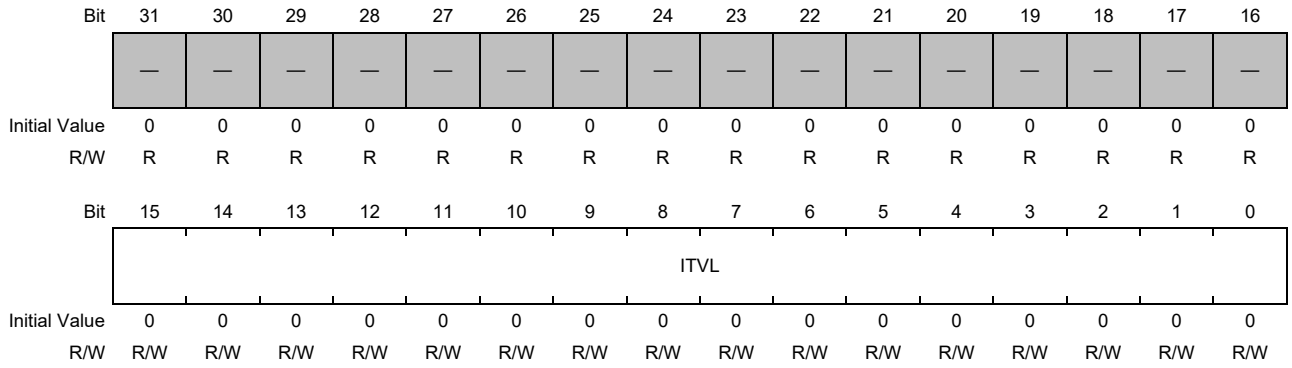
11	—	0	R	Reserved. Set 0. A read operation results in 0 being read.
10 to 8	AM[2:0]	000	R/W	ACK Mode Sets the DMAACK output mode. 000: (initial value) 001: Level mode (active until the transfer request from an on-chip peripheral module input becomes inactive) 01x: Bus cycle mode (active while the DMA transfer is in a bus cycle) 1xx: DMAACK not to be output (this setting should be made when an auto request is made by STG (CHCTRL_n/nS) or when the SCIFA transfer is performed)

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved. Set 0. A read operation results in 0 being read.
6	LVL	0	R/W	Level Selects whether to detect a DMA request based on the level or edge of the signal. 0: Detects based on the edge (initial value). 1: Detects based on the level.
5	HIEN	0	R/W	High Enable Selects whether to detect a DMA request using the High level or rising edge of the signal. When LVL = 0: HIEN = 1: Detects a request in response to the rising edge of the signal. HIEN = 0: Does not detect a request in response to the rising edge of the signal (initial value). When LVL = 1: HIEN = 1: Detects a request when the signal is at the High level. HIEN = 0: Does not detect a request even when the signal is at the High level (initial value).
4	LOEN	0	R/W	Low Enable Selects whether to detect a DMA request using the Low level or falling edge of the signal. When LVL = 0: LOEN = 1: Detects a request in response to the falling edge of the signal. LOEN = 0: Does not detect a request in response to the falling edge of the signal (initial value). When LVL = 1: LOEN = 1: Detects a request when the signal is at the Low level. LOEN = 0: Does not detect a request even when the signal is at the Low level (initial value).
3	REQD	0	R/W	Request Direction Selects whether DMAREQ selected by the SEL bit is the source or destination. This bit is also used to define when DMAACK is to become active. 0: Source; DMAACK is to become active when read (initial value). 1: Destination; DMAACK is to become active when written.
2 to 0	SEL[2:0]	000	R/W	These bits are used to set a DMAC channel. Set one of the following values so that the channel set by the SEL bits matches the CHCFG_n/nS channel. 000: CH0/CH8 001: CH1/CH9 010: CH2/CH10 011: CH3/CH11 100: CH4/CH12 101: CH5/CH13 110: CH6/CH14 111: CH7/CH15

14.4.10 Channel Interval Register n/nS (CHITVL_n/nS)

This register sets the transfer interval for DMA channel n (n = 0 to 15).

For details, see **Section 14.7.6, Interval Count Function**.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
15 to 0	ITVL	All 0	R/W	Sets the channel transfer interval.

14.4.11 Channel Extension Register n/nS (CHEXT_n/nS)

This is an extension register for DMA channel n (n = 0 to 15).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCA[3:0]				—	DPR[2:0]			SCA[3:0]				—	SPR[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Set 0. A read operation results in 0 being read.
15 to 12	DCA[3:0]	0000	R/W	Destination CACHE Sets the value to be output to AWCACHE[3:0] for DMA write transfer. See Note 1 below.
11	—	0	R	Set 0. A read operation results in 0 being read.
10 to 8	DPR[2:0]	000	R/W	Destination PROT Sets the value to be output to AWPROT[2:0] for DMA write transfer. In the case of non-secure access, the value output as AWPROT[1] is fixed to 1 regardless of the setting of the DPR[1] bit. See Note 2 below.
7 to 4	SCA[3:0]	0000	R/W	Source CACHE Sets the value to be output to ARCACHE[3:0] for DMA read transfer. See Note 1 below.
3	—	0	R	Set 0. A read operation results in 0 being read.
2 to 0	SPR[2:0]	000	R/W	Source PROT Sets the value to be output to ARPROT[2:0] for DMA read transfer. In the case of non-secure access, the value output as ARPROT[1] is fixed to 1 regardless of the setting of the SPR[1] bit. See Note 2 below.

Note 1. Cache support: Bits SCA and DCA are used to change the settings.

Note 2. Protection unit support: Bits SPR and DPR are used to change the settings.
For the setting value, see AMBA AXI Protocol Specification from Arm Limited.

14.4.12 Next Link Address Register n/nS (NXLA_n/nS)

This is a 32-bit register that sets the link address of DMA channel n (n = 0 to 15).

For information about the link mode, see **Section 14.6.3, Link Mode**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NXLA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NXLA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NXLA	All 0	R/W	Sets a link address. The low-order 2 bits are masked with 0s. Only an address aligned with a 4-byte boundary can be set.

14.4.13 Current Link Address Register n/nS (CRLA_n/nS)

This is a 32-bit register that indicates the link address of DMA channel n (n = 0 to 15).

For information about the link mode, see **Section 14.6.3, Link Mode**.

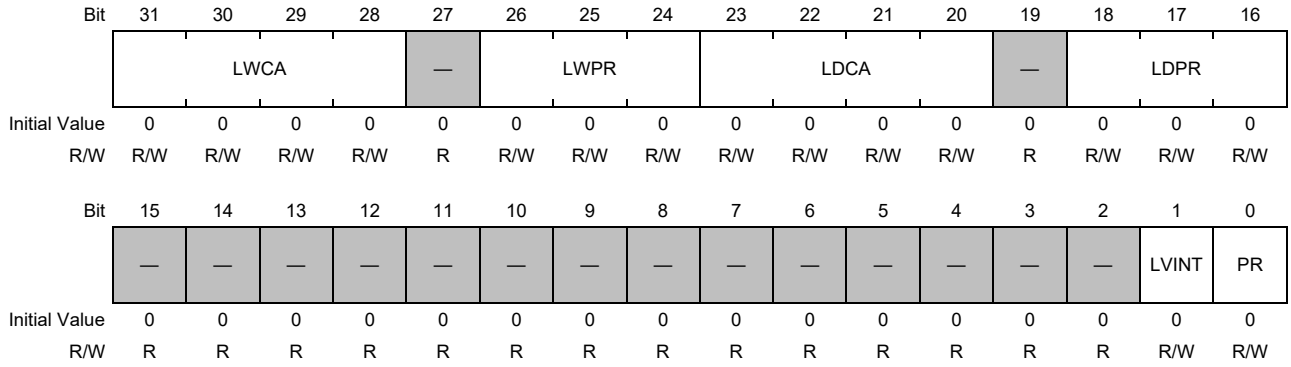
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRLA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRLA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRLA	All 0	R	Indicates the address of the currently executed descriptor.

14.4.14 DMA Control Register (DCTRL_0_7/0_7S, DCTRL_8_15/8_15S)

This register sets the transfer type for descriptor access and the arbitration between channels.
(DCTRL_0_7/0_7S is common for channels 0 to 7 and DCTRL_8_15/8_15S is common for channels 8 to 15.)



Bit	Bit Name	Initial Value	R/W	Description
31 to 28	LWCA	0000	R/W	Link WriteBack CACHE Sets the value to be output to AWCACHE[3:0] during descriptor writeback in link mode. For the setting value, see Note in **Section 14.4.11, Channel Extension Register n/nS (CHEXT_n/nS)**.
27	—	0	R	Reserved. Set 0. The initial value is 0.
26 to 24	LWPR	000	R/W	Link WriteBack PROT Sets the value to be output to AWPROT[2:0] during descriptor writeback in link mode. In the case of non-secure access, the value output as AWPROT[1] is fixed to 1 regardless of the setting of the LWPR[1] bit. For the setting value, see AMBA AXI Protocol Specification from Arm Limited.
23 to 20	LDCA	0000	R/W	Link Descriptor CACHE Sets the value to be output to ARCACHE[3:0] during descriptor load in link mode. For the setting value, see Note in **Section 14.4.11, Channel Extension Register n/nS (CHEXT_n/nS)**.
19	—	0	R	Reserved. Set 0. The initial value is 0.
18 to 16	LDPR	000	R/W	Link Descriptor PROT Sets the value to be output to ARPROT[2:0] during descriptor load in link mode. In the case of non-secure access, the value output as ARPROT[1] is fixed to 1 regardless of the setting of the LDPR[1] bit. For the setting value, see AMBA AXI Protocol Specification from Arm Limited.
15 to 2	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
1	LVINT	0	R/W	Sets whether to use pulse output or level output for the DMA transfer end interrupt and DMA error interrupt. Set pulse output for this product. 0: Pulse output (initial value) 1: Level output
0	PR	0	R/W	Sets the transfer priority control mode between channels (see **Section 14.7.2, Priority Control for DMA Channels**). 0: Fixed priority mode (initial value) 1: Round robin mode

14.4.15 DMA Status EN Register (DSTAT_EN_0_7/0_7S)

This register indicates the EN bit status of the CHSTAT_n/nS register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	EN7	0	R	Indicates the EN bit status of DMA channel 7.
6	EN6	0	R	Indicates the EN bit status of DMA channel 6.
5	EN5	0	R	Indicates the EN bit status of DMA channel 5.
4	EN4	0	R	Indicates the EN bit status of DMA channel 4.
3	EN3	0	R	Indicates the EN bit status of DMA channel 3.
2	EN2	0	R	Indicates the EN bit status of DMA channel 2.
1	EN1	0	R	Indicates the EN bit status of DMA channel 1.
0	EN0	0	R	Indicates the EN bit status of DMA channel 0.

14.4.16 DMA Status EN Register (DSTAT_EN_8_15/8_15S)

This register indicates the EN bit status of the CHSTAT_n/nS register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	EN15	EN14	EN13	EN12	EN11	EN10	EN9	EN8
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	EN15	0	R	Indicates the EN bit status of DMA channel 15.
6	EN14	0	R	Indicates the EN bit status of DMA channel 14.
5	EN13	0	R	Indicates the EN bit status of DMA channel 13.
4	EN12	0	R	Indicates the EN bit status of DMA channel 12.
3	EN11	0	R	Indicates the EN bit status of DMA channel 11.
2	EN10	0	R	Indicates the EN bit status of DMA channel 10.
1	EN9	0	R	Indicates the EN bit status of DMA channel 9.
0	EN8	0	R	Indicates the EN bit status of DMA channel 8.

14.4.17 DMA Status ER Register (DSTAT_ER_0_7/0_7S)

This register indicates the ER bit status of the CHSTAT_n/nS register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	ER7	0	R	Indicates the ER bit status of DMA channel 7.
6	ER6	0	R	Indicates the ER bit status of DMA channel 6.
5	ER5	0	R	Indicates the ER bit status of DMA channel 5.
4	ER4	0	R	Indicates the ER bit status of DMA channel 4.
3	ER3	0	R	Indicates the ER bit status of DMA channel 3.
2	ER2	0	R	Indicates the ER bit status of DMA channel 2.
1	ER1	0	R	Indicates the ER bit status of DMA channel 1.
0	ER0	0	R	Indicates the ER bit status of DMA channel 0.

14.4.18 DMA Status ER Register (DSTAT_ER_8_15/8_15S)

This register indicates the ER bit status of the CHSTAT_n/nS register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ER15	ER14	ER13	ER12	ER11	ER10	ER9	ER8
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	ER15	0	R	Indicates the ER bit status of DMA channel 15.
6	ER14	0	R	Indicates the ER bit status of DMA channel 14.
5	ER13	0	R	Indicates the ER bit status of DMA channel 13.
4	ER12	0	R	Indicates the ER bit status of DMA channel 12.
3	ER11	0	R	Indicates the ER bit status of DMA channel 11.
2	ER10	0	R	Indicates the ER bit status of DMA channel 10.
1	ER9	0	R	Indicates the ER bit status of DMA channel 9.
0	ER8	0	R	Indicates the ER bit status of DMA channel 8.

14.4.19 DMA Status END Register (DSTAT_END_0_7/0_7S)

This register indicates the END bit status of the CHSTAT_n/nS register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	END7	END6	END5	END4	END3	END2	END1	END0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	END7	0	R	Indicates the END bit status of DMA channel 7.
6	END6	0	R	Indicates the END bit status of DMA channel 6.
5	END5	0	R	Indicates the END bit status of DMA channel 5.
4	END4	0	R	Indicates the END bit status of DMA channel 4.
3	END3	0	R	Indicates the END bit status of DMA channel 3.
2	END2	0	R	Indicates the END bit status of DMA channel 2.
1	END1	0	R	Indicates the END bit status of DMA channel 1.
0	END0	0	R	Indicates the END bit status of DMA channel 0.

14.4.20 DMA Status END Register (DSTAT_END_8_15/8_15S)

This register indicates the END bit status of the CHSTAT_n/nS register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	END15	END14	END13	END12	END11	END10	END9	END8
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	END15	0	R	Indicates the END bit status of DMA channel 15.
6	END14	0	R	Indicates the END bit status of DMA channel 14.
5	END13	0	R	Indicates the END bit status of DMA channel 13.
4	END12	0	R	Indicates the END bit status of DMA channel 12.
3	END11	0	R	Indicates the END bit status of DMA channel 11.
2	END10	0	R	Indicates the END bit status of DMA channel 10.
1	END9	0	R	Indicates the END bit status of DMA channel 9.
0	END8	0	R	Indicates the END bit status of DMA channel 8.

14.4.21 DMA Status TC Register (DSTAT_TC_0_7/0_7S)

This register indicates the TC bit status of the CHSTAT_n/nS register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	TC7	0	R	Indicates the TC bit status of DMA channel 7.
6	TC6	0	R	Indicates the TC bit status of DMA channel 6.
5	TC5	0	R	Indicates the TC bit status of DMA channel 5.
4	TC4	0	R	Indicates the TC bit status of DMA channel 4.
3	TC3	0	R	Indicates the TC bit status of DMA channel 3.
2	TC2	0	R	Indicates the TC bit status of DMA channel 2.
1	TC1	0	R	Indicates the TC bit status of DMA channel 1.
0	TC0	0	R	Indicates the TC bit status of DMA channel 0.

14.4.22 DMA Status TC Register (DSTAT_TC_8_15/8_15S)

This register indicates the TC bit status of the CHSTAT_n/nS register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TC15	TC14	TC13	TC12	TC11	TC10	TC9	TC8
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	TC15	0	R	Indicates the TC bit status of DMA channel 15.
6	TC14	0	R	Indicates the TC bit status of DMA channel 14.
5	TC13	0	R	Indicates the TC bit status of DMA channel 13.
4	TC12	0	R	Indicates the TC bit status of DMA channel 12.
3	TC11	0	R	Indicates the TC bit status of DMA channel 11.
2	TC10	0	R	Indicates the TC bit status of DMA channel 10.
1	TC9	0	R	Indicates the TC bit status of DMA channel 9.
0	TC8	0	R	Indicates the TC bit status of DMA channel 8.

14.4.23 DMA Status SUS Register (DSTAT_SUS_0_7/0_7S)

This register indicates the SUS bit status of the CHSTAT_n/nS register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SUS7	SUS6	SUS5	SUS4	SUS3	SUS2	SUS1	SUS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	SUS7	0	R	Indicates the SUS bit status of DMA channel 7.
6	SUS6	0	R	Indicates the SUS bit status of DMA channel 6.
5	SUS5	0	R	Indicates the SUS bit status of DMA channel 5.
4	SUS4	0	R	Indicates the SUS bit status of DMA channel 4.
3	SUS3	0	R	Indicates the SUS bit status of DMA channel 3.
2	SUS2	0	R	Indicates the SUS bit status of DMA channel 2.
1	SUS1	0	R	Indicates the SUS bit status of DMA channel 1.
0	SUS0	0	R	Indicates the SUS bit status of DMA channel 0.

14.4.24 DMA Status SUS Register (DSTAT_SUS_8_15/8_15S)

This register indicates the SUS bit status of the CHSTAT_n/nS register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SUS15	SUS14	SUS13	SUS12	SUS11	SUS10	SUS9	SUS8
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	SUS15	0	R	Indicates the SUS bit status of DMA channel 15.
6	SUS14	0	R	Indicates the SUS bit status of DMA channel 14.
5	SUS13	0	R	Indicates the SUS bit status of DMA channel 13.
4	SUS12	0	R	Indicates the SUS bit status of DMA channel 12.
3	SUS11	0	R	Indicates the SUS bit status of DMA channel 11.
2	SUS10	0	R	Indicates the SUS bit status of DMA channel 10.
1	SUS9	0	R	Indicates the SUS bit status of DMA channel 9.
0	SUS8	0	R	Indicates the SUS bit status of DMA channel 8.

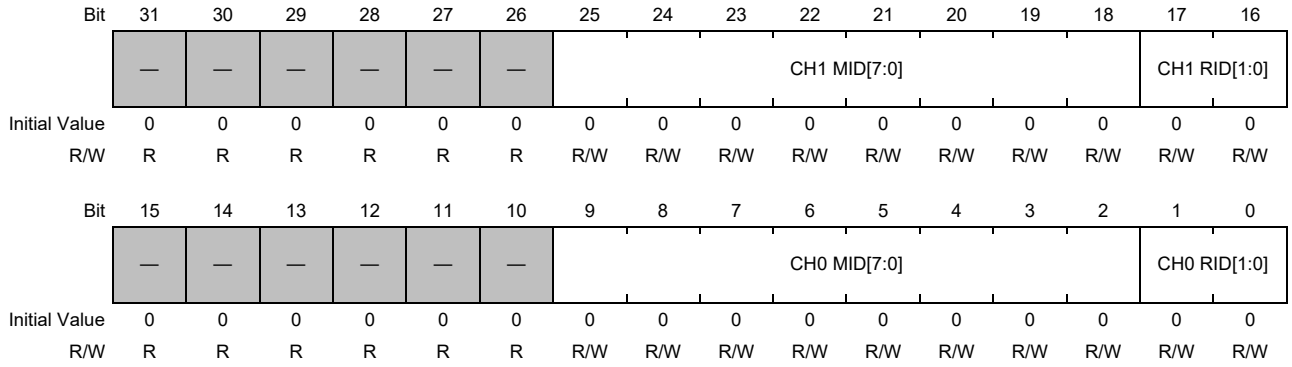
14.4.25 DMA Extension Resource Selectors 0/0S to 7/7S (DMARS0/0S to DMARS7/7S)

DMARSn/nS (n = 0 to 7) are 32-bit readable/writable registers that specify the DMA transfer sources from peripheral modules in each channel. DMARS0/0S is for channels 0/0S and 1/1S, DMARS1/1S is for channels 2/2S and 3/3S, and so on.

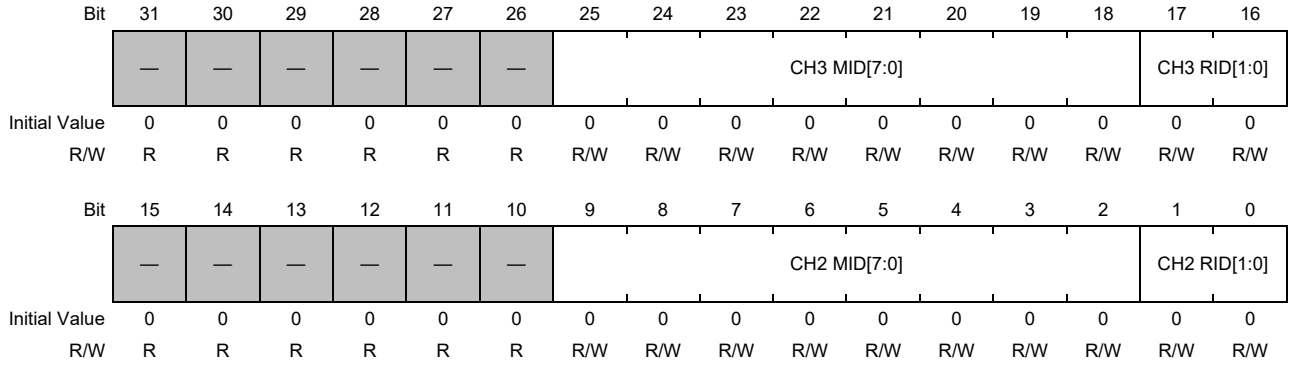
Table 14.3 shows the specifiable combinations.

Some on-chip peripheral modules in this product use the same signal both for an interrupt request and for a DMA transfer request. If such a module is selected by a DMARSn/nS register, the signal works as a DMA transfer request signal and interrupt requests to the interrupt controller are masked. To enable the interrupt, clear the setting of DMARSn/nS (set all MID[7:0] and RID[1:0] to 0). See **Table 14.3** for possible combinations of MID[7:0] and RID[1:0].

- DMARS0/0S



- DMARS1/1S



- DMARS2/2S

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH5 MID[7:0]								CH5 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH4 MID[7:0]								CH4 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS3/3S

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH7 MID[7:0]								CH7 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH6 MID[7:0]								CH6 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS4/4S

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH9 MID[7:0]								CH9 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH8 MID[7:0]								CH8 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS5/5S

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH11 MID[7:0]								CH11 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH10 MID[7:0]								CH10 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS6/6S

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH13 MID[7:0]								CH13 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH12 MID[7:0]								CH12 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS7/7S

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH15 MID[7:0]								CH15 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH14 MID[7:0]								CH14 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

14.5 Operation

When there is a DMA transfer request, the DMAC starts the transfer according to the predetermined channel priority order; when the transfer end conditions are satisfied, it ends the transfer. Transfers can be requested in two modes: auto request, and on-chip peripheral module request.

14.5.1 Transfer Flow

After the next source address register (N0SA_n/nS, N1SA_n/nS), next destination address register (N0DA_n/nS, N1DA_n/nS), next transaction byte register (N0TB_n/nS, N1TB_n/nS), channel control register (CHCTRL_n/nS), channel configuration register (CHCFG_n/nS), channel extension register (CHEXT_n/nS), DMA control register (DCTRL_0_7/0_7S, DCTRL_8_15/8_15S), and DMA extension resource selector (DMARSn/nS) are set for the target transfer conditions, the DMAC transfers data according to the following procedure:

1. Checks to see if transfer is enabled (EN = 0 and TACT = 0 in channel status register).
2. Clears the channel status register (set 1 in the SWRST bit of the channel control register).
3. Enables DMA transfer (set 1 in the SETEN bit of the channel control register).
4. When a transfer request comes and transfer is enabled, the DMAC transfers one transfer unit of data (depending on the DDS[3:0] and SDS[3:0] bit settings). For an auto request, the transfer begins automatically when 1 is set in the STG bit of the channel control register. The CRTB_n/nS value will be decremented by 1 for each transfer.
5. If 0 is set in the REN bit of the channel configuration register when transfer has been completed for the specified count (when CRTB_n/nS reaches 0), transfer ends normally. If the DEM bit of the channel configuration register is set to 0 at this time, a DMA transfer end interrupt is sent to the CPU. If the REN bit is 1 when CRTB_n/nS reaches 0, transfer operations are continued with the values of N0SA_n/nS, N1SA_n/nS, N0DA_n/nS, N1DA_n/nS, N0TB_n/nS, and N1TB_n/nS set by the RSEL bit of the channel configuration register until there are no more transfer requests.
6. When an address error in the DMAC is generated, the transfer is stopped. Transfers are also stopped when 1 is set in the CLREN bit of CHCTRL_n/nS.

14.5.2 DMA Transfer Requests

DMA transfer requests are basically generated in either the data transfer source or destination, but they can also be generated in on-chip peripheral modules that are neither the transfer source nor destination. Transfers can be requested in two modes: auto request, and on-chip peripheral module request. On-chip peripheral module request is selected by the DMARS0/0S to DMARS7/7S registers.

(1) Auto-Request Mode

When there is no transfer request signal from an external source, as in a memory-to-memory transfer or a transfer between memory and an on-chip peripheral module unable to request a transfer, auto-request mode allows the DMAC to automatically generate a transfer request signal internally. When the STG bit in channel control register n is set to 1, the transfer begins so long as the TACT bit in channel status register is 0.

(2) On-Chip Peripheral Module Request Mode

In this mode, the transfer is performed in response to the DMA transfer request signal from an on-chip peripheral module.

When a transfer request signal is sent in on-chip peripheral module request mode while DMA transfer is enabled, the DMA transfer is performed.

The DMA transfer request signals to be sent from on-chip peripheral modules or external pin input are listed in **Table 14.3**.

The transfer source or destination is fixed for some on-chip peripheral module requests. For details, see **Table 14.3**.

Table 14.3 On-Chip Module Requests (1/9)

DMA Transfer

Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS*1						
				MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
GTM ch0	OSTM0TINT (compare match)	Arbitrary	Arbitrary	0000_1000	11	0/1	010	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
GTM ch1	OSTM1TINT (compare match)	Arbitrary	Arbitrary	0000_1001	11	0/1	010	0	1	0	0/1	
GTM ch2	OSTM2TINT (compare match)	Arbitrary	Arbitrary	0000_1010	11	0/1	010	0	1	0	0/1	
GTM ch3	OSTM3TINT (compare match)	Arbitrary	Arbitrary	0000_1011	11	0/1	010	0	1	0	0/1	
GTM ch4	OSTM4TINT (compare match)	Arbitrary	Arbitrary	0000_1100	11	0/1	010	0	1	0	0/1	
GTM ch5	OSTM5TINT (compare match)	Arbitrary	Arbitrary	0000_1101	11	0/1	010	0	1	0	0/1	
GTM ch6	OSTM6TINT (compare match)	Arbitrary	Arbitrary	0000_1110	11	0/1	010	0	1	0	0/1	
GTM ch7	OSTM7TINT (compare match)	Arbitrary	Arbitrary	0000_1111	11	0/1	010	0	1	0	0/1	
MTU3 ch0	TGIA0 (input capture/compare match)	Arbitrary	Arbitrary	0001_0000	11	0/1	001	0	1	0	0/1	
	TGIB0 (input capture/compare match)	Arbitrary	Arbitrary	0001_0001	11	0/1	001	0	1	0	0/1	
	TGIC0 (input capture/compare match)	Arbitrary	Arbitrary	0001_0010	11	0/1	001	0	1	0	0/1	
	TGID0 (input capture/compare match)	Arbitrary	Arbitrary	0001_0011	11	0/1	001	0	1	0	0/1	
MTU3 ch1	TGIA1 (input capture/compare match)	Arbitrary	Arbitrary	0001_0100	11	0/1	001	0	1	0	0/1	
	TGIB1 (input capture/compare match)	Arbitrary	Arbitrary	0001_0101	11	0/1	001	0	1	0	0/1	
MTU3 ch2	TGIA2 (input capture/compare match)	Arbitrary	Arbitrary	0001_0110	11	0/1	001	0	1	0	0/1	
	TGIB2 (input capture/compare match)	Arbitrary	Arbitrary	0001_0111	11	0/1	001	0	1	0	0/1	
MTU3 ch3	TGIA3 (input capture/compare match)	Arbitrary	Arbitrary	0001_1000	11	0/1	001	0	1	0	0/1	
	TGIB3 (input capture/compare match)	Arbitrary	Arbitrary	0001_1001	11	0/1	001	0	1	0	0/1	
	TGIC3 (input capture/compare match)	Arbitrary	Arbitrary	0001_1010	11	0/1	001	0	1	0	0/1	
	TGID3 (input capture/compare match)	Arbitrary	Arbitrary	0001_1011	11	0/1	001	0	1	0	0/1	

Table 14.3 On-Chip Module Requests (2/9)

DMA Transfer

Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS*1						
				MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
MTU3 ch4	TGIA4 (input capture/compare match)	Arbitrary	Arbitrary	0001_1100	11	0/1	001	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
	TGIB4 (input capture/compare match)	Arbitrary	Arbitrary	0001_1101	11	0/1	001	0	1	0	0/1	
	TGIC4 (input capture/compare match)	Arbitrary	Arbitrary	0001_1110	11	0/1	001	0	1	0	0/1	
	TGID4 (input capture/compare match)	Arbitrary	Arbitrary	0001_1111	11	0/1	001	0	1	0	0/1	
	TCIV4 (over flow/under flow)	Arbitrary	Arbitrary	0010_0000	11	0/1	001	0	1	0	0/1	
MTU3 ch5	TGIU5 (input capture/compare match)	Arbitrary	Arbitrary	0010_0001	11	0/1	001	0	1	0	0/1	
	TGIV5 (input capture/compare match)	Arbitrary	Arbitrary	0010_0010	11	0/1	001	0	1	0	0/1	
	TGIW5 (input capture/compare match)	Arbitrary	Arbitrary	0010_0011	11	0/1	001	0	1	0	0/1	
MTU3 ch6	TGIA6 (input capture/compare match)	Arbitrary	Arbitrary	0010_0100	11	0/1	001	0	1	0	0/1	
	TGIB6 (input capture/compare match)	Arbitrary	Arbitrary	0010_0101	11	0/1	001	0	1	0	0/1	
	TGIC6 (input capture/compare match)	Arbitrary	Arbitrary	0010_0110	11	0/1	001	0	1	0	0/1	
	TGID6 (input capture/compare match)	Arbitrary	Arbitrary	0010_0111	11	0/1	001	0	1	0	0/1	
MTU3 ch7	TGIA7 (input capture/compare match)	Arbitrary	Arbitrary	0010_1000	11	0/1	001	0	1	0	0/1	
	TGIB7 (input capture/compare match)	Arbitrary	Arbitrary	0010_1001	11	0/1	001	0	1	0	0/1	
	TGIC7 (input capture/compare match)	Arbitrary	Arbitrary	0010_1010	11	0/1	001	0	1	0	0/1	
	TGID7 (input capture/compare match)	Arbitrary	Arbitrary	0010_1011	11	0/1	001	0	1	0	0/1	
	TCIV7 (over flow/under flow)	Arbitrary	Arbitrary	0010_1100	11	0/1	001	0	1	0	0/1	

Table 14.3 On-Chip Module Requests (3/9)

DMA Transfer

Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS*1						
				MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
MTU3 ch8	TGIA8 (input capture/compare match)	Arbitrary	Arbitrary	0010_1101	11	0/1	001	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
	TGIB8 (input capture/compare match)	Arbitrary	Arbitrary	0010_1110	11	0/1	001	0	1	0	0/1	
	TGIC8 (input capture/compare match)	Arbitrary	Arbitrary	0010_1111	11	0/1	001	0	1	0	0/1	
	TGID8 (input capture/compare match)	Arbitrary	Arbitrary	0011_0000	11	0/1	001	0	1	0	0/1	
PWM(GPT) ch0	CCMPA0 (input capture/compare match)	Arbitrary	Arbitrary	0011_0001	11	0	001	0	1	0	0/1	
	CCMPB0 (input capture/compare match)	Arbitrary	Arbitrary	0011_0010	11	0	001	0	1	0	0/1	
	CMPC0 (compare match)	Arbitrary	Arbitrary	0011_0011	11	0	001	0	1	0	0/1	
	CMPD0 (compare match)	Arbitrary	Arbitrary	0011_0100	11	0	001	0	1	0	0/1	
	CMPE0 (compare match)	Arbitrary	Arbitrary	0011_0101	11	0	001	0	1	0	0/1	
	CMPF0 (compare match)	Arbitrary	Arbitrary	0011_0110	11	0	001	0	1	0	0/1	
	ADTRGA0 (compare match)	Arbitrary	Arbitrary	0011_0111	11	0	001	0	1	0	0/1	
	ADTRGB0 (compare match)	Arbitrary	Arbitrary	0011_1000	11	0	001	0	1	0	0/1	
	OVF0 (overflow)	Arbitrary	Arbitrary	0011_1001	11	0	001	0	1	0	0/1	
	UNF0 (underflow)	Arbitrary	Arbitrary	0011_1010	11	0	001	0	1	0	0/1	
PWM(GPT) ch1	CCMPA1 (input capture/compare match)	Arbitrary	Arbitrary	0011_1110	11	0	001	0	1	0	0/1	
	CCMPB1 (input capture/compare match)	Arbitrary	Arbitrary	0011_1111	11	0	001	0	1	0	0/1	
	CMPC1 (compare match)	Arbitrary	Arbitrary	0100_0000	11	0	001	0	1	0	0/1	
	CMPD1 (compare match)	Arbitrary	Arbitrary	0100_0001	11	0	001	0	1	0	0/1	
	CMPE1 (compare match)	Arbitrary	Arbitrary	0100_0010	11	0	001	0	1	0	0/1	
	CMPF1 (compare match)	Arbitrary	Arbitrary	0100_0011	11	0	001	0	1	0	0/1	
	ADTRGA1 (compare match)	Arbitrary	Arbitrary	0100_0100	11	0	001	0	1	0	0/1	
	ADTRGB1 (compare match)	Arbitrary	Arbitrary	0100_0101	11	0	001	0	1	0	0/1	

Table 14.3 On-Chip Module Requests (4/9)

DMA Transfer

Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS*1						
				MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
PWM(GPT) ch1	OVF1 (overflow)	Arbitrary	Arbitrary	0100_0110	11	0	001	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
	UNF1 (underflow)	Arbitrary	Arbitrary	0100_0111	11	0	001	0	1	0	0/1	
PWM(GPT) ch2	CCMPA2 (input capture/compare match)	Arbitrary	Arbitrary	0100_1011	11	0	001	0	1	0	0/1	
	CCMPB2 (input capture/compare match)	Arbitrary	Arbitrary	0100_1100	11	0	001	0	1	0	0/1	
	CMPC2 (compare match)	Arbitrary	Arbitrary	0100_1101	11	0	001	0	1	0	0/1	
	CMPD2 (compare match)	Arbitrary	Arbitrary	0100_1110	11	0	001	0	1	0	0/1	
	CMPE2 (compare match)	Arbitrary	Arbitrary	0100_1111	11	0	001	0	1	0	0/1	
	CMPF2 (compare match)	Arbitrary	Arbitrary	0101_0000	11	0	001	0	1	0	0/1	
	ADTRGA2 (compare match)	Arbitrary	Arbitrary	0101_0001	11	0	001	0	1	0	0/1	
	ADTRGB2 (compare match)	Arbitrary	Arbitrary	0101_0010	11	0	001	0	1	0	0/1	
	OVF2 (overflow)	Arbitrary	Arbitrary	0101_0011	11	0	001	0	1	0	0/1	
	UNF2 (underflow)	Arbitrary	Arbitrary	0101_0100	11	0	001	0	1	0	0/1	
PWM(GPT) ch3	CCMPA3 (input capture/compare match)	Arbitrary	Arbitrary	0101_1000	11	0	001	0	1	0	0/1	
	CCMPB3 (input capture/compare match)	Arbitrary	Arbitrary	0101_1001	11	0	001	0	1	0	0/1	
	CMPC3 (compare match)	Arbitrary	Arbitrary	0101_1010	11	0	001	0	1	0	0/1	
	CMPD3 (compare match)	Arbitrary	Arbitrary	0101_1011	11	0	001	0	1	0	0/1	
	CMPE3 (compare match)	Arbitrary	Arbitrary	0101_1100	11	0	001	0	1	0	0/1	
	CMPF3 (compare match)	Arbitrary	Arbitrary	0101_1101	11	0	001	0	1	0	0/1	
	ADTRGA3 (compare match)	Arbitrary	Arbitrary	0101_1110	11	0	001	0	1	0	0/1	
	ADTRGB3 (compare match)	Arbitrary	Arbitrary	0101_1111	11	0	001	0	1	0	0/1	
	OVF3 (overflow)	Arbitrary	Arbitrary	0110_0000	11	0	001	0	1	0	0/1	
	UNF3 (underflow)	Arbitrary	Arbitrary	0110_0001	11	0	001	0	1	0	0/1	

Table 14.3 On-Chip Module Requests (5/9)

DMA Transfer

Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS*1						
				MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
PWM(GPT) ch4	CCMPA4 (input capture/compare match)	Arbitrary	Arbitrary	0110_0101	11	0	001	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
	CCMPB4 (input capture/compare match)	Arbitrary	Arbitrary	0110_0110	11	0	001	0	1	0	0/1	
	CMPC4 (compare match)	Arbitrary	Arbitrary	0110_0111	11	0	001	0	1	0	0/1	
	CMPD4 (compare match)	Arbitrary	Arbitrary	0110_1000	11	0	001	0	1	0	0/1	
	CMPE4 (compare match)	Arbitrary	Arbitrary	0110_1001	11	0	001	0	1	0	0/1	
	CMPF4 (compare match)	Arbitrary	Arbitrary	0110_1010	11	0	001	0	1	0	0/1	
	ADTRGA4 (compare match)	Arbitrary	Arbitrary	0110_1011	11	0	001	0	1	0	0/1	
	ADTRGB4 (compare match)	Arbitrary	Arbitrary	0110_1100	11	0	001	0	1	0	0/1	
	OVF4 (overflow)	Arbitrary	Arbitrary	0110_1101	11	0	001	0	1	0	0/1	
	UNF4 (underflow)	Arbitrary	Arbitrary	0110_1110	11	0	001	0	1	0	0/1	
PWM(GPT) ch5	CCMPA5 (input capture/compare match)	Arbitrary	Arbitrary	0111_0010	11	0	001	0	1	0	0/1	
	CCMPB5 (input capture/compare match)	Arbitrary	Arbitrary	0111_0011	11	0	001	0	1	0	0/1	
	CMPC5 (compare match)	Arbitrary	Arbitrary	0111_0100	11	0	001	0	1	0	0/1	
	CMPD5 (compare match)	Arbitrary	Arbitrary	0111_0101	11	0	001	0	1	0	0/1	
	CMPE5 (compare match)	Arbitrary	Arbitrary	0111_0110	11	0	001	0	1	0	0/1	
	CMPF5 (compare match)	Arbitrary	Arbitrary	0111_0111	11	0	001	0	1	0	0/1	
	ADTRGA5 (compare match)	Arbitrary	Arbitrary	0111_1000	11	0	001	0	1	0	0/1	
	ADTRGB5 (compare match)	Arbitrary	Arbitrary	0111_1001	11	0	001	0	1	0	0/1	
	OVF5 (overflow)	Arbitrary	Arbitrary	0111_1010	11	0	001	0	1	0	0/1	
	UNF5 (underflow)	Arbitrary	Arbitrary	0111_1011	11	0	001	0	1	0	0/1	

Table 14.3 On-Chip Module Requests (6/9)

DMA Transfer

Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS*1						
				MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
PWM(GPT) ch6	CCMPA6 (input capture/compare match)	Arbitrary	Arbitrary	0111_1111	11	0	001	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
	CCMPB6 (input capture/compare match)	Arbitrary	Arbitrary	1000_0000	11	0	001	0	1	0	0/1	
	CMPC6 (compare match)	Arbitrary	Arbitrary	1000_0001	11	0	001	0	1	0	0/1	
	CMPD6 (compare match)	Arbitrary	Arbitrary	1000_0010	11	0	001	0	1	0	0/1	
	CMPE6 (compare match)	Arbitrary	Arbitrary	1000_0011	11	0	001	0	1	0	0/1	
	CMPF6 (compare match)	Arbitrary	Arbitrary	1000_0100	11	0	001	0	1	0	0/1	
	ADTRGA6 (compare match)	Arbitrary	Arbitrary	1000_0101	11	0	001	0	1	0	0/1	
	ADTRGB6 (compare match)	Arbitrary	Arbitrary	1000_0110	11	0	001	0	1	0	0/1	
	OVF6 (overflow)	Arbitrary	Arbitrary	1000_0111	11	0	001	0	1	0	0/1	
	UNF6 (underflow)	Arbitrary	Arbitrary	1000_1000	11	0	001	0	1	0	0/1	
PWM(GPT) ch7	CCMPA7 (input capture/compare match)	Arbitrary	Arbitrary	1000_1100	11	0	001	0	1	0	0/1	
	CCMPB7 (input capture/compare match)	Arbitrary	Arbitrary	1000_1101	11	0	001	0	1	0	0/1	
	CMPC7 (compare match)	Arbitrary	Arbitrary	1000_1110	11	0	001	0	1	0	0/1	
	CMPD7 (compare match)	Arbitrary	Arbitrary	1000_1111	11	0	001	0	1	0	0/1	
	CMPE7 (compare match)	Arbitrary	Arbitrary	1001_0000	11	0	001	0	1	0	0/1	
	CMPF7 (compare match)	Arbitrary	Arbitrary	1001_0001	11	0	001	0	1	0	0/1	
	ADTRGA7 (compare match)	Arbitrary	Arbitrary	1001_0010	11	0	001	0	1	0	0/1	
	ADTRGB7 (compare match)	Arbitrary	Arbitrary	1001_0011	11	0	001	0	1	0	0/1	
	OVF7 (overflow)	Arbitrary	Arbitrary	1001_0100	11	0	001	0	1	0	0/1	
	UNF7 (underflow)	Arbitrary	Arbitrary	1001_0101	11	0	001	0	1	0	0/1	

Table 14.3 On-Chip Module Requests (7/9)

DMA Transfer

Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS*1						
				MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
SSIF ch0	INT_ssif_dma_rx_0 (receive data full)	SSIF RDR_0	Arbitrary	1001_1001	10	0	010	0	1	0	0	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
	INT_ssif_dma_tx_0 (transmit data empty)	Arbitrary	SSIF TDR_0	1001_1001	01	0	010	0	1	0	1	
SSIF ch1	INT_ssif_dma_rx_1 (receive data full)	SSIF RDR_1	Arbitrary	1001_1010	10	0	010	0	1	0	0	
	INT_ssif_dma_tx_1 (transmit data empty)	Arbitrary	SSIF TDR_1	1001_1010	01	0	010	0	1	0	1	
SSIF ch2	INT_ssif_dma_rx_2 (receive data full)	SSIF RDR_2	Arbitrary	1001_1011	10	0	010	0	1	0	0	
	INT_ssif_dma_rx_2 (transmit data empty)	Arbitrary	SSIF TDR_2	1001_1011	01	0	010	0	1	0	1	
SSIF ch3	INT_ssif_dma_rx_3 (receive data full)	SSIF RDR_3	Arbitrary	1001_1100	10	0	010	0	1	0	0	
	INT_ssif_dma_tx_3 (transmit data empty)	Arbitrary	SSIF TDR_3	1001_1100	01	0	010	0	1	0	1	
SRC	SRC_IDEI (input data FIFO empty)	Arbitrary	SRCID	1001_1101	10	0	010	0	1	0	1	
	SRC_ODFI (output data FIFO full)	SRCOD	Arbitrary	1001_1101	01	0	010	0	1	0	0	
I2C ch0	INTRIIC_RI0 (receive data full)	RIIC0DRR	Arbitrary	1010_0010	10	0	010	0	1	0	0	
	INTRIIC_TI0 (transmit data empty)	Arbitrary	RIIC0DRT	1010_0010	01	0	010	0	1	0	1	
I2C ch1	INTRIIC_RI1 (receive data full)	RIIC1DRR	Arbitrary	1010_0011	10	0	010	0	1	0	0	
	INTRIIC_TI1 (transmit data empty)	Arbitrary	RIIC1DRT	1010_0011	01	0	010	0	1	0	1	
I2C ch2	INTRIIC_RI2 (receive data full)	RIIC2DRR	Arbitrary	1010_0100	10	0	010	0	1	0	0	
	INTRIIC_TI2 (transmit data empty)	Arbitrary	RIIC2DRT	1010_0100	01	0	010	0	1	0	1	
I2C ch3	INTRIIC_RI3 (receive data full)	RIIC3DRR	Arbitrary	1010_0101	10	0	010	0	1	0	0	
	INTRIIC_TI3 (transmit data empty)	Arbitrary	RIIC3DRT	1010_0101	01	0	010	0	1	0	1	
I3C	INTRESP (Normal Response buffer full)	I3C NRSPQP	Arbitrary	1010_0110	11	0/1	100	0	1	0	0	
	INTCMD (Normal Command buffer empty)	Arbitrary	I3C NCMDQP	1010_0111	11	0/1	100	0	1	0	1	
	INTIBI (Normal IBI Status buffer full)	I3C NIBIQP	Arbitrary	1010_1000	11	0/1	100	0	1	0	0	
	INTRX (Normal Rx Data buffer full)	I3C NTDTBP0	Arbitrary	1010_1001	10	0/1	100	0	1	0	0	
	INTTX (Normal Tx Data buffer empty)	Arbitrary	I3C NTDTBP0	1010_1001	01	0/1	100	0	1	0	1	
	INTRCV (Normal Receive Status buffer full)	I3C NRSQP	Arbitrary	1010_1010	11	0/1	100	0	1	0	0	

Table 14.3 On-Chip Module Requests (8/9)

DMA Transfer

Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS*1						
				MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
I3C	INTHRESP (Response buffer full (high criteria))	I3C HRSPQP	Arbitrary	1010_1011	11	0/1	100	0	1	0	0	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
	IITHCMD (Command buffer empty (high criteria))	Arbitrary	I3C HCMDQP	1010_1100	11	0/1	100	0	1	0	1	
	INTHRX (Rx Data buffer full (high criteria))	I3C HTDTBP	Arbitrary	1010_1101	10	0/1	100	0	1	0	0	
	INTHTX (Tx Data buffer empty (high criteria))	Arbitrary	I3C HTDTBP	1010_1101	01	0/1	100	0	1	0	1	
SCIF ch0	RXI0 (receive FIFO data full)	FRDR0	Arbitrary	1010_1110	10	0	100	1	1	0	0	
	TXI0 (transmit FIFO data empty)	Arbitrary	FTDR0	1010_1110	01	0	100	1	1	0	1	
SCIF ch1	RXI1 (receive FIFO data full)	FRDR1	Arbitrary	1010_1111	10	0	100	1	1	0	0	
	TXI1 (transmit FIFO data empty)	Arbitrary	FTDR1	1010_1111	01	0	100	1	1	0	1	
SCIF ch2	RXI2 (receive FIFO data full)	FRDR2	Arbitrary	1011_0000	10	0	100	1	1	0	0	
	TXI2 (transmit FIFO data empty)	Arbitrary	FTDR2	1011_0000	01	0	100	1	1	0	1	
SCIF ch3	RXI3 (receive FIFO data full)	FRDR3	Arbitrary	1011_0001	10	0	100	1	1	0	0	
	TXI3 (transmit FIFO data empty)	Arbitrary	FTDR3	1011_0001	01	0	100	1	1	0	1	
SCIF ch4	RXI4 (receive FIFO data full)	FRDR4	Arbitrary	1011_0010	10	0	100	1	1	0	0	
	TXI4 (transmit FIFO data empty)	Arbitrary	FTDR4	1011_0010	01	0	100	1	1	0	1	
SCIF ch5	RXI5 (receive FIFO data full)	FRDR5	Arbitrary	1011_0011	10	0	100	1	1	0	0	
	TXI5 (transmit FIFO data empty)	Arbitrary	FTDR5	1011_0011	01	0	100	1	1	0	1	
SCIg ch0	RXI0 (receive FIFO data full)	RDR0	Arbitrary	1011_0100	10	0	010	1	1	0	0	
	TXI0 (transmit FIFO data empty)	Arbitrary	TDR0	1011_0100	01	0	010	1	1	0	1	
SCIg ch1	RXI1 (receive FIFO data full)	RDR1	Arbitrary	1011_0101	10	0	010	1	1	0	0	
	TXI1 (transmit FIFO data empty)	Arbitrary	TDR1	1011_0101	01	0	010	1	1	0	1	

Table 14.3 On-Chip Module Requests (9/9)

DMA Transfer

Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS*1						
				MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
RSPI ch0	SPRI0 (receive buffer full)	SPDR0	Arbitrary	1011_0110	10	0	010	1	1	0	0	CH0:000 CH1:001
	SPTI0 (transmit buffer empty)	Arbitrary	SPDR0	1011_0110	01	0	010	1	1	0	1	CH2:010 CH3:011
RSPI ch1	SPRI1 (receive buffer full)	SPDR1	Arbitrary	1011_0111	10	0	010	1	1	0	0	CH4:100 CH5:101
	SPTI1 (transmit buffer empty)	Arbitrary	SPDR1	1011_0111	01	0	010	1	1	0	1	CH6:110 CH7:111
RSPI ch2	SPRI2 (receive buffer full)	SPDR2	Arbitrary	1011_1000	10	0	010	1	1	0	0	CH8:000 CH9:001
	SPTI2 (transmit buffer empty)	Arbitrary	SPDR2	1011_1000	01	0	010	1	1	0	1	CH10:010 CH11:011
RSPI ch3	SPRI3 (receive buffer full)	SPDR3	Arbitrary	1011_1001	10	0	010	1	1	0	0	CH12:100 CH13:101
	SPTI3 (transmit buffer empty)	Arbitrary	SPDR3	1011_1001	01	0	010	1	1	0	1	CH14:110 CH15:111
RSPI ch4	SPRI4 (receive buffer full)	SPDR4	Arbitrary	1011_1010	10	0	010	1	1	0	0	
	SPTI4 (transmit buffer empty)	Arbitrary	SPDR4	1011_1010	01	0	010	1	1	0	1	
CANFD	RXF_DMA0	RSCFD0CFDRFDF0_0	Arbitrary	1011_1011	11	0	001	0	1	0	0/1	
	RXF_DMA1	RSCFD0CFDRFDF0_1	Arbitrary	1011_1100	11	0	001	0	1	0	0/1	
	RXF_DMA2	RSCFD0CFDRFDF0_2	Arbitrary	1011_1101	11	0	001	0	1	0	0/1	
	RXF_DMA3	RSCFD0CFDRFDF0_3	Arbitrary	1011_1110	11	0	001	0	1	0	0/1	
	RXF_DMA4	RSCFD0CFDRFDF0_4	Arbitrary	1011_1111	11	0	001	0	1	0	0/1	
	RXF_DMA5	RSCFD0CFDRFDF0_5	Arbitrary	1100_0000	11	0	001	0	1	0	0/1	
	RXF_DMA6	RSCFD0CFDRFDF0_6	Arbitrary	1100_0001	11	0	001	0	1	0	0/1	
	RXF_DMA7	RSCFD0CFDRFDF0_7	Arbitrary	1100_0010	11	0	001	0	1	0	0/1	
	COM_DMA0 (transceiver FIFO receive mode only)	RSCFD0CFDCFDF0_0	Arbitrary	1100_0011	11	0	001	0	1	0	0/1	
	COM_DMA1 (transceiver FIFO Receive mode only)	RSCFD0CFDCFDF0_1	Arbitrary	1100_0100	11	0	001	0	1	0	0/1	
SPDIF	rbdmareqn_tx	Arbitrary	TDAD	1001_1110	01	0	001	1	1	0	1	
	rbdmareqn_rx	RDAD	Arbitrary	1001_1110	10	0	001	1	1	0	0	
PDM	INT_PDM_DAT0	PDDRRCH0	Arbitrary	1001_1111	11	0	100	1	1	0	0	
	INT_PDM_DAT1	PDDRRCH1	Arbitrary	1010_0000	11	0	100	1	1	0	0	
	INT_PDM_DAT2	PDDRRCH2	Arbitrary	1010_0001	11	0	100	1	1	0	0	

Note 1. CHCFG_n/nS setting value

TM	0: Single transfer
	1: Block transfer
AM	001: ACK level output
	010: ACK bus cycle output
	100: No ACK
LVL	0: REQ edge detection
	1: REQ level detection
REQD	0: ACK output at read
	1: ACK output at write

Note 2. Only in products with a Trusted Secure IP

14.6 DMA Mode

14.6.1 Mode Setting

The DMS field of the CHCFG_n/nS register can be used to toggle between register mode and link mode.

Table 14.4 DMA Mode Setting

DMS (CHCFG_n/nS)	Mode	Description
0	Register mode	A DMA transfer is executed using the values set in the Next Register Set.
1	Link mode	A DMA transfer is executed using the descriptor set in the Current register. The DMAC repeatedly loads the descriptor and executes the DMA transfer unless otherwise set by the descriptor or stopped by the control register.

14.6.2 Register Mode

In register mode, a DMA transfer is executed using the values set in the internal registers.

Two sets of the source address, destination address, and transfer byte count (Next0 Register Set and Next1 Register Set) can be set. It is possible to select the Next register set to be used for the DMA transfer, as well as to execute two Next register sets continuously for the DMA transfer.

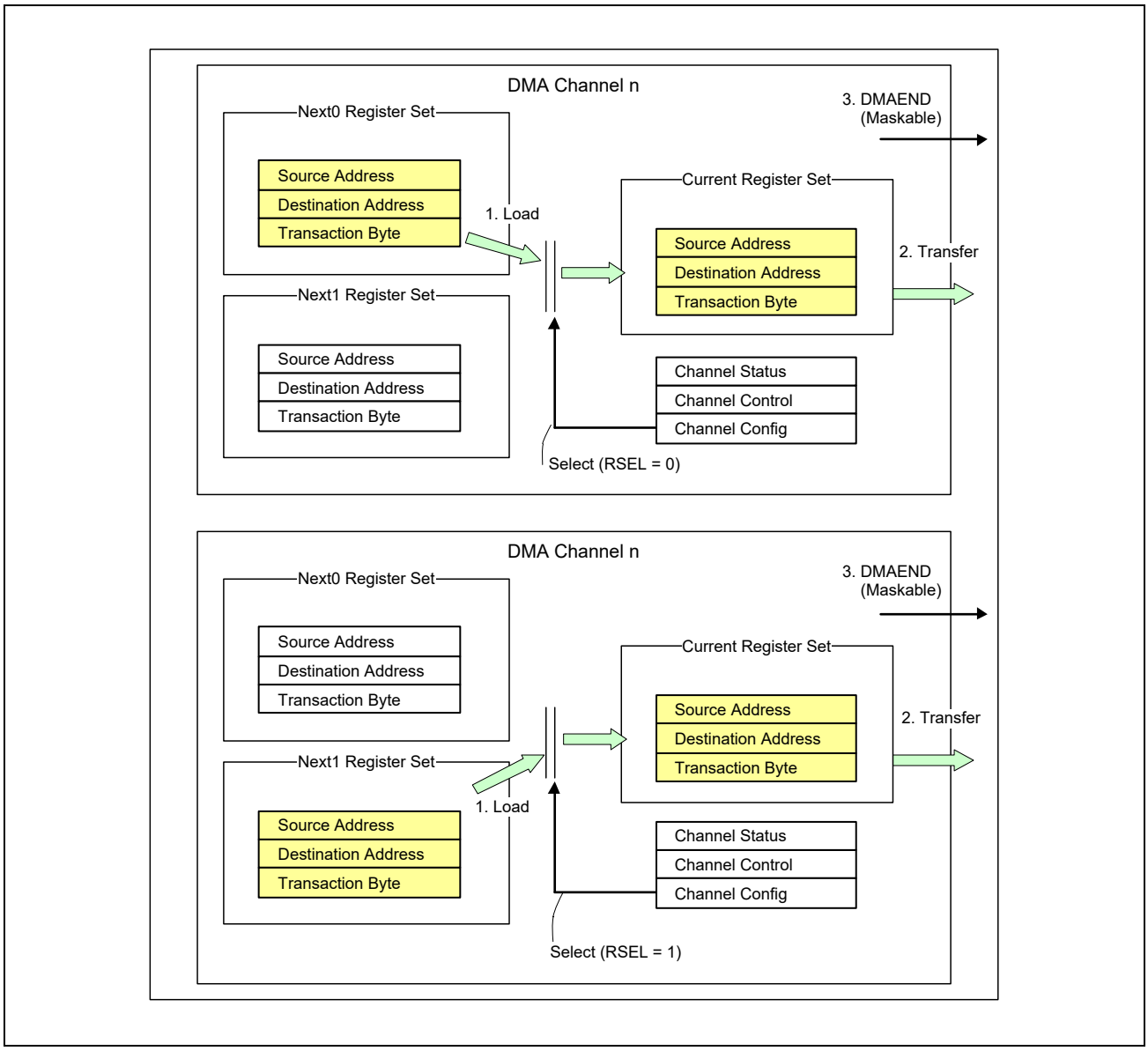


Figure 14.2 Outline of Normal Register Mode

The above figure shows how the transfer is executed when the Next0 Register Set is used (upper part of the figure) and when the Next1 Register Set is used (lower part of the figure).

(1) Operation Flow

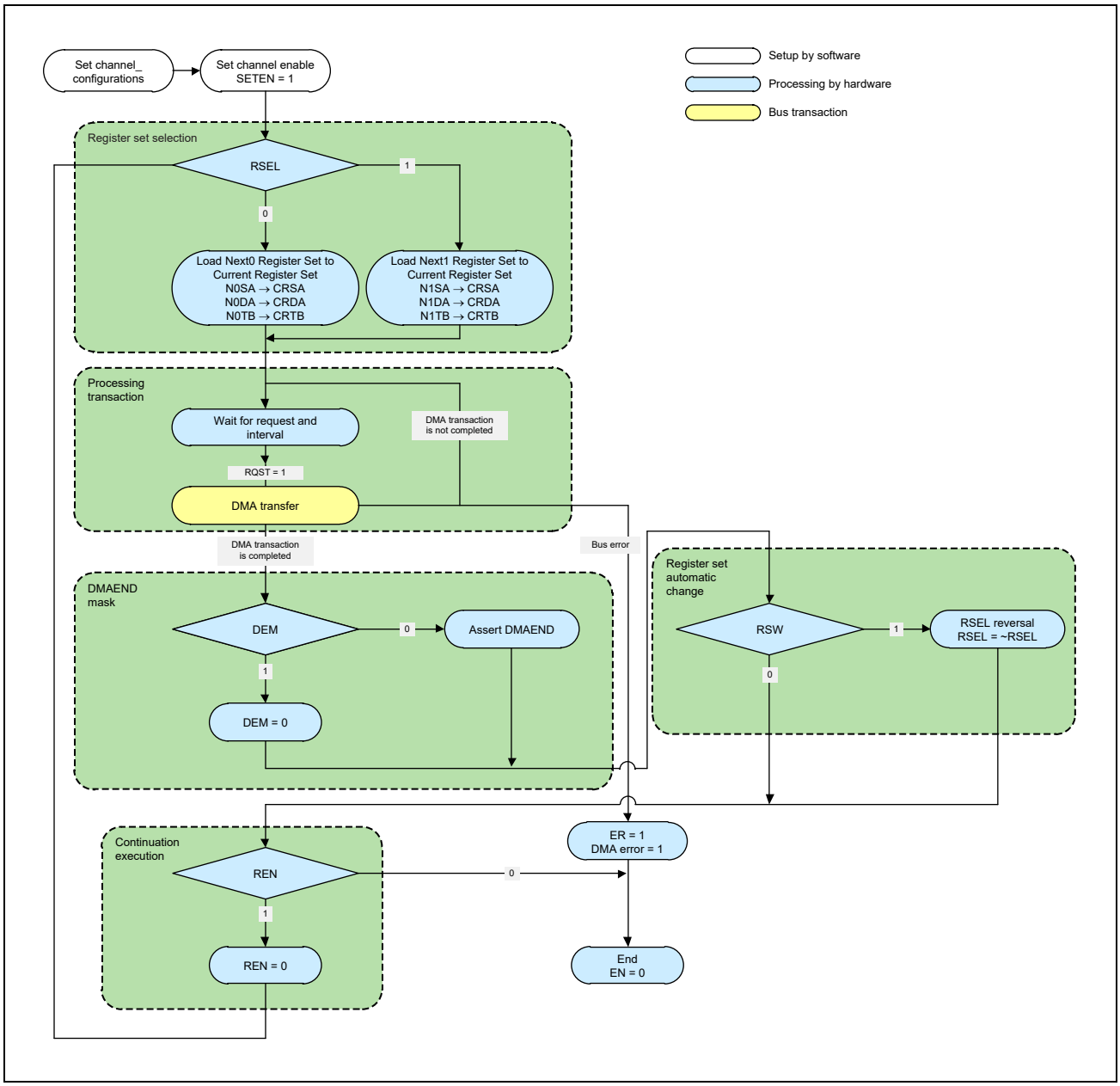


Figure 14.3 Register Mode Flow

<Explanation of the register mode flow>

1. Channel setting (set channel configuration)

   The Next0 or Next1 register set (destination address, source address, and total transfer byte count) is set.

   In the Channel register set, the DMA register set (REQ, DMAACK, transfer size, etc.) is set. (See **Section 14.7, DMA Transfer**.)

2. Register set selection (register set selection)

   When 1 is set in EN, the values set in the Next register set selected by RSEL are loaded to the Current register set.

3. DMA transaction (processing transaction)

   A DMA transfer is executed according to the set values. For details of the transfer, see **Section 14.7, DMA Transfer**.

4. DMA transfer end interrupt mask (DMAINT mask)

   The DMA transfer end interrupt is masked according to the value set in the DEM bit of CHCFG_n/nS. When 1 is set in DEM, the DMA transfer end interrupt is not output. Also, immediately after that, DEM is automatically cleared to 0.

5. Automatic register set change (register set automatic change)

   Whether to use the other Next register set is determined by the value set in the RSW bit of CHCFG_n/nS.

6. Continuation of execution (continuation execution)

   Whether to continue the execution of the DMA transfer is determined by the value set in the REN bit of CHCFG_n/ nS. When 1 is set in REN, the execution of the DMA transfer is continued. Also, immediately after that, REN is automatically cleared to 0.

(2) Register Setting

(a) Register mode setting

Select the register set to be executed.

Table 14.5 Register Mode Setting

DMS (CHCFG_n/nS)	RSEL (CHCFG_n/nS)	Description
0	0	Executes the Next0 Register Set.
	1	Executes the Next1 Register Set.

(b) DMA transfer end interrupt mask setting

The DMA transfer end interrupt can be masked individually for each register set.

Table 14.6 DMAINT Mask Setting

DEM (CHCFG_n/nS)	Operation
0	When the DMA transaction is completed, a DMA transfer end interrupt is issued.
1	Even when the DMA transaction is completed, a DMA transfer end interrupt is not issued. After the DMA transaction is completed, DEM is cleared to 0 by hardware.

(c) Automatic register set execution setting

After DMA transfers, the DMA transaction of the selected register set is automatically executed.

Table 14.7 Automatic Register Set Execution Setting

REN (CHCFG_n/nS)	Operation	Remark
0	When the DMA transaction of the register set selected by RSEL is completed, the EN bit is cleared and the DMA operation ends.	Set this value to execute a DMA transaction once.
1	When a DMA transaction is completed, the DMAC continues to execute a DMA transfer by using the data set in the selected register set. When continuous transfers are successful, REN is cleared to 0.	Set this value to continuously execute DMA transfers by using the data set in separate register sets.

(d) Automatic register set change setting

When 1 is set in REN, the DMAC can automatically change to the register set to be executed next, after a DMA transaction is completed.

Table 14.8 Automatic Register Set Change Setting

RSW (CHCFG_n/nS)	Operation	Remark
0	If 1 is set in REN when a DMA transaction is completed, the register set is not changed.	Set this value to use only one register set.
1	If 1 is set in REN when a DMA transaction is completed, the value of RSEL is automatically inverted and the other register set is selected.	Set this value to change the register set.

(3) Setting Examples

(a) When only the Next0 register set is used

Table 14.9 Register Mode Setting Example 1

DMS (CHCFG_n/nS)	RSEL (CHCFG_n/nS)	DEM (CHCFG_n/nS)	RSW (CHCFG_n/nS)	REN (CHCFG_n/nS)
0 (Register mode)	0 (Next0)	0 (not masked)	0 (not switched)	0 (not continuously executed)

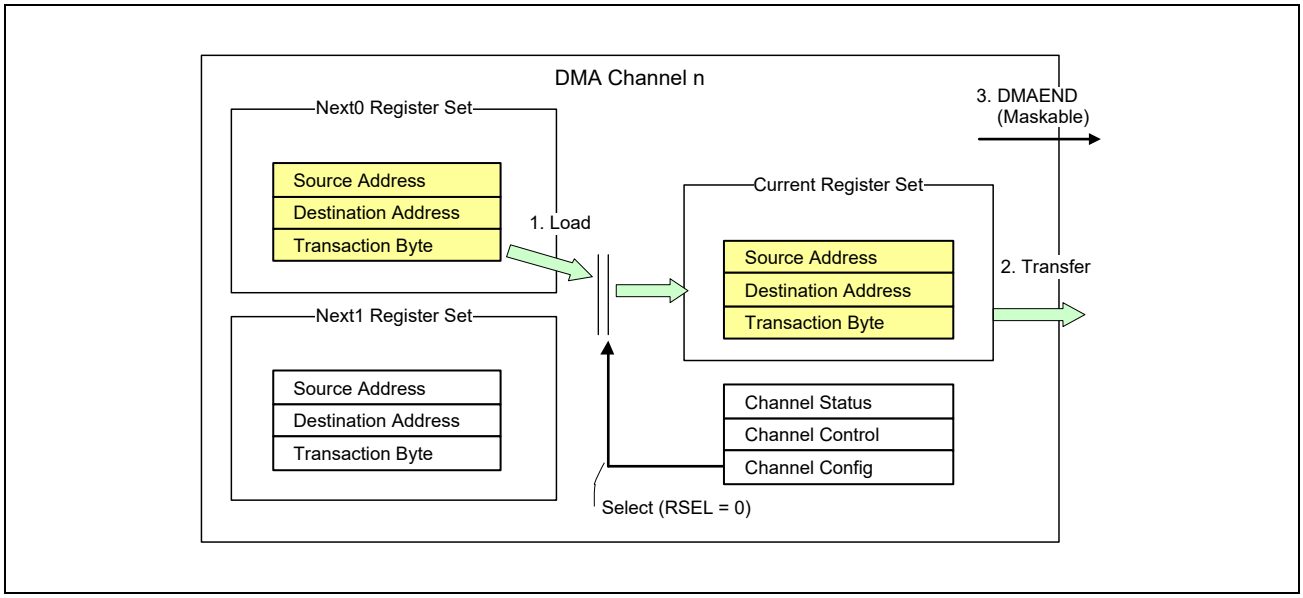


Figure 14.4 Register Mode Setting Example 1

1. 1 is set in EN (SETEN = 1), and the Next0 register set is loaded to the Current register set.
2. A DMA transaction is executed according to the values set in the Current register set and Channel register set.
3. Because 0 is set in DEM, the DMA transfer end interrupt is issued after the DMA transaction is completed.
4. Because 0 is set in REN, EN is cleared to 0 and the DMA transaction ends.

(b) When two register sets are used continuously

Table 14.10 Automatic Register Set Execution Setting

DMS (CHCFG_n/nS)	RSEL (CHCFG_n/nS)	DEM (CHCFG_n/nS)	RSW (CHCFG_n/nS)	REN (CHCFG_n/nS)
0 (Register mode)	0 (Next0)	1 (masked)	1 (switched)	1 (continuously executed)

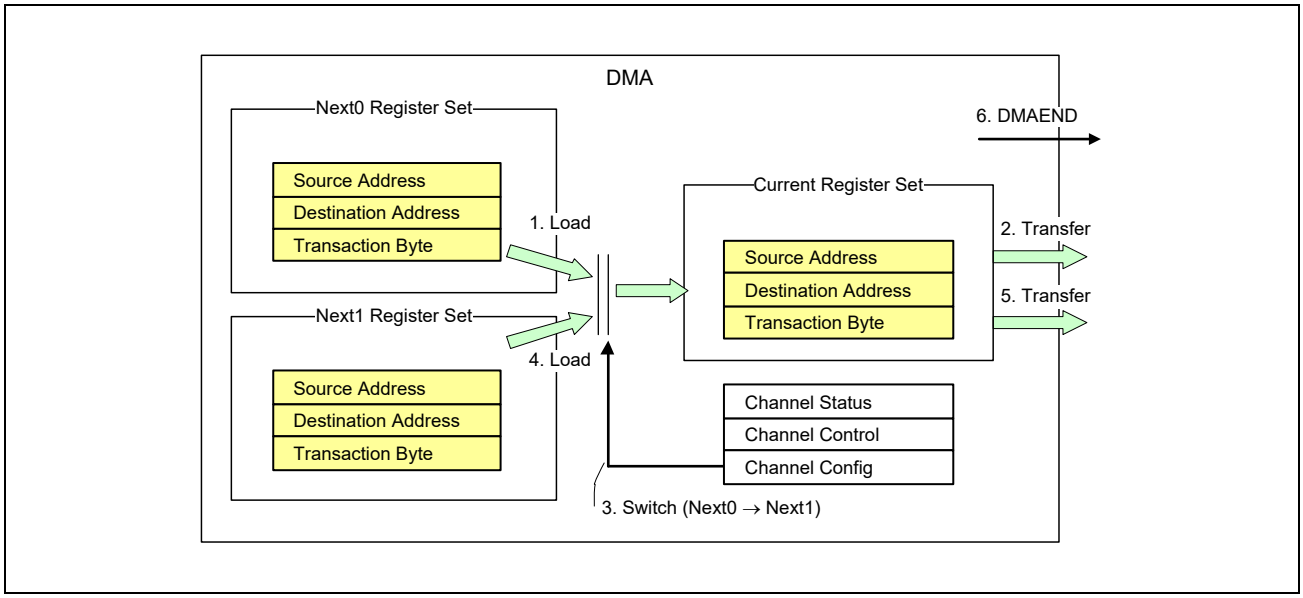


Figure 14.5 Register Mode Setting Example 2

1. 1 is set in EN (SETEN = 1), and the Next0 register set is loaded to the Current register set.
2. A DMA transaction is executed according to the values set in the Current register set and Channel register set.
3. Because 1 is set in DEM, DMA transfer end interrupt is not output after the DMA transaction is completed. Also, DEM is automatically cleared to 0.
4. Because 1 is set in REN, the execution is continued. Also, REN is automatically cleared to 0.
5. Because 1 is set in RSW, the register set to be executed next is switched (RSEL = 0 → 1).
6. The Next1 register set is loaded to the Current register set.
7. A DMA transaction is executed according to the values set in the Current register set and Channel register set.
8. Because 0 is set in DEM, the DMA transfer end interrupt is issued after the DMA transaction is completed.
9. Because 0 is set in REN, EN is cleared to 0 and the DMA transaction ends.

14.6.3 Link Mode

In link mode, a descriptor stored in external memory is loaded as set values and a DMA transaction is executed using the loaded values. The DMAC contains a Next Link address and a Current Link address for each channel, and these addresses are used to set the descriptor address to be executed next and to indicate the descriptor address of the currently executed DMA transaction, respectively.

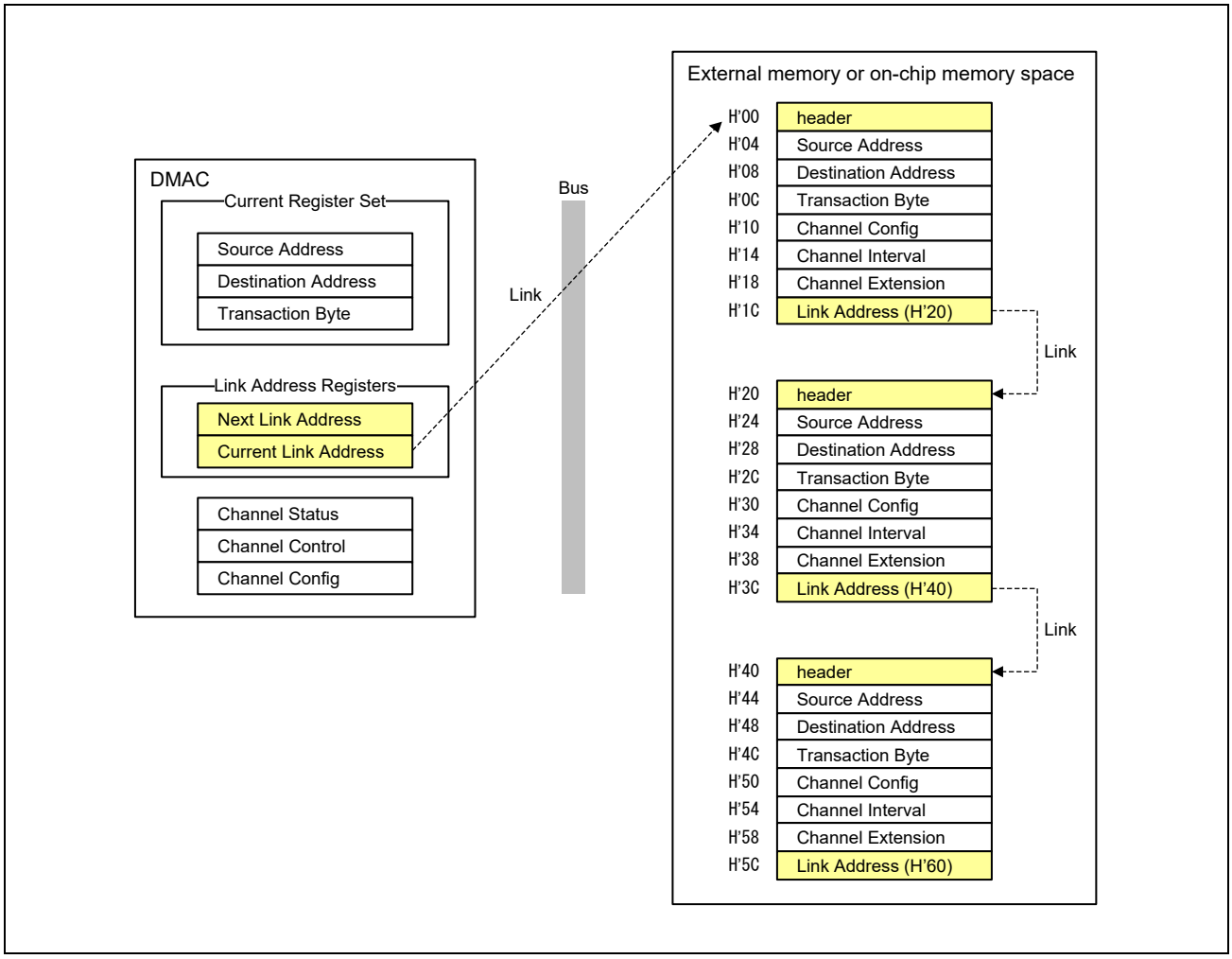


Figure 14.6 Link Mode Outline

(1) Operation Flow

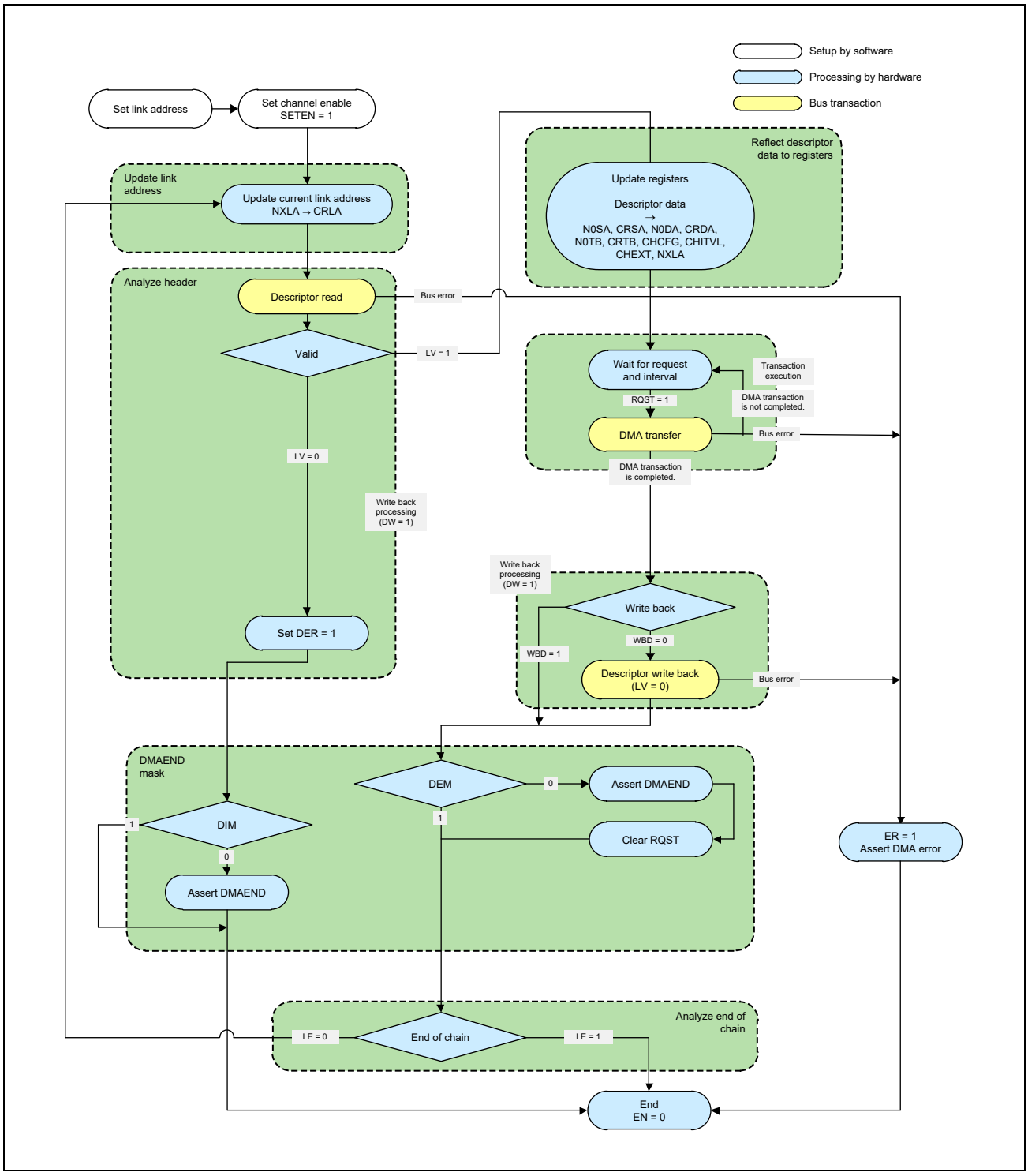


Figure 14.7 Link Mode Flow

<Explanation of the link mode flow>

1. Channel setting

   The start address of the link destination is set in NXLA_n/nS.

2. Link address update

   When 1 is set in EN (1 is set in SETEN), the Link address set in NXLA_n/nS is loaded to CRLA_n/nS.

3. Descriptor load and header analysis

   The DMAC begins to load the descriptor and then analyzes the content of the header. When LV is 0, the DMAC discards the loaded descriptor and sets 1 in DER to end the operation (EN = 0). In this case, if 0 is set in DIM of the header, DMAEND is issued.

4. Descriptor setting

   The loaded descriptor is set in the Current register set and Channel register set. Also, the next link address is set in NXLA_n/nS.

5. DMA transaction

   A DMA transaction is executed according to the set values.

6. Header writeback

   When 0 is set in WBD of the header, the DMAC writes back the header with 0 set in its LV bit.

7. DMAINT mask

   When 0 is set in the DEM bit of CHCFG_n/nS, the DMA transfer end interrupt is issued.

8. Link end analysis

   When 1 is set in LE of the header, the operation is ended by clearing EN to 0, after transfer using the settings of the descriptor is completed. If the setting of LE is 0, the Current registers are then updated and loading of the next descriptor begins. The TEND signal is issued after the transfer of each descriptor.

(2) Register Setting

(a) Link mode setting

To use the link mode, set 1 in the DMS bit of the CHCFG_n/nS register.

Table 14.11 Link Mode Setting

DMS (CHCFG_n/nS)	Description
1	Operates in link mode. This bit cannot be changed using a descriptor.

(b) Link address setting

There are two registers that indicate a link address:
Next Link address register and Current Link address register.
To start the link mode, set a link address in the Next Link address register.

The Next Link address indicates the next link address after a descriptor is loaded. The Current Link address indicates the currently executed link address.

Table 14.12 Link Address Register Set

Register	Description
Next Link Address Register (NXLA_n/nS)	Sets and indicates the next link address. Before starting the link mode, set a link address in this register.
Current Link Address Register (CRLA_n/nS)	Indicates the currently executed link address. This register is read-only.

CAUTION

In link mode, the settings can be changed by reading a descriptor. It is not possible, however, to synchronize the change of the settings with a peripheral module request or external request. Therefore, when using a peripheral module request or external request, set AM, LVL, HIEN, LOEN, and SEL of the CHCFG_n/nS register before setting Enable and do not change any of these bits in the descriptor.

(3) Descriptor Setting

In a link address, prepare a descriptor with data arranged in the order shown below.

The DMAC reads the descriptor in burst mode.

(a) Descriptor data arrangement

Table 14.13 Descriptor Data Arrangement

Address	Data	Remarks
Link address + H'00	header	
Link address + H'04	Source Address	
Link address + H'08	Destination Address	
Link address + H'0C	Transaction Byte	
Link address + H'10	Config	The register mode cannot be set.
Link address + H'14	Interval	
Link address + H'18	Extension	
Link address + H'1C	Next Link Address	

Note: As a link address, set an address aligned along the 32-bit boundary.

(b) header

The header indicates the status of the descriptor, as shown below.

The DMAC reads this area when a DMA transfer is started in link mode. Also, after a DMA transaction is completed, the DMAC writes back the transfer status to the area.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	—	—	—	—	—	—	—	—	—	DIM	WBD	LE	LV

If 0 is set in WBD when the header is read, the DMAC writes back to this 4-byte area after a DMA transaction using a descriptor is completed.
0 is written back to LV, and the values obtained when reading the header are written back to the other fields.

Figure 14.8 Header Area

Table 14.14 Header Area

Bit Position	Bit Name	Meaning
31 to 4	—	—
3	DIM	Descriptor Interrupt Mask Sets whether to mask the DMA transfer end interrupt if 0 is set in LV when the header is loaded. 0: Issues a DMA transfer end interrupt. 1: Does not issue a DMA transfer end interrupt.
2	WBD	Write Back Disable Sets whether to mask LV bit writeback. When 1 is set in this bit, the DMAC does not perform writeback. 0: Writes the LV bit back to 0. 1: Does not write back the LV bit.
1	LE	Link End Indicates whether the link ends with the DMA transaction of this descriptor. Set 1 in this bit to indicate the end of the link. 0: The link continues. 1: The link ends.
0	LV	Link Valid Indicates whether this descriptor is valid. If 0 is set in WBD, the DMAC writes 0 in this bit after the DMA transaction written in the descriptor is executed. When setting the header, set 1 in this bit. 0: Descriptor invalid 1: Descriptor valid

(c) Descriptor data other than the header

The data items of the descriptor other than the header are the same as defined in the internal register specifications (note that the DMS bit of the CHCFG_n/nS register cannot be changed using the descriptor). For information about the internal register specifications, see **Section 14.4, Register Descriptions**.

For descriptor setting examples, see **Section 14.8, DMA Setting Examples**.

(d) CACHE settings for descriptor access

The CACHE settings for descriptor access can be set in LWCA and LDCA of the DMA control register (DCTRL_0_7/ 0_7S, DCTRL_8_15/8_15S). Make these settings as appropriate for the access destination in which the descriptor is prepared.

(e) Descriptor area and DMA transfer area

The following figure outlines the descriptor area and DMA transfer area that are accessed by the DMAC.

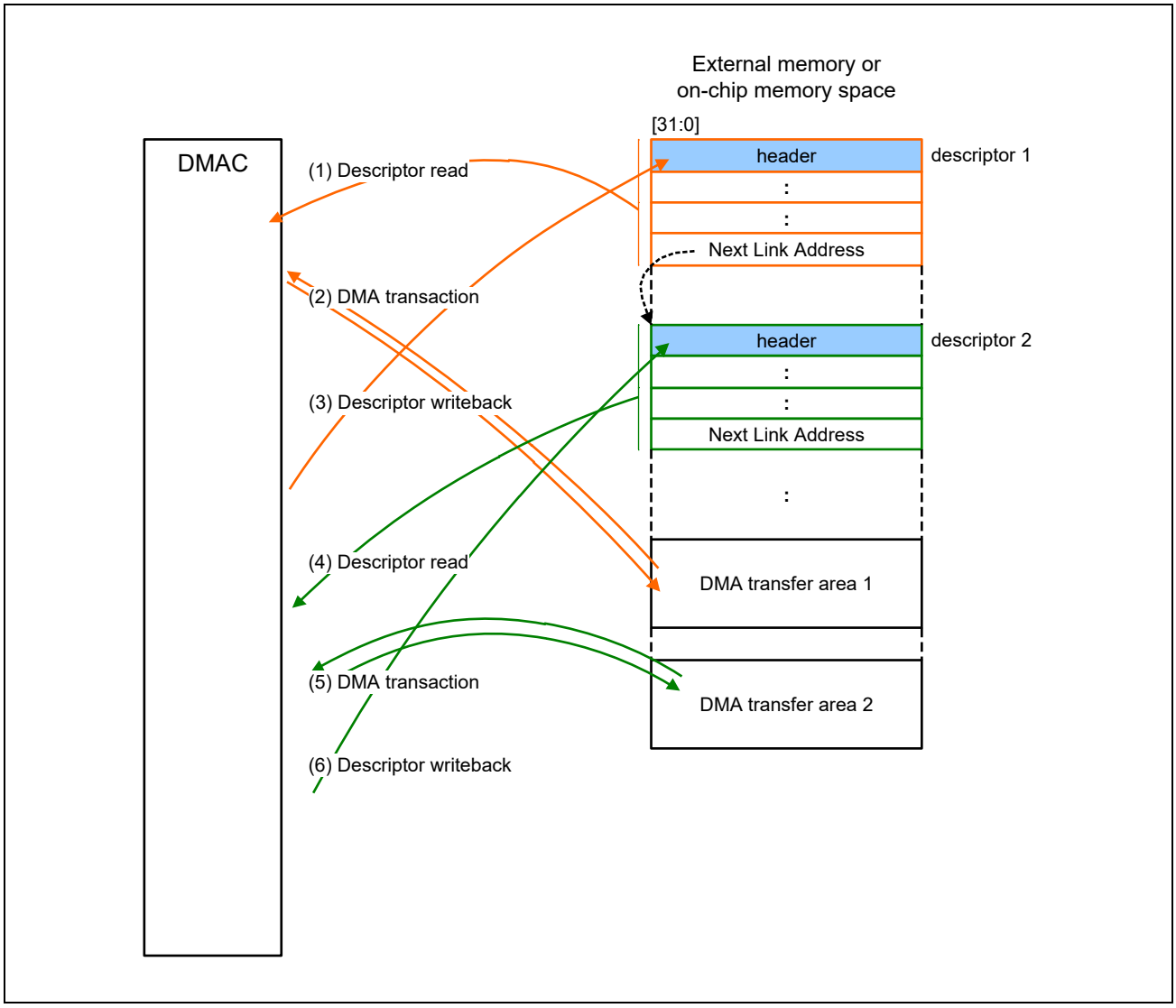


Figure 14.9 Outline of the Descriptor Area and DMA Transfer Area

1. Descriptor read

   The values set in the internal Next Link Address register are loaded to the Current Link Address register, and a descriptor is read from the external memory space (descriptor1) pointed to by the Current Link Address register.

2. DMA transfer

   When 1 is set in the LV bit of the header in the descriptor, a DMA transfer is executed according to the descriptor data.

3. Descriptor writeback

   When 0 is set in the WBD bit of the header after the DMA transfer of the set number of bytes is completed, the DMAC writes back data in word size to the header of descriptor1, with 0 set in LV and the other bits containing the values read in <1>.

4. Descriptor read

   When 0 is set in the LE bit of the header in the last read descriptor (<1>), the next descriptor is read from the address (descriptor2) indicated by Next Link Address in the descriptor.

5. DMA transfer

   When 1 is set in the LV bit of the header in the descriptor, a DMA transfer is executed according to the descriptor data.

6. Descriptor writeback

   When 0 is set in the WBD bit of the header after the DMA transfer of the set number of bytes is completed, the DMAC writes back data in word size to the header of descriptor2, with 0 set in LV and the other bits containing the values read in <4>.

4 through 6 are repeated.

When the header contains 1 in LE and 0 in WBD, the DMAC executes a DMA transfer using the settings of that descriptor, writes back data with 0 set in the LV bit of the header and ends the operation.

When the header contains 1 in both LE and WBD, the DMAC executes a DMA transfer using the settings of that descriptor and ends the operation (without writing back).

When the header contains 0 in LV, the DMAC ends the operation (without executing a DMA transfer).

(4) Descriptor Configuration Examples

In link mode, a descriptor can be configured as shown below.

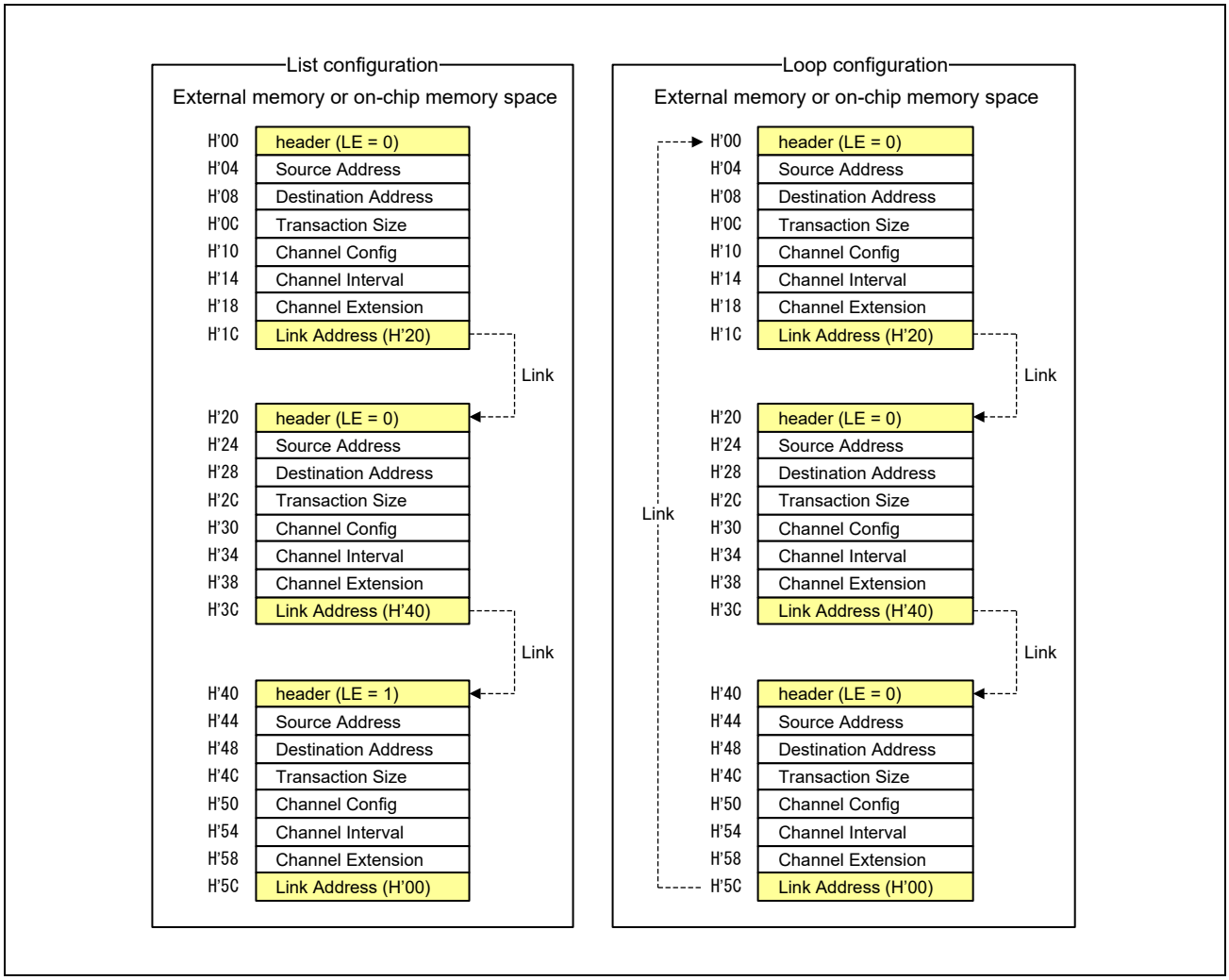


Figure 14.10 Descriptor Configuration Examples

- List configuration
  The link is ended by setting 1 in the LE bit of the header in the last descriptor.
- Loop configuration
  A descriptor can be created with a loop configuration, by setting the address of the top descriptor in the link address of the last descriptor. To end the loop, change the value of the LE bit of the header to 1 before the DMAC reads the descriptor, or follow the transfer suspension procedure.

14.7 DMA Transfer

The basic operation of DMA transfer is described here.

14.7.1 Transfer Mode

Two transfer modes are supported: single transfer mode and block transfer mode.

To select a transfer mode, set the TM bit of CHCFG_n/nS for each channel.

Table 14.15 Basic Transfer Setting

Transfer Mode	TM (CHCFG_n/nS)	Function
Single transfer	0	A single DMA transfer is executed in response to a DMAREQ.
Block transfer	1	In response to a DMAREQ, the DMAC continues to execute the transfer until the DMA transaction is completed.

(1) Single Transfer Mode

When a DMA transfer request is received, a DMA transfer is executed once in the direction indicated by REQD (source or destination). A DMA transfer is executed once each time a transfer request is received, and this operation continues until the number of bytes loaded to CRTB_n/nS is reached (arbitration between channels is accomplished for each DMA transfer).

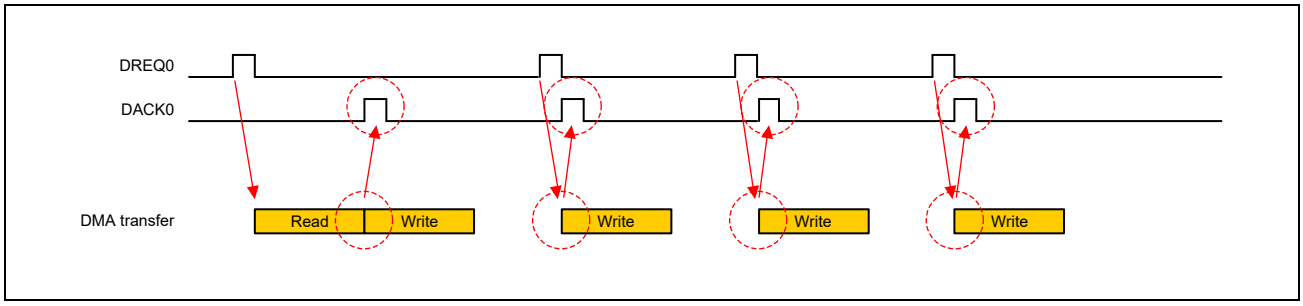


Figure 14.11 Single Transfer Mode (REQD = 1, SDS > DDS)

(2) Block Transfer Mode

Once a DMA transfer request is received, the DMAC continues to execute the transfer until data equivalent to the number of bytes loaded to the DMA transfer byte register (CRTB_n/nS register) is transferred (the DMA transaction is completed) (arbitration between channels is accomplished for each DMA transfer).

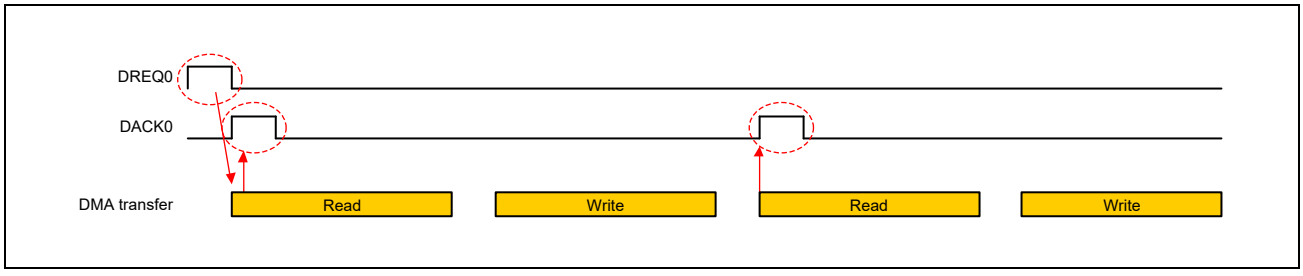


Figure 14.12 Block Transfer Mode (REQD = 0, SDS < DDS)

14.7.2 Priority Control for DMA Channels

Within channels 0 to 7 and 8 to 15, two priority control modes are supported: fixed priority mode and round robin mode. Only round robin mode is supported for priority control between the group of channels 0 to 7 and the group of channels 8 to 15. To select a priority control mode, use the PR bit of the DMA control register (DCTRL register). The fixed priority mode is selected when 0 is set in the PR bit, and the round robin mode is selected when 1 is set.

Read priority and write priority are controlled independently.

The DMAC issues transfer requests to different channels concurrently without waiting for the completion of any particular transfer and processes responses in the order it receives them. Therefore, the order in which the channels start transactions is not necessarily consistent with the order in which the transactions end.

Table 14.16 Priority Control Setting

Mode	PR (DCTRL)	Function	Purpose
Fixed priority	0	Requests are controlled based on the fixed order of priority for channels 0 to 7 and 8 to 15 (High: CH0 (CH8) > CH1 (CH9) > CH2 (CH10) > CH3 (CH11) > CH4 (CH12) > CH5 (CH13) > CH6 (CH14) > CH7 (CH15): Low).	Use this mode when the channels have a specific order of priority.
Round robin	1	Requests are controlled in a round robin fashion.	Use this mode to execute all requests evenly.

(1) Fixed Priority Mode

In fixed priority mode, the channels have a fixed order of priority in channels 0 to 7 and 8 to 15. Round robin mode is used to determine the priority between the group of channels 0 to 7 and the group of channels 8 to 15.

Immediately after a reset, the order of priority is as follows.

High CH0 > CH8 > CH1 > CH9 > CH2 > CH10 > CH3 > CH11 > CH4 > CH12 > CH5 > CH13 > CH6 > CH14 > CH7 > CH15 Low

If there is a transfer request from DMA channel 0 in this state, a transfer is executed on DMA channel 0. After the transfer is completed, the order of priority is as follows.

High CH8 > CH0 > CH9 > CH1 > CH10 > CH2 > CH11 > CH3 > CH12 > CH4 > CH13 > CH5 > CH14 > CH6 > CH15 > CH7 Low

If a DMA transfer request occurs on multiple channels simultaneously, the DMA transfer request of the channel having the smallest channel number is given priority. The following figure shows an example where a DMA transfer request occurs on a channel having a higher priority while a DMA transfer is being executed in fixed priority mode.

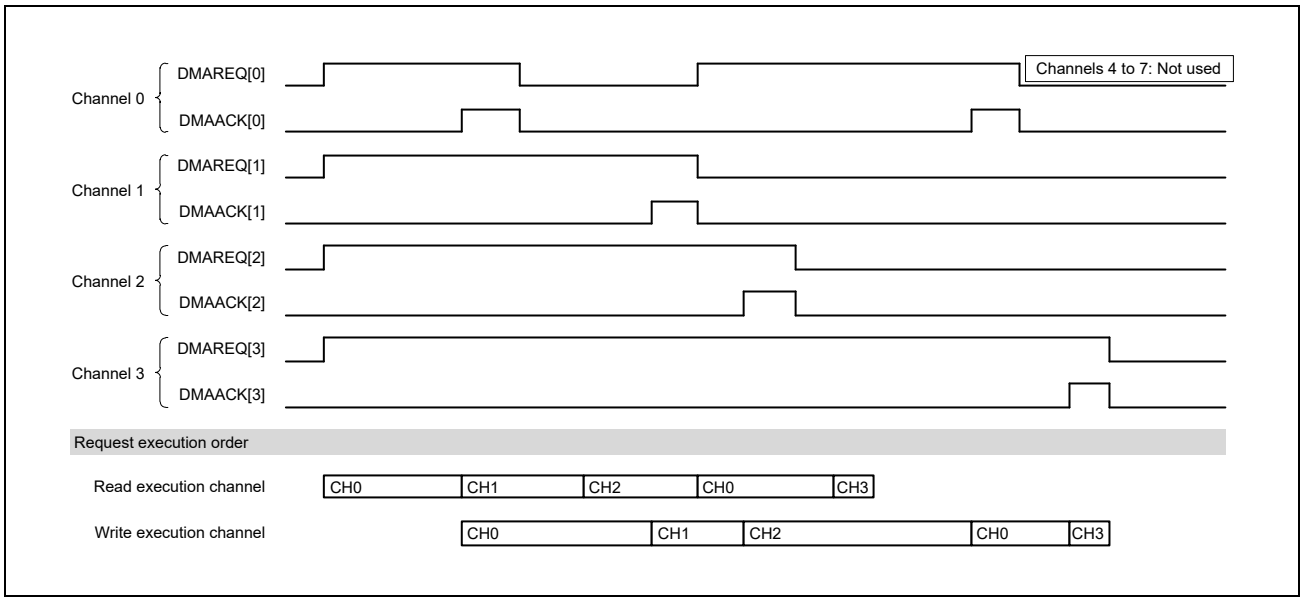


Figure 14.13 Fixed Priority Mode (Number of Channels = 4, REQD = 1)

(2) Round Robin Mode

In round robin mode, each time a transfer request is received from a channel in the group of channels 0 to 7 and the group of channels 8 to 15, the order of priority is changed in such a way that the channel that executed a transfer last has the lowest priority.
Round robin mode is used to determine the priority between the group of channels 0 to 7 and the group of channels 8 to 15.

Immediately after a reset, the order of priority is the same as that of the fixed priority mode, which is as follows.
High CH0 > CH8 > CH1 > CH9 > CH2 > CH10 > CH3 > CH11 > CH4 > CH12 > CH5 > CH13 > CH6 > CH14 > CH7 > CH15 Low

If a transfer request is received from DMA channel 2 in this state, a transfer is executed on DMA channel 2. After the transfer is completed, the order of priority is as follows.
High CH8 > CH3 > CH9 > CH4 > CH10 > CH5 > CH11 > CH6 > CH12 > CH7 > CH13 > CH0 > CH14 > CH1 > CH15 > CH2 Low

The following figure shows an example where DMA transfers are executed in round robin mode.

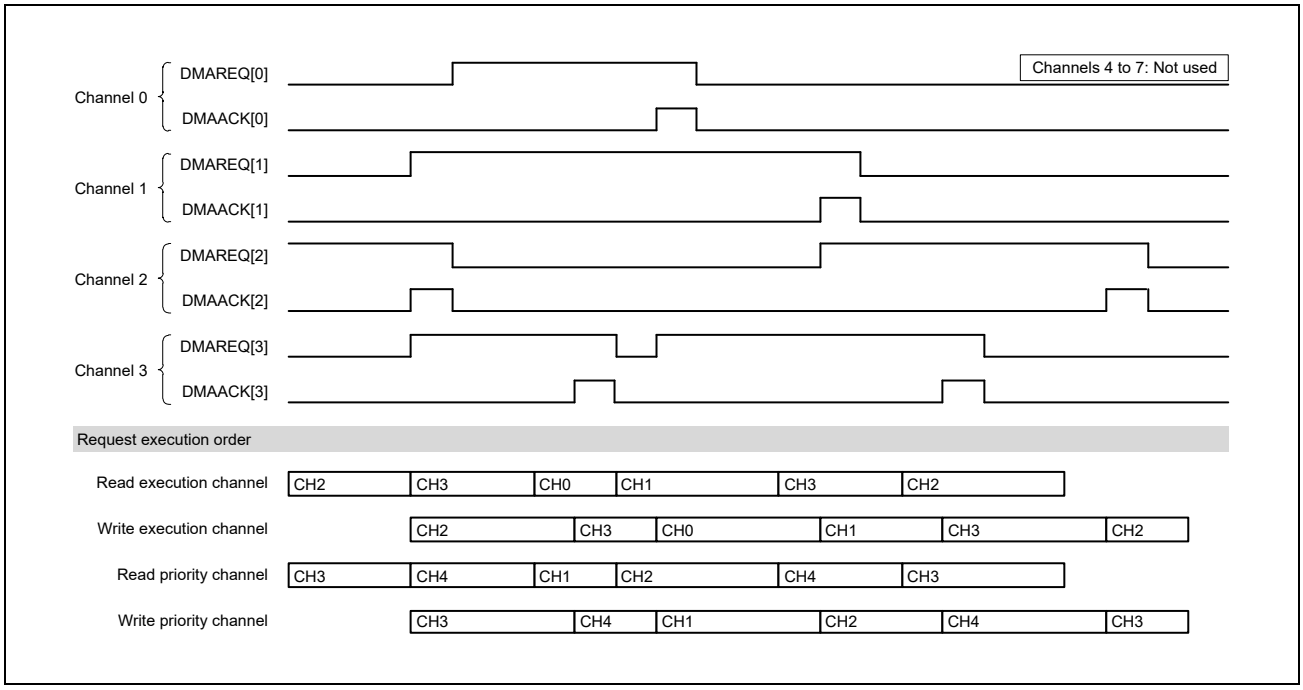


Figure 14.14 Round Robin Mode (Number of Channels = 4, REQD = 0)

The channel whose channel number is the number of the currently transferring channel + 1 gets to execute a DMA transfer next. If there is no transfer request from this channel, the channel whose channel number is the number of this channel + 1 gets to execute a DMA transfer.

14.7.3 DMA Transfer Request

Edge detection or level detection can be selected using the LVL bit of the CHCFG_n/nS register.

The HIEN and LOEN bits of the CHCFG_n/nS register are used to select either the rising edge or falling edge in the case of edge detection or either the high level or low level in the case of level detection.

When the transfer request is by an on-chip peripheral module, set the CHCFG_n/nS register according to **Table 14.3**.

(1) Edge Detection

Setting 0 in the LVL bit of the CHCFG_n/nS register enables edge detection.

(2) Level Detection

Setting 1 in the LVL bit of the CHCFG_n/nS register enables level detection.

14.7.4 DMA Transfer End Interrupt

The DMA transfer end interrupt is an interrupt request signal that indicates that a DMA transaction is completed. There is an independent DMA transfer end interrupt for each channel.

When the transfer of data equivalent to the total transfer byte count loaded to the CRTB (Current Transaction Byte) is completed, 1 is set in END of the CHSTAT_n/nS register. In this case, when 0 is set in DEM of the CHCFG_n/nS register, the DMA transfer end interrupt (DMAINTn_NS/S) is output (n = 0 to 15). (When writeback is performed in link mode, the signal is output after the writeback operation.)

When 0 is set in LV of the header in the read descriptor in link mode, 1 is set in DER of the CHSTAT_n/nS register. In this case, when 0 is set in DIM of the header, the DMA transfer end interrupt is output.

Table 14.17 Assertion Conditions of DMA Transfer End Interrupt

Source	Condition	DMA Transfer End Interrupt Mask Signal
DMA transaction end	When the transfer of data equivalent to the total transfer byte count loaded to the CRTB (Current Transaction Byte) is completed with an OKAY response (or after the writeback operation when writeback is performed in link mode)	DEM bit of the CHCFG_n/nS register
Descriptor invalid	When 0 is set in LV of the header in the read descriptor in link mode while 0 is set in DIM of the header	DIM bit of the header

14.7.5 DMA Error Interrupt

If an error response is received for a DMA transfer or descriptor access, the DMAC regards it as an error and stops the transfer. Upon receiving an error response, the EN bit of the CHSTAT_n/nS register of transferring channel n is cleared to 0 and 1 is set in the ER bit (n = 0 to 15). Also, the DMA error interrupt (DMAERR_NS/S) is output.

The DMA error interrupt cannot be masked.

Once an error occurs, the data of the whole transfer cannot be guaranteed. Be sure to start the transaction again from the beginning by following the procedure below.

1. Set 1 in the SWRST bit of the CHCTRL_n/nS register.
2. Set each register again.

14.7.6 Interval Count Function

The interval at which a DMA transfer is executed can be adjusted by setting the ITVL bit of the channel interval register (CHITVL_n/nS). This function is intended to prevent the DMA controller from occupying the bus all the time.

When a read or write operation is completed, a countdown starts from the value set in CHITVL_n/nS. The next internal request is not executed until the count value reaches 0.

The following figure shows an example of how this works.

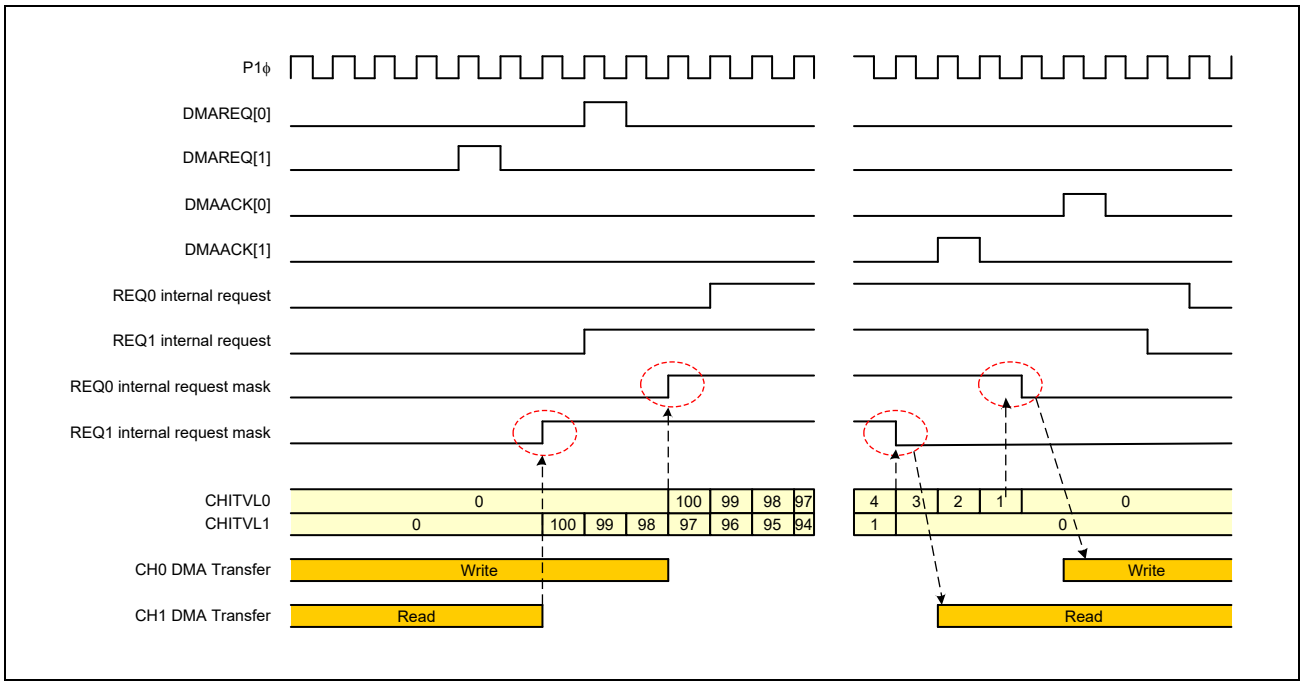


Figure 14.15 Interval Count

14.7.7 Difference in Operation Due to the Transfer Size

(1) When the Source Transfer Size Is Smaller

When the read of data equivalent to the destination data size is completed, the data is written to the destination.

The following figure shows a timing chart where the source transfer size is 8 bits and the destination transfer size is 32 bits (in the case of rising edge detection).

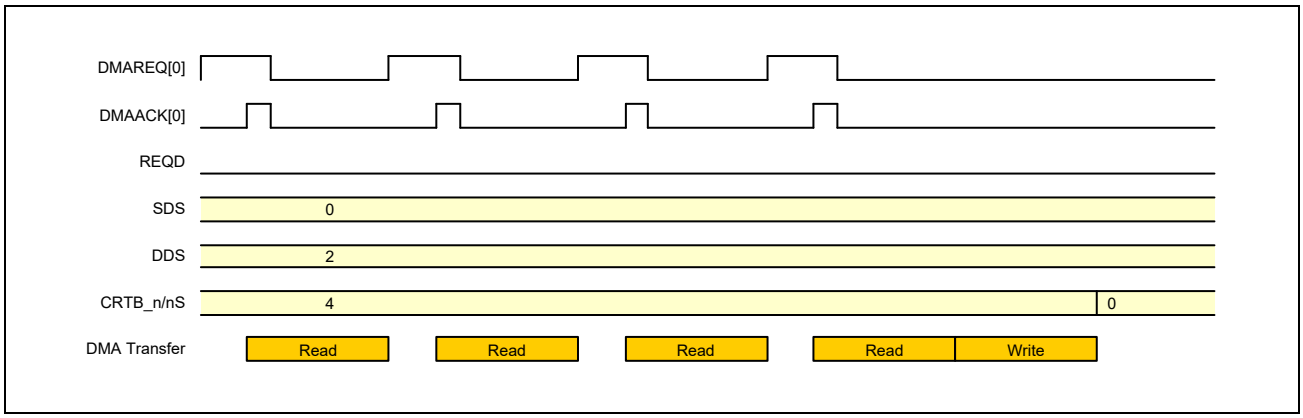


Figure 14.16 When the Source Transfer Size Is Smaller (LVL = 0, HIEN = 1, REQD = 0, SDS < DDS in CHCFG_n/nS)

(2) When the Destination Transfer Size Is Smaller

Since the source transfer size is larger, multiple destination writes occur after a single source read. The following figure shows a timing chart where the source transfer size is 64 bits and the destination transfer size is 16 bits (in the case of rising edge detection) (1 is set in REQD of the CHCFG_n/nS register).

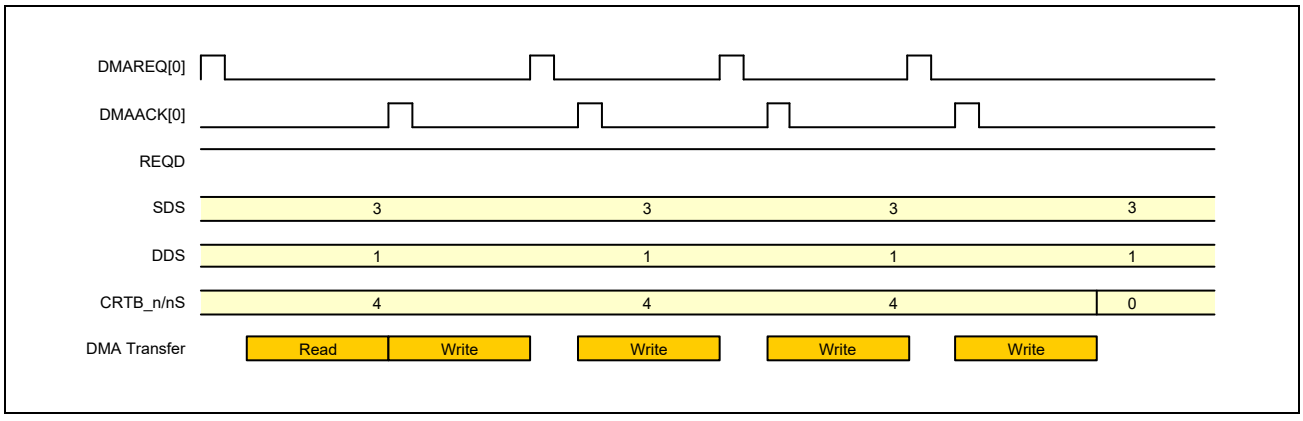


Figure 14.17 When the Destination Transfer Size Is Smaller (LVL = 0, HIEN = 1, REQD = 1, SDS > DDS in CHCFG_n/nS)

(3) When the Source Transfer Size Is the Same as the Destination Transfer Size

Every time a DMA transfer request is detected, a source read and a destination write occur.

The following figure shows a timing chart where the source transfer size and the destination transfer size are both 8 bits (in the case of rising edge detection, with 1 set in REQD of the CHCFG_n/nS register).

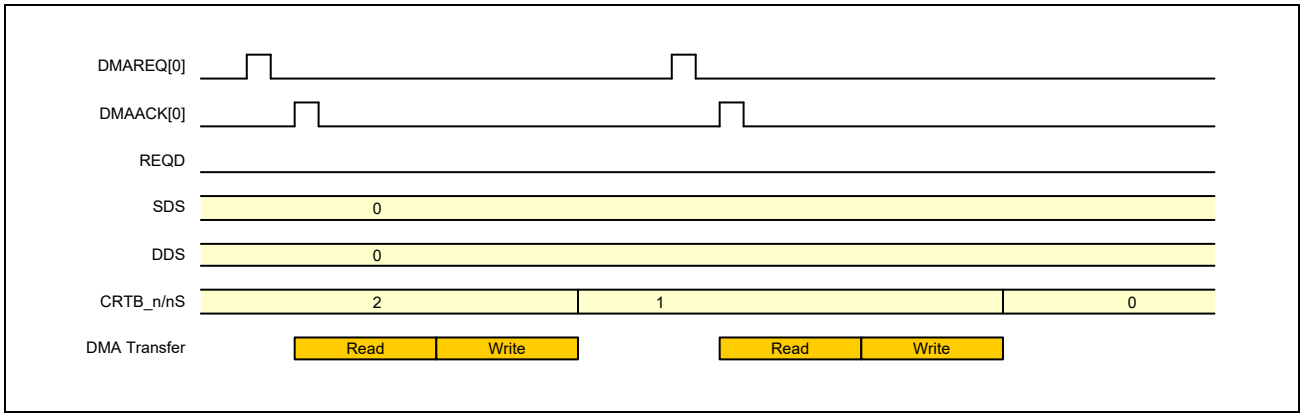


Figure 14.18 When the Source Transfer Size Is the Same as the Destination Transfer Size (LVL = 0, HIEN = 1, REQD = 0, SDS = DDS in CHCFG_n/nS)

14.7.8 Transfer Status

The channel status register indicates the status of DMA transfer execution on a channel.

(1) Suspend

A DMA transfer can be suspended by using the SETSUS bit of CHCTRL_n/nS. In this case, if an ongoing bus cycle exists, the DMAC waits for that cycle to end before suspending the transfer. Writing 1 in the CLRSUS bit restores the DMA transfer from the suspend status.

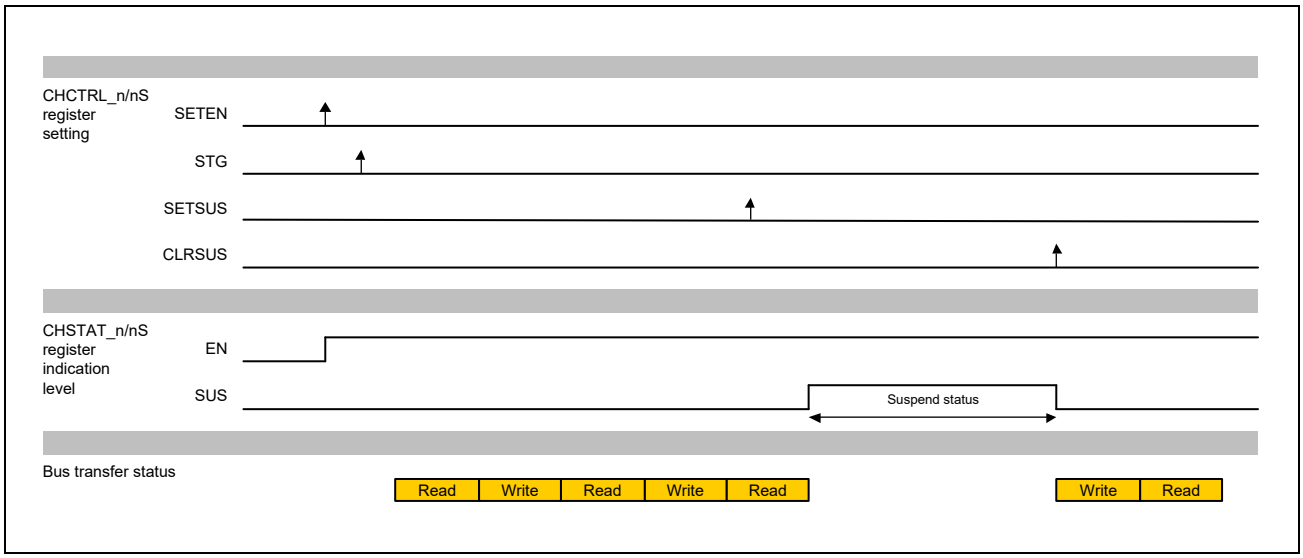


Figure 14.19 DMAC Suspend Status (Auto Request/Block Transfer)

In the above case, the DMA transfer is suspended after the read transfer is completed.

If there is any ongoing DMA transfer, the suspend status starts when that transfer is completed. To make sure that the transfer is suspended, read the CHSTAT or DSTAT_SUS register, after setting SETSUS, and check that 1 is set in the SUS bit for the relevant channel.

(2) Transfer Stop

If 1 is written to CLREN while a DMA transaction is in progress, the DMA transaction for the corresponding channel can be stopped. For the post-stop processing, two modes are supported: one sweeps out the data remaining in the buffer when the transaction is stopped (SBE = 1) and the other does not (SBE = 0). One of these modes can be selected using the SBE bit of the CHCFG_n/nS register. By default, SBE is set to 0.

When this sweep mode is enabled and CLREN is set to 1, and if a DMA transaction is stopped with data remaining in the DMAC buffer, the transaction is completed after the DMAC sweeps the data.

(a) Transfer Stop (Buffer Sweep Disabled - SBE = 0)

If 1 is set in CLREN during a DMA transfer, the DMA transfer is stopped. The stop timing depends on the value set in REQD. After stopping a DMA transfer, be sure to set 1 in SWRST to clear the DMA internal status before setting the next transfer.

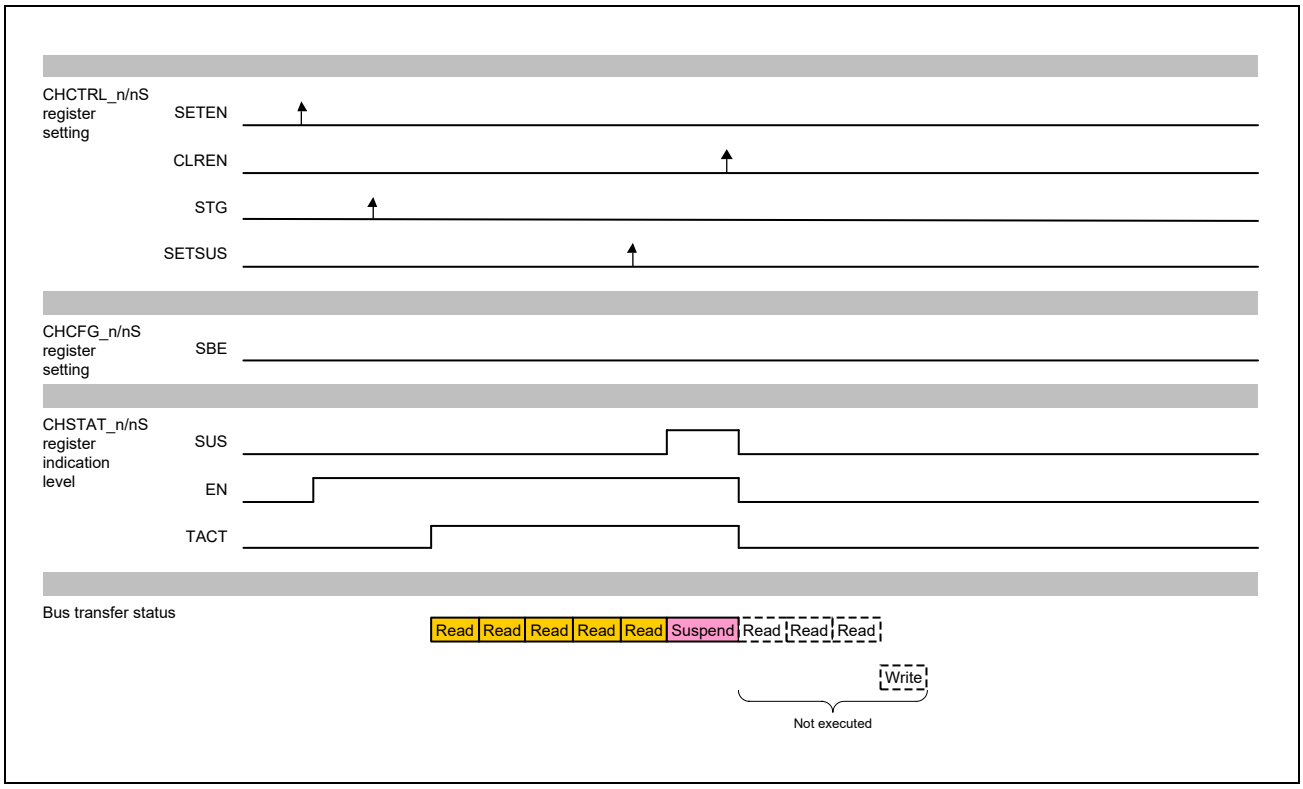


Figure 14.20 DMAC Transfer Stop

- The setting of the TACT bit being 0 indicates that the channel has been brought to a complete stop.
- If an ongoing DMA transfer is stopped before it is completed, the DMA transfer end interrupt is not asserted.
- If 0 is set in REQD, the DMA transfer is stopped when the next read is completed. (If the buffer contains any data that can be written, the DMA transfer is stopped after the data is written.)
- If 1 is set in REQD, the DMA transfer is stopped when the next write is completed.

(b) Transfer Stop (Buffer Sweep Enabled - SBE = 1)

If 1 is set in CLREN during a DMA transfer, the DMA transfer is stopped. When 0 is set in REQD, the DMA transfer is stopped after the DMAC sweeps (writes) the already read data. If 1 is set in REQD to use hardware requests, do not use the sweep mode. After stopping a DMA transfer, be sure to set 1 in SWRST to clear the DMAC internal status before setting the next transfer.

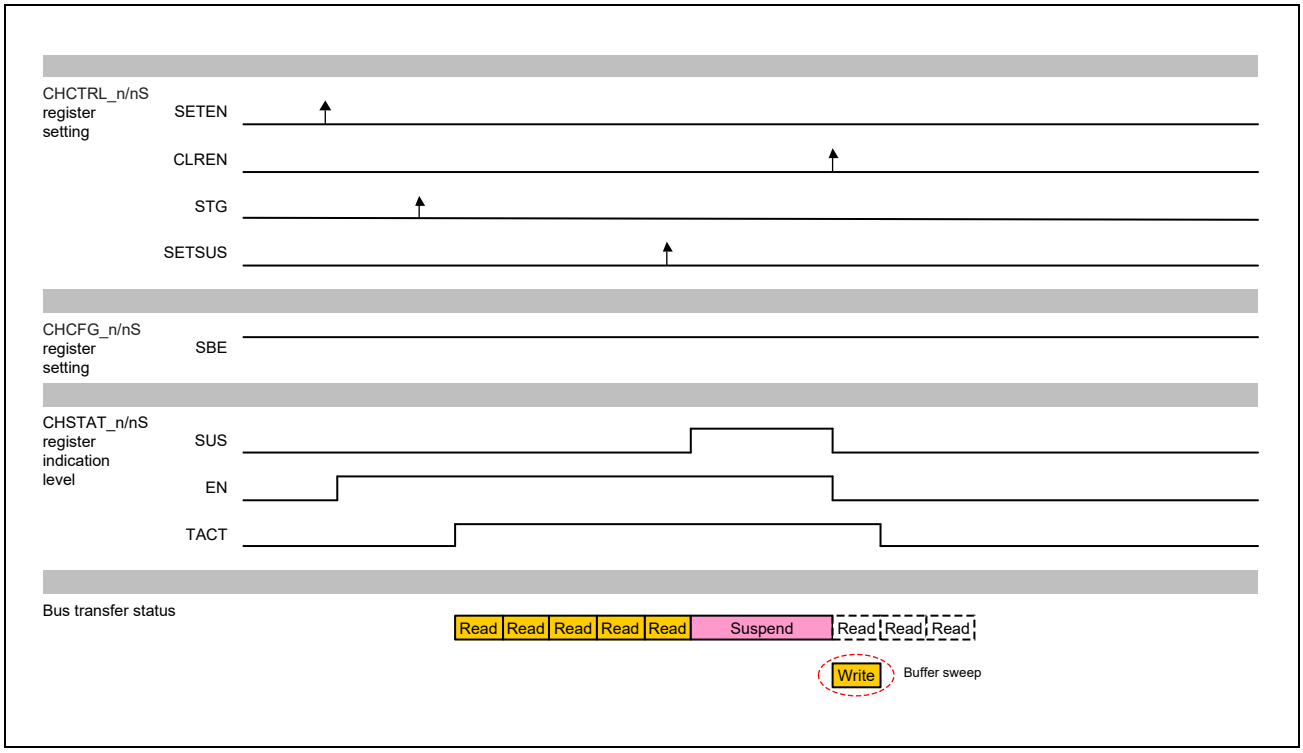


Figure 14.21 DMA Transfer Stop (Buffer Sweep Mode)

- The setting of the TACT bit being 0 indicates that the channel has been brought to a complete stop.
- If a transfer is stopped in sweep mode (SBE = 1) during the fifth read transfer by setting SETSUS and then CLREN, the read data is written before the DMA transfer is stopped.

(c) Channel Stop Check Method

Even when the EN bit is cleared to 0, the DMA transfer cannot be stopped immediately, if the bus is already executing the transfer. Therefore, in order to make sure that the DMAC has been brought to a complete stop, check that the EN bit and TACT bit are both set to 0.

(d) Transfer Stop Procedure

The transfer stop procedure is described below.

1. Set 1 in SETSUS of CHCTRL_n/nS.
2. Repeat polling until the SUS bit of CHSTAT_n/nS is set to 1. (If EN is already set to 0, the DMAC has already been stopped. Go to step 6.)
3. Set 1 in CLREN of CHCTRL_n/nS.

4. When 0 is set in SBE, the transfer is stopped according to the value of REQD. When 1 is set in SBE, the sweep mode is enabled. When 1 is set in SBE, set 0 in REQD.
5. Read CHSTAT_n/nS to check that 0 is set in the TACT bit. When TACT is set to 0, it means that the DMAC has been brought to a complete stop. When TACT is set to 1, repeat polling until this bit is set to 0.
6. To execute the next DMA transfer after stopping a transfer, be sure to set 1 in the SWRST (software reset) bit of CHCTRL_n/nS before the next DMA transfer starts.

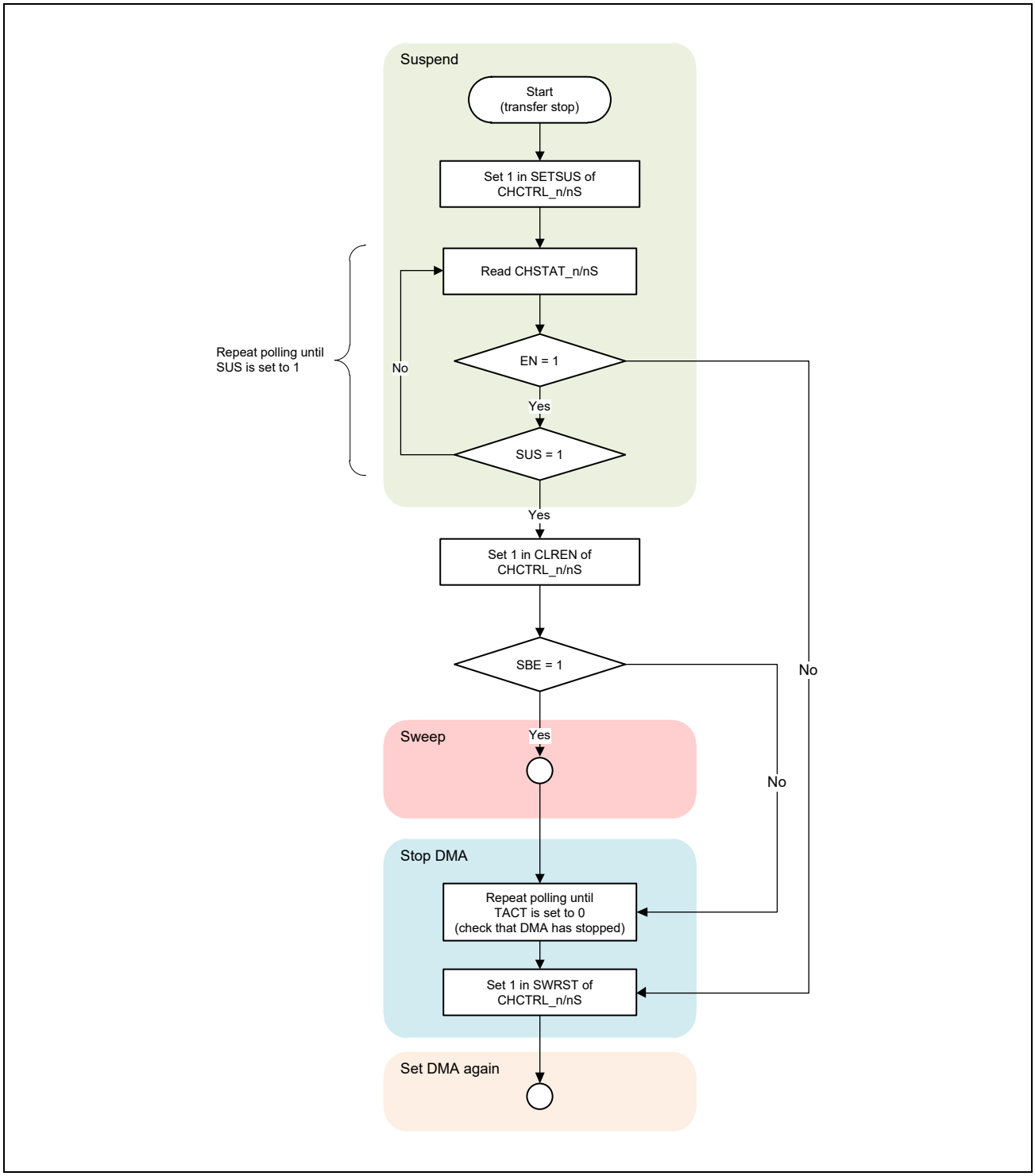


Figure 14.22 Transfer Stop Flow

14.8 DMA Setting Examples

Setting examples applicable when DMA transfer is executed using the direct memory access controller are shown in the following.

The transfer conditions for these setting examples are as follows.

Table 14.18 Transfer Condition List for DMA Transfer Setting Examples

	DMA Mode	Transfer Mode	Transfer Request
Setting example 1	Register	Single	Hardware
Setting example 2	Register	Block	Software
Setting example 3	Register (continuous execution)	Block	Software
Setting example 4	Link	Block	Software

For details of the settings, see the individual setting examples.

14.8.1 Setting Example 1 (Register Mode/Hardware Request)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

Table 14.19 DMA Transfer Setting Example 1

Item		Description	
Channel used		3	
DMA mode		Register	
Transfer mode		Single transfer	
Register set used		Next0	
Source/destination		Source	Destination
	Start address	H’1111_0000	H’2222_0000
	Address direction	Increment	Increment
	Data size	32 bits	32 bits
DMA transfer byte count		64 bytes	
DMA transfer request		Rising edge detection by hardware	
DMAACK signal		Level output during read	
DMA transfer end interrupt mask		Not masked	
CACHE setting		Default value	

Setting example 1

N0SA = H’1111_0000 (source address)
N0DA = H’2222_0000 (destination address)
N0TB = H’0000_0040 (transfer byte count)

CHCFG = H’0002_2123 (configuration)
CHITVL = H’0000_0000 (interval)
CHEXT = H’0000_0000 (CACHE setting)

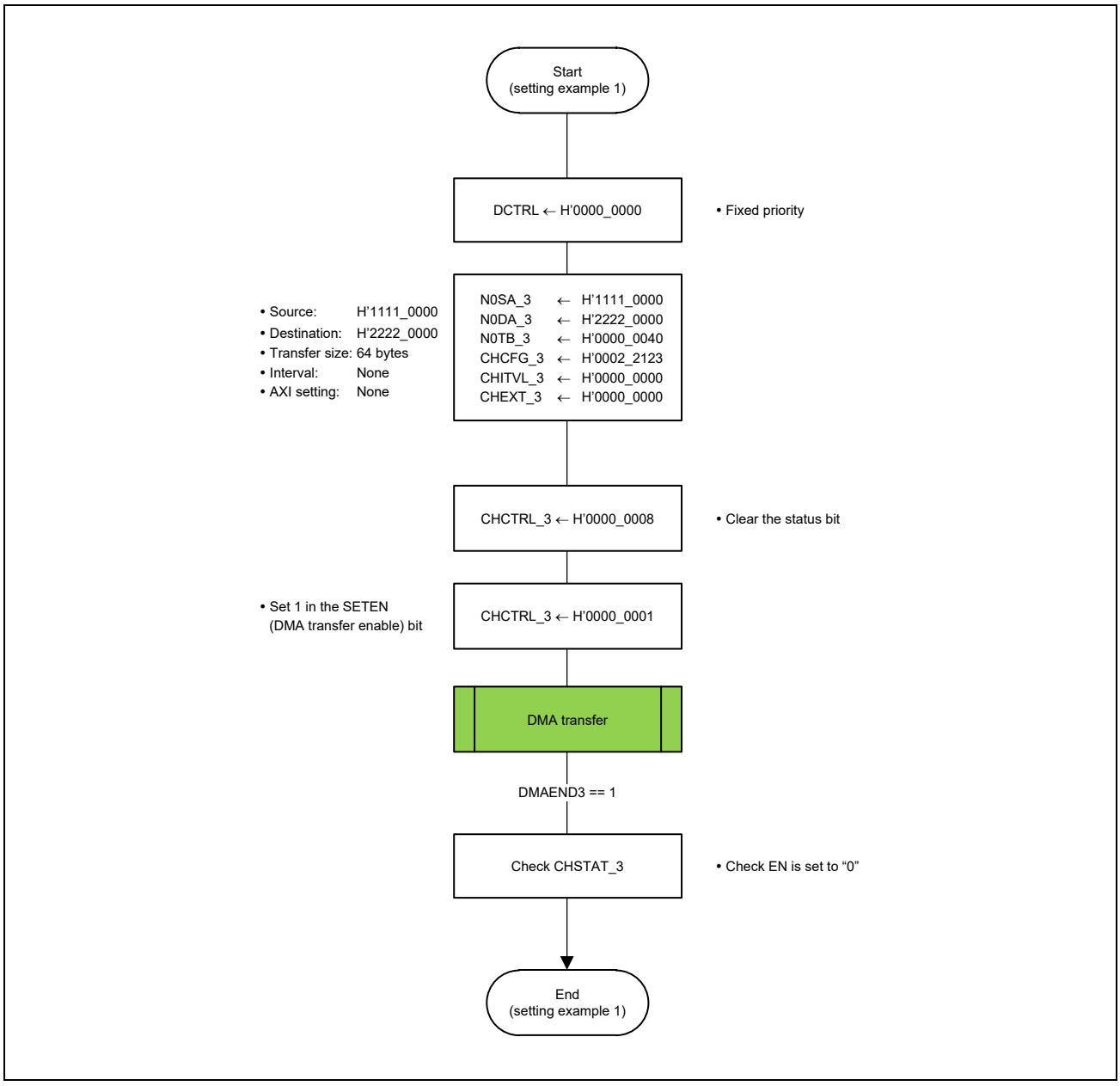


Figure 14.23 Setting Example 1

14.8.2 Setting Example 2 (Register Mode/Software Request)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

Table 14.20 DMA Transfer Setting Example 2

Item		Description	
Channel used		2	
Priority control		Round robin	
DMA mode		Register	
Transfer mode		Block transfer	
Register set used		Next1	
Source/destination		Source	Destination
	Start address	H'0FFF_E000	H'3333_0000
	Address direction	Increment	Increment
	Data size	8 bits	256 bits
DMA transfer byte count		128 bytes	
DMA transfer request		Auto request	
DMAACK signal		Masked	
DMA transfer end interrupt mask		Not masked	
CACHE setting		Default value	

Setting example 2

DCTRL = H'0000_0001 (DMA setting)

N1SA = H'0FFF_E000 (source address)
N1DA = H'3333_0000 (destination address)
N1TB = H'0000_0080 (transfer byte count)

CHCFG = H'1045_0402 (configuration)
CHITVL = H'0000_0000 (interval)
CHEXT = H'0000_0000 (CACHE setting)

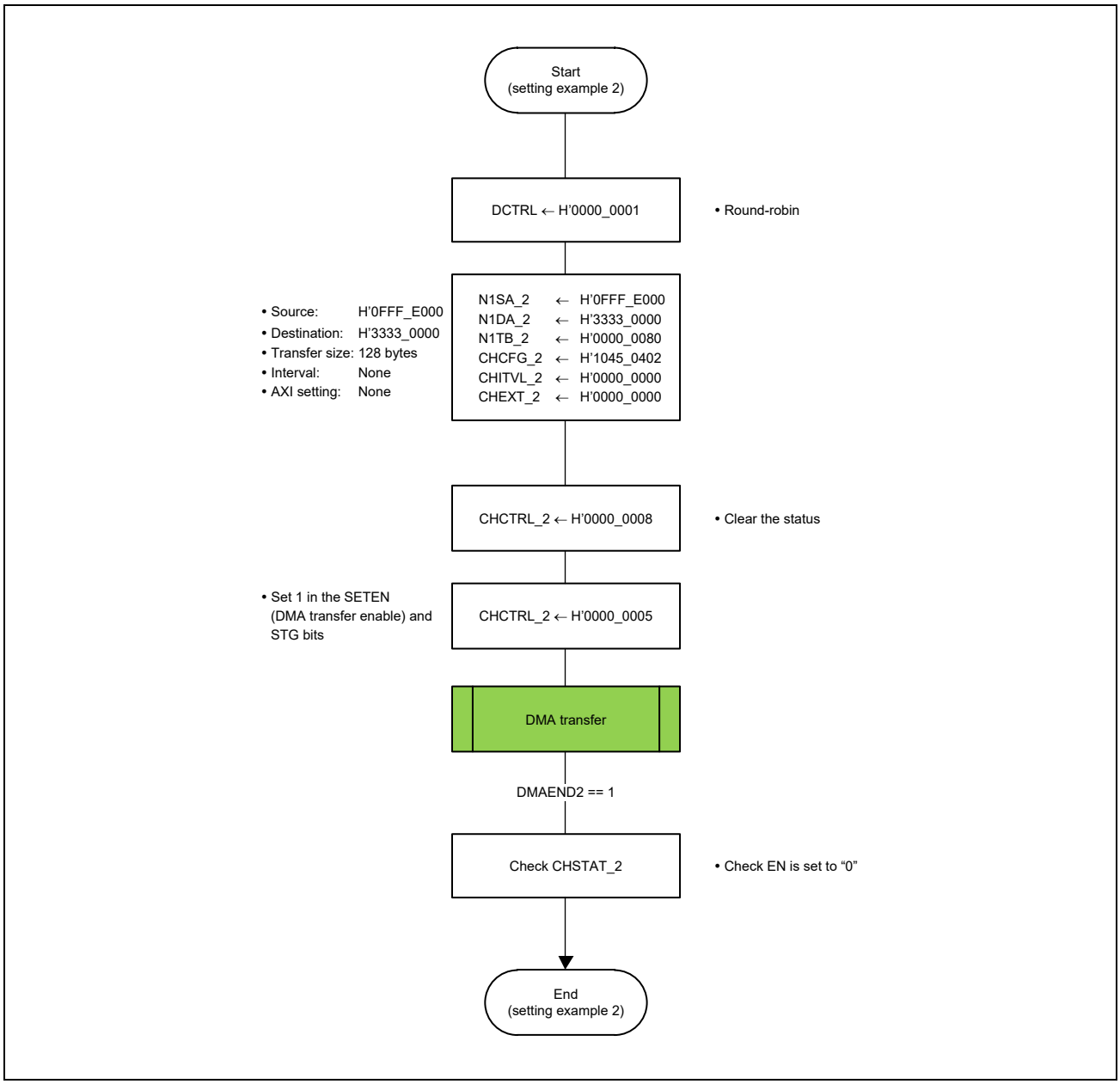


Figure 14.24 Setting Example 2

14.8.3 Setting Example 3 (Register Mode/Continuous Execution)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

Table 14.21 DMA Transfer Setting Example 3

Item		Description	
Channel used		1	
Priority control		Round robin	
DMA mode		Register	
Transfer mode		Block transfer	
Register set used		Use Next0 and then Next1 continuously	
Next0		Source	Destination
	Start address	H'1111_0000	H'3333_0000
	Address direction	Fixed	Fixed
	Data size	32 bits	512 bits
	DMA transfer byte count	512 bytes	
Next1		Source	Destination
	Start address	H'2222_0000	H'4444_0000
	Address direction	Fixed	Fixed
	Data size	32 bits	512 bits
	DMA transfer byte count	2048 bytes	
DMA transfer request		Auto request	
DMAACK signal		Not output	
DMA transfer end interrupt mask		Mask the DMA transfer end interrupt upon completion of Next0	
CACHE setting		Default value	

Setting example 3

DCTRL = H'0000_0001 (DMA setting)

N0SA = H'1111_0000 (source address)
N0DA = H'3333_0000 (destination address)
N0TB = H'0000_0200 (transfer byte count)

N1SA = H'2222_0000 (source address)
N1DA = H'4444_0000 (destination address)
N1TB = H'0000_0800 (transfer byte count)

CHCFG = H'6176_2001 (configuration)
CHITVL = H'0000_0000 (interval)
CHEXT = H'0000_0000 (CACHE setting)

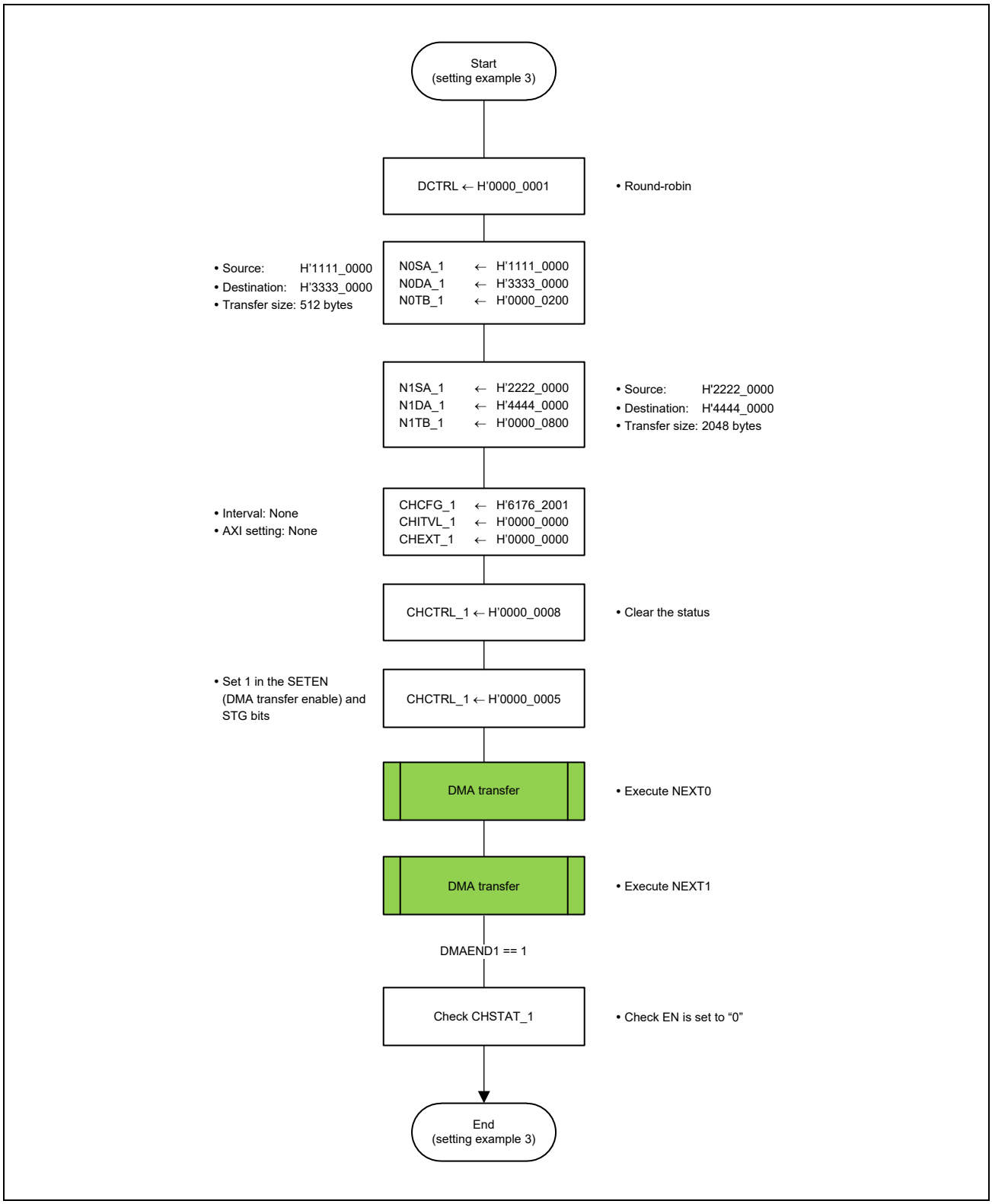


Figure 14.25 Setting Example 3

14.8.4 Setting Example 4 (Link Mode)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

Table 14.22 DMA Transfer Setting Example 4

Item	Description
Channel used	0
Priority control	Round robin
DMA mode	Link
Transfer mode	Block transfer
Register set used	—
Descriptor start address	H'0000_1000

Table 14.23 DMA Transfer Setting Example 4 (Descriptor 1)

Item		Description	
Descriptor start address		H’0000_1000	
Next descriptor start address		H’0000_2000	
Transfer mode		Block transfer	
Next0		Source	Destination
	Start address	H’1111_0000	H’3333_0000
	Address direction	Increment	Increment
	Data size	32 bits	32 bits
	DMA transfer byte count	2048 bytes	
DMA transfer request		Auto request trigger (STG)	
DMAACK signal		Not output	
DMA transfer end interrupt mask		Masked	
CACHE setting		Default value	
header			
	DMA interrupt when LV = 1	Issued (DIM = 0)	
	LV writeback	Done (WBD = 0)	
	Next link address	Available (LE = 0)	
	Descriptor valid	Valid (LV = 1)	

Table 14.24 DMA Transfer Setting Example 4 (Descriptor 2)

Item		Description	
Descriptor start address		H'0000_2000	
Next descriptor start address		H'0000_5000	
Transfer mode		Block transfer	
Next0		Source	Destination
	Start address	H'4444_0000	H'5555_0000
	Address direction	Increment	Increment
	Data size	64 bits	256 bits
	DMA transfer byte count	1024 bytes	
DMA transfer request		Auto request trigger (STG)	
DMAACK signal		Not output	
DMA transfer end interrupt mask		Masked	
CACHE setting		Default value	
header			
	DMA interrupt when LV = 1	Issued (DIM = 0)	
	LV writeback	Done (WBD = 0)	
	Next link address	Available (LE = 0)	
	Descriptor valid	Valid (LV = 1)	

Table 14.25 DMA Transfer Setting Example 4 (Descriptor 3)

Item		Description	
Descriptor start address		H'0000_5000	
Next descriptor start address		—	
Transfer mode		Block transfer	
Next0		Source	Destination
	Start address	H'7777_0000	H'AAAA_0000
	Address direction	Increment	Increment
	Data size	512 bits	512 bits
	DMA transfer byte count	4096 bytes	
DMA transfer request		Auto request trigger (STG)	
DMAACK signal		Not output	
DMA transfer end interrupt mask		Not masked	
CACHE setting		Default value	
header			
	DMA interrupt when LV = 1	Issued (DIM = 0)	
	LV writeback	Done (WBD = 0)	
	Next link address	Not available (LE = 1)	
	Descriptor valid	Valid (LV = 1)	

Setting example 4

DCTRL= H'0000_0001 (DMA setting)
NXLA = H'0000_1000 (descriptor start address)
CHCFG = H'8000_0000 (configuration)

Table 14.26 Descriptor Setting

	Descriptor 1	Descriptor 2	Descriptor 3
header	H'0000_0001	H'0000_0001	H'0000_0003
SA (Source Address)	H'1111_0000	H'4444_0000	H'7777_0000
DA (Destination Address)	H'3333_0000	H'5555_0000	H'AAAA_0000
TB (Transaction Byte)	H'0000_0800	H'0000_0400	H'0000_1000
CFG (Configuration)	H'8142_2008	H'8145_3008	H'8046_6008
ITVL (Interval)	H'0000_0000	H'0000_0000	H'0000_0000
EXT (Extension)	H'0000_0000	H'0000_0000	H'0000_0000
NXLA (Next Link Address)	H'0000_2000	H'0000_5000	H'0000_0000

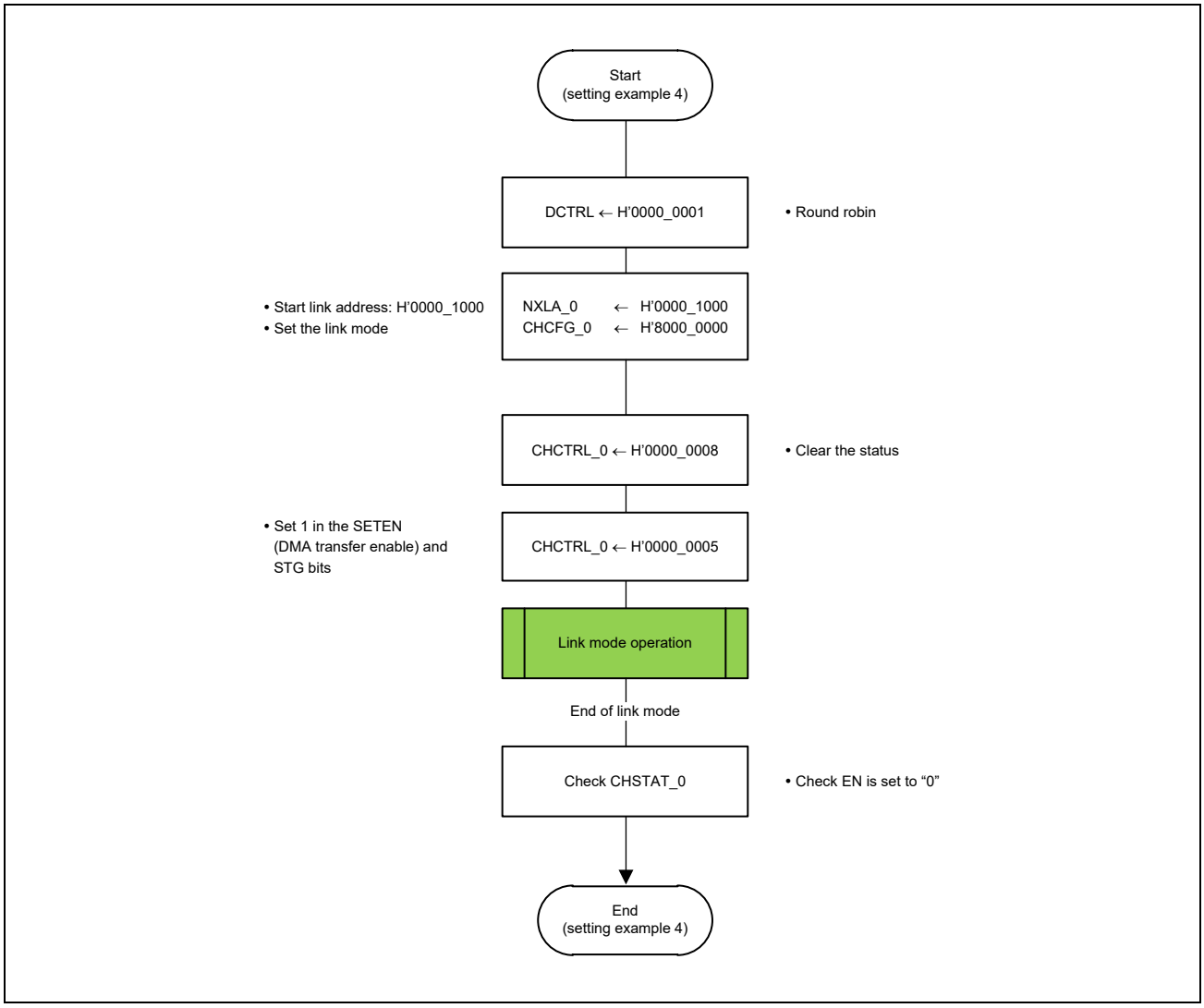


Figure 14.26 Setting Example 4

14.8.5 Next Register Set Continuous Execution Setting

The following figure shows the flowchart for executing DMA transfers continuously by using two Next register sets in register mode. While a DMA transaction is being executed using one Next register set, the other Next register set is set in order to continue to execute DMA transfers.

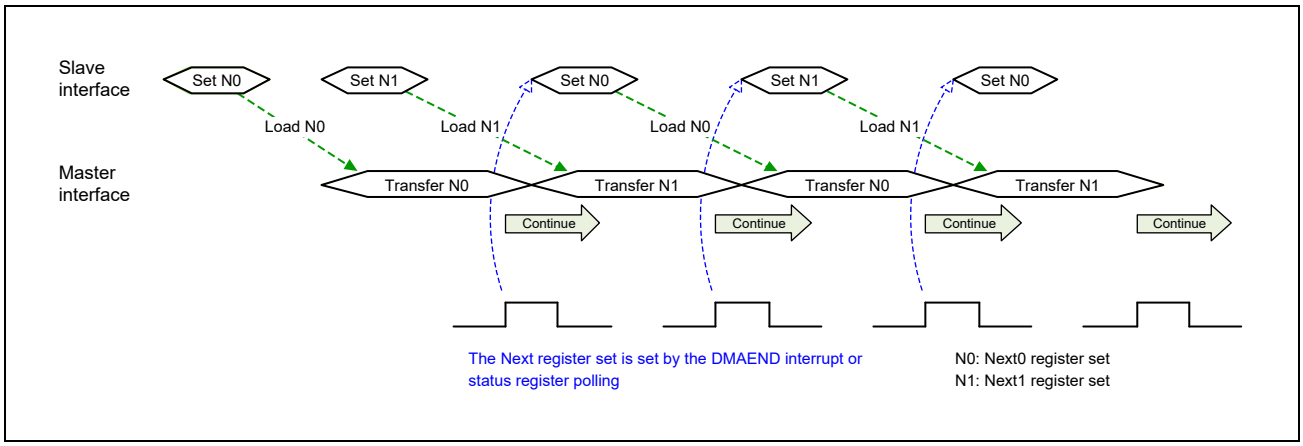


Figure 14.27 Image of Next Register Set Continuous Execution

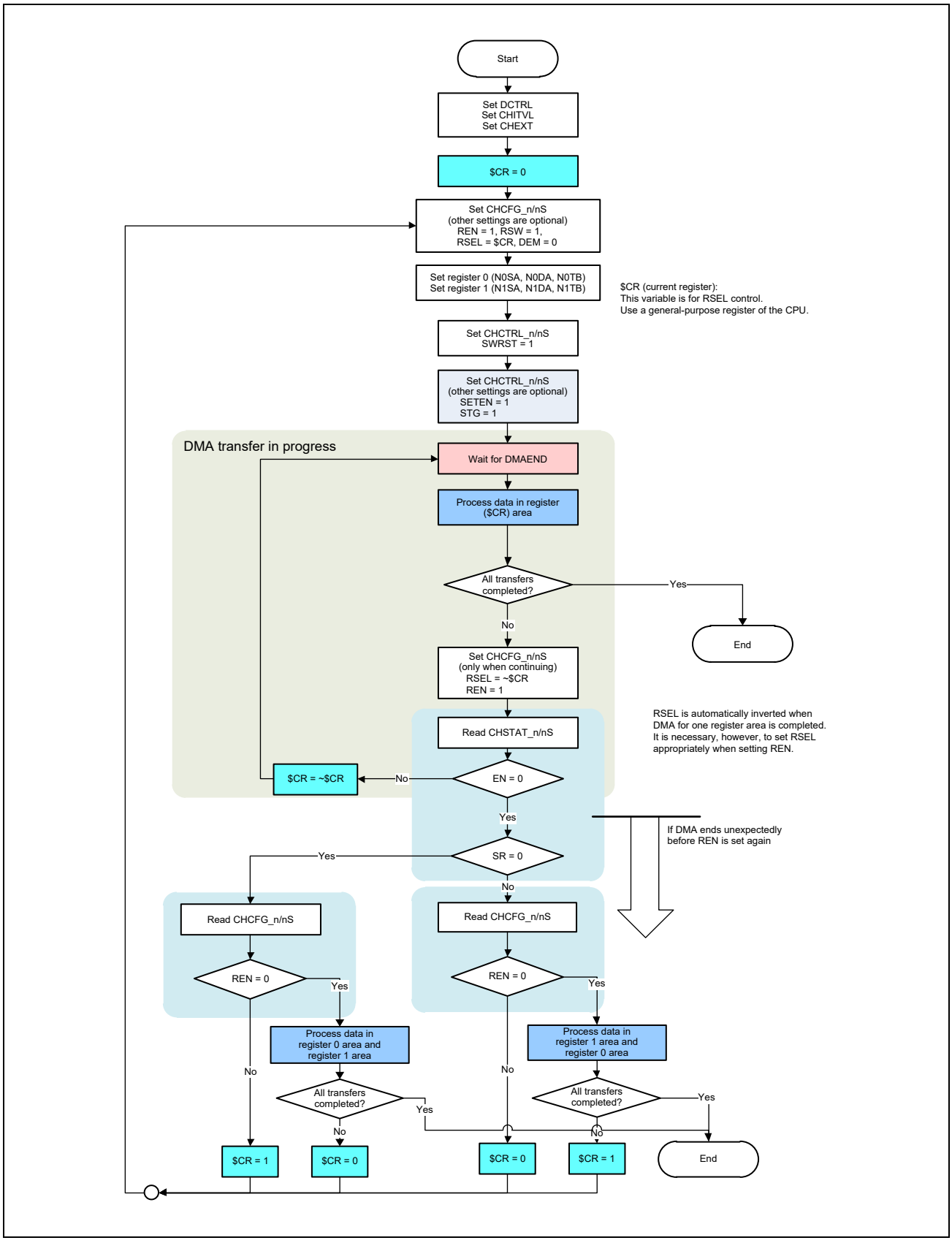


Figure 14.28 Example of Continuous DMA Execution by Using a Next Register Set

- Supplementary information

  First, save the data of the register sets to be used for DMA transfers (0 (N0SA, N0DA, and N0TB) and 1 (N1SA, N1DA, and N1TB)) to a general-purpose register of the CPU (the values of this register is referred to as $CR for the sake of convenience).

  Each time the DMA transfers for one register set are completed (the DMA transfer end interrupt is output), REN is automatically cleared to 0. In order to continue to execute DMA transfers, it is necessary to set REN of the CHCFG_n/nS register every time the DMA transfer end interrupt is asserted. This register also contains the RSEL bit, and the value of this bit needs to be set appropriately as well. Therefore, use $CR.

  In this mode, two Next register sets are executed continuously. However, if CLREN is not set before the DMA transaction is completed (the next DMA transfer end interrupt is output), continuous execution stops. In this case, how much of data has been transferred can be checked by reading the SR and EN bits of the CHSTAT_n/nS register and the REN bit of the CHCFG_n/nS register. To restart the DMA transaction, follow the flowchart shown above.

15. System Counter (SYC)

SYC generates the count value used by the generic timer built into Cortex-A55.

It uses the Timestamp generator, which is one of the components of Arm CoreSight SoC-400, to generate a 64-bit count value. For more information on the Timestamp generator, please refer to *the ARM CoreSight SoC-400 Technical Reference Manual*.

15.1 Overview

15.1.1 Features

- Count by 24 MHz clock (SYC_CNT_CLK)
- 64 bits gray code counter value generation
  - Convert the value generated by Timestamp generator to Gray code and output
- Access control
  - Access control for the two interfaces of the Timestamp generator
  - Control secure access/non-secure access
- Halt on Debug function
  - Counting can be stopped/restarted during debugging according to request from CoreSight

15.1.2 Block Diagram of SYC

The SYC connection diagram is shown in **Figure 15.1** and the SYC block diagram is shown in **Figure 15.2**.

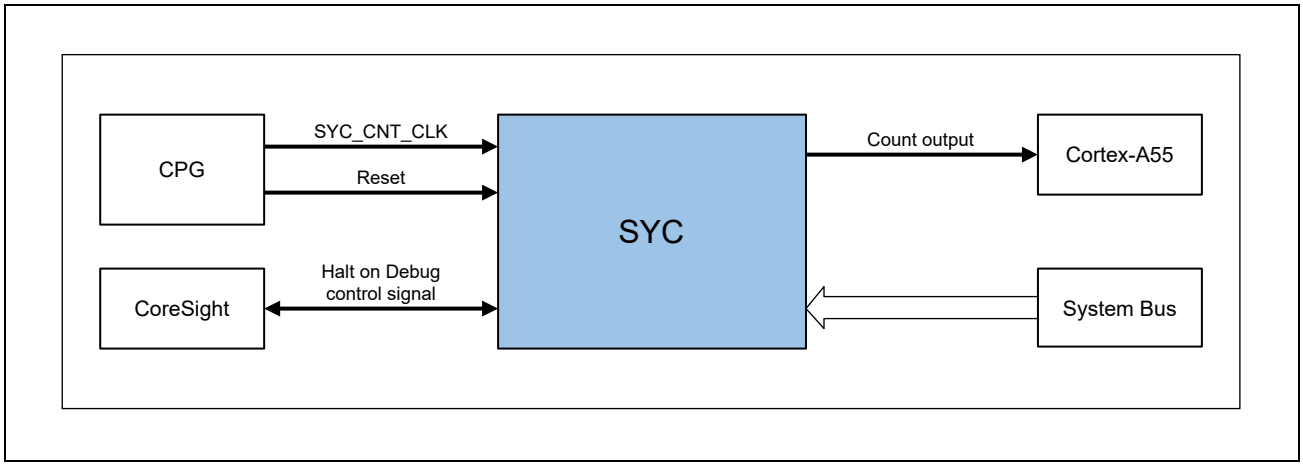


Figure 15.1 SYC Connection Diagram

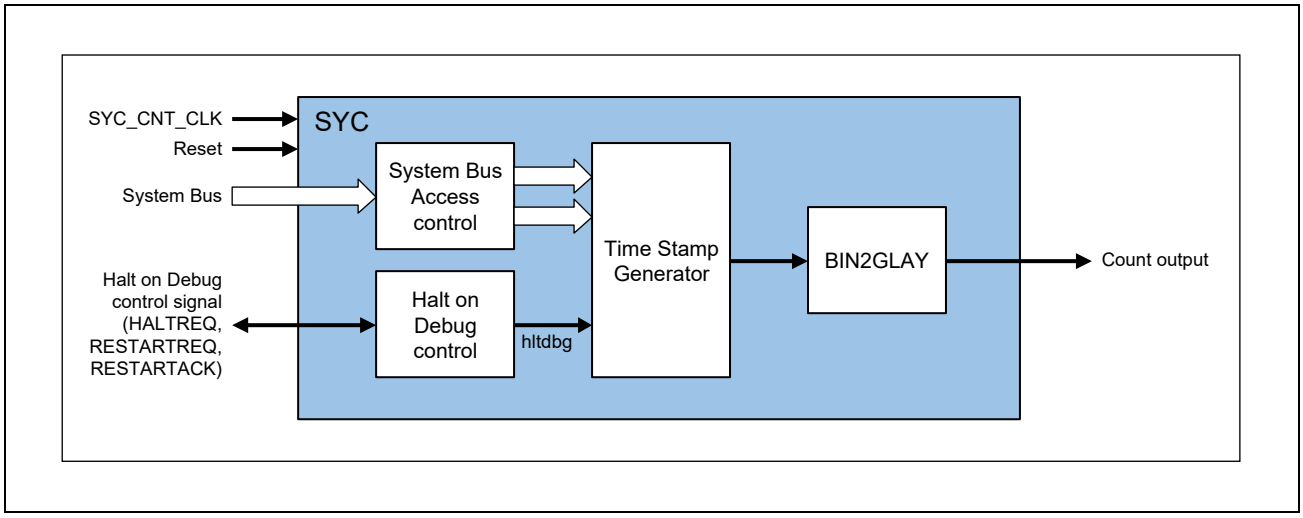


Figure 15.2 SYC Block Diagram

15.2 Address Space

Table 15.1 shows the SYC address space. SYC has an address space of 8 KB. The SYC address space is offset from the base address. The base address of SYC is as follows.

SYC base address: H'0_1100_0000 (Overall Address Space)
SYC base address: H'4100_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
SYC base address: H'5100_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 15.1 SYC Address Space

Offset Address	Region Name	Access Target
H'0000 to H'0FFF	PSELCTRL region	PSELCTRL region & PSELCTRL region Management registers
H'1000 to H'1FFF	PSELREAD region	PSELREAD region & PSELREAD region Management registers

Note: Secure access and non-secure access are possible for the PSELCTRL region and PSELREAD region.

15.3 Register Descriptions

For more information on SYC registers, please refer to *the ARM CoreSight SoC-400 Technical Reference Manual 3.10 Timestamp generator*.

15.4 Operation

15.4.1 System Bus Access Control

Access control from one system bus interface of SYC to two APB interfaces of Timestamp Generator is performed.

Controls so that the PSELCTRL area can be accessed when accessing the first half 4 KB of the 8 KB address area of SYC, and the PSELREAD area can be accessed when accessing the latter 4 KB.

The PSELCTRL area and PSELREAD area accept secure access and non-secure access.

15.4.2 Halt on Debug Function

SYC supports the Halt on Debug function included in Timestamp Generator. When HALTREQ is asserted from CoreSight, the counting operation of Timestamp Generator is stopped.

When RESTARTREQ is asserted from CoreSight, the counting operation of Timestamp Generator is restarted.

HALTREQ and RESTARTREQ are controlled by CoreSight's CTI (Cross Trigger Interface).

16. Multi-Function Timer Pulse Unit 3 (MTU3a)

16.1 Overview

This LSI has an on-chip multi-function timer pulse unit 3 (MTU3a), consisting of eight 16-bit timer channels and one 32- bit timer channel.

Table 16.1 shows the specifications of the MTU and **Table 16.2** lists the functions of the MTU. **Figure 16.1** and **Figure 16.2** are block diagrams of the MTU.

Table 16.1 MTU Specifications (1/2)

Item	Description
Pulse input/output	28 lines max.
Pulse input	3 lines
Count clock	11 clocks for each channel (14 clocks for MTU0, 12 clocks for MTU2, and 10 clocks for MTU5, four clocks for MTU1-MTU2 combination (when LWA = 1))
Operating frequency	Up to 100 MHz
Available operations	[MTU0 to MTU4, MTU6, MTU7, and MTU8] • Waveform output on compare match • Input capture function (noise filter setting available) • Counter-clearing operation • Simultaneous writing to multiple timer counters (TCNT) (excluding MTU8) • Simultaneous clearing on compare match or input capture (excluding MTU8) • Simultaneous input and output to registers in synchronization with counter operations (excluding MTU8) • Up to 12-phase PWM output in combination with synchronous operation (excluding MTU8)
	[MTU0 MTU3, MTU4, MTU6, MTU7, and MTU8] • Buffer operation specifiable
	[MTU1, MTU2] • Phase counting mode can be specified independently • 32-bit phase counting mode can be specified for interlocked operation of MTU1 and MTU2 (when TMDR3.LWA = 1) • Cascade connection operation available
	[MTU3, MTU4, MTU6, and MTU7] • Through interlocked operation of MTU3/4 and MTU6/7, the positive and negative signals in six phases (12 phases in total) can be output in complementary PWM and reset-synchronized PWM operation. • In complementary PWM mode, values can be transferred from buffer registers to temporary registers at crests and troughs of the timer-counter values or when the buffer registers (TGRD registers in MTU4 and MTU7) are written to. • Double-buffering selectable in complementary PWM mode
	[MTU3 and MTU4] • Through interlocking with MTU0, a mode for driving AC synchronous motors (brushless DC motors) by using complementary PWM output and reset-synchronized PWM output is settable and allows the selection of two types of waveform output (chopping or level)
	[MTU5] • Capable of operation as a dead-time compensation counter
	[MTU0/MTU5, MTU1, MTU2, and MTU8] 32-bit phase counting mode specifiable by combining MTU1 and MTU2 and through interlocked operation with MTU0/MTU5 and MTU8
Interrupt-skipping function	• In complementary PWM mode, interrupts on crests and troughs of counter values and triggers to start conversion by the A/D converter can be skipped
Interrupt sources	43 sources
Buffer operation	Automatic transfer of register data (transfer from the buffer register to the timer register)

Table 16.1 MTU Specifications (2/2)

Item	Description
Trigger generation	A/D converter start triggers can be generated
	A/D converter start request delaying function enables A/D converter to be started with any desired timing and to be synchronized with PWM output
Low power consumption function	The MTU3a can be placed in the module-stop state.

Table 16.2 MTU Functions (1/2)

Item		MTU0	MTU1	MTU2	MTU1 & MTU2 (LWA = 1)	MTU3	MTU4	MTU5	MTU6	MTU7	MTU8
Count clock		P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB MTCLKC MTCLKD MTIOC1A	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB MTCLKC	MTCLKA MTCLKB MTCLKC MTCLKD	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTIOC1A	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB
External clocks in phase-counting mode		—	MTCLKA MTCLKB	MTCLKA MTCLKB MTCLKC MTCLKD	MTCLKA MTCLKB MTCLKC MTCLKD	—	—	—	—	—	—
General registers (TGR)		TGRA TGRB TGRE	TGRA TGRB	TGRA TGRB	TGRALW TGRBLW	TGRA TGRB	TGRA TGRB	TGRU TGRV TGRW	TGRA TGRB	TGRA TGRB	TGRA TGRB
General registers/ buffer registers		TGRC TGRD TGRF	—	—	—	TGRC TGRD TGRE	TGRC TGRD TGRE TGRF	—	TGRC TGRD TGRE	TGRC TGRD TGRE TGRF	TGRC TGRD
I/O pins		MTIOC0A MTIOC0B MTIOC0C MTIOC0D	MTIOC1A MTIOC1B	MTIOC2A MTIOC2B	MTIOC1A MTIOC1B	MTIOC3A MTIOC3B MTIOC3C MTIOC3D	MTIOC4A MTIOC4B MTIOC4C MTIOC4D	MTIC5U MTIC5V MTIC5W	MTIOC6A MTIOC6B MTIOC6C MTIOC6D	MTIOC7A MTIOC7B MTIOC7C MTIOC7D	MTIOC8A MTIOC8B MTIOC8C MTIOC8D
Counter clear function		TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGRALW/ TGRBLW input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture
Compare match output	0 output	✓	✓	✓	—	✓	✓	—	✓	✓	✓
	1 output	✓	✓	✓	—	✓	✓	—	✓	✓	✓
	Toggle output	✓	✓	✓	—	✓	✓	—	✓	✓	✓
Input capture function		✓	✓	✓	✓*1	✓	✓	✓	✓	✓	✓*2
Synchronous operation		✓	✓	✓	—	✓	✓	—	✓	✓	—
PWM mode 1		✓	✓	✓	—	✓	✓	—	✓	✓	—
PWM mode 2		✓	✓	✓	—	—	—	—	—	—	—
Complementary PWM mode		—	—	—	—	✓	✓	—	✓	✓	—
Reset-synchronized PWM mode		—	—	—	—	✓	✓	—	✓	✓	—
AC synchronous motor drive mode		✓	—	—	—	✓	✓	—	—	—	—
Phase counting mode		—	✓	✓	✓	—	—	—	—	—	—
Buffer operation		✓	—	—	—	✓	✓	—	✓	✓	✓
Dead time compensation counter function		—	—	—	—	—	—	—	—	—	—
DMAC activation		TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGRALW/ TGRBLW input capture	TGR compare match or input capture	TGR compare match or input capture and TCNT overflow/ underflow*3	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture and TCNT overflow/ underflow*3	TGR compare match or input capture

Table 16.2 MTU Functions (2/2)

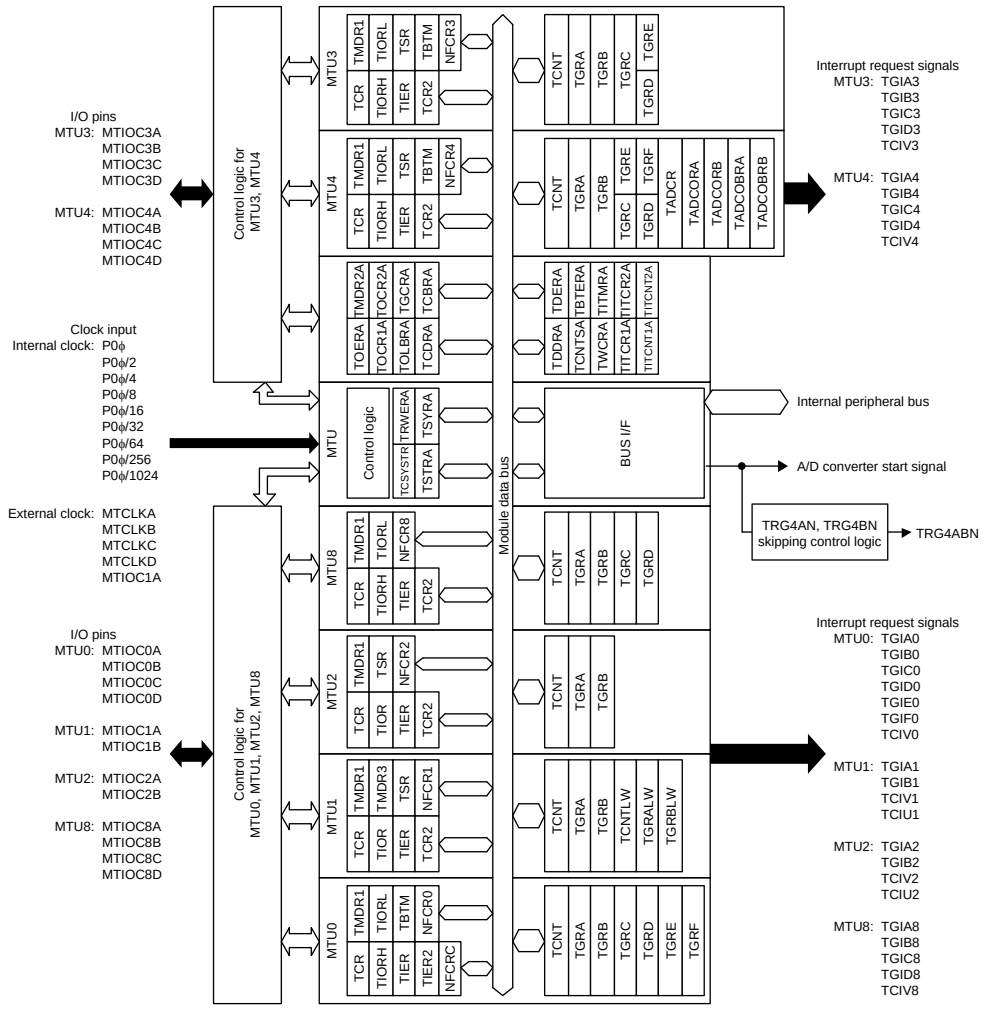
Item	MTU0	MTU1	MTU2	MTU1 & MTU2 (LWA = 1)	MTU3	MTU4	MTU5	MTU6	MTU7	MTU8
A/D converter start trigger	TGRA compare match or input capture TGRE compare match	TGRA compare match or input capture	TGRA compare match or input capture	TGRALW input capture	TGRA compare match or input capture	TGRA compare match or input capture, or TCNT underflow (trough) in complementary PWM mode	—	TGRA compare match or input capture	TGRA compare match or input capture, or TCNT underflow (trough) in complementary PWM mode	—
Interrupt sources	Seven sources • Compare match or input capture 0A • Compare match or input capture 0B • Compare match or input capture 0C • Compare match or input capture 0D • Compare match 0E • Compare match 0F • Overflow	Four sources • Compare match or input capture 1A • Compare match or input capture 1B • Overflow • Underflow	Four sources • Compare match or input capture 2A • Compare match or input capture 2B • Overflow • Underflow	Four sources • Input capture 1A • Input capture 1B • Overflow • Underflow	Five sources • Compare match or input capture 3A • Compare match or input capture 3B • Compare match or input capture 3C • Compare match or input capture 3D • Overflow	Five sources • Compare match or input capture 4A • Compare match or input capture 4B • Compare match or input capture 4C • Compare match or input capture 4D • Overflow or underflow*3	Three sources • Compare match or input capture 5U • Compare match or input capture 5V • Compare match or input capture 5W	Five sources • Compare match or input capture 6A • Compare match or input capture 6B • Compare match or input capture 6C • Compare match or input capture 6D • Overflow	Five sources • Compare match or input capture 7A • Compare match or input capture 7B • Compare match or input capture 7C • Compare match or input capture 7D • Overflow or underflow*3	Five sources • Compare match or input capture 8A • Compare match or input capture 8B • Compare match or input capture 8C • Compare match or input capture 8D • Overflow
A/D converter start request delaying function	—	—	—	—	—	• A/D converter start request at a match between TADCORA and TCNT or A/D converter start request at a match between TADCORB and TCNT	—	—	• A/D converter start request at a match between TADCORA and TCNT or A/D converter start request at a match between TADCORB and TCNT	—
Interrupt skipping 1	—	—	—	—	• Skips TGRA compare match interrupts	• Skips TCIV interrupts	—	• Skips TGRA compare match interrupts	• Skips TCIV interrupts	—
Interrupt skipping 2	—	—	—	—	—	• Skipping in compare count between TADCORA and TCNT, and TADCORB and TCNT	—	—	• Skipping in compare count between TADCORA and TCNT, and TADCORB and TCNT	—

Remarks: ✓: Possible —: Not possible

Note 1. When LWA is 1, the TGRALW capture source can be selected from either of the following: an input from MTIOC1A or MTU0.TGRA compare match/input capture event.
The TGRBLW capture source can be selected from any of the following: an input from MTIOC1B, MTU0.TGRC compare match/ input capture event, or MTU8.TGRC compare match event.

Note 2. Capture in MTU8 is supported only in normal mode.

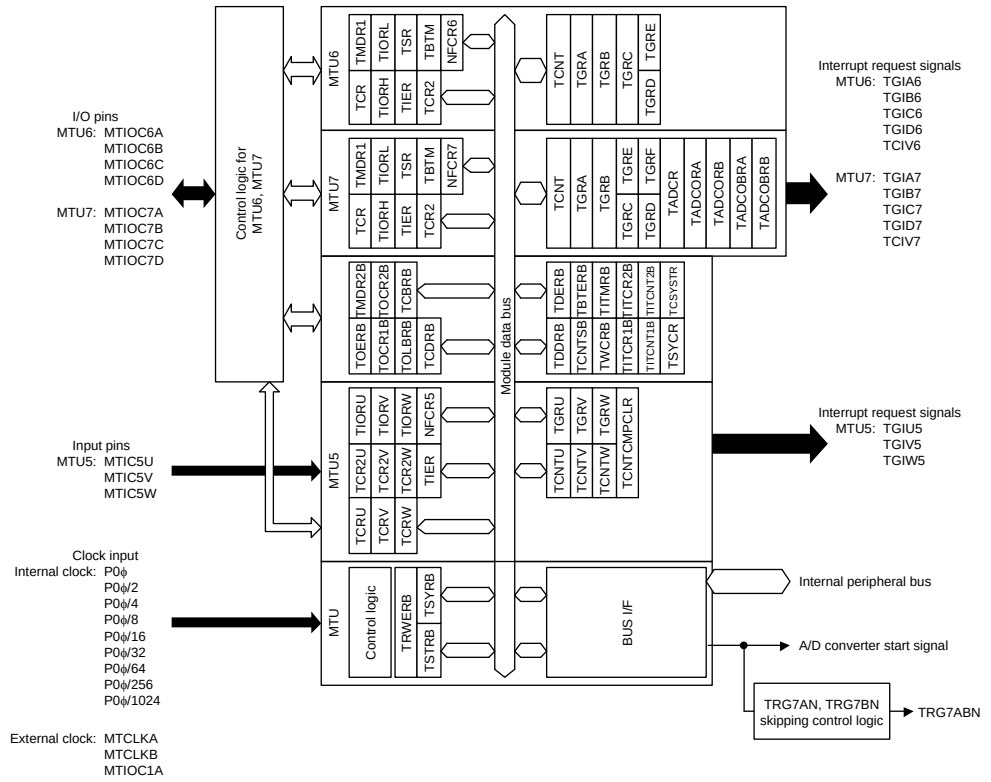
Note 3. The underflow interrupt source is valid only in complementary PWM mode.



TCR: Timer control register
TCR2: Timer control register 2
TMDR: Timer mode register 1
TMDR2A: Timer mode register 2
TMDR3: Timer mode register 3
TIOR: Timer I/O control register
TIORH: Timer I/O control register H
TIORL: Timer I/O control register L
TIER: Timer interrupt enable register
TIER2: Timer interrupt enable register 2
TSR: Timer status register
TBTM: Timer buffer operation transfer mode register
TCNT: Timer counter
TCNTLW: Timer longword counter
TGRA: Timer general register A
TGRALW: Timer longword general register A
TGRB: Timer general register B
TGRBLW: Timer longword general register B
TGRC: Timer general register C
TGRD: Timer general register D
TGRE: Timer general register E
TGRF: Timer general register F
TSTRA: Timer start register
TSYRA: Timer synchronous register
TCSYSTR: Timer counter synchronous start register

TRWERA: Timer read/write enable register
TOERA: Timer output master enable register
TOCR1A: Timer output control register 1
TOCR2A: Timer output control register 2
TOLBRA: Timer output level buffer register
TGCRA: Timer gate control register A
TCNTSA: Timer subcounter
TCDRA: Timer cycle data register
TCBRA: Timer cycle buffer register
TDDRA: Timer dead time data register
TDERA: Timer dead time enable register
TBTERA: Timer buffer transfer set register
TWCRA: Timer waveform control register
TICCR: Timer input capture control register
TADCR: Timer A/D converter start request control register
TADCORA: Timer A/D converter start request cycle set register A
TADCORB: Timer A/D converter start request cycle set register B
TADCOBRA: Timer A/D converter start request cycle set buffer register A
TADCOBRB: Timer A/D converter start request cycle set buffer register B
TITMRA: Timer interrupt skipping mode register
TITCR1A: Timer interrupt skipping set register 1
TITCR2A: Timer interrupt skipping set register 2
TITCNT1A: Timer interrupt skipping counter 1
TITCNT2A: Timer interrupt skipping counter 2
NFCRn: Noise filter control register n

Figure 16.1 Block Diagram of MTU (MTU0 to MTU4, MTU8)



TCR:	Timer control register
TCRU:	Timer control register U
TCR2U:	Timer control register 2U
TCRV:	Timer control register V
TCR2V:	Timer control register 2V
TCRW:	Timer control register W
TCR2W:	Timer control register 2W
TMDR1:	Timer mode register 1
TMDR2B:	Timer mode register 2
TIORH:	Timer I/O control register H
TIORL:	Timer I/O control register L
TIORU:	Timer I/O control register U
TIORV:	Timer I/O control register V
TIORW:	Timer I/O control register W
TIER:	Timer interrupt enable register
TSR:	Timer status register
TCNT:	Timer counter
TCNTU:	Timer counter U
TCNTV:	Timer counter V
TCNTW:	Timer counter W
TGRA:	Timer general register A
TGRB:	Timer general register B
TGRC:	Timer general register C
TGRD:	Timer general register D
TGRE:	Timer general register E
TGRF:	Timer general register F
TGRU:	Timer general register U
TGRV:	Timer general register V
TGRW:	Timer general register W
TCNTCMPCLR:	Timer compare match clear register

TSTRB:	Timer start register
TSYRB:	Timer synchronous register
TSYCR:	Timer synchronous clear register
TCSYSTR:	Timer counter synchronous start register
TRWERB:	Timer read/write enable register
TOERB:	Timer output master enable register
TOCR1B:	Timer output control register 1
TOCR2B:	Timer output control register 2
TOLBRB:	Timer output level buffer register
TCDRB:	Timer cycle data register
TCBRB:	Timer cycle buffer register
TDDRB:	Timer dead time data register
TDERB:	Timer dead time enable register
TCNTSB:	Timer subcounter
TBTERB:	Timer buffer transfer set register
TWCRB:	Timer waveform control register
TADCR:	Timer A/D converter start request control register
TADCORA:	Timer A/D converter start request cycle set register A
TADCORB:	Timer A/D converter start request cycle set register B
TADCOBRA:	Timer A/D converter start request cycle set buffer register A
TADCOBRB:	Timer A/D converter start request cycle set buffer register B
TITMRB:	Timer interrupt skipping mode register
TITCR1B:	Timer interrupt skipping set register 1
TITCR2B:	Timer interrupt skipping set register 2
TITCNT1B:	Timer interrupt skipping counter 1
TITCNT2B:	Timer interrupt skipping counter 2
NFCRn	Noise filter control register n

Figure 16.2 Block Diagram of MTU (MTU5 to MTU7)

Table 16.3 shows the configuration of pins for the MTU.

Table 16.3 Pin Configuration of the MTU

Channel	Pin Name	I/O	Function
MTU	MTCLKA	Input	External clock A input pin (MTU1/MTU2 phase counting mode A phase input)
	MTCLKB	Input	External clock B input pin (MTU1/MTU2 phase counting mode B phase input)
	MTCLKC	Input	External clock C input pin (MTU2 phase counting mode A phase input)
	MTCLKD	Input	External clock D input pin (MTU2 phase counting mode B phase input)
MTU0	MTIOC0A	I/O	MTU0 TGRA input capture input/output compare output/PWM output pin
	MTIOC0B	I/O	MTU0 TGRB input capture input/output compare output/PWM output pin
	MTIOC0C	I/O	MTU0 TGRC input capture input/output compare output/PWM output pin
	MTIOC0D	I/O	MTU0 TGRD input capture input/output compare output/PWM output pin
MTU1	MTIOC1A	I/O	MTU1 TGRA input capture input/output compare output/PWM output pin
	MTIOC1B	I/O	MTU1 TGRB input capture input/output compare output/PWM output pin
MTU2	MTIOC2A	I/O	MTU2 TGRA input capture input/output compare output/PWM output pin
	MTIOC2B	I/O	MTU2 TGRB input capture input/output compare output/PWM output pin
MTU3	MTIOC3A	I/O	MTU3 TGRA input capture input/output compare output/PWM output pin
	MTIOC3B	I/O	MTU3 TGRB input capture input/output compare output/PWM output pin
	MTIOC3C	I/O	MTU3 TGRC input capture input/output compare output/PWM output pin
	MTIOC3D	I/O	MTU3 TGRD input capture input/output compare output/PWM output pin
MTU4	MTIOC4A	I/O	MTU4 TGRA input capture input/output compare output/PWM output pin
	MTIOC4B	I/O	MTU4 TGRB input capture input/output compare output/PWM output pin
	MTIOC4C	I/O	MTU4 TGRC input capture input/output compare output/PWM output pin
	MTIOC4D	I/O	MTU4 TGRD input capture input/output compare output/PWM output pin
MTU5	MTIC5U	Input	MTU5 TGRU input capture input/external pulse input pin
	MTIC5V	Input	MTU5 TGRV input capture input/external pulse input pin
	MTIC5W	Input	MTU5 TGRW input capture input/external pulse input pin
MTU6	MTIOC6A	I/O	MTU6 TGRA input capture input/output compare output/PWM output pin
	MTIOC6B	I/O	MTU6 TGRB input capture input/output compare output/PWM output pin
	MTIOC6C	I/O	MTU6 TGRC input capture input/output compare output/PWM output pin
	MTIOC6D	I/O	MTU6 TGRD input capture input/output compare output/PWM output pin
MTU7	MTIOC7A	I/O	MTU7 TGRA input capture input/output compare output/PWM output pin
	MTIOC7B	I/O	MTU7 TGRB input capture input/output compare output/PWM output pin
	MTIOC7C	I/O	MTU7 TGRC input capture input/output compare output/PWM output pin
	MTIOC7D	I/O	MTU7 TGRD input capture input/output compare output/PWM output pin
MTU8	MTIOC8A	I/O	MTU8.TGRA input capture input/output compare output pin
	MTIOC8B	I/O	MTU8.TGRB input capture input/output compare output pin
	MTIOC8C	I/O	MTU8.TGRC input capture input/output compare output pin
	MTIOC8D	I/O	MTU8.TGRD input capture input/output compare output pin

16.2 Register Descriptions

Table 16.4 shows the register configuration. The MTU address space is offset from the base address. The base address of MTU is as follows.

MTU base address: H'0_1000_0000 (Overall Address Space)
MTU base address: H'4000_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
MTU base address: H'5000_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 16.4 Register configuration (1/5)

Channel	Register Name	Abbreviation	Address	Access size
MTU0	Timer control register	TCR	H'1300	8
	Timer mode register 1	TMDR1	H'1301	8
	Timer I/O control register H	TIORH	H'1302	8
	Timer I/O control register L	TIORL	H'1303	8
	Timer interrupt enable register	TIER	H'1304	8
	Timer counter	TCNT	H'1306	16
	Timer general register A	TGRA	H'1308	16
	Timer general register B	TGRB	H'130A	16
	Timer general register C	TGRC	H'130C	16
	Timer general register D	TGRD	H'130E	16
	Timer general register E	TGRE	H'1320	16
	Timer general register F	TGRF	H'1322	16
	Timer interrupt enable register 2	TIER2	H'1324	8
	Timer buffer operation transfer mode register	TBTM	H'1326	8
	Timer control register 2	TCR2	H'1328	8
	Noise filter control register 0	NFCR0	H'1290	8
	Noise filter control register C	NFCRC	H'1299	8
MTU1	Timer control register	TCR	H'1380	8
	Timer mode register 1	TMDR1	H'1381	8
	Timer I/O control register	TIOR	H'1382	8
	Timer interrupt enable register	TIER	H'1384	8
	Timer status register	TSR	H'1385	8
	Timer counter	TCNT	H'1386	16
	Timer general register A	TGRA	H'1388	16
	Timer general register B	TGRB	H'138A	16
	Timer input capture control register	TICCR	H'1390	8
	Timer mode register 3	TMDR3	H'1391	8
	Timer control register 2	TCR2	H'1394	8
	Timer longword counter	TCNTLW	H'13A0	32
	Timer longword general register A	TGRALW	H'13A4	32
	Timer longword general register B	TGRBLW	H'13A8	32
	Noise filter control register 1	NFCR1	H'1291	8

Table 16.4 Register configuration (2/5)

Channel	Register Name	Abbreviation	Address	Access size
MTU2	Timer control register	TCR	H'1400	8
	Timer mode register 1	TMDR1	H'1401	8
	Timer I/O control register	TIOR	H'1402	8
	Timer interrupt enable register	TIER	H'1404	8
	Timer status register	TSR	H'1405	8
	Timer counter	TCNT	H'1406	16
	Timer general register A	TGRA	H'1408	16
	Timer general register B	TGRB	H'140A	16
	Timer control register 2	TCR2	H'140C	8
	Noise filter control register 2	NFCR2	H'1292	8
MTU3	Timer control register	TCR	H'1200	8
	Timer mode register 1	TMDR1	H'1202	8
	Timer I/O control register H	TIORH	H'1204	8
	Timer I/O control register L	TIORL	H'1205	8
	Timer interrupt enable register	TIER	H'1208	8
	Timer counter	TCNT	H'1210	16
	Timer general register A	TGRA	H'1218	16
	Timer general register B	TGRB	H'121A	16
	Timer general register C	TGRC	H'1224	16
	Timer general register D	TGRD	H'1226	16
	Timer control register 2	TCR2	H'124C	8
	Timer general register E	TGRE	H'1272	16
	Timer status register	TSR	H'122C	8
	Timer buffer operation transfer mode register	TBTM	H'1238	8
	Noise filter control register 3	NFCR3	H'1293	8
MTU4	Timer control register	TCR	H'1201	8
	Timer mode register 1	TMDR1	H'1203	8
	Timer I/O control register H	TIORH	H'1206	8
	Timer I/O control register L	TIORL	H'1207	8
	Timer interrupt enable register	TIER	H'1209	8
	Timer counter	TCNT	H'1212	16
	Timer general register A	TGRA	H'121C	16
	Timer general register B	TGRB	H'121E	16
	Timer general register C	TGRC	H'1228	16
	Timer general register D	TGRD	H'122A	16
	Timer control register 2	TCR2	H'124D	8
	Timer general register E	TGRE	H'1274	16
	Timer general register F	TGRF	H'1276	16
	Timer status register	TSR	H'122D	8
	Timer buffer operation transfer mode register	TBTM	H'1239	8
	Timer A/D converter start request control register	TADCR	H'1240	16
	Timer A/D converter start request cycle set register A	TADCORA	H'1244	16
	Timer A/D converter start request cycle set register B	TADCORB	H'1246	16
	Timer A/D converter start request cycle set buffer register A	TADCOBRA	H'1248	16
	Timer A/D converter start request cycle set buffer register B	TADCOBRB	H'124A	16

Table 16.4 Register configuration (3/5)

Channel	Register Name	Abbreviation	Address	Access size
MTU4	Noise filter control register 4	NFCR4	H'1294	8
MTU5	Timer counter U	TCNTU	H'1C80	16
	Timer general register U	TGRU	H'1C82	16
	Timer control register U	TCRU	H'1C84	8
	Timer control register 2U	TCR2U	H'1C85	8
	Timer I/O control register U	TIORU	H'1C86	8
	Timer counter V	TCNTV	H'1C90	16
	Timer general register V	TGRV	H'1C92	16
	Timer control register V	TCRV	H'1C94	8
	Timer control register 2V	TCR2V	H'1C95	8
	Timer I/O control register V	TIORV	H'1C96	8
	Timer counter W	TCNTW	H'1CA0	16
	Timer general register W	TGRW	H'1CA2	16
	Timer control register W	TCRW	H'1CA4	8
	Timer control register 2W	TCR2W	H'1CA5	8
	Timer I/O control register W	TIORW	H'1CA6	8
	Timer interrupt enable register	TIER	H'1CB2	8
	Timer start register	TSTR	H'1CB4	8
	Timer compare match clear register	TCNTCMPCLR	H'1CB6	8
	Noise filter control register 5	NFCR5	H'1A95	8
MTU6	Timer control register	TCR	H'1A00	8
	Timer mode register 1	TMDR1	H'1A02	8
	Timer I/O control register H	TIORH	H'1A04	8
	Timer I/O control register L	TIORL	H'1A05	8
	Timer interrupt enable register	TIER	H'1A08	8
	Timer counter	TCNT	H'1A10	16
	Timer general register A	TGRA	H'1A18	16
	Timer general register B	TGRB	H'1A1A	16
	Timer general register C	TGRC	H'1A24	16
	Timer general register D	TGRD	H'1A26	16
	Timer control register 2	TCR2	H'1A4C	8
	Timer general register E	TGRE	H'1A72	16
	Timer synchronous clear register	TSYCR	H'1A50	8
	Timer status register	TSR	H'1A2C	8
	Timer buffer operation transfer mode register	TBTM	H'1A38	8
	Noise filter control register 6	NFCR6	H'1A93	8
MTU7	Timer control register	TCR	H'1A01	8
	Timer mode register 1	TMDR1	H'1A03	8
	Timer I/O control register H	TIORH	H'1A06	8
	Timer I/O control register L	TIORL	H'1A07	8
	Timer interrupt enable register	TIER	H'1A09	8
	Timer counter	TCNT	H'1A12	16
	Timer general register A	TGRA	H'1A1C	16
	Timer general register B	TGRB	H'1A1E	16
	Timer general register C	TGRC	H'1A28	16

Table 16.4 Register configuration (4/5)

Channel	Register Name	Abbreviation	Address	Access size
MTU7	Timer general register D	TGRD	H'1A2A	16
	Timer control register 2	TCR2	H'1A4D	8
	Timer general register E	TGRE	H'1A74	16
	Timer general register F	TGRF	H'1A76	16
	Timer status register	TSR	H'1A2D	8
	Timer buffer operation transfer mode register	TBTM	H'1A39	8
	Timer A/D converter start request control register	TADCR	H'1A40	16
	Timer A/D converter start request cycle set register A	TADCORA	H'1A44	16
	Timer A/D converter start request cycle set register B	TADCORB	H'1A46	16
	Timer A/D converter start request cycle set buffer register A	TADCOBRA	H'1A48	16
	Timer A/D converter start request cycle set buffer register B	TADCOBRB	H'1A4A	16
	Noise filter control register 7	NFCR7	H'1A94	8
MTU8	Timer control register	TCR	H'1600	8
	Timer mode register 1	TMDR1	H'1601	8
	Timer I/O control register H	TIORH	H'1602	8
	Timer I/O control register L	TIORL	H'1603	8
	Timer interrupt enable register	TIER	H'1604	8
	Timer control register 2	TCR2	H'1606	8
	Timer counter	TCNT	H'1608	32
	Timer general register A	TGRA	H'160C	32
	Timer general register B	TGRB	H'1610	32
	Timer general register C	TGRC	H'1614	32
	Timer general register D	TGRD	H'1618	32
	Noise filter control register 8	NFCR8	H'1298	8
MTU	Timer output master enable register A	TOERA	H'120A	8
	Timer gate control register A	TGCRA	H'120D	8
	Timer output control register 1A	TOCR1A	H'120E	8
	Timer output control register 2A	TOCR2A	H'120F	8
	Timer cycle data register A	TCDRA	H'1214	16
	Timer dead time data register A	TDDRA	H'1216	16
	Timer sub counter A	TCNTSA	H'1220	16
	Timer cycle buffer register A	TCBRA	H'1222	16
	Timer interrupt skipping set register 1A	TITCR1A	H'1230	8
	Timer interrupt skipping counter 1A	TITCNT1A	H'1231	8
	Timer buffer transfer set register A	TBTERA	H'1232	8
	Timer dead time enable register A	TDERA	H'1234	8
	Timer output level buffer register A	TOLBRA	H'1236	8
	Timer interrupt skipping mode register A	TITMRA	H'123A	8
	Timer interrupt skipping set register 2A	TITCR2A	H'123B	8
	Timer interrupt skipping counter 2A	TITCNT2A	H'123C	8
	Timer waveform control register A	TWCRA	H'1260	8
	Timer mode register 2A	TMDR2A	H'1270	8
	Timer read/write enable register A	TRWERA	H'1284	8
	Timer start register A	TSTRA	H'1280	8
	Timer synchronous register A	TSYRA	H'1281	8

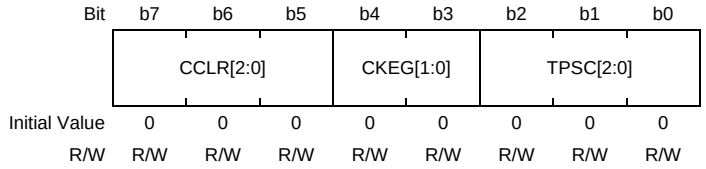
Table 16.4 Register configuration (5/5)

Channel	Register Name	Abbreviation	Address	Access size
MTU	Timer output master enable register B	TOERB	H'1A0A	8
	Timer output control register 1B	TOCR1B	H'1A0E	8
	Timer output control register 2B	TOCR2B	H'1A0F	8
	Timer cycle data register B	TCDRB	H'1A14	16
	Timer dead time data register B	TDDRB	H'1A16	16
	Timer sub counter B	TCNTSB	H'1A20	16
	Timer cycle buffer register B	TCBRB	H'1A22	16
	Timer interrupt skipping set register 1B	TITCR1B	H'1A30	8
	Timer interrupt skipping counter 1B	TITCNT1B	H'1A31	8
	Timer buffer transfer set register B	TBTERB	H'1A32	8
	Timer dead time enable register B	TDERB	H'1A34	8
	Timer output level buffer register B	TOLBRB	H'1A36	8
	Timer interrupt skipping mode register B	TITMRB	H'1A3A	8
	Timer interrupt skipping set register 2B	TITCR2B	H'1A3B	8
	Timer interrupt skipping counter 2B	TITCNT2B	H'1A3C	8
	Timer waveform control register B	TWCRB	H'1A60	8
	Timer mode register 2B	TMDR2B	H'1A70	8
	Timer start register B	TSTRB	H'1A80	8
	Timer synchronous register B	TSYRB	H'1A81	8
	Timer read/write enable register B	TRWERB	H'1A84	8
	Timer counter synchronous start register	TCSYSTR	H'1282	8

16.2.1 Timer Control Register (TCR)

(1) MTU0.TCR, MTU1.TCR, MTU2.TCR, MTU3.TCR, MTU4.TCR, MTU6.TCR, MTU7.TCR, MTU8.TCR

Address(es): MTU0.TCR H'0_1000_1300, MTU1.TCR H'0_1000_1380, MTU2.TCR H'0_1000_1400, MTU3.TCR H'0_1000_1200, MTU4.TCR H'0_1000_1201, MTU6.TCR H'0_1000_1A00, MTU7.TCR H'0_1000_1A01, MTU8.TCR H'0_1000_1600



Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	TPSC[2:0]	All 0	R/W	Time Prescaler Select See **Table 16.7** to **Table 16.10**.
b4, b3	CKEG[1:0]	All 0	R/W	Clock Edge Select b4 b3 0 0: Count at rising edge 0 1: Count at falling edge 1 x: Count at both edges
b7 to b5	CCLR[2:0]	All 0	R/W	Counter Clear Source Select See **Table 16.5** and **Table 16.6**.

Remarks: x: Don't care

The TCR register controls the TCNT operation for each channel in combination with the TCR2 register. The MTU has a total of 11 TCR registers, one each for MTU0 to MTU4, MTU 6, MTU7, and MTU8 and three (TCRU, TCRV, and TCRW) for MTU5. TCR values should be specified only while TCNT operation is stopped.

TPSC[2:0] Bits (Time Prescaler Select)

These bits select the TCNT count clock source. The clock source can be selected independently for each channel. See **Table 16.7** to **Table 16.10** for details.

CKEG[1:0] Bits (Clock Edge Select)

These bits select the count clock source edge, including the MTIOC1A pin. When the internal clock is counted at both edges, the count clock period is halved (e.g. P0ϕ/4 at both edges = P0ϕ/2 at rising edge). If phase counting mode is used on MTU1 and MTU2, the setting of these bits is ignored and the phase counting mode setting has priority. Internal clock edge selection is valid when the count clock source is P0ϕ/2 or slower. When P0ϕ/1 or the overflow/underflow in another channel is selected for the count clock source, a value can be written to these bits but counter operation compiles with the initial value.

CCLR[2:0] Bits (Counter Clear Source Select)

These bits select the TCNT counter clearing source. See **Table 16.5** and **Table 16.6** for details.

Table 16.5 CCLR[2:0] (MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8)

Channel	Bit 7 CCLR2	Bit 6 CCLR1	Bit 5 CCLR0	Description
MTU0 MTU3 MTU4 MTU6 MTU7 MTU8	0	0	0	TCNT clearing disabled
	0	0	1	TCNT cleared by TGRA compare match/input capture
	0	1	0	TCNT cleared by TGRB compare match/input capture
	0	1	1	TCNT cleared by counter clearing in another channel where synchronous clearing or synchronous operation is selected*1
	1	0	0	TCNT clearing disabled
	1	0	1	TCNT cleared by TGRC compare match/input capture*2
	1	1	0	TCNT cleared by TGRD compare match/input capture*2
	1	1	1	TCNT cleared by counter clearing in another channel where synchronous clearing or synchronous operation is selected*1

Note 1. Synchronous operation is selected by setting a SYNC bit in TSYRA or TSYRB to 1 except for MTU8.

Note 2. When TGRC or TGRD is used as a buffer register, TCNT is not cleared because the buffer register setting has priority and compare match/input capture does not occur.

Table 16.6 CCLR[2:0] (MTU1 and MTU2)

Channel	Bit 7 Reserved*2	Bit 6 CCLR1	Bit 5 CCLR0	Description
MTU1 MTU2	0	0	0	TCNT clearing disabled
	0	0	1	TCNT cleared by TGRA compare match/input capture
	0	1	0	TCNT cleared by TGRB compare match/input capture
	0	1	1	TCNT cleared by counter clearing in another channel where synchronous clearing or synchronous operation is selected*1

Note 1. Synchronous operation is selected by setting a SYNC bit in TSYRA or TSYRB to 1.

Note 2. Bit-7 is reserved in MTU1 and MTU2. It is read as 0. The write value is ignored.

(2) MTU5.TCRU, MTU5.TCRV, MTU5.TCRW

Address(es): MTU5.TCRU H'0_1000_1C84, MTU5.TCRV H'0_1000_1C94, MTU5.TCRW H'0_1000_1CA4

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	TPSC[1:0]	
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	TPSC[1:0]	All 0	R/W	Time Prescaler Select See **Table 16.11**.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TPSC[1:0] Bits (Time Prescaler Select)

These bits select the TCNT count clock source. See **Table 16.11** for details.

16.2.2 Timer Control Register 2 (TCR2)

(1) MTU0.TCR2, MTU3.TCR2, MTU4.TCR2, MTU6.TCR2, MTU7.TCR2, MTU8.TCR2

Address(es): MTU0.TCR2 H'0_1000_1328, MTU3.TCR2 H'0_1000_124C, MTU4.TCR2 H'0_1000_124D, MTU6.TCR2 H'0_1000_1A4C, MTU7.TCR2 H'0_1000_1A4D, MTU8.TCR2 H'0_1000_1606

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TPSC2[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) MTU1.TCR2, MTU2.TCR2

Address(es): MTU1.TCR2 H'0_1000_1394, MTU2.TCR2 H'0_1000_140C

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PCB[1:0]		TPSC2[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	TPSC2[2:0]	All 0	R/W	Time Prescaler Select See **Table 16.7** to **Table 16.10**.
b4, b3	PCB[1:0]	All 0	R/W	Phase Counting Mode Function Expansion Control Functional Expansion Control for Phase Counting Modes 2, 3, and 5
b7 to b5	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

The TCR2 register controls the TCNT operation for each channel in combination with the TCR register. The MTU has a total of 11 TCR2 registers, one each for MTU0 to MTU4, MTU 6, MTU7, and MTU8 and three (TCR2U, TCR2V, and TCR2W) for MTU5. TCR2 values should be specified only while TCNT operation is stopped.

TPSC2[2:0] Bits (Time Prescaler Select)

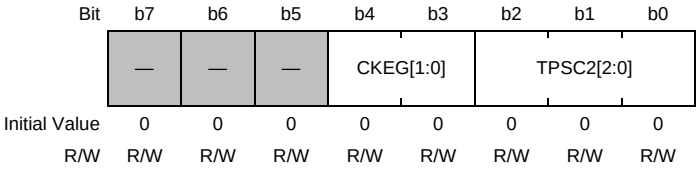
These bits select the TCNT count clock source. The clock source can be selected independently for each channel. See **Table 16.7** to **Table 16.10** for details.

PCB[1:0] Bits (Phase Counting Mode Function Expansion Control)

These bits control extended functions for phase counting mode 2, 3, and 5 in MTU1 and MTU2. See **Section 16.3.6, Phase Counting Mode**.

(3) MTU5.TCR2U, MTU5.TCR2V, MTU5.TCR2W

Address(es): MTU5.TCR2U H'0_1000_1C85, MTU5.TCR2V H'0_1000_1C95, MTU5.TCR2W H'0_1000_1CA5



Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	TPSC2[2:0]	All 0	R/W	Time Prescaler Select See **Table 16.11**.
b4, b3	CKEG[1:0]	All 0	R/W	Clock Edge Select b4 b3 0 0: Counts at the rising edge. 0 1: Counts at the falling edge. 1 x: Counts at both edges.
b7 to b5	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Remarks: x: Don't care

TPSC2[2:0] Bits (Time Prescaler Select)

These bits select the TCNT count clock source. See **Table 16.11** for details.

CKEG[1:0] Bits (Clock Edge Select)

These bits select the edge of the count clock source signal output from the MTIOC1A pin.

Table 16.7 TPSC[2:0], TPSC2[2:0] (MTU0)

	TCR2[2:0]			TCR[2:0]			
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC2	TPSC1	TPSC0	Description
MTU0	0	0	0	0	0	0	Internal clock: counts on P0ϕ/1
	0	0	0	0	0	1	Internal clock: counts on P0ϕ/4
	0	0	0	0	1	0	Internal clock: counts on P0ϕ/16
	0	0	0	0	1	1	Internal clock: counts on P0ϕ/64
	0	0	0	1	0	0	External clock: counts on MTCLKA pin input
	0	0	0	1	0	1	External clock: counts on MTCLKB pin input
	0	0	0	1	1	0	External clock: counts on MTCLKC pin input
	0	0	0	1	1	1	External clock: counts on MTCLKD pin input
	0	0	1	x	x	x	Internal clock: counts on P0ϕ/2
	0	1	0	x	x	x	Internal clock: counts on P0ϕ/8
	0	1	1	x	x	x	Internal clock: counts on P0ϕ/32
	1	0	0	x	x	x	Internal clock: counts on P0ϕ/256
	1	0	1	x	x	x	Internal clock: counts on P0ϕ/1024
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	External clock: counts on MTIOC1A pin input

Remarks: x: Don’t care

Table 16.8 TPSC[2:0], TPSC2[2:0] (MTU1)

	TCR2[2:0]			TCR[2:0]			
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC2	TPSC1	TPSC0	Description
MTU1	0	0	0	0	0	0	Internal clock: counts on P0ϕ/1
	0	0	0	0	0	1	Internal clock: counts on P0ϕ/4
	0	0	0	0	1	0	Internal clock: counts on P0ϕ/16
	0	0	0	0	1	1	Internal clock: counts on P0ϕ/64
	0	0	0	1	0	0	External clock: counts on MTCLKA pin input
	0	0	0	1	0	1	External clock: counts on MTCLKB pin input
	0	0	0	1	1	0	Internal clock: counts on P0ϕ/256
	0	0	0	1	1	1	Overflow/underflow of MTU2.TCNT
	0	0	1	x	x	x	Internal clock: counts on P0ϕ/2
	0	1	0	x	x	x	Internal clock: counts on P0ϕ/8
	0	1	1	x	x	x	Internal clock: counts on P0ϕ/32
	1	0	0	x	x	x	Internal clock: counts on P0ϕ/1024
	1	0	1	x	x	x	Setting prohibited
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	Setting prohibited

Note: This setting has no effect when MTU1 is in phase counting mode.

Remarks: x: Don’t care

Table 16.9 TPSC[2:0], TPSC2[2:0] (MTU2)

	TCR2[2:0]			TCR[2:0]			
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC2	TPSC1	TPSC0	Description
MTU2	0	0	0	0	0	0	Internal clock: counts on P0ϕ/1
	0	0	0	0	0	1	Internal clock: counts on P0ϕ/4
	0	0	0	0	1	0	Internal clock: counts on P0ϕ/16
	0	0	0	0	1	1	Internal clock: counts on P0ϕ/64
	0	0	0	1	0	0	External clock: counts on MTCLKA pin input
	0	0	0	1	0	1	External clock: counts on MTCLKB pin input
	0	0	0	1	1	0	External clock: counts on MTCLKC pin input
	0	0	0	1	1	1	Internal clock: counts on P0ϕ/1024
	0	0	1	x	x	x	Internal clock: counts on P0ϕ/2
	0	1	0	x	x	x	Internal clock: counts on P0ϕ/8
	0	1	1	x	x	x	Internal clock: counts on P0ϕ/32
	1	0	0	x	x	x	Internal clock: counts on P0ϕ/256
	1	0	1	x	x	x	Setting prohibited
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	Setting prohibited

Note: This setting has no effect when the MTU2 is in phase counting mode.

Remarks: x: Don't care

Table 16.10 TPSC[2:0], TPSC2[2:0] (MTU3, MTU4, MTU6, MTU7, and MTU8)

	TCR2[2:0]			TCR[2:0]			
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC2	TPSC1	TPSC0	Description
MTU3 MTU4 MTU6 MTU7 MTU8	0	0	0	0	0	0	Internal clock: counts on P0ϕ/1
	0	0	0	0	0	1	Internal clock: counts on P0ϕ/4
	0	0	0	0	1	0	Internal clock: counts on P0ϕ/16
	0	0	0	0	1	1	Internal clock: counts on P0ϕ/64
	0	0	0	1	0	0	Internal clock: counts on P0ϕ/256
	0	0	0	1	0	1	Internal clock: counts on P0ϕ/1024
	0	0	0	1	1	0	External clock: counts on MTCLKA pin input
	0	0	0	1	1	1	External clock: counts on MTCLKB pin input
	0	0	1	x	x	x	Internal clock: counts on P0ϕ/2
	0	1	0	x	x	x	Internal clock: counts on P0ϕ/8
	0	1	1	x	x	x	Internal clock: counts on P0ϕ/32
	1	0	0	x	x	x	Setting prohibited
	1	0	1	x	x	x	Setting prohibited
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	Setting prohibited

Remarks: x: Don't care

Table 16.11 TPSC[1:0], TPSC2[2:0] (MTU5)

	TCR2[2:0]			TCR[1:0]		
	Bit 2	Bit 1	Bit 0	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC1	TPSC0	Description
MTU5	0	0	0	0	0	Internal clock: counts on P0ϕ/1
	0	0	0	0	1	Internal clock: counts on P0ϕ/4
	0	0	0	1	0	Internal clock: counts on P0ϕ/16
	0	0	0	1	1	Internal clock: counts on P0ϕ/64
	0	0	1	x	x	Internal clock: counts on P0ϕ/2
	0	1	0	x	x	Internal clock: counts on P0ϕ/8
	0	1	1	x	x	Internal clock: counts on P0ϕ/32
	1	0	0	x	x	Internal clock: counts on P0ϕ/256
	1	0	1	x	x	Internal clock: counts on P0ϕ/1024
	1	1	0	x	x	Setting prohibited
	1	1	1	x	x	Internal clock: counts on MTIOC1A pin input

Remarks: x: Don’t care

16.2.3 Timer Mode Register 1 (TMDR1)

(1) MTU0.TMDR1

Address(es): MTU0.TMDR1 H'0_1000_1301

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	BFE	BFB	BFA	MD[3:0]			
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) MTU1.TMDR1, MTU2.TMDR1

Address(es): MTU1.TMDR1 H'0_1000_1381, MTU2.TMDR1 H'0_1000_1401

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	MD[3:0]			
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(3) MTU3.TMDR1, MTU4.TMDR1, MTU6.TMDR1, MTU7.TMDR1, MTU8.TMDR1

Address(es): MTU3.TMDR1 H'0_1000_1202, MTU4.TMDR1 H'0_1000_1203, MTU6.TMDR1 H'0_1000_1A02, MTU7.TMDR1 H'0_1000_1A03, MTU8.TMDR1 H'0_1000_1601

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	BFB	BFA	MD[3:0]			
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b3 to b0	MD[3:0]	All 0	R/W	Mode Select These bits specify the timer operating mode. See **Table 16.12** for details.
b4	BFA	0	R/W	Buffer Operation A 0: TGRA and TGRC operate normally 1: TGRA and TGRC used together for buffer operation
b5	BFB	0	R/W	Buffer Operation B 0: TGRB and TGRD operate normally 1: TGRB and TGRD used together for buffer operation
b6	BFE	0	R/W	Buffer Operation E 0: MTU0.TGRE and MTU0.TGRF operate normally 1: MTU0.TGRE and MTU0.TGRF used together for buffer operation
b7	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.

The TMDR1 register specifies the operating mode of each channel. The MTU has a total of eight TMDR1 registers, one each for MTU0 to MTU4, MTU 6, MTU7, and MTU8. TMDR1 register values should be specified only while TCNT operation is stopped.

Table 16.12 Operating Mode Setting by MD[3:0] Bits (MTU0 to MTU4, MTU6 to MTU8)

Bit 3	Bit 2	Bit 1	Bit 0		MTU0	MTU1	MTU2	MTU1 & MTU2 (LWA = 1)	MTU3	MTU4	MTU6	MTU7	MTU8
MD3	MD2	MD1	MD0	Description									
0	0	0	0	Normal mode	✓	✓	✓		✓	✓	✓	✓	✓
0	0	0	1	Setting prohibited									
0	0	1	0	PWM mode 1	✓	✓	✓		✓	✓	✓	✓	
0	0	1	1	PWM mode 2	✓	✓	✓						
0	1	0	0	Phase counting mode 1		✓	✓	✓					
0	1	0	1	Phase counting mode 2		✓	✓	✓					
0	1	1	0	Phase counting mode 3		✓	✓	✓					
0	1	1	1	Phase counting mode 4		✓	✓	✓					
1	0	0	0	Reset-synchronized PWM mode*1					✓		✓		
1	0	0	1	Phase counting mode 5		✓	✓	✓					
1	0	1	x	Setting prohibited									
1	1	0	0	Setting prohibited									
1	1	0	1	Complementary PWM mode 1 (transfer at crest)*1					✓		✓		
1	1	1	0	Complementary PWM mode 2 (transfer at trough)*1					✓		✓		
1	1	1	1	Complementary PWM mode 3 (transfer at crest and trough)*1					✓		✓		

Note: Only set the corresponding operating mode listed above for each channel.

Remarks: x: Don't care

Note 1. Reset-synchronized PWM mode and complementary PWM mode can only be set for MTU3 and MTU6. When MTU3 or MTU6 is set to reset-synchronized PWM mode or complementary PWM mode, the MTU4 or MTU7 settings become ineffective and automatically conform to the MTU3 or MTU6 setting, respectively. MTU4 and MTU7 should be set to the initial values (normal mode).

BFA Bit (Buffer Operation A)

This bit specifies whether to operate TGRA in the normal way or to use TGRA and TGRC together for buffer operation. When TGRC is used as a buffer register, TGRC input capture/output compare does not take place in modes other than complementary PWM mode, but compare match with TGRC occurs in complementary PWM mode.

If a compare match occurs on MTU4 in the Tb interval in complementary PWM mode, the TGIEC bit in timer interrupt enable register (MTU4.TIER) should be set to 0.

In reset-synchronized PWM mode or complementary PWM mode, buffer operation in MTU3 and MTU4 (or MTU6 and MTU7) depends on the settings in the BFA bit of MTU3.TMDR1 (MTU6.TMDR1). The BFA bit of MTU4.TMDR1 (MTU7.TMDR1) should be set to 0.

In MTU1 and MTU2, which have no TGRC, this bit is reserved. It is read as 0. The write value should be 0. See **Figure 16.50** for an illustration of the Tb interval in complementary PWM mode.

BFB Bit (Buffer Operation B)

This bit specifies whether to operate TGRB in the normal way or to use TGRB and TGRD together for buffer operation. When TGRD is used as a buffer register, TGRD input capture/output compare does not take place in modes other than complementary PWM mode, but compare match with TGRD occurs in complementary PWM mode.

If a compare match occurs on MTU4 in the Tb interval in complementary PWM mode, the TGIED bit in timer interrupt enable register (MTU4.TIER) should be set to 0.

In reset-synchronized PWM mode or complementary PWM mode, buffer operation in MTU3 and MTU4 (or MTU6 and MTU7) depends on the settings in the BFB bit of MTU3.TMDR1 (MTU6.TMDR1). The BFB bit of MTU4.TMDR1 (MTU7.TMDR1) should be set to 0.

In MTU1 and MTU2, which have no TGRD, this bit is reserved. It is read as 0. The write value should be 0. See **Figure 16.50** for an illustration of the Tb interval in complementary PWM mode.

BFE Bit (Buffer Operation E)

This bit specifies whether to operate MTU0.TGRE and MTU0.TGRF in the normal way or to use them together for buffer operation. Compare match with TGRF occurs even when TGRF is used as a buffer register.

In MTU0 to MTU4, MTU6, MTU7, and MTU8, this bit is reserved. It is read as 0. The write value should be 0.

16.2.4 Timer Mode Registers 2 (TMDR2A and TMDR2B)

Address(es): MTU.TMDR2A H'0_1000_1270, MTU.TMDR2B H'0_1000_1A70

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	DRS
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	DRS	0	R/W	Double Buffer Select 0: Double buffer function is disabled 1: Double buffer function is enabled
b7 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TMDR2A and TMDR2B specify the double buffer function in complementary PWM mode 3 (transfer at the crest and trough of the counter value). The MTU has two TMDR2 registers, one each for MTU3 (TMDR2A) and MTU6 (TMDR2B). TMDR2A and TMDR2B values should be specified only while TCNT operation is stopped.

DRS Bit (Double Buffer Select)

This bit enables or disables the double buffer function in complementary PWM mode.

16.2.5 Timer Mode Register 3 (TMDR3)

Address(es): MTU1.TMDR3 H'0_1000_1391

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	PHCKSEL	LWA
Initial Value	0	0	0	0	0	0	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	LWA	0	R/W	MTU1/MTU2 Combination Longword Access Control 0: 16-bit access is enabled. 1: 32-bit access is enabled.
b1	PHCKSEL	1	R/W	External Input Phase Clock Select Selects the external clock pin for phase counting mode. 0: MTCLKA and MTCLKB are selected for the external phase clock. 1: MTCLKC and MTCLKD are selected for the external phase clock.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

The TMDR3 register controls longword access to a 32-bit register or counter in a combination of MTU1 and MTU2. There is only one TMDR3 register in MTU1. The counter (TCNTLW), general register A (TGRALW), and general register B (TGRBLW) of MTU1 and MTU2 are accessed in the combinations listed in **Table 16.13**.

LWA Bit (MTU1/MTU2 Combination Longword Access Control)

This bit selects a 32-bit access in a combination of MTU1 and MTU2.

When the LWA bit is set to 0, MTU1 and MTU2 operate independently as 16-bit timers and the TCNTLW, TGRALW, and TGRBLW registers cannot be accessed.

When the LWA bit is set to 1, MTU1 and MTU2 are cascaded and operate as a 32-bit timer, which is controlled through the TCR, TCR2, TIOR, and TMDR1 registers in MTU1. The TCR, TCR2, TIOR, and TMDR1 register settings in MTU2 are ignored. The 16-bit registers (TCNT, TGRA, and TGRB) in MTU1 and MTU2 cannot be accessed. MTU2 input capture and compare match are disabled.

Note that MTU1 and MTU2 can be cascaded by setting the LWA bit to 1 only in phase-counting mode. Cascade connection cannot be used in normal, PWM1, or PWM2 mode. When setting the LWA bit to 1, be sure to select phase-counting mode.

In addition, initialize the TCNT, TGRA, and TGRB registers in MTU1 and MTU2 before setting the LWA bit to 1.

PHCKSEL Bit (External Input Phase Clock Select)

When the MTU1 and MTU2 registers are combined for 32-bit phase counting mode or MTU2 phase counting mode, this bit selects either the A- or B-phase signal from the external clock. See **Table 16.66, Clock Input Pins in Phase Counting Mode** for details.

Table 16.13 Setting and Combination of the TMDR3 Register

	TMDR3.LWA = 0		TMDR3.LWA = 1	
Register	Symbol	Access mode	Symbol	Access mode
Counter in MTU1*1	MTU1.TCNT	Word	MTU1.TCNT_1_LW	Longword
Counter in MTU2	MTU2.TCNT	Word		
General register A in MTU1	MTU1.TGRA	Word	MTU1.TGRA_1_LW	Longword
General register A in MTU2	MTU2.TGRA	Word		
General register B in MTU1	MTU1.TGRB	Word	MTU1.TGRB_1_LW	Longword
General register B in MTU2	MTU2.TGRB	Word		

Note 1. When the LWA bit is set to 1, setting the count clock for MTU1 as MTU2.TCNT overflow/underflow is not required.

16.2.6 Timer I/O Control Register (TIOR)

(1) MTU0.TIORH, MTU1.TIOR, MTU2.TIOR, MTU3.TIORH, MTU4.TIORH, MTU6.TIORH, MTU7.TIORH, MTU8.TIORH

Address(es): MTU0.TIORH H'0_1000_1302, MTU1.TIOR H'0_1000_1382, MTU2.TIOR H'0_1000_1402, MTU3.TIORH H'0_1000_1204, MTU4.TIORH H'0_1000_1206, MTU6.TIORH H'0_1000_1A04, MTU7.TIORH H'0_1000_1A06, MTU8.TIORH H'0_1000_1602

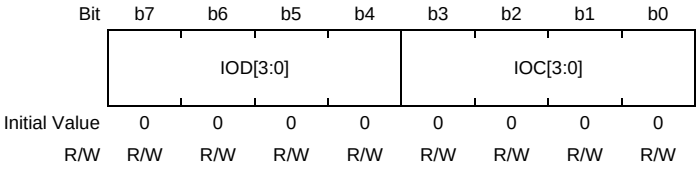
Bit	b7	b6	b5	b4	b3	b2	b1	b0
	IOB[3:0]				IOA[3:0]			
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b3 to b0	IOA[3:0]	All 0	R/W	I/O Control A*1 See the following tables. MTU0.TIORH: **Table 16.28** MTU1.TIOR: **Table 16.30** MTU2.TIOR: **Table 16.31** MTU3.TIORH: **Table 16.32** MTU4.TIORH: **Table 16.34** MTU6.TIORH: **Table 16.36** MTU7.TIORH: **Table 16.38** MTU8.TIORH: **Table 16.40**
b7 to b4	IOB[3:0]	All 0	R/W	I/O Control B*1 See the following tables. MTU0.TIORH: **Table 16.14** MTU1.TIOR: **Table 16.16** MTU2.TIOR: **Table 16.17** MTU3.TIORH: **Table 16.18** MTU4.TIORH: **Table 16.20** MTU6.TIORH: **Table 16.22** MTU7.TIORH: **Table 16.24** MTU8.TIORH: **Table 16.26**

Note 1. When the value of IOn[3:0] (n = A, B) is changed to the output-prohibited state (0000b or 0100b) during low, high, or toggle output at compare match, this register is in Hi-Z.

(2) MTU0.TIORL, MTU3.TIORL, MTU4.TIORL, MTU6.TIORL, MTU7.TIORL, MTU8.TIORL

Address(es): MTU0.TIORL H'0_1000_1303, MTU3.TIORL H'0_1000_1205, MTU4.TIORL H'0_1000_1207, MTU6.TIORL H'0_1000_1A05,MTU7.TIORL H'0_1000_1A07, MTU8.TIORL H'0_1000_1603

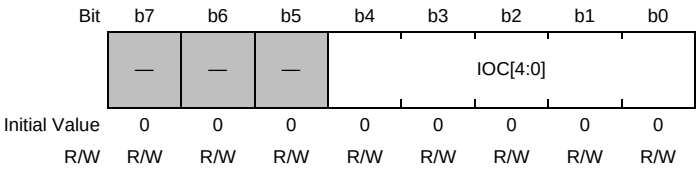


Bit	Bit Name	Initial Value	R/W	Description
b3 to b0	IOC[3:0]	All 0	R/W	I/O Control C*1 See the following tables. MTU0.TIORL: **Table 16.29** MTU3.TIORL: **Table 16.33** MTU4.TIORL: **Table 16.35** MTU6.TIORL: **Table 16.37** MTU7.TIORL: **Table 16.39** MTU8.TIORL: **Table 16.41**
b7 to b4	IOD[3:0]	All 0	R/W	I/O Control D*1 See the following tables. MTU0.TIORL: **Table 16.15** MTU3.TIORL: **Table 16.19** MTU4.TIORL: **Table 16.21** MTU6.TIORL: **Table 16.23** MTU7.TIORL: **Table 16.25** MTU8.TIORL: **Table 16.27**

Note 1. When the value of IOn[3:0] (n = C, D) is changed to the output-prohibited state (0000b or 0100b) during low, high, or toggle output at compare match, this register is in Hi-Z.

(3) MTU5.TIORU, MTU5.TIORV, MTU5.TIORW

Address(es): MTU5.TIORU H'0_1000_1C86, MTU5.TIORV H'0_1000_1C96, MTU5.TIORW H'0_1000_1CA6



Bit	Bit Name	Initial Value	R/W	Description
b4 to b0	IOC[4:0]	All 0	R/W	I/O Control C See the following table. MTU5.TIORU, MTU5.TIORV, MTU5.TIORW: **Table 16.42**
b7 to b5	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

The TIOR register controls the TGR register. The MTU has a total of 17 TIOR registers, two each for MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8, one each for MTU1 and MTU2, and three (MTU5.TIORU/TIORV/TIORW) for MTU5. The TIOR register should be set when the TMDR register setting is normal mode, PWM mode, or phase counting mode.

Note that TIOR is affected by the TMDR1 setting.

The initial output specified by TIOR is valid when the counter is stopped (the CSTn bits in TSTRA and TSTRB are cleared to 0). Also note that, in PWM mode 2, the output at the point at which the counter is cleared to 0 is specified. When TGRC or TGRD is designated for buffer operation, this setting is invalid and the register operates as a buffer register.

Table 16.14 TIORH (MTU0)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU0.TGRB Function	MTIOC0B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by TCNT (LWA = 0) or TCNTLW (LWA = 1) in MTU1*1

Remarks: Don’t care

Note 1. Input capture will not be generated in MTU0 if P0ϕ/1 is selected as the count clock for MTU1. Select a clock other than P0ϕ/1.

Table 16.15 TIORL (MTU0)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU0.TGRD Function	MTIOC0D Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register*1	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by TCNT (LWA = 0) or TCNTLW (LWA = 1) in MTU1*2

Remarks: Don’t care

Note 1. When the MTU0.TMDR1.BFB is set to 1 and MTU0.TGRD is used as a buffer register, this setting is invalid and input capture/ output compare is not generated.

Note 2. Input capture will not be generated in MTU0 if P0ϕ/1 is selected as the count clock for MTU1. Select a clock other than P0ϕ/1.

Table 16.16 TIOR (MTU1)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU1.TGRB/TGRBLW Function	MTIOC1B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	0	0		Input capture at occurrence of compare match or input capture in the MTU0.TGRC register
1	1	1	x		Input capture at occurrence of compare match in the MTU8.TGRC register

Remarks: x: Don't care

Table 16.17 TIOR (MTU2)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU2.TGRB Function	MTIOC2B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don't care

Table 16.18 TIORH (MTU3)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU3.TGRB Function	MTIOC3B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.19 TIORL (MTU3)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU3.TGRD Function	MTIOC3D Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the BFB bit in MTU3.TMDR1 is set to 1 and MTU3.TGRD is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.20 TIORH (MTU4)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU4.TGRB Function	MTIOC4B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.21 TIORL (MTU4)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU4.TGRD Function	MTIOC4D Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the BFB bit in MTU4.TMDR1 is set to 1 and MTU4.TGRD is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.22 TIORH (MTU6)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU6.TGRB Function	MTIOC6B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.23 TIORL (MTU6)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU6.TGRD Function	MTIOC6D Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the BFB bit in MTU6.TMDR1 is set to 1 and MTU6.TGRD register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.24 TIORH (MTU7)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU7.TGRB Function	MTIOC7B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.25 TIORL (MTU7)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU7.TGRD Function	MTIOC7D Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the BFB bit in MTU7.TMDR1 is set to 1 and MTU7.TGRD register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.26 TIORH (MTU8)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU8.TGRB Function	MTIOC8B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by MTU1.TCNT (LWA = 0) or MTU1.TCNTLW (LWA = 1)*1

Remarks: x: Don’t care

Note 1. Input capture will not be generated in MTU8 if P0ϕ/1 is selected as the count clock for MTU1. Select a clock other than P0ϕ/1.

Table 16.27 TIORL (MTU8)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU8.TGRD Function	MTIOC8D Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the BFB bit of the TMDR1 register is set to 1 and the TGRD register is used as a buffer register in MTU8, this setting is invalid and input capture/output compare is not generated.

Table 16.28 TIORH (MTU0)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU0.TGRA Function	MTIOC0A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	0	0		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by TCNT (LWA = 0) or TCNTLW (LWA = 1) in MTU1*1
1	1	1	x		Input capture on generation of compare match with MTU8.TGRC

Remarks: x: Don't care

Note 1. Input capture will not be generated in MTU0 if P0ϕ/1 is selected as the count clock for MTU1. Select a clock other than P0ϕ/1.

Table 16.29 TIORL (MTU0)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU0.TGRC Function	MTIOC0C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register*1	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by TCNT (LWA = 0) or TCNTLW (LWA = 1) in MTU1*2

Remarks: x: Don’t care

Note 1. When the BFA bit in MTU0.TMDR1 is set to 1 and MTU0.TGRC is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Note 2. Input capture will not be generated in MTU0 if P0ϕ/1 is selected as the count clock for MTU1. Select a clock other than P0ϕ/1.

Table 16.30 TIOR (MTU1)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU1.TGRA/TGRALW Function	MTIOC1A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Input capture at generation of MTU0.TGRA compare match/input capture.

Remarks: x: Don’t care

Table 16.31 TIOR (MTU2)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU2.TGRA Function	MTIOC2A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.32 TIORH (MTU3)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU3.TGRA Function	MTIOC3A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.33 TIORL (MTU3)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU3.TGRC Function	MTIOC3C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the BFA bit in MTU3.TMDR1 is set to 1 and MTU3.TGRC is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.34 TIORH (MTU4)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU4.TGRA Function	MTIOC4A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.35 TIORL (MTU4)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU4.TGRC Function	MTIOC4C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the BFA bit in MTU4.TMDR1 is set to 1 and MTU4.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.36 TIORH (MTU6)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU6.TGRA Function	MTIOC6A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.37 TIORL (MTU6)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU6.TGRC Function	MTIOC6C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the BFA bit in MTU6.TMDR1 is set to 1 and MTU6.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.38 TIORH (MTU7)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU7.TGRA Function	MTIOC7A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don't care

Table 16.39 TIORL (MTU7)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU7.TGRC Function	MTIOC7C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don't care

Note 1. When the BFA bit in MTU7.TMDR1 is set to 1 and MTU7.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.40 TIORH (MTU8)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU8.TGRA Function	MTIOC8A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.41 TIORL (MTU8)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU8.TGRC Function	MTIOC8C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the MTU8.TMDR1.BFA bit is set to 1 and the MTU8.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.42 TIORU, TIORV, and TIORW (MTU5)

Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC4	IOC3	IOC2	IOC1	IOC0	MTU5.TGRU, MTU5.TGRV, MTU5.TGRW Function	MTIC5U, MTIC5V, MTIC5W Pin Function
0	0	0	0	0	Output compare register	No function
0	0	0	0	1		Setting prohibited
0	0	0	1	x		Setting prohibited
0	0	1	x	x		Setting prohibited
0	1	x	x	x		Setting prohibited
1	0	0	0	0	Input capture register*1	Setting prohibited
1	0	0	0	1		Input capture at rising edge.
1	0	0	1	0		Input capture at falling edge.
1	0	0	1	1		Input capture at both edges.
1	0	1	x	x		Input capture on generation of compare match with MTU8.TGRC
1	1	0	0	0		Setting prohibited
1	1	0	0	1		Measurement of low pulse width of external input signal. Capture at trough in complementary PWM mode.
1	1	0	1	0		Measurement of low pulse width of external input signal. Capture at crest of complementary PWM mode.
1	1	0	1	1		Measurement of low pulse width of external input signal. Capture at crest and trough of complementary PWM mode.
1	1	1	0	0		Setting prohibited
1	1	1	0	1		Measurement of high pulse width of external input signal. Capture at trough in complementary PWM mode.
1	1	1	1	0		Measurement of high pulse width of external input signal. Capture at crest of complementary PWM mode.
1	1	1	1	1		Measurement of high pulse width of external input signal. Capture at crest and trough of complementary PWM mode.

Remarks: x: Don’t care

Note 1. Setting the IOC[4:0] bits to “H’19”, “H’1A”, “H’1B”, “H’1D”, “H’1E”, or “H’1F” is only allowed when the external pulse width measurement function is used or the dead time compensation function, in coordination with MTU6 and MTU7, is used. For details, refer to **Section 16.3.11, External Pulse Width Measurement**, and **Section 16.3.12, Dead Time Compensation**.

16.2.7 Timer Compare Match Clear Register (TCNTCMPCLR)

Address(es): MTU5.TCNTCMPCLR H'0_1000_1CB6

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	CMPCLR5U	CMPCLR5V	CMPCLR5W
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	CMPCLR5W	0	R/W	TCNT Compare Clear 5W 0: Disables MTU5.TCNTW to be cleared to H'0000 at MTU5.TCNTW and MTU5.TGRW compare match or input capture 1: Enables MTU5.TCNTW to be cleared to H'0000 at MTU5.TCNTW and MTU5.TGRW compare match or input capture
b1	CMPCLR5V	0	R/W	TCNT Compare Clear 5V 0: Disables MTU5.TCNTV to be cleared to H'0000 at MTU5.TCNTV and MTU5.TGRV compare match or input capture 1: Enables MTU5.TCNTV to be cleared to H'0000 at MTU5.TCNTV and MTU5.TGRV compare match or input capture
b2	CMPCLR5U	0	R/W	TCNT Compare Clear 5U 0: Disables MTU5.TCNTU to be cleared to H'0000 at MTU5.TCNTU and MTU5.TGRU compare match or input capture 1: Enables MTU5.TCNTU to be cleared to H'0000 at MTU5.TCNTU and MTU5.TGRU compare match or input capture
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TCNTCMPCLR specifies requests to clear MTU5.TCNTU, MTU5.TCNTV, and MTU5.TCNTW. The MTU has one TCNTCMPCLR (on MTU5).

16.2.8 Timer Interrupt Enable Register (TIER)

(1) MTU1.TIER, MTU2.TIER

Address(es): MTU1.TIER H'0_1000_1384, MTU2.TIER H'0_1000_1404

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) MTU0.TIER, MTU3.TIER, MTU6.TIER

Address(es): MTU0.TIER H'0_1000_1304, MTU3.TIER H'0_1000_1208, MTU6.TIER H'0_1000_1A08

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(3) MTU4.TIER, MTU7.TIER

Address(es): MTU4.TIER H'0_1000_1209, MTU7.TIER H'0_1000_1A09

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	TTGE2	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(4) MTU8.TIER

Address(es): MTU8.TIER H'0_1000_1604

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	TGIEA	0	R/W	TGR Interrupt Enable A 0: Interrupt requests (TGIA) disabled 1: Interrupt requests (TGIA) enabled
b1	TGIEB	0	R/W	TGR Interrupt Enable B 0: Interrupt requests (TGIB) disabled 1: Interrupt requests (TGIB) enabled
b2	TGIEC	0	R/W	TGR Interrupt Enable C 0: Interrupt requests (TGIC) disabled 1: Interrupt requests (TGIC) enabled
b3	TGIED	0	R/W	TGR Interrupt Enable D 0: Interrupt requests (TGID) disabled 1: Interrupt requests (TGID) enabled
b4	TCIEV	0	R/W	Overflow Interrupt Enable 0: Interrupt requests (TCIV) disabled 1: Interrupt requests (TCIV) enabled
b5	TCIEU	0	R/W	Underflow Interrupt Enable 0: Interrupt requests (TCIU) disabled 1: Interrupt requests (TCIU) enabled
b6	TTGE2	0	R/W	A/D Converter Start Request Enable 2 0: A/D converter start request generation by MTUn.TCNT underflow (trough) disabled 1: A/D converter start request generation by MTUn.TCNT underflow (trough) enabled
b7	TTGE	0	R/W	A/D Converter Start Request Enable 0: A/D converter start request generation disabled 1: A/D converter start request generation enabled

Note: n = 4, 7

The TIER register enables or disables interrupt requests from each channel. The MTU has a total of ten TIER registers, two for MTU0 and one each for MTU1 to MTU8.

TGIEA and TGIEB Bits (TGR Interrupt Enable A and B)

Each bit enables or disables interrupt requests (TGIn) (n = A, B).

TGIEC and TGIED Bits (TGR Interrupt Enable C and D)

Each bit enables or disables an interrupt request (TGIn) (n = C, D).

In MTU1 and MTU2, these bits are reserved. They are read as 0. The write value should be 0.

TCIEV Bit (Overflow Interrupt Enable)

This bit enables or disables interrupt requests (TCIV).

TCIEU Bit (Underflow Interrupt Enable)

This bit enables or disables interrupt requests (TCIU).

In MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8, this bit is reserved. It is read as 0. The write value should be 0.

TTGE2 Bit (A/D Converter Start Request Enable 2)

This bit enables or disables generation of A/D converter start requests by MTUn.TCNT underflow (trough) in complementary PWM mode (n = 4, 7).

In MTU0 to MTU3, MTU6, and MTU8, this bit is reserved. It is read as 0. The write value should be 0.

TTGE Bit (A/D Converter Start Request Enable)

This bit enables or disables generation of A/D converter start requests by TGRA input capture/compare match. MTU8 is a reserved bit. It is read as 0. The write value should be 0.

(5) MTU0.TIER2

Address(es): MTU0.TIER2 H'0_1000_1324

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE2	—	—	—	—	—	TGIEF	TGIEE
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	TGIEE	0	R/W	TGR Interrupt Enable E 0: Interrupt requests (TGIE) disabled 1: Interrupt requests (TGIE) enabled
b1	TGIEF	0	R/W	TGR Interrupt Enable F 0: Interrupt requests (TGIF) disabled 1: Interrupt requests (TGIF) enabled
b6 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b7	TTGE2	0	R/W	A/D Converter Start Request Enable 2 0: A/D converter start request generation by compare match between MTU0.TCNT and MTU0.TGRE disabled 1: A/D converter start request generation by compare match between MTU0.TCNT and MTU0.TGRE enabled

TGIEE and TGIEF Bits (TGR Interrupt Enable E and F)

Each bit enables or disables interrupt requests by compare match between MTU0.TCNT and MTU0.TGRn (n = E, F).

TTGE2 Bit (A/D Converter Start Request Enable 2)

This bit enables or disables AD converter start requests by compare match between TCNT and TGRE in MTU0.

(6) MTU5.TIER

Address(es): MTU5.TIER H'0_1000_1CB2

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TGIE5U	TGIE5V	TGIE5W
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	TGIE5W	0	R/W	TGR Interrupt Enable 5W 0: Interrupt requests TGIW5 disabled 1: Interrupt requests TGIW5 enabled
b1	TGIE5V	0	R/W	TGR Interrupt Enable 5V 0: Interrupt requests TGIV5 disabled 1: Interrupt requests TGIV5 enabled
b2	TGIE5U	0	R/W	TGR Interrupt Enable 5U 0: Interrupt requests TGIU5 disabled 1: Interrupt requests TGIU5 enabled
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TGIE5n Bits (TGR Interrupt Enable 5n)

Each bit enables or disables interrupt requests (TGIn5) (n = U, V, W).

16.2.9 Timer Status Register (TSR)

(1) MTU1.TSR, MTU2.TSR

Address(es): MTU1.TSR H'0_1000_1385, MTU2.TSR H'0_1000_1405

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TCFD	—	—	—	—	—	—	—
Initial Value	1	1	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) MTU3.TSR, MTU4.TSR, MTU6.TSR, MTU7.TSR

Address(es): MTU3.TSR H'0_1000_122C, MTU4.TSR H'0_1000_122D, MTU6.TSR H'0_1000_1A2C, MTU7.TSR H'0_1000_1A2D

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TCFD	—	—	—	—	—	—	—
Initial Value	1	1	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b5 to b0	—	All 0	R/W	Reserved The read value is undefined. The write value should be 1.
b6	—	1	R/W	Reserved This bit is read as 1. The write value should be 1.
b7	TCFD	1	R	Count Direction Flag 0: TCNT counts down 1: TCNT counts up

TSR indicates the states of each of the channels. The MTU has a total of six TSR registers, one each for MTU1 to MTU4, MTU6, and MTU7.

TCFD Flag (Count Direction Flag)

Status flag that indicates the direction in which TCNT is counting in MTU1 to MTU4, MTU6, and MTU7.

16.2.10 Timer Buffer Operation Transfer Mode Register (TBTM)

(1) MTU0.TBTM

Address(es): MTU0.TBTM H'0_1000_1326

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TTSE	TTSB	TTSA
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) MTU3.TBTM, MTU4.TBTM, MTU6.TBTM, MTU7.TBTM

Address(es): MTU3.TBTM H'0_1000_1238, MTU4.TBTM H'0_1000_1239, MTU6.TBTM H'0_1000_1A38, MTU7.TBTM H'0_1000_1A39

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	TTSB	TTSA
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	TTSA	0	R/W	Timing Select A 0: When compare match A occurs in each channel, data is transferred from TGRC to TGRA 1: When TCNT is cleared in each channel, data is transferred from TGRC to TGRA
b1	TTSB	0	R/W	Timing Select B 0: When compare match B occurs in each channel, data is transferred from TGRD to TGRB 1: When TCNT is cleared in each channel, data is transferred from TGRD to TGRB
b2	TTSE	0	R/W	Timing Select E 0: When compare match E occurs in MTU0, data is transferred from MTU0.TGRF to MTU0.TGRE 1: When MTU0.TCNT is cleared in MTU0, data is transferred from MTU0.TGRF to MTU0.TGRE
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TBTM specifies the timing for transferring data from the buffer register to the timer general register in PWM mode. The MTU has a total of five TBTM registers, one each for MTU0, MTU3, MTU4, MTU6, and MTU7.

TTSA Bit (Timing Select A)

This bit specifies the timing for transferring data from TGRC to TGRA in each channel when they are used together for buffer operation. When a channel is not set to PWM mode, do not set the TTSA bit in the channel to 1.

TTSB Bit (Timing Select B)

This bit specifies the timing for transferring data from TGRD to TGRB in each channel when they are used together for buffer operation. When a channel is not set to PWM mode, do not set the TTSB bit in the channel to 1.

TTSE Bit (Timing Select E)

This bit specifies the timing for transferring data from MTU0.TGRF to MTU0.TGRE when they are used together for buffer operation.

In MTU3, MTU4, MTU6, and MTU7, this bit is reserved. It is read as 0 and the write value should be 0. When a channel is not set to PWM mode, do not set the TTSE bit in the channel to 1.

16.2.11 Timer Input Capture Control Register (TICCR)

Address(es): MTU1.TICCR H'0_1000_1390

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	I2BE	I2AE	I1BE	I1AE
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	I1AE	0	R/W	Input Capture Enable 0: Does not include the MTIOC1A pin in the MTU2.TGRA input capture conditions 1: Includes the MTIOC1A pin in the MTU2.TGRA input capture conditions
b1	I1BE	0	R/W	Input Capture Enable 0: Does not include the TMTIOC1B pin in the MTU2.TGRB input capture conditions 1: Includes the MTIOC1B pin in the MTU2.TGRB input capture conditions
b2	I2AE	0	R/W	Input Capture Enable 0: Does not include the MTIOC2A pin in the MTU1.TGRA input capture conditions 1: Includes the MTIOC2A pin in the MTU1.TGRA input capture conditions
b3	I2BE	0	R/W	Input Capture Enable 0: Does not include the MTIOC2B pin in the MTU1.TGRB input capture conditions 1: Includes the MTIOC2B pin in the MTU1.TGRB input capture conditions
b7 to b4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TICCR specifies input capture conditions when MTU1.TCNT and MTU2.TCNT are cascaded. The MTU has one TICCR for MTU1.

16.2.12 Timer Synchronous Clear Register (TSYCR)

Address(es): MTU6.TSYCR H'0_1000_1A50

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	CE0A	CE0B	CE0C	CE0D	CE1A	CE1B	CE2A	CE2B
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	CE2B	0	R/W	Clear Enable 2B 0: Disables counter clearing by the MTU2.TGIB2 interrupt generation timing. 1: Enables counter clearing by the MTU2.TGIB2 interrupt generation timing.
b1	CE2A	0	R/W	Clear Enable 2A 0: Disables counter clearing by the MTU2.TGIA2 interrupt generation timing*1. 1: Enables counter clearing by the MTU2.TGIA2 interrupt generation timing*1.
b2	CE1B	0	R/W	Clear Enable 1B 0: Disables counter clearing by the MTU1.TGIB1 interrupt generation timing*1. 1: Enables counter clearing by the MTU1.TGIB1 interrupt generation timing*1.
b3	CE1A	0	R/W	Clear Enable 1A 0: Disables counter clearing by the MTU1.TGIA1 interrupt generation timing*1. 1: Enables counter clearing by the MTU1.TGIA1 interrupt generation timing*1.
b4	CE0D	0	R/W	Clear Enable 0D 0: Disables counter clearing by the MTU0.TGID0 interrupt generation timing*1. 1: Enables counter clearing by the MTU0.TGID0 interrupt generation timing*1.
b5	CE0C	0	R/W	Clear Enable 0C 0: Disables counter clearing by the MTU0.TGIC0 interrupt generation timing*1. 1: Enables counter clearing by the MTU0.TGIC0 interrupt generation timing*1.
b6	CE0B	0	R/W	Clear Enable 0B 0: Disables counter clearing by the MTU0.TGIB0 interrupt generation timing*1. 1: Enables counter clearing by the MTU0.TGIB0 interrupt generation timing*1.
b7	CE0A	0	R/W	Clear Enable 0A 0: Disables counter clearing by the MTU0.TGIA0 interrupt generation timing*1. 1: Enables counter clearing by the MTU0.TGIA0 interrupt generation timing*1.

Note 1. This does not depend on the TIERn.TGIEm bit setting. (n = 0, 1, 2; m = A, B, C, D)

TSYCR specifies synchronous clear conditions for MTU6.TCNT and MTU7.TCNT. The MTU has one TSYCR for MTU1.

CEnm Bits (Clear Enable nm; n = 0, 1, 2; m = A, B, C, D)

These bits enable or disable counter clearing by the MTUn.TGImn interrupt generation timing.

16.2.13 Timer Counter (TCNT)

(1) MTU0.TCNT to MTU7.TCNT

Address(es): MTU0.TCNT H'0_1000_1306, MTU1.TCNT H'0_1000_1386, MTU2.TCNT H'0_1000_1406, MTU3.TCNT H'0_1000_1210, MTU4.TCNT H'0_1000_1212, MTU5.TCNTU H'0_1000_1C80, MTU5.TCNTV H'0_1000_1C90, MTU5.TCNTW H'0_1000_1CA0, MTU6.TCNT H'0_1000_1A10, MTU7.TCNT H'0_1000_1A12

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TCNT must not be accessed in eight bits; it should be accessed in 16 bits.

(2) MTU8.TCNT

Address(es): MTU8.TCNT H'0_1000_1608

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TCNT must not be accessed in 8 or 16 bits; it should be accessed in 32 bits.

MTU0.TCNT to MTU7.TCNT are 16-bit readable/writable counters and MTU8.TCNT is a 32-bit readable/writable counter. The MTU has a total of 11 TCNT counters, one each for MTU0 to MTU4, MTU6, MTU7, and MTU8 and three (TCNTU, TCNTV, and TCNTW) for MTU5. The TCNT counters in MTU0 to MTU4, MTU6, and MTU7 are initialized to H'0000 by a reset, and the TCNT counter in MTU8 is initialized to H'0000_0000 by a reset. The TCNTU, TCNTV, and TCNTW counters in MTU5 are initialized to H'0000 by a reset. In MTU0 to MTU4, MTU6, and MTU7, the TCNT counters must not be accessed in 8-bit units; they should be accessed in 16-bit units. The MTU8.TCNT counter must not be accessed in 8- or 16-bit units; it should be accessed in 32-bit units.

The MTU1.TCNT and MTU2.TCNT counters are read as H'0000 when TMDR3.LWA is 1. See **Section 16.2.5, Timer Mode Register 3 (TMDR3)** for details.

16.2.14 Timer Longword Counter (TCNTLW)

Address(es): MTU1.TCNTLW H'0_1000_13A0

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TCNTLW must not be accessed in 8 or 16 bits; it should be accessed in 32 bits.

The TCNTLW counter is a 32-bit readable/writable counter. Only one counter of this type is provided, and is formed by combining MTU1.TCNT and MTU2.TCNT. Such operation is only effective when TMDR3.LWA is 1. The TCNTLW counter is initialized to H'0000_0000 by a reset. This counter is read as H'0000_0000 when TMDR3.LWA is 0. See **Section 16.2.5, Timer Mode Register 3 (TMDR3)** for details. This register can only be used in 32-bit phase counting mode.

16.2.15 Timer General Register (TGR)

(1) MTU0.TGR to MTU7.TGR

Address(es): MTU0.TGRA H'0_1000_1308, MTU0.TGRB H'0_1000_130A, MTU0.TGRC H'0_1000_130C, MTU0.TGRD H'0_1000_130E, MTU0.TGRE H'0_1000_1320, MTU0.TGRF H'0_1000_1322, MTU1.TGRA H'0_1000_1388, MTU1.TGRB H'0_1000_138A, MTU2.TGRA H'0_1000_1408, MTU2.TGRB H'0_1000_140A, MTU3.TGRA H'0_1000_1218, MTU3.TGRB H'0_1000_121A, MTU3.TGRC H'0_1000_1224, MTU3.TGRD H'0_1000_1226, MTU3.TGRE H'0_1000_1272, MTU4.TGRA H'0_1000_121C, MTU4.TGRB H'0_1000_121E, MTU4.TGRC H'0_1000_1228, MTU4.TGRD H'0_1000_122A, MTU4.TGRE H'0_1000_1274, MTU4.TGRF H'0_1000_1276, MTU5.TGRU H'0_1000_1C82, MTU5.TGRV H'0_1000_1C92, MTU5.TGRW H'0_1000_1CA2, MTU6.TGRA H'0_1000_1A18, MTU6.TGRB H'0_1000_1A1A, MTU6.TGRC H'0_1000_1A24, MTU6.TGRD H'0_1000_1A26, MTU6.TGRE H'0_1000_1A72, MTU7.TGRA H'0_1000_1A1C, MTU7.TGRB H'0_1000_1A1E, MTU7.TGRC H'0_1000_1A28, MTU7.TGRD H'0_1000_1A2A, MTU7.TGRE H'0_1000_1A74, MTU7.TGRF H'0_1000_1A76

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TGR must not be accessed in eight bits; it should be accessed in 16 bits. The initial value of TGR is H'FFFF.

(2) MTU8.TGR

Address(es): MTU8.TGRA H'0_1000_160C, MTU8.TGRB H'0_1000_1610, MTU8.TGRC H'0_1000_1614, MTU8.TGRD H'0_1000_1618

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TGR must not be accessed in 8 or 16 bits; it should be accessed in 32 bits.

The MTU0.TGR to MTU7.TGR registers are 16-bit readable/writable registers; the MTU8.TGR register is a 32-bit readable/writable register. The MTU has a total of 39 TGR registers, six for MTU0, two each for MTU1 and MTU2, five each for MTU3 and MTU6, six each for MTU4 and MTU7, three for MTU5 and four for MTU8.

The TGRA, TGRB, TGRC, and TGRD registers function as either output compare or input capture registers. The TGRC and TGRD registers for MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8 can also be designated for operation as buffer registers. TGR buffer register combinations are TGRA and TGRC, and TGRB and TGRD.

MTU0.TGRE and MTU0.TGRF function as compare registers. When the MTU0.TCNT count matches the MTU0.TGRE value, an A/D converter start request can be issued. The TGRF register can also be designated for operation as a buffer register. TGR buffer register combination is TGRE and TGRF. MTU5.TGRU, MTU5.TGRV, and MTU5.TGRW function as compare match, input capture, or external pulse width measurement registers.

The MTU1.TGRA, MTU2.TGRA, MTU1.TGRB, and MTU2.TGRB registers are read as H'0000 when TMDR3.LWA is 1. See **Section 16.2.5, Timer Mode Register 3 (TMDR3)** for details.

16.2.16 Timer Longword General Registers (TGRALW and TGRBLW)

Address(es): MTU1.TGRALW H'0_1000_13A4, MTU1.TGRBLW H'0_1000_13A8

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TGRALW and TGRBLW must not be accessed in 8 or 16 bits; they should be accessed in 32 bits.

The TGRALW (TGRBLW) register is a 32-bit readable/writable register. Two general registers of this type are provided, and are formed by combining the TGRA (TGRB) registers in MTU1 and MTU2. Such operation is only effective when TMDR3.LWA is 1.

The TGRALW (TGRBLW) register is initialized to H'FFFF_FFFF by a reset, but it is read as H'0000_0000 when TMDR3.LWA is 0. See **Section 16.2.5, Timer Mode Register 3 (TMDR3)** for details.

The TGRALW (TGRBLW) register functions as an output compare or input capture register when TMDR3.LWA is 1. This register can only be used in 32-bit phase counting mode.

16.2.17 Timer Start Registers (TSTRA, TSTRB, and TSTR)

(1) MTU.TSTRA (MTU0, MTU1, MTU2, MTU3, MTU4, MTU8)

Address(es): MTU.TSTRA H'0_1000_1280

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	CST4	CST3	—	—	CST8	CST2	CST1	CST0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	CST0	0	R/W	Counter Start 0 0: MTU0.TCNT counting is stopped 1: MTU0.TCNT performs count operation
b1	CST1	0	R/W	Counter Start 1 0: MTU1.TCNT counting is stopped 1: MTU1.TCNT performs count operation
b2	CST2	0	R/W	Counter Start 2 0: MTU2.TCNT counting is stopped 1: MTU2.TCNT performs count operation
b3	CST8	0	R/W	Counter start 8 0: MTU8.TCNT counting is stopped 1: MTU8.TCNT performs count operation.
b5, b4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b6	CST3	0	R/W	Counter Start 3 0: MTU3.TCNT counting is stopped 1: MTU3.TCNT performs count operation
b7	CST4	0	R/W	Counter Start 4 0: MTU4.TCNT counting is stopped 1: MTU4.TCNT performs count operation

Note: When 1 is written to a bit in TCSYSTR, the corresponding bit in TSTRA is also set to 1 automatically.

The TSTRA register starts or stops TCNT operation in MTU0 to MTU4 and MTU8. TSTRA starts or stops TCNT operation in MTU0 to MTU4.

TSTRB starts or stops TCNT operation in MTU6 and MTU7. TSTR starts or stops TCNT operation in MTU5.

Before setting the operating mode in TMDR1 or setting the TCNT count clock in TCR, be sure to stop the TCNT counting.

CSTn Bits (Counter Start n) (n = 0, 1, 2, 3, 4, 8)

Each bit starts or stops TCNT in the corresponding channel.

If 0 is written to the CSTn bit during operation with the MTIOC pin designated for output, the counter stops. In this case, the initial output value specified by the TOCR1A or TOCR2A register is output from the MTIOC pin in complementary PWM mode or reset-synchronized PWM mode.

In the other modes, the level of the output compare signal from the MTIOC pin is retained. If TIOR is written to while the CSTn bit is 0, the level of output from the pin will be updated to the specified initial output value.

(2) MTU.TSTRB(MTU6, MTU7)

Address(es): MTU.TSTRB H'0_1000_1A80

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	CST7	CST6	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b5 to b0	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b6	CST6	0	R/W	Counter Start 6 0: MTU6.TCNT counting is stopped 1: MTU6.TCNT performs count operation
b7	CST7	0	R/W	Counter Start 7 0: MTU7.TCNT counting is stopped 1: MTU7.TCNT performs count operation

Note: When 1 is written to a bit in TCSYSTR, the corresponding bit in TSTRB is also set to 1 automatically.

CSTn Bits (Counter Start n) (n = 6, 7)

Each bit starts or stops TCNT in the corresponding channel.

If 0 is written to the CSTn bit during operation with the MTIOC pin designated for output, the counter stops. In this case, the initial output value specified by the TOCR1B or TOCR2B register is output from the MTIOC pin in complementary PWM mode or reset-synchronized PWM mode.

In the other modes, the level of the output compare signal from the MTIOC pin is retained. If TIOR is written to while the CSTn bit is 0, the level of output from the pin will be updated to the specified initial output value.

(3) MTU5.TSTR(MTU5)

Address(es): MTU5.TSTR H'0_1000_1CB4

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	CSTU5	CSTV5	CSTW5
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	CSTW5	0	R/W	Counter Start W5 0: MTU5.TCNTW counting is stopped 1: MTU5.TCNTW performs count operation
b1	CSTV5	0	R/W	Counter Start V5 0: MTU5.TCNTV counting is stopped 1: MTU5.TCNTV performs count operation
b2	CSTU5	0	R/W	Counter Start U5 0: MTU5.TCNTU counting is stopped 1: MTU5.TCNTU performs count operation
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

16.2.18 Timer Synchronous Registers (TSYRA and TSYRB)

(1) MTU.TSYRA(MTU0, MTU1, MTU2, MTU3, MTU4)

Address(es): MTU.TSYRA H'0_1000_1281

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	SYNC4	SYNC3	—	—	—	SYNC2	SYNC1	SYNC0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	SYNC0	0	R/W	Timer Synchronous Operation 0 0: MTU0.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU0.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)
b1	SYNC1	0	R/W	Timer Synchronous Operation 1 0: MTU1.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU1.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)
b2	SYNC2	0	R/W	Timer Synchronous Operation 2 0: MTU2.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU2.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)
b5 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b6	SYNC3	0	R/W	Timer Synchronous Operation 3 0: MTU3.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU3.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)
b7	SYNC4	0	R/W	Timer Synchronous Operation 4 0: MTU4.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU4.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)

TSYRA selects independent operation or synchronous operation of TCNT in MTU0 to MTU4. TSYRB selects independent operation or synchronous operation of TCNT in MTU6 and MTU7. A channel performs synchronous operation when the corresponding bit in TSYR is set to 1.

SYNCn Bits (Timer Synchronous Operation n) (n = 0, 1, 2, 3, 4)

Each bit selects whether operation is independent of or synchronized with other channels.

When synchronous operation is selected, the TCNT synchronous setting of multiple channels and synchronous clearing by counter clearing on another channel are possible.

To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of the TCR.CCLR[2:0] bits.

(2) MTU.TSYRB(MTU6, MTU7)

Address(es): MTU.TSYRB H'0_1000_1A81

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	SYNC7	SYNC6	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b5 to b0	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b6	SYNC6	0	R/W	Timer Synchronous Operation 6 0: MTU6.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU6.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)
b7	SYNC7	0	R/W	Timer Synchronous Operation 7 0: MTU7.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU7.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)

SYNCn Bits (Timer Synchronous Operation n) (n = 6, 7)

Each bit selects whether operation is independent of or synchronized with other channels.

When synchronous operation is selected, the TCNT synchronous setting of multiple channels and synchronous clearing by counter clearing on another channel are possible.

To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of bits TCR.CCLR[2:0].

16.2.19 Timer Counter Synchronous Start Register (TCSYSTR)

Address(es): MTU.TCSYSTR H'0_1000_1282

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	SCH0	SCH1	SCH2	SCH3	SCH4	—	SCH6	SCH7
Initial Value	0	0	0	0	0	0	0	0
R/W	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R	R/(W)*1	R/(W)*1

Bit	Bit Name	Initial Value	R/W	Description
b0	SCH7	0	R/(W) *1	Synchronous Start 7 0: Does not specify synchronous start for MTU7.TCNT 1: Specifies synchronous start for MTU7.TCNT
b1	SCH6	0	R/(W) *1	Synchronous Start 6 0: Does not specify synchronous start for MTU6.TCNT 1: Specifies synchronous start for MTU6.TCNT
b2	—	0	R	Reserved This bit is read as 0. The write value should be 0.
b3	SCH4	0	R/(W) *1	Synchronous Start 4 0: Does not specify synchronous start for MTU4.TCNT 1: Specifies synchronous start for MTU4.TCNT
b4	SCH3	0	R/(W) *1	Synchronous Start 3 0: Does not specify synchronous start for MTU3.TCNT 1: Specifies synchronous start for MTU3.TCNT
b5	SCH2	0	R/(W) *1	Synchronous Start 2 0: Does not specify synchronous start for MTU2.TCNT 1: Specifies synchronous start for MTU2.TCNT
b6	SCH1	0	R/(W) *1	Synchronous Start 1 0: Does not specify synchronous start for MTU1.TCNT 1: Specifies synchronous start for MTU1.TCNT
b7	SCH0	0	R/(W) *1	Synchronous Start 0 0: Does not specify synchronous start for MTU0.TCNT 1: Specifies synchronous start for MTU0.TCNT

Note 1. Only 1 can be written to this bit. It is automatically cleared to 0 when counting begins.

TCSYSTR specifies synchronous start of the counters.

SCH7 Bit (Synchronous Start 7)

This bit controls synchronous start of TCNT in MTU7. [Clearing condition]

- When the CST7 bit in TSTRB is set to 1 while SCH7 = 1

SCH6 Bit (Synchronous Start 6)

This bit controls synchronous start of TCNT in MTU6. [Clearing condition]

- When the CST6 bit in TSTRB is set to 1 while SCH6 = 1

SCH4 Bit (Synchronous Start 4)

This bit controls synchronous start of TCNT in MTU4. [Clearing condition]

- When the CST4 bit in TSTRA is set to 1 while SCH4 = 1

SCH3 Bit (Synchronous Start 3)

This bit controls synchronous start of TCNT in MTU3. [Clearing condition]

- When the CST3 bit in TSTRA is set to 1 while SCH3 = 1

SCH2 Bit (Synchronous Start 2)

This bit controls synchronous start of TCNT in MTU2. [Clearing condition]

- When the CST2 bit in TSTRA is set to 1 while SCH2 = 1

SCH1 Bit (Synchronous Start 1)

This bit controls synchronous start of TCNT in MTU1. [Clearing condition]

- When the CST1 bit in TSTRA is set to 1 while SCH1 = 1

SCH0 Bit (Synchronous Start 0)

This bit controls synchronous start of TCNT in MTU0. [Clearing condition]

- When the CST0 bit in TSTRA is set to 1 while SCH0 = 1

16.2.20 Timer Read/Write Enable Registers (TRWERA and TRWERB)

Address(es): MTU.TRWERA H'0_1000_1284, MTU.TRWERB H'0_1000_1A84

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	RWE
Initial Value	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	RWE	1	R/W	Read/Write Enable 0: Read/write access to the registers is disabled 1: Read/write access to the registers is enabled
b7 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TRWERA enables or disables access to the registers and counters that have write-protection capability against accidental modification in MTU3 and MTU4.

TRWERB enables or disables access to the registers and counters that have write-protection capability against accidental modification in MTU6 and MTU7.

RWE Bit (Read/Write Enable)

This bit enables or disables access to the registers that have write-protection capability against accidental modification.

[Clearing condition]

When 0 is written to the RWE bit after reading RWE = 1

- Registers and Counters having Write-Protection Capability against Accidental Modification (TRWERA)

  24 registers: MTUn.TCR, MTUn.TCR2, MTUn.TMDR1, MTUn.TIORH, MTUn.TIORL, MTUn.TIER, MTUn.TGRA, MTUn.TGRB, TOERA, TOCR1A, TOCR2A, TGCRA, TCDRA, TDDRA, and MTUn.TCNT (n = 3, 4)

- Registers and Counters having Write-Protection Capability against Accidental Modification (TRWERB)

  23 registers: MTUn.TCR, MTUn.TCR2, MTUn.TMDR1, MTUn.TIORH, MTUn.TIORL, MTUn.TIER, MTUn.TGRA, MTUn.TGRB, TOERB, TOCR1B, TOCR2B, TCDRB, TDDRB, and MTUn.TCNT (n = 6, 7)

16.2.21 Timer Output Master Enable Registers (TOERA and TOERB)

(1) MTU.TOERA

Address(es): MTU.TOERA H'0_1000_120A

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B
Initial Value	1	1	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	OE3B	0	R/W	Master Enable MTIOC3B 0: MTU output is disabled*1 1: MTU output is enabled
b1	OE4A	0	R/W	Master Enable MTIOC4A 0: MTU output is disabled*1 1: MTU output is enabled
b2	OE4B	0	R/W	Master Enable MTIOC4B 0: MTU output is disabled*1 1: MTU output is enabled
b3	OE3D	0	R/W	Master Enable MTIOC3D 0: MTU output is disabled*1 1: MTU output is enabled
b4	OE4C	0	R/W	Master Enable MTIOC4C 0: MTU output is disabled*1 1: MTU output is enabled
b5	OE4D	0	R/W	Master Enable MTIOC4D 0: MTU output is disabled*1 1: MTU output is enabled
b7, b6	—	All 1	R/W	Reserved These bits are read as 1. The write value should be 1.

Note 1. To output the inactive level from each pin when the MTU output is set to disabled, first set the port mode register (PMn) and port register (Pn) of I/O ports to output the inactive level from general I/O ports, and then set the port mode control register (PMCn) to use general I/O ports. For details, refer to **Section 45**, **General Purpose Input Output Port (GPIO)**.

TOERA enables or disables output settings for output pins MTIOC4D, MTIOC4C, MTIOC3D, MTIOC4B, MTIOC4A, and MTIOC3B.

These pins do not output values correctly if the bits of the TOERA register are not appropriately set. Write a desired value to the TOERA register before setting up the TIOR registers in MTU3 and MTU4.

Set the TOERA register after clearing the CST3 and CST4 bits in the TSTRA register to 0 (see **Figure 16.44** and **Figure 16.48**).

(2) MTU.TOERB

Address(es): MTU.TOERB H'0_1000_1A0A

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OE7D	OE7C	OE6D	OE7B	OE7A	OE6B
Initial Value	1	1	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	OE6B	0	R/W	Master Enable MTIOC6B 0: MTU output is disabled*1 1: MTU output is enabled
b1	OE7A	0	R/W	Master Enable MTIOC7A 0: MTU output is disabled*1 1: MTU output is enabled
b2	OE7B	0	R/W	Master Enable MTIOC7B 0: MTU output is disabled*1 1: MTU output is enabled
b3	OE6D	0	R/W	Master Enable MTIOC6D 0: MTU output is disabled*1 1: MTU output is enabled
b4	OE7C	0	R/W	Master Enable MTIOC7C 0: MTU output is disabled*1 1: MTU output is enabled
b5	OE7D	0	R/W	Master Enable MTIOC7D 0: MTU output is disabled*1 1: MTU output is enabled
b7, b6	—	All 1	R/W	Reserved These bits are read as 1. The write value should be 1.

Note 1. To output the inactive level from each pin when the MTU output is set to disabled, first set the port mode register (PMn) and port register (Pn) of I/O ports to output the inactive level from general I/O ports, and then set the port mode control register (PMCn) to use general I/O ports. For details, refer to **Section 45**, **General Purpose Input Output Port (GPIO)**.

TOERB enables or disables output settings for output pins MTIOC7D, MTIOC7C, MTIOC6D, MTIOC7B, MTIOC7A, and MTIOC6B.

These pins do not output values correctly if the bits of the TOERB register are not appropriately set. Write a desired value to the TOERB register before setting up the TIOR registers in MTU6 and MTU7.

Set the TOERB register after clearing the CST6 and CST7 bits in the TSTRB register to 0 (see **Figure 16.44** and **Figure 16.48**).

16.2.22 Timer Output Control Registers 1 (TOCR1A and TOCR1B)

Address(es): MTU.TOCR1A H'0_1000_120E, MTU.TOCR1B H'0_1000_1A0E

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	PSYE	—	—	TOCL	TOCS	OLSN	OLSP
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	OLSP	0	R/W	Output Level Select P*1 *3 See **Table 16.43**.
b1	OLSN	0	R/W	Output Level Select N*1 *3 See **Table 16.44**.
b2	TOCS	0	R/W	TOC Select 0: TOCR1j setting is selected (j = A, B) 1: TOCR2j setting is selected
b3	TOCL	0	R/W	TOC Register Write Protection*2 *4 0: Write access to the TOCS, OLSN, and OLSP bits is enabled 1: Write access to the TOCS, OLSN, and OLSP bits is disabled
b5, b4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b6	PSYE	0	R/W	PWM Synchronous Output Enable 0: Toggle output is disabled 1: Toggle output is enabled
b7	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.

Note 1. Clearing the TOCR1j.TOCS bit to 0 makes this bit setting valid.

Note 2. Setting the TOCR1j.TOCL bit to 1 prevents accidental modification when the CPU goes out of control.

Note 3. If dead-time is not generated, the negative-phase output is the exact inverse of the positive-phase output. In this case, only the OLSP bit is valid.

Note 4. This bit can be set to 1 only once after a reset. After 1 is written, 0 cannot be written to the bit.

TOCR1A and TOCR1B enable or disable PWM-synchronized toggle output in complementary PWM mode and reset-synchronized PWM mode, and control inversion of PWM output level.

OLSP Bit (Output Level Select P)

This bit selects the positive-phase output level in reset-synchronized PWM mode and complementary PWM mode. The initial output is selected while the counter is stopped.

OLSN Bit (Output Level Select N)

This bit selects the negative-phase output level in reset-synchronized PWM mode and complementary PWM mode. The initial output is selected while the counter is stopped.

TOCS Bit (TOC Select)

This bit selects either the TOCR1j or TOCR2j (j = A, B) setting to be used for the output level in complementary PWM mode and reset-synchronized PWM mode.

TOCL Bit (TOC Register Write Protection)

This bit enables or disables write access to the TOCS, OLSN, and OLSP bits in TOCR1j (j = A, B).

PSYE Bit (PWM Synchronous Output Enable)

This bit enables or disables toggle output synchronized with the PWM cycle.

Table 16.43 Output Level Select Function

Bit 0	Function			
			Compare Match Output	
OLSP	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Table 16.44 Output Level Select Function

Bit 1	Function			
			Compare Match Output	
OLSN	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Figure 16.3 shows an example of output in complementary PWM mode (one phase) when OLSN = 1 and OLSP = 1.

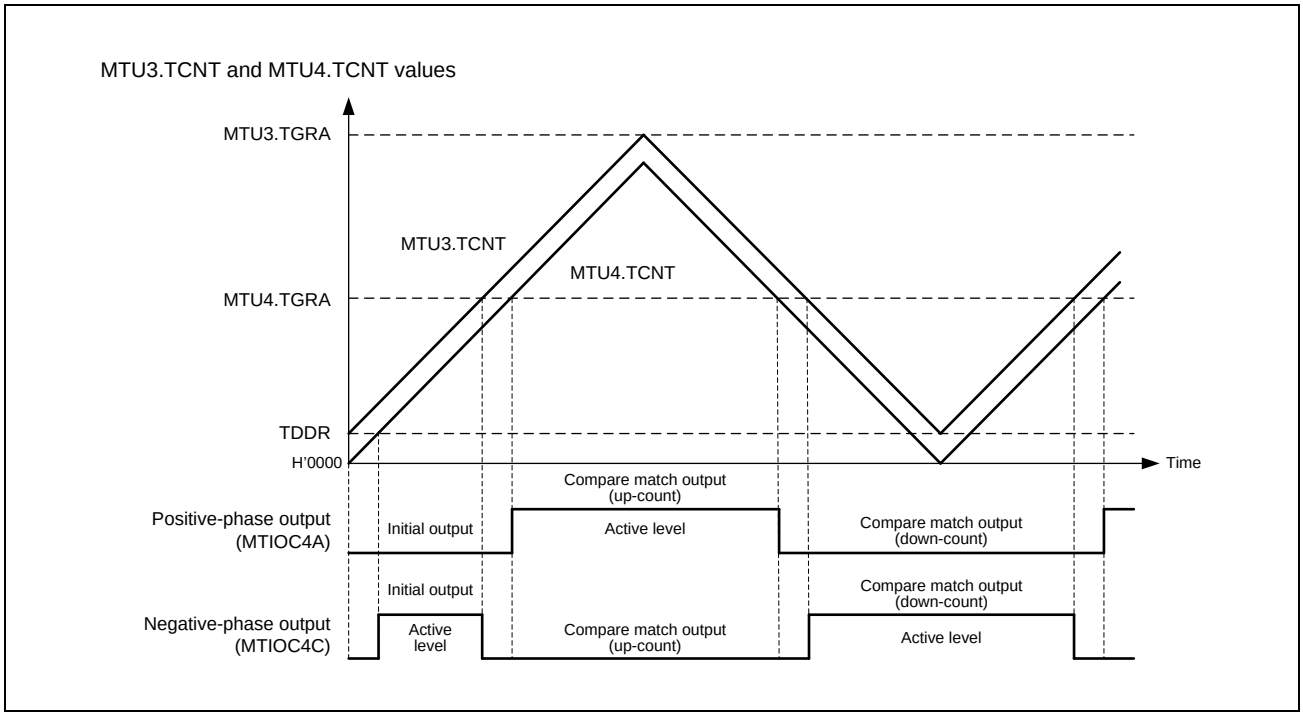


Figure 16.3 Example of Output in Complementary PWM Mode

16.2.23 Timer Output Control Registers 2 (TOCR2A and TOCR2B)

Address(es): MTU.TOCR2A H'0_1000_120F, MTU.TOCR2B H'0_1000_1A0F

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	BF[1:0]		OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	OLS1P	0	R/W	Output Level Select 1P*1 *2 This bit selects the output level on MTIOC3B or MTIOC6B in reset-synchronized PWM mode and complementary PWM mode. See **Table 16.45**.
b1	OLS1N	0	R/W	Output Level Select 1N*1 *2 This bit selects the output level on MTIOC3D or MTIOC6D in reset-synchronized PWM mode and complementary PWM mode. See **Table 16.46**.
b2	OLS2P	0	R/W	Output Level Select 2P*1 *2 This bit selects the output level on MTIOC4A or MTIOC7Ain reset-synchronized PWM mode and complementary PWM mode. See **Table 16.47**.
b3	OLS2N	0	R/W	Output Level Select 2N*1 *2 This bit selects the output level on MTIOC4C or MTIOC7C in reset-synchronized PWM mode and complementary PWM mode. See **Table 16.48**.
b4	OLS3P	0	R/W	Output Level Select 3P*1 *2 This bit selects the output level on MTIOC4B or MTIOC7B in reset-synchronized PWM mode and complementary PWM mode. See **Table 16.49**.
b5	OLS3N	0	R/W	Output Level Select 3N*1 *2 This bit selects the output level on MTIOC4D or MTIOC7D in reset-synchronized PWM mode and complementary PWM mode. See **Table 16.50**.
b7, b6	BF[1:0]	All 0	R/W	TOLBR Buffer Transfer Timing Select These bits select the timing for transferring data from TOLBRj to TOCR2j. See **Table 16.51** for details.

Note: j = A, B

Note 1. Setting the TOCR1j.TOCS bit to 1 makes this bit setting valid.

Note 2. If dead-time is not generated, the negative-phase output is the exact inverse of the positive-phase output. In this case, only the OLSiP bits are valid (i = 1 to 3).

TOCR2A and TOCR2B control inversion of PWM output level in complementary PWM mode and reset-synchronized PWM mode.

The initial output is selected while the counter is stopped.

Table 16.45 MTIOCmB Output Level Select Function

Bit 0	Function			
			Compare Match Output	
OLS1P	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Note: m = 3, 6

Table 16.46 MTIOCmD Output Level Select Function

Bit 1	Function			
			Compare Match Output	
OLS1N	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: m = 3, 6

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Table 16.47 MTIOCmA Output Level Select Function

Bit 2	Function			
			Compare Match Output	
OLS2P	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Note: m = 4, 7

Table 16.48 MTIOCmC Output Level Select Function

Bit 3	Function			
			Compare Match Output	
OLS2N	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: m = 4, 7

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Table 16.49 MTIOCmB Output Level Select Function

Bit 4	Function			
			Compare Match Output	
OLS3P	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Note: m = 4, 7

Table 16.50 MTIOCmD Output Level Select Function

Bit 5	Function			
			Compare Match Output	
OLS3N	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: m = 4, 7

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Table 16.51 Setting of TOCR2j.BF[1:0] Bits

Bit 7	Bit 6	Description	
BF1	BF0	Complementary PWM Mode	Reset-Synchronized PWM Mode
0	0	Does not transfer data from the buffer register (TOLBRj) to TOCR2j.	Does not transfer data from the buffer register (TOLBRj) to TOCR2j.
	1	Transfers data from the buffer register (TOLBRj) to TOCR2j at the crest of the MTUn.TCNT count.	Transfers data from the buffer register (TOLBRj) to TOCR2j when MTUm.TCNT or MTUn.TCNT is cleared.
1	0	Transfers data from the buffer register (TOLBRj) to TOCR2j at the trough of the MTUn.TCNT count.	Setting prohibited
	1	Transfers data from the buffer register (TOLBRj) to TOCR2j at the crest and trough of the MTUn.TCNT count.	Setting prohibited

Note: n = 4, 7;
m = 3, 6;
j = A, B

16.2.24 Timer Output Level Buffer Registers (TOLBRA and TOLBRB)

Address(es): MTU.TOLBRA H'0_1000_1236, MTU.TOLBRB H'0_1000_1A36

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	OLS1P	0	R/W	Output Level Select 1P Specify the buffer value to be transferred to the OLS1P bit in TOCR2j.
b1	OLS1N	0	R/W	Output Level Select 1N Specify the buffer value to be transferred to the OLS1N bit in TOCR2j.
b2	OLS2P	0	R/W	Output Level Select 2P Specify the buffer value to be transferred to the OLS2P bit in TOCR2j.
b3	OLS2N	0	R/W	Output Level Select 2N Specify the buffer value to be transferred to the OLS2N bit in TOCR2j.
b4	OLS3P	0	R/W	Output Level Select 3P Specify the buffer value to be transferred to the OLS3P bit in TOCR2j.
b5	OLS3N	0	R/W	Output Level Select 3N Specify the buffer value to be transferred to the OLS3N bit in TOCR2j.
b7, b6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note: j = A, B

TOLBRA and TOLBRB are buffer registers for TOCR2A and TOCR2B and specify the PWM output level in complementary PWM mode and reset-synchronized PWM mode.

Figure 16.4 shows an example of the PWM output level setting procedure in buffer operation.

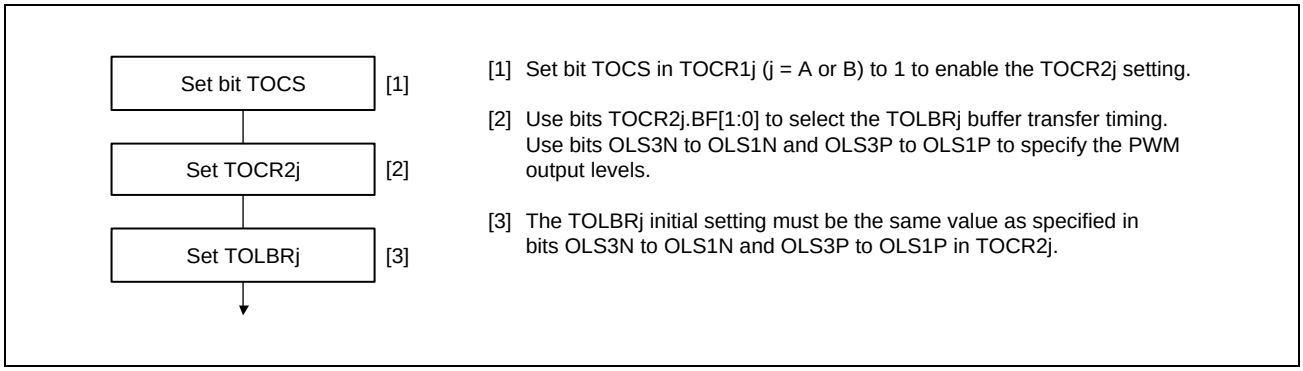


Figure 16.4 Example of PWM Output Level Setting Procedure in Buffer Operation

16.2.25 Timer Gate Control Register A (TGCRA)

Address(es): MTU.TGCRA H'0_1000_120D

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	BDC	N	P	FB	WF	VF	UF
Initial Value	1	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	UF	0	R/W	Output Phase Switch These bits turn on or off the positive-phase/negative-phase output. The setting of these bits is valid only when the FB bit is set to 1. In this case, the setting of b0 to b2 is used instead of the external input. See **Table 16.52**.
b1	VF	0	R/W	
b2	WF	0	R/W	
b3	FB	0	R/W	External Feedback Signal Enable 0: Output is switched by external input (input sources are TGRA, TGRB, and TGRC input capture signals in MTU0) 1: Output is switched by software (TGCRA's UF, VF, and WF settings)
b4	P	0	R/W	Positive-Phase Output (P) Control 0: Level output 1: Reset-synchronized PWM or complementary PWM output
b5	N	0	R/W	Negative-Phase Output (N) Control 0: Level output 1: Reset-synchronized PWM or complementary PWM output
b6	BDC	0	R/W	Brushless DC Motor 0: Ordinary output 1: Functions of this register are effective.
b7	—	1	R/W	Reserved This bit is read as 1. The write value should be 1.

TGCRA controls the output waveform necessary for brushless DC motor control in reset-synchronized PWM mode and complementary PWM mode. TGCRA register settings are ineffective for anything other than complementary PWM mode and reset-synchronized PWM mode.

UF, VF, and WF Bits (Output Phase Switch)

The setting of these bits is valid only when the FB bit is set to 1. In this case, the setting of b0 to b2 is used instead of the external input. See **Table 16.52** for details.

FB Bit (External Feedback Signal Enable)

This bit selects whether the positive-/negative-phase output is switched automatically with the TGRA, TGRB, and TGRC input capture signals in MTU0 or by writing 0 or 1 to bits 2 to 0 in TGCRA.

When the FB bit in the TGCRA register is 0, the output signals from MTU3 and MTU4 are switched automatically with the TGRA, TGRB, and TGRC input capture signals in MTU0.

P Bit (Positive-Phase Output (P) Control)

This bit selects either the level output or the reset-synchronized PWM/complementary PWM output for the positive-phase output pins (MTIOC3B, MTIOC4A, and MTIOC4B pins).

N Bit (Negative-Phase Output (N) Control)

This bit selects the level output or the reset-synchronized PWM/complementary PWM output for the negative-phase output pins (MTIOC3D, MTIOC4C, and MTIOC4D pins).

BDC Bit (Brushless DC Motor)

This bit selects whether to make the functions of TGCRA effective or ineffective.

Table 16.52 Output Level Select Function

Bit 2	Bit 1	Bit 0	Function					
			MTIOC3B	MTIOC4A	MTIOC4B	MTIOC3D	MTIOC4C	MTIOC4D
WF	VF	UF	U Phase	V Phase	W Phase	U Phase	V Phase	W Phase
0	0	0	OFF	OFF	OFF	OFF	OFF	OFF
		1	ON	OFF	OFF	OFF	OFF	ON
	1	0	OFF	ON	OFF	ON	OFF	OFF
		1	OFF	ON	OFF	OFF	OFF	ON
1	0	0	OFF	OFF	ON	OFF	ON	OFF
		1	ON	OFF	OFF	OFF	ON	OFF
	1	0	OFF	OFF	ON	ON	OFF	OFF
		1	OFF	OFF	OFF	OFF	OFF	OFF

16.2.26 Timer Subcounters (TCNTSA and TCNTSB)

Address(es): MTU.TCNTSA H'0_1000_1220, MTU.TCNTSB H'0_1000_1A20

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note: TCNTSA and TCNTSB must not be accessed in eight bits; it should be accessed in 16 bits.

TCNTSA and TCNTSB are 16-bit read-only counters that are used only in complementary PWM mode. The initial value of TCNTSA and TCNTSB after a reset is H'0000.

16.2.27 Timer Cycle Data Registers (TCDRA and TCDRB)

Address(es): MTU.TCDRA H'0_1000_1214, MTU.TCDRB H'0_1000_1A14

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TCDRA and TCDRB must not be accessed in eight bits; it should be accessed in 16 bits.

TCDRA and TCDRB are 16-bit readable/writable registers used only in complementary PWM mode. Set half the PWM carrier cycle as the TCDRA and TCDRB values. The TCDRA and TCDRB registers are constantly compared with the TCNTSA and TCNTSB counters in complementary PWM mode, respectively. When a match occurs, the TCNTSA and TCNTSB counters switch the count direction (down-count to up-count). The initial value of TCDRA and TCDRB after a reset is H'FFFF.

16.2.28 Timer Cycle Buffer Registers (TCBRA and TCBRB)

Address(es): MTU.TCBRA H'0_1000_1222, MTU.TCBRB H'0_1000_1A22

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TCBRA and TCBRB must not be accessed in eight bits; it should be accessed in 16 bits.

TCBRA and TCBRB are 16-bit readable/writable registers, used only in complementary PWM mode, that function as buffer registers for TCDRA and TCDRB. The TCBRA and TCBRB values are transferred to TCDRA and TCDRB with the transfer timing set in TMDR1. The initial value of TCBRA and TCBRB after a reset is H'FFFF.

16.2.29 Timer Dead Time Data Registers (TDDRA and TDDRB)

Address(es): MTU.TDDRA H'0_1000_1216, MTU.TDDRB H'0_1000_1A16

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TDDRA and TDDRB must not be accessed in eight bits; it should be accessed in 16 bits.

TDDRA and TDDRB are 16-bit readable/writable registers, used only in complementary PWM mode, which specify the MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) counter offset value. In complementary PWM mode, when the MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) counters are cleared and then restarted, the TDDRA (TDDRB) value is loaded into the MTU3.TCNT (MTU6.TCNT) counter and the count operation starts. The initial value of TDDRA and TDDRB after a reset is H'FFFF.

16.2.30 Timer Dead Time Enable Registers (TDERA and TDERB)

Address(es): MTU.TDERA H'0_1000_1234, MTU.TDERB H'0_1000_1A34

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	TDER
Initial Value	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/(W)

Bit	Bit Name	Initial Value	R/W	Description
b0	TDER	1	R/(W)	Dead Time Enable 0: No dead time is generated 1: Dead time is generated*1
b7 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. TDDRA and TDDRB must be set to 1 or a larger value.

TDERA and TDERB control dead time generation in complementary PWM mode. The MTU has one TDER each for MTU3 and MTU6. TDERA and TDERB should be modified only while TCNT stops.

TDER Bit (Dead Time Enable)

This bit specifies whether to generate dead time.

[Clearing condition]

- When 0 is written to TDER after reading TDER = 1

16.2.31 Timer Buffer Transfer Set Registers (TBTERA and TBTERB)

Address(es): MTU.TBTERA H'0_1000_1232, MTU.TBTERB H'0_1000_1A32

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	BTE[1:0]	
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	BTE[1:0]	All 0	R/W	Buffer Transfer Disable and Interrupt Skipping Link Setting These bits enable or disable transfer from the buffer registers*1 used in complementary PWM mode to the temporary registers, and specify whether to link the transfer with interrupt skipping function 1. For details, see **Table 16.53**.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Applicable buffer registers (TBTERA):
MTU3.TGRC, MTU3.TGRD, MTU4.TGRC, MTU4.TGRD, and TCBRA
Applicable buffer registers (TBTERB):
MTU6.TGRC, MTU6.TGRD, MTU7.TGRC, MTU7.TGRD, and TCBRB

TBTERA and TBTERB enable or disable transfer from the buffer registers used in complementary PWM mode to the temporary registers, and specify whether to link the transfer with interrupt skipping 1 operation.

Table 16.53 Setting of BTE[1:0] Bits in TBTERA and TBTERB

Bit 1	Bit 0	
BTE1	BTE0	Description
0	0	Enables transfer from the buffer registers to the temporary registers*1 and does not link the transfer with interrupt skipping function 1.
	1	Disables transfer from the buffer registers to the temporary registers.
1	0	Links transfer from the buffer registers to the temporary registers with interrupt skipping function 1.*2
	1	Setting prohibited

Note 1. Data is transferred according to the MD3 to MD0 bit setting in TMDR1. For details, refer to **Section 16.3.8, Complementary PWM Mode**.

Note 2. When interrupt skipping is disabled (the T3AEN and T4VEN (T6AEN and T7VEN) bits are cleared to 0 in the timer interrupt skipping set register (TITCR1A (TITCR1B)) or the skipping count set bits (T3ACOR and T4VCOR (T6ACOR and T7VCOR)) in TITCR1A (TITCR1B) are cleared to 0), be sure to disable link of buffer transfer with interrupt skipping (clear the BTE1 bit in the timer buffer transfer set register (TBTERA (TBTERB)) to 0). If link with interrupt skipping is enabled while interrupt skipping is disabled, buffer transfer will not be performed.

16.2.32 Timer Waveform Control Registers (TWCRA and TWCRB)

Address(es): MTU.TWCRA H'0_1000_1260, MTU.TWCRB H'0_1000_1A60

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	CCE	—	—	—	—	—	SCC	WRE
Initial Value	0*2	0	0	0	0	0	0	0
R/W	R/(W)	R/W	R/W	R/W	R/W	R/W	R/(W)	R/(W)*3

Bit	Bit Name	Initial Value	R/W	Description
b0	WRE	0	R/(W) *3	Waveform Retain Enable 0: Initial values specified in TOCR1A and TOCR2A (TOCR1B and TOCR2B) are output 1: Initial output is inhibited
b1	SCC*1 *3	0	R/(W)	Synchronous Clearing Control (Only valid in register TWCRB) 0: Clearing of MTU6.TCNT and MTU7.TCNT in response to synchronous clearing for MTU0, MTU1, MTU2–MTU6, MTU7 is enabled. 1: Clearing of MTU6.TCNT and MTU7.TCNT in response to synchronous clearing for MTU0, MTU1, MTU2–MTU6, MTU7 is disabled.
b6 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b7	CCE*2	0	R/(W)	Compare Match Clear Enable 0: Counters are not cleared at MTU3.TGRA (MTU6.TGRA) compare match 1: Counters are cleared at MTU3.TGRA (MTU6.TGRA) compare match

Note 1. This bit is only valid in register TWCRB and is a reserved bit in register TWCRA.

Note 2. Do not set to 1 when complementary PWM mode 1 is not selected.

Note 3. Do not set to 1 when complementary PWM mode is not selected.

TWCRA and TWCRB control the output waveform when synchronous counter clearing occurs in MTU3.TNCT and MTU4.TNCT (MTU6.TNCT and MTU7.TNCT) in complementary PWM mode and specifies whether to clear the counters at MTU3.TGRA (MTU6.TGRA) compare match.

The CCE bit and WRE bit in TWCRA and TWCRB should be modified only while TCNT stops.

WRE Bit (Waveform Retain Enable)

This bit selects the waveform output when synchronous counter clearing occurs in complementary PWM mode. The initial output is inhibited with this function only when synchronous clearing occurs within the Tb interval at the trough in complementary PWM mode. When synchronous clearing occurs outside this interval, the initial values

specified in TOCR1A and TOCR2A (TOCR1B and TOCR2B) are output regardless of the WRE bit setting. The initial values specified in TOCR1A and TOCR2A (TOCR1B and TOCR2B) are also output when synchronous clearing occurs in the Tb interval at the trough immediately after MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT) start operation.

For the Tb interval at the trough in complementary PWM mode, see **Figure 16.50**.

[Setting condition]

- When 1 is written to the WRE bit after reading WRE = 0

SCC Bit (Synchronous Clearing Control)

The setting of this bit selects whether MTU6.TCNT and MTU7.TCNT are or are not cleared when counter-synchronous clearing is generated for MTU0, MTU1, MTU2–MTU6, MTU7 in complementary PWM mode.

Make the complementary PWM mode settings for MTU6 and MTU7 when this function is in use. When writing a new value to the SCC bit while the counter is operating, do not change the values of the CCE and WRE bits.

Synchronous clearing from the MTU module only becomes disabled due to the setting of the SCC bit when synchronous clearing is generated outside the Tb interval in the trough. If synchronous clearing is generated within the Tb interval in the trough including immediately after the value at which MTU6.TCNT and MTU7.TCNT start, MTU6.TCNT and MTU7.TCNT are cleared.

Regarding the Tb interval in the trough in complementary PWM mode, see **Figure 16.50**.

[Setting condition]

- Writing of 1 to the SCC bit after reading it as 0

The corresponding bit in register TWCRA is reserved and is read as 0. When writing to TWCRA, write 0 to this bit.

CCE Bit (Compare Match Clear Enable)

This bit specifies whether to clear counters at MTU3.TGRA (MTU6.TGRA) compare match in complementary PWM mode.

[Setting condition]

When 1 is written to CCE after reading CCE = 0

16.2.33 Noise Filter Control Register n (NFCRn) (n = 0 to 4, 6, 7, 8, C)

(1) MTU0.NFCR0, MTU1.NFCR1, MTU2.NFCR2, MTU3.NFCR3, MTU4.NFCR4, MTU6.NFCR6, MTU7.NFCR7, MTU8.NFCR8

Address(es): MTU0.NFCR0 H'0_1000_1290, MTU1.NFCR1 H'0_1000_1291, MTU2.NFCR2 H'0_1000_1292, MTU3.NFCR3 H'0_1000_1293, MTU4.NFCR4 H'0_1000_1294, MTU6.NFCR6 H'0_1000_1A93, MTU7.NFCR7 H'0_1000_1A94, MTU8.NFCR8 H'0_1000_1298

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	NFCS[1:0]		NFDEN	NFCEN	NFBEN	NFAEN
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	NFAEN	0	R/W	Noise Filter A Enable 0: The noise filter for the MTIOCnA pin is disabled. 1: The noise filter for the MTIOCnA pin is enabled.
b1	NFBEN	0	R/W	Noise Filter B Enable 0: The noise filter for the MTIOCnB pin is disabled. 1: The noise filter for the MTIOCnB pin is enabled.
b2	NFCEN	0	R/W*1	Noise Filter C Enable 0: The noise filter for the MTIOCnC pin is disabled. 1: The noise filter for the MTIOCnC pin is enabled.
b3	NFDEN	0	R/W*1	Noise Filter D Enable 0: The noise filter for the MTIOCnD pin is disabled. 1: The noise filter for the MTIOCnD pin is enabled.
b5, b4	NFCS[1:0]	All 0	R/W	Noise Filter Clock Select b5 b4 0 0: P0ϕ/1 0 1: P0ϕ/8 1 0: P0ϕ/32 1 1: Clock source for counting
b7, b6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. These bits are reserved in the NFCR1 and NFCR2 registers. These bits are read as 0 and writing to them has no effect.

The NFCRn register (n = 0 to 4, 6, 7, or 8) specifies the noise filter function of the input capture input pin of the corresponding channel.

NFAEN Bit (Noise Filter A Enable)

This bit disables or enables the noise filter for input from the MTIOCnA pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFBEN Bit (Noise Filter B Enable)

This bit disables or enables the noise filter for input from the MTIOCnB pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFCEN Bit (Noise Filter C Enable)

This bit disables or enables the noise filter for input from the MTIOCnC pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFDEN Bit (Noise Filter D Enable)

This bit disables or enables the noise filter for input from the MTIOCnD pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFCS[1:0] Bits (Noise Filter Clock Select)

These bits set the sampling interval for the noise filters. When setting the NFCS[1:0] bits, wait for two cycles of the selected sampling interval before setting the input-capture function. When the NFCS[1:0] bits are set to 11b, i.e. selecting the external clock as the source to drive counting, wait for two cycles of the external clock before setting the input capture function.

(2) MTU0.NFCRC

Address(es): MTU0.NFCRC H'0_1000_1299

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	NFCS[1:0]		NFDEN	NFCEN	NFBEN	NFAEN
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	NFAEN	0	R/W	Noise Filter A Enable 0: The noise filter for the MTCLKA pin is disabled. 1: The noise filter for the MTCLKA pin is enabled.
b1	NFBEN	0	R/W	Noise Filter B Enable 0: The noise filter for the MTCLKB pin is disabled. 1: The noise filter for the MTCLKB pin is enabled.
b2	NFCEN	0	R/W	Noise Filter C Enable 0: The noise filter for the MTCLKC pin is disabled. 1: The noise filter for the MTCLKC pin is enabled.
b3	NFDEN	0	R/W	Noise Filter D Enable 0: The noise filter for the MTCLKD pin is disabled. 1: The noise filter for the MTCLKD pin is enabled.
b5, b4	NFCS[1:0]	All 0	R/W	Noise Filter Clock Select b5 b4 0 0: P0ϕ/1 0 1: P0ϕ/2 1 0: P0ϕ/8 1 1: P0ϕ/32
b7, b6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

The NFCRC register is used in common with all channels and specifies the noise filter function of the external clock pins.

NFAEN Bit (Noise Filter A Enable)

This bit disables or enables the noise filter for input from the MTCLKA pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFBEN Bit (Noise Filter B Enable)

This bit disables or enables the noise filter for input from the MTCLKB pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFCEN Bit (Noise Filter C Enable)

This bit disables or enables the noise filter for input from the MTCLKC pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFDEN Bit (Noise Filter D Enable)

This bit disables or enables the noise filter for input from the MTCLKD pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFCS[1:0] Bits (Noise Filter Clock Select)

These bits set the sampling interval for the noise filters. After setting the NFCS[1:0] bits, wait for two cycles of the selected sampling interval to set the input capture function.

16.2.34 Noise Filter Control Register 5 (NFCR5)

Address(es): MTU5.NFCR5 H'0_1000_1A95

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	NFCS[1:0]		—	NFWEN	NFVEN	NFUEN
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	NFUEN	0	R/W	Noise Filter U Enable 0: The noise filter for the MTIC5U pin is disabled. 1: The noise filter for the MTIC5U pin is enabled.
b1	NFVEN	0	R/W	Noise Filter V Enable 0: The noise filter for the MTIC5V pin is disabled. 1: The noise filter for the MTIC5V pin is enabled.
b2	NFWEN	0	R/W	Noise Filter W Enable 0: The noise filter for the MTIC5W pin is disabled. 1: The noise filter for the MTIC5W pin is enabled.
b3	—	0	R/W	Reserved These bits are read as 0. The write value should be 0.
b5, b4	NFCS[1:0]	All 0	R/W	Noise Filter Clock Select b5 b4 0 0: P0ϕ/1 0 1: P0ϕ/8 1 0: P0ϕ/32 1 1: Clock source for counting
b7, b6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

NFUEN Bit (Noise Filter U Enable)

This bit disables or enables the noise filter for input from the MTIC5U pin. Since unexpected edges may be internally generated when the value of this bit is changed, select the output compare function for the relevant pin in the timer I/O control register.

NFVEN Bit (Noise Filter V Enable)

This bit disables or enables the noise filter for input from the MTIC5V pin. Since unexpected edges may be internally generated when the value of this bit is changed, select the output compare function for the relevant pin in the timer I/O control register.

NFWEN Bit (Noise Filter W Enable)

This bit disables or enables the noise filter for input from the MTIC5W pin. Since unexpected edges may be internally generated when the value of this bit is changed, select the output compare function for the relevant pin in the timer I/O control register.

NFCS[1:0] Bits (Noise Filter Clock Select)

These bits set the sampling interval for the noise filters. When setting the NFCS[1:0] bits, wait for two cycles of the selected sampling interval before setting the input-capture function.

16.2.35 Timer A/D Converter Start Request Control Register (TADCR)

(1) MTU4.TADCR

Address(es): MTU4.TADCR H'0_1000_1240

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BF[1:0]		—	—	—	—	—	—	UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	ITB4VE	0	R/W	TCIV4 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG4BN and TCIV4 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG4BN and TCIV4 interrupt skipping 1 are linked
b1	ITB3AE	0	R/W	TGIA3 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG4BN and TGIA3 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG4BN and TGIA3 interrupt skipping 1 are linked
b2	ITA4VE	0	R/W	TCIV4 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG4AN and TCIV4 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG4AN and TCIV4 interrupt skipping 1 are linked
b3	ITA3AE	0	R/W	TGIA3 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG4AN and TGIA3 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG4AN and TGIA3 interrupt skipping 1 are linked
b4	DT4BE	0	R/W	Down-Count TRG4BN Enable*3 0: A/D converter start requests (TRG4BN) disabled during MTU4.TCNT down-count operation 1: A/D converter start requests (TRG4BN) enabled during MTU4.TCNT down-count operation
b5	UT4BE	0	R/W	Up-Count TRG4BN Enable 0: A/D converter start requests (TRG4BN) disabled during MTU4.TCNT up-count operation 1: A/D converter start requests (TRG4BN) enabled during MTU4.TCNT up-count operation
b6	DT4AE	0	R/W	Down-Count TRG4AN Enable*3 0: A/D converter start requests (TRG4AN) disabled during MTU4.TCNT down-count operation 1: A/D converter start requests (TRG4AN) enabled during MTU4.TCNT down-count operation
b7	UT4AE	0	R/W	Up-Count TRG4AN Enable 0: A/D converter start requests (TRG4AN) disabled during MTU4.TCNT up-count operation 1: A/D converter up requests (TRG4AN) enabled during MTU4.TCNT down-count operation
b13 to b8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Bit	Bit Name	Initial Value	R/W	Description
b15, b14	BF[1:0]	All 0	R/W	MTU4.TADCOBRA/TADCOBRB Transfer Timing Select See **Table 16.54** for details. These bits specify the transfer timing from MTU4.TADCOBRA and MTU4.TADCOBRB to MTU4.TADCORA and MTU4.TADCORB.

Note: The TADCR register in MTU4 must not be accessed in eight bits; it should be accessed in 16 bits.

Note 1. Clear the bit to 0 when interrupt skipping is disabled (the T3AEN or T4VEN bit in TITCR1A is cleared to 0 or the T3ACOR or T4VCOR bits in TITCR1A are cleared to 0).

Note 2. If link with interrupt skipping is enabled while interrupt skipping is disabled, A/D converter start requests will not be issued.

Note 3. Clear the bit to 0 when complementary PWM mode is not selected.

TADCR enables or disables A/D converter start requests and specifies whether to link A/D converter start requests with interrupt skipping function. The MTU has one TADCR each for MTU4 and MTU7.

Table 16.54 Setting of Transfer Timing by TADCR.BF[1:0] Bits (MTU4)

Bit 15	Bit 14	Description			
BF[1]	BF[0]	Complementary PWM Mode	Reset-Synchronized PWM Mode	PWM Mode 1	Normal Mode
0	0	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4).
0	1	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) at the crest of the TCNT count in MTU4.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) when a compare match occurs between TCNT and TGRA in MTU3.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) when a compare match occurs between TCNT and TGRA in MTU4.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) when a compare match occurs between TCNT and TGRA in MTU4.
1	0	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) at the trough of the TCNT count in MTU4.	Setting prohibited	Setting prohibited	Setting prohibited
1	1	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) at the crest and trough of the TCNT count in MTU4.	Setting prohibited	Setting prohibited	Setting prohibited

(2) MTU7.TADCR

Address(es): MTU7.TADCR H'0_1000_1A40

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BF[1:0]		—	—	—	—	—	—	UT7AE	DT7AE	UT7BE	DT7BE	ITA6AE	ITA7VE	ITB6AE	ITB7VE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	ITB7VE	0	R/W	TCIV7 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG7BN and TCIV7 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG7BN and TCIV7 interrupt skipping 1 are linked
b1	ITB6AE	0	R/W	TGIA6 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG7BN and TGIA6 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG7BN and TGIA6 interrupt skipping 1 are linked
b2	ITA7VE	0	R/W	TCIV7 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG7AN and TCIV7 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG7AN and TCIV7 interrupt skipping 1 are linked
b3	ITA6AE	0	R/W	TGIA6 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG7AN and TGIA6 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG7AN and TGIA6 interrupt skipping 1 are linked
b4	DT7BE	0	R/W	Down-Count TRG7BN Enable*3 0: A/D converter start requests (TRG7BN) disabled during MTU7.TCNT down-count operation 1: A/D converter start requests (TRG7BN) enabled during MTU7.TCNT down-count operation
b5	UT7BE	0	R/W	Up-Count TRG7BN Enable 0: A/D converter start requests (TRG7BN) disabled during MTU7.TCNT up-count operation 1: A/D converter start requests (TRG7BN) enabled during MTU7.TCNT up-count operation
b6	DT7AE	0	R/W	Down-Count TRG7AN Enable*3 0: A/D converter start requests (TRG7AN) disabled during MTU7.TCNT down-count operation 1: A/D converter start requests (TRG7AN) enabled during MTU7.TCNT down-count operation
b7	UT7AE	0	R/W	Up-Count TRG7AN Enable 0: A/D converter start requests (TRG7AN) disabled during MTU7.TCNT up-count operation 1: A/D converter up requests (TRG7AN) enabled during MTU7.TCNT down-count operation
b13 to b8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Bit	Bit Name	Initial Value	R/W	Description
b15, b14	BF[1:0]	All 0	R/W	MTU7.TADCOBRA/TADCOBRB Transfer Timing Select See **Table 16.55** for details. These bits specify the transfer timing from MTU7.TADCOBRA and MTU7.TADCOBRB to MTU7.TADCORA and MTU7.TADCORB.

Note: The TADCR register in MTU7 must not be accessed in eight bits; it should be accessed in 16 bits.

Note 1. Clear the bit to 0 when interrupt skipping is disabled (the T6AEN or T7VEN bit in TITCR1B is cleared to 0 or the T6ACOR or T7VCOR bits in TITCR1B are cleared to 0).

Note 2. If link with interrupt skipping is enabled while interrupt skipping is disabled, A/D converter start requests will not be issued.

Note 3. Clear the bit to 0 when complementary PWM mode is not selected.

Table 16.55 Setting of Transfer Timing by TADCR.BF[1:0] Bits (MTU7)

Bit 15	Bit 14	Description			
BF[1]	BF[0]	Complementary PWM Mode	Reset-Synchronized PWM Mode	PWM Mode 1	Normal Mode
0	0	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7).
0	1	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) at the crest of the TCNT count in MTU7.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) when a compare match occurs between TCNT and TGRA in MTU6.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) when a compare match occurs between TCNT and TGRA in MTU7.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) when a compare match occurs between TCNT and TGRA in MTU7.
1	0	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) at the trough of the TCNT count in MTU7.	Setting prohibited	Setting prohibited	Setting prohibited
1	1	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) at the crest and trough of the TCNT count in MTU7.	Setting prohibited	Setting prohibited	Setting prohibited

16.2.36 Timer A/D Converter Start Request Cycle Set Registers (TADCORA and TADCORB)

Address(es): MTU4.TADCORA H'0_1000_1244, MTU4.TADCORB H'0_1000_1246, MTU7.TADCORA H'0_1000_1A44, MTU7.TADCORB H'0_1000_1A46

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TADCORA and TADCORB must not be accessed in eight bits; it should be accessed in 16 bits.

Note 1. When the A/D converter start request delaying function linked with skipping function 1 (for details, see **Section 16.3.9(5), A/D Converter Start Request Delaying Function Linked with Interrupt Skipping Function 1**) is used, the value of this register should be H'0002 to TCDRA setting - 2 in MTU4 and H'0002 to TCDRB setting - 2 in MTU7.

Note 2. When interrupt skipping function 2 is used and the difference between the TADCORA value and the TADCORB value is small, the skipping count may not be counted correctly and the A/D converter start request may not be generated with the expected timing in some cases. The TADCORA and TADCORB values should satisfy the following conditions.

(1) When skipping function 2 is specified with the skipping count set to 0

- The difference between the TADCORA and TADCORB values should be equal to or greater than 4.
- The TADCORA compare interval should be equal to or greater than 4 P0ϕ cycles (the TADCORA update value should be the previous value + 4 or greater, or previous value − 4 or smaller).
- The TADCORB compare interval should be equal to or greater than 4 P0ϕ cycles (the TADCORB update value should be the previous value + 4 or greater, or previous value − 4 or smaller).

(2) When skipping function 2 is specified with the skipping count set to 1 or greater

- The difference between the TADCORA and TADCORB values should be equal to or greater than 2.
- The TADCORB compare interval should be equal to or greater than 2 P0ϕ cycles (the TADCORB update value should be the previous value + 2 or greater, or previous value − 2 or smaller).

TADCORA and TADCORB are 16-bit readable/writable registers that issue a corresponding A/D converter start request when the MTUn.TCNT (n = 4, 7) count reaches the value in TADCORA or TADCORB.

MTUn.TADCORA and TADCORB are initialized to H'FFFF by a reset.

16.2.37 Timer A/D Converter Start Request Cycle Set Buffer Registers (TADCOBRA and TADCOBRB)

Address(es): MTU4.TADCOBRA H'0_1000_1248, MTU4.TADCOBRB H'0_1000_124A, MTU7.TADCOBRA H'0_1000_1A48, MTU7.TADCOBRB H'0_1000_1A4A

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TADCOBRA and TADCOBRB must not be accessed in eight bits; it should be accessed in 16 bits.

TADCOBRA and TADCOBRB are 16-bit readable/writable registers whose values are transferred to TADCORA and TADCORB, respectively, when the crest or trough of the MTUn.TCNT count is reached.

TADCOBRA and TADCOBRB are initialized to H'FFFF by a reset.

16.2.38 Timer Interrupt Skipping Mode Registers (TITMRA and TITMRB)

Address(es): MTU.TITMRA H'0_1000_123A, MTU.TITMRB H'0_1000_1A3A

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	TITM
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	TITM	0	R/W	Interrupt Skipping Function Select Selects one of the two types of interrupt skipping functions. 0: Selects interrupt skipping function 1*1 1: Selects interrupt skipping function 2*2
b7 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. TITCR1A or TITCR1B is used to enable interrupt skipping function 1.

Note 2. TITCR2A or TITCR2B is used to enable interrupt skipping function 2.

TITMRA and TITMRB are used to select either of two skipping functions for the TITMRA and TITMRB registers.

16.2.39 Timer Interrupt Skipping Set Registers 1 (TITCR1A and TITCR1B)

(1) MTU.TITCR1A

Address(es): MTU.TITMRA H'0_1000_1230

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	T3AEN	T3ACOR[2:0]			T4VEN	T4VCOR[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	T4VCOR[2:0]	All 0	R/W	TCIV4 Interrupt Skipping Count Setting These bits specify the TCIV4 interrupt skipping count within the range from 0 to 7.*1 For details, see **Table 16.56**.
b3	T4VEN	0	R/W	T4VEN 0: TCIV4 interrupt skipping disabled 1: TCIV4 interrupt skipping enabled
b6 to b4	T3ACOR[2:0]	All 0	R/W	TGIA3 Interrupt Skipping Count Setting These bits specify the TGIA3 interrupt skipping count within the range from 0 to 7. For details, see **Table 16.57**.
b7	T3AEN	0	R/W	T3AEN 0: TGIA3 interrupt skipping disabled 1: TGIA3 interrupt skipping enabled

Note 1. When 0 is specified for the interrupt skipping count, no interrupt skipping will be performed.
Before changing the interrupt skipping count, be sure to clear the TITCR1A.T3AEN and TITCR1A.T4VEN bits to 0 to clear the skipping counter (TITCNT1A).

(2) MTU.TITCR1B

Address(es): MTU.TITCR1B H'0_1000_1A30

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	T6AEN	T6ACOR[2:0]			T7VEN	T7VCOR[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	T7VCOR[2:0]	All 0	R/W	TCIV7 Interrupt Skipping Count Setting These bits specify the TCIV7 interrupt skipping count within the range from 0 to 7.*1 For details, see **Table 16.58**.
b3	T7VEN	0	R/W	T7VEN 0: TCIV7 interrupt skipping disabled 1: TCIV7 interrupt skipping enabled
b6 to b4	T6ACOR[2:0]	All 0	R/W	TGIA6 Interrupt Skipping Count Setting These bits specify the TGIA6 interrupt skipping count within the range from 0 to 7.*1 For details, see **Table 16.59**.
b7	T6AEN	0	R/W	T6AEN 0: TGIA6 interrupt skipping disabled 1: TGIA6 interrupt skipping enabled

Note 1. When 0 is specified for the interrupt skipping count, no interrupt skipping will be performed.
Before changing the interrupt skipping count, be sure to clear the TITCR1B.T6AEN and TITCR1B.T7VEN bits to 0 to clear the skipping counter (TITCNT1B).

TITCR1A and TITCR1B enable or disable interrupt skipping and specify the interrupt skipping count.

The setting in the TITCR1A (TITCR1B) register is valid only while the TITM bit in TITMRA (TITMRB) is set to 0; when the TITM bit in TITMRA (TITMRB) is set to 1, the setting in TITCR1A (TITCR1B) is cleared.

Table 16.56 Setting of Interrupt Skipping Count by T4VCOR[2:0] Bits

Bit 2	Bit 1	Bit 0	
T4VCOR2	T4VCOR1	T4VCOR0	Description
0	0	0	Does not skip TCIV4 interrupts.
		1	Sets the TCIV4 interrupt skipping count to 1.
	1	0	Sets the TCIV4 interrupt skipping count to 2.
		1	Sets the TCIV4 interrupt skipping count to 3.
1	0	0	Sets the TCIV4 interrupt skipping count to 4.
		1	Sets the TCIV4 interrupt skipping count to 5.
	1	0	Sets the TCIV4 interrupt skipping count to 6.
		1	Sets the TCIV4 interrupt skipping count to 7.

Table 16.57 Setting of Interrupt Skipping Count by T3ACOR[2:0] Bits

Bit 6	Bit 5	Bit 4	
T3ACOR2	T3ACOR1	T3ACOR0	Description
0	0	0	Does not skip TGIA3 interrupts.
		1	Sets the TGIA3 interrupt skipping count to 1.
	1	0	Sets the TGIA3 interrupt skipping count to 2.
		1	Sets the TGIA3 interrupt skipping count to 3.
1	0	0	Sets the TGIA3 interrupt skipping count to 4.
		1	Sets the TGIA3 interrupt skipping count to 5.
	1	0	Sets the TGIA3 interrupt skipping count to 6.
		1	Sets the TGIA3 interrupt skipping count to 7.

Table 16.58 Setting of Interrupt Skipping Count by T7VCOR[2:0] Bits

Bit 2	Bit 1	Bit 0	
T7VCOR2	T7VCOR1	T7VCOR0	Description
0	0	0	Does not skip TCIV7 interrupts.
		1	Sets the TCIV7 interrupt skipping count to 1.
	1	0	Sets the TCIV7 interrupt skipping count to 2.
		1	Sets the TCIV7 interrupt skipping count to 3.
1	0	0	Sets the TCIV7 interrupt skipping count to 4.
		1	Sets the TCIV7 interrupt skipping count to 5.
	1	0	Sets the TCIV7 interrupt skipping count to 6.
		1	Sets the TCIV7 interrupt skipping count to 7.

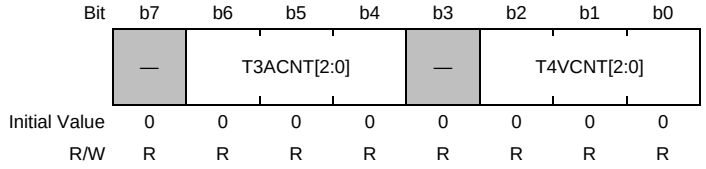
Table 16.59 Setting of Interrupt Skipping Count by T6ACOR[2:0] Bits

Bit 6	Bit 5	Bit 4	
T6ACOR2	T6ACOR1	T6ACOR0	Description
0	0	0	Does not skip TGIA6 interrupts.
		1	Sets the TGIA6 interrupt skipping count to 1.
	1	0	Sets the TGIA6 interrupt skipping count to 2.
		1	Sets the TGIA6 interrupt skipping count to 3.
1	0	0	Sets the TGIA6 interrupt skipping count to 4.
		1	Sets the TGIA6 interrupt skipping count to 5.
	1	0	Sets the TGIA6 interrupt skipping count to 6.
		1	Sets the TGIA6 interrupt skipping count to 7.

16.2.40 Timer Interrupt Skipping Counters 1 (TITCNT1A and TITCNT1B)

(1) MTU.TITCNT1A

Address(es): MTU.TITCNT1A H'0_1000_1231



Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	T4VCNT[2:0]	All 0	R	TCIV4 Interrupt Counter While the T4VEN bit in TITCR1A is set to 1, the count in these bits is incremented every time a TCIV4 interrupt occurs.
b3	—	0	R	Reserved This bit is read as 0.
b6 to b4	T3ACNT[2:0]	All 0	R	TGIA3 Interrupt Counter While the T3AEN bit in TITCR1A is set to 1, the count in these bits is incremented every time a TGIA3 interrupt occurs.
b7	—	0	R	Reserved This bit is read as 0.

Note: To clear the TITCNT1A, clear the TITCR1A.T3AEN and TITCR1A.T4VEN bits to 0.

TITCNT1A and TITCNT1B are 8-bit readable/writable counters. TITCNTA and TITCNTB retain their values even after stopping the count operation of MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT).

T4VCNT[2:0] Bits (TCIV4 Interrupt Counter)

[Clearing conditions]

- When the TITM bit in TITMRA is 1
- When the T4VEN bit in TITCR1A is cleared to 0
- When the T4VCOR[2:0] bits in TITCR1A are cleared to 000b
- When the T4VCNT[2:0] bits in TITCNT1A match the T4VCOR[2:0] bits in TITCR1A

T3ACNT[2:0] Bits (TGIA3 Interrupt Counter)

[Clearing conditions]

- When the TITM bit in TITMRA is 1
- When the T3AEN bit in TITCR1A is cleared to 0
- When the T3ACOR[2:0] bits in TITCR1A are cleared to 000b
- When the T3ACNT[2:0] bits in TITCNT1A match the T3ACOR[2:0] bits in TITCR1A

(2) MTU.TITCNT1B

Address(es): MTU.TITCNT1B H'0_1000_1A31

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	T6ACNT[2:0]			—	T7VCNT[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	T7VCNT[2:0]	All 0	R	TCIV7 Interrupt Counter While the T7VEN bit in TITCR1B is set to 1, the count in these bits is incremented every time a TCIV7 interrupt occurs.
b3	—	0	R	Reserved This bit is read as 0.
b6 to b4	T6ACNT[2:0]	All 0	R	TGIA6 Interrupt Counter While the T6AEN bit in TITCR1B is set to 1, the count in these bits is incremented every time a TGIA6 interrupt occurs.
b7	—	0	R	Reserved This bit is read as 0.

Note: To clear the TITCNT1B, clear the TITCR1B.T6AEN and TITCR1B.T7VEN bits to 0.

T7VCNT[2:0] Bits (TCIV7 Interrupt Counter)

[Clearing conditions]

- When the TITM bit in TITMRB is 1
- When the T7VEN bit in TITCR1B is cleared to 0
- When the T7VCOR[2:0] bits in TITCR1B are cleared to 000b
- When the T7VCNT[2:0] bits in TITCNT1B match the T7VCOR[2:0] bits in TITCR1B

T6ACNT[2:0] Bits (TGIA6 Interrupt Counter)

[Clearing conditions]

- When the TITM bit in TITMRB is 1
- When the T6AEN bit in TITCR1B is cleared to 0
- When the T6ACOR[2:0] bits in TITCR1B are cleared to 000b
- When the T6ACNT[2:0] bits in TITCNT1B match the T6ACOR[2:0] bits in TITCR1B

16.2.41 Timer Interrupt Skipping Set Registers 2 (TITCR2A and TITCR2B)

(1) MTU.TITCR2A

Address(es): MTU.TITCR2A H'0_1000_123B

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG4COR[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	TRG4COR [2:0]	All 0	R/W	TRG4AN/TRG4BN Interrupt Skipping Count Setting These bits specify the TRG4AN/TRG4BN interrupt skipping count within the range from 0 to 7. For details, see **Table 16.60**.
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TITCR2A and TITCR2B specify the interrupt skipping count for TRG4AN and TRG4BN (TRG7AN and TRG7BN). This setting is valid only while TITMRA or TITMRB is set to 1.

Table 16.60 Setting of Interrupt Skipping Count by TRG4COR[2:0] Bits

Bit 2	Bit 1	Bit 0	
TRG4COR2	TRG4COR1	TRG4COR0	Description
0	0	0	Does not skip TRG4AN and TRG4BN interrupts.
		1	Sets the TRG4AN and TRG4BN interrupt skipping count to 1.
	1	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 2.
		1	Sets the TRG4AN and TRG4BN interrupt skipping count to 3.
1	0	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 4.
		1	Sets the TRG4AN and TRG4BN interrupt skipping count to 5.
	1	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 6.
		1	Sets the TRG4AN and TRG4BN interrupt skipping count to 7.

(2) MTU.TITCR2B

Address(es): MTU.TITCR2B H'0_1000_1A3B

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG7COR[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	TRG7COR [2:0]	All 0	R/W	TRG7AN/TRG7BN Interrupt Skipping Count Setting These bits specify the TRG7AN/TRG7BN interrupt skipping count within the range from 0 to 7. For details, see **Table 16.61**.
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Table 16.61 Setting of Interrupt Skipping Count by TRG7COR[2:0] Bits

Bit 2	Bit 1	Bit 0	
TRG7COR2	TRG7COR1	TRG7COR0	Description
0	0	0	Does not skip TRG7AN and TRG7BN interrupts.
		1	Sets the TRG7AN and TRG7BN interrupt skipping count to 1.
	1	0	Sets the TRG7AN and TRG7BN interrupt skipping count to 2.
		1	Sets the TRG7AN and TRG7BN interrupt skipping count to 3.
1	0	0	Sets the TRG7AN and TRG7BN interrupt skipping count to 4.
		1	Sets the TRG7AN and TRG7BN interrupt skipping count to 5.
	1	0	Sets the TRG7AN and TRG7BN interrupt skipping count to 6.
		1	Sets the TRG7AN and TRG7BN interrupt skipping count to 7.

16.2.42 Timer Interrupt Skipping Counters 2 (TITCNT2A and TITCNT2B)

(1) MTU.TITCNT2A

Address(es): MTU.TITCNT2A H'0_1000_123C

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG4CNT[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	TRG4CNT [2:0]	All 0	R	TRG4AN/TRG4BN Interrupt Counter These bits start counting from the value set in TRG4COR[2:0] and the count decrements every time TRG4AN or TRG4BN is generated. When the count reaches 0 and is reloaded, the TRG4AN and TRG4BN interrupts become valid.
b7 to b3	—	All 0	R	Reserved These bits are read as 0.

TITCNT2A and TITCNT2B start counting from the values set in the TRG4COR[2:0] and TRG7COR[2:0] bits and the count decrements every time TRG4AN or TRG4BN (TITCNT2A) is generated or TRG7AN or TRG7BN (TITCNT2B) is generated. When the count reaches 0 and is reloaded, the TRG4AN and TRG4BN interrupts or the TRG7AN and TRG7BN interrupts become valid.

TRG4CNT[2:0] Bits (TRG4AN/TRG4BN Interrupt Counter)

These bits start counting from the value set in the TRG4COR[2:0] bits and the count decrements every time a TRG4AN or TRG4BN interrupt is generated. When the count reaches 0 and is reloaded, the TRG4AN and TRG4BN interrupts become valid.

[Clearing conditions]

- When the TITM bit in TITMRA is 0
- When the TRG4COR[2:0] bits in TITCR2A are cleared to 000b
- When the count of TRG4AN and TRG4BN occurrence matches the TRG4COR[2:0] value in TITCR2A

(2) MTU.TITCNT2B

Address(es): MTU.TITCNT2B H'0_1000_1A3C

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG7CNT[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	TRG7CNT [2:0]	All 0	R	TRG7AN/TRG7BN Interrupt Counter These bits start counting from the value set in TRG7COR[2:0] and the count decrements every time TRG7AN or TRG7BN is generated. When the count reaches 0 and is reloaded, the TRG7AN and TRG7BN interrupts become valid.
b7 to b3	—	All 0	R	Reserved These bits are read as 0.

TRG7CNT[2:0] Bits (TRG7AN/TRG7BN Interrupt Counter)

These bits start counting from the value set in the TRG7COR[2:0] bits and the count decrements every time a TRG7AN or TRG7BN interrupt is generated. When the count reaches 0 and is reloaded, the TRG7AN and TRG7BN interrupts become valid.

[Clearing conditions]

- When the TITM bit in TITMRB is 0
- When the TRG7COR[2:0] bits in TITCR2B are cleared to 000b
- When the count of TRG7AN and TRG7BN occurrence matches the TRG7COR[2:0] value in TITCR2B

16.2.43 Bus Master Interface

The timer counter (MTU8.TCNT), general registers (MTU8.TGRn) for MTU8, and MTU1.TCNTLW, MTU1.TGRALW, and MTU1.TGRBLW registers when TMDR3.LWA = 1 are 32-bit registers. A 32-bit data bus to the bus master enables 32-bit read/write access. 8- and 16-bit read/write are not allowed. Access these registers in 32-bit units.

Excluding MTU8, the timer counters (MTU0.TCNT to MTU7.TCNT), general registers (MTU0.TGRn to MTU7.TGRn), timer sub counters (TCNTSA and TCNTSB), timer cycle buffer registers (TCBRA and TCBRB), timer dead time data registers (TDDRA and TDDRB), timer cycle data registers (TCDRA and TCDRB), timer A/D converter start request control registers (MTU4.TADCR and MTU7.TADCR), timer A/D converter start request cycle set registers (MTU4.TADCORA, MTU4.TADCORB, MTU7.TADCORA, and MTU7.TADCORB), and timer A/D converter start request cycle set buffer registers (MTU4.TADCOBRA, MTU4.TADCOBRB, MTU7.TADCOBRA, and MTU7.TADCOBRB) are 16-bit registers. A 16-bit data bus to the bus master enables 16-bit read/write access. 8-bit read/ write is not allowed. Access the registers in 16-bit units.

All registers other than the above registers are 8-bit registers. Read from/write to these registers in 8-bit units.

16.3 Operation

16.3.1 Basic Functions

Each channel has TCNT and TGR. TCNT in each channel performs up-counting and is also capable of free-running count, periodic count, and external event count operations.

Each TGR can be used as an input capture register or an output compare register.

(1) Counter Operation

When one of bits CST0 to CST4 and CST8 in TSTRA, bits CST6 and CST7 in TSTRB, and bits CSTU5, CSTV5, and CSTW5 in MTU5.TSTR is set to 1, TCNT for the corresponding channel begins counting. TCNT can perform various count operations such as free-running count or periodic count operations.

(a) Example of Count Operation Setting Procedure

Figure 16.5 shows an example of the count operation setting procedure.

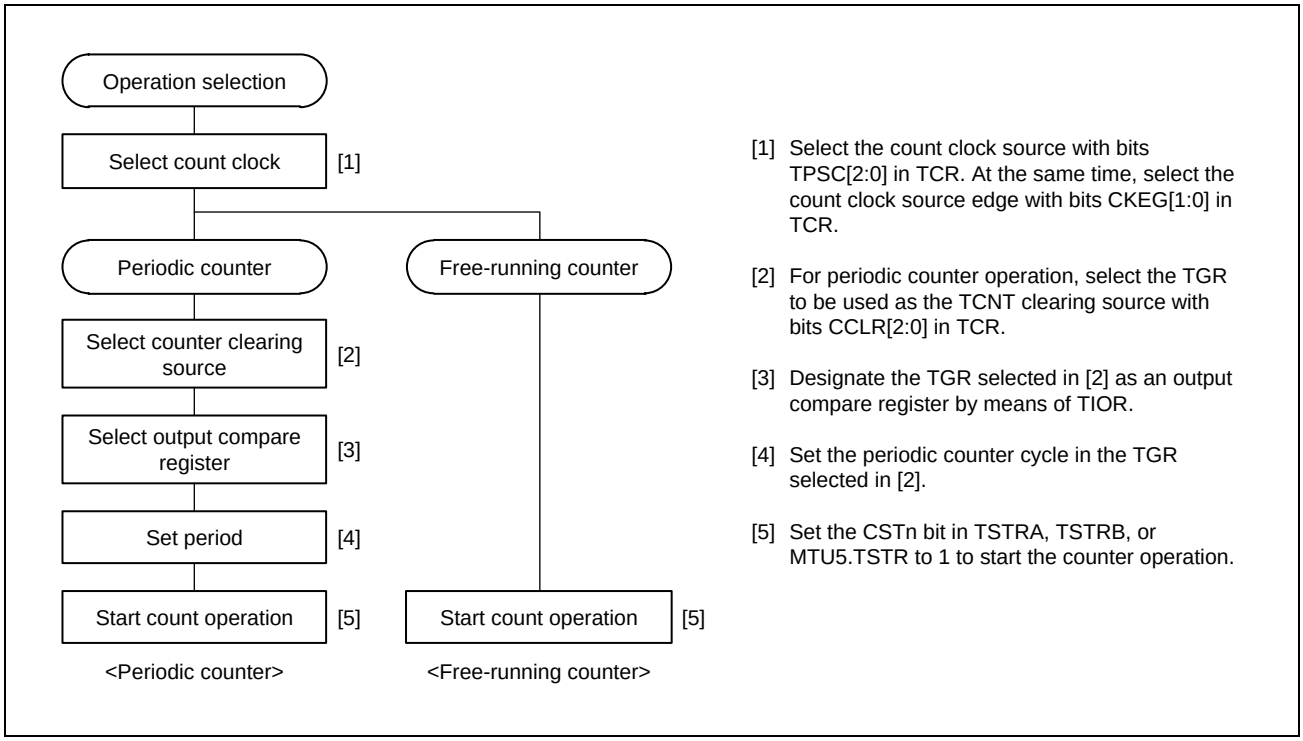


Figure 16.5 Example of Count Operation Setting Procedure

(b) Free-Running Count Operation and Periodic Count Operation

Immediately after a reset, all TCNT counters are designated as free-running counters. When the CSTn bit in TSTRA, TSTRB, or TSTR in MTU5 is set to 1, the corresponding TCNT counter starts up-counting as a free-running counter. When TCNT overflows (from H'FFFF to H'0000), if the corresponding TIER.TCIEV bit is 1, an interrupt request is issued to the CPU. After an overflow, TCNT starts counting up again from H'0000.

Figure 16.6 illustrates free-running counter operation.

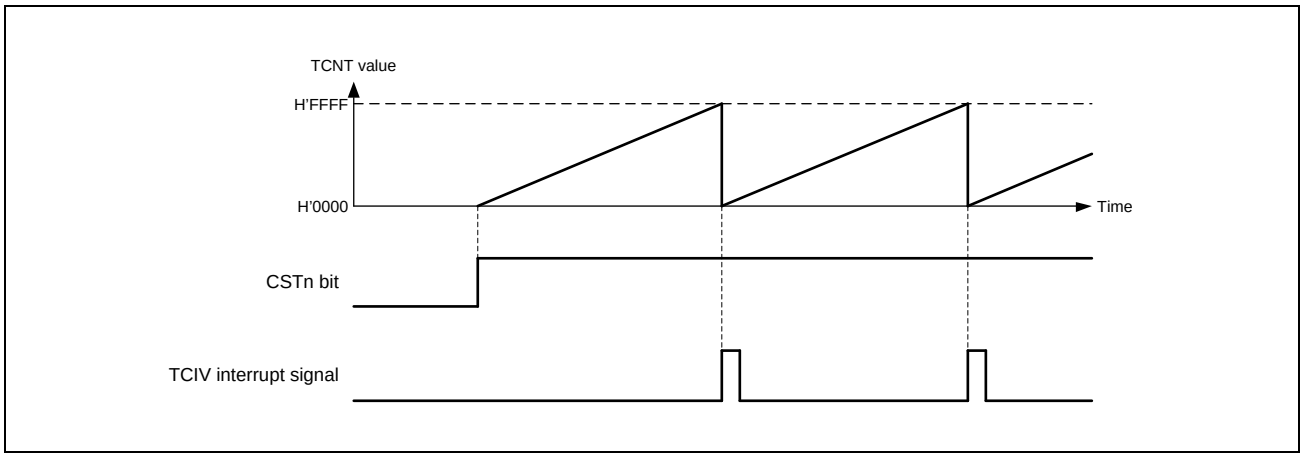


Figure 16.6 Free-Running Counter Operation

When compare match is selected as the TCNT clearing source, TCNT for the relevant channel performs periodic count operation. TGR for setting the cycle is designated as an output compare register, and counter clearing by compare match is selected by means of bits CCLR[2:0] in TCR. After the settings have been made, TCNT starts up-counting as a periodic counter when the corresponding CSTn bit in TSTRA, TSTRB, or TSTR in MTU5 is set to 1. When the count matches the value in TGR, TCNT is cleared to H'0000.

If the value of the corresponding TIER.TGIE bit is 1 at this point, an interrupt request is issued to the CPU. After a compare match, TCNT starts counting up again from H'0000.

Figure 16.7 illustrates periodic counter operation.

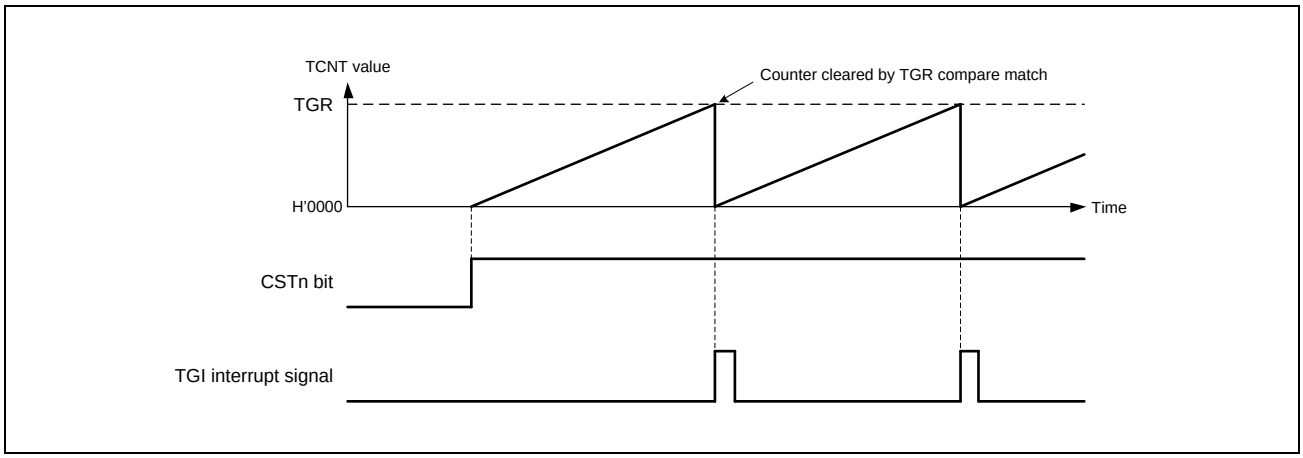


Figure 16.7 Periodic Counter Operation

(2) Waveform Output by Compare Match

Upon compare match, low, high, or toggle output from the corresponding pin can be performed. The compare match output operation is not available in MTU5.

(a) Example of Procedure for Setting Waveform Output by Compare Match

Figure 16.8 shows an example of the procedure for setting waveform output by compare match

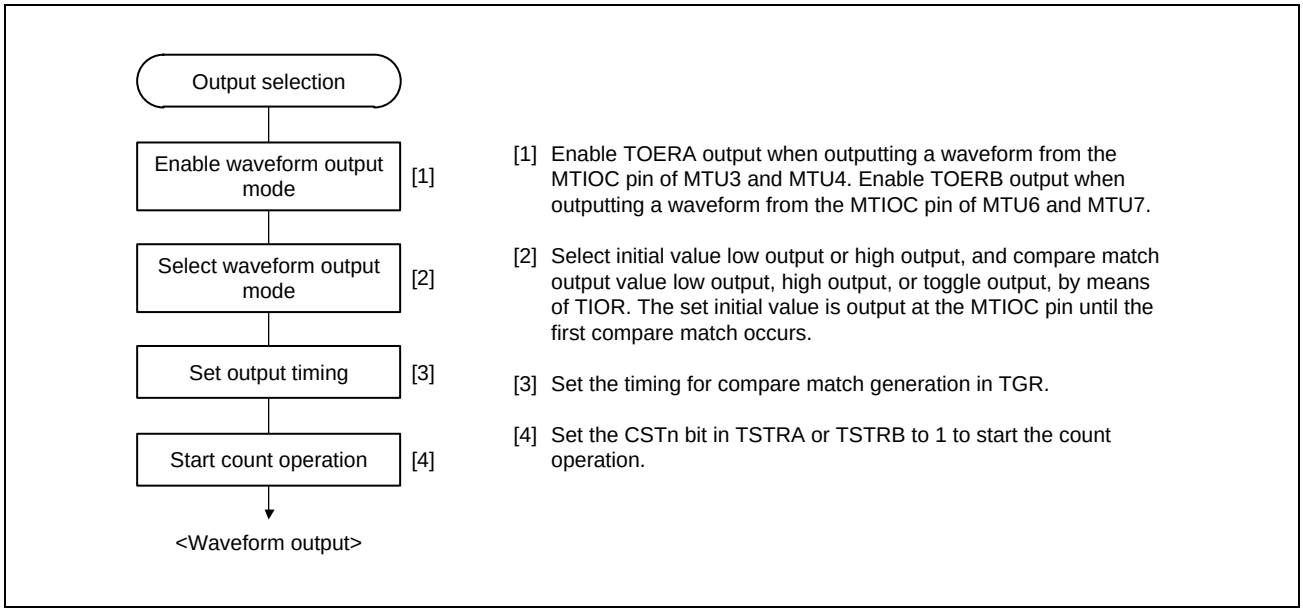


Figure 16.8 Example of Procedure for Setting Waveform Output by Compare Match

(b) Examples of Waveform Output Operation

Figure 16.9 shows an example of low output and high output.

In this example, TCNT has been designated as a free-running counter, and settings have been made so that high is output by compare match A and low is output by compare match B. When the pin level is the same as the specified level, the pin level does not change.

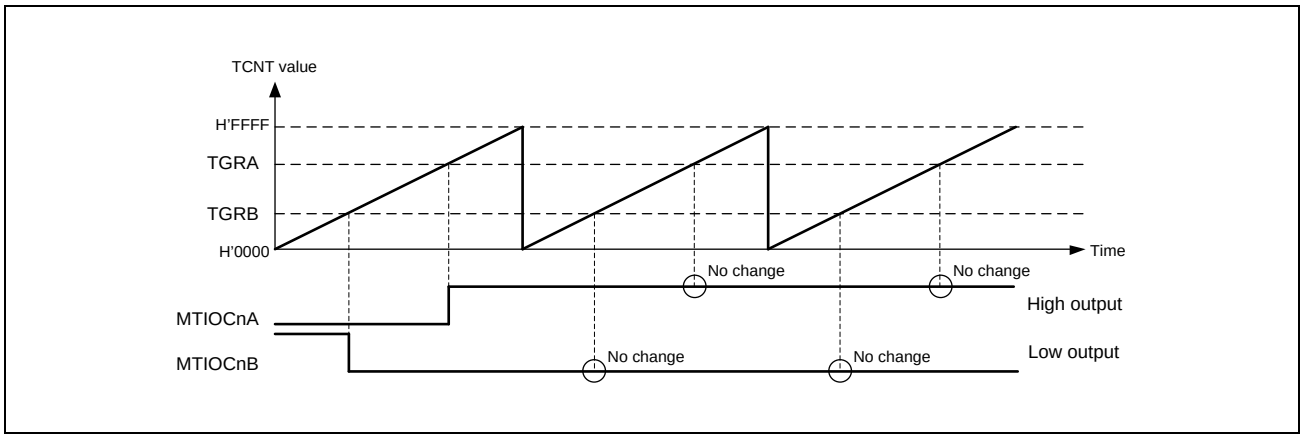


Figure 16.9 Example of low output and high output Operation (n = 0 to 4, 6, 7, 8)

Figure 16.10 shows an example of toggle output.

In this example, TCNT has been designated as a periodic counter (with counter clearing on compare match B), and settings have been made so that the output is toggled by both compare match A and compare match B.

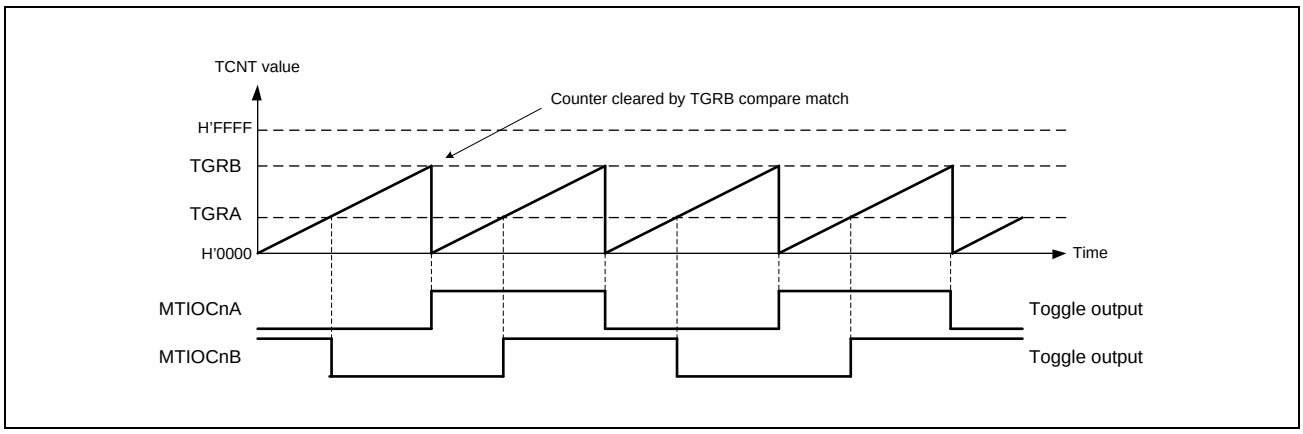


Figure 16.10 Example of Toggle Output Operation (n = 0 to 4, 6, 7, 8)

(3) Input Capture Function

The TCNT value can be transferred to TGR on detection of the MTIOCnm, MTIC5U, MTIC5V, or MTIC5W pin (n = 0 to 4, 6, 7, 8; m = A to D), input edge.

The rising edge, falling edge, or both edges can be selected as the detection edge. For MTU0 and MTU1, another channel's count clock or compare match signal can also be specified as the input capture source.

NOTE

When another channel's count clock is used as the input capture input for MTU0 and MTU1, P0ϕ/1 should not be selected as the count clock used for input capture input. Input capture will not be generated if P0ϕ/1 is selected.

(a) Example of Input Capture Operation Setting Procedure

Figure 16.11 shows an example of the input capture operation setting procedure.

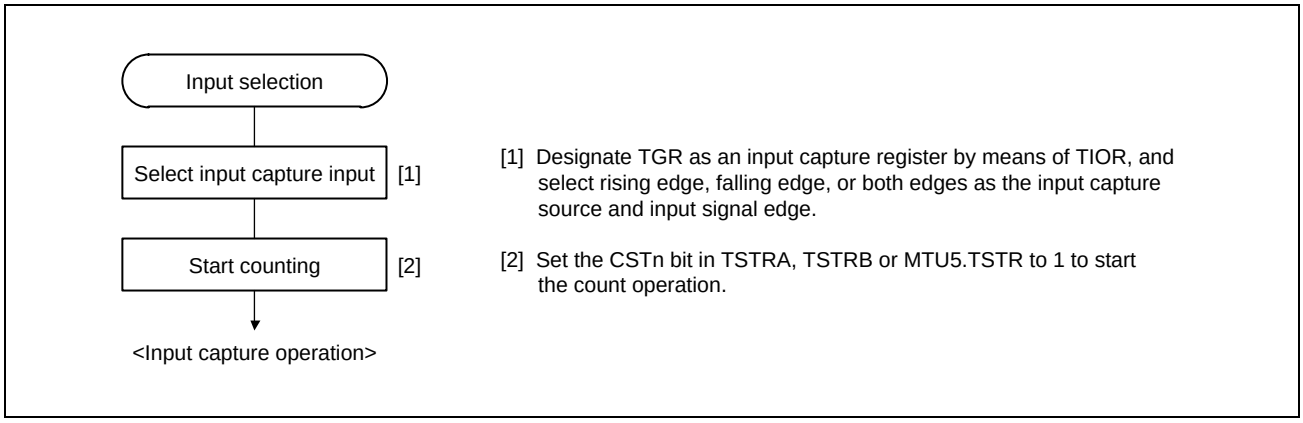


Figure 16.11 Example of Input Capture Operation Setting Procedure

(b) Example of Input Capture Operation

Figure 16.12 shows an example of input capture operation.

In this example, both rising and falling edges have been selected as the MTIOCnA pin input capture input edge, the falling edge has been selected as the MTIOCnB pin input capture input edge, and counter clearing by TGRB input capture has been designated for TCNT. (n = 0 to 4, 6, 7, 8)

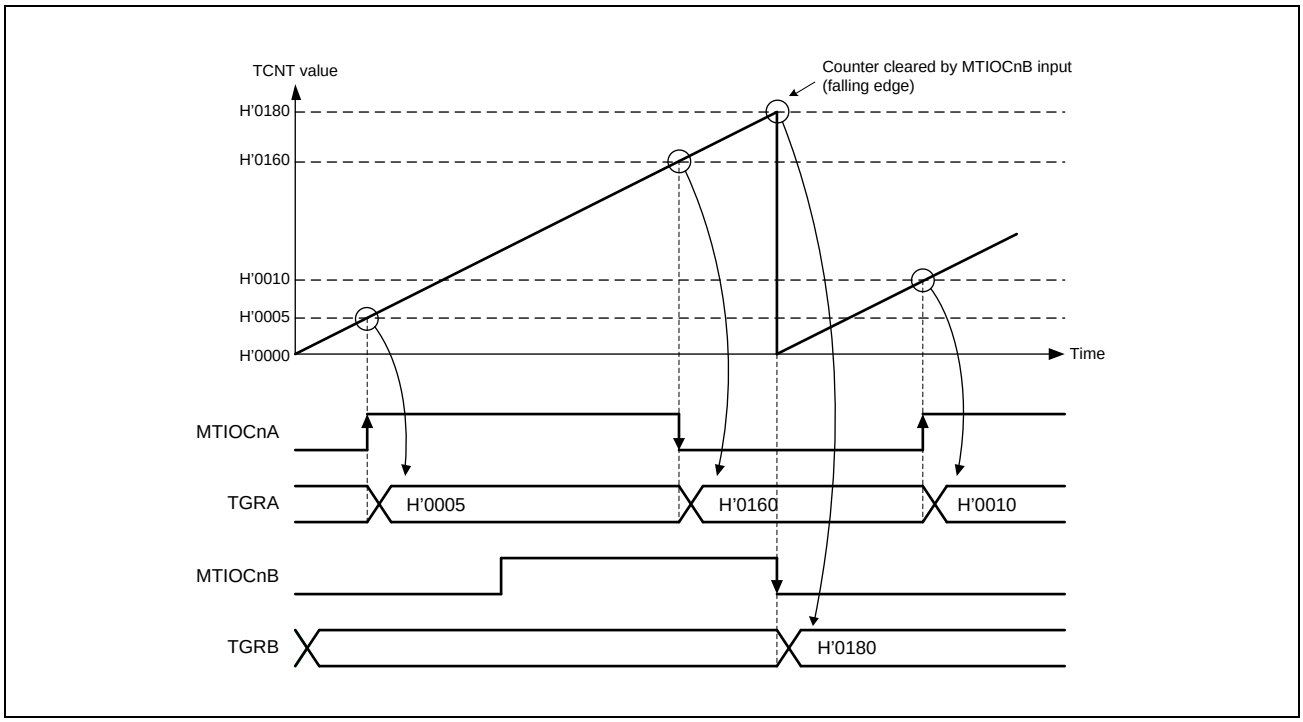


Figure 16.12 Example of Input Capture Operation (n = 0 to 4, 6, 7, 8)

16.3.2 Synchronous Operation

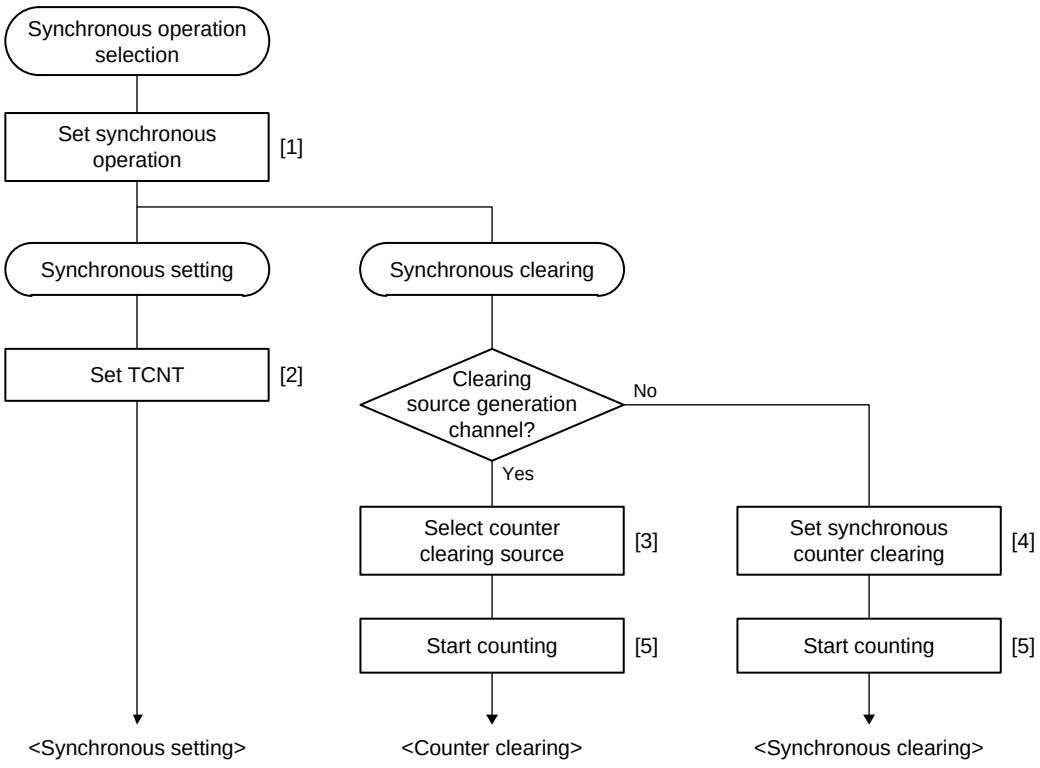
In synchronous operation, the values in multiple TCNT counters can be modified simultaneously (synchronous setting). In addition, multiple TCNT counters can be cleared simultaneously (synchronous clearing) by making the appropriate setting in TCR.

Synchronous operation can increase the number of TGR registers assigned to the same time base.

MTU0 to MTU4, MTU6, and MTU7 can all be designated for synchronous operation. MTU5 and MTU8 cannot be used for synchronous operation.

(1) Example of Synchronous Operation Setting Procedure

Figure 16.13 shows an example of the synchronous operation setting procedure.



[1] Set to 1 in the SYNC bits in TSYRA and TSYRB corresponding to the channels to be designated for synchronous operation.
[2] When the TCNT counter of any of the channels designated for synchronous operation is written to, the same value is simultaneously written to the other TCNT counters.
[3] Use the CCLR[2:0] bits in TCR to specify TCNT clearing by input capture/output compare, etc.
[4] Use the CCLR[2:0] bits in TCR to designate synchronous clearing for the counter clearing source.
[5] Set to 1 in the CSTn bits in TSTRA and TSTRB for the relevant channels, to start the count operation.

Figure 16.13 Example of Synchronous Operation Setting Procedure

(2) Example of Synchronous Operation

Figure 16.14 shows an example of synchronous operation.

In this example, synchronous operation and PWM mode 1 have been designated for MTU0 to MTU 2, MTU0.TGRB compare match has been set as the counter clearing source in MTU0, and synchronous clearing has been set for the counter clearing source in MTU1 and MTU2.

Three-phase PWM waveforms are output from pins MTIOC0A, MTIOC1A, and MTIOC2A. At this time, synchronous setting and synchronous clearing by MTU0.TGRB compare match are performed for the TCNT counters in MTU0 to MTU2, and the data set in MTU0.TGRB is used as the PWM cycle.

For details of PWM modes, see **Section 16.3.5, PWM Modes**.

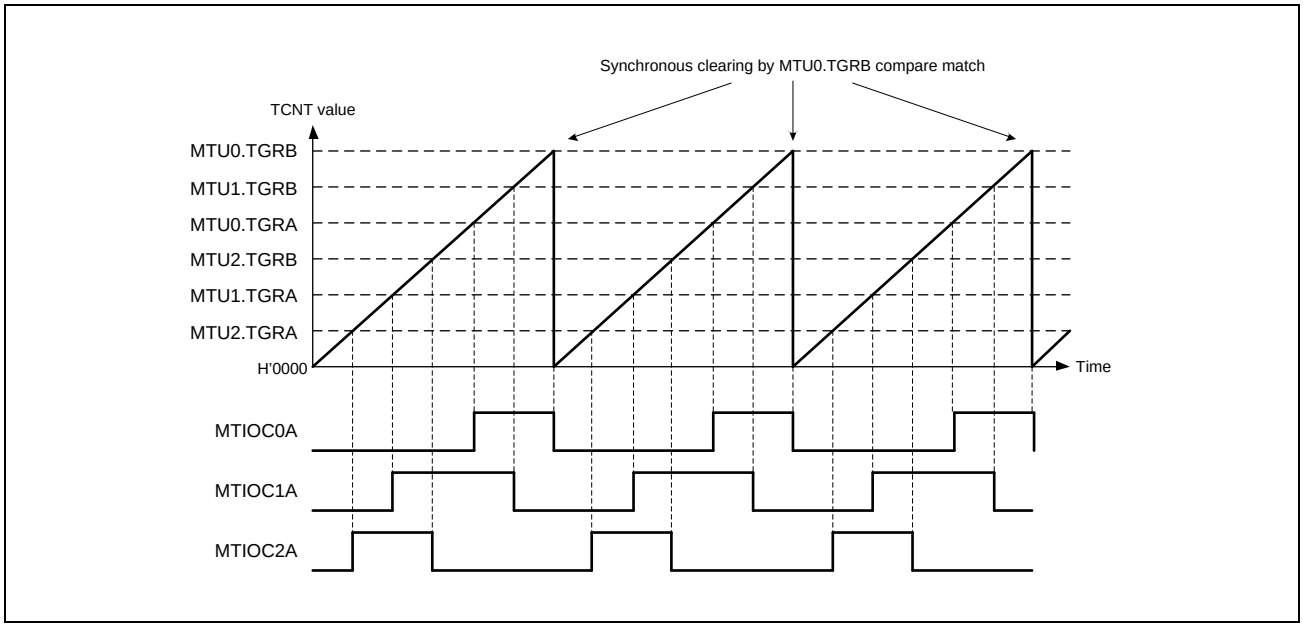


Figure 16.14 Example of Synchronous Operation

16.3.3 Buffer Operation

Buffer operation, provided for MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8, enables TGRC and TGRD to be used as buffer registers. In MTU0, TGRF can also be used as a buffer register.

Buffer operation differs depending on whether TGR has been designated as an input capture register or as a compare match register.

NOTE

MTU0.TGRE cannot be designated as an input capture register and can only operate as a compare match register.

Table 16.62 shows the register combinations used in buffer operation.

Table 16.62 Register Combinations in Buffer Operation

Channel	Timer General Register	Buffer Register
MTU0	TGRA	TGRC
	TGRB	TGRD
	TGRE	TGRF
MTU3	TGRA	TGRC
	TGRB	TGRD
MTU4	TGRA	TGRC
	TGRB	TGRD
MTU6	TGRA	TGRC
	TGRB	TGRD
MTU7	TGRA	TGRC
	TGRB	TGRD
MTU8	TGRA	TGRC
	TGRB	TGRD

When TGR is an output compare register

When a compare match occurs, the value in the buffer register for the corresponding channel is transferred to the timer general register.

This operation is illustrated in **Figure 16.15**.

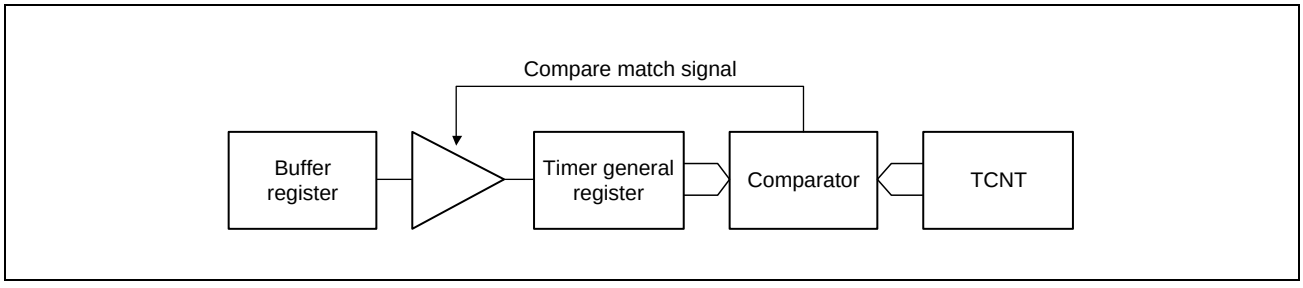


Figure 16.15 Compare Match Buffer Operation

When TGR is an input capture register

When an input capture occurs, the value in TCNT is transferred to TGR and the value previously held in TGR is transferred to the buffer register.

This operation is illustrated in **Figure 16.16**.

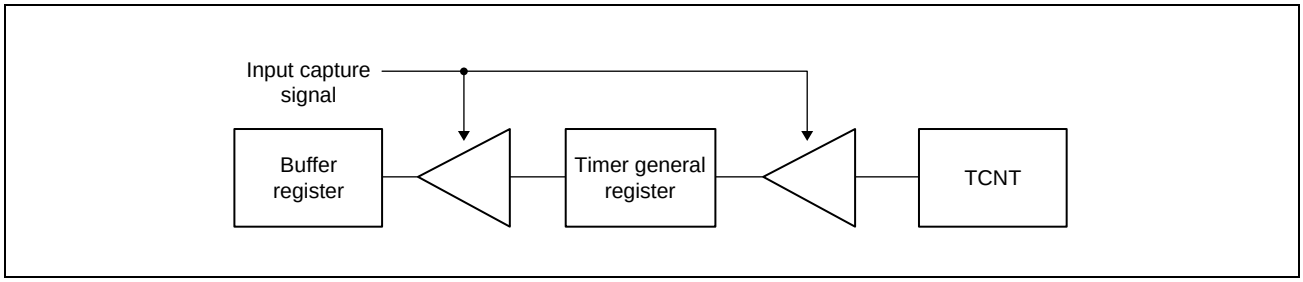


Figure 16.16 Input Capture Buffer Operation

(1) Example of Buffer Operation Setting Procedure

Figure 16.17 shows an example of the buffer operation setting procedure.

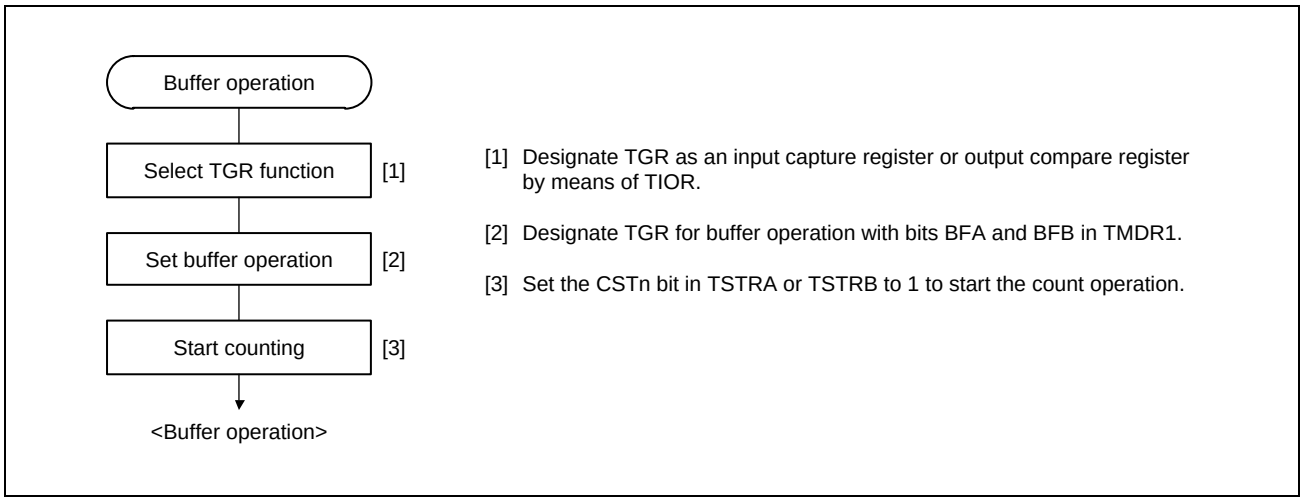


Figure 16.17 Example of Buffer Operation Setting Procedure

(2) Examples of Buffer Operation

(a) When TGR is an Output Compare Register

Figure 16.18 shows an operation example in which PWM mode 1 has been designated for MTU0, and buffer operation has been designated for TGRA and TGRC. The settings used in this example are TCNT clearing by compare match B, high output at compare match A, and low output at compare match B. In this example, the TTSA bit in TBTM is cleared to 0.

As buffer operation has been set, when compare match A occurs, the output changes and the value in buffer register TGRC is simultaneously transferred to timer general register TGRA. This operation is repeated each time compare match A occurs.

For details of PWM modes, see **Section 16.3.5, PWM Modes**.

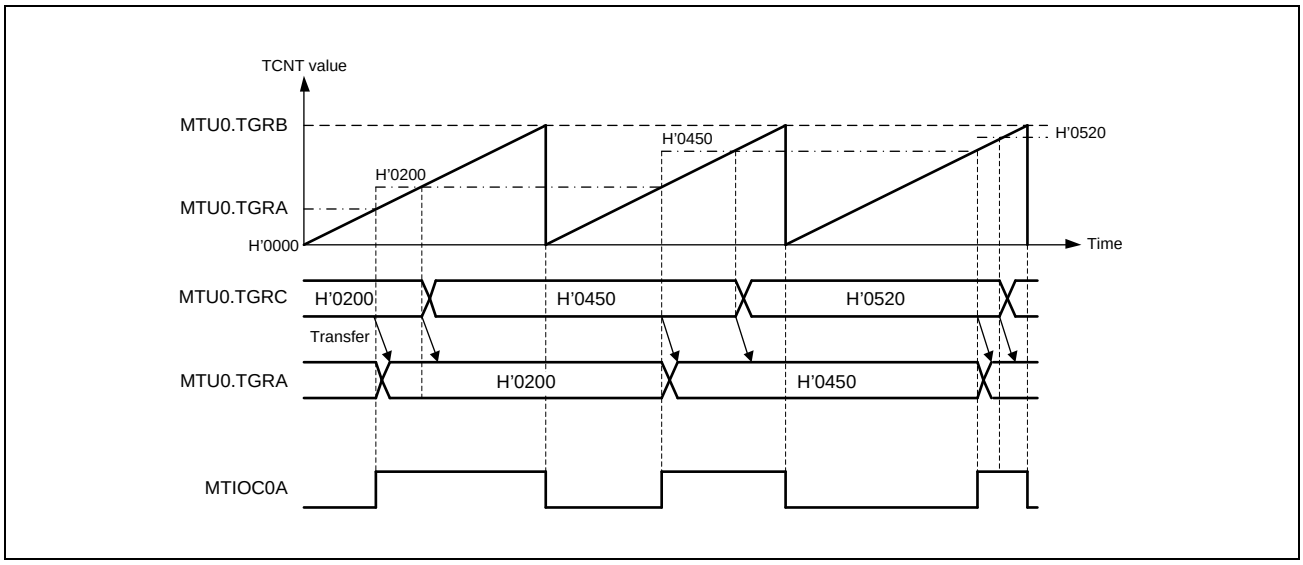


Figure 16.18 Example of Buffer Operation (1)

(b) When TGR is an Input Capture Register

Figure 16.19 shows an operation example in which TGRA has been designated as an input capture register, and buffer operation has been designated for TGRA and TGRC.

Counter clearing by TGRA input capture has been set for TCNT, and both rising and falling edges have been selected as the MTIOCnA pin input capture input edge. (n = 0 to 4, 6, 7, 8)

As buffer operation has been set, when the TCNT value is transferred to TGRA upon occurrence of input capture A, the value previously stored in TGRA is simultaneously transferred to TGRC.

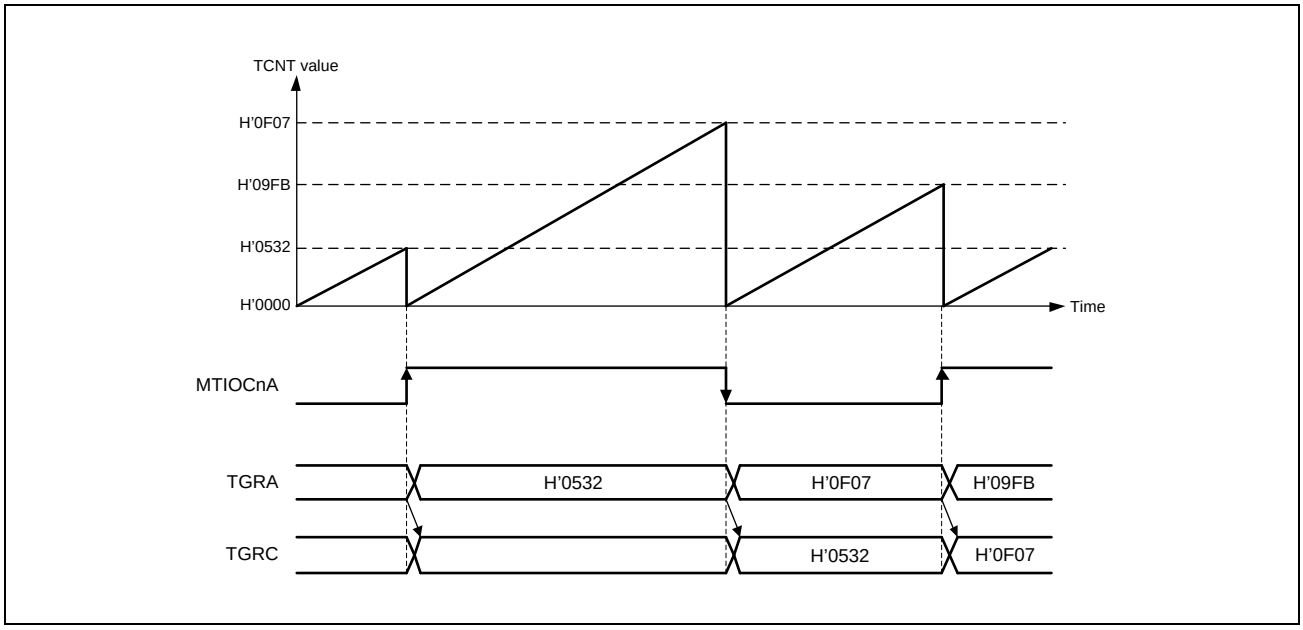


Figure 16.19 Example of Buffer Operation (2) (n = 0 to 4, 6, 7, 8)

(3) Selecting Timing for Transfer from Buffer Registers to Timer General Registers in Buffer Operation

The timing for transfer from buffer registers to timer general registers can be selected in PWM mode 1 or 2 for MTU0 or in PWM mode 1 for MTU3, MTU4, MTU6, and MTU7 by setting the buffer operation transfer mode registers (MTUn.TBTM (n = 0, 3, 4, 6, 7)). Either compare match (value after reset) or TCNT clearing can be selected for the transfer timing. TCNT clearing as transfer timing is one of the following cases.

- When TCNT overflows (H'FFFF to H'0000)
- When H'0000 is written to TCNT during counting
- When TCNT is cleared to H'0000 under the condition specified in the CCLR[2:0] bits in TCR

NOTE

TBTM must be modified only while TCNT stops.

Figure 16.20 shows an operation example in which PWM mode 1 is designated for MTU0 and buffer operation is designated for MTU0.TGRA and MTU0.TGRC. The settings used in this example are MTU0.TCNT clearing by compare match B, high output at compare match A, and low output at compare match B. The TTSA bit in MTU0.TBTM is set to 1.

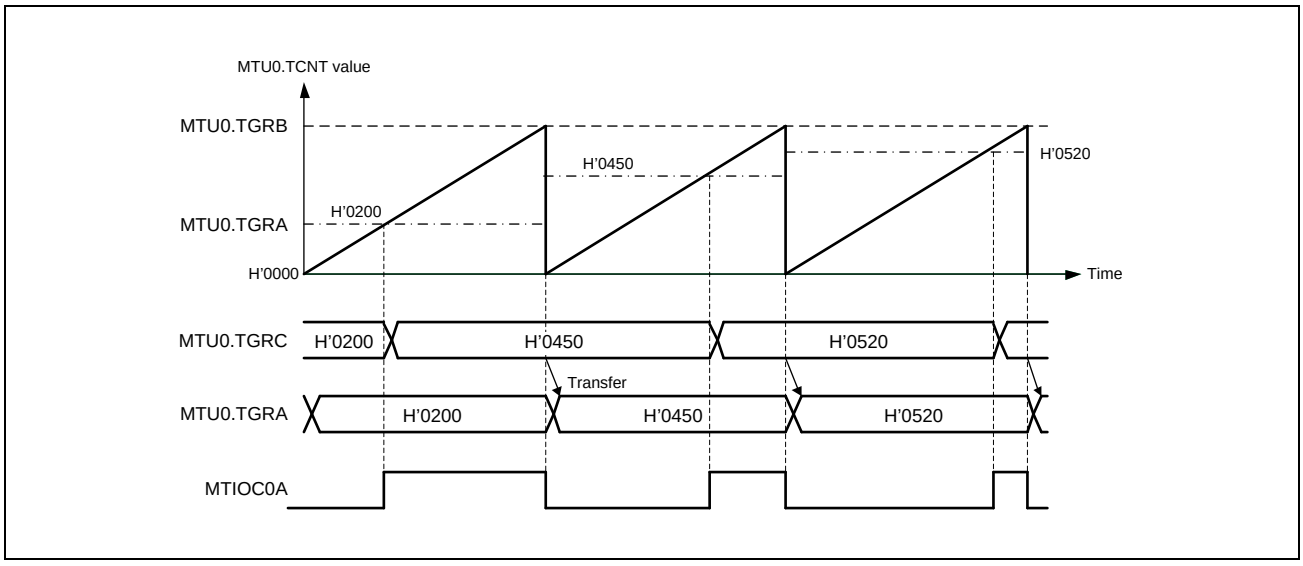


Figure 16.20 Example of Buffer Operation When MTU0.TCNT Clearing is Selected for MTU0.TGRC to MTU0.TGRA Transfer Timing

16.3.4 Cascaded Operation

In cascaded operation, two 16-bit counters in different channels are used together as a 32-bit counter.

There are two functions for connecting MTU1 and MTU2 to use as a 32-bit counter: cascade connection to be set when the MTU1.TMDR3.LWA bit is 0, and cascade connection 32-bit phase counting mode to be set when the MTU1.TMDR3.LWA bit is 1. For details on cascade connection 32-bit phase counting mode, refer to **Section 16.3.6.2, Cascade Connection 32-Bit Phase Counting Mode**. This section describes the cascade connection function to be set when the MTU1.TMDR3.LWA bit is 0.

This function works when the LWA bit of TMDR3 in MTU1 is set to 0 and the TPSC[2:0] bits of TCR in MTU1 are set so that TCNT in MTU1 counts at an overflow/underflow of TCNT in MTU2.

Underflow occurs only when the MTU2 to which the lower 16 bits allocated is in phase counting mode.

Table 16.63 shows the register combinations used in cascaded operation.

NOTE

When phase counting mode is set for MTU1, the count clock setting is invalid and the counters operate independently in phase counting mode.

Table 16.63 Cascaded Combinations

Combination	Upper 16 Bits	Lower 16 Bits
MTU1 and MTU2	MTU1.TCNT	MTU2.TCNT

For simultaneous input capture of MTU1.TCNT and MTU2.TCNT during cascaded operation, additional input capture input pins can be specified by the input capture control register (TICCR). The input-capture condition is of edges in the signal produced by taking the logical OR of the input level on the main input pin and the input level on the added input pin. Accordingly, if either is at the high level, a change in the level of the other will not produce an edge for detection. For details, see **Section 16.3.4(4), Cascaded Operation Example (c)**. For input capture in cascade connection, refer to **Section 16.6.20, Simultaneous Input Capture in MTU1.TCNT and MTU2.TCNT in Cascade Connection**.

Table 16.64 shows the TICCR setting and input capture input pins.

Table 16.64 TICCR Setting and Input Capture Input Pins

Target Input Capture	TICCR Setting	Input Capture Input Pin
Input capture from MTU1.TCNT to MTU1.TGRA	I2AE bit = 0 (Initial value)	MTIOC1A
	I2AE bit = 1	MTIOC1A, MTIOC2A
Input capture from MTU1.TCNT to MTU1.TGRB	I2BE bit = 0 (Initial value)	MTIOC1B
	I2BE bit = 1	MTIOC1B, MTIOC2B
Input capture from MTU2.TCNT to MTU2.TGRA	I1AE bit = 0 (Initial value)	MTIOC2A
	I1AE bit = 1	MTIOC2A, MTIOC1A
Input capture from MTU2.TCNT to MTU2.TGRB	I1BE bit = 0 (Initial value)	MTIOC2B
	I1BE bit = 1	MTIOC2B, MTIOC1B

(1) Example of Cascaded Operation Setting Procedure

Figure 16.21 shows an example of the cascaded operation setting procedure.

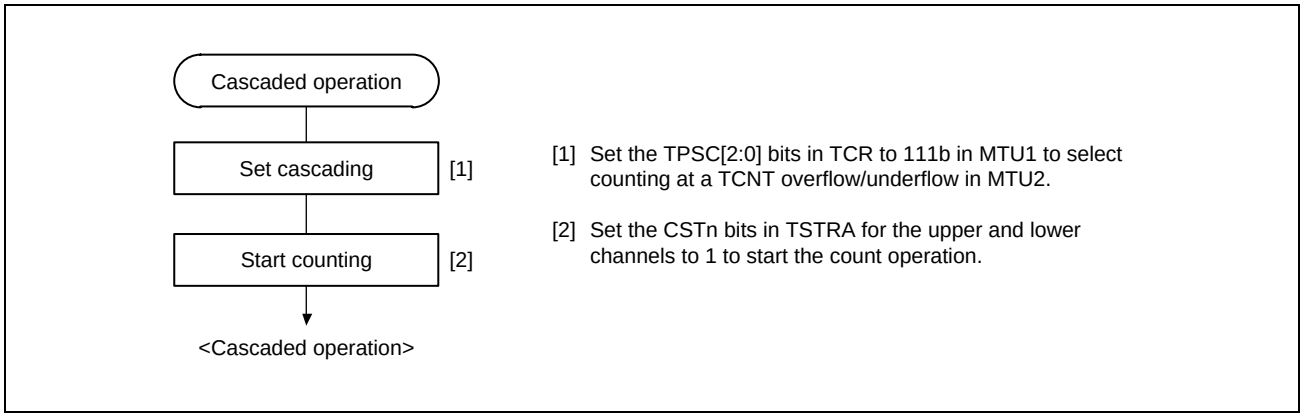


Figure 16.21 Cascaded Operation Setting Procedure

(2) Cascaded Operation Example (a)

Figure 16.22 shows the operation when MTU1.TCNT is set for counting at MTU2.TCNT overflow/underflow and MTU2 is set for phase counting mode 1 while MTU1.TCNT and MTU2.TCNT are cascaded.

MTU1.TCNT is incremented by MTU2.TCNT overflow and decremented by MTU2.TCNT underflow.

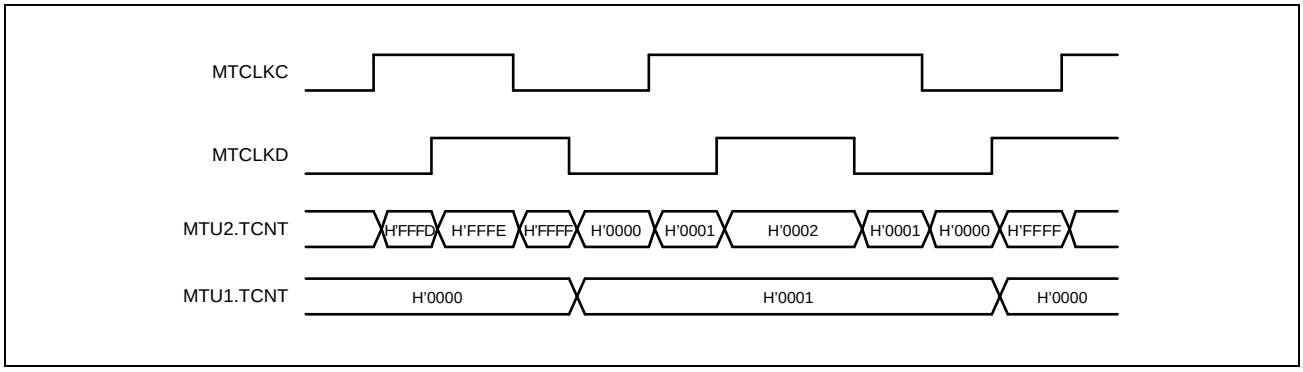


Figure 16.22 Cascaded Operation Example (a)

(3) Cascaded Operation Example (b)

Figure 16.23 illustrates the operation when MTU1.TCNT and MTU2.TCNT have been cascaded and the I2AE bit in TICCR has been set to 1 to include the MTIOC2A pin in the MTU1.TGRA input capture conditions. In this example, the MTU1.TIOR.IOA[3:0] bits have selected the MTIOC1A rising edge for the input capture timing while the MTU2.TIOR.IOA[3:0] bits have selected the MTIOC2A rising edge for the input capture timing.

Under these conditions, the rising edge of both MTIOC1A and MTIOC2A is used for the MTU1.TGRA input capture condition. For the MTU2.TGRA input capture condition, the MTIOC2A rising edge is used.

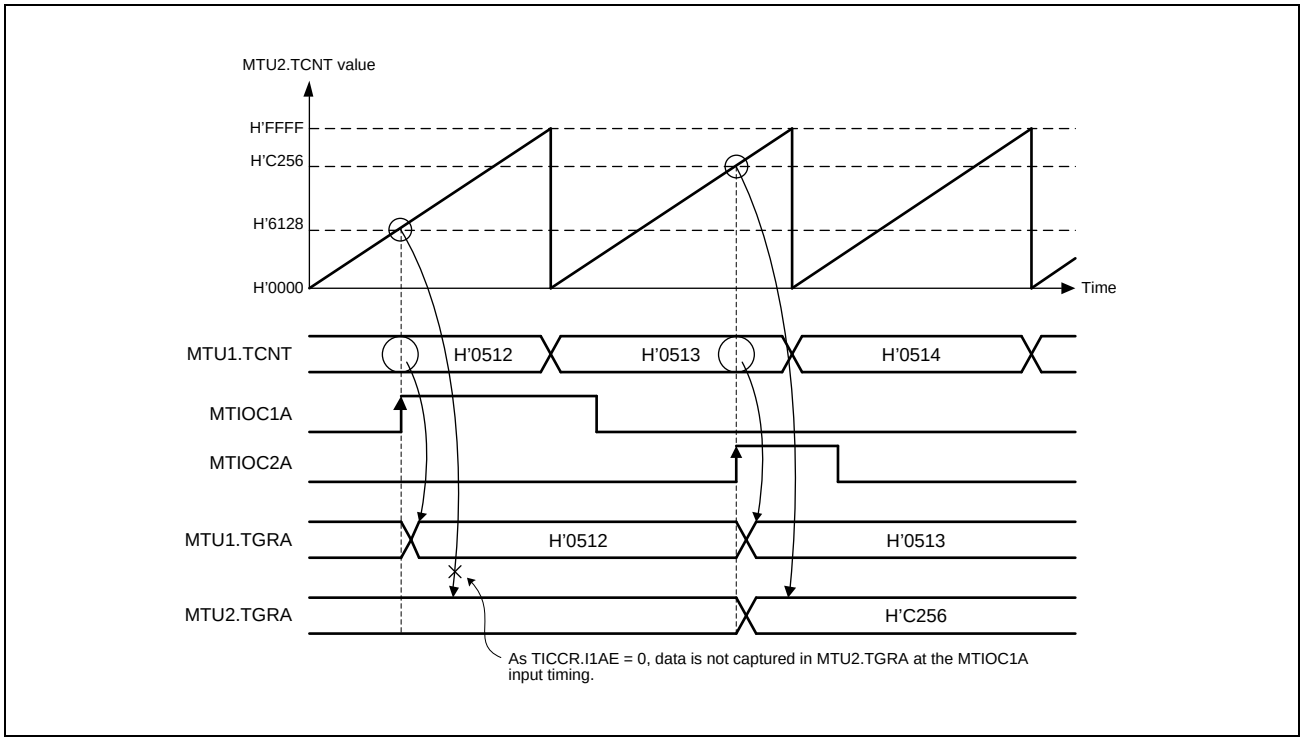
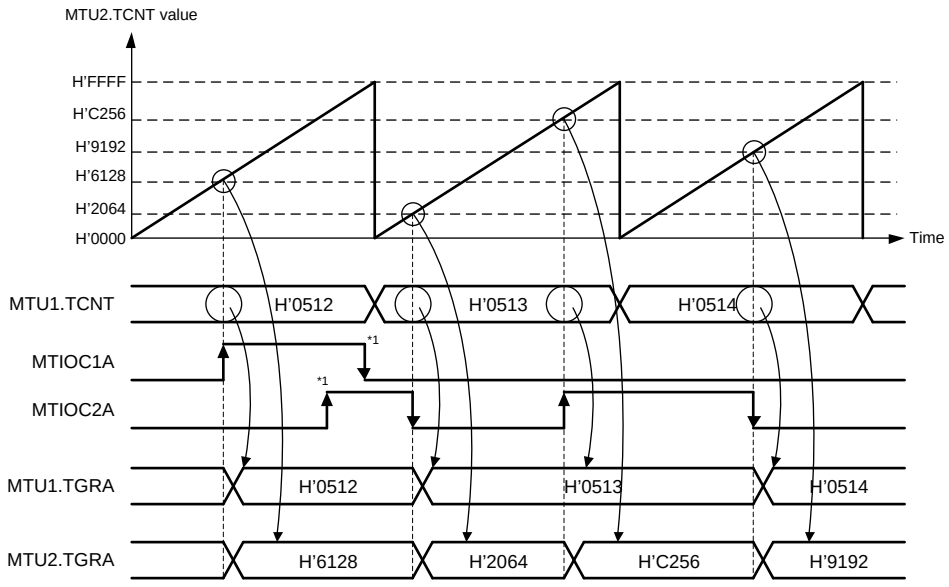


Figure 16.23 Cascaded Operation Example (b)

(4) Cascaded Operation Example (c)

Figure 16.24 illustrates the operation when MTU1.TCNT and MTU2.TCNT have been cascaded and the TICCR.I2AE and I1AE bits have been set to 1 to include the MTIOC2A and MTIOC1A pins in the MTU1.TGRA and MTU2.TGRA input capture conditions, respectively. In this example, the IOA[3:0] bits in both MTU1.TIOR and MTU2.TIOR have selected both the rising and falling edges for the input capture timing. Under these conditions, the ORed result of MTIOC1A and MTIOC2A input is used for the MTU1.TGRA and MTU2.TGRA input capture conditions.



Note 1. When either of the input signals is driven high, the selected edge of the other cannot act as the input capture condition.

Figure 16.24 Cascaded Operation Example (c)

(5) Cascaded Operation Example (d)

Figure 16.25 illustrates the operation when MTU1.TCNT and MTU2.TCNT have been cascaded and the TICCR.I2AE bit has been set to 1 to include the MTIOC2A pin in the MTU1.TGRA input capture conditions. In this example, the IOA[3:0] bits in MTU1.TIOR have selected occurrence of MTU0.TGRA compare match or input capture for the input capture timing while the IOA[3:0] bits in MTU2.TIOR have selected the MTIOC2A rising edge for the input capture timing.

Under these conditions, as MTU1.TIOR has selected occurrence of MTU0.TGRA compare match or input capture for the input capture timing, the MTIOC2A edge is not used for MTU1.TGRA input capture condition although the I2AE bit in TICCR has been set to 1.

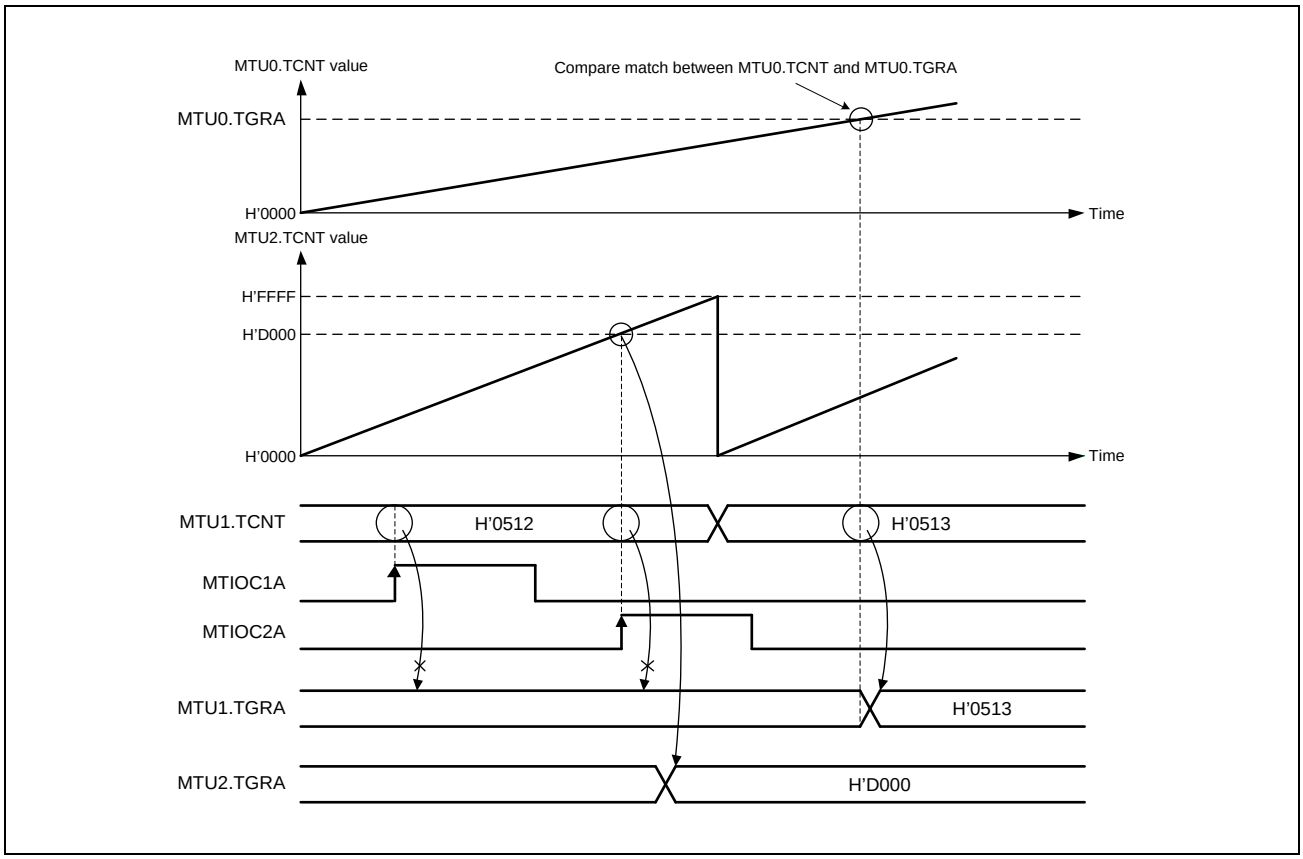


Figure 16.25 Cascaded Operation Example (d)

16.3.5 PWM Modes

PWM modes are provided to output PWM waveforms from the external pins. The output level can be selected as low, high, or toggle output in response to a compare match of each TGR.

PWM waveforms in the range of 0% to 100% duty cycle can be output according to the TGR settings.

By designating TGR compare match as the counter clearing source, the PWM cycle can be specified in that register. Every channel except MTU5 and MTU8 can be set to PWM mode independently. Synchronous operation is also possible between the channels placed in PWM mode or between a channel in PWM mode and another channel in a different mode.

There are two PWM modes as described below.

PWM Mode 1

PWM waveforms are output from the MTIOCnA and MTIOCnC pins by pairing TGRA with TGRB and TGRC with TGRD. The levels specified by the TIOR.IOA[3:0] and IOC[3:0] bits are output from the MTIOCnA and MTIOCnC pins at compare matches A and C, and the level specified by the TIOR.IOB[3:0] and IOD[3:0] bits are output at compare matches B and D (n = 0 to 4, 6, 7). The initial output value is set in TGRA or TGRC. If the values set in paired TGRs are identical, the output value does not change even when a compare match occurs.

In PWM mode 1, PWM waveforms in up to 12 phases can be output.

PWM Mode 2

PWM waveform output is generated using one TGR as the cycle register and the others as duty registers. The level specified in TIOR is output at compare matches. Upon counter clearing by a cycle register compare match, the initial value set in TIOR is output from each pin. If the values set in the cycle and duty registers are identical, the output value does not change even when a compare match occurs.

In PWM mode 2, up to eight phases of PWM waveforms can be output when synchronous clearing is used as synchronous operation in the channels that cannot be placed in PWM mode 2.

The correspondence between PWM output pins and registers is shown in **Table 16.65**.

Table 16.65 PWM Output Registers and Output Pins

Channel	Register	Output Pins	
		PWM Mode 1	PWM Mode 2
MTU0	TGRA	MTIOC0A	MTIOC0A
	TGRB		MTIOC0B
	TGRC	MTIOC0C	MTIOC0C
	TGRD		MTIOC0D
MTU1	TGRA	MTIOC1A	MTIOC1A
	TGRB		MTIOC1B
MTU2	TGRA	MTIOC2A	MTIOC2A
	TGRB		MTIOC2B
MTU3	TGRA	MTIOC3A	Setting prohibited
	TGRB		
	TGRC	MTIOC3C	
	TGRD		
MTU4	TGRA	MTIOC4A	
	TGRB		
	TGRC	MTIOC4C	
	TGRD		
MTU6	TGRA	MTIOC6A	
	TGRB		
	TGRC	MTIOC6C	
	TGRD		
MTU7	TGRA	MTIOC7A	
	TGRB		
	TGRC	MTIOC7C	
	TGRD		

Note: In PWM mode 2, PWM waveform output is not possible for the TGR register in which the PWM cycle is set.

(1) Example of PWM Mode Setting Procedure

Figure 16.26 shows an example of the PWM mode setting procedure.

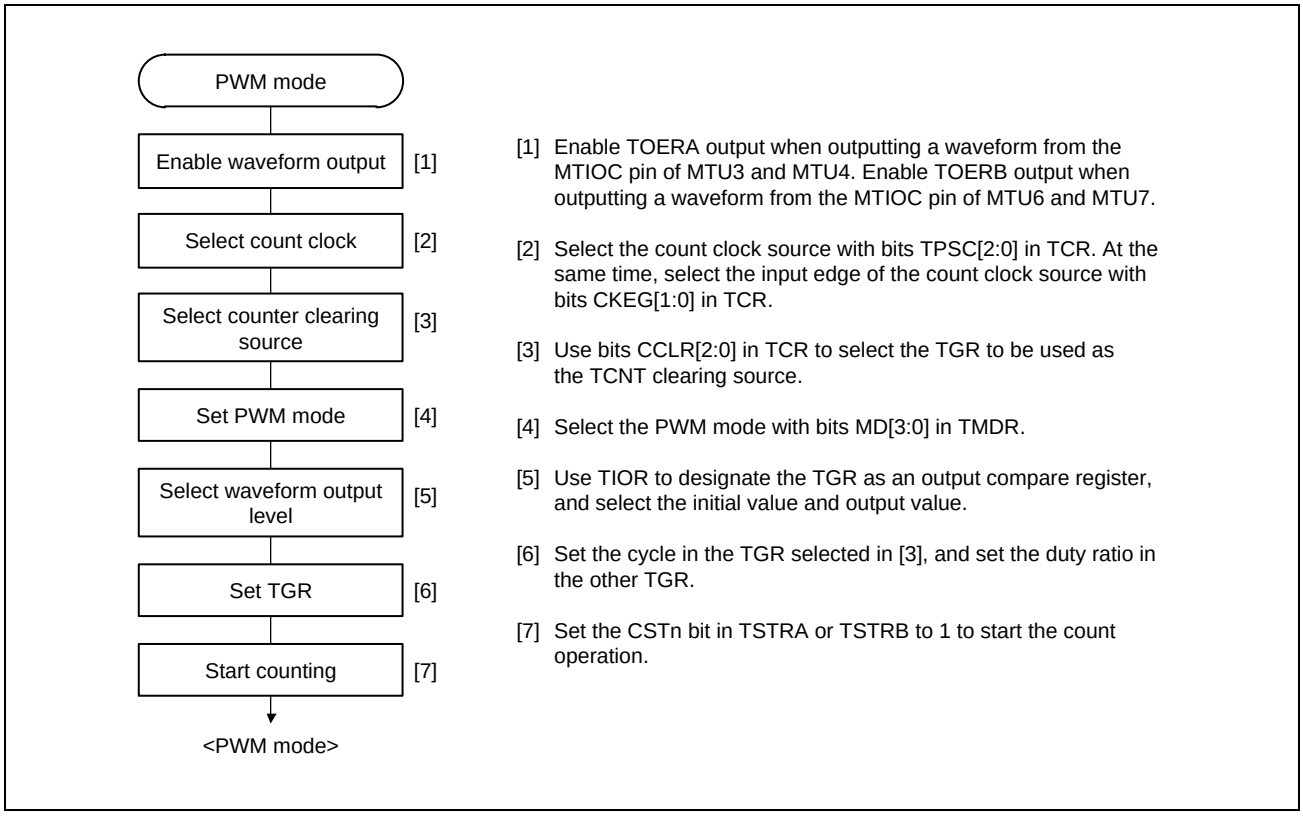


Figure 16.26 Example of PWM Mode Setting Procedure

(2) Examples of PWM Mode Operation

Figure 16.27 shows an example of operation in PWM mode 1.

In this example, TGRA compare match is set as the TCNT clearing source, 0 is set as the initial output value and output value for TGRA, and 1 is set as the output value for TGRB.

In this case, the value set in TGRA is used as the cycle, and the value set in TGRB is used as the duty ratio.

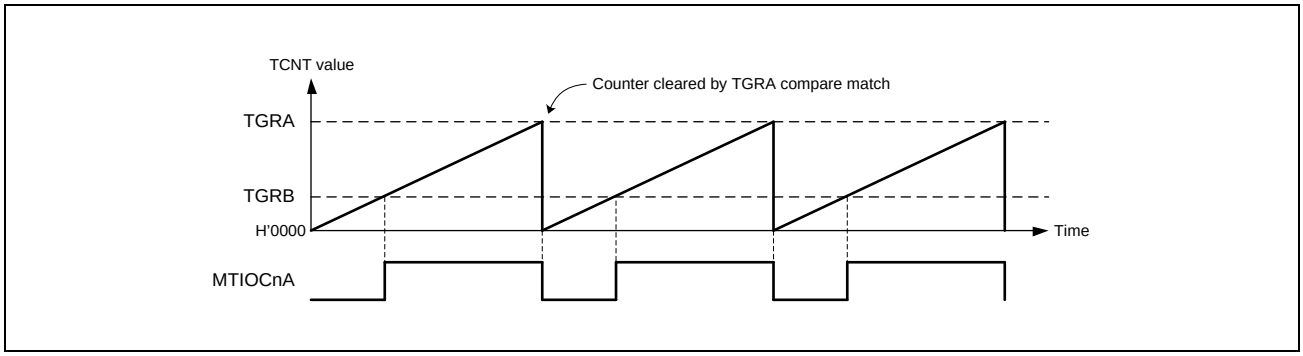


Figure 16.27 Example of PWM Mode 1 Operation (n = 0 to 4, 6, 7)

Figure 16.28 shows an example of operation in PWM mode 2.

In this example, synchronous operation is designated for MTU0 and MTU1, MTU1.TGRB compare match is set as the TCNT clearing source, and low is set as the initial output value and high as the output value for the other TGR registers (MTU0.TGRA to MTU0.TGRD and MTU1.TGRA), outputting 5-phase PWM waveforms.

In this case, the value set in MTU1.TGRB is used as the cycle, and the values set in the other TGRs are used as the duty ratio.

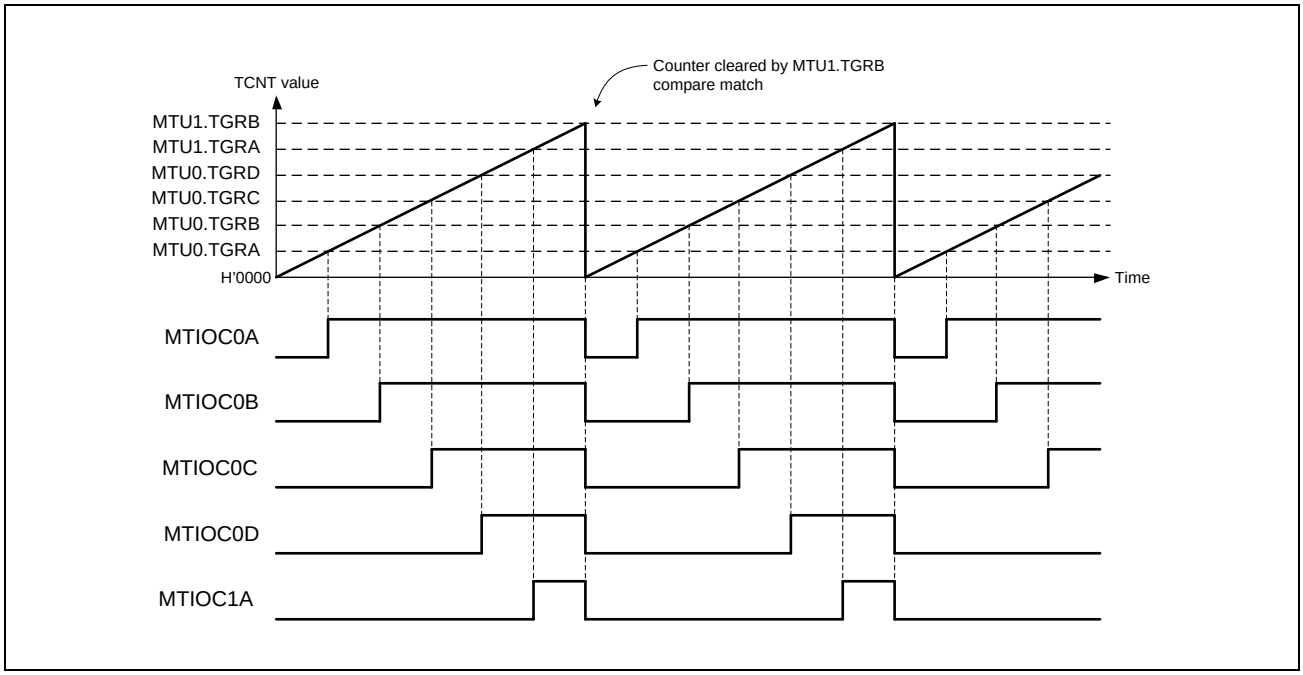


Figure 16.28 Example of PWM Mode 2 Operation

Figure 16.29 shows examples of PWM waveform output with 0% duty and 100% duty in PWM mode 1. In these examples, TGRA compare match is selected as the TCNT clearing source, the initial output value and the output value for TGRA are set to the low level, and the output value for TGRB is set to the high level.

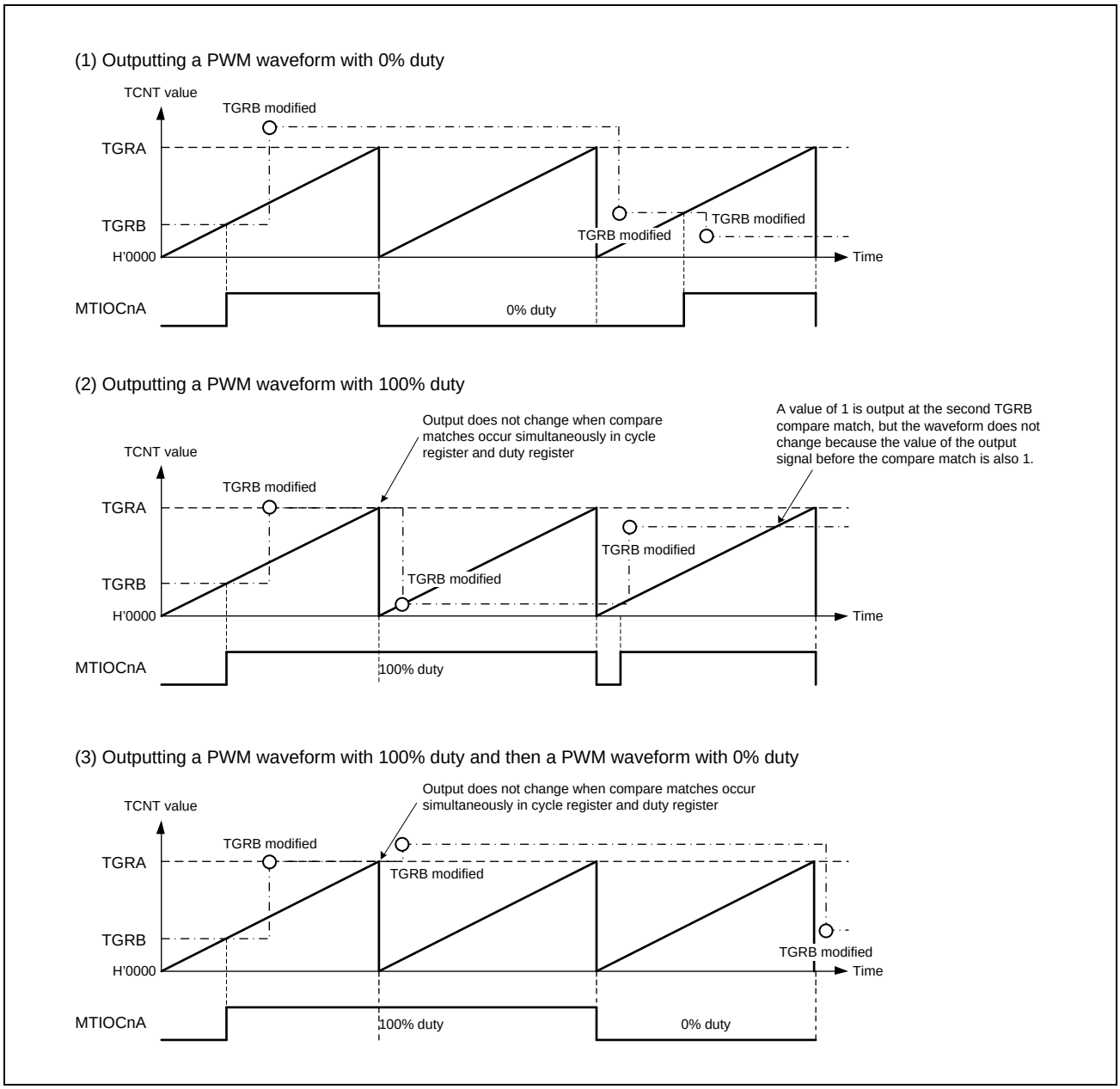


Figure 16.29 Examples of PWM Mode Operation (PWM Waveform Output with 0% Duty and 100% Duty) (n = 0 to 4, 6, 7)

16.3.6 Phase Counting Mode

There are two phase counting modes: 16-bit phase counting mode in which MTU1 and MTU2 operate independently, and cascade connection 32-bit phase counting mode in which MTU1 and MTU2 are cascaded.

In phase counting mode, the phase difference between two external input clocks is detected and the corresponding TCNT is incremented or decremented.

Two external clock input pins for each phase counting mode are not affected by the settings of TCR.TPSC[2:0], TCR2.TPSC2[2:0], and CKEG[1:0]. The two external clock input pins used in 16-bit phase counting mode and cascade connection 32-bit phase counting mode of MTU2 can be selected by MTU1.TMDR3.PHCKSEL. In a phase counting mode other than 16-bit phase counting mode and cascade connection 32-bit phase counting mode of MTU2, MTCLKA and MTCLKB are selected for A-phase and B-phase, respectively. In phase counting mode, the external clock pins MTCLKA, MTCLKB, MTCLKC, and MTCLKD are used for two-phase encoder pulse input.

Table 16.66 lists the external clock input pins to be connected in each phase counting mode.

Table 16.66 Clock Input Pins in Phase Counting Mode

		External Clock Input Pins	
Phase Counting Mode	TMDR3.PHCKSEL bit	A-Phase	B-Phase
MTU1 16-bit phase counting mode	x (Don't care)	MTCLKA	MTCLKB
MTU2 16-bit phase counting mode	0	MTCLKA	MTCLKB
	1 (initial value)	MTCLKC	MTCLKD
Cascade connection 32-bit phase counting mode	0	MTCLKA	MTCLKB
	1 (initial value)	MTCLKC	MTCLKD

16.3.6.1 16-Bit Phase Counting Mode

When the MTU1.TMDR3.LWA is 0, 16-bit phase counting mode can be set individually for MTU1 and MTU2.

In 16-bit phase counting mode, the phase difference between two external input clocks is detected and the 16-bit counter TCNT of the corresponding channel is incremented or decremented.

When 16-bit phase counting mode is specified, an external clock is selected as the count clock and TCNT operates as an up-counter/down-counter regardless of the setting of TCR.TPSC[2:0], TCR2.TPSC2[2:0], and CKEG[1:0]. However, the functions of TCR.CCLR[1:0],TIOR, TIER, and TGR are enabled, and input capture/compare match and interrupt functions can be used.

These external input pins can be used for two-phase encoder pulse input.

If an overflow occurs during TCNT up-counting, a TCIV interrupt is generated when the corresponding TCIEV bit in the TIER register is set to 1.

If an underflow occurs during TCNT down-counting, a TCIU interrupt is generated when the corresponding TCIEU bit in the TIER register is set to 1.

The TCFD bit in TSR is the count direction flag. Read the TCFD flag to check whether TCNT is counting up or down.

(1) Example of 16-Bit Phase Counting Mode Setting Procedure

Figure 16.30 shows an example of the phase counting mode setting procedure.

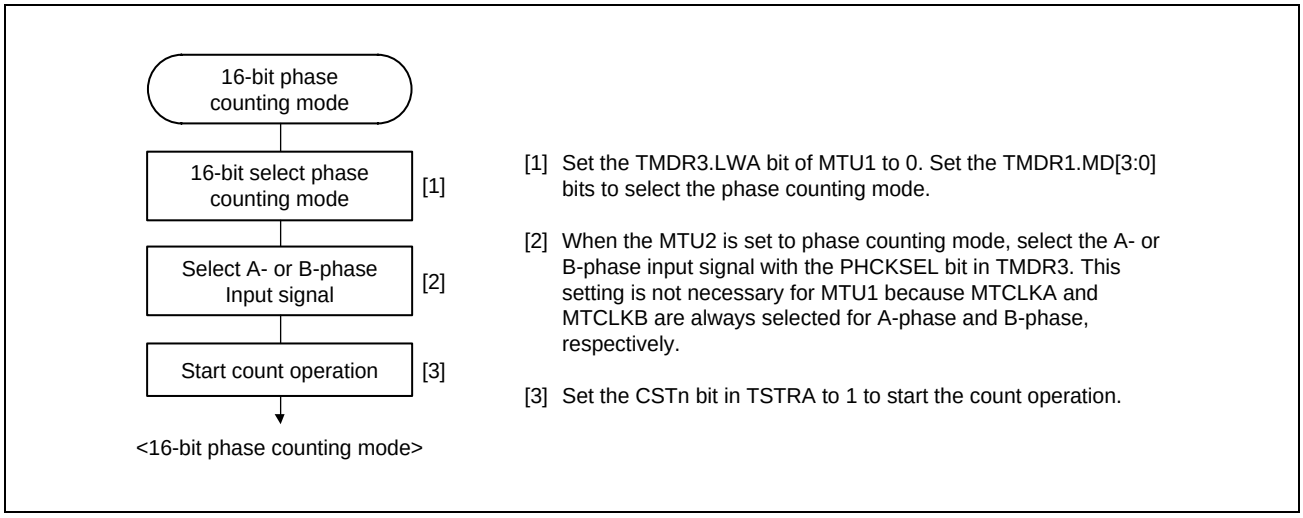


Figure 16.30 Example of 16-Bit Phase Counting Mode Setting Procedure

(2) Examples of 16-Bit Phase Counting Mode Operation

In phase counting mode, TCNT is incremented or decremented according to the phase difference between two external clocks. There are five modes according to the count conditions. Each mode operates under the condition PHCKSEL = 1, which means the phase clock for MTU1 is input from MTCLKA or MTCLKB and that for MTU2 is input from MTCLKC or MTCLKD.

(a) Phase Counting Mode 1

Figure 16.31 shows an example of operation in phase counting mode 1, and **Table 16.67** summarizes the TCNT up- counting and down-counting conditions.

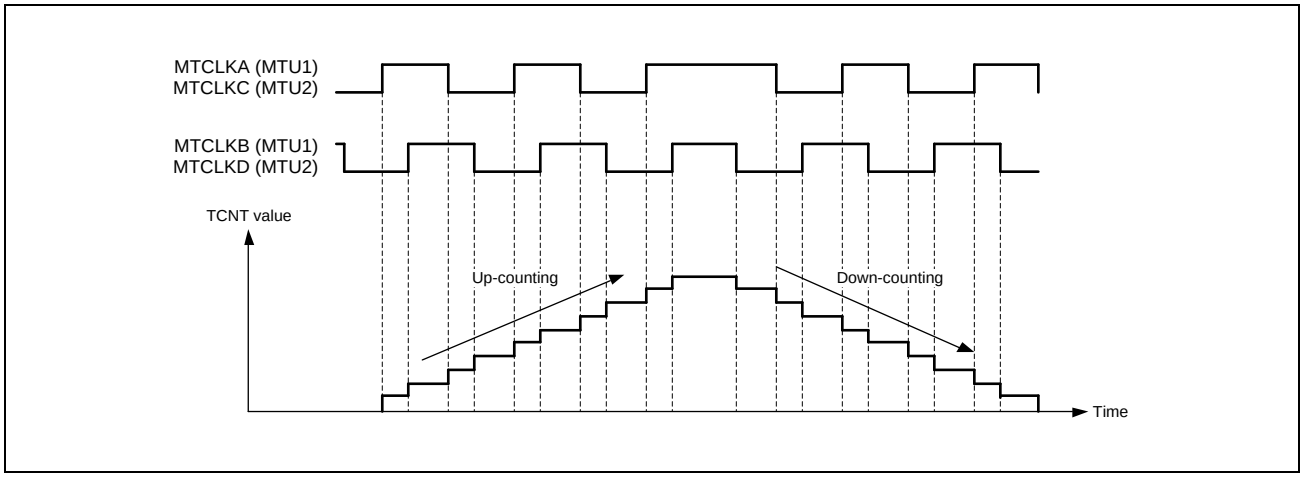


Figure 16.31 Example of Operation in Phase Counting Mode 1

Table 16.67 Up-Counting and Down-Counting Conditions in Phase Counting Mode 1

MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
High	↑	Up-counting
Low	↓	
↑	Low	
↓	High	
High	↓	Down-counting
Low	↑	
↑	High	
↓	Low	

Remarks: ↑: Rising edge
↓: Falling edge

(b) Phase Counting Mode 2

Figure 16.32 to **Figure 16.34** show the examples of operation in phase counting mode 2 and **Table 16.68** summarizes the TCNT up-counting and down-counting conditions.

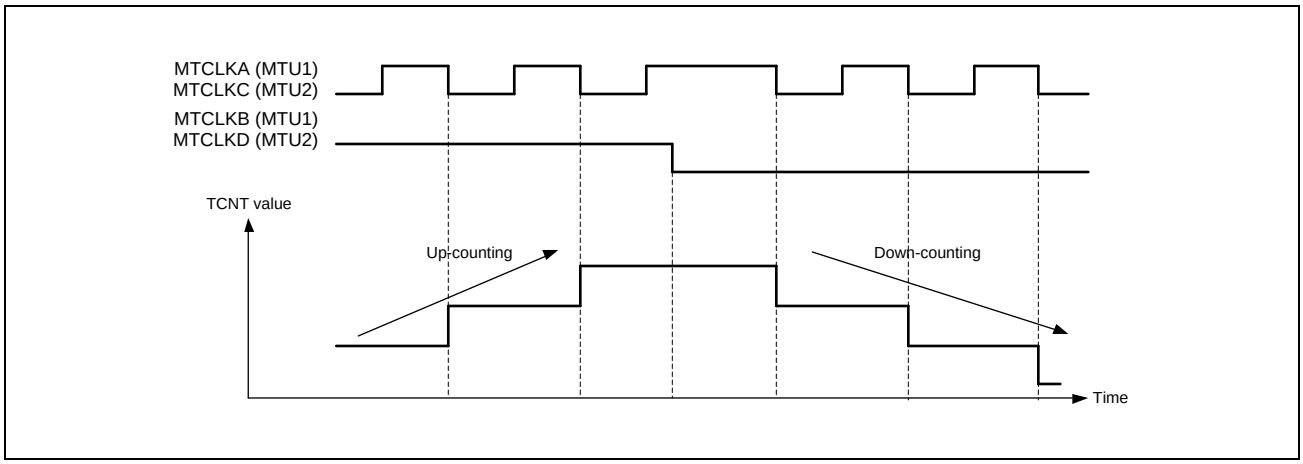


Figure 16.32 Example of Operation in Phase Counting Mode 2 (When MTUn.TCR2.PCB[1:0] is 00b (n = 1, 2))

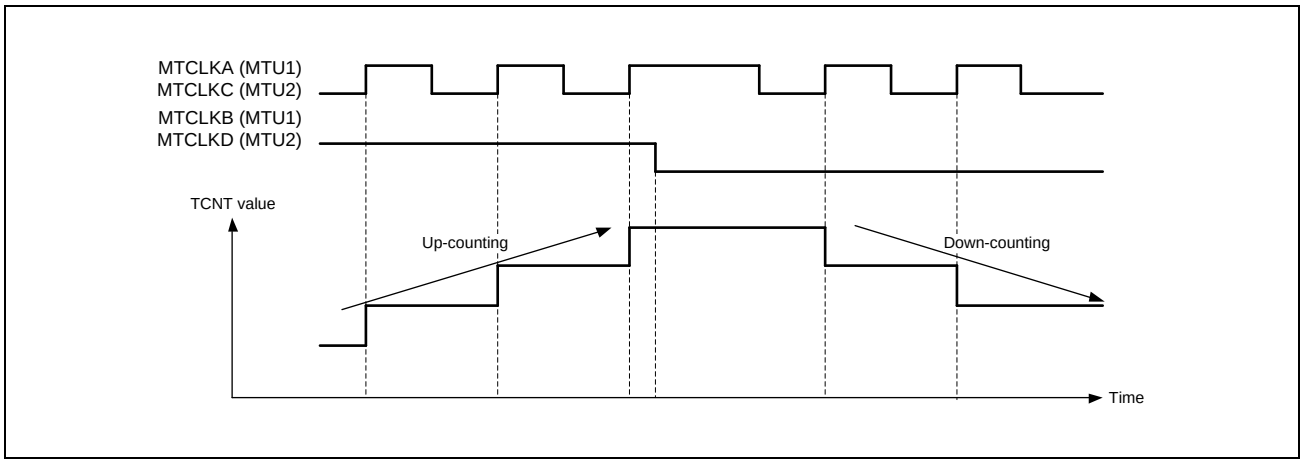


Figure 16.33 Example of Operation in Phase Counting Mode 2 (When MTUn.TCR2.PCB[1:0] is 01b (n = 1, 2))

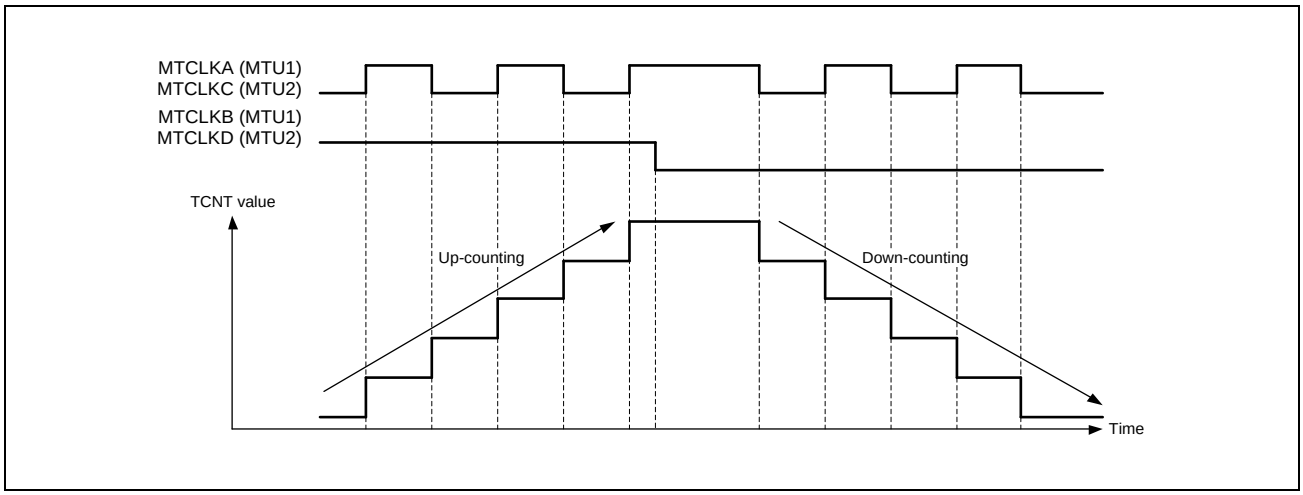


Figure 16.34 Example of Operation in Phase Counting Mode 2 (When MTUn.TCR2.PCB[1:0] is 1xb (n = 1, 2))

Table 16.68 Up-Counting and Down-Counting Conditions in Phase Counting Mode 2

PCB[1:0]	MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
00b	High	↑	Not counted (Don't care)
	Low	↓	
	↑	Low	
	↓	High	Up-counting
	High	↓	
	Low	↑	
	↑	High	
	↓	Low	Down-counting
01b	High	↑	Not counted (Don't care)
	Low	↓	
	↑	Low	Down-counting
	↓	High	Not counted (Don't care)
	High	↓	
	Low	↑	
	↑	High	Up-counting
	↓	Low	Not counted (Don't care)
1xb	High	↑	Not counted (Don't care)
	Low	↓	
	↑	Low	Down-counting
	↓	High	Up-counting
	High	↓	Not counted (Don't care)
	Low	↑	
	↑	High	Up-counting
	↓	Low	Down-counting

Remarks: ↑ : Rising edge
↓ : Falling edge

(c) Phase Counting Mode 3

Figure 16.35 to **Figure 16.37** show the examples of operation in phase counting mode 3 and **Table 16.69** summarizes the TCNT up-counting and down-counting conditions.

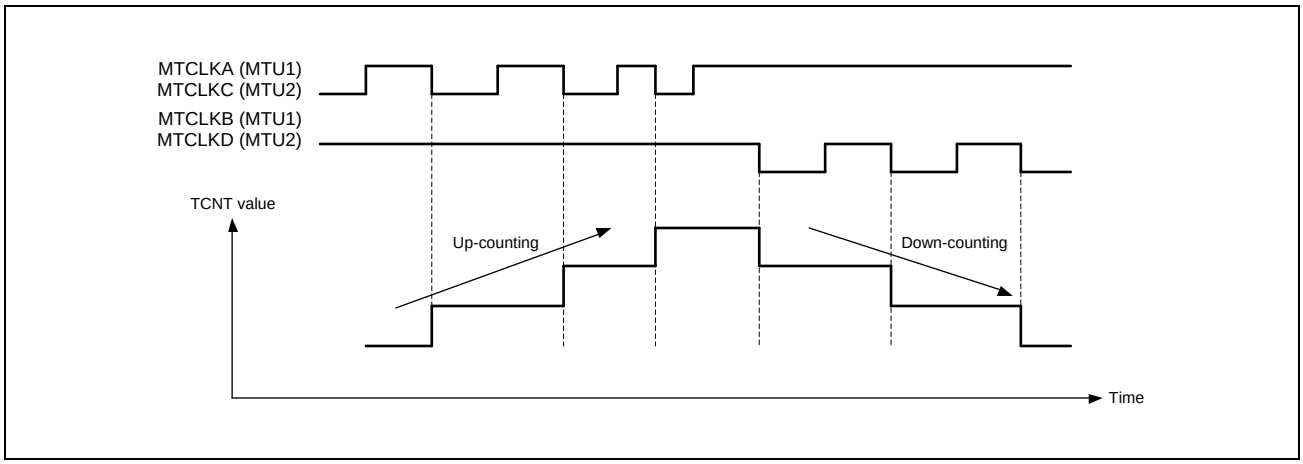


Figure 16.35 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] is 00b (n = 1, 2))

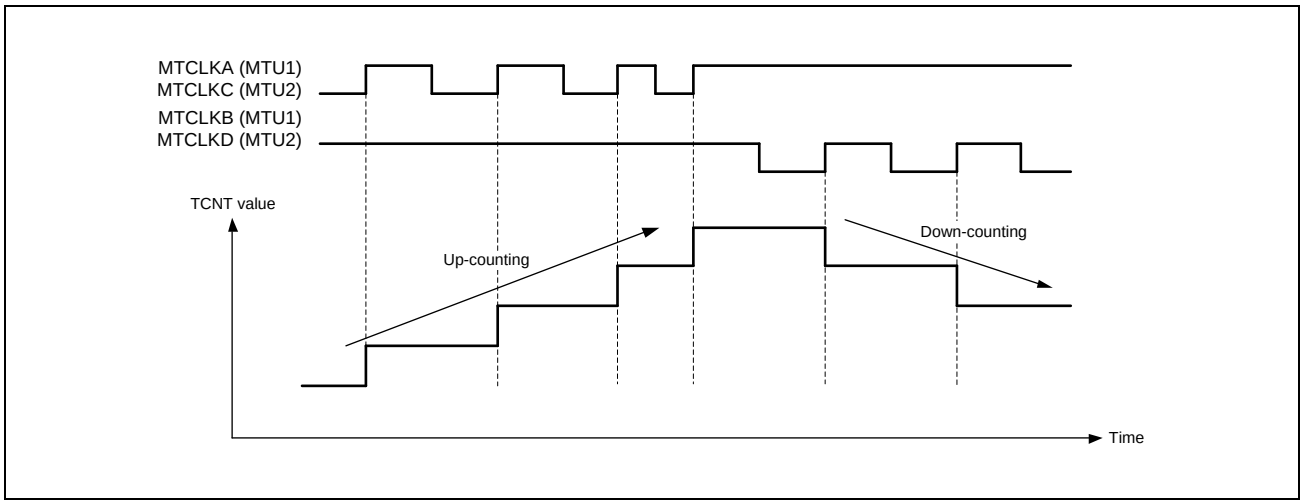


Figure 16.36 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] is 01b (n = 1, 2))

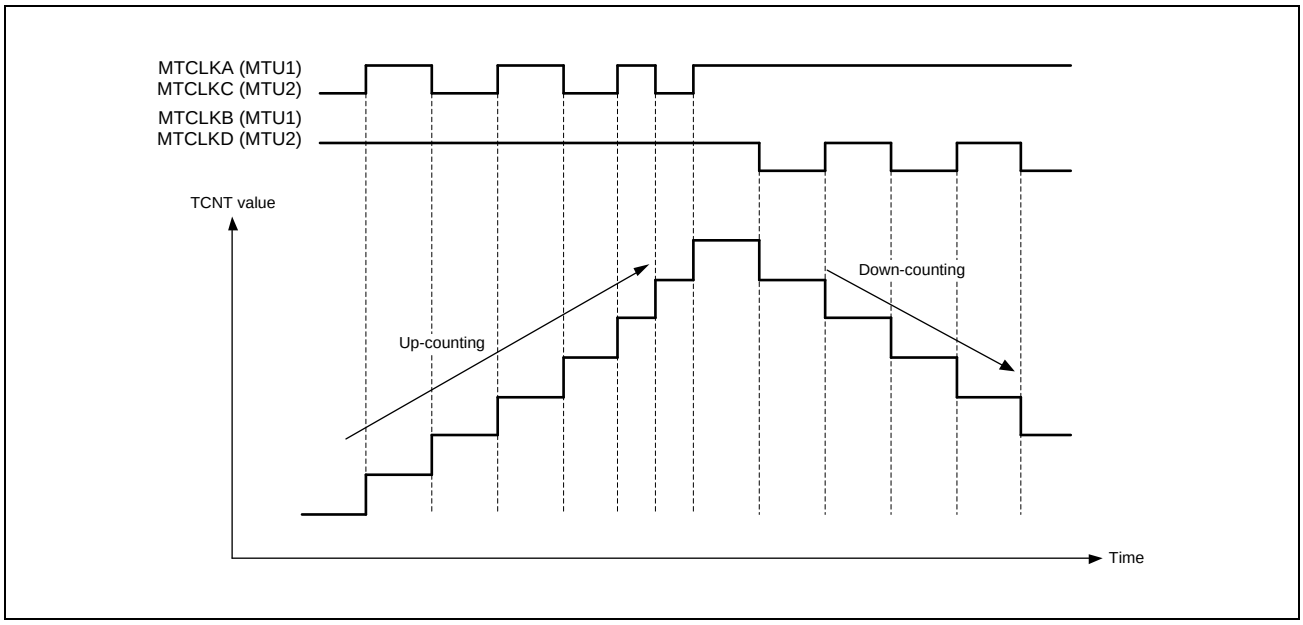


Figure 16.37 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] is 1xb (n = 1, 2))

Table 16.69 Up-Counting and Down-Counting Conditions in Phase Counting Mode 3

PCB[1:0]	MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
00b	High	↑	Not counted (Don't care)
	Low	↓	
	↑	Low	
	↓	High	Up-counting
	High	↓	Down-counting
	Low	↑	Not counted (Don't care)
	↑	High	
	↓	Low	
01b	High	↑	Down-counting
	Low	↓	Not counted (Don't care)
	↑	Low	
	↓	High	
	High	↓	
	Low	↑	
	↑	High	Up-counting
	↓	Low	Not counted (Don't care)
1xb	High	↑	Down-counting
	Low	↓	Not counted (Don't care)
	↑	Low	
	↓	High	Up-counting
	High	↓	Down-counting
	Low	↑	Not counted (Don't care)
	↑	High	Up-counting
	↓	Low	Not counted (Don't care)

Remarks: ↑ : Rising edge
↓ : Falling edge

(d) Phase Counting Mode 4

Figure 16.38 shows an example of operation in phase counting mode 4, and **Table 16.70** summarizes the TCNT up- counting and down-counting conditions.

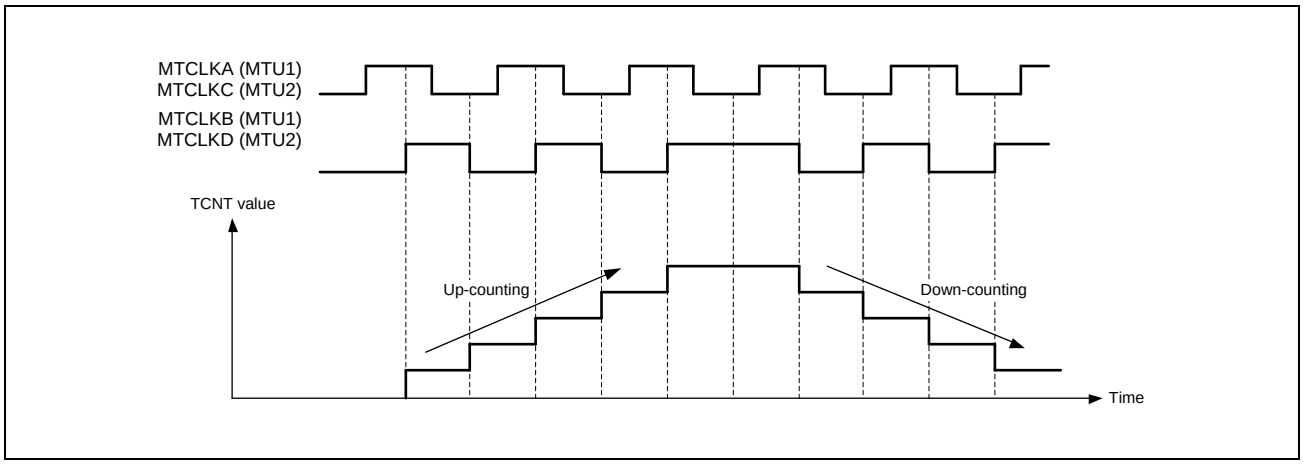


Figure 16.38 Example of Operation in Phase Counting Mode 4

Table 16.70 Up-Counting and Down-Counting Conditions in Phase Counting Mode 4

MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
High	↑	Up-counting
Low	↓	
↑	Low	Not counted (Don’t care)
↓	High	
High	↓	Down-counting
Low	↑	
↑	High	Not counted (Don’t care)
↓	Low	

Remarks: ↑ : Rising edge
↓ : Falling edge

(e) Phase Counting Mode 5

Figure 16.39 and **Figure 16.40** show the examples of operation in phase counting mode 5 and **Table 16.71** summarizes the TCNT up-counting and down-counting conditions.

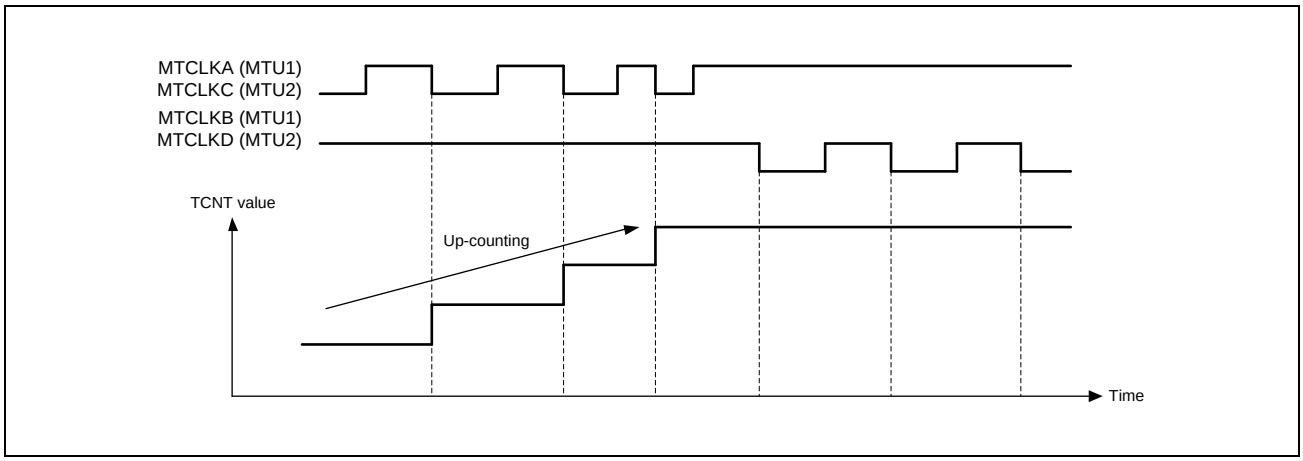


Figure 16.39 Example of Operation in Phase Counting Mode 5 (When MTUn.TCR2.PCB[1:0] = 0xb (n = 1, 2))

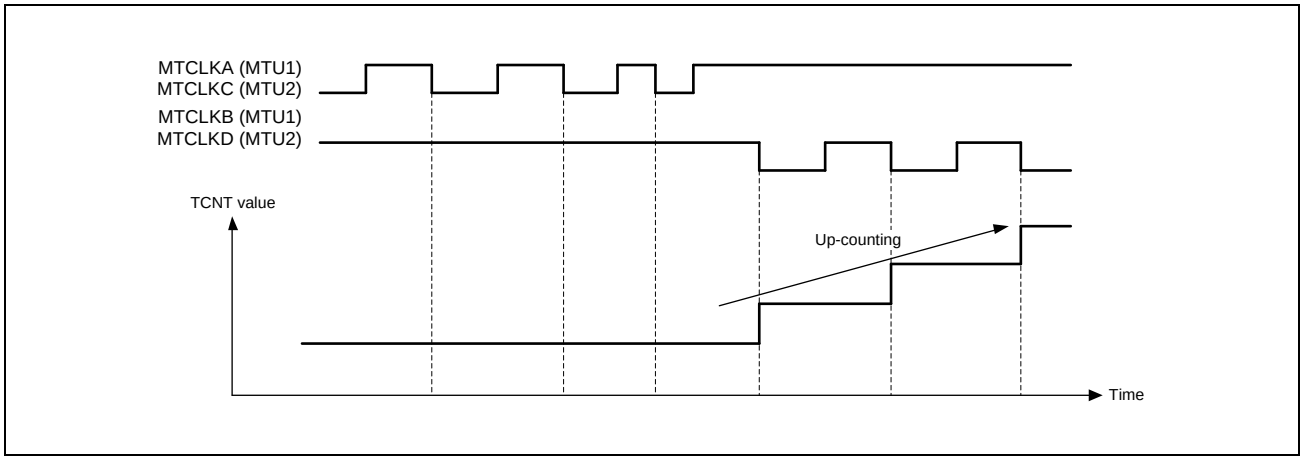


Figure 16.40 Example of Operation in Phase Counting Mode 5 (When MTUn.TCR2.PCB[1:0] = 1xb (n = 1, 2))

Table 16.71 Up-Counting and Down-Counting Conditions in Phase Counting Mode 5

PCB[1:0]	MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
0xb	High	↑	Not counted (Don't care)
	Low	↓	
	↑	Low	
	↓	High	Up-counting
	High	↓	Not counted (Don't care)
	Low	↑	
	↑	High	
	↓	Low	Up-counting
1xb	High	↑	Not counted (Don't care)
	Low	↓	Up-counting
	↑	Low	Not counted (Don't care)
	↓	High	
	High	↓	Up-counting
	Low	↑	Not counted (Don't care)
	↑	High	
	↓	Low	

Remarks: ↑: Rising edge
↓: Falling edge

(3) 16-Bit Phase Counting Mode Application Example

Figure 16.41 shows an example in which MTU1 is in phase counting mode, and MTU1 is coupled with MTU0 to input 2-phase encoder pulses of a servo motor in order to detect position or speed.

MTU1 is set to phase counting mode 1, and the encoder pulse A-phase and B-phase are input to MTCLKA and MTCLKB.

In MTU0, MTU0.TGRC compare match is specified as the TCNT clearing source and MTU0.TGRA and MTU0.TGRC are used for the compare match function and are set with the speed control cycle and position control cycle.

MTU0.TGRB is used for input capture, with MTU0.TGRB and MTU0.TGRD operating in buffer mode. The MTU1 count clock is designated as the MTU0.TGRB input capture source, and the widths of 2-phase encoder 4-multiplication pulses are detected.

MTU1.TGRA and MTU1.TGRB for MTU1 are designated for the input capture function and MTU0.TGRA and MTU0.TGRC compare matches in MTU0 are selected as the input capture sources to store the up/down-counter values for the control cycles.

This procedure enables the accurate detection of position and speed.

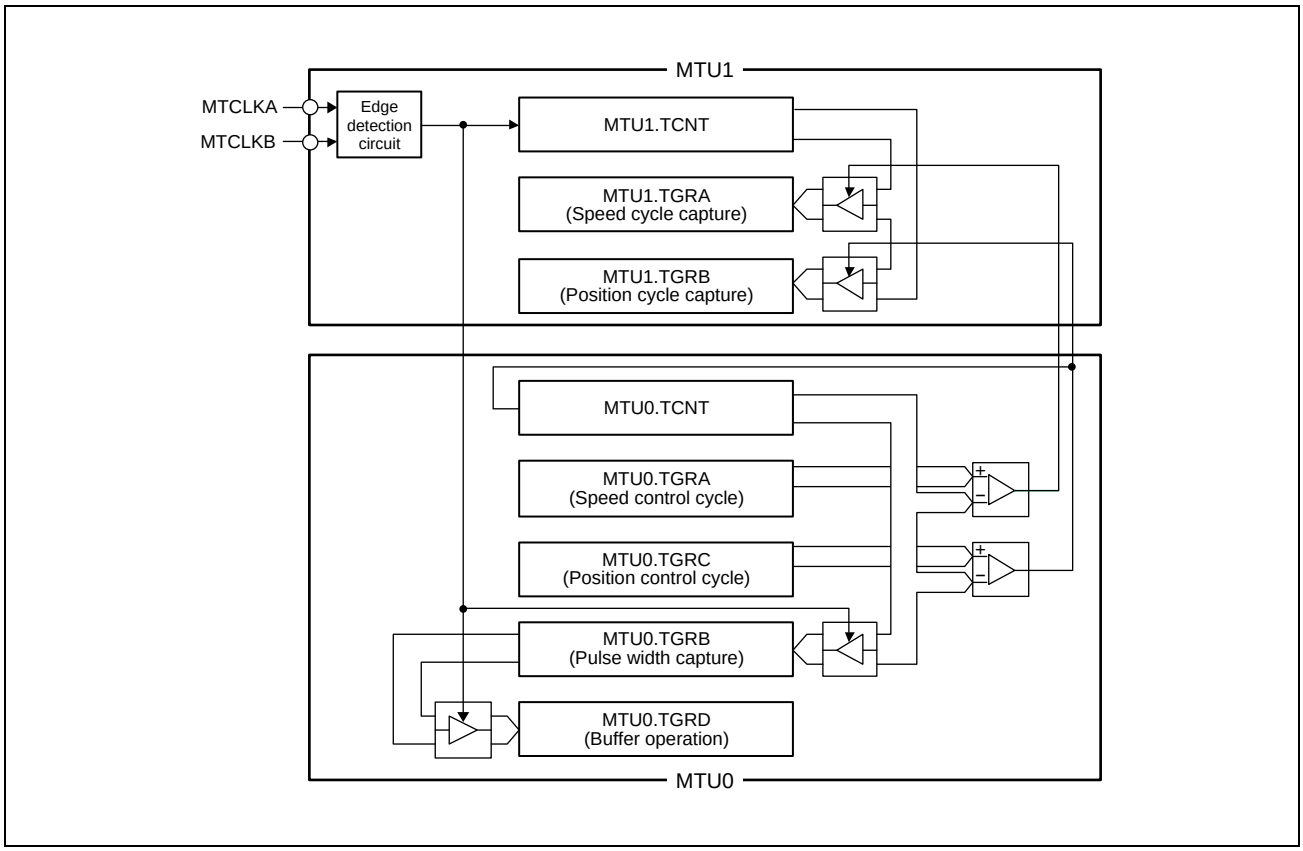


Figure 16.41 16-Bit Phase Counting Mode Application Example

16.3.6.2 Cascade Connection 32-Bit Phase Counting Mode

When MTU1 is set to phase counting mode by setting MTU1.TMDR3.LWA = 1, MTU1 and MTU2 are connected to operate in cascade connection 32-bit phase counting mode as shown in **Figure 16.42**. When this mode is used, the TCR, TCR2, TIOR, TIER, TGR, and TSR registers are controlled by MTU1 and the settings of MTU2 are disabled. See **Figure 16.43** for the procedure for setting cascade connection 32-bit phase counting mode.

In this mode, three-phase (A, B, and Z) signals can be input. As an encoder pulse signal, the external input phase clocks MTCLKA and MTCLKB or MTCLKC and MTCLKD can be selected for A-phase and B-phase, and MTIOC1A can be selected for Z-phase, respectively. See **Table 16.70** for selecting external clock input of A-phase and B-phase. A counter event is generated using an A-phase or B-phase pulse and counted by the 32-bit counter MTU1.TCNTLW.

An input capture can also be generated using a Z-phase signal; thus, an angular velocity can be measured using the value captured in the general register.

Furthermore, MTU8 can be used as a channel for measuring a 1-ms interval, and a compare match signal can be output at a 1-ms interval to the MTU1 and MTU2, which operate in cascade connection 32-bit phase counting mode. That is, a compare match signal of MTU8 is used as a capture signal of MTU1 and MTU2, and the number of A-phase and B-phase pulses for a 1-ms period can be measured.

When MTU0 or MTU5 is specified as the channel for measuring Z-phase signal pulses, this compare match signal of TGRC in MTU8 can be output as a capture signal or a clear signal to MTU0 or MTU5; thus, the Z-phase count can be measured at 1-ms intervals.

In addition, a counter event signal of combined MTU1 and MTU2 can be used as a capture signal of TGRD in MTU8, and measurement can be performed including the intervals of A, B, or both phase pulses. In this case, the TGRD register in MTU8 should be set to buffer operation.

Refer to **Section 16.3.4, Cascaded Operation**, for details on the cascade connection function for connecting MTU1 and MTU2 in a mode other than cascade connection 32-bit phase counting mode.

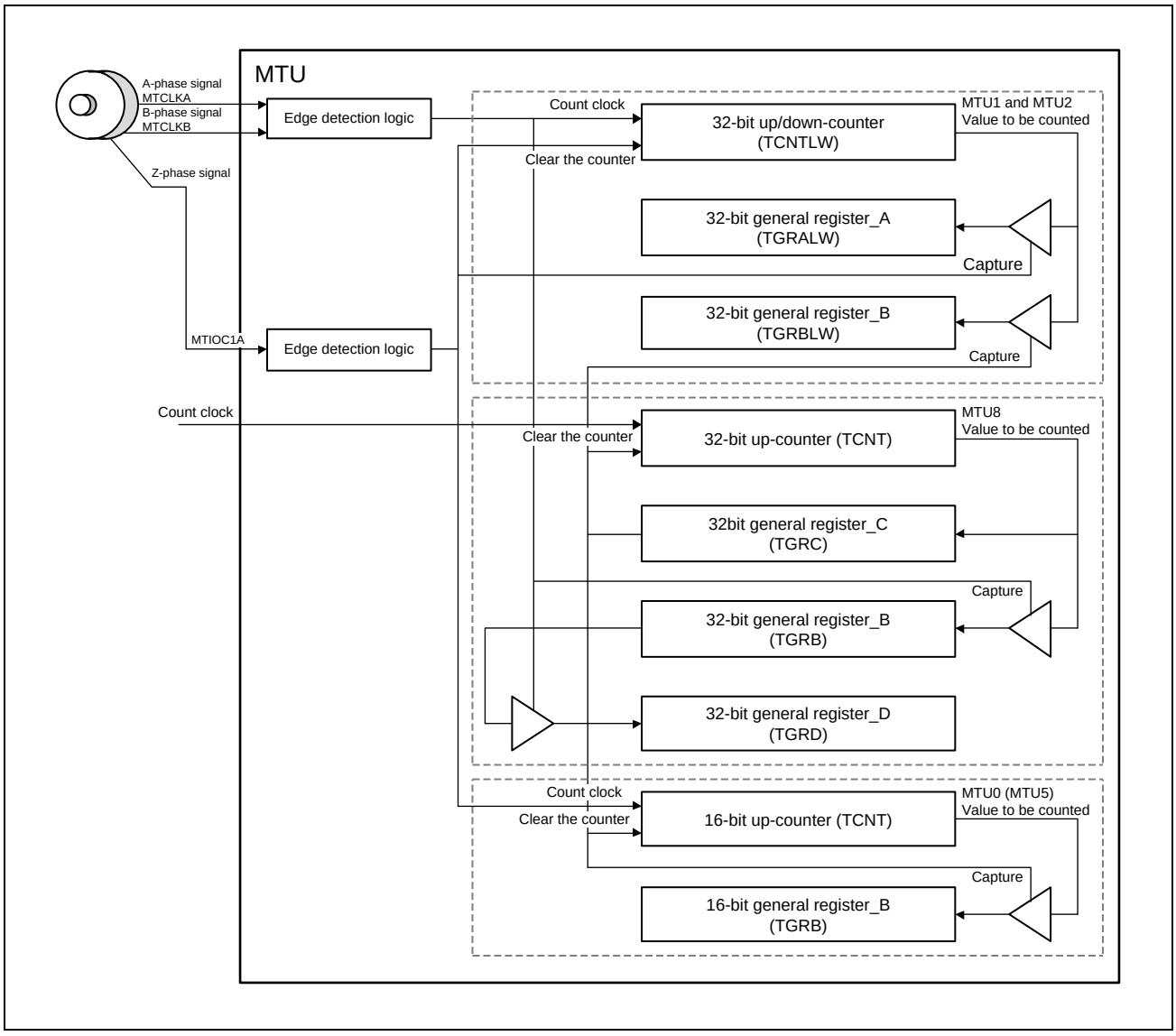
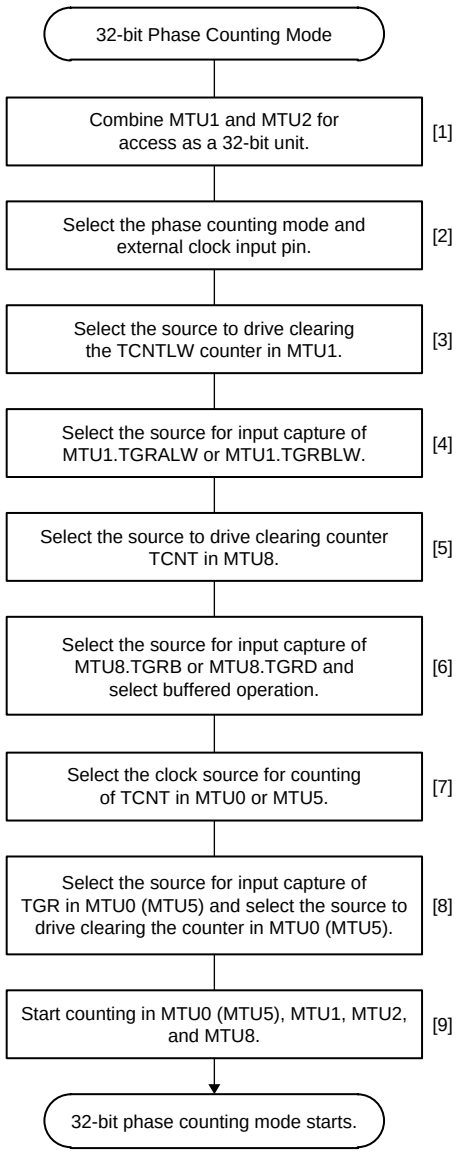


Figure 16.42 Block Diagram for Operation in Cascade Connection 32-Bit Phase Counting Mode

(1) Example of Setting Cascade Connection 32-Bit Phase Counting Mode

Figure 16.43 shows an example of the procedure for setting cascade connection 32-bit phase counting mode.



[1] Set the LWA bit in MTU1.TMDR3 to 1 and combine MTU1 and MTU3 for access as a 32-bit unit.*1

[2] Select phase counting mode with the MD[3:0] bits in MTU1.TMDR1 and select the external clock input pin with the PHCKSEL bits in MTU1.TMDR3.

[3] Set the CCLR[2:0] bits of TCR in MTU1 to 001b to select input capture of TGRALW in MTU1 as the source of the signal to drive clearing the TCNTLW counter in MTU1.

[4] Set the IOA[3:0] bits in MTU1.TIOR to select the effective edge (rising, falling or both) of the signal produced by input capture of TGRALW in MTU1. After that, set the IOB[3:0] bits in MTU1.TIOR to 1110b to select the compare match signal of MTU8.TGRC as the source to drive input capture of MTU1.TGRBLW.

[5] Set the CCLR[2:0] bits in MTU8.TCR to 101b to select the compare match signal of MTU8.TGRC as the source to drive clearing counter TCNT in MTU8.

[6] Select the source for input capture of MTU8.TGRB as the clock for counting of TCNTLW in MTU1 by setting the IOB[3:0] bits in MTU8.TIORH to 1110b. At the same time, select buffered operation for TGRB and TGRD by setting the BFB bit in MTU8.TMDR1 to 1.

[7] Set the TPSC2[2:0] bits in MTU0.TCR2 to 111b to select the MTIOC1A pin as the clock source for counting of TCNT in MTU0.
(When using TCNT in MTU5, set the TPSC2[2:0] bits in MTU5.TCR2 to 111b).

[8] Select the compare match signal of MTU8.TGRC as the source to drive input capture of MTU0.TGRA by setting the IOA[3:0] bits in MTU0.TIORH to 1110b. (When using TCNT in MTU5, set the TIORU, TIORV, or TIORW bit in MTU5 is set to 10100b).
After that, select input capture of MTU0.TGRA as the source to drive clearing counter TCNT in MTU0 by setting the CCLR[2:0] bits in MTU0.TCR to 001b.

[9] Start counting in MTU0, MTU1, MTU2, and MTU8 by setting the CST0, CST1, CST2, and CST8 bits in TSTRA to 1 at the same time.
(When using TCNT in MTU5, set the counter corresponding to TSTR in MTU5 to 1).

Note 1. When the LWA bit in MTU1.TMDR3 is set to 1, the TCR, TCR2, TMDR, and TIOR registers in MTU1 are effective in controlling cascaded operation. In this case, the settings of these registers in MTU2 are ignored.

Figure 16.43 Procedure for Setting Cascade Connection 32-Bit Phase Counting Mode

16.3.7 Reset-Synchronized PWM Mode

In the reset-synchronized PWM mode, three phases of positive and negative PWM waveforms (six phases in total) that share a common wave transition point can be output by combining MTU3 and MTU4 and MTU6 and MTU7.

When set for reset-synchronized PWM mode, the MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4C, MTIOC4B, MTIOC4D, MTIOC6B, MTIOC6D, MTIOC7A, MTIOC7C, MTIOC7B, and MTIOC7D pins function as PWM output pins and timer counters 6 and 12 (MTU3.TCNT and MTU6.TCNT) functions as an up-counter.

Table 16.72 shows the PWM output pins used. **Table 16.73** shows the settings of the registers.

Table 16.72 Output Pins for Reset-Synchronized PWM Mode

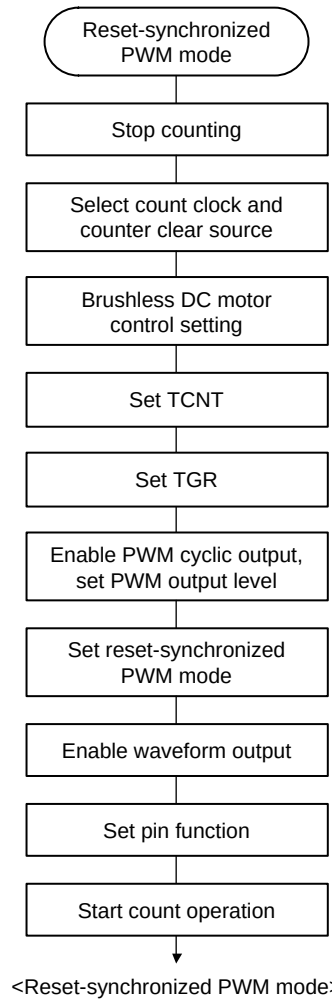
Channel	Output Pin	Description
MTU3	MTIOC3B	PWM output pin 1
	MTIOC3D	PWM output pin 1' (negative-phase waveform of PWM output 1)
MTU4	MTIOC4A	PWM output pin 2
	MTIOC4C	PWM output pin 2' (negative-phase waveform of PWM output 2)
	MTIOC4B	PWM output pin 3
	MTIOC4D	PWM output pin 3' (negative-phase waveform of PWM output 3)
MTU6	MTIOC6B	PWM output pin 4
	MTIOC6D	PWM output pin 4' (negative-phase waveform of PWM output 4)
MTU7	MTIOC7A	PWM output pin 5
	MTIOC7C	PWM output pin 5' (negative-phase waveform of PWM output 5)
	MTIOC7B	PWM output pin 6
	MTIOC7D	PWM output pin 6' (negative-phase waveform of PWM output 6)

Table 16.73 Register Settings for Reset-Synchronized PWM Mode

Register	Setting
MTU3.TCNT	Initial setting (H’0000)
MTU4.TCNT	Initial setting (H’0000)
MTU3.TGRA	Set the count cycle for MTU3.TCNT
MTU3.TGRB	Set the transition point of the PWM waveform to be output from the MTIOC3B and MTIOC3D pins
MTU4.TGRA	Set the transition point of the PWM waveform to be output from the MTIOC4A and MTIOC4C pins
MTU4.TGRB	Set the transition point of the PWM waveform to be output from the MTIOC4B and MTIOC4D pins
MTU6.TCNT	Initial setting (H’0000)
MTU7.TCNT	Initial setting (H’0000)
MTU6.TGRA	Set the count cycle for MTU6.TCNT
MTU6.TGRB	Set the transition point of the PWM waveform to be output from the MTIOC6B and MTIOC6D pins
MTU7.TGRA	Set the transition point of the PWM waveform to be output from the MTIOC7A and MTIOC7C pins
MTU7.TGRB	Set the transition point of the PWM waveform to be output from the MTIOC7B and MTIOC7D pins

(1) Example of Procedure for Setting Reset-Synchronized PWM Mode

Figure 16.44 shows an example of the procedure for specifying the reset-synchronized PWM mode.



[1] Clear the TSTRA.CST3 (TSTRB.CST6) and TSTRA.CST4 (TSTRB.CST7) bits to 0 to stop the TCNT count operation. Specify the reset-synchronized PWM mode while MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) are stopped.

[2] Set the TCR.TPSC[2:0] and TCR.CKEG[1:0] bits of MTU3 (MTU6) to select the count clock source and clock edge for MTU3 (MTU6). Set the TCR.CCLR[2:0] bits to select TGRA compare match as the counter clear source.

[3] To control a brushless DC motor, set the TGCRA.BDC bit and set the feedback signal input source and output chopping or gate signal direct output (only when MTU3 and MTU4 are used).

[4] Set MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) to H'0000.

[5] MTU3.TGRA (MTU6.TGRA) is the cycle register. Set the PWM cycle in MTU3.TGRA (MTU6.TGRA).
Set the PWM output transition timing in MTU3.TGRB (MTU6.TGRB), MTU4.TGRA (MTU7.TGRA), and MTU4.TGRB (MTU7.TGRB).
Note that the setting values should be within the range of compare match with MTU3.TCNT (MTU6.TCNT).

$X \leq$ MTU3.TGRA (X: Set value)

[6] Set the PSYE bit in the TOCR1A (TOCR1B) register to enable or disable toggle output synchronized with the PMW cycle, and set the OLSP and OLSN bits to select the PWM output level.
When specifying the PWM output level using the TOCR2A (TOCR2B) and TOLBRA (TOLBRB) in buffer operation, see Timer Output Level Buffer Registers (TOLBRA and TOLBRB).

[7] Set the MD[3:0] bits in MTU3.TMDR1 (MTU6.TMDR1) register to 1000b to select the reset-synchronized PWM mode. Specify the buffer operation (BFA = 1) of the MTU3.TGRA (MTU6.TGRA) register and the buffer operation (BFB = 1) of the MTU3.TGRB register when necessary. Set the MTU4.TMDR1 (MTU7.TMDR1) register to the initial value (normal operation; BFA = 0 and BFB = 0).

[8] Set the TOERA (TOERB) register to enable or disable output from the PWM output pin. Then, set the MTU3.TIORH (MTU6.TIORH), MTU3.TIORL (MTU6.TIORL), MTU4.TIORH (MTU7.TIORH), and MTU4.TIORL (MTU7.TIORL) registers to H'00.

[9] Set the port mode control registers (PMCn) and Pmn port fuction control reigister (PFCm) of the GPIO ports.

[10] Set the TSTRA.CST3 (TSTRB.CST6) bit to 1 to start count operation.

Note: The output waveform starts to toggle at the point of MTU3.TCNT (MTU6.TCNT) = MTU3.TGRA (MTU6.TGRA) = X by setting X = MTU3.TGRA (MTU6.TGRA), i.e., cycle = duty ratio.

Figure 16.44 Procedure for Selecting Reset-Synchronized PWM Mode

(2) Example of Reset-Synchronized PWM Mode Operation

Figure 16.45 shows an example of operation in the reset-synchronized PWM mode.

MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT) operate as up-counters. The counters are cleared when a compare match occurs between MTU3.TCNT (MTU6.TCNT) and MTU3.TGRA (MTU6.TGRA), and then begin incrementing from H’0000. The output from the PWM pins toggles every time a compare match occurs in MTU3.TGRB (MTU6.TGRB), MTU4.TGRA (MTU7.TGRA), and MTU4.TGRB (MTU7.TGRB) and the counters are cleared.

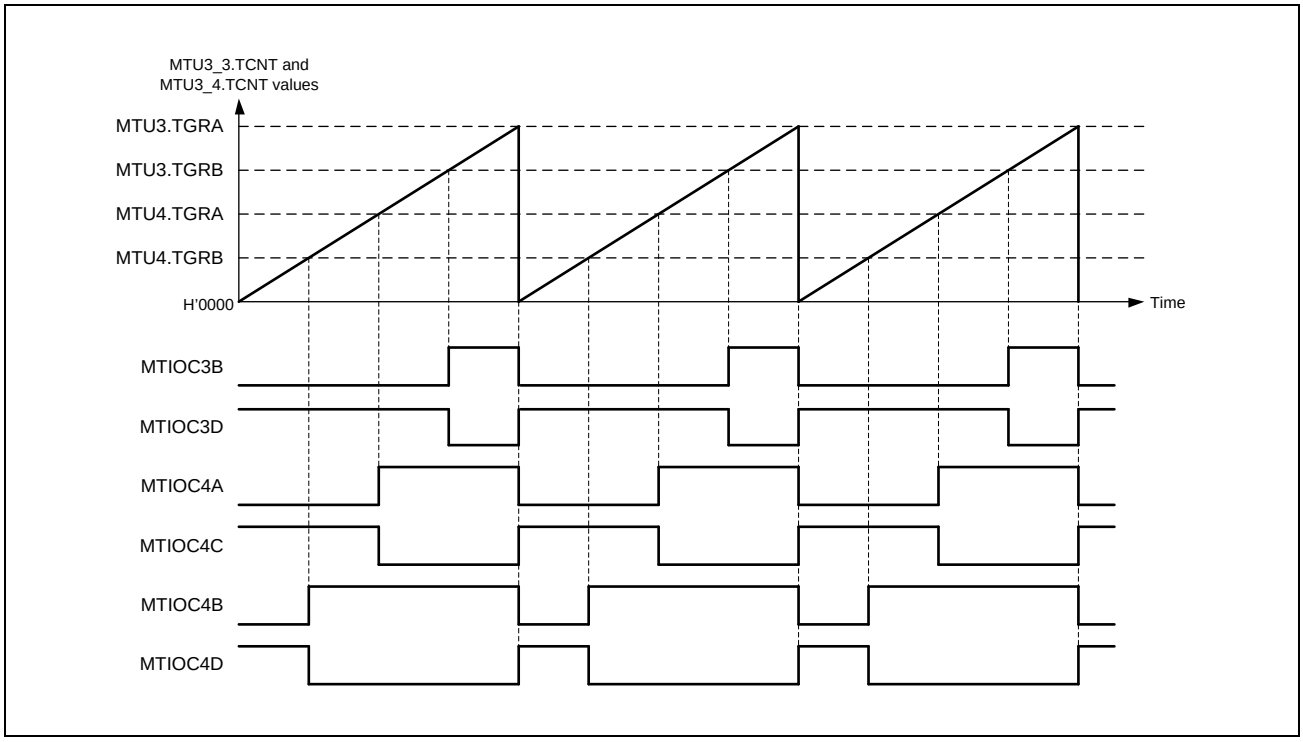


Figure 16.45 Example of Reset-Synchronized PWM Mode Operation
(When OLSN = 1 and OLSP = 1 in MTU3.TOCR1 and MTU4.TOCR1)

16.3.8 Complementary PWM Mode

In complementary PWM mode, dead time can be set for PWM waveforms to be output. The dead time is the period during which the upper and lower arm transistors are set to the inactive level in order to prevent short-circuiting of the arms.

Six positive-phase and six negative-phase PWM waveforms (12 phases in total) with dead time can be output by combining MTU3/ MTU4 and MTU6/MTU7. PWM waveforms without dead time can also be output.

In complementary PWM mode, MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D, MTIOC6B,

MTIOC6D, MTIOC7A, MTIOC7B, MTIOC7C, and MTIOC7D pins function as PWM output pins, and the MTIOC3A and MTIOC6A pins can be set for toggle output synchronized with the PWM cycle.

MTU3.TCNT, MTU4.TCNT, MTU6.TCNT, and MTU7.TCNT function as up/down-counters.

Table 16.74 shows the PWM output pins used. **Table 16.75** shows the settings of the registers used.

A function to directly cut off the PWM output by using an external signal is supported as a port function.

Table 16.74 Output Pins for Complementary PWM Mode

Channel	Output Pin	Description
MTU3	MTIOC3A	Toggle output synchronized with PWM cycle (or I/O port)
	MTIOC3B	PWM output pin 1
	MTIOC3C	I/O port*1
	MTIOC3D	PWM output pin 1' (negative-phase waveform output of PWM output 1)
MTU4	MTIOC4A	PWM output pin 2
	MTIOC4C	PWM output pin 2' (negative-phase waveform output of PWM output 1)
	MTIOC4B	PWM output pin 3
	MTIOC4D	PWM output pin 3' (negative-phase waveform output of PWM output 1)
MTU6	MTIOC6A	Toggle output synchronized with PWM cycle (or I/O port)
	MTIOC6B	PWM output pin 4
	MTIOC6C	I/O port*1
	MTIOC6D	PWM output pin 4' (negative-phase waveform output of PWM output 4)
MTU7	MTIOC7A	PWM output pin 5
	MTIOC7C	PWM output pin 5' (negative-phase waveform output of PWM output 5)
	MTIOC7B	PWM output pin 6
	MTIOC7D	PWM output pin 6' (negative-phase waveform output of PWM output 6)

Note 1. Avoid setting the MTIOC3C and MTIOC6C pins as timer I/O pins in complementary PWM mode.

Table 16.75 Register Settings for Complementary PWM Mode (1/2)

Channel	Counter / Register	Description	Read/Write from CPU
MTU3	TCNT	Starts up-counting from the value set in the dead time register	Maskable by TRWERA setting*1
	TGRA	Set MTU3.TCNT upper limit value (1/2 carrier cycle + dead time)	Maskable by TRWERA setting*1
	TGRB	PWM output 1 compare register	Maskable by TRWERA setting*1
	TGRC	MTU3.TGRA buffer register	Readable/writable
	TGRD	PWM output 1/MTU3.TGRB buffer register	Readable/writable
	TGRE	MTU3.TGRB buffer register B (when double buffer function is used)	Readable/writable
MTU4	TCNT	Starts up-counting after being initialized to H'0000	Maskable by TRWERA setting*1
	TGRA	PWM output 2 compare register	Maskable by TRWERA setting*1
	TGRB	PWM output 3 compare register	Maskable by TRWERA setting*1
	TGRC	PWM output 2/MTU4.TGRA buffer register	Readable/writable
	TGRD	PWM output 3/MTU4.TGRB buffer register	Readable/writable
	TGRE	MTU4.TGRA buffer register B (when double buffer function is used)	Readable/writable
	TGRF	MTU4.TGRB buffer register B (when double buffer function is used)	Readable/writable
MTU6	TCNT	Starts up-counting from the value set in the dead time register	Maskable by TRWERB setting*2
	TGRA	Set MTU6.TCNT upper limit value (1/2 carrier cycle + dead time)	Maskable by TRWERB setting*2
	TGRB	PWM output 4 compare register	Maskable by TRWERB setting*2
	TGRC	MTU6.TGRA buffer register	Readable/writable
	TGRD	PWM output 4/MTU6.TGRB buffer register	Readable/writable
	TGRE	MTU6.TGRB buffer register B (when double buffer function is used)	Readable/writable
MTU7	TCNT	Starts up-counting after being initialized to H'0000	Maskable by TRWERB setting*2
	TGRA	PWM output 5 compare register	Maskable by TRWERB setting*2
	TGRB	PWM output 6 compare register	Maskable by TRWERB setting*2
	TGRC	PWM output 5/MTU7.TGRA buffer register	Readable/writable
	TGRD	PWM output 6/MTU7.TGRB buffer register	Readable/writable
	TGRE	MTU7.TGRA buffer register B (when double buffer function is used)	Readable/writable
	TGRF	MTU7.TGRB buffer register B (when double buffer function is used)	Readable/writable

Note 1. Access can be enabled or disabled according to the setting in TRWERA (timer read/write enable register A).

Note 2. Access can be enabled or disabled according to the setting in TRWERB (timer read/write enable register B).

Table 16.76 Register Settings for Complementary PWM Mode (2/2)

Counter / Register	Description	Read/Write from CPU
Timer dead time data register A (TDDRA)	Set MTU4.TCNT and MTU3.TCNT offset value (dead time value)	Maskable by TRWERA setting*1
Timer dead time data register B (TDDRB)	Set MTU7.TCNT and MTU6.TCNT offset value (dead time value)	Maskable by TRWERB setting*2
Timer cycle data register A (TCDRA)	Set MTU4.TCNT upper limit value (1/2 carrier cycle)	Maskable by TRWERA setting*1
Timer cycle data register B (TCDRB)	Set MTU7.TCNT upper limit value (1/2 carrier cycle)	Maskable by TRWERB setting*2
Timer cycle buffer register A (TCBRA)	TCDRA buffer register	Readable/writable
Timer cycle buffer register B (TCBRB)	TCDRB buffer register	Readable/writable
Sub counter A (TCNTSA)	Sub counter A for dead time generation	Read-only
Sub counter B (TCNTSB)	Sub counter B for dead time generation	Read-only
Temporary register 1A (TEMP1A)	PWM output 1/MTU3.TGRB temporary register A	Not readable or writable
Temporary register 1B (TEMP1B)	PWM output 1/MTU3.TGRB temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 2A (TEMP2A)	PWM output 2/MTU4.TGRA temporary register A	Not readable or writable
Temporary register 2B (TEMP2B)	PWM output 2/MTU4.TGRA temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 3A (TEMP3A)	PWM output 3/MTU4.TGRB temporary register A	Not readable or writable
Temporary register 3B (TEMP3B)	PWM output 3/MTU4.TGRB temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 4A (TEMP4A)	PWM output 4/MTU6.TGRB temporary register A	Not readable or writable
Temporary register 4B (TEMP4B)	PWM output 4/MTU6.TGRB temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 5A (TEMP5A)	PWM output 5/MTU7.TGRA temporary register A	Not readable or writable
Temporary register 5B (TEMP5B)	PWM output 5/MTU7.TGRA temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 6A (TEMP6A)	PWM output 6/MTU7.TGRB temporary register A	Not readable or writable
Temporary register 6B (TEMP6B)	PWM output 6/MTU7.TGRB temporary register B (when double buffer function is used)	Not readable or writable

Note 1. Access can be enabled or disabled according to the setting in TRWERA (timer read/write enable register A).

Note 2. Access can be enabled or disabled according to the setting in TRWERB (timer read/write enable register B).

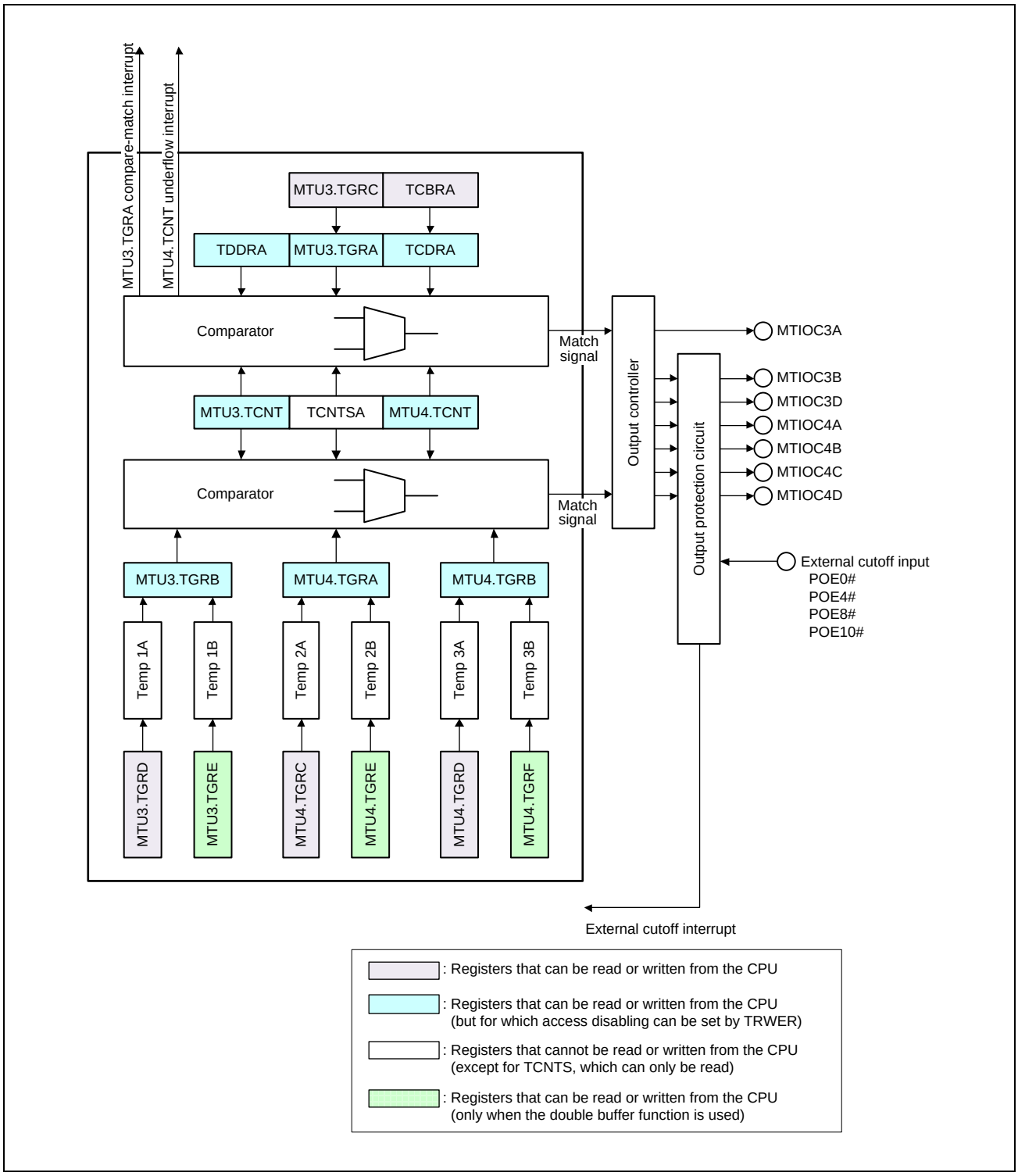


Figure 16.46 Block Diagram of MTU3 and MTU4 in Complementary PWM Mode

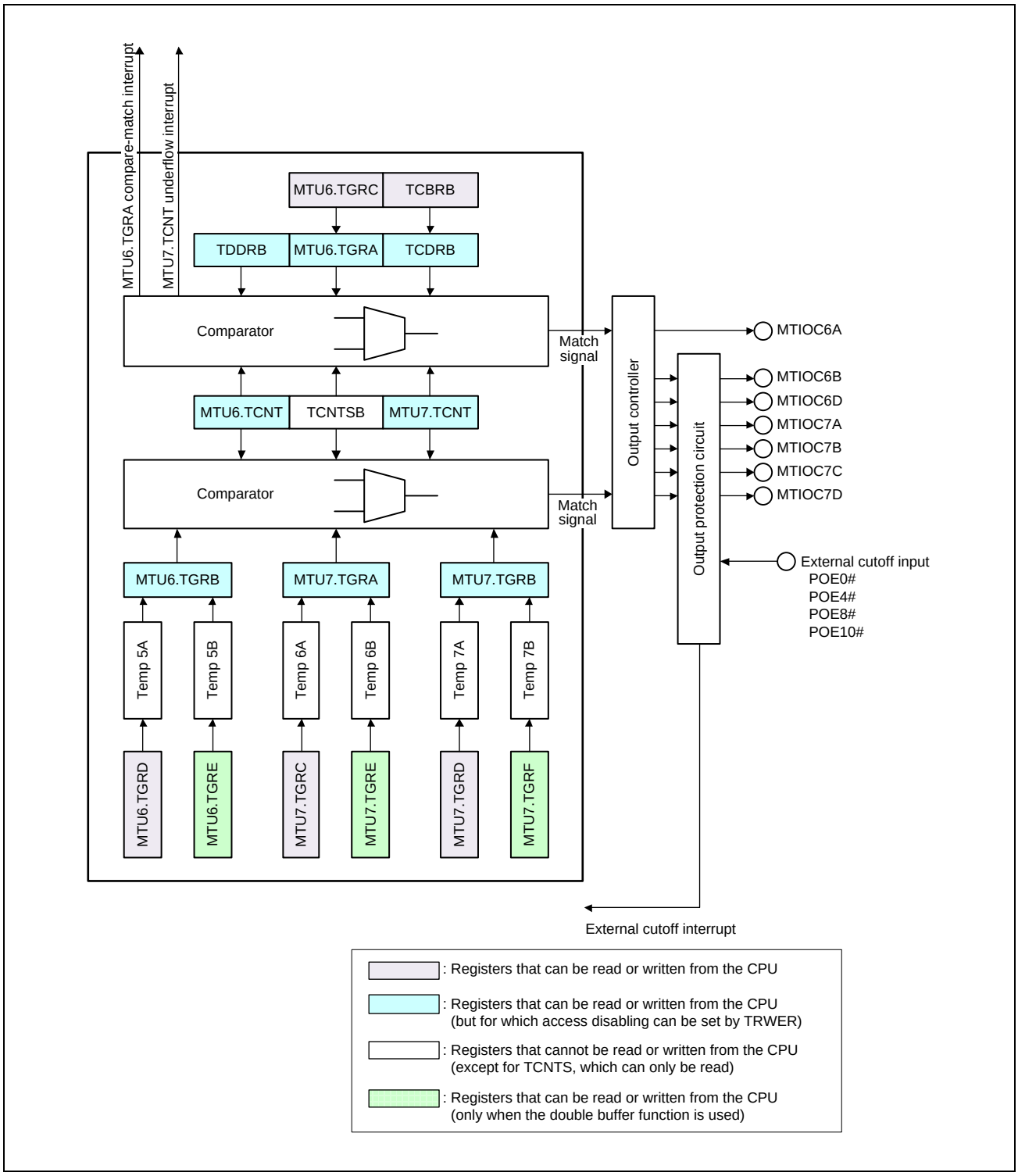
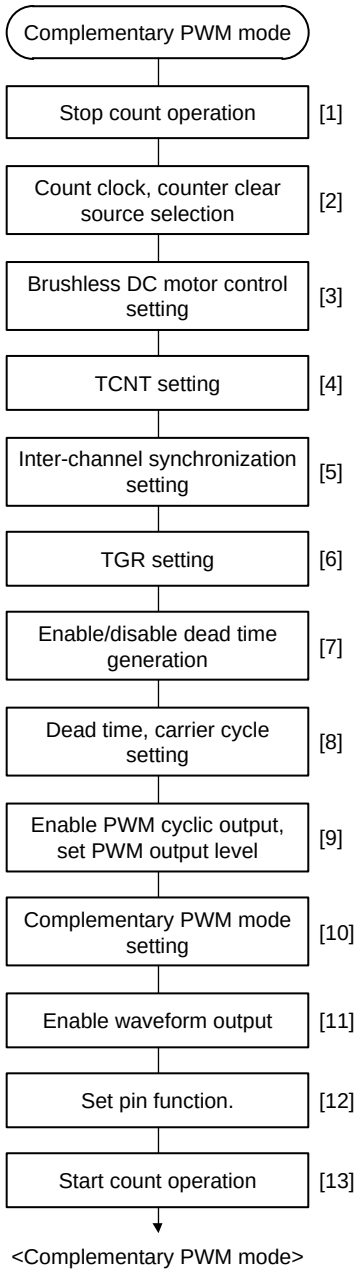


Figure 16.47 Block Diagram of MTU6 and MTU7 in Complementary PWM Mode

(1) Example of Complementary PWM Mode Setting Procedure

Figure 16.48 shows an example of the complementary PWM mode setting procedure.



[1] Clear the TSTRA.CST3 (TSTRB.CST6) and TSTRA.CST4 (TSTRB.CST7) bits to 0 to stop the TCNT count operation.
Specify complementary PWM mode while MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) are stopped.

[2] Set the TCR.TPSC[2:0] and TCR.CKEG[1:0] bits of MTU3 and MTU4 (MTU6 and MTU7) to select the same count clock source and clock edge for MTU3 and MTU4 (MTU6 and MTU7).
Set the TCR.CCLR[2:0] bits only when restarting by synchronous clearing with another channel during complementary PWM operation.

[3] To control a brushless DC motor, set the TGCRA.BDC bit and set the feedback signal input source and output chopping or gate signal direct output (only when MTU3 and MTU4 are used).

[4] Set the dead time in MTU3.TCNT (MTU6.TCNT). Set MTU4.TCNT (MTU7.TCNT) to H'0000.

[5] Make this setting only when restarting by synchronous clearing with another channel during complementary PWM mode operation. In this case, set the TSYRA register (TSYRB) to synchronize the channel generating the synchronous clear with MTU3 and MTU4 (MTU6 and MTU7).

[6] When MTU3 and MTU6 are used, set the PWM output duty ratio in the duty registers (MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB) and buffer registers (MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD).
Be sure to set the same value in each pair of the duty registers (MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD) and buffer registers (MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB).
Only when the double buffer function is used, set the PWM output duty value − 1 in the buffer registers (MTU3.TGRE, MTU4.TGRE, and MTU4.TGRF).
When MTU6 and MTU7 are used, set the value to the following registers:
Duty registers (MTU6.TGRB, MTU7.TGRA, and MTU7.TGRB)
Buffer registers (MTU6.TGRD, MTU7.TGRC, and MTU7.TGRD)
Buffer registers when the double buffer function is used (MTU6.TGRE, MTU7.TGRE, and MTU7.TGRF)

[7] This setting is necessary only when no dead time should be generated. Make appropriate settings in the TDERA (TDERB) register so that no dead time generated.

[8] Set the dead time in the TDDRA (TDDRB) register, 1/2 the PWM cycle in the TCDRA (TCDRB) register and TCBRA (TCBRB) register, and 1/2 the PWM cycle plus the dead time in MTU3.TGRA (MTU6.TGRA) and MTU3.TGRC (MTU6.TGRC).
When no dead time generation is selected, set 1 in the TDDRA (TDDRB) register and 1/2 the PWM cycle + 1 in MTU3.TGRA (MTU6.TGRA) and MTU3.TGRC (MTU6.TGRC).

[9] Set the PSYE bit in the TOCR1A register to enable or disable toggle output synchronized with the PWM cycle, and set the OLSP and OLSN bits to select the PWM output level. When specifying the PWM output level using the TOCR2A (TOCR2B) and TOLBRA (TOLBRA) in buffer operation, see Timer Output Level Buffer Registers (TOLBRA and TOLBRB).

[10] Select complementary PWM mode in MTU3.TMDR1 (MTU6.TMDR1). Specify the buffer operation (BFA = 1) of MTU3.TGRA (MTU6.TGRA) and the buffer operation (BFB = 1) of MTU3.TGRB (MTU6.TGRB).
Only when the double buffer function is used, set the TMDR2A (TMDR2B) register to enable the double buffer function.

[11] Set the TOERA (TOERB) register to enable or disable output from the PWM output pin. Then, set the MTU3.TIORH (MTU6.TIORH), MTU3.TIORL (MTU6.TIORL), MTU4.TIORH (MTU7.TIORH), and MTU4.TIORL (MTU7.TIORL) to H'00.

[12] Set the port mode registers (PMn) and Pmn port function control register (PFCm) of the GPIO ports.

[13] Set the TSTRA.CST3 (TSTRB.CST6) and TSTRA.CST4 (TSTRB.CST7) bits to 1 simultaneously to start count operation.

Figure 16.48 Example of Complementary PWM Mode Setting Procedure

(2) Outline of Complementary PWM Mode Operation

In complementary PWM mode, six phases (three positive and three negative) PWM waveforms can be output. **Figure 16.49** illustrates counter operation in complementary PWM mode (MTU3 and MTU4), and **Figure 16.50** shows an example of operation in complementary PWM mode.

(a) Counter Operation

In complementary PWM mode, three counters — MTU3.TCNT, MTU4.TCNT, and TCNTSA (MTU6.TCNT, MTU7.TCNT, and TCNTSB) — in each unit perform up-/down-count operations.

MTU3.TCNT (MTU6.TCNT) is automatically initialized to the value set in TDDRA (TDDRB) when complementary PWM mode is selected and the CST3 bit in TSTRA (TSTRB) is 0. When the CST3 bit is set to 1, MTU3.TCNT (MTU6.TCNT) counts up to the value set in MTU3.TGRA (MTU6.TGRA), then switches to down-counting when it matches MTU3.TGRA (MTU6.TGRA). When the MTU4.TCNT (MTU7.TCNT) value matches H’0000, MTU3.TCNT (MTU6.TCNT) switches to up-counting, and the operation is repeated in this way.

TCNT in MTU4 (MTU7) should be initialized to H’0000 after a reset. When the CST4 bit is set to 1, MTU4.TCNT (MTU7.TCNT) counts up in synchronization with MTU3.TCNT (MTU6.TCNT), and switches to down-counting when MTU3.TCNT (MTU6.TCNT) matches MTU3.TGRA (MTU6.TGRA). On reaching H’0000, MTU4.TCNT (MTU7.TCNT) switches to up-counting, and the operation is repeated in this way. TCNTSA (TCNTSB) is a read-only counter. It does not need to be initialized after a reset.

In counting up by MTU3.TCNT and MTU4.TCNT (or MTU6.TCNT and MTU7.TCNT), MTU3.TCNT (or MTU6.TCNT) starts counting up when it matches TCDRA (or TCDRB) and switches to counting down when it matches MTU3.TGRA (or MTU6.TGRA). Furthermore, when MTU4.TCNT (or MTU7.TCNT) matches TDDRA (or TDDRB), TCNTSA (or TCNTSB) is set to the value in MTU3.TGRA (or MTU6.TGRA) and counting is stopped.

When MTU4.TCNT (MTU7.TCNT) matches TDDRA (TDDRB) during down-counting of MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT), TCNTSA (TCNTSB) starts up-counting, and when MTU4.TCNT (MTU7.TCNT) matches H’0000, the operation switches to down-counting. When MTU3.TCNT (MTU6.TCNT) matches TCDRA (TCDRB), TCNTSA (TCNTSB) is cleared to H’0000 and stops counting.

TCNTSA (TCNTSB) is compared with the compare register and temporary register, in which the PWM duty is specified, only during the count operation.

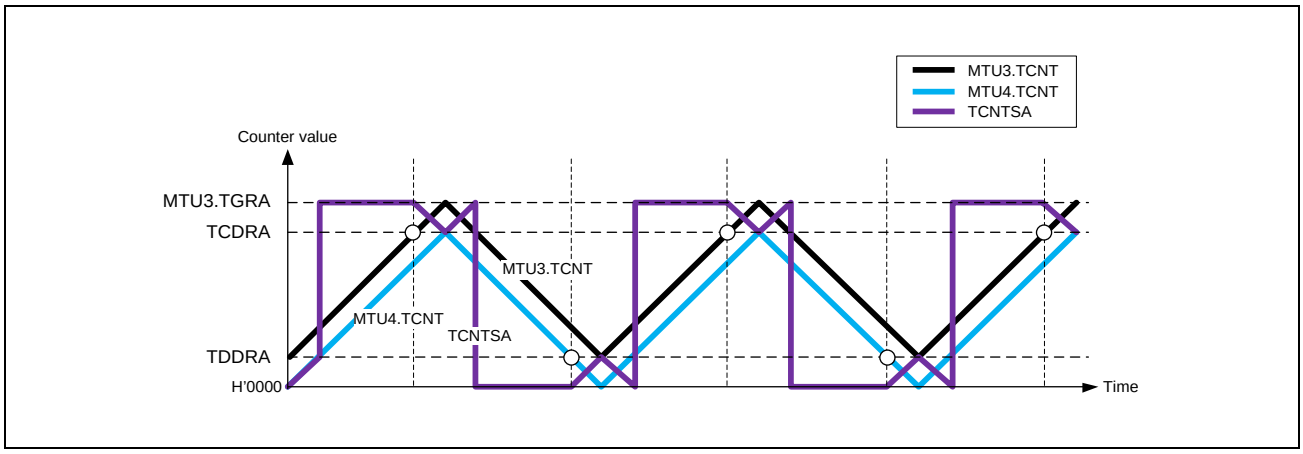


Figure 16.49 Count Operation in Complementary PWM Mode (MTU3 and MTU4)

(b) Register Operation

In complementary PWM mode, nine registers (compare registers, buffer registers, and temporary registers) are used to control the duty ratio for the PWM output. **Figure 16.50** shows an example of operation in complementary PWM mode (MTU3 and MTU4).

MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB (MTU6.TGRB, MTU7.TGRA, and MTU7.TGRB) are constantly compared with the counters to generate PWM waveforms. When these registers match the counter, the value set in the OLSN and OLSP bits in the timer output control registers (TOCR1A and TOCR1B) is output from the PWM output pin. MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD (MTU6.TGRD, MTU7.TGRC, and MTU7.TGRD) are buffer registers for these compare registers.

When the double buffer function is used, MTU3.TGRE, MTU4.TGRE, and MTU4.TGRF (MTU6.TGRE, MTU7.TGRE, and MTU7.TGRF) are also used as buffer registers B. For details of double buffer operation, refer to **Section 16.3.8(2)(s), Double Buffer Function in Complementary PWM Mode**.

Data in a compare register can be changed by writing new data to the corresponding buffer register. The buffer registers can be read or written at any time.

When modifying data in a buffer register, write to MTU4.TGRD (MTU7.TGRD) last and enable data transfer from the buffer register to a temporary register. At this time, transfer from the TCBRA (TCBRB) register and MTU3.TGRC (MTU6.TGRC) register, which operate as buffer registers for the timer cycle registers, to temporary registers is also enabled. Data is transferred to all five temporary registers at the same time.

When transfer is enabled in the Ta interval, data written to a buffer register is transferred to the temporary register. The data is not transferred to the temporary register in the Tb1 and Tb2 intervals. Data enabled for transfer in this interval is transferred to the temporary register at the end of this interval.

The value transferred to a temporary register is transferred to the compare register at the end of the Tb1 interval (when matches MTU3.TGRA (MTU6.TGRA) while TCNTSA (TCNTSB) is counting up), or at the end of the Tb2 interval (when matches H’0000 while TCNTSA (TCNTSB) is counting down). The timing for transfer from the temporary register to the compare register can be selected with bits MD[3:0] in the timer mode register 1 (TMDR1). **Figure 16.50** shows an example in which the trough is selected for the transfer timing.

In the Tb interval in which data is not transferred to the temporary register (Tb1 in **Figure 16.50**), the temporary register has the same function as the compare register and is compared with the counter. In this interval, therefore, there are two compare match registers for one output phase; the compare register contains the pre-change data and the temporary register contains new data. In this interval, three counters MTU3.TCNT, MTU4.TCNT, and TCNTSA (MTU6.TCNT, MTU7.TCNT, and TCNTSB) and two registers (compare register and temporary register) are compared, and PWM output is controlled accordingly.

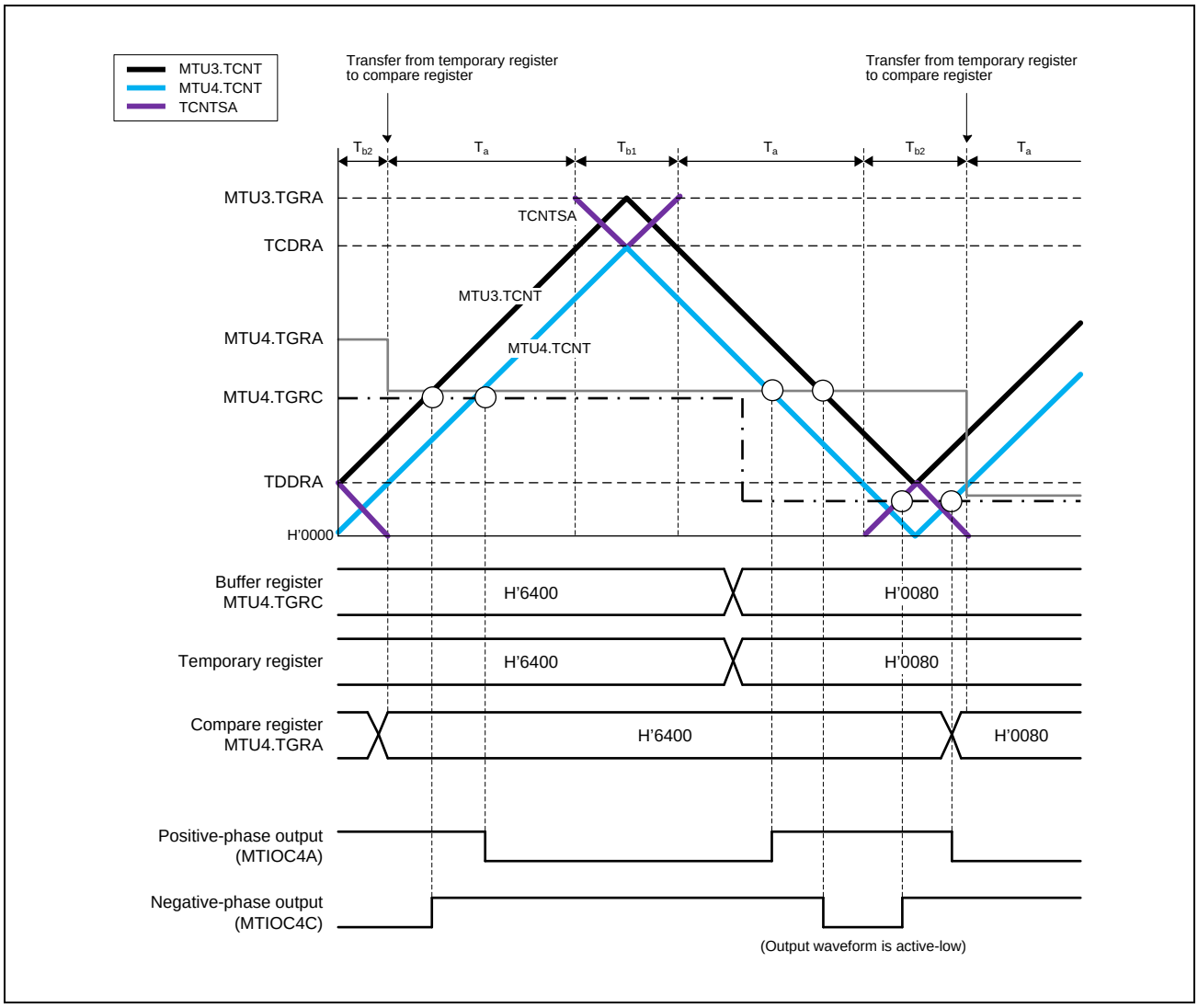


Figure 16.50 Example of Operation in Complementary PWM Mode (MTU3 and MTU4)

(c) Initial Setting

In complementary PWM mode, there are nine registers that require initial setting. In addition, there is a register that specifies whether to generate dead time (it should be used only when dead time generation should be disabled).

Before setting complementary PWM mode with MTU3.TMDR1.MD[3:0] (MTU6.TMDR1.MD[3:0]) bits, initial values should be set in the following registers.

The TOCR1A, TOCR2A, TOCR1B, and TOCR2B registers are used to set the PWM output level. MTU3.TGRC (MTU6.TGRC) operates as the buffer register for MTU3.TGRA (MTU6.TGRA), and should be set with 1/2 the PWM cycle + dead time Td. The timer cycle buffer register (TCBRA or TCBRB) operates as the buffer register for the timer cycle data register (TCDRA or TCDRB), and should be set with 1/2 the PWM cycle. Set dead time Td in the timer dead time data register (TDDRA or TDDRB).

When dead time is not needed, the TDER bit in the timer dead time enable register (TDERA or TDERB) should be cleared to 0, MTU3.TGRC and MTU3.TGRA (MTU6.TGRC and MTU6.TGRA) should be set to 1/2 the PWM carrier cycle + 1, and TDDRA (TDDRB) should be set to 1.

Set the respective initial PWM duty values in three buffer registers A (MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD (MTU6.TGRD, MTU7.TGRC, and MTU7.TGRD)).

Set the respective (initial PWM duty – 1) values in three buffer registers B (MTU3.TGRE, MTU4.TGRE, and MTU4.TGRF (MTU6.TGRE, MTU7.TGRE, and MTU7.TGRF)) only when the double buffer function is used.

The values set in the five buffer registers excluding TDDRA (TDDRB) are transferred to the corresponding compare registers as soon as complementary PWM mode is set.

Set MTU4.TCNT (MTU7.TCNT) to H’0000 before setting complementary PWM mode.

Table 16.77 Registers and Counters Requiring Initial Setting

Register and Counter	Setting
TOCR1A, TOCR2A, TOCR1B, TOCR2B	PWM output level
MTU3.TGRC MTU6.TGRC	1/2 PWM cycle + dead time Td (1/2 PWM cycle + 1 when dead time generation is disabled by the TDERA or TDERB setting)
TDDRA, TDDRB	Dead time Td (1 when dead time generation is disabled by the TDERA or TDERB setting)
TCBRA, TCBRB	1/2 PWM cycle
MTU3.TGRD, MTU4.TGRC, MTU4.TGRD MTU6.TGRD, MTU7.TGRC, MTU7.TGRD	Initial PWM duty ratio value for each phase
MTU3.TGRE, MTU4.TGRE, MTU4.TGRF MTU6.TGRE, MTU7.TGRE, MTU7.TGRF	Initial PWM duty ratio – 1 value for each phase (only when double buffer function is used)
MTU4.TCNT MTU7.TCNT	H’0000

Note: The value set in MTU3.TGRC (MTU6.TGRC) should be the sum of 1/2 the PWM cycle set in TCBRA (TCBRB) and dead time Td set in TDDRA (TDDRB). When dead time generation is disabled by TDERA (TDERB), TGRC should be set to 1/2 the PWM cycle + 1.

(d) PWM Output Level Setting

In complementary PWM mode, the PWM output level is set with bits OLSN and OLSP in timer output control register 1 (TOCR1A or TOCR1B) or bits OLS1P to OLS3P and OLS1N to OLS3N in timer output control register 2 (TOCR2A or TOCR2B).

The output level can be set for each of the three positive phases and three negative phases of 6-phase output. Complementary PWM mode should be cleared before setting or changing output levels.

(e) Dead Time Setting

In complementary PWM mode, dead time can be set for PWM output.

The dead time is set in the timer dead time data register (TDDRA or TDDRB). The value set in TDDRA (TDDRB) is used as the MTU3.TCNT (MTU6.TCNT) counter start value and creates a non-overlapping interval between MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT). Complementary PWM mode should be cleared before changing the contents of TDDRA (TDDRB).

(f) Dead Time Suppressing

Dead time generation is suppressed by clearing the TDER bit in the timer dead time enable register (TDERA or TDERB) to 0. TDERA (TDERB) can be cleared to 0 only when 0 is written to it after reading TDER = 1.

MTU3.TGRA and MTU4.TGRC (MTU6.TGRA and MTU7.TGRC) should be set to 1/2 PWM cycle + 1 and the timer dead time data register (TDDRA or TDDRB) should be set to 1.

By the above settings, PWM waveforms without dead time can be obtained. **Figure 16.51** shows an example of operation without dead time (MTU3 and MTU4).

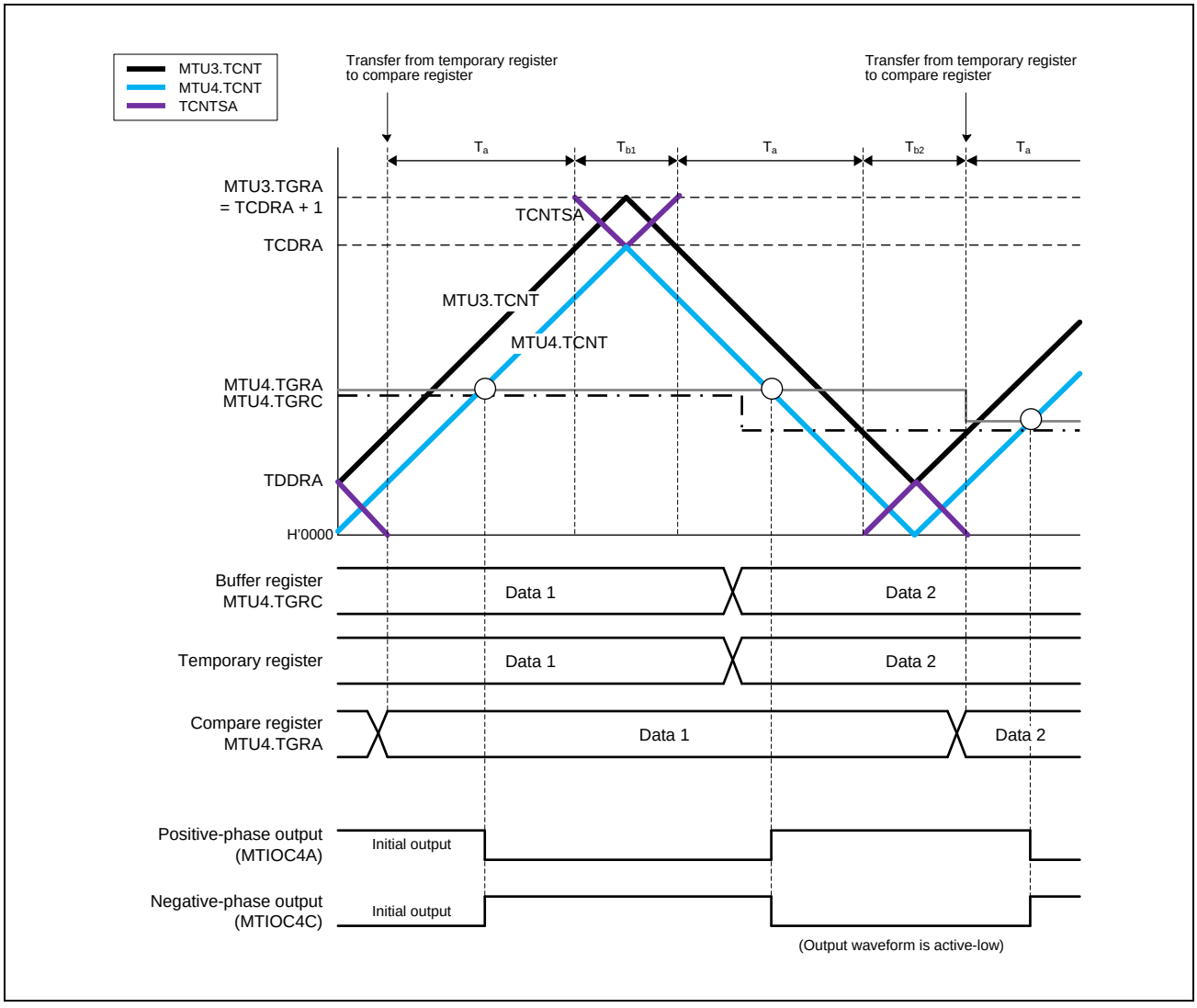


Figure 16.51 Example of Operation without Dead Time (MTU3 and MTU4)

(g) PWM Cycle Setting

In complementary PWM mode, the PWM cycle is set in two registers—MTU3.TGRA (MTU6.TGRA), in which the MTU3.TCNT (MTU6.TCNT) upper limit value is set, and TCDRA (TCDRB), in which the MTU4.TCNT (MTU7.TCNT) upper limit value is set. The settings should be made so as to achieve the following relationship between these two registers:

With dead time: MTU3.TGRA (MTU6.TGRA) setting = TCDRA (TCDRB) setting + TDDRA (TDDRB) setting
Without dead time: MTU3.TGRA (MTU6.TGRA) setting = TCDRA (TCDRB) setting + 1

In addition, the settings should be made so as to achieve the following relationship between the TCDRA (TCDRB) register and the TDDRA (TDDRB) register:

TCDRA (TCDRB) setting > TDDRA (TDDRB) setting × 2 + 2

The MTU3.TGRA and TCDRA (MTU6.TGRA and TCDRB) settings are made by setting values in buffer registers MTU3.TGRC and TCBRA (MTU6.TGRC and TCBRB). When data is written to MTU4.TGRD (MTU7.TGRD) to enable transfers, the values set in MTU3.TGRC and TCBRA (MTU6.TGRC and TCBRB) are transferred simultaneously to the MTU3.TGRA and TCDRA (MTU6.TGRA and TCDRB) with the transfer timing selected with the MTU3.TMDR1.MD[3:0] (MTU6.TMDR1.MD[3:0]) bits.

The new PWM cycle is reflected from the next cycle when data is updated at the crest, or from the current cycle when updated in the trough. **Figure 16.52** illustrates the operation when the PWM cycle is updated at the crest.

See the following **Section 16.3.8(2)(h), Register Data Updating**, for the method of updating the data in each buffer register.

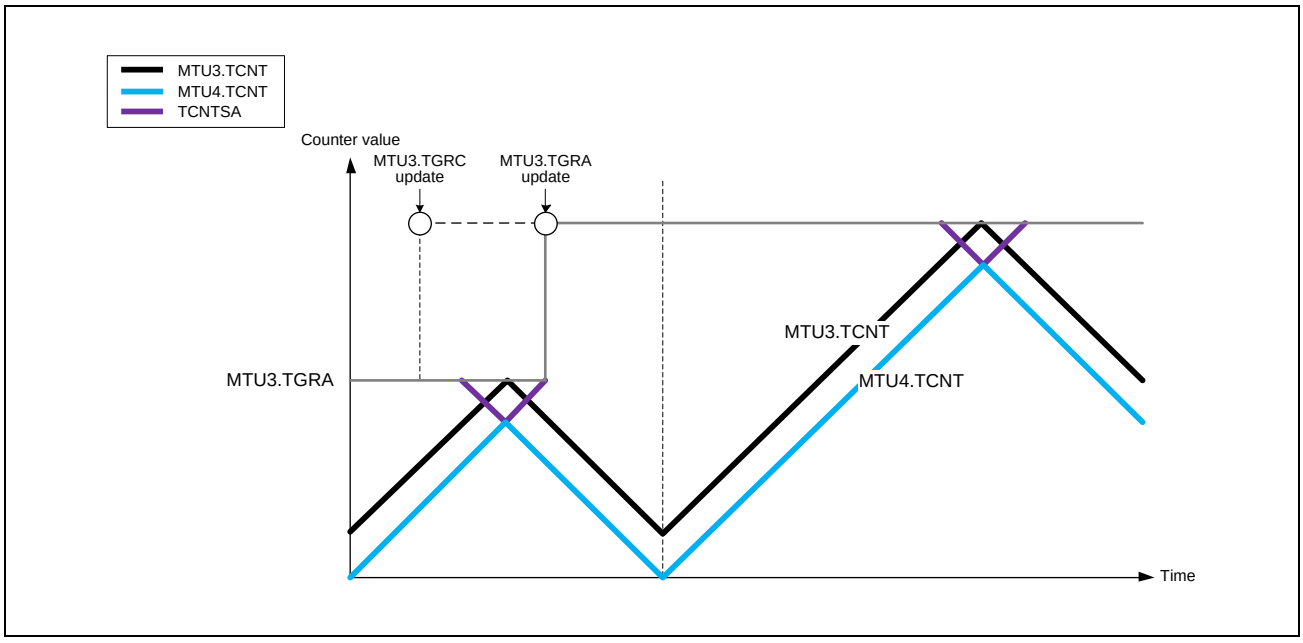


Figure 16.52 Example of PWM Cycle Updating (MTU3 and MTU4)

(h) Register Data Updating

In complementary PWM mode, buffer registers are used to update the data in five compare registers for PWM duty and PWM cycle. The update data can be written to the buffer registers at any time.

There is a temporary register between each of these registers and its buffer register. While sub counter TCNTSA (TCNTSB) is not counting, if buffer register data is updated, the temporary register value also changes. Data is not transferred from buffer registers to temporary registers while TCNTSA (TCNTSB) is counting; in this case, the value written to a buffer register is transferred after TCNTSA (TCNTSB) halts.

The temporary register value is transferred to the compare register at the data update timing set with MTU3.TMDR1.MD[3:0] (MTU6.TMDR1.MD[3:0]) bits. **Figure 16.53** shows an example of data updating in complementary PWM mode (MTU3 and MTU4). This example shows the mode in which data is updated at both the counter crest and trough.

When updating buffer register data, be sure to write to MTU4.TGRD (MTU7.TGRD) at the end of the update. Data is transferred from buffer registers to the temporary registers simultaneously for all five registers after the write to MTU4.TGRD (MTU7.TGRD).

Even when not updating all five registers or when not updating the MTU4.TGRD (MTU7.TGRD) data, be sure to write to MTU4.TGRD (MTU7.TGRD) after writing data to the registers to be updated. In this case, the data written to MTU4.TGRD (MTU7.TGRD) should be the same as the data prior to the write operation.

See **Section 16.3.8(2)(s), Double Buffer Function in Complementary PWM Mode**, for data updating when the double buffer function is used.

● Data update timing: counter crest and trough

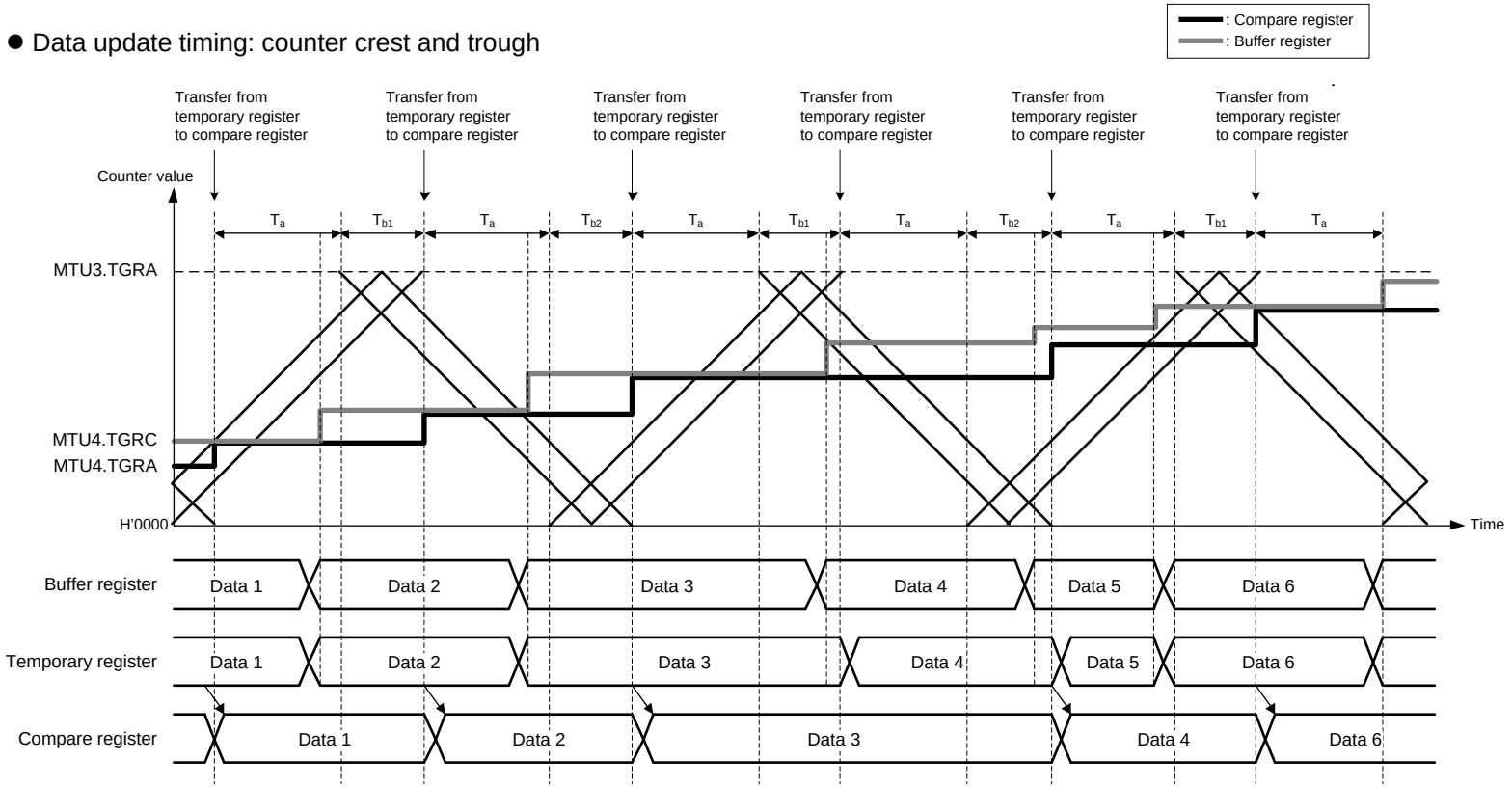


Figure 16.53 Example of Data Updating in Complementary PWM Mode (MTU3 and MTU4)

(i) Initial Output in Complementary PWM Mode

In complementary PWM mode, the initial output is determined by the setting of the OLSN and OLSP bits in the TOCR1A (TOCR1B) register or the OLS1N to OLS3N and OLS1P to OLS3P bits in the TOCR2A register (TOCR2B). This initial output is the non-active level of the PWM output and continues from when complementary PWM mode is set with the MTU3.TMDR1 (MTU6.TMDR1) until MTU4.TCNT (MTU7.TCNT) exceeds the value set in the TDDRA (TDDRB) register. **Figure 16.54** shows an example of the initial output in complementary PWM mode.

An example of the waveform when the initial PWM duty ratio value is smaller than the TDDRA (TDDRB) value is shown in **Figure 16.55**.

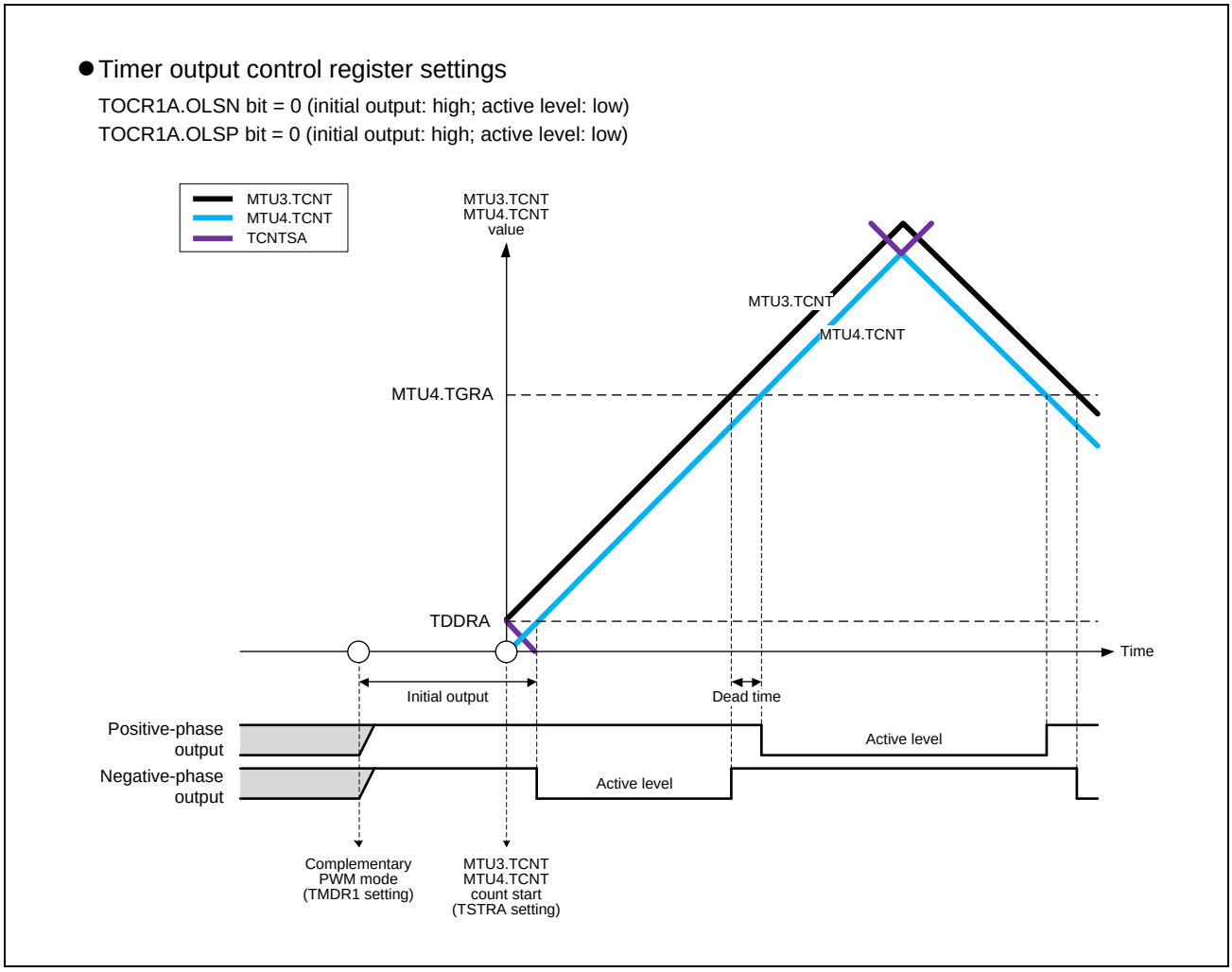


Figure 16.54 Example of Initial Output in Complementary PWM Mode (MTU3 and MTU4) (1)

● Timer output control register settings

TOCR1A.OLSN bit = 0 (initial output: high; active level: low)
TOCR1A.OLSP bit = 0 (initial output: high; active level: low)

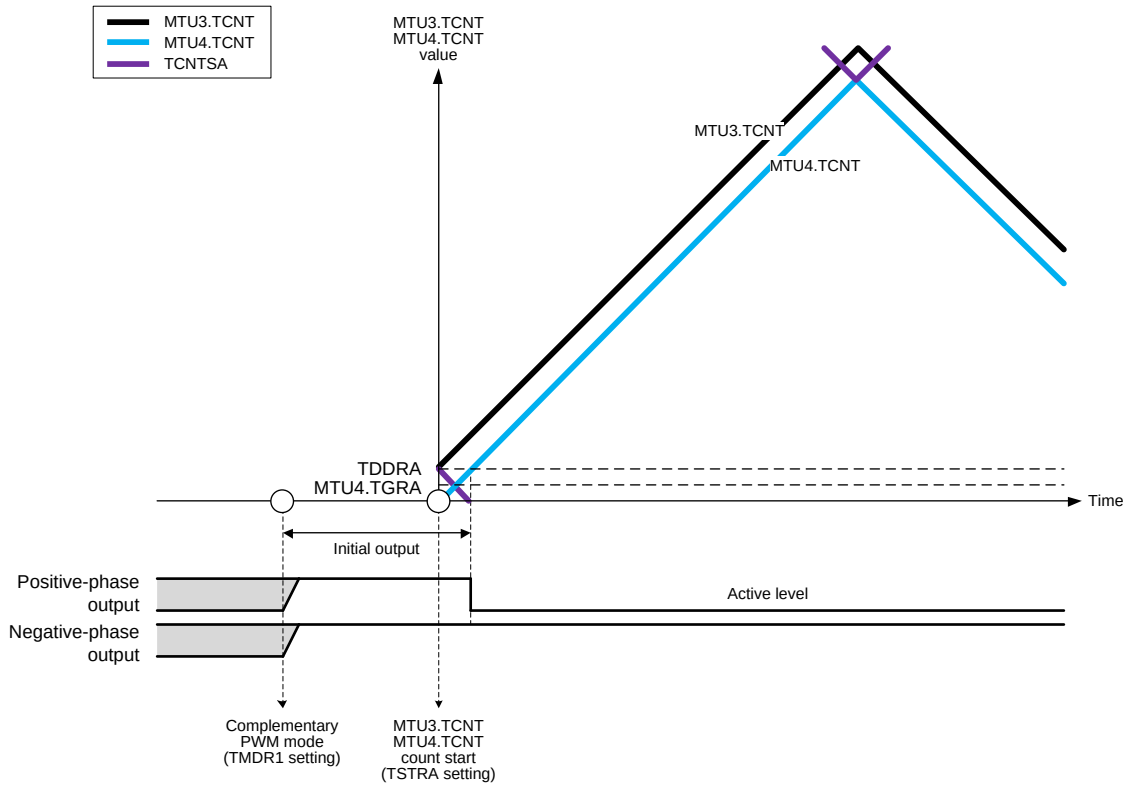


Figure 16.55 Example of Initial Output in Complementary PWM Mode (MTU3 and MTU4) (2)

(j) Method for Generating PWM Output in Complementary PWM Mode

In complementary PWM mode, six phases (three positive and three negative) PWM waveforms can be output. Dead time can be set for PWM waveforms to be output.

A PWM waveform is generated by output of the level selected in the timer output control register in the event of a compare match between a counter and a compare register. While TCNTSA (TCNTSB) is counting, the compare register and temporary register values are simultaneously compared to create consecutive PWM output from 0 to 100% duty ratio. The relative timing of turn-on and turn-off compare match occurrence may vary, but the compare match that turns off each phase takes precedence to secure the dead time and ensure that the positive-phase and negative-phase turn-on times do not overlap. **Figure 16.56** to **Figure 16.58** show examples of waveform generation in complementary PWM mode.

The positive-phase and negative-phase turn-off timing is generated by a compare match with the counter indicated by a solid line, and the turn-on timing is generated by a compare match with the counter indicated by a dotted line, which operates with a delay equal to the dead time behind the counter indicated by a solid line. In the T1 period, compare match a that turns off the negative phase has the highest priority, and compare matches before a are ignored. In the T2 period, compare match c that turns off the positive phase has the highest priority, and compare matches before c are ignored.

In most cases, compare matches occur in the order a → b → c → d (or c → d → a' → b') as shown in **Figure 16.56**.

If compare matches deviate from the a → b → c → d order, since the time for which the negative phase is off is shorter than twice the dead time, the positive phase is not turned on. If compare matches deviate from the c → d → a' → b' order, since the time for which the positive phase is off is shorter than twice the dead time, the negative phase is not turned on. As shown in **Figure 16.57**, if compare match c follows compare match a before compare match b, compare match b is ignored and the negative phase is turned on by compare match d. This is because turning off the positive phase has priority due to the occurrence of compare match c (positive-phase off timing) before compare match b (positive-phase on timing) (consequently, the waveform does not change because the positive phase goes from off to off).

Similarly, in the example in **Figure 16.58**, turning off the negative phase has priority due to the occurrence of compare match a' (negative-phase off timing) before compare match d (negative-phase on timing). As a result, the negative phase is not turned on.

Thus, in complementary PWM mode, compare matches at turn-off timings take precedence, and turn-on timing compare matches that occur before a turn-off timing compare match are ignored.

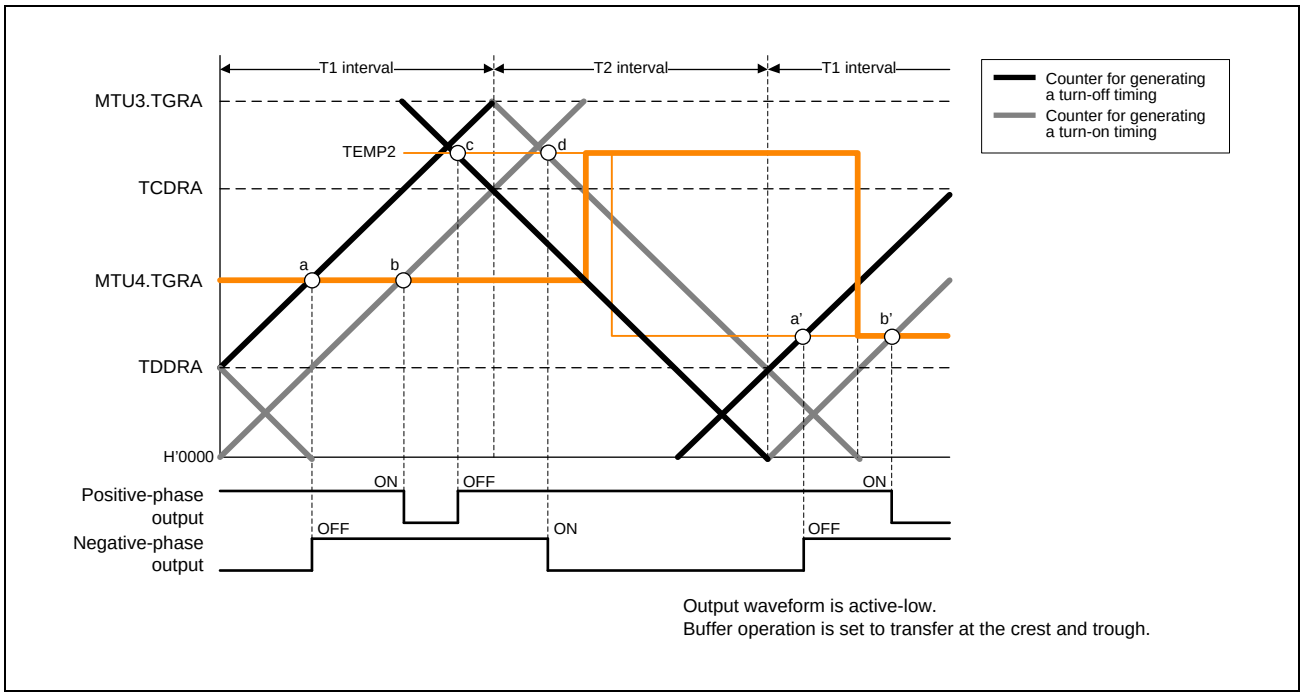


Figure 16.56 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (1)

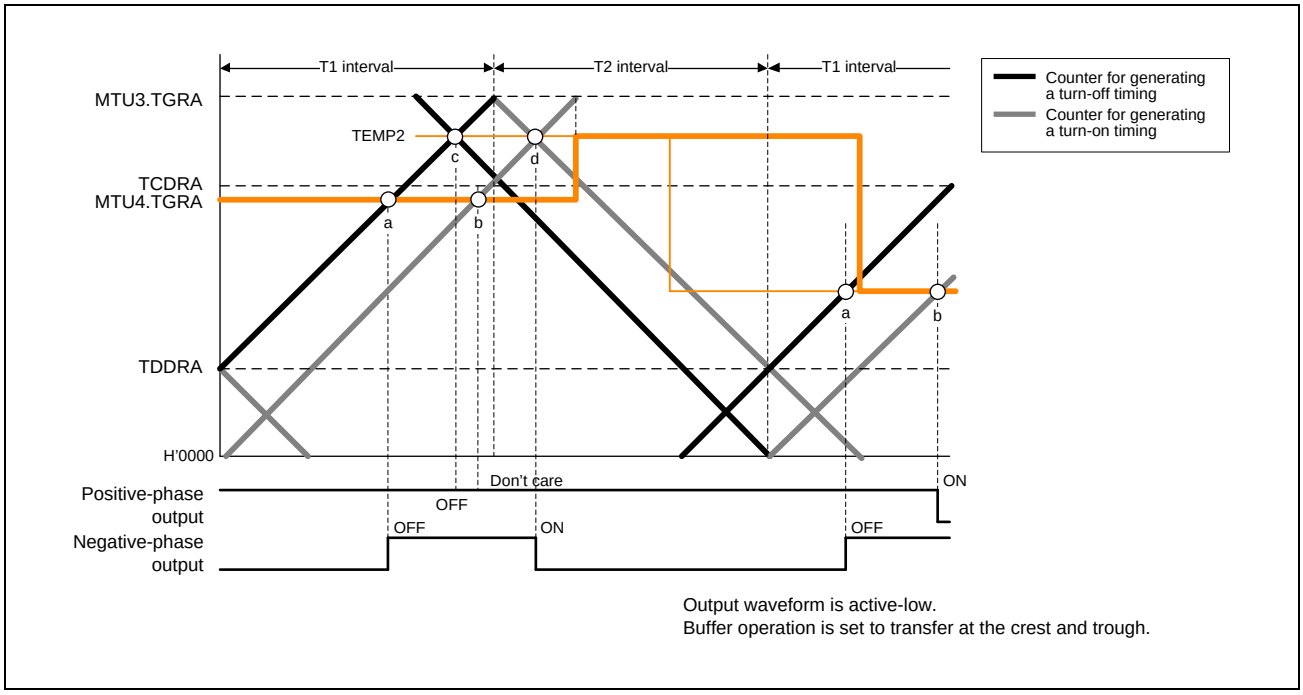


Figure 16.57 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (2)

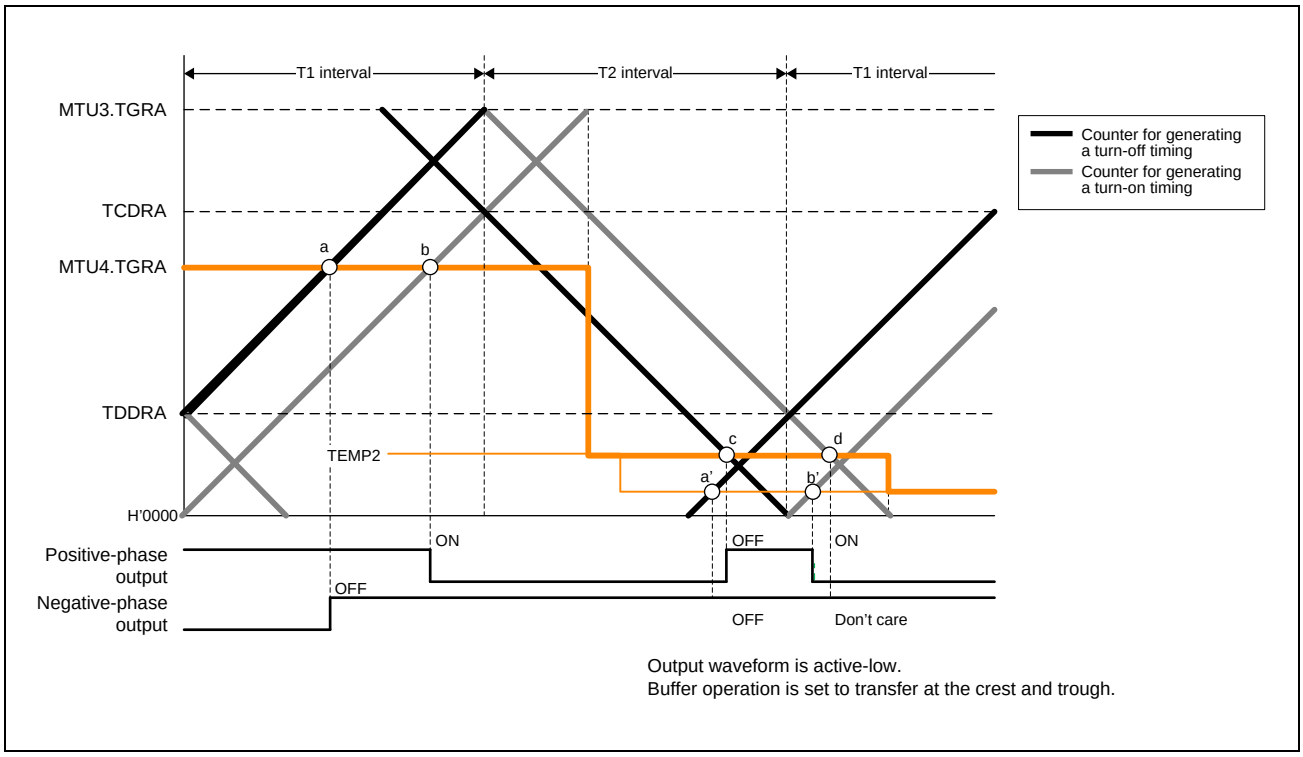


Figure 16.58 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (3)

(k) 0% and 100% Duty Ratio Output in Complementary PWM Mode

In complementary PWM mode, 0% and 100% duty PWM output can be output as required. **Figure 16.59** to **Figure 16.63** show output examples.

A 100% duty waveform is output when the compare register value is set to H’0000. The waveform in this case has a positive phase with a 100% on-state. A 0% duty waveform is output when the compare register value is set to the same value as MTU3.TGRA (MTU6.TGRA). The waveform in this case has a positive phase with a 100% off-state.

Turn-on and turn-off compare matches occur simultaneously, but if a turn-on compare match and turn-off compare match for the same phase occur simultaneously, both compare matches are ignored and the waveform does not change.

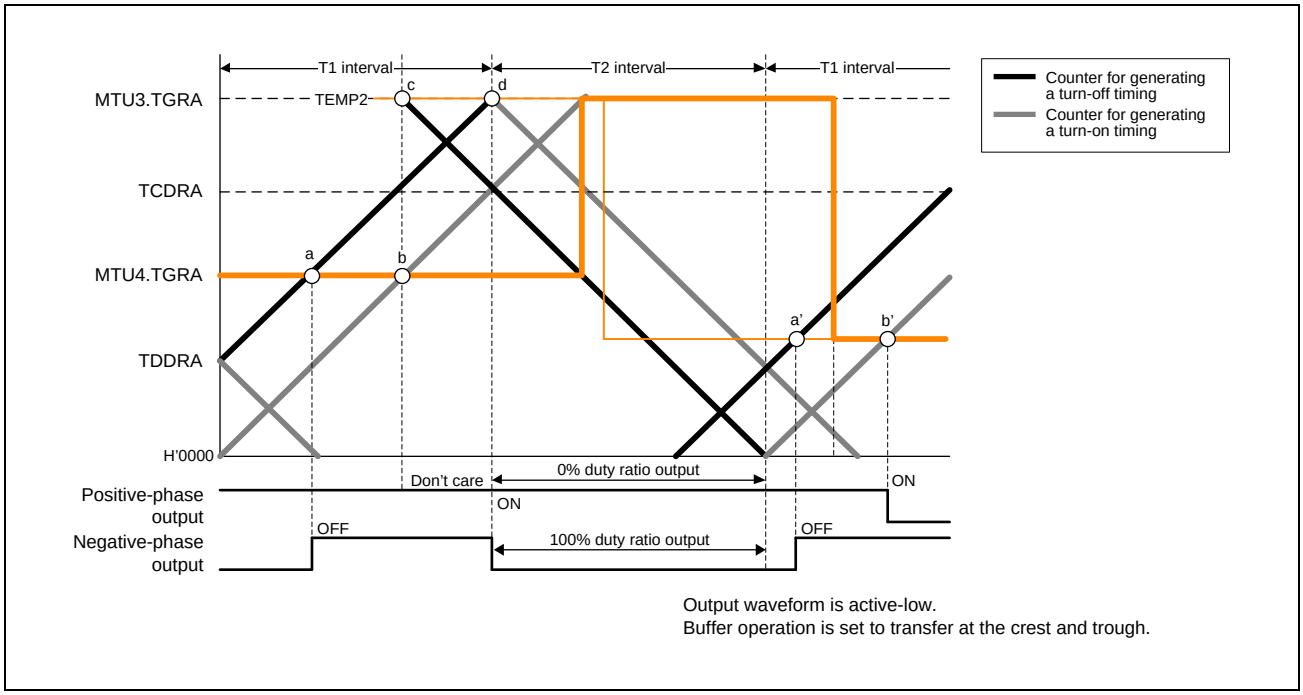


Figure 16.59 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (1)

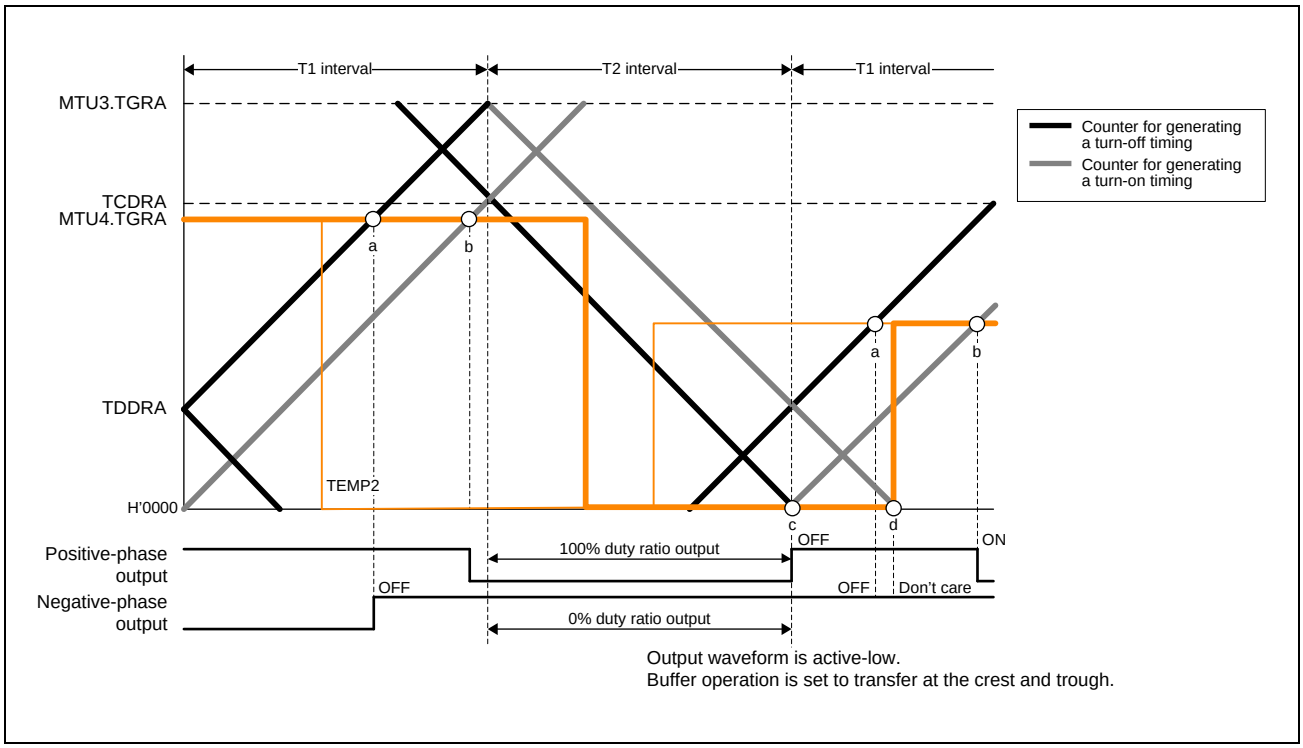


Figure 16.60 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (2)

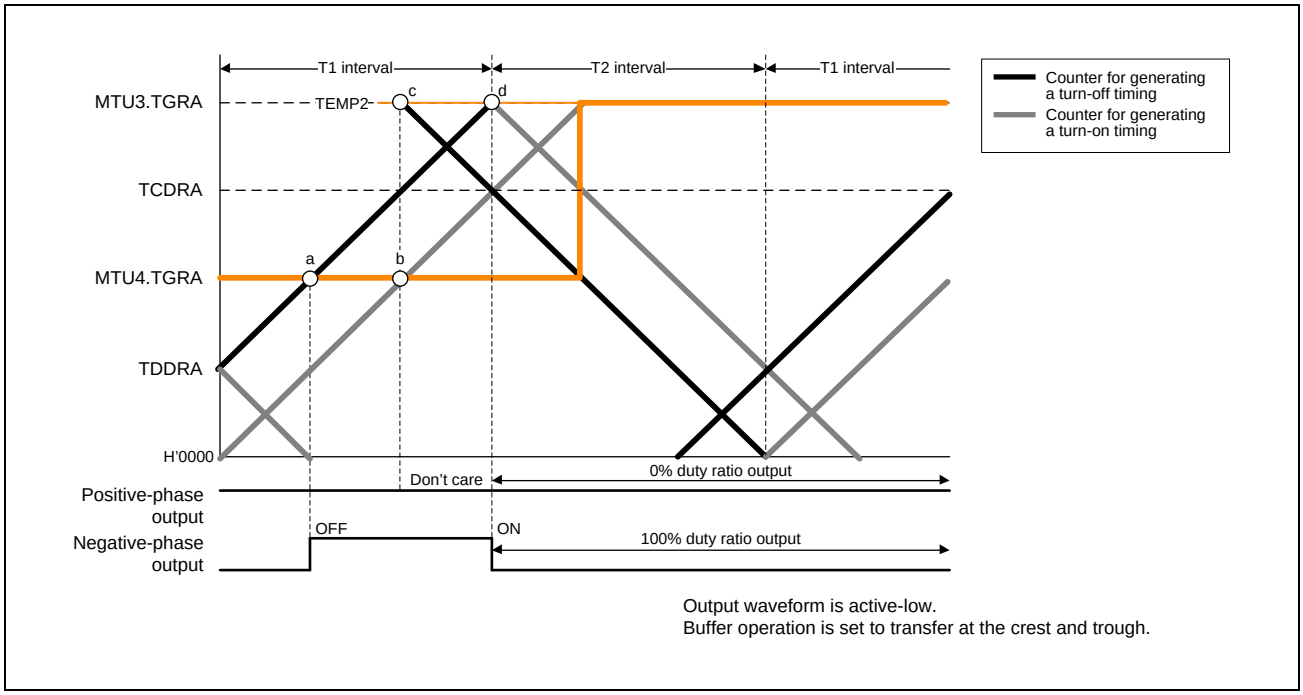


Figure 16.61 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (3)

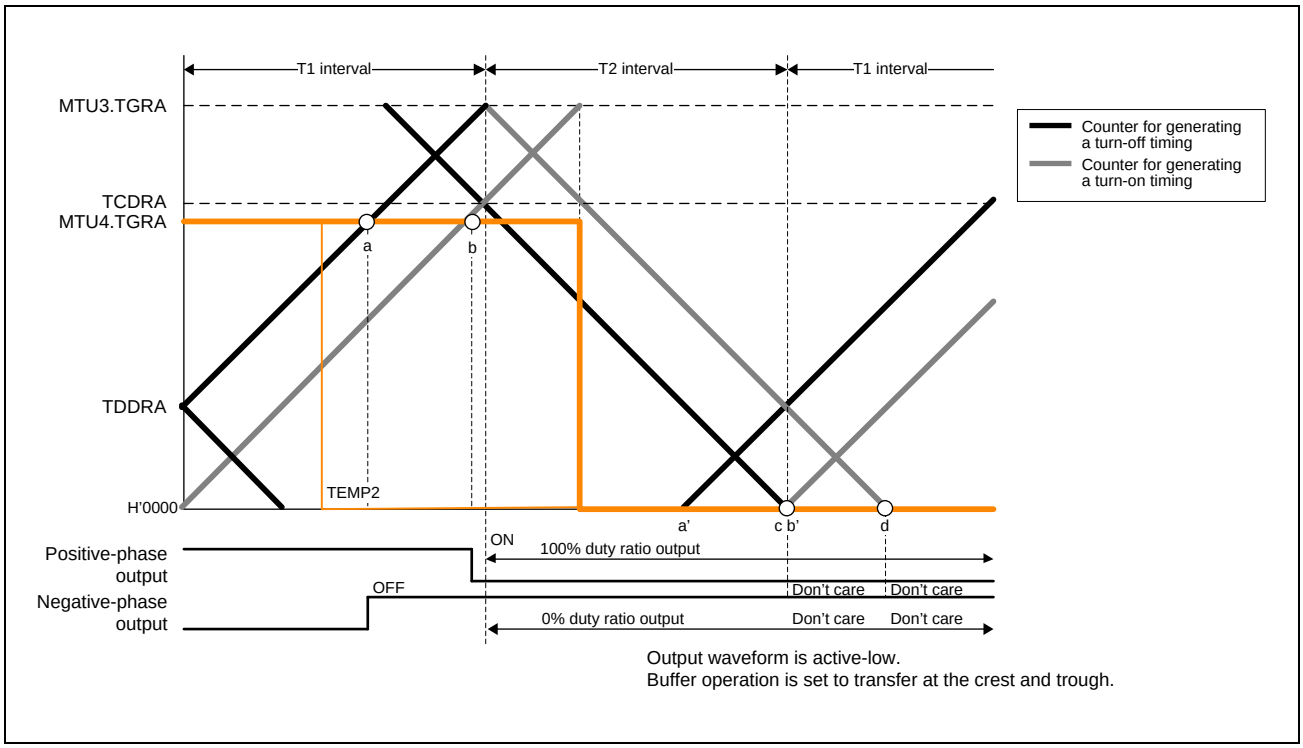


Figure 16.62 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (4)

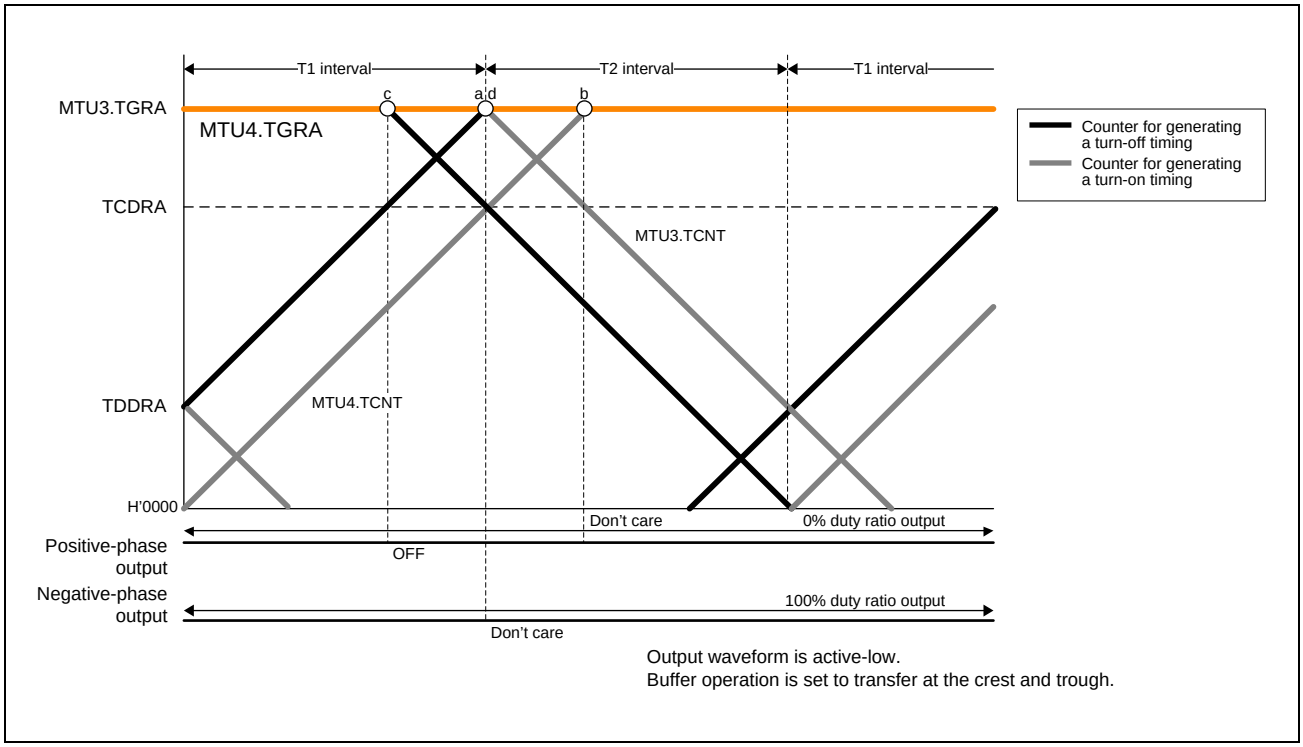


Figure 16.63 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (5)

(l) Toggle Output Synchronized with PWM Cycle

In complementary PWM mode, toggle output from the PWM output pin in synchronization with the PWM cycle can be enabled by setting the PSYE bit in the TOCR1A (TOCR1B) register to 1. An example of a toggle output waveform is shown in **Figure 16.64**.

This output is toggled by a compare match between MTU3.TCNT and MTU3.TGRA (MTU6.TCNT and MTU6.TGRA) and a compare match between MTU4.TCNT (MTU7.TCNT) and H’0000.

The MTIOC3A (MTIOC6A) pin is assigned for this toggle output. The initial output is high-level output.

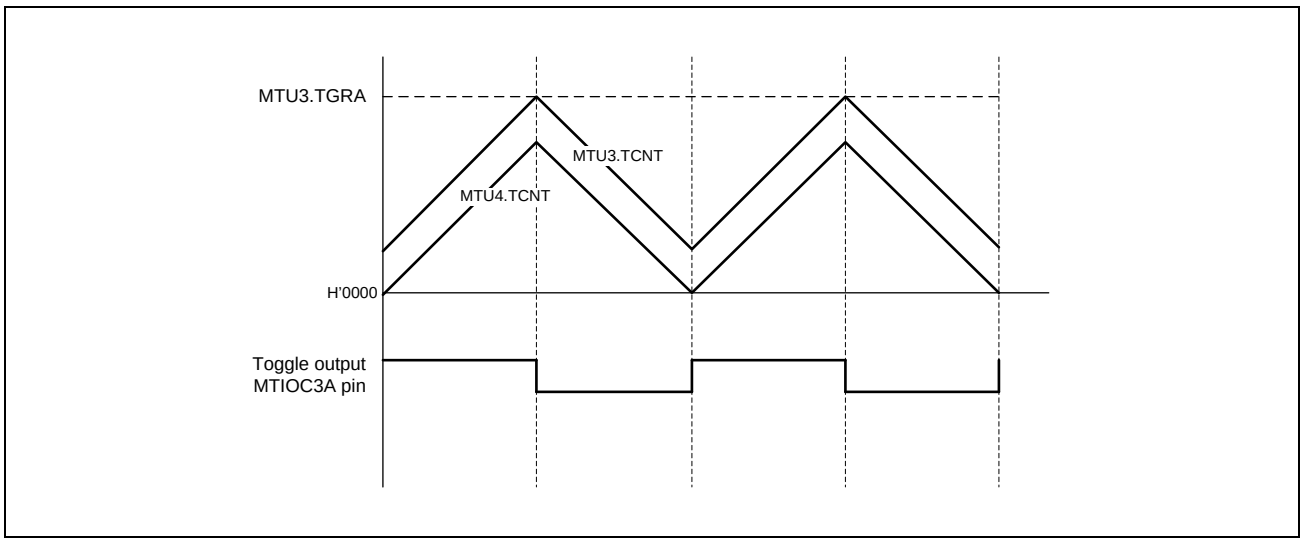


Figure 16.64 Example of Toggle Output Waveform Synchronized with PWM Output (MTU3 and MTU4)

(m) Counter Clearing by Another Channel

In complementary PWM mode, MTU3.TCNT, MTU4.TCNT, and TCNTSA (MTU6.TCNT, MTU7.TCNT, and TCNTSB) can be cleared by another channel source when a mode for synchronization with another channel is specified through the TSYRA (TSYRB) register and synchronous clearing is selected with MTU3.TCR.CCLR[2:0] (MTU6.TCR.CCLR[2:0]) bits.

Figure 16.65 illustrates an example of this operation.

Use of this function enables a counter to be cleared and restarted through an external signal.

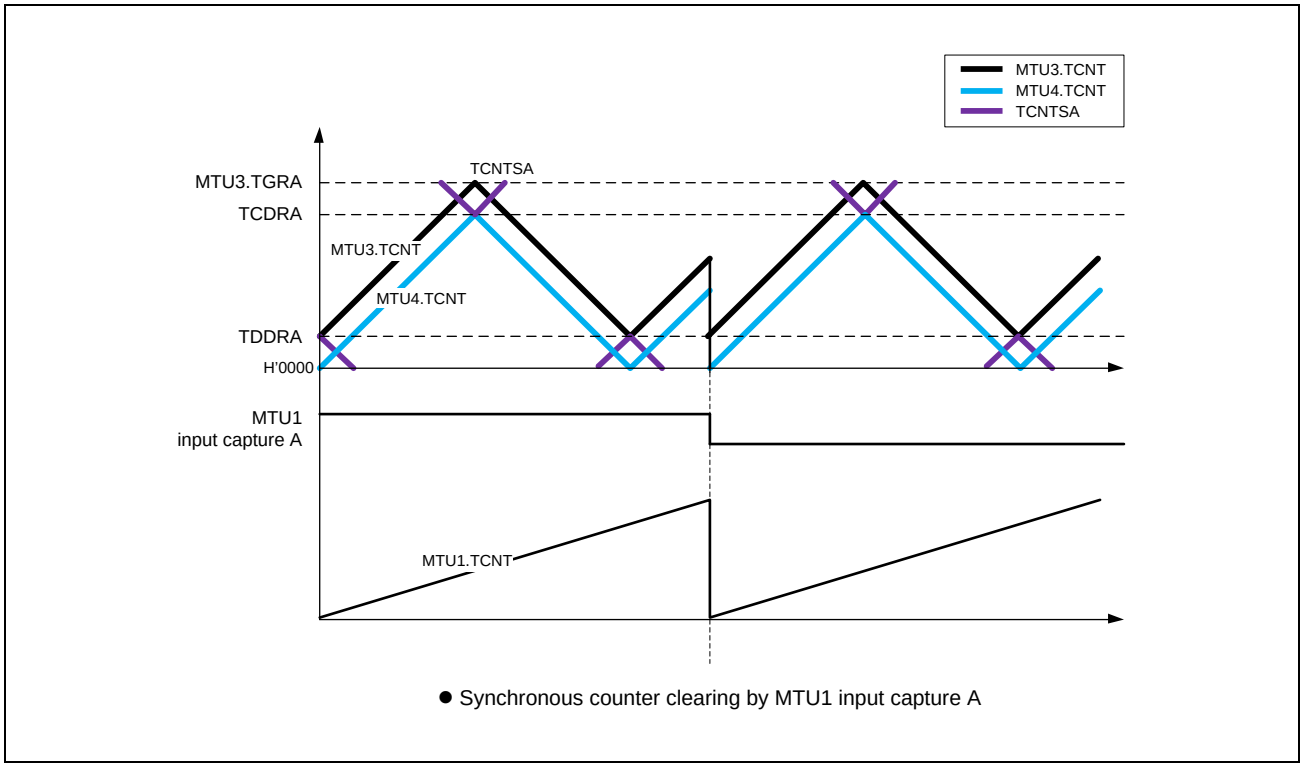


Figure 16.65 Counter Clearing Synchronized with Another Channel (MTU3 and MTU4)

(n) Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

Setting the WRE bit in TWCRA (TWCRB) to 1 suppresses initial output when synchronous counter clearing occurs in the Tb interval (Tb2 interval) at the trough in complementary PWM mode and controls abrupt change in duty cycle at synchronous counter clearing.

Initial output suppression through the WRE bit = 1 is applicable only when synchronous clearing occurs in the Tb2 interval as indicated by (10) or (11) in **Figure 16.66**. When synchronous clearing occurs outside that interval, the initial value specified by the OLSN and OLSP bits in TOCR1A (TOCR1B) is output. Even in the Tb2 interval, if synchronous clearing occurs in the initial value output period (indicated by (1) in **Figure 16.66**) immediately after the counters start operation, initial value output is not suppressed.

This function can be used in both channel combinations of MTU 3 and MTU4 and MTU6 and MTU7. In MTU3 and MTU4, synchronous clearing in MTU0, MTU1, and MTU2 can cause counter clearing; in MTU6 and MTU7, compare match or input capture generated in MTU0, MTU1, and MTU2 can cause counter clearing.

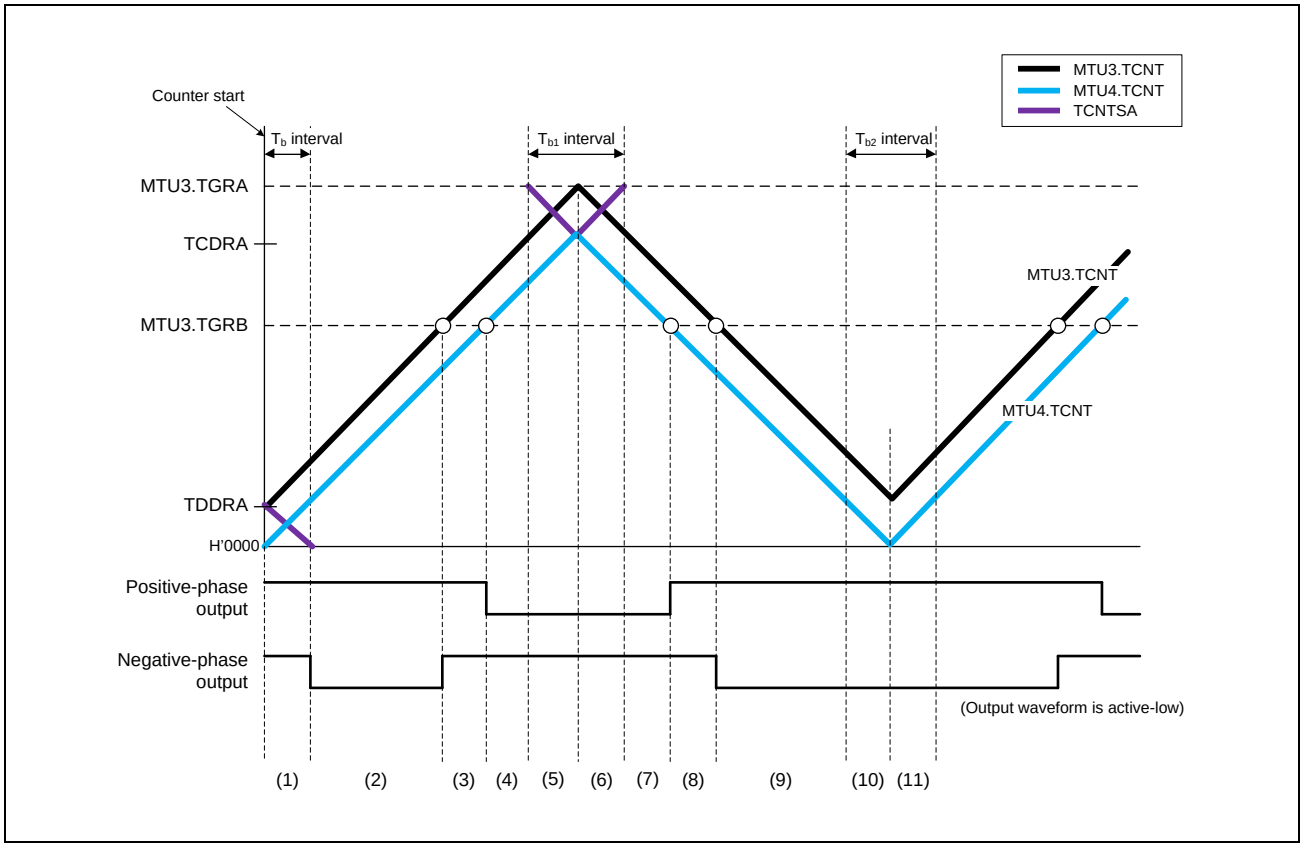


Figure 16.66 Timing for Synchronous Counter Clearing (MTU3 and MTU4)

- Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode.

  An example of the procedure for setting output waveform control at synchronous counter clearing in complementary PWM mode is shown in **Figure 16.67**.

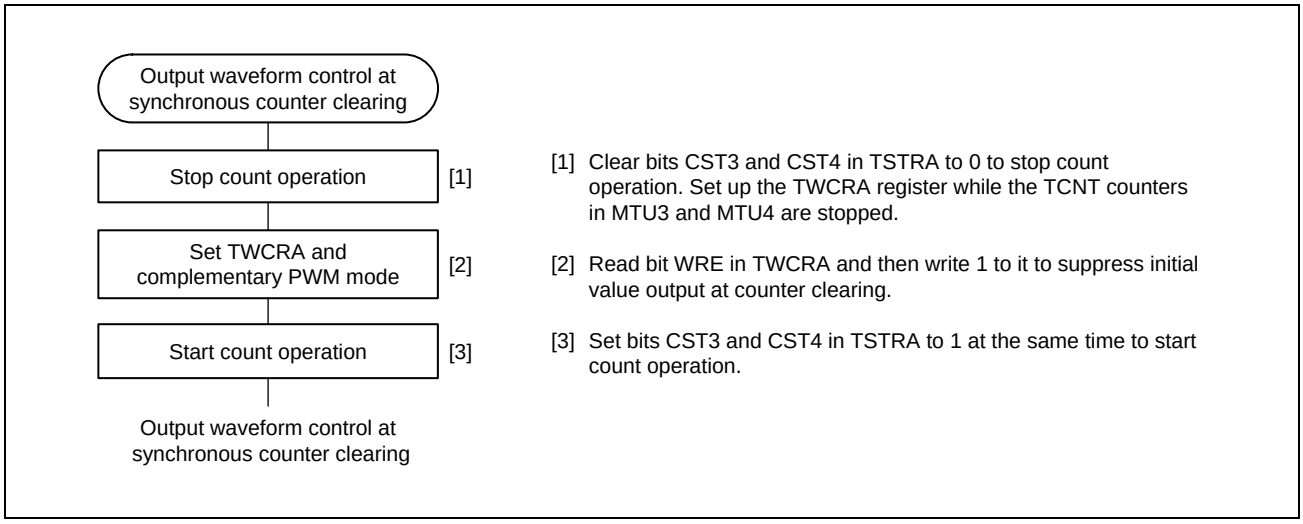


Figure 16.67 Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode (MTU3 and MTU4)

- Examples of Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

Figure 16.68 to **Figure 16.71** show examples of output waveform control in which MTU3 and MTU4 operate in complementary PWM mode and synchronous counter clearing is generated while the WRE bit in TWCRA is set to 1. In the examples shown in **Figure 16.68** to **Figure 16.71**, synchronous counter clearing occurs at timing (3), (6), (8), and (11) shown in **Figure 16.66**, respectively.
In MTU6 and MTU7, these examples are equivalent to the cases when MTU6 and MTU7 operate in complementary PWM mode and synchronous counter clearing is generated while the SCC bit is cleared to 0 and the WRE bit is set to 1 in TWCRB.

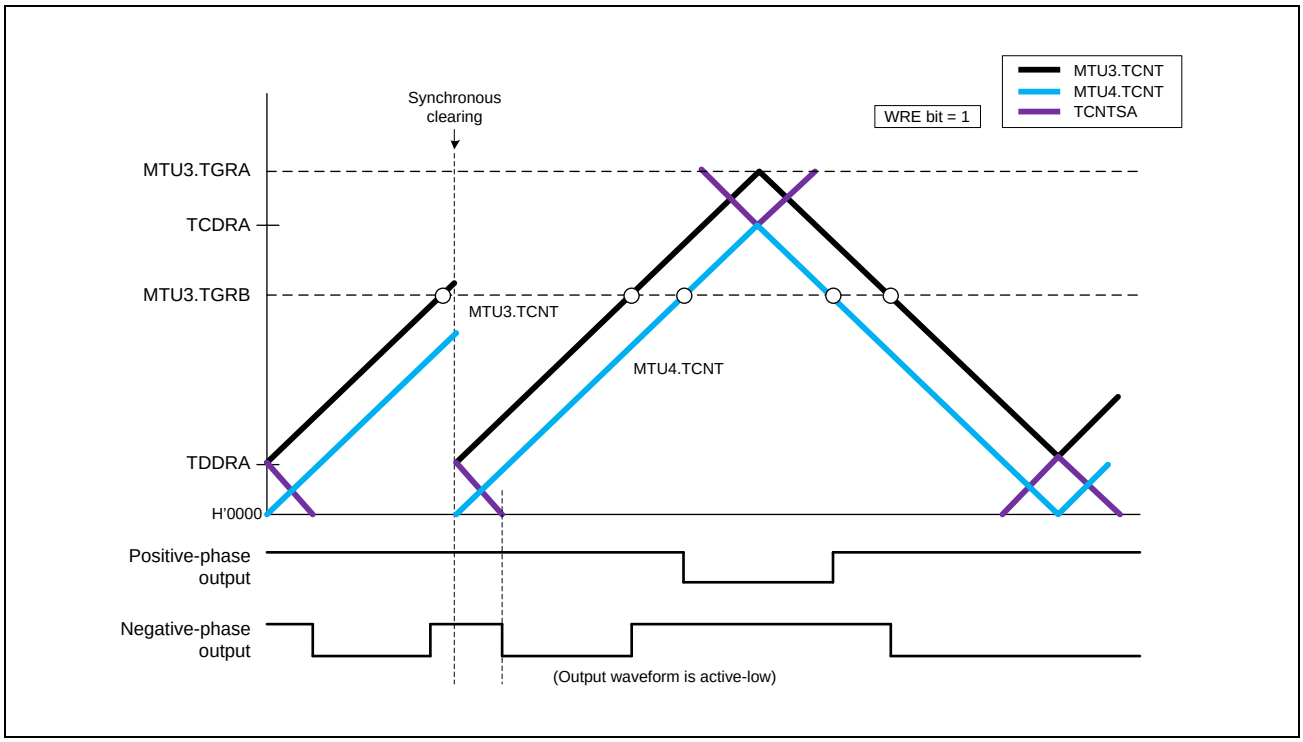


Figure 16.68 Example of Synchronous Clearing in Dead Time during Up-Counting
(Timing (3) in **Figure 16.66**; TWCRA.WRE Bit is 1)

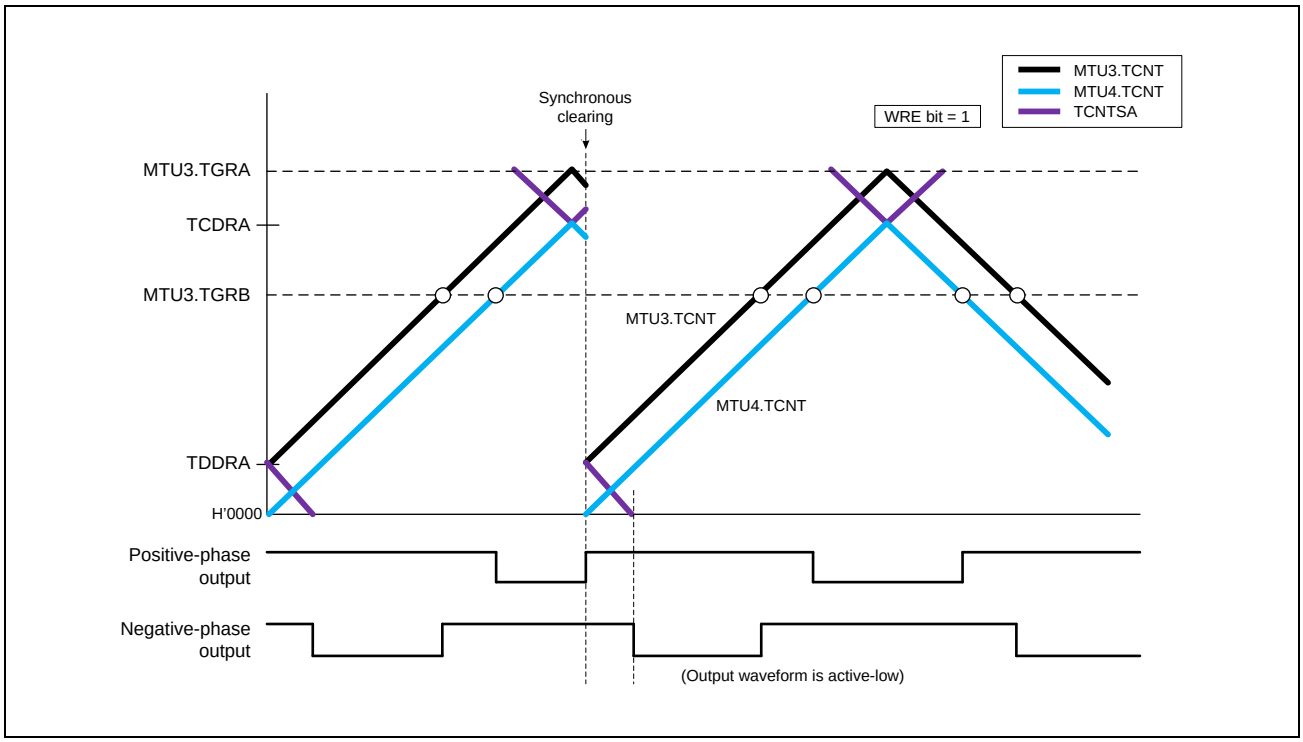


Figure 16.69 Example of Synchronous Clearing in Tb1 interval
(Timing (6) in **Figure 16.66**; TWCRA.WRE Bit is 1)

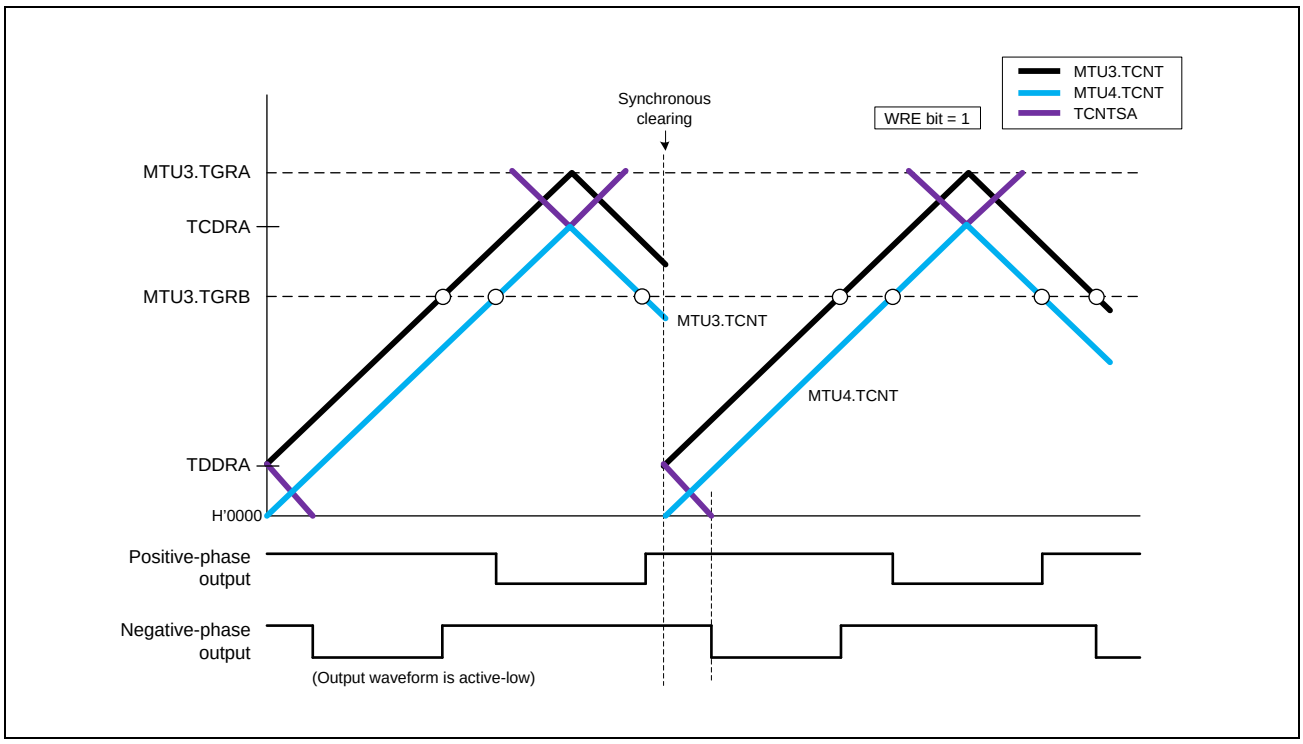


Figure 16.70 Example of Synchronous Clearing in Dead Time during Down-Counting
(Timing (8) in **Figure 16.66**; TWCRA.WRE Bit is 1)

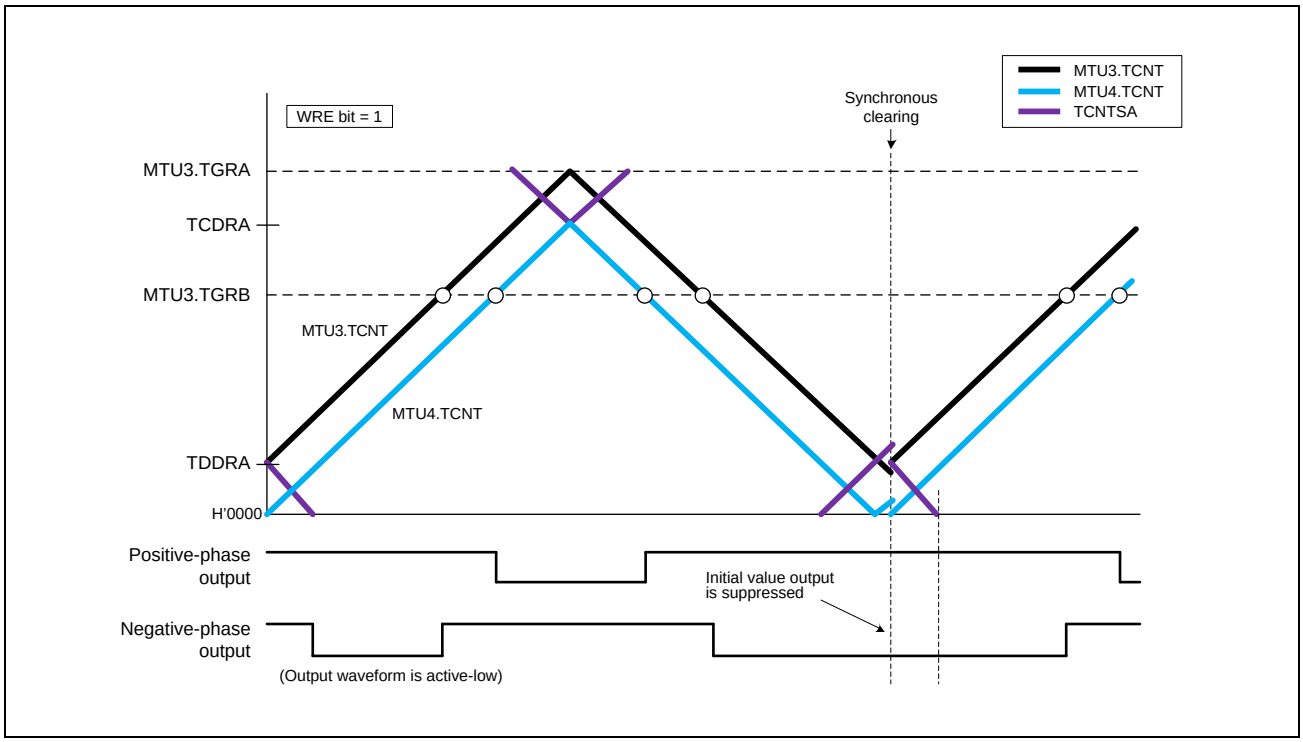


Figure 16.71 Example of Synchronous Clearing in Tb2 interval
(Timing (11) in **Figure 16.66**; TWCRA.WRE Bit is 1)

(o) Suppressing Synchronous Counter Clearing for MTU0 to MTU2 and MTU6 and MTU7

In MTU6 and MTU7, setting the SCC bit in TWCRB to 1 suppresses synchronous counter clearing caused by MTU0 to MTU2.

Synchronous counter clearing is suppressed only within the interval shown in **Figure 16.72**. When using this function, MTU6 and MTU7 should be set to complementary PWM mode.

For details of synchronous clearing caused by MTU0 to MTU2, refer to **Section 16.3.10(2), Synchronous Counter Clearing for MTU6 and MTU7**.

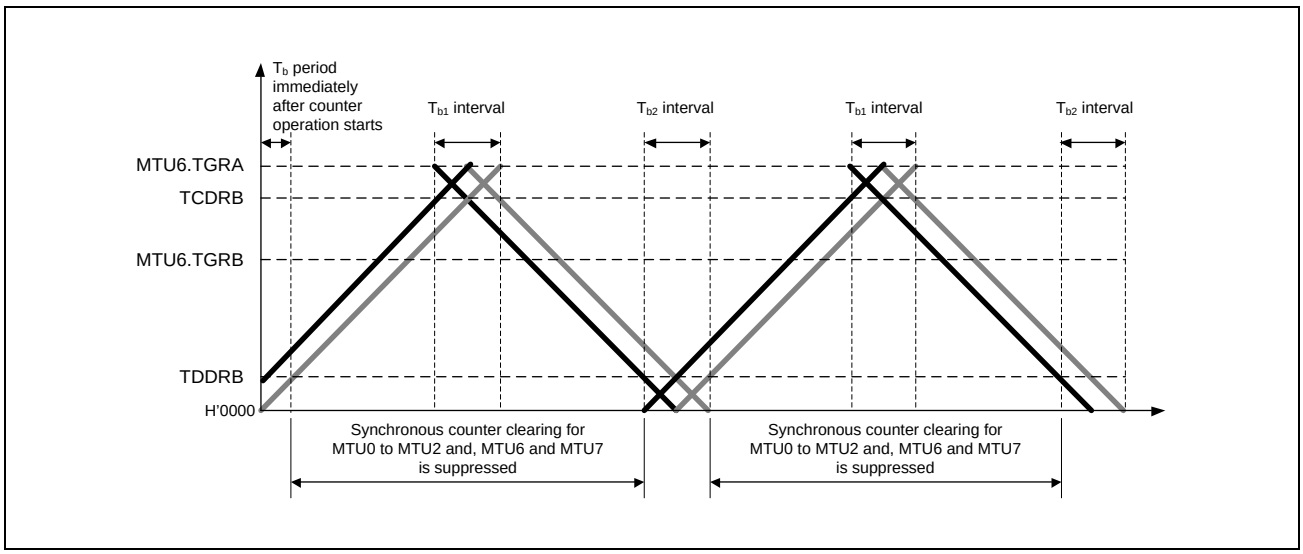
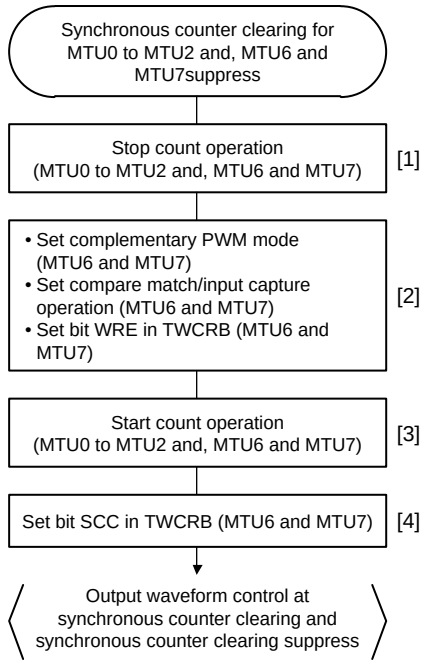


Figure 16.72 Synchronous Clearing-Suppressed Interval Specified by TWCRB.SCC Bit for MTU0, MTU1 and MTU2, and MTU6 and MTU7

- Example of Procedure for Suppressing Synchronous Counter Clearing for MTU0 to MTU2, and MTU6 and MTU7

An example of the procedure for suppressing synchronous counter clearing for MTU0 to MTU2, and MTU6 and MTU7 is shown in **Figure 16.73**.



[1] Clear bits CSTn of the timer start register (TSTRB) in MTU6 and MTU7 to 0 to stop count operation. Clear bits CSTn of TSTRA in MTU0 to MTU2 to 0 to stop count operation.

[2] Set the complementary PWM mode in MTU6 and MTU7 and compare match/input capture operation in MTU0 to MTU2. When bit WRE in TWCRA or TWCRB should be set, make appropriate setting here.

[3] Set bits CST6 and CST7 in MTU6 and MTU7 to 1 to start count operation. For synchronous counter clearing for MTU0 to MTU2 and MTU6 and MTU7, set bits CSTn of TSTRA in MTU0 to MTU2 to 1 to start count operation in any one of MTU0.TCNT to MTU2.TCNT.

[4] Read TWCRB and then set bit SCC in TWCRB to 1 to suppress synchronous counter clearing for MTU0 to MTU2 and MTU6 and MTU7*1. Here, do not modify the CCE and WRE bit values in TWCRB of MTU6 and MTU7. Synchronous counter clearing for MTU0 to MTU2 and MTU6 and MTU7 is suppressed in the synchronous counter clearing-suppressed periods.

Note 1. The TWCRB.SCC bit value can be modified during count operation. However, if a synchronous clearing occurs when bit TWCRB.SCC is modified from 0 to 1, the synchronous clearing may not be suppressed. If a synchronous clearing occurs when bit TWCRB.SCC is modified from 1 to 0, the synchronous clearing may be suppressed.

Figure 16.73 Example of Procedure for Suppressing Synchronous Counter Clearing for MTU0 to MTU2, and MTU6 and MTU7

- Examples of Suppression of Synchronous Counter Clearing for MTU 0 to MTU2, and MTU6 and MTU7

Figure 16.74 to **Figure 16.77** show examples of operation in which MTU6 and MTU7 operate in complementary PWM mode and synchronous counter clearing for MTU 0 to MTU2, and MTU6 and MTU7 is suppressed by setting the SCC bit in TWCRB in MTU6 and MTU7 to 1. In the examples shown in **Figure 16.74** to **Figure 16.77**, synchronous counter clearing occurs at timing (3), (6), (8), and (11) shown in **Figure 16.66**, respectively. In these examples, the WRE bit in TWCRB in MTU6 and MTU7 is set to 1.

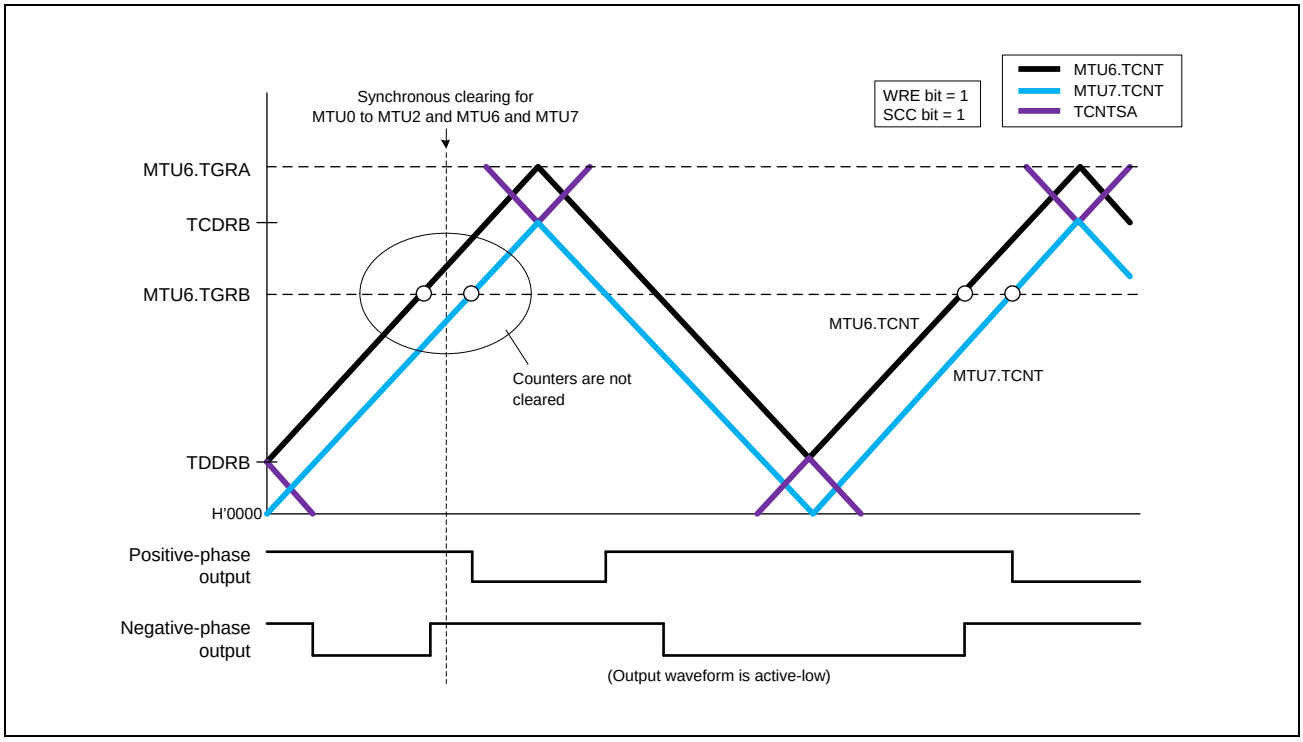


Figure 16.74 Example of Synchronous Clearing in Dead Time during Up-Counting
(Timing (3) in **Figure 16.66**; TWCRB.WRE Bit is 1 and SCC Bit is 1 in MTU6 and MTU7)

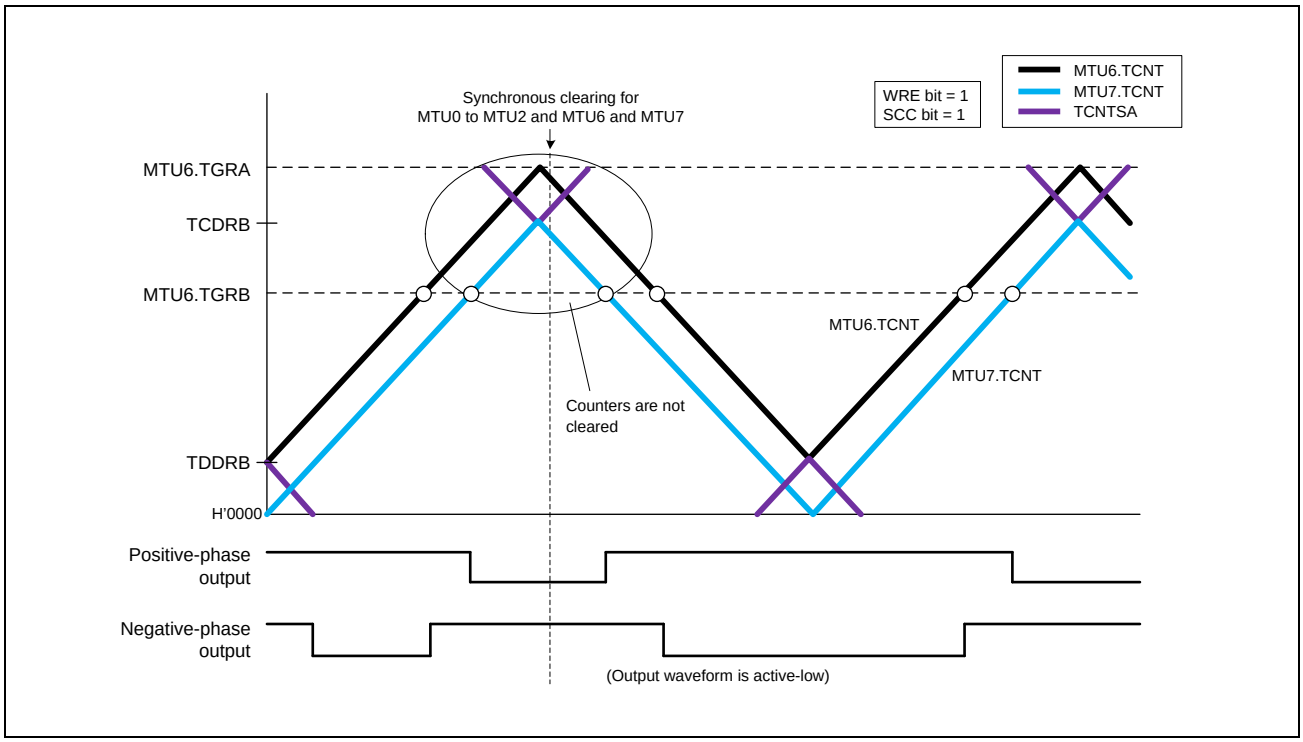


Figure 16.75 Example of Synchronous Clearing in Tb1 interval
(Timing (6) in **Figure 16.66**; TWCRB.WRE Bit is 1 and SCC Bit is 1 in MTU6 and MTU7)

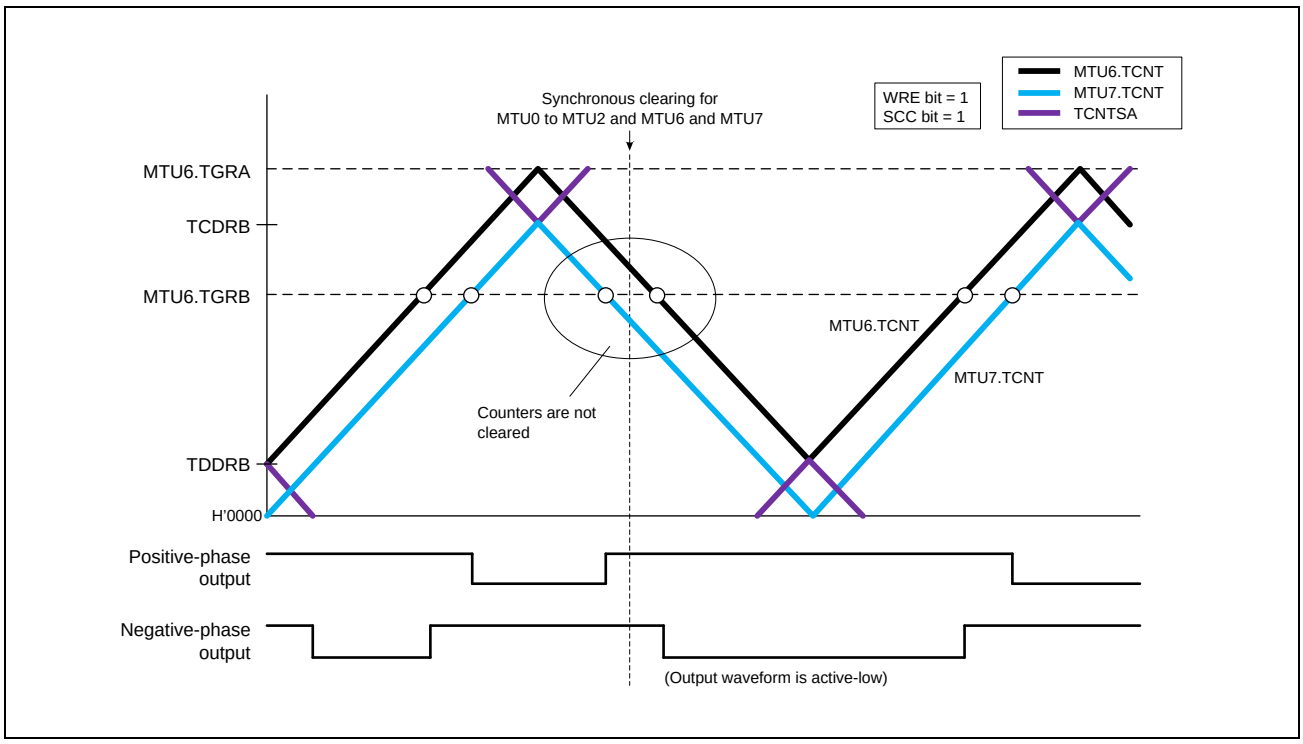


Figure 16.76 Example of Synchronous Clearing in Dead Time during Down-Counting
(Timing (8) in **Figure 16.66**; TWCRB.WRE Bit is 1 and SCC Bit is 1 in MTU6 and MTU7)

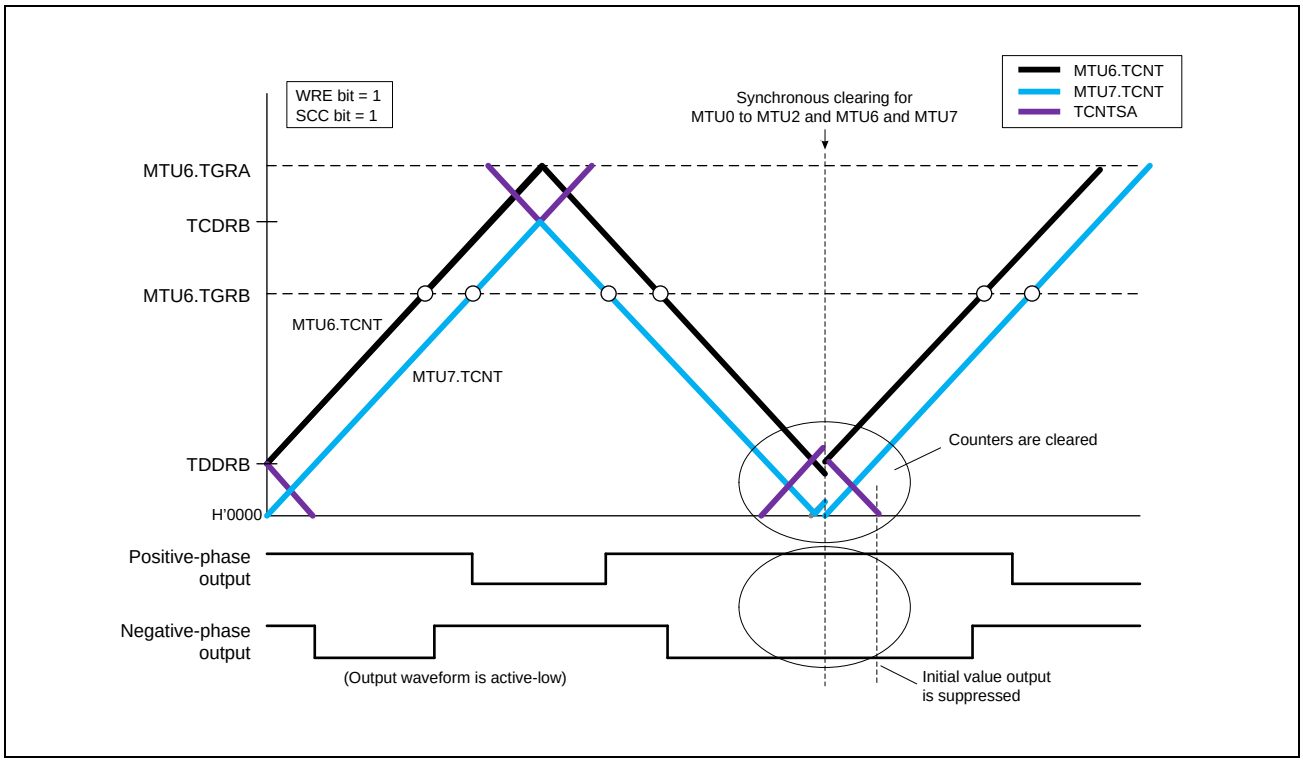


Figure 16.77 Example of Synchronous Clearing in Tb2 interval
(Timing (11) in **Figure 16.66**; TWCRB.WRE Bit is 1 and SCC Bit is 1 in MTU6 and MTU7)

(p) Counter Clearing by MTU3.TGRA (MTU6.TGRA) Compare Match

In complementary PWM mode, MTU3.TCNT, MTU4.TCNT, and TCNTSA (MTU6.TCNT, MTU7.TCNT, and TCNTSB) can be cleared by MTU3.TGRA (MTU6.TGRA) compare match when the TWCRA.CCE (TWCRB.CCE) bit.

Figure 16.78 illustrates an operation example.

NOTES

1. Use this function only in complementary PWM mode 1 (transfer at crest).
2. Do not specify synchronous clearing by another channel (do not set 1 in the SYNC0 to SYNC4 or SYNC6 to SYNC7 bits in the timer synchronous register (TSYRA or TSYRB) and the CEOA to CEOD or CEIA to CEID bits in TSYCR).
3. Do not set the PWM duty value to H'0000.
4. Do not set the PSYE bit in timer output control register 1 (TOCR1A or TOCR1B) to 1.

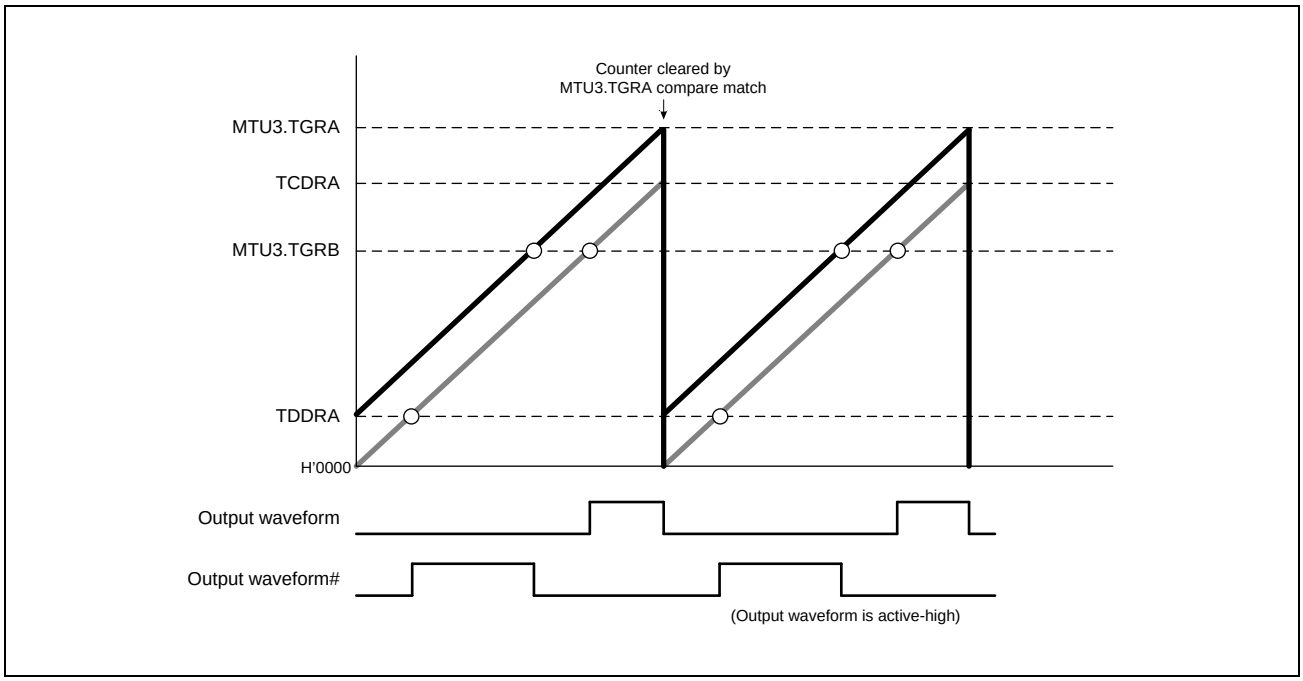


Figure 16.78 Example of Counter Clearing Operation by MTU3.TGRA Compare Match

(q) Example of Waveform Output for Driving AC Synchronous Motor (Brushless DC Motor)

In complementary PWM mode when MTU3 and MTU4 are used, a brushless DC motor can easily be controlled using TGCRA. **Figure 16.79** to **Figure 16.82** show examples of brushless DC motor driving waveforms created using TGCRA.

To switch the output phases for a 3-phase brushless DC motor by means of external signals detected with a Hall element, etc., clear the TGCRA.FB bit to 0. In this case, the external signals indicating the magnetic pole position should be input to pins MTIOC0A, MTIOC0B, and MTIOC0C in MTU0 (make appropriate settings with the port mode registers (PMn) and Pmn port function control register (PFCm) of the GPIO ports). When an edge is detected at pin MTIOC0A, MTIOC0B, or MTIOC0C, the output on/off state is switched automatically.

When the TGCRA.FB bit is 1, the output on/off state is switched when the UF, VF, or WF bit in TGCRA is cleared to 0 or set to 1.

The driving waveforms are output from the 6-phase PWM output pins for complementary PWM mode.

With this 6-phase output, while the output is turned on, chopping output is available through complementary PWM mode output function by setting the N bit or P bit in TGCRA to 1. When the N bit or P bit is 0, the level output is selected.

The active level of the 6-phase output (on output level) can be set with the TOCR1A.OLSN and TOCR1A.OLSP bits regardless of the setting of the N and P bits.

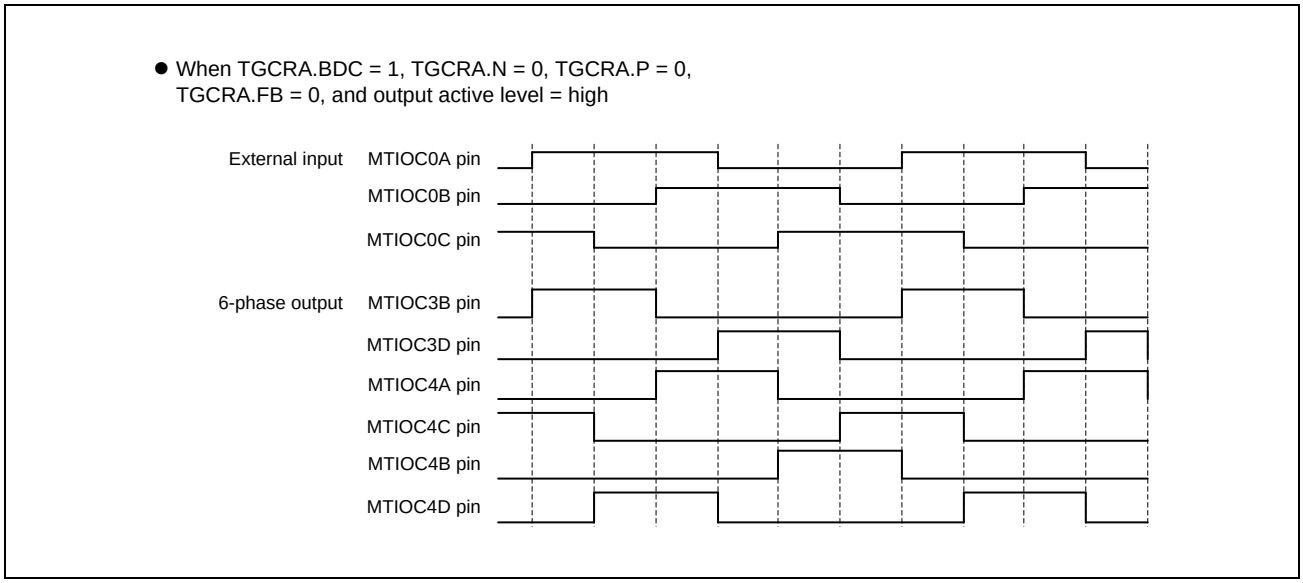


Figure 16.79 Example of Output Phase Switching by External Input (1)

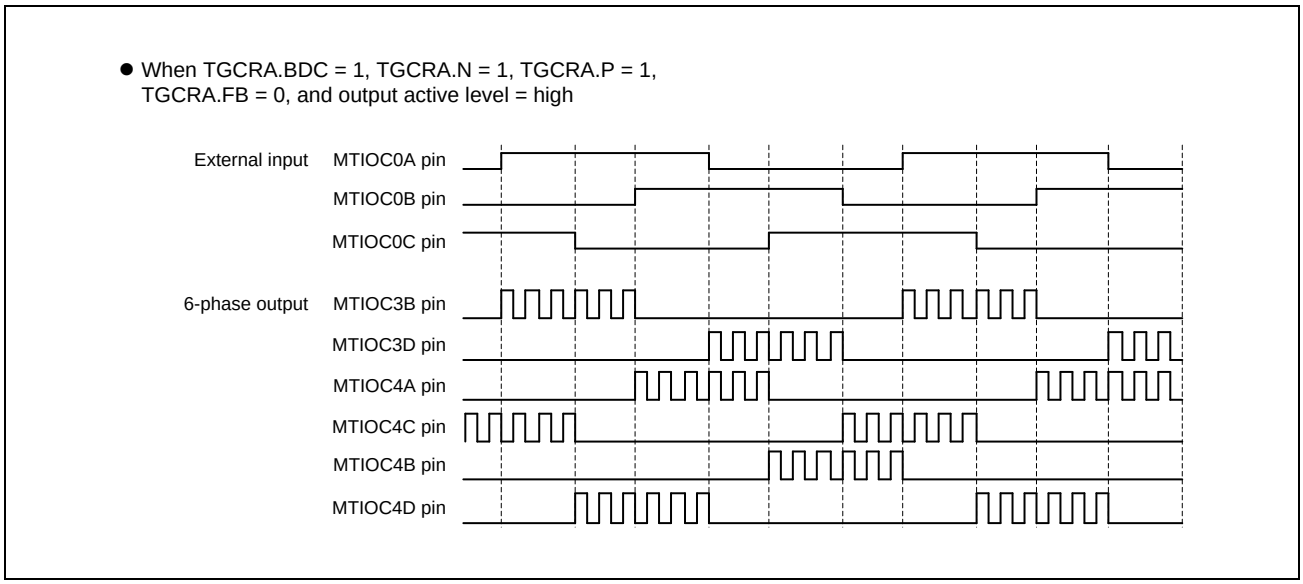


Figure 16.80 Example of Output Phase Switching by External Input (2)

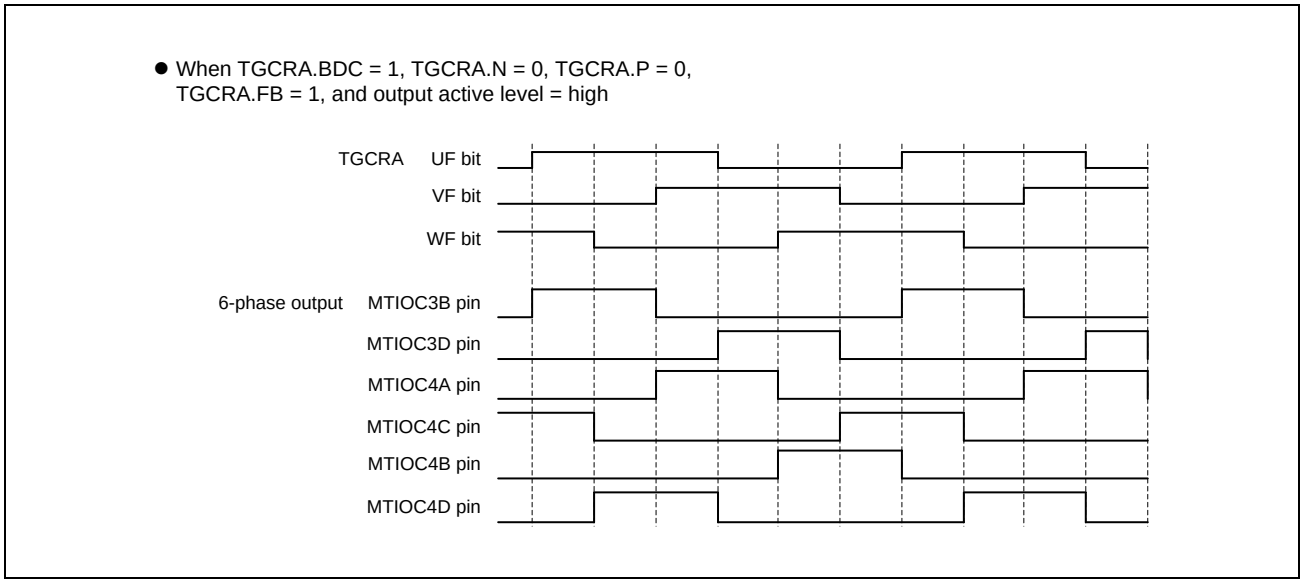


Figure 16.81 Example of Output Phase Switching through UF, VF, and WF Bit Settings (1)

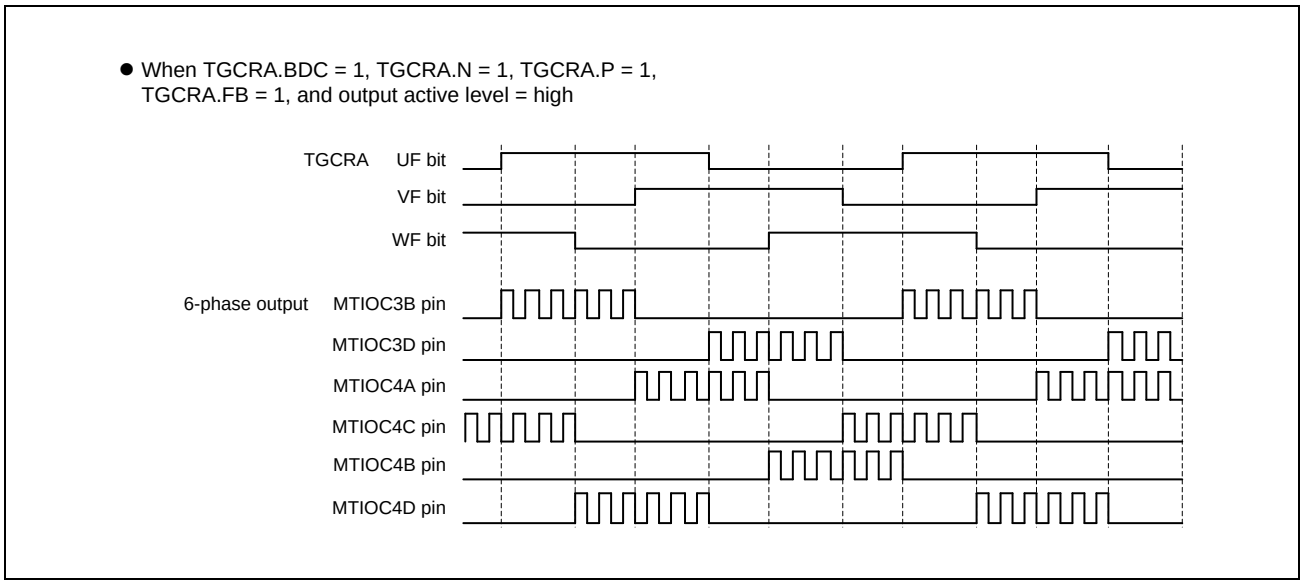


Figure 16.82 Example of Output Phase Switching through UF, VF, and WF Bit Settings (2)

(r) A/D Converter Start Request Setting

In complementary PWM mode, an A/D converter start request can be issued using MTU3.TGRA (MTU6.TGRA) compare match, MTU4.TCNT (MTU7.TCNT) underflow (trough), or compare match on a channel other than MTU3 and MTU4 (MTU6 and MTU7).

When start requests using MTU3.TGRA (MTU6.TGRA) compare match are specified, A/D conversion can be started at the crest of the MTU3.TCNT (MTU6.TCNT) count.

A/D converter start requests can be specified by setting the TIER.TTGE bit. To issue an A/D converter start request at an MTU4.TCNT (MTU7.TCNT) underflow (trough), set the MTU4.TIER.TTGE2 (MTU7.TIER.TTGE2) bit to 1.

(s) Double Buffer Function in Complementary PWM Mode

In complementary PWM mode 3 (transfer at the crest and trough), the PWM output setting resolution can be improved from ±2 to ±1 by setting the TMDR2A.DRS (TMDR2B.DRS) bit to 1.

When setting buffer registers A (TGRD in MTU3, TGRC and TGRD in MTU4, TGRD in MTU6, and TGRC and TGRD in MTU7), set also buffer registers B (TGRE in MTU3, TGRE and TGRF in MTU4, TGRE in MTU6, and TGRE and TGRF in MTU7) at the same time. Each buffer register B should be set to the buffer register A value or (buffer register A value – 1). For details of the setting procedure, refer to **Section 16.3.8(1), Example of Complementary PWM Mode Setting Procedure**.

NOTE

When a buffer register B is set to the buffer register A value, symmetric PWM waveforms are output. When a buffer register B is set to (buffer register A value – 1), asymmetric PWM waveforms are output.

Figure 16.83 shows an example of double buffer operation. Each register data is transferred as follows.

- After TGRD (buffer A) in MTU4 or MTU7 is written to, data is transferred from TGRD (buffer A) in MTU4 or MTU7 to Temp3A or Temp6A (temporary A) and from TGRF (buffer B) in MTU4 or MTU7 to Temp3B or Temp6B (temporary B).
- With timing (1) in the figure, data is transferred from Temp3A or Temp6A (temporary A) to TGRB (compare) in MTU4 or MTU7.
- With timing (2) in the figure, data is transferred from Temp3B or Temp6B (temporary B) to TGRB (compare) in MTU4 or MTU7.

In the crest interval (Tb1 interval), the compare register and temporary register A are valid; in the trough interval (Tb2 interval), the compare register and temporary register B are valid.

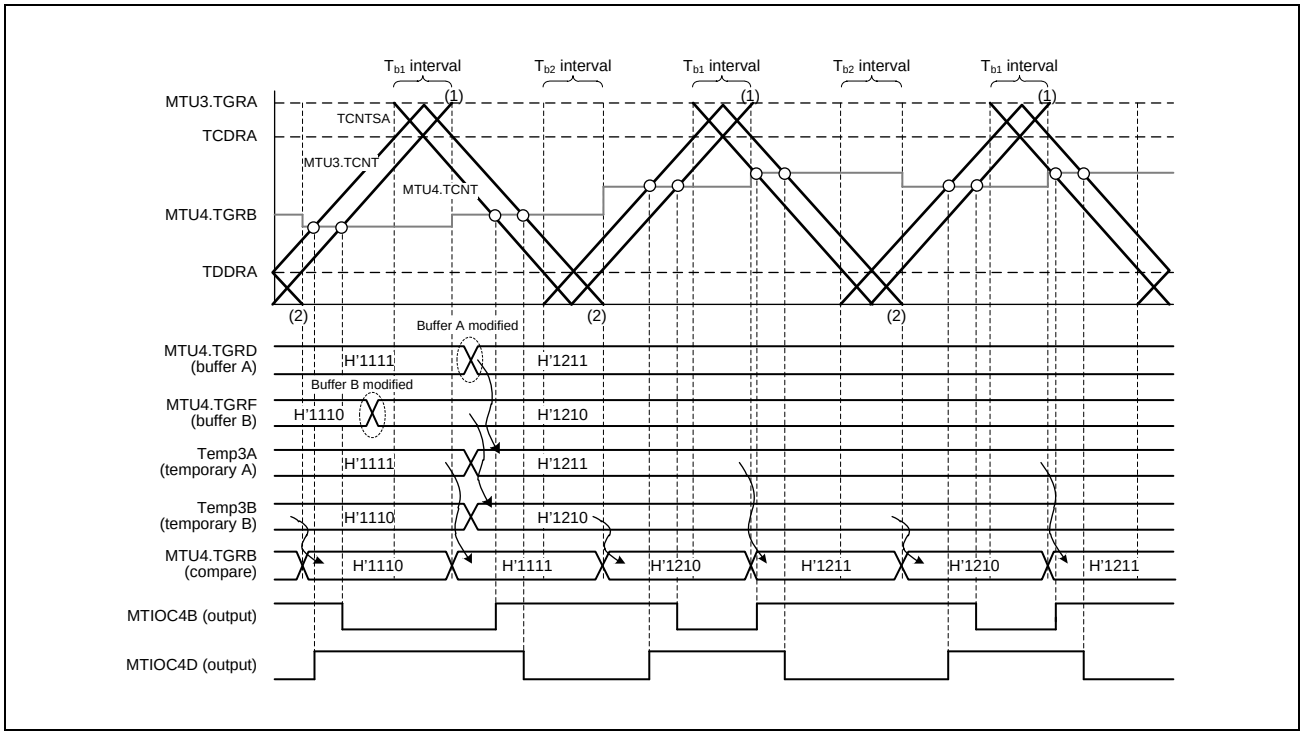


Figure 16.83 Example of Double Buffer Operation

Figure 16.84 shows an example when the buffer write value is smaller than the TDDRA (TDDRB) value, and **Figure 16.85** shows an example when the write value is greater than TCDRA (TCDRB).

In the crest interval, the output is controlled according to the compare match with the compare register or temporary register A; in the trough interval, the output is controlled according to the compare match with the compare register or temporary register B.

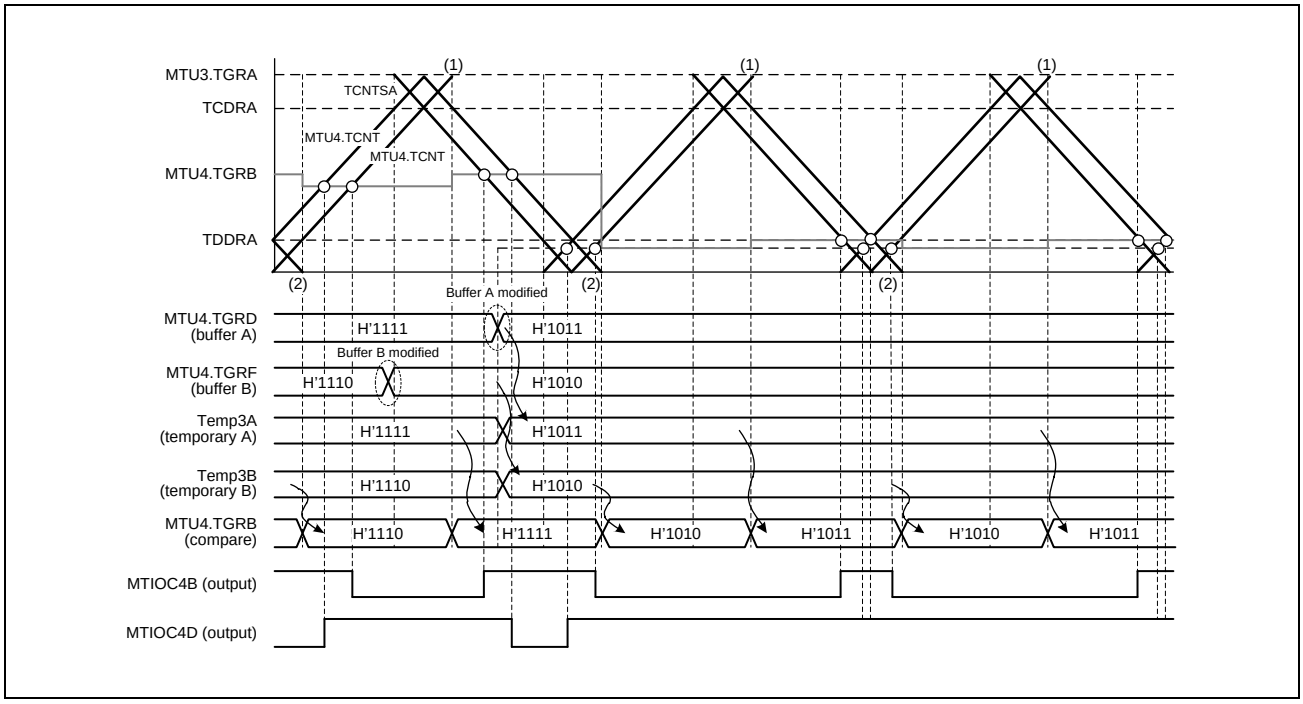


Figure 16.84 Example of Double Buffer Operation (Buffer Write Value is Smaller than TDDRA)

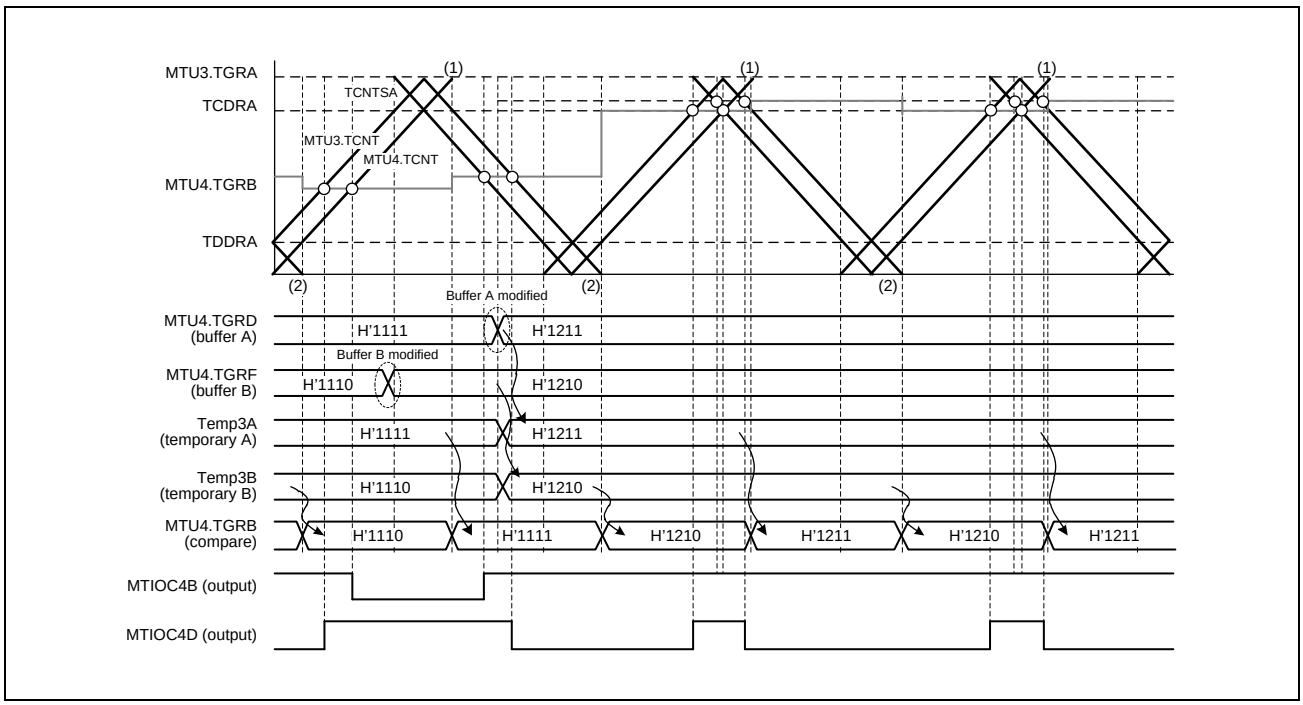


Figure 16.85 Example of Double Buffer Operation (Buffer Write Value is Greater than TCDRA)

(3) Interrupt Skipping Function 1 in Complementary PWM Mode

Interrupts TGIA3 (TGIA6) (at the crest) and TCIV4 (TCIV7) (at the trough) in MTU3 and MTU4 (MTU6 and MTU7) can be skipped up to seven times by making settings in the TITCR1A (TITCR1B) register.

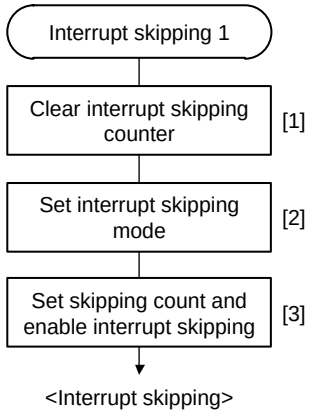
Transfers from a buffer register to a temporary register or a compare register can be skipped in coordination with interrupt skipping by making settings in the TBTERA (TBTERB) register. For the linkage with buffer registers, refer to description (c), Buffer Transfer Control Linked with Interrupt Skipping, below.

A/D converter start requests generated by the A/D converter start request delaying function can also be skipped in coordination with interrupt skipping by making settings in the MTU4.TADCR (MTU7.TADCR) register. For the linkage with the A/D converter start request delaying function, refer to **Section 16.3.9, A/D Converter Start Request Delaying Function**.

The TITCR1A (TITCR1B) register should be set while interrupt skipping function 1 is selected by setting the TITM bit in the timer interrupt skipping mode register (TITMRA or TITMRB) to 0, TGIA3 (TGIA6) interrupt requests are disabled by setting the MTU3.TIER (MTU6.TIER) register, TCIV4 (TCIV7) interrupt requests are disabled by setting the MTU4.TIER (MTU7.TIER) register, and a compare match is not generated. Before changing the skipping count, be sure to clear the T3AEN (T6AEN) and T4VEN (T7VEN) bits to 0 to clear the skipping counter.

(a) Example of Interrupt Skipping Function 1 Setting Procedure

Figure 16.86 shows an example of the interrupt skipping function 1 setting procedure. **Figure 16.87** shows the periods during which interrupt skipping count can be changed.



[1] Set bits T3AEN and T4VEN in the timer interrupt skipping set register 1 (TITCR1A) to 0 to clear the skipping counter.

[2] Set bit TITM in the timer interrupt skipping mode register (TITMRA) to 0 to select interrupt skipping function 1.

[3] Specify the interrupt skipping count within the range from 0 to 7 times in bits T3ACOR2 to T3ACOR0 and T4VCOR2 to T4VCOR0 in TITCR1A, and enable interrupt skipping through bits T3AEN and T4VEN.

Note: The setting of TITCR1A must be done while the TGIA3 and TCIV4 interrupt requests are disabled by the settings of MTU3.TIER and MTU4.TIER under the conditions in which a compare match is not generated. Before changing the skipping count, be sure to clear the T3AEN and T4VEN bits to 0 to clear the skipping counter.

Figure 16.86 Example of Interrupt Skipping Function 1 Setting Procedure

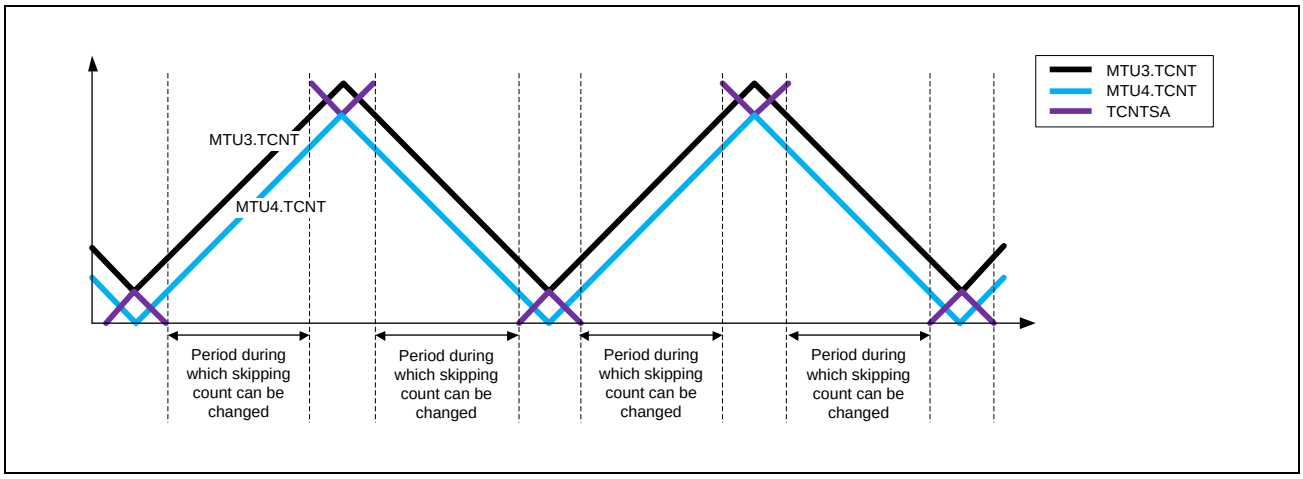


Figure 16.87 Periods during which Interrupt Skipping Count can be Changed

(b) Example of Interrupt Skipping Function 1

Figure 16.88 shows an example of TGIA3 (TGIA6) interrupt skipping in which the interrupt skipping count is set to three by the T3ACOR (T6ACOR) bits and the T3AEN (T6AEN) bit is set to 1 in the TITCR1A (TITCR1B) register.

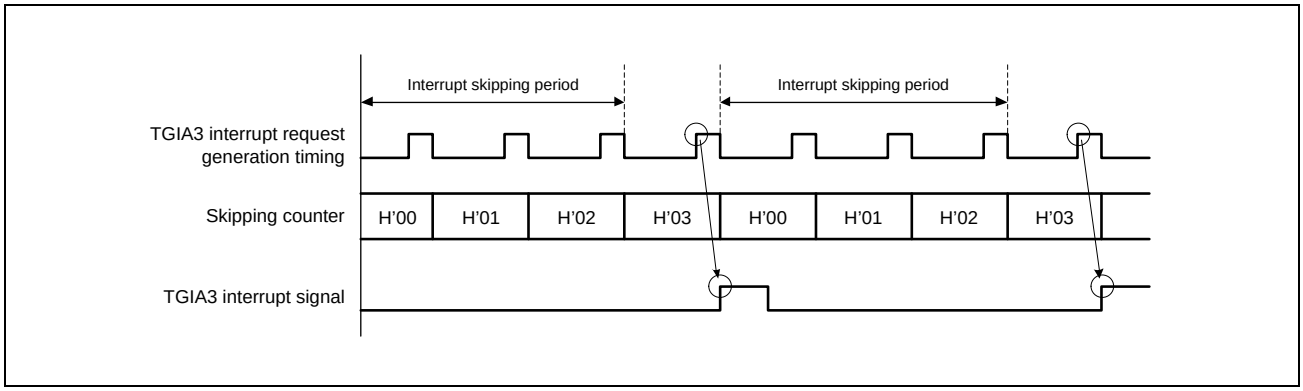


Figure 16.88 Example of Interrupt Skipping Function 1

(c) Buffer Transfer Control Linked with Interrupt Skipping

In complementary PWM mode, whether to transfer data from a buffer register to a temporary register and whether to link the transfer with interrupt skipping can be specified with the BTE[1:0] bits in the TBTERA (TBTERB) register.

Figure 16.89 shows an example of operation when buffer transfer is disabled (BTE[1:0] = 01b). While this setting is valid, data is not transferred from the buffer register to the temporary register.

Figure 16.90 shows an example of operation when buffer transfer is linked with interrupt skipping (BTE[1:0] = 10b). While this setting is valid, data is not transferred from the buffer register outside the buffer transfer-enabled period.

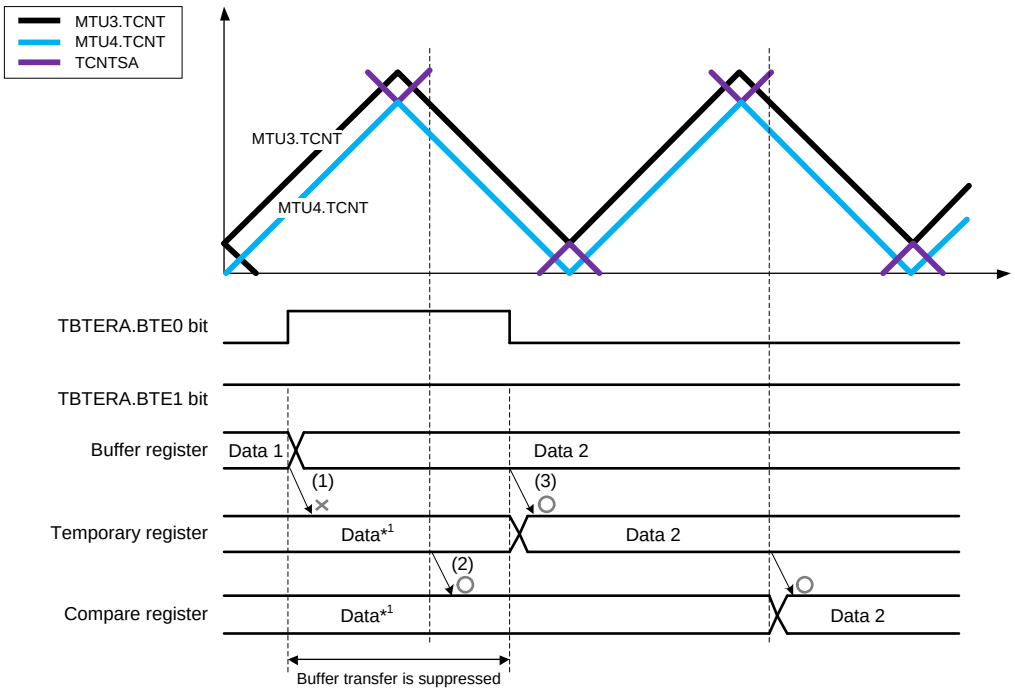
Note that the buffer transfer-enabled period differs depending on whether only the T3AEN (T6AEN) bit in the TITCR1A (TITCR1B) register is set to 1, only the T4VEN (T7VEN) bit in the TITCR1A (TITCR1B) register is set to 1, or both the T3AEN and T4VEN (T6AEN and T7VEN) bits are set to 1. **Figure 16.91** shows the relationship between the T3AEN (T6AEN) and T4VEN (T7VEN) bit settings in TITCR1A (TITCR1B) and buffer transfer-enabled period.

NOTE

This function must be used in combination with interrupt skipping function 1.

When interrupt skipping is disabled (the T3AEN and T4VEN (T6AEN and T7VEN) bits in the timer interrupt skipping set register 1 (TITCR1A or TITCR1B) are cleared to 0 or the skipping count set bits (T3ACOR and T4VCOR (T6ACOR and T7VCOR)) in TITCR1A (TITCR1B) are cleared to 0), make sure that buffer transfer is not linked with interrupt skipping (clear the BTE1 bit in TBTERA or TBTERB to 0).

If buffer transfer is linked with interrupt skipping while interrupt skipping is disabled, buffer transfer is never performed.



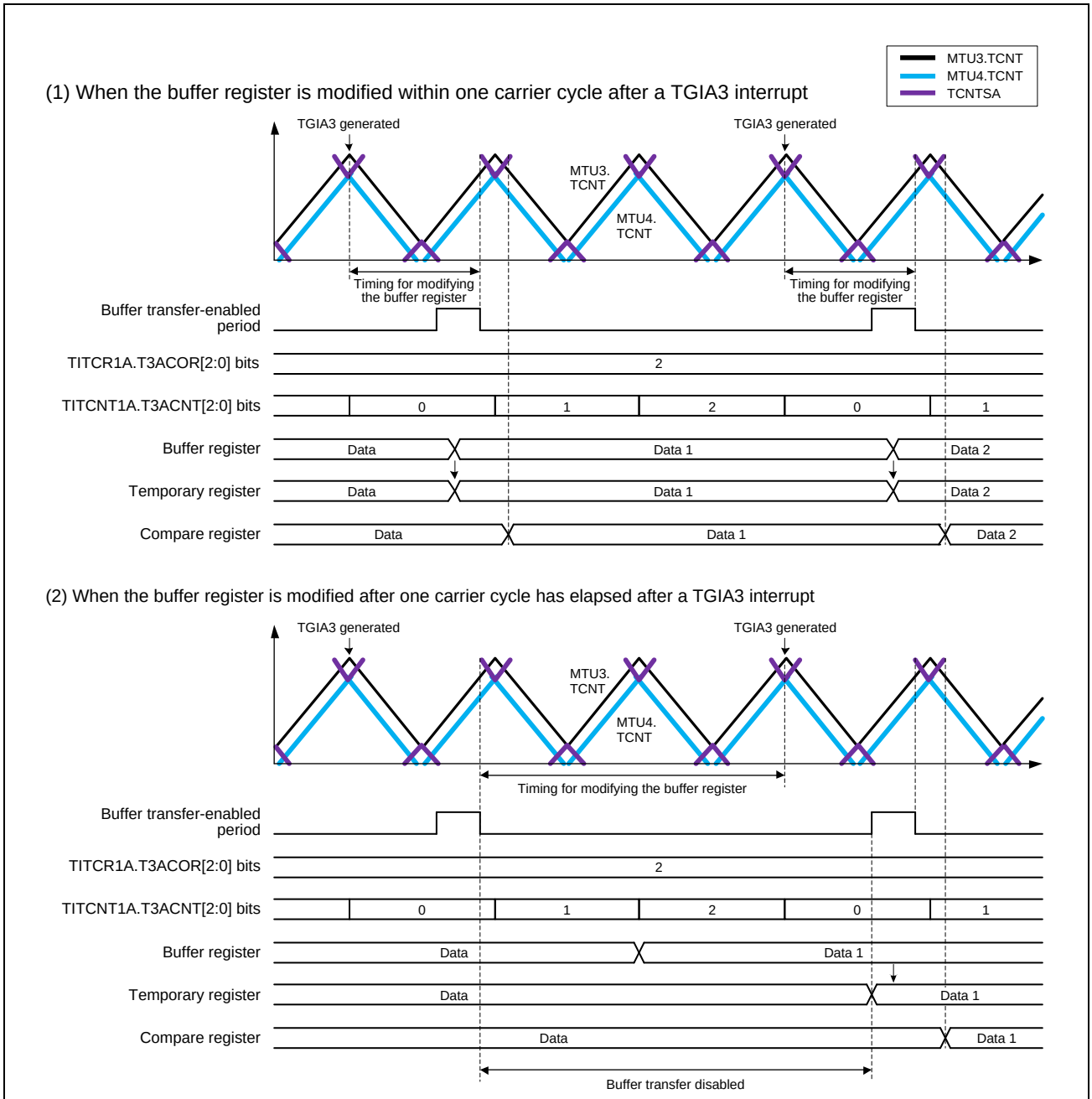
(1) No data is transferred from the buffer register to the temporary register in the buffer transfer-disabled period (bits BTE[1:0] in TBTERA are set to 01b).

(2) Data is transferred from the temporary register to the compare register even in the buffer transfer-disabled period.

(3) After buffer transfer is enabled, data is transferred from the buffer register to the temporary register.

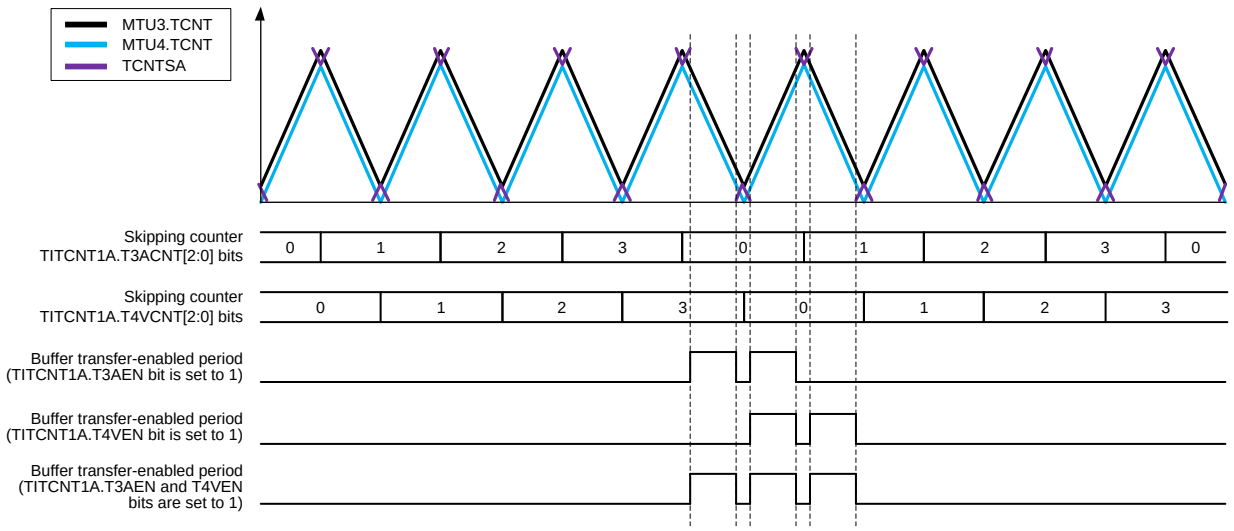
Note 1. When buffer transfer at the crest is selected.

Figure 16.89 Example of Operation When Buffer Transfer is Disabled (BTE[1:0] = 01b)



Note: Buffer transfer at the crest is selected.
The skipping count is set to two. The TITCR1A.T3AEN bit is set to 1.

Figure 16.90 Example of Operation When Buffer Transfer is Linked with Interrupt Skipping (BTE[1:0] = 10b)



Note: The skipping count is set to three.
Buffer operation is set to transfer at the crest and trough.

Figure 16.91 Relationship between Bits T3AEN and T4VEN in TITCR1A and Buffer Transfer-Enabled Period

(4) Complementary PWM Mode Output Protection Functions

The following output protection functions are provided for complementary PWM mode.

(a) Register and Counter Miswrite Prevention Function

Access from the CPU to the mode registers, control registers, compare registers, and counters can be enabled or disabled by setting the RWE bit in the TRWERA (TRWERB) register. The applicable registers are some of the registers in MTU3, MTU4, MTU6, and MTU7 shown below:

MTU3.TCR, MTU4.TCR, MTU3.TCR2, MTU4.TCR2, MTU3.TMDR1, MTU4.TMDR1, MTU3.TIORH, MTU4.TIORH, MTU3.TIORL, MTU4.TIORL, MTU3.TIER, MTU4.TIER, MTU3.TCNT, MTU4.TCNT, MTU3.TGRA, MTU4.TGRA, MTU3.TGRB, MTU4.TGRB, MTU.TOERA, MTU.TOCR1A, MTU.TOCR2A, MTU.TGCRA, MTU.TCDRA, MTU.TDDRA MTU6.TCR, MTU7.TCR, MTU6.TCR2, MTU7.TCR2, MTU6.TMDR1, MTU7.TMDR1, MTU6.TIORH, MTU7.TIORH, MTU6.TIORL, MTU7.TIORL, MTU6.TIER, MTU7.TIER, MTU6.TCNT, MTU7.TCNT, MTU6.TGRA, MTU7.TGRA, MTU6.TGRB, MTU7.TGRB, MTU.TOERB, MTU.TOCR1B, MTU.TOCR2B, MTU.TCDRB, and MTU.TDDRB

47 registers in total

This function can disable CPU access to the mode registers, control registers, and counters to prevent miswriting due to CPU runaway. In the access-disabled state, the applicable registers are read as undefined and writing to these registers is ignored.

(b) Halting of PWM Output by External Signal

The PWM output pins of MTU0, MTU3, MTU4, MTU6, and MTU7 can be set to the high-impedance state automatically.

See **Section 17, Port Output Enable 3 (POE3)**, for details.

16.3.9 A/D Converter Start Request Delaying Function

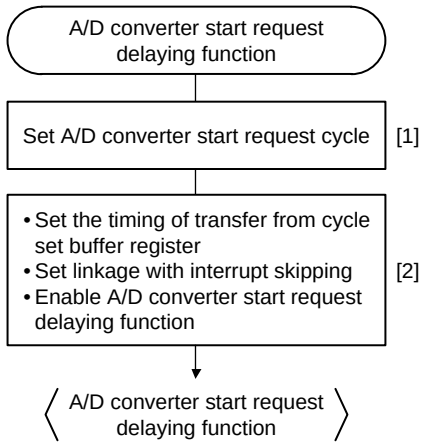
A/D converter start requests can be issued in MTU4 or MTU7 by making settings in the timer A/D converter start request control register (MTU4.TADCR or MTU7.TADCR), timer A/D converter start request cycle set registers (MTU4.TADCORA and MTU4.TADCORB, or MTU7.TADCORA and MTU7.TADCORB), and timer A/D converter start request cycle set buffer registers (MTU4.TADCOBRA and MTU4.TADCOBRB, or MTU7.TADCOBRA and MTU7.TADCOBRB).

The A/D converter start request delaying function compares MTU4.TCNT with MTU4.TADCORA or MTU4.TADCORB (MTU7.TCNT with MTU7.TADCORA or MTU7.TADCORB), and when their values match, the function issues a respective A/D converter start request (TRG4AN or TRG4BN (TRG7AN or TRG7BN)).

A/D converter start requests (TRG4AN and TRG4BN (TRG7AN and TRG7BN)) can be skipped in coordination with interrupt skipping by making settings in the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the TADCR register in MTU4 (the ITA6AE, ITA7VE, ITB6AE, and ITB7VE bits in the TADCR register in MTU7).

(1) Example of Procedure for Specifying A/D Converter Start Request Delaying Function

Figure 16.92 shows an example of procedure for specifying the A/D converter start request delaying function.



[1] Set the cycle in the timer A/D converter start request cycle buffer register (MTU4.TADCOBRA or MTU4.TADCOBRB) and timer A/D converter start request cycle register (MTU4.TADCORA or MTU4.TADCORB).
(When the timing of transfer from the cycle buffer register to cycle register is set to the trough, or both trough and crest in step [2] below, the same initial value must be specified in the cycle buffer register and cycle register because transfer is performed immediately after counting starts.)

[2] Use bits BF[1:0] in the timer A/D converter start request control register (TADCR in MTU4) to specify the timing of transfer from the timer A/D converter start request cycle buffer register to the A/D converter start request cycle register.
- Specify whether to link with interrupt skipping through bits ITA3AE, ITA4VE, ITB3AE, and ITB4VE.
- Use bits UT4AE, DT4AE, UT4BE, and DT4BE to enable A/D converter start requests (TRG4AN or TRG4BN).

Note 1. Set up TADCR in MTU4 while TCNT in MTU4 is stopped.

Note 2. Set bit BF1 of TADCR in MTU4 to 0 when complementary PWM mode is not selected.

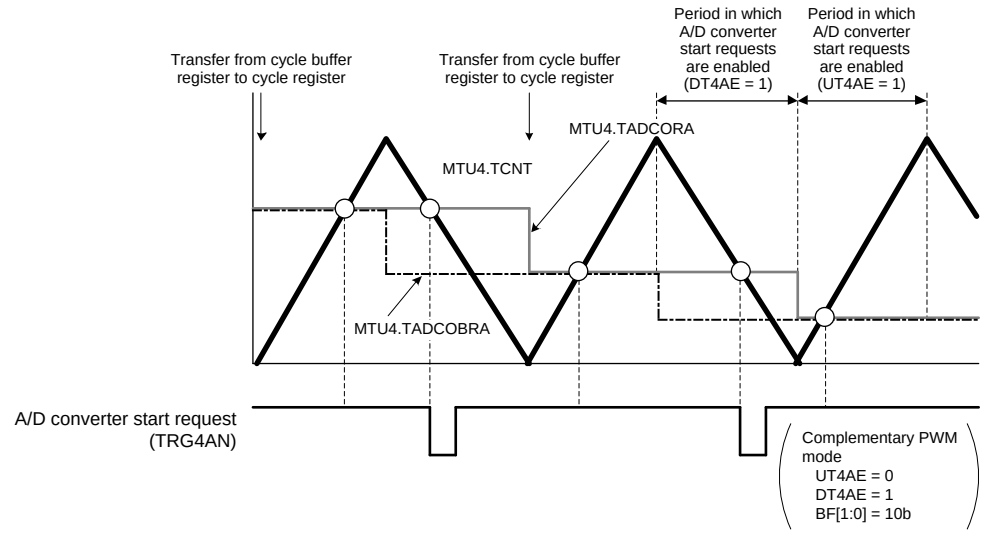
Note 3. Set bits ITA3AE, ITA4VE, ITB3AE, ITB4VE, DT4AE, and DT4BE to 0 when complementary PWM mode is not selected.

Note 4. Set bits ITA3AE, ITA4VE, ITB3AE, and ITB4VE to 0 when interrupt skipping is disabled.

Figure 16.92 Example of Procedure for Specifying A/D Converter Start Request Delaying Function (MTU3 and MTU4)

(2) Basic Example of A/D Converter Start Request Delaying Function Operation

Figure 16.93 shows a basic example of A/D converter start request signal (TRG4AN (TRG7AN)) operation when the trough of MTU4.TCNT (MTU7.TCNT) is specified for the buffer transfer timing and an A/D converter start request signal is output during MTU4.TCNT (MTU7.TCNT) down-counting.



Period in which A/D converter start requests are enabled (UT4AE = 1): 0 ≤ MTU4.TCNT ≤ TCDR – 1
Period in which A/D converter start requests are enabled (DT4AE = 1): TCDR ≥ MTU4.TCNT ≥ 1
UT4AE, DT4AE, BF[1:0]: Bits in TADCR

Figure 16.93 Basic Example of A/D Converter Start Request Signal (TRG4AN) Operation

(3) Period in which A/D Converter Start Requests are Enabled

An A/D converter start request (TRG4AN or TRG4BN) can be issued when the TCNT counter value in MTU4 (MTU7) matches the value of the corresponding cycle set register (TADCORA or TADCORB) in MTU4 (MTU7) within the period specified by the UT4AE or UT4BE (UT7AE or UT7BE) bit of TADCR in MTU4 (MTU7).

When the UT4AE or UT4BE (UT7AE or UT7BE) bit is set to 1 in complementary PWM mode, issuing of AD converter start requests is enabled for the period of TCNT up-counting (0 ≤ TCNT ≤ TCDR - 1) in MTU4 (MTU7). When the DT4AE or DT4BE (DT7AE or DT7BE) bit is set to 1, issuing of AD converter start requests is enabled for the period of TCNT down-counting (TCDR ≥ TCNT ≥ 1) in MTU4 (MTU7) (**Figure 16.93**).

(4) Buffer Transfer

The data in the timer A/D converter start request cycle set registers (MTU4.TADCORA and MTU4.TADCORB, or MTU7.TADCORA and MTU7.TADCORB) is updated by writing data to the timer A/D converter start request cycle set buffer registers (MTU4.TADCOBRA and MTU4.TADCOBRB, or MTU7.TADCOBRA and MTU7.TADCOBRB). Data is transferred from the buffer registers to the respective cycle set registers at the timing selected with the BF[1:0] bits in the MTU4.TADCR (MTU7.TADCR) register.

In complementary PWM mode, data is also transferred from the timer A/D converter start request cycle set buffer registers to the timer A/D converter start request cycle set registers when MTU4.TGRD (MTU7.TGRD) register is updated.

When using buffer transfer in complementary PWM mode, pay attention to the buffer transfer timing. For details, refer to **Section 16.6.26, Notes on A/D Converter Start Request Delaying Function in Complementary PWM Mode**. When complementary PWM mode is not selected, be sure to clear the BF1 bit of the TADCR register in MTU4 or MTU7.

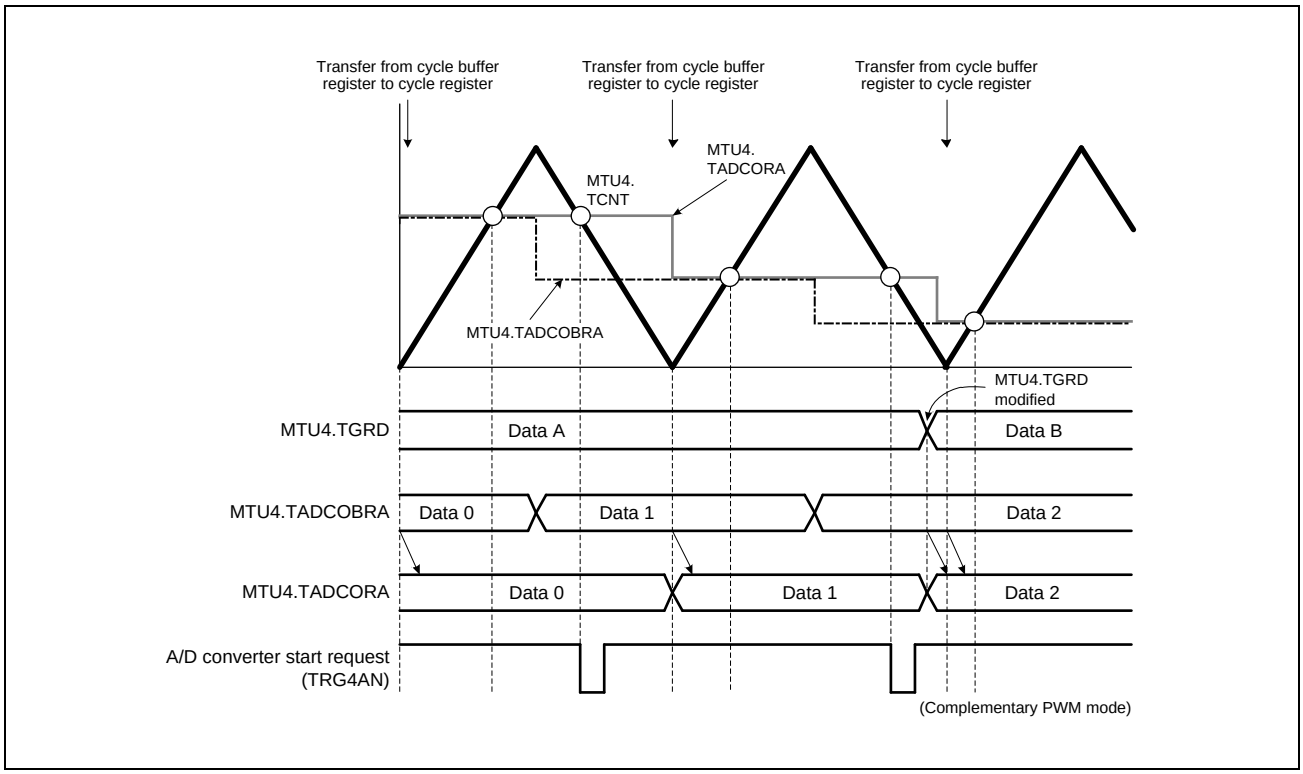


Figure 16.94 Example of A/D Converter Start Request Signal (TRG4AN) and Buffer Transfer Operation

(5) A/D Converter Start Request Delaying Function Linked with Interrupt Skipping Function 1

In complementary PWM mode, A/D converter start requests (TRG4AN and TRG4BN (TRG7AN and TRG7BN)) can be issued in coordination with interrupt skipping 1 by setting the ITA3AE, ITA4VE, ITB3AE, and ITB4VE (ITA6AE, ITA7VE, ITB6AE, and ITB7VE) bits of the TADCR register in MTU4 (MTU7).

Figure 16.95 shows an example of A/D converter start request signal (TRG4AN (TRG7AN)) operation when TRG4AN (TRG7AN) output is enabled during MTU4.TCNT (MTU7.TCNT) up-counting and down-counting and A/D converter start requests are linked with interrupt skipping 1.

Figure 16.96 shows another example of A/D converter start request signal (TRG4AN (TRG7AN)) operation when TRG4AN (TRG7AN) output is enabled during MTU4.TCNT (MTU7.TCNT) up-counting and A/D converter start requests are linked with interrupt skipping 1.

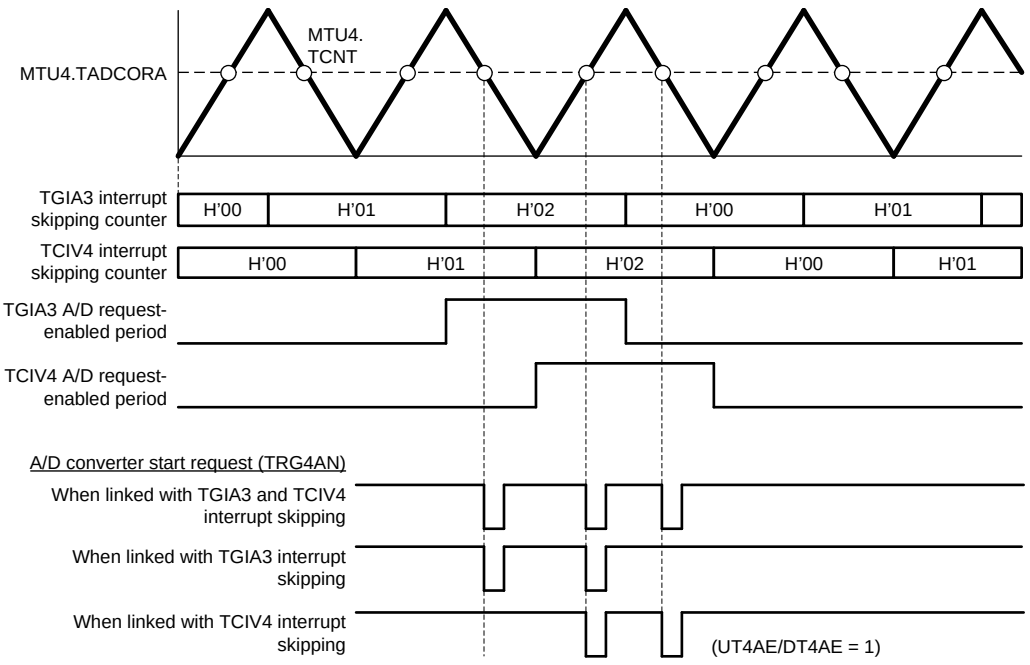
When complementary PWM mode is not selected, A/D converter start request delaying function cannot be used in coordination with interrupt skipping 1; clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE (ITA6AE, ITA7VE, ITB6AE, and ITB7VE) bits of the TADCR register to 0 in MTU4 (MTU7).

NOTE

This function should be used in combination with interrupt skipping 1.

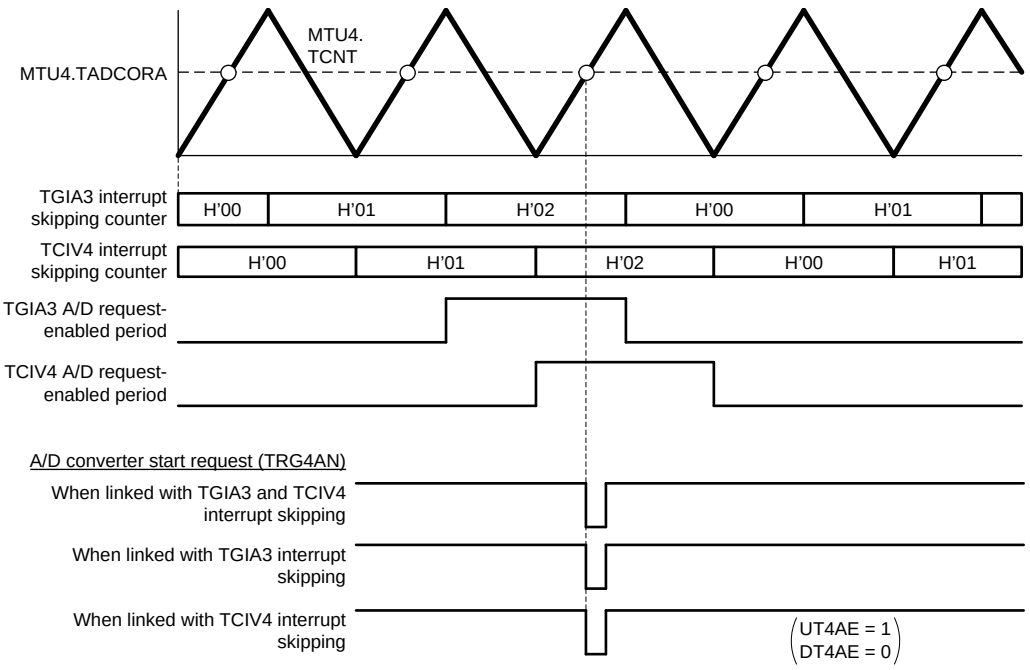
When interrupt skipping is disabled (the T3AEN and T4VEN (T6AEN and T7VEN) bits in the timer interrupt skipping set register (TITCR1A (TITCR1B)) are cleared to 0 or the skipping count set bits (T3ACOR and T4VCOR (T6ACOR and T7VCOR)) in TITCR1A (TITCR1B) are cleared to 0), make sure that A/D converter start requests are not linked with interrupt skipping 1 (clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE (ITA6AE, ITA7VE, ITB6AE, and ITB7VE) bits in the timer A/D converter start request control register (MTU4.TADCR (MTU7.TADCR)) to 0).

When this function is used, MTU4.TADCORA and MTU4.TADCORB (MTU7.TADCORA and MTU7.TADCORB) should be set with the value ranging H'0002 to the value set in TCDRA minus 2 (value set in TCDRB minus 2).



Note: When the interrupt skipping count is set to two.

Figure 16.95 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping Function 1 (UT4AE and DT4AE = 1)



Note: When the interrupt skipping count is set to two.

Figure 16.96 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping Function 1 (UT4AE = 1, DT4AE = 0)

(6) A/D Converter Start Request Delaying Function Linked with Interrupt Skipping Function 2

By setting the TITM bit to 1 in the TITMRA (TITMRB) register, the counter starts down-counting from the value (0 to 7) set in the TRG4COR[2:0] (TRG7COR[2:0]) bits in TITCR2A (TITCR2B) register every time an A/D converter start trigger (TRG4AN or TRG4BN (TRG7AN or TRG7BN)) is generated. When the counter value reaches 0 and is reloaded, the TRG4AN and TRG4BN (TRG7AN and TRG7BN) interrupts become valid and an A/D converter start request signal (TRG4ABN (TRG7ABN)) is output.

This function is valid only when the A/D converter request delaying function is enabled.

(a) Example of Procedure for Setting Interrupt Skipping Function 2

Figure 16.97 shows an example of procedure for setting interrupt skipping function 2.

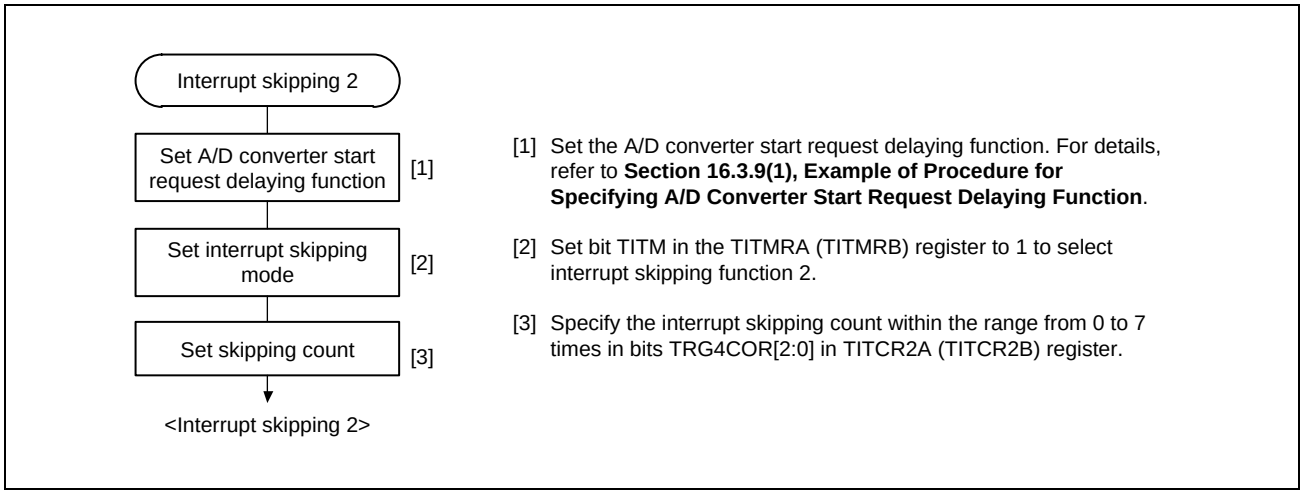


Figure 16.97 Example of Procedure for Setting Interrupt Skipping Function 2

(b) Example of Interrupt Skipping Function 2 Operation

Figure 16.98 shows an example of interrupt skipping 2 operation.

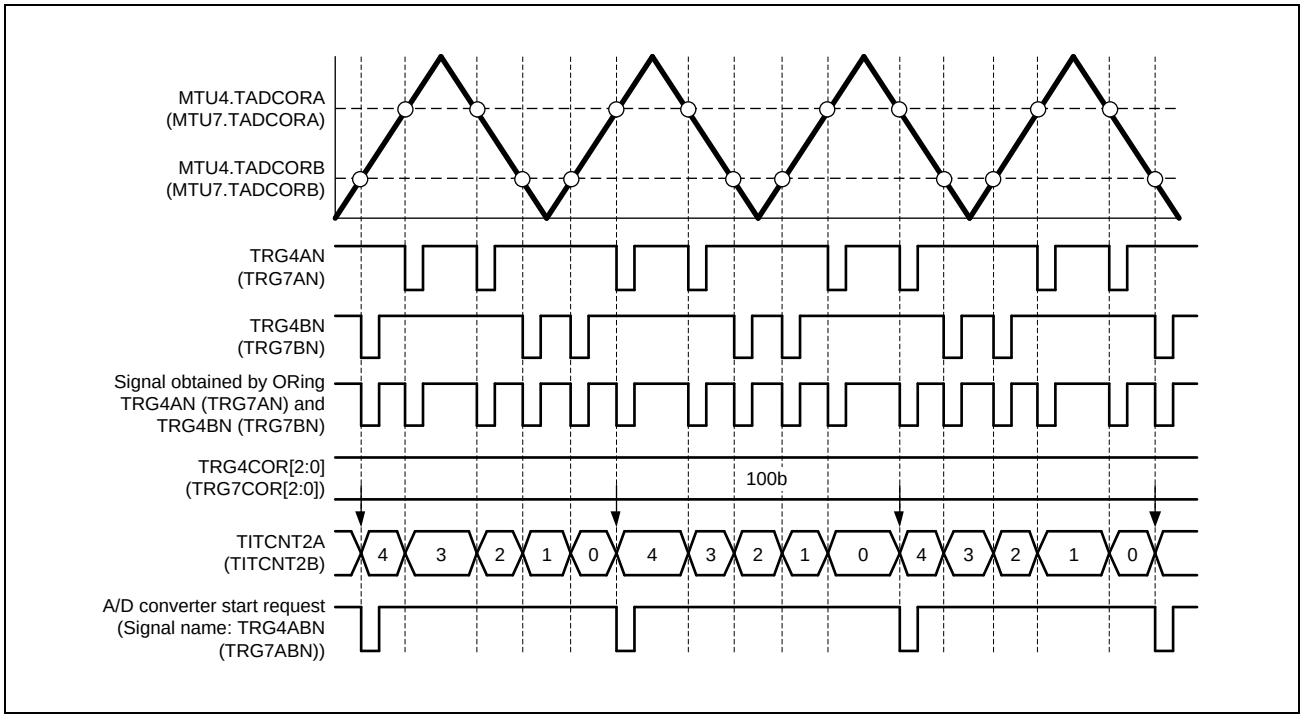


Figure 16.98 Example of Interrupt Skipping 2 Operation (Skipping Count is Set to Four)

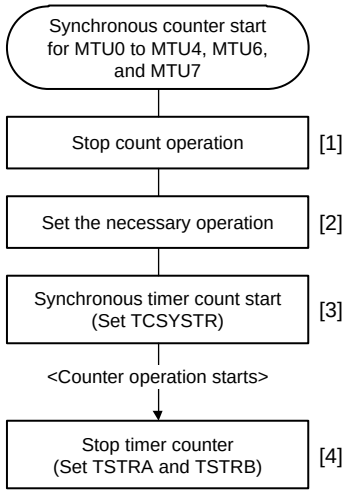
16.3.10 Synchronous Operation of MTU0 to MTU4, MTU6, and MTU7

(1) Synchronous Counter Start for MTU0 to MTU4, MTU6, and MTU7

The counters in MTU0 to MTU4, MTU6, and MTU7 can be started synchronously by making the TCSYSTR settings.

(a) Example of Procedure for Setting Synchronous Counter Start for MTU0 to MTU4, MTU6, and MTU7

Figure 16.99 shows an example of the procedure for specifying synchronous counter start in MTU0 to MTU4, MTU6, and MTU7.



[1] Set TSTRA and TSTRB to stop the counters used for synchronous start operation.

[2] Specify the necessary operation with appropriate registers such as TCR, TMDR1, and TMDR2.

[3] In TCSYSTR, set the bits corresponding to the counters to be started synchronously to 1 at the same time. When TCSYSTR is set, TSTRA and TSTRB for the target timer counters are automatically set appropriately, the counters start synchronously, and TCSYSTR is automatically cleared.

[4] To stop a timer counter, clear the bit in TSTRA or TSTRB that corresponds to the counter to 0.

Note 1. Clearing a bit in TCSYSTR corresponding to an operating counter to 0 does not stop the counter.

Note 2. To start counters synchronously in reset-synchronized PWM mode or complementary PWM mode using MTU3 and MTU4 or MTU6 and MTU7, set the bits in TCSYSTR that correspond to MTU 3 to MTU4, MTU6, and MTU7 to 1 at the same time.

Figure 16.99 Example of Procedure for Specifying Synchronous Counter Start in MTU0 to MTU4, MTU6, and MTU7

(b) Examples of Synchronous Counter Start Operation

Figure 16.100 shows an example of the synchronous counter start operation in MTU0 to MTU4, MTU6, and MTU7.

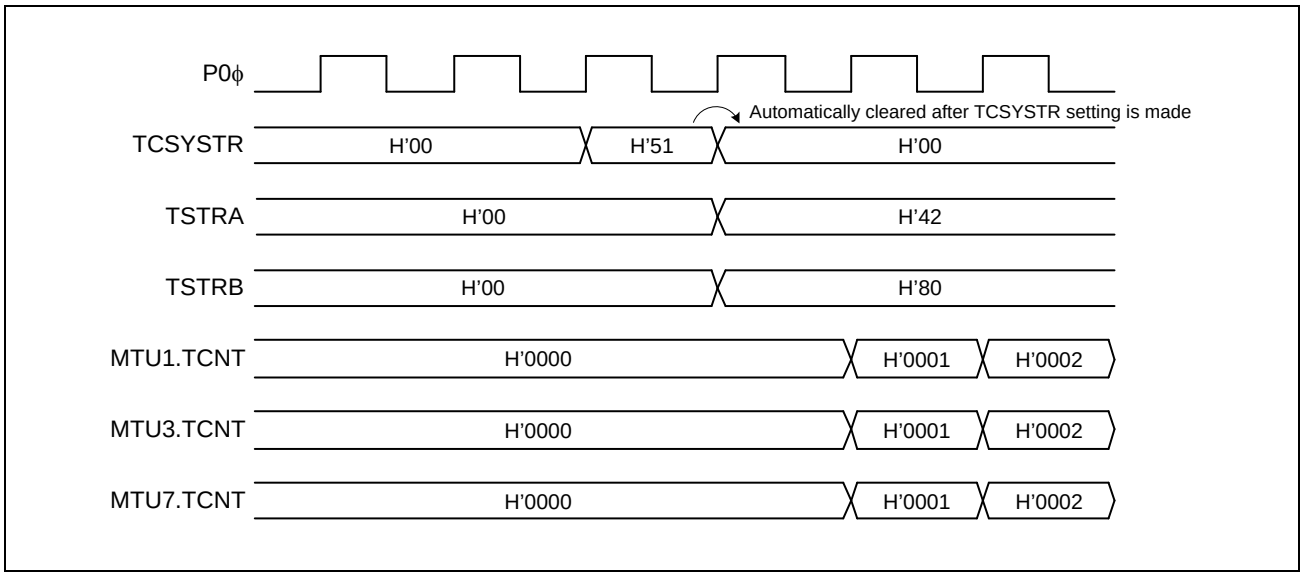


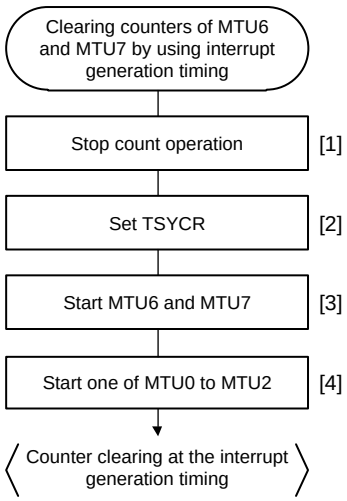
Figure 16.100 Example of Synchronous Counter Start Operation in MTU0 to MTU4, MTU6, and MTU7

(2) Synchronous Counter Clearing for MTU6 and MTU7

The counters in MTU6 and MTU7 can be cleared by the TGImn interrupt generation timing (m = A to D; n = 0 to 2) through the TSYCR setting.

(a) Example of Procedure for Specifying Synchronous Counter Clearing for MTU6 and MTU7

Figure 16.101 shows an example of the procedure for specifying synchronous counter clearing in MTU6 and MTU7 by using the interrupt generation timing.



[1] Set TSTRA and TSTRB to stop the counters used for this function.

[2] Use TSYCR to specify the source to be used for the MTU6.TCNT and MTU7.TCN counter clearing.

[3] Start MTU6.TCNT or MTU7.TCNT.

[4] Start MTU0.TCNT, MTU1.TCNT, or MTU2.TCNT.

Note: The TSYCR setting is ignored while the counter is stopped. The setting becomes valid after MTU6.TCNT or MTU7.TCNT is started.

Figure 16.101 Example of Procedure for Specifying Synchronous Counter Clearing in MTU6 and MTU7

(b) Examples of Synchronous Counter Clearing in MTU6 and MTU7

Figure 16.102 and **Figure 16.103** show examples of synchronous counter clearing in MTU6 and MTU7 by using the interrupt generation timing.

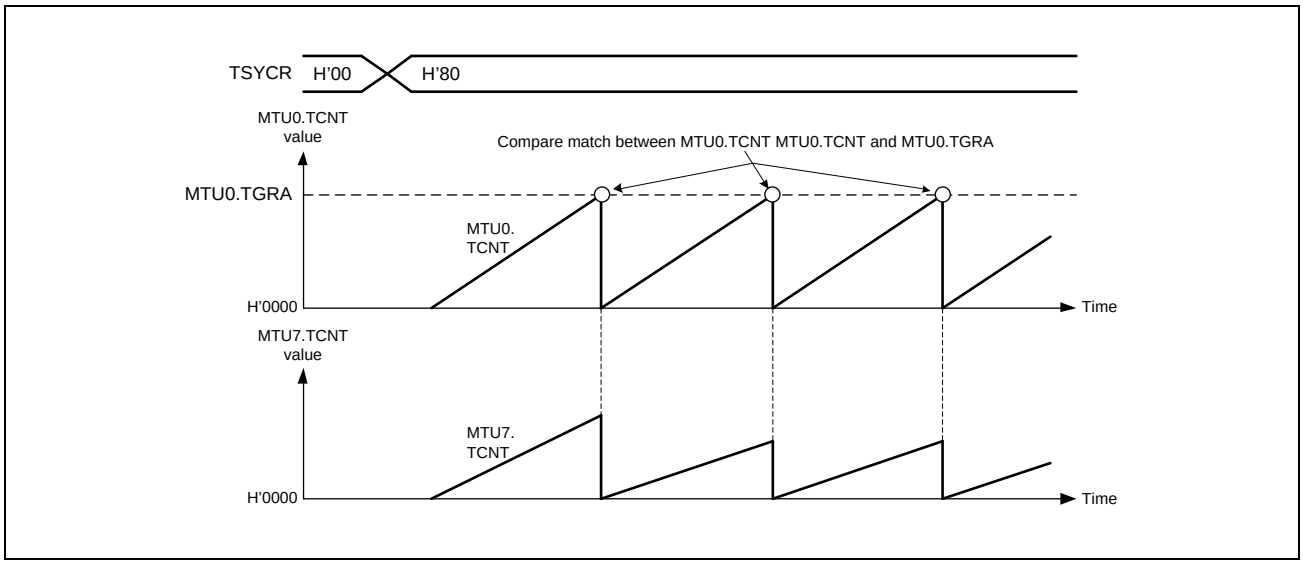


Figure 16.102 Example of Synchronous Counter Clearing in MTU6 and MTU7 (1)

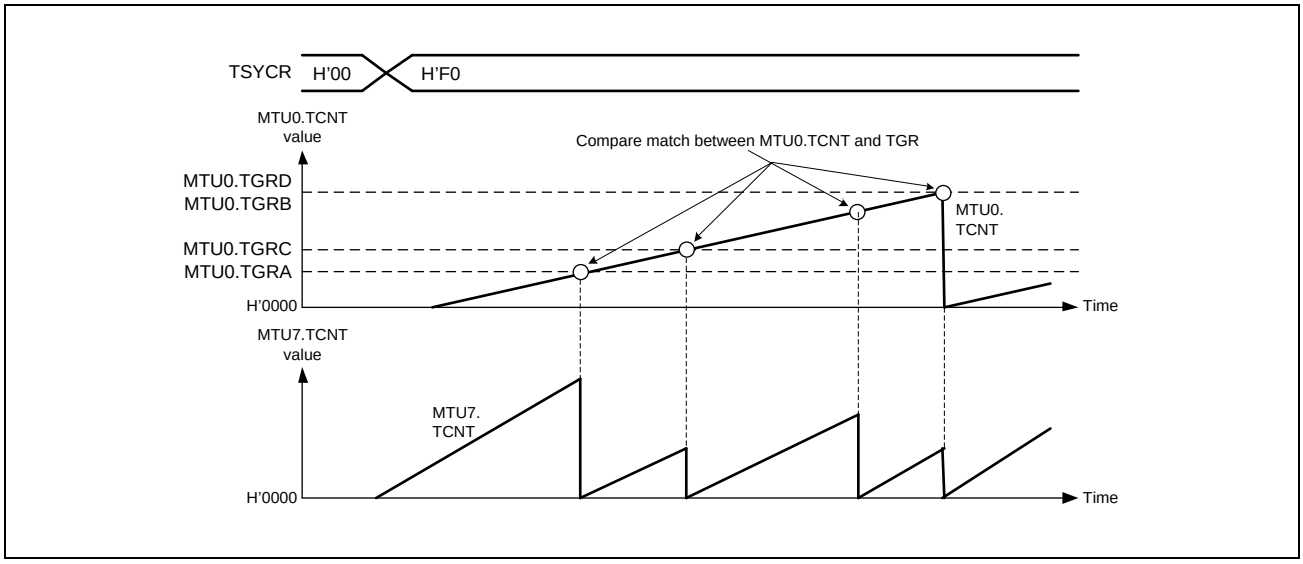


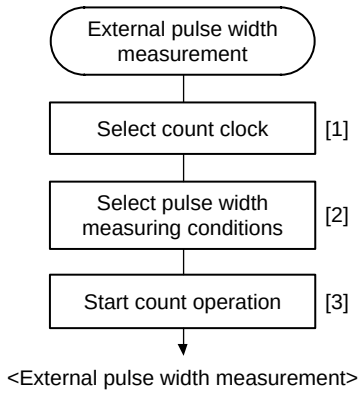
Figure 16.103 Example of Synchronous Counter Clearing in MTU6 and MTU7 (2)

16.3.11 External Pulse Width Measurement

The pulse widths of up to three external input lines can be measured in MTU5.

When the IOC[4:0] bits in MTU5.TIORU, MTU5.TIORV, MTU5.TIORW are set for pulse width measurement, the pulse width of the signal input to the MTIC5U, MTIC5V, and MTIC5W pins are measured. TCNTU, TCNTV, and TCNTW count up while the level specified by the IOC[4:0] bits is input.

Figure 16.104 shows an example of setting external pulse width measurement, and **Figure 16.105** an example of external pulse width measurement.



[1] Use the TPSC1 and TPSC0 bits in MTU5.TCRU, MTU5.TCRV, or MTU5.TCRW to select the count clock.

[2] In MTU5.TIORU, MTU5.TIORV, or MTU5.TIORW, select the high level or low level for the pulse width measuring condition.

[3] Set bit CSTU5, CSTV5, or CSTW5 in MTU5.TSTR to 1 to start count operation.

Note: When MTU6 and MTU7 are not operating in complementary PWM mode, note the following 1 to 3.
1. Set the CMPCLR5U, CMPCLR5V, and CMPCLR5W bits in the TCNTCMPCLR register to 0.
2. Set the TGIE5U, TGIE5V, and TGIE5W bits in MTU5.TIER to 0.
3. The TCNT value is not captured in the TGR register.

Figure 16.104 Example of External Pulse Width Measurement Setting Procedure

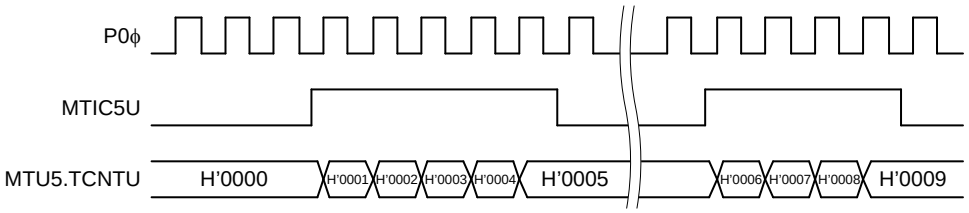


Figure 16.105 Example of External Pulse Width Measurement (Measuring High Pulse Width)

16.3.12 Dead Time Compensation

MTU5 to MTU7 can be used in combination to compensate for the delay in the dead time (the delay between complementary PWM output and inverter output).

Figure 16.106 shows an example of a motor control circuit in which the combination of MTU5 to MTU7 is used to compensate for the dead time delay.

The external pulse measurement function of MTU5 allows the specification of a correction to the duty cycle specified in the PWM output compare registers by measuring the delay between the complementary PWM output and inverter output. This can be used for dead time compensation for the PWM output waveform during complementary PWM operation of MTU6 and MTU7 (**Figure 16.107**).

Figure 16.108 shows the procedure for setting dead time compensation using MTU5 to MTU7. For details on MTU5 operation at this time, refer to **Section 16.3.13, TCNTU, TCNTV, and TCNTW Capture at Crest and/or Trough in Complementary PWM Mode**.

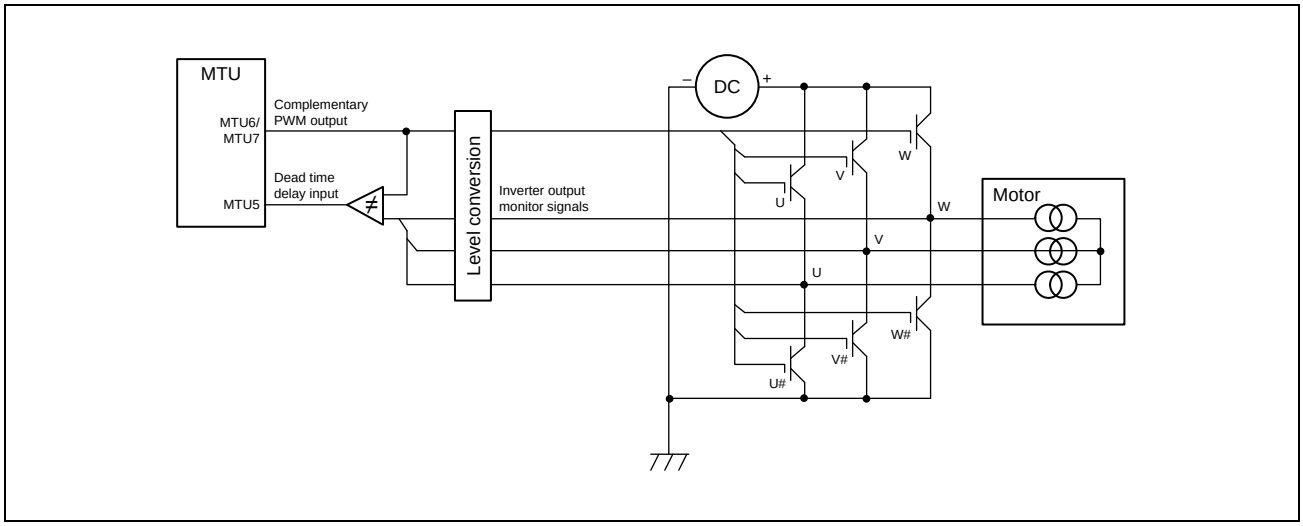


Figure 16.106 Motor Control Circuit Example

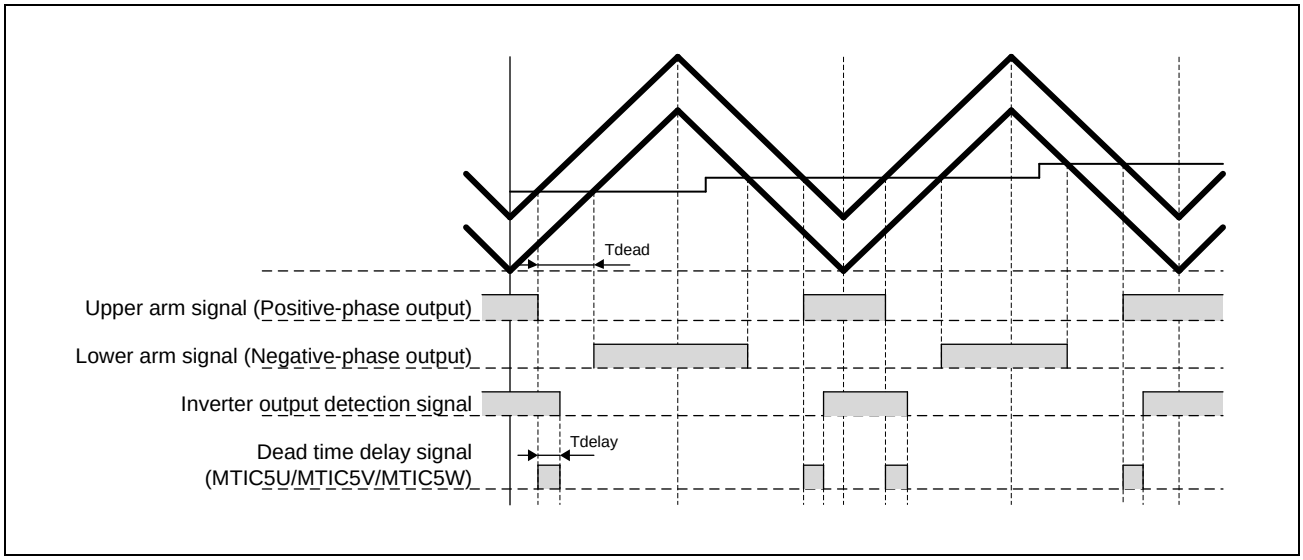
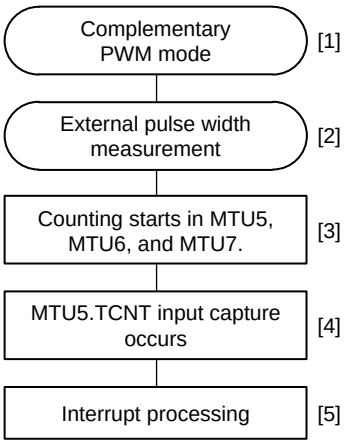


Figure 16.107 Delay in Dead Time in Complementary PWM Operation

(1) Example of Dead Time Compensation Setting Procedure

Figure 16.108 shows an example of dead time compensation setting procedure by using three counters in MTU5.



[1] Set MTU6 and MTU7 to complementary PWM mode. For details, refer to **Section 16.3.8, Complementary PWM Mode**.

[2] Specify the external pulse width measurement function for the target TIOR in MTU5. For details, refer to **Section 16.3.11, External Pulse Width Measurement**.

[3] Set bits CST6 and CST7 in TSTRB and bits CSTU5, CSTV5, and CSTW5 in MTU5.TSTR to 1 to start count operation.

[4] When the capture condition specified in TIOR is satisfied, the MTU5.TCNT value is captured in MTU5.TGR.
To prevent the generation of the MTU5.TGIm5 interrupt (m = U, V, W) request under the capture condition specified in TIOR, do not enable interrupt requests in MTU5.TIER.

[5] For U-phase dead time compensation, when an interrupt is generated at the crest (TGIA6) or trough (TCIV7) in complementary PWM mode, read the MTU5.TGRU value, calculate the difference in time between MTU6.TGRB and MTU5.TGRU, and write the corrected value to MTU6.TGRD in the interrupt processing.
For the V phase and W phase, read the MTU5.TGRV and MTU5.TGRW values and write the corrected values to MTU7.TGRC and TGRD, respectively, in the same way as for U-phase compensation.
The MTU5.TCNT value should be cleared through the TCNTCMPCLR setting or by software.

Note: The port mode registers (PMn) and Pmn port function control register (PFCm) of the GPIO ports should be set up in advance.
As the interrupt flag is set under the capture condition specified in TIOR, do not enable interrupt requests in MTU5.TIER.

Figure 16.108 Example of Dead Time Compensation Setting Procedure

16.3.13 TCNTU, TCNTV, and TCNTW Capture at Crest and/or Trough in Complementary PWM Mode

The external pulse width measurement function of MTU5 can be used to transfer the values of TCNTU, TCNTV, and TCNTW to TGRU, TGRV, and TGRW at the crests, troughs, or both crests and troughs of the complementary PWM output when MTU6 and MTU7 are being used in complementary PWM mode. The type of transfer timing is specified in TIORU, TIORV, and TIORW. When the CMPCLR5U, CMPCLR5V, and CMPCLR5W bits in the TCNTCNPCLR register are set to 1, TCNTU, TCNTV, and TCNTW are cleared to H'0000 at the timing for transfer to TGRU, TGRV, and TGRW.

Note that the TCNTU, TCNTV, and TCNTW capture operation at the crest and/or trough of complementary PWM output in MTU5 is not available when MTU3 and MTU4 are used in complementary PWM mode.

Figure 16.109 shows an operation example in which TCNTU is used as a free-running counter without being cleared and the value is captured in TGRU at the crest and trough in complementary PWM mode.

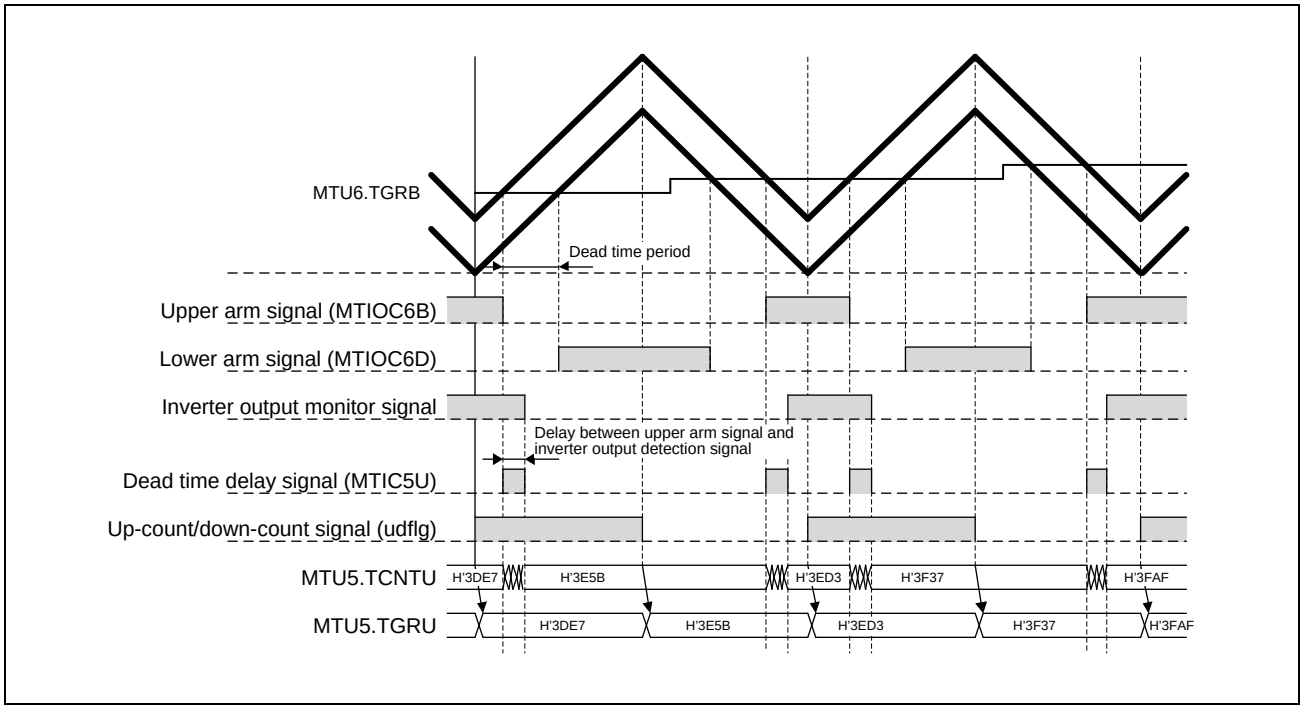


Figure 16.109 TCNTU Capture at Crest and Trough in Complementary PWM Operation

16.3.14 Noise Filter Function

The input capture input pins and external pulse input pins have a noise filter function.

Set the NFCRn register (n = 0 to 7, C) to enable or disable the noise filter function and set the sampling clock. The noise filter for each pin can be enabled or disabled individually, and the sampling clock can be set for each channel.

Figure 16.110 shows the timing of noise filtering.

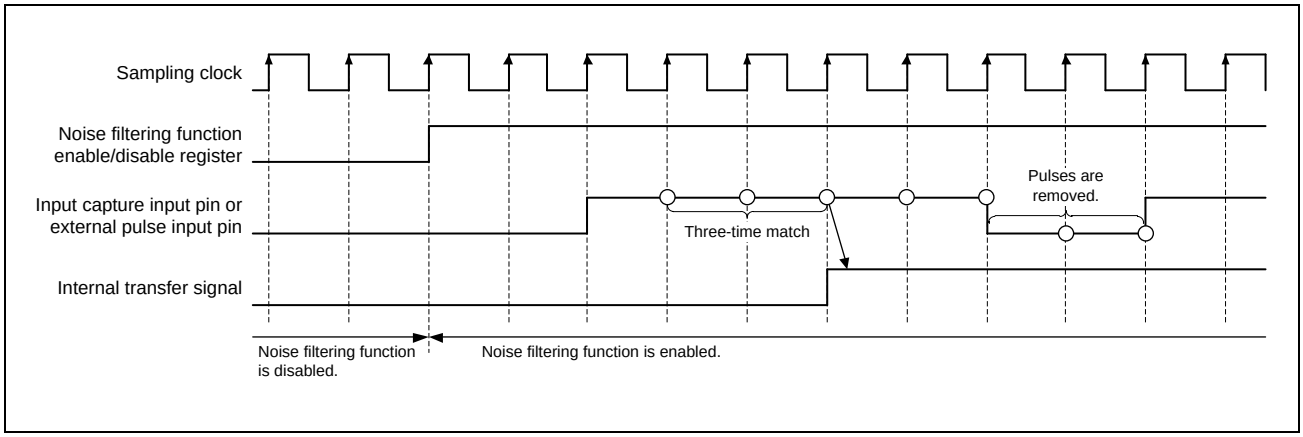


Figure 16.110 Timing of Noise Filtering

16.4 Interrupt Sources

16.4.1 Interrupt Sources and Priorities

There are three kinds of interrupt source; TGR input capture/compare match, TCNT overflow, and TCNT underflow. Each interrupt source has its own enable/disable bit, allowing the generation of interrupt request signals to be enabled or disabled individually.

When an interrupt source is generated, if the corresponding enable/disable bit in TIER is set to 1, an interrupt is requested.

Relative channel priorities can be changed by the interrupt controller; however the priority within a channel is fixed. For details, refer to **Section 8, Interrupt Controller**. **Table 16.78** lists the MTU interrupt sources.

Table 16.78 MTU Interrupt Sources (1/2)

Channel	Name	Interrupt Source	DMAC Activation
MTU0	TGIA0	MTU0.TGRA input capture/compare match	Possible
	TGIB0	MTU0.TGRB input capture/compare match	Possible
	TGIC0	MTU0.TGRC input capture/compare match	Possible
	TGID0	MTU0.TGRD input capture/compare match	Possible
	TCIV0	MTU0.TCNT overflow	Not possible
	TGIE0	MTU0.TGRE compare match	Not possible
	TGIF0	MTU0.TGRF compare match	Not possible
MTU1	TGIA1	MTU1.TGRA input capture/compare match	Possible
	TGIB1	MTU1.TGRB input capture/compare match	Possible
	TCIV1	MTU1.TCNT overflow	Not possible
	TCIU1	MTU1.TCNT underflow	Not possible
MTU2	TGIA2	MTU2.TGRA input capture/compare match	Possible
	TGIB2	MTU2.TGRB input capture/compare match	Possible
	TCIV2	MTU2.TCNT overflow	Not possible
	TCIU2	MTU2.TCNT underflow	Not possible
MTU3	TGIA3	MTU3.TGRA input capture/compare match	Possible
	TGIB3	MTU3.TGRB input capture/compare match	Possible
	TGIC3	MTU3.TGRC input capture/compare match	Possible
	TGID3	MTU3.TGRD input capture/compare match	Possible
	TCIV3	MTU3.TCNT overflow	Not possible
MTU4	TGIA4	MTU4.TGRA input capture/compare match	Possible
	TGIB4	MTU4.TGRB input capture/compare match	Possible
	TGIC4	MTU4.TGRC input capture/compare match	Possible
	TGID4	MTU4.TGRD input capture/compare match	Possible
	TCIV4	MTU4.TCNT overflow/underflow*1	Possible
MTU5	TGIU5	MTU5.TGRU input capture/compare match	Possible
	TGIV5	MTU5.TGRV input capture/compare match	Possible
	TGIW5	MTU5.TGRW input capture/compare match	Possible
MTU6	TGIA6	MTU6.TGRA input capture/compare match	Possible
	TGIB6	MTU6.TGRB input capture/compare match	Possible
	TGIC6	MTU6.TGRC input capture/compare match	Possible
	TGID6	MTU6.TGRD input capture/compare match	Possible
	TCIV6	MTU6.TCNT overflow	Not possible

Table 16.78 MTU Interrupt Sources (2/2)

Channel	Name	Interrupt Source	DMAC Activation
MTU7	TGIA7	MTU7.TGRA input capture/compare match	Possible
	TGIB7	MTU7.TGRB input capture/compare match	Possible
	TGIC7	MTU7.TGRC input capture/compare match	Possible
	TGID7	MTU7.TGRD input capture/compare match	Possible
	TCIV7	MTU7.TCNT overflow/underflow*1	Possible
MTU8	TGIA8	MTU8.TGRA input capture/compare match	Possible
	TGIB8	MTU8.TGRB input capture/compare match	Possible
	TGIC8	MTU8.TGRC input capture/compare match	Possible
	TGID8	MTU8.TGRD input capture/compare match	Possible
	TCIV8	MTU8.TCNT overflow	Not possible

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

Note 1. The underflow interrupt source is valid only in complementary PWM mode.

(1) Input Capture/Compare Match Interrupt

If the TIER.TGIE bit is set to 1 when a TGR input capture/compare match occurs on a channel, an interrupt is requested. The MTU has 33 input capture/compare match interrupts (six for MTU0, four each for MTU3, MTU4, MTU6, MTU7, and MTU8, two each for MTU1 and MTU2, and three for MTU5).

(2) Overflow Interrupt

If the TIER.TCIEV bit is set to 1 when a TCNT overflow occurs on a channel, clearing an interrupt is requested. The MTU provides a total of eight overflow interrupts (one for each channel except MTU5).

In complementary PWM mode, an overflow interrupt is generated even when an underflow occurs in TCNT in MTU4 or MTU7.

(3) Underflow Interrupt

If the TIER.TCIEU bit is set to 1 when a TCNT underflow occurs on a channel, an interrupt is requested. The MTU has two underflow interrupts (one each for MTU1, and MTU2).

16.4.2 DMAC Activation

(1) DMAC Activation

The DMAC can be activated by the TGR input capture/compare match interrupt and the overflow interrupt in MTU4 and MTU7. For details, see **Section 14, Direct Memory Access Controller**.

The MTU provides a total of 33 interrupts (input capture/compare match and overflow interrupts) that can be used as DMAC activation sources: four each for MTU0, MTU3, MTU6, and MTU8, two each for MTU1 and MTU2, five each for MTU4 and MTU7, and three for MTU5.

16.4.3 A/D Converter Activation

The A/D converter can be activated by one of the following three methods in the MTU. **Table 16.79** shows the relationship between interrupt sources and A/D converter start request signals.

(1) A/D Converter Activation by TGRA Input Capture/Compare Match or at MTU4.TCNT (MTU7.TCNT) Trough in Complementary PWM Mode

The A/D converter can be activated by the occurrence of a TGRA input capture/compare match in each channel. In addition, if complementary PWM operation is performed while the TTGE2 bit in MTU4.TIER (MTU7.TIER) is set to 1, the A/D converter can be activated at the trough of MTU4.TCNT (MTU7.TCNT) count (MTU4.TCNT (MTU7.TCNT) = H'0000).

A/D converter start request signal TRGAnN is issued to the A/D converter under either of the following conditions (n = 0 to 4, 6, or 7).

- When a TGRA input capture/compare match occurs on a channel while the TIER.TTG bit is set to 1
- When the MTU4.TCNT (MTU7.TCNT) count reaches the trough (MTU4.TCNT (MTU7.TCNT) = H'0000) during complementary PWM operation while the TTGE2 bit in MTU4.TIER (MTU7.TIER) is set to 1

When either condition is satisfied, if A/D converter start signal TRGAnN from the MTU is selected as the trigger in the A/D converter, A/D conversion will start.

(2) A/D Converter Activation by Compare Match between MTU0.TCNT and MTU0.TGRE

A/D converter start request signal TRG0N is issued to the A/D converter when a compare match occurs between MTU0.TCNT and MTU0.TGRE.

When a compare match occurs between MTU0.TCNT and MTU0.TGRE in MTU0 while the TTGE2 bit.TIER2 is set to 1, A/D converter start request TGR0N is issued to the A/D converter. If A/D converter start signal TRG0N from the MTU is selected as the trigger in the A/D converter, A/D conversion will start.

(3) A/D Converter Activation by A/D Converter Start Request Delaying Function

The A/D converter can be activated by generating A/D converter start request signal TRG4AN or TRG4BN (TRG7AN or TRG7BN) when the MTU4.TCNT (MTU7.TCNT) count matches the MTU4.TADCORA or MTU4.TADCORB (MTU7.TADCORA or MTU7.TADCORB) value if the UT4AE, DT4AE, UT4BE, or DT4BE (UT7AE, DT7AE, UT7BE, or DT7BE) bit in the A/D converter start request control register (MTU4.TADCR (MTU7.TADCR)) is set to 1. For details, refer to **Section 16.3.9, A/D Converter Start Request Delaying Function**.

A/D conversion will start when TRG4AN (TRG7AN) is generated if A/D converter start signal TRG4AN (TRG7AN) from the MTU is selected as the trigger in the A/D converter, when TRG4BN (TRG7BN) is generated if TRG4BN (TRG7BN) from the MTU is selected as the trigger in the A/D converter, or when TRG4ABN (TRG7ABN) is generated if TRG4ABN (TRG7ABN) from the MTU is selected as the trigger in the A/D converter.

Table 16.79 Interrupt Sources and A/D Converter Start Request Signals

Target Registers	Interrupt Source	A/D Converter Start Request Signal
MTU0.TGRA and MTU0.TCNT	Input capture/compare match	TRGA0N
MTU1.TGRA and MTU1.TCNT		TRGA1N
MTU2.TGRA and MTU2.TCNT		TRGA2N
MTU3.TGRA and MTU3.TCNT		TRGA3N
MTU4.TGRA and MTU4.TCNT*1		TRGA4N
MTU4.TCNT	MTU4.TCNT trough in complementary PWM mode	
MTU6.TGRA and MTU6.TCNT	Input capture/compare match	TRGA6N
MTU7.TGRA and MTU7.TCNT*1		TRGA7N
MTU7.TCNT	MTU7.TCNT trough in complementary PWM mode	
MTU0.TGRE and MTU0.TCNT	Compare match	TRG0N
MTU4.TADCORA and MTU4.TCNT		TRG4AN
MTU4.TADCORB and MTU4.TCNT		TRG4BN
MTU7.TADCORA and MTU7.TCNT		TRG7AN
MTU7.TADCORB and MTU7.TCNT		TRG7BN
MTU4.TADCORA and MTU4.TCNT, MTU4.TADCORB and MTU4.TCNT	Compare match (interrupt skipping function 2)	TRG4ABN
MTU7.TADCORA and MTU7.TCNT, MTU7.TADCORB and MTU7.TCNT		TRG7ABN

Note 1. Since PWM waveforms are generated in complementary PWM mode, MTU4.TGRA (MTU7.TGRA) compare match not only with MTU4.TCNT (MTU7.TCNT) but also with MTU3.TCNT (MTU6.TCNT) and TCNTSA (TCNTSB) is detected. Accordingly, when compare match with MTU3.TCNT (MTU6.TCNT) and TCNTSA (TCNTSB) occurs, TRGA4N (TRGA7N) is also generated.
When MTU3 and MTU 4 (MTU 6 and MTU7) are made to operate in complementary PWM mode for generating an A/D converter start request, use the A/D converter start request by compare match between MTU4.TCNT (MTU7.TCNT) and MTU4.TADCORA or TADCORB (MTU7.TADCORA or TADCORB).

16.5 Operation Timing

16.5.1 Input/Output Timing

(1) TCNT Count Timing

Figure 16.111 and **Figure 16.112** show the TCNT count timing in internal clock operation, **Figure 16.113** shows the TCNT count timing in external clock operation (normal mode), and **Figure 16.114** shows the TCNT count timing in external clock operation (phase counting mode).

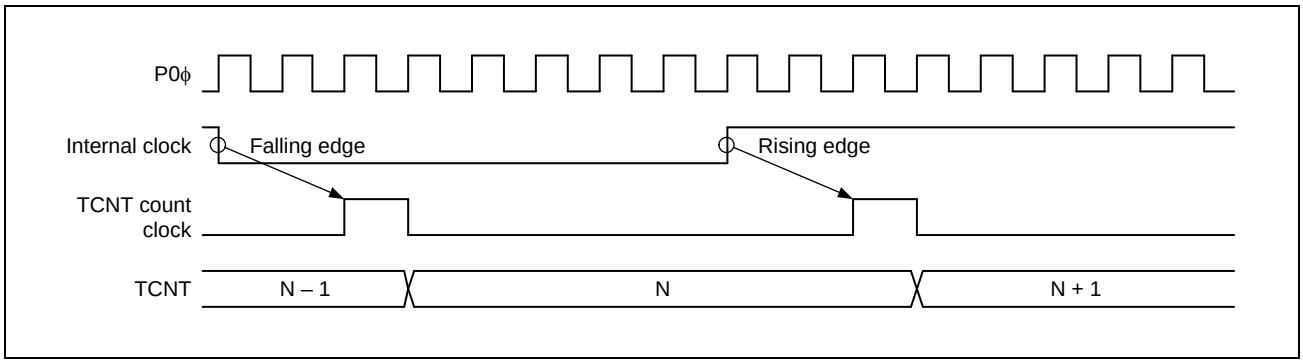


Figure 16.111 Count Timing in Internal Clock Operation (MTU0 to MTU4 and MTU6 to MTU8)

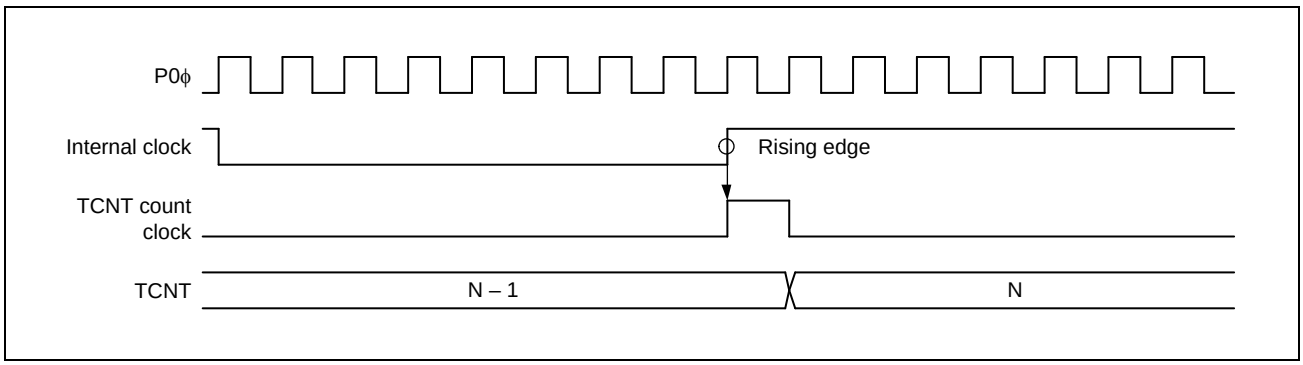


Figure 16.112 Count Timing in Internal Clock Operation (MTU5)

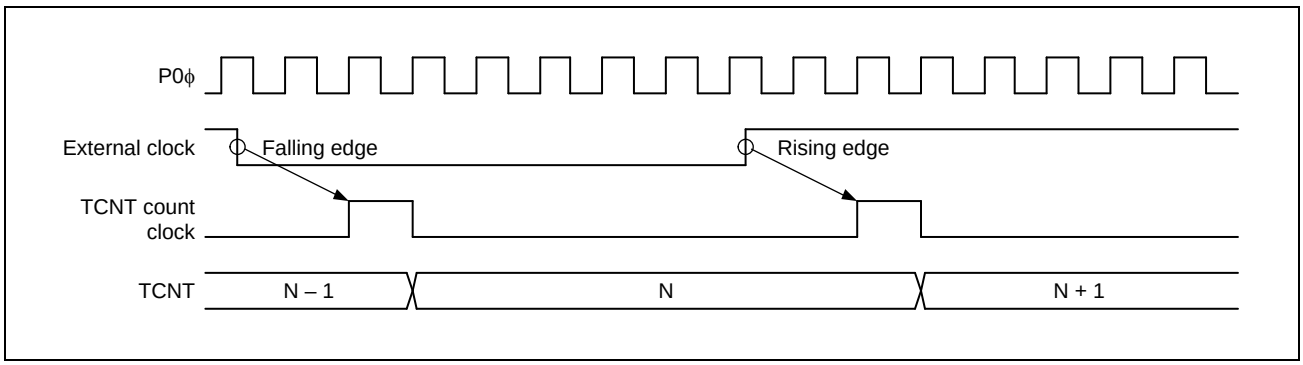


Figure 16.113 Count Timing in External Clock Operation (MTU0 to MTU4, MTU6 to MTU8)

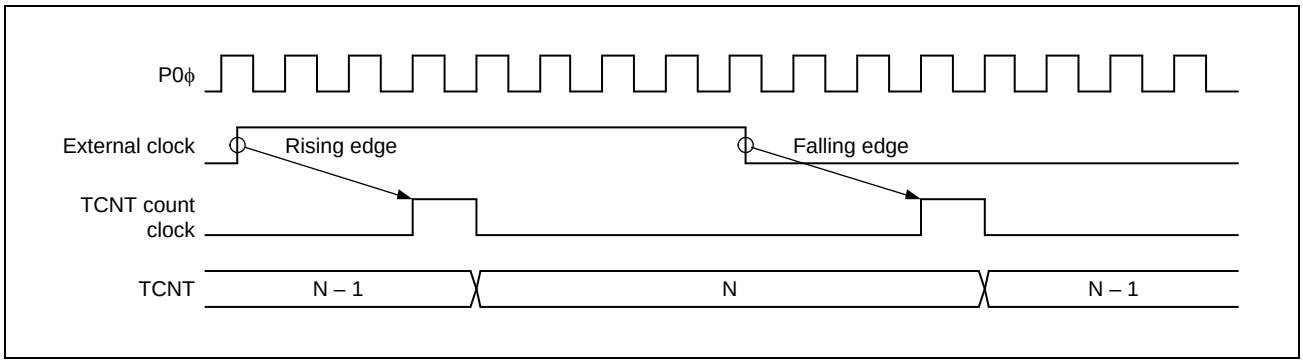


Figure 16.114 Count Timing in External Clock Operation (Phase Counting Mode)

(2) Output Compare Output Timing

A compare match signal is generated in the final state in which TCNT and TGR match (the point at which the count value matched by TCNT is updated). When a compare match signal is generated, the value set in TIOR is output from MTIOCnm pin (n = 0 to 4, 6, 7, 8; m = A to D). After a match between TCNT and TGR, the compare match signal is not generated until the TCNT count clock is generated.

Figure 16.115 shows the output compare output timing (normal mode or PWM mode) and **Figure 16.116** shows the output compare output timing (complementary PWM mode or reset-synchronized PWM mode).

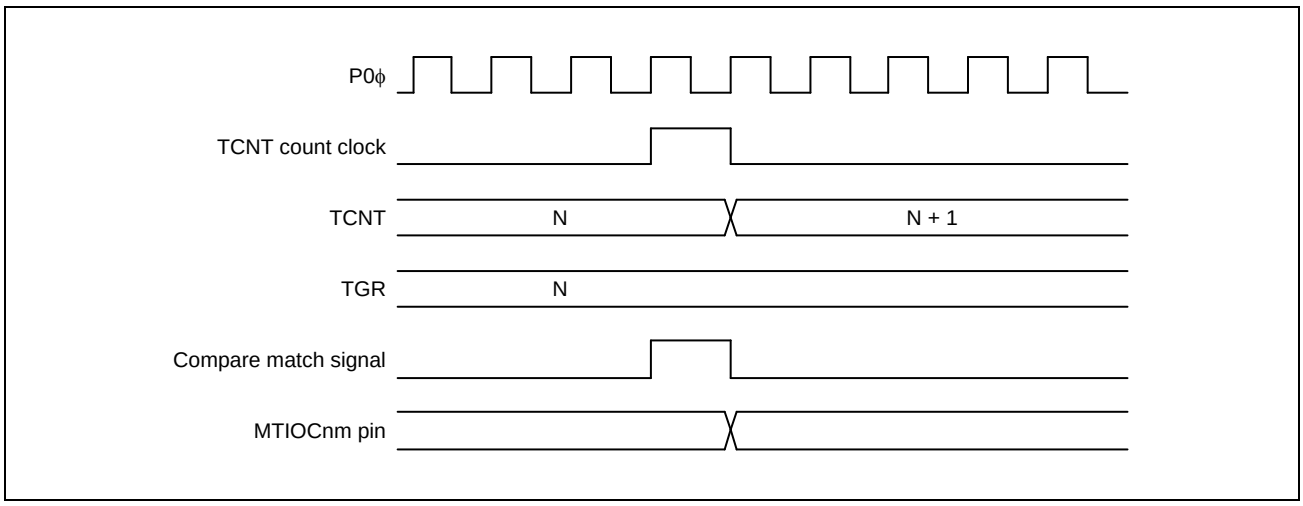


Figure 16.115 Output Compare Output Timing (Normal Mode or PWM Mode) (n = 0 to 4, 6, 7, 8; m = A to D)

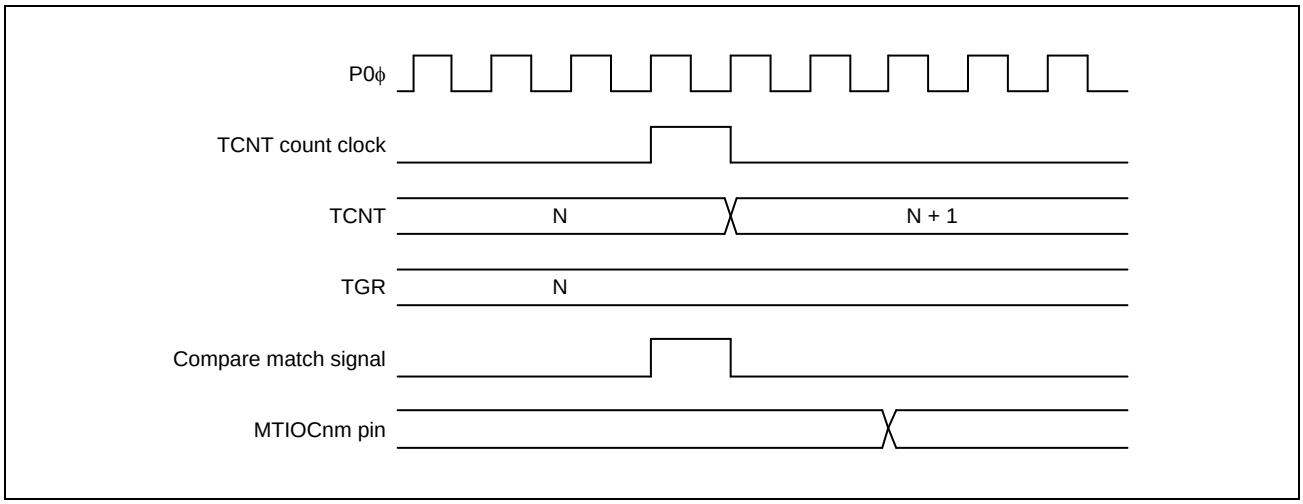


Figure 16.116 Output Compare Output Timing (Complementary PWM Mode or Reset-Synchronized PWM Mode) (n = 0 to 4, 6, 7, 8; m = A to D)

(3) Input Capture Signal Timing

Figure 16.117 shows the input capture signal timing.

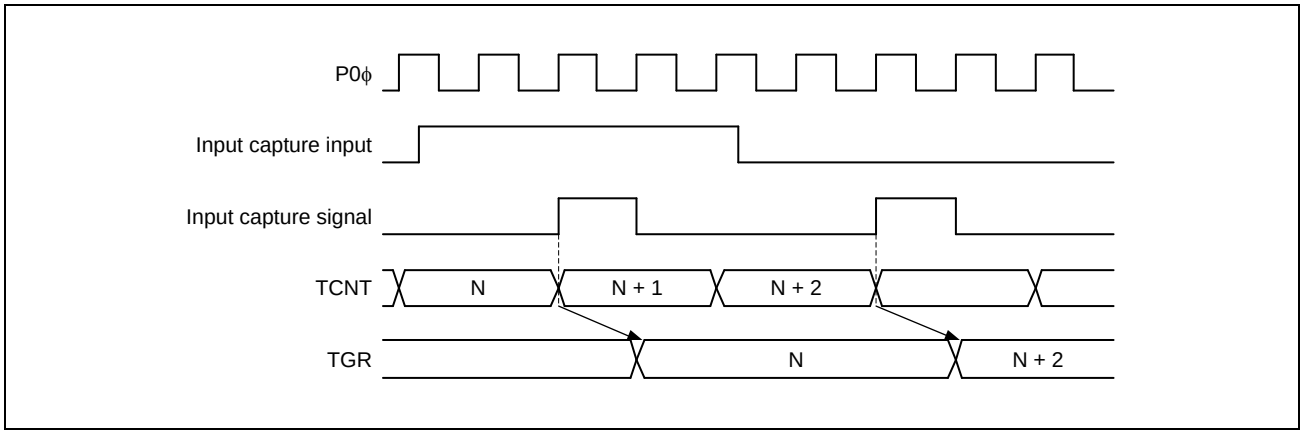


Figure 16.117 Input Capture Input Signal Timing

(4) Timing for Counter Clearing by Compare Match/Input Capture

Figure 16.118 and **Figure 16.119** show the timing when counter clearing on compare match is specified, and **Figure 16.120** shows the timing when counter clearing on input capture is specified.

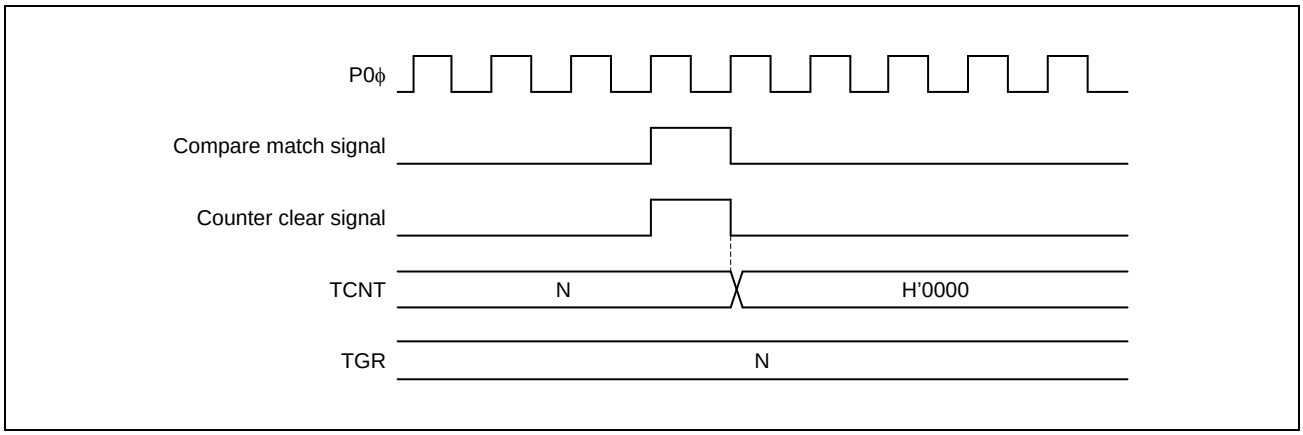


Figure 16.118 Counter Clear Timing (Compare Match) (MTU0 to MTU4 and MTU6 to MTU8)

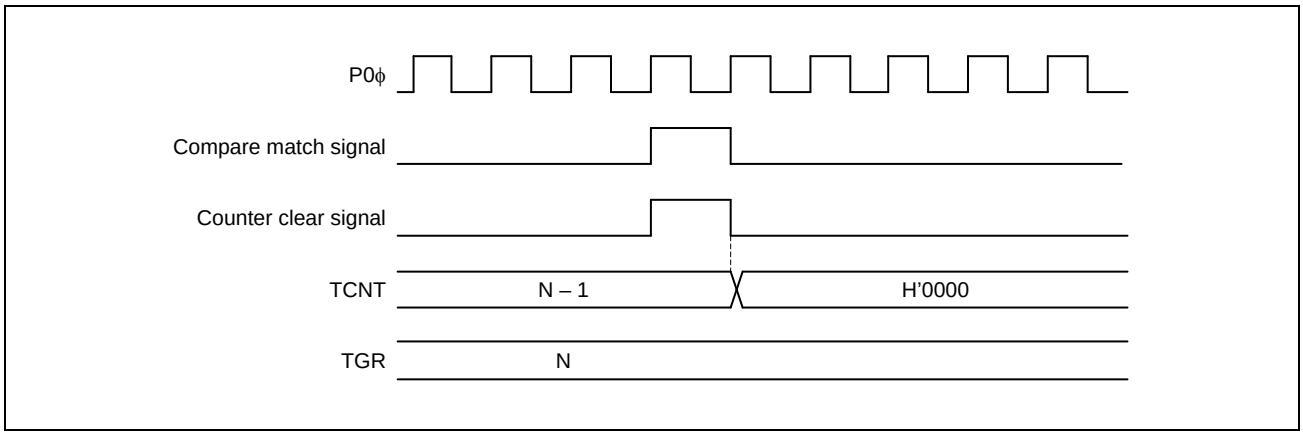


Figure 16.119 Counter Clear Timing (Compare Match) (MTU5)

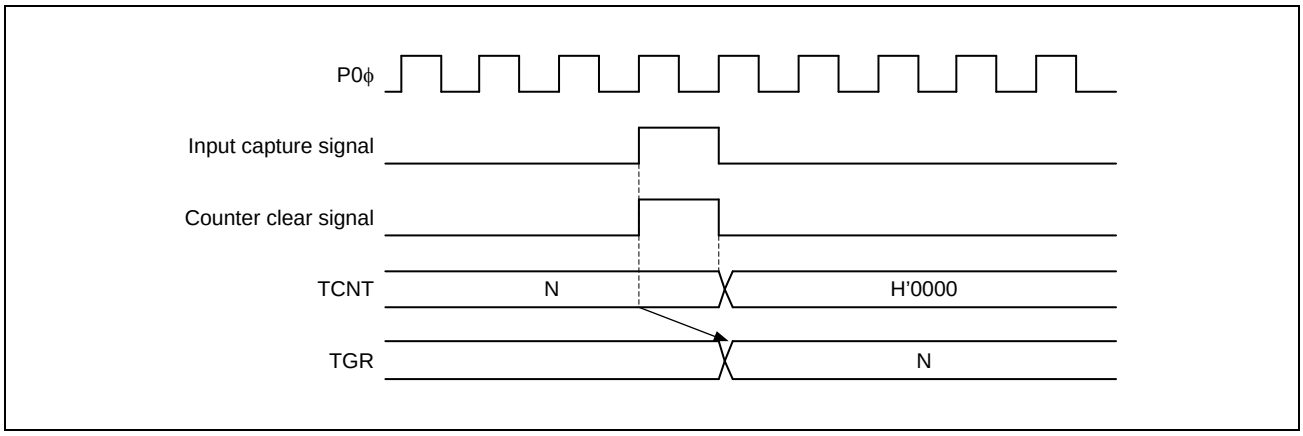


Figure 16.120 Counter Clear Timing (Input Capture) (MTU0 to MTU8)

(5) Buffer Operation Timing

Figure 16.121 to **Figure 16.123** show the timing in buffer operation.

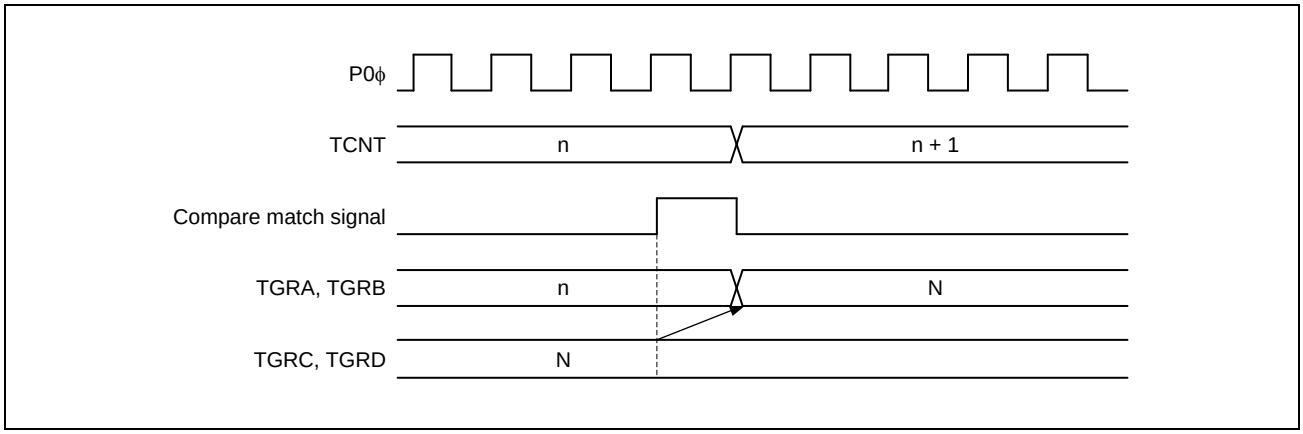


Figure 16.121 Buffer Operation Timing (Compare Match)

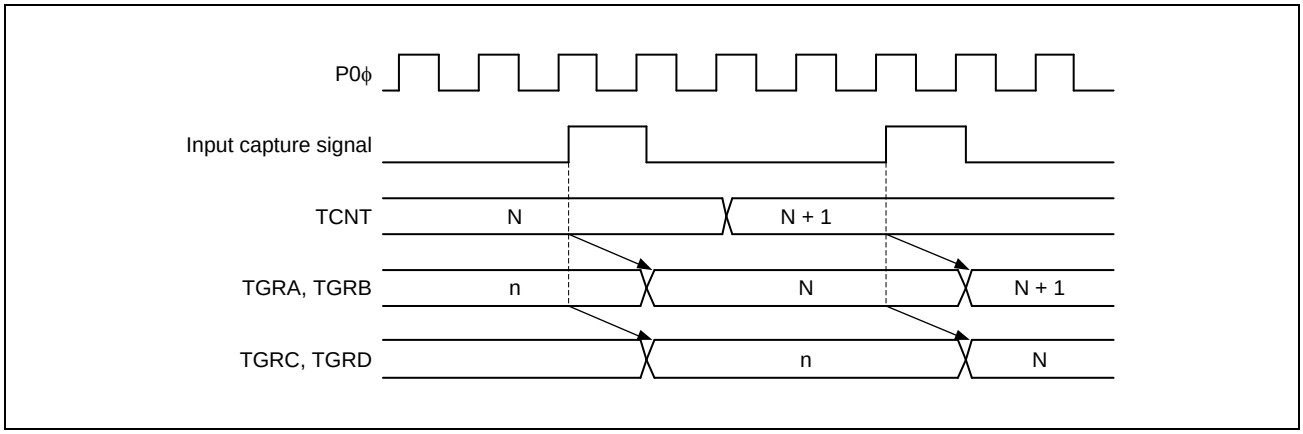


Figure 16.122 Buffer Operation Timing (Input Capture)

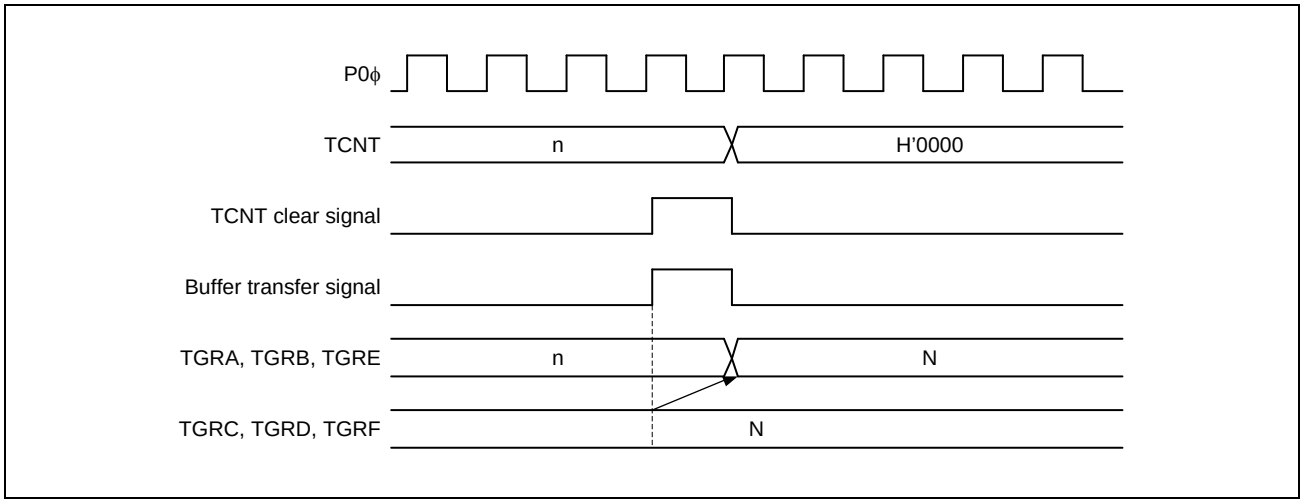


Figure 16.123 Buffer Operation Timing (When TCNT Cleared)

(6) Buffer Transfer Timing (Complementary PWM Mode)

Figure 16.124 to **Figure 16.126** show the buffer transfer timing in complementary PWM mode.

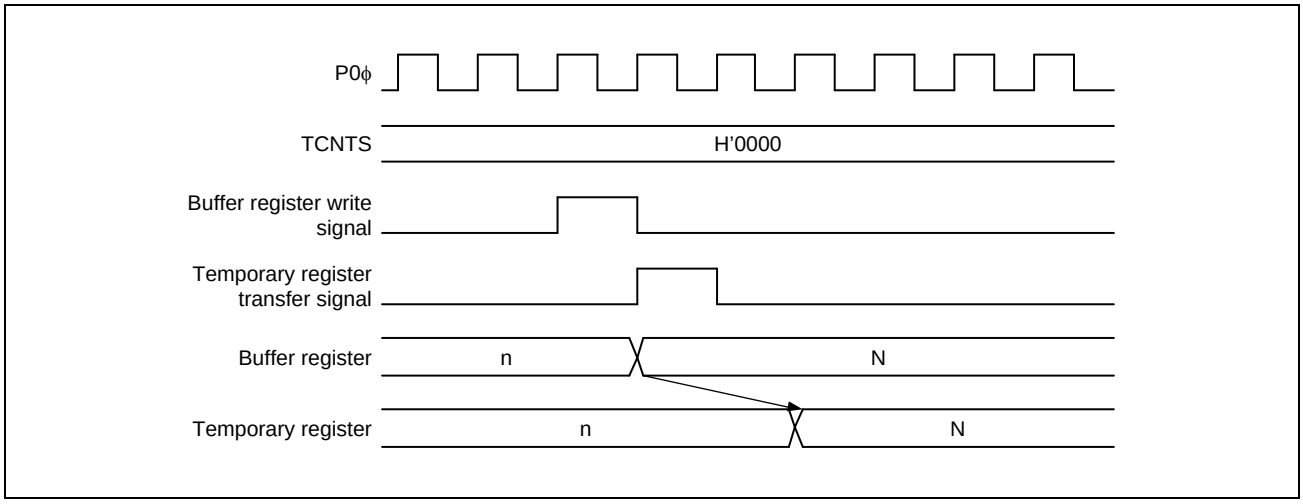


Figure 16.124 Transfer Timing from Buffer Register to Temporary Register (TCNTSA Stopped)

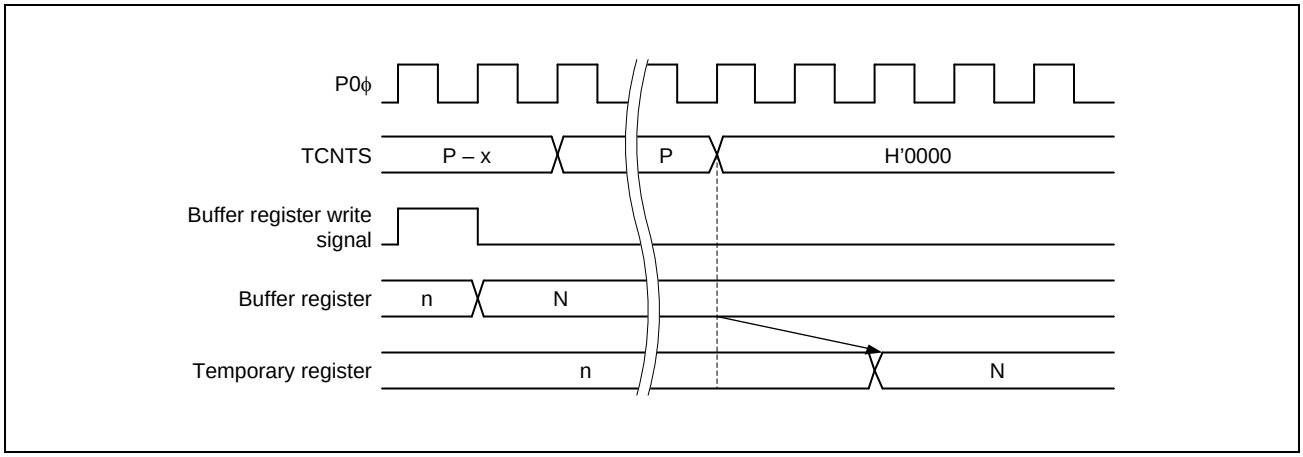


Figure 16.125 Transfer Timing from Buffer Register to Temporary Register (TCNTSA Operating)

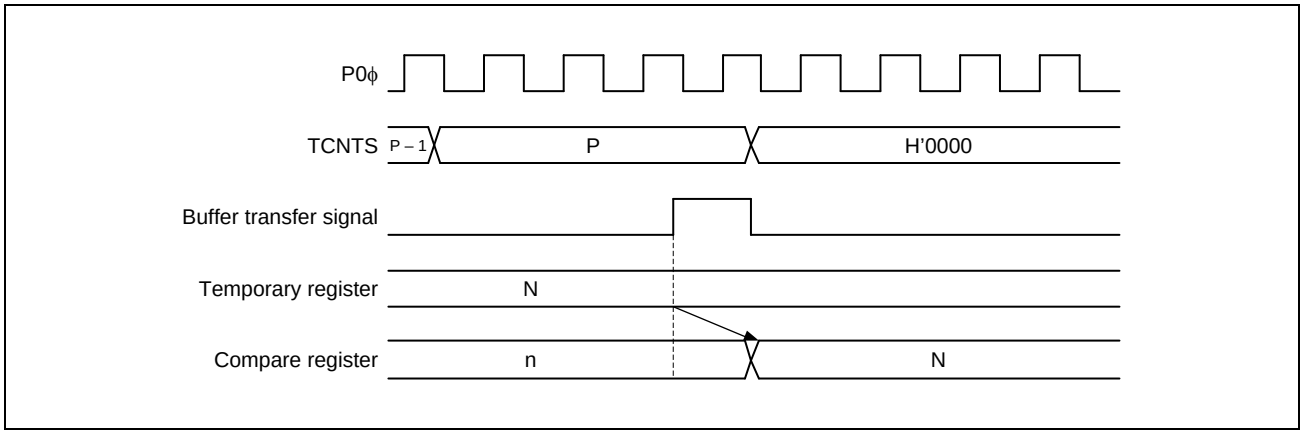


Figure 16.126 Transfer Timing from Temporary Register to Compare Register

16.5.2 Interrupt Signal Timing

(1) TGI Interrupt Timing by Compare Match

Figure 16.127 and **Figure 16.128** show the TGI interrupt request signal timing when a compare match occurs.

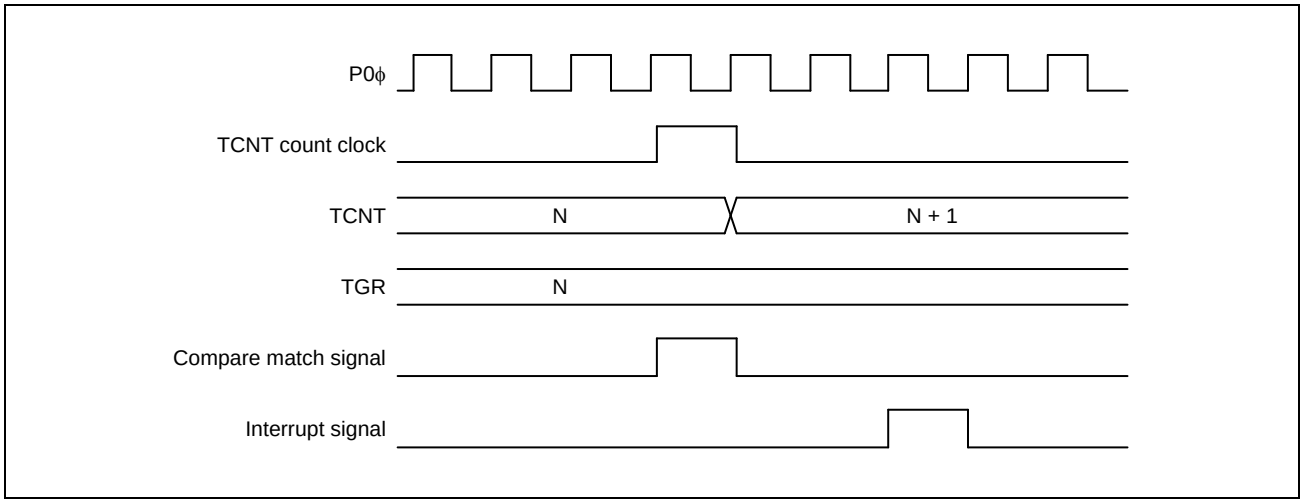


Figure 16.127 TGI Interrupt Timing (Compare Match) (MTU0 to MTU4 and MTU6 to MTU8)

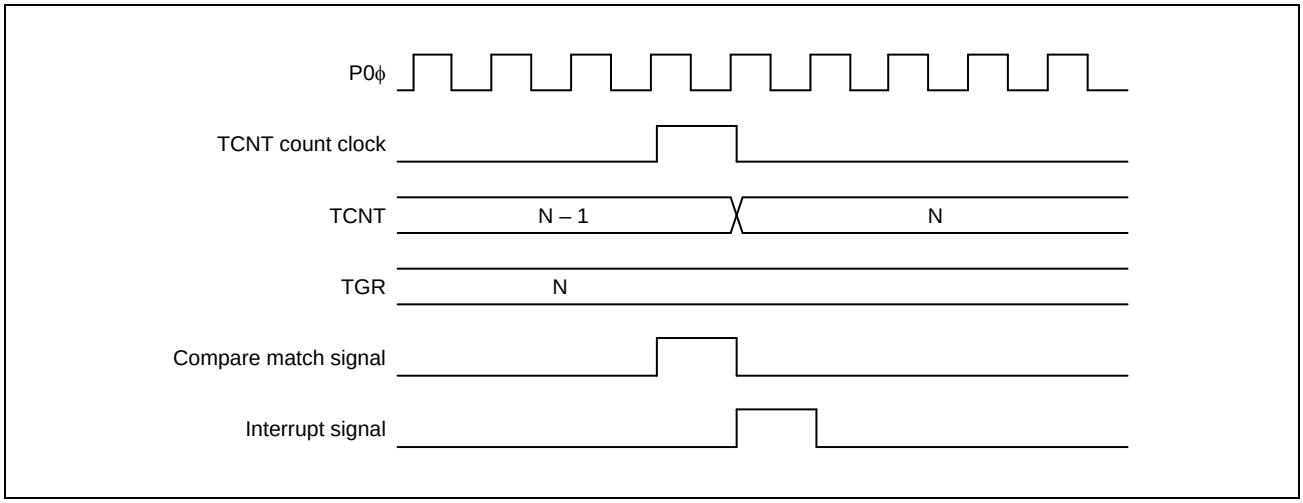


Figure 16.128 TGI Interrupt Timing (Compare Match) (MTU5)

(2) TGI Interrupt Timing by Input Capture

Figure 16.129 and **Figure 16.130** show the TGI interrupt request signal timing when an input capture occurs.

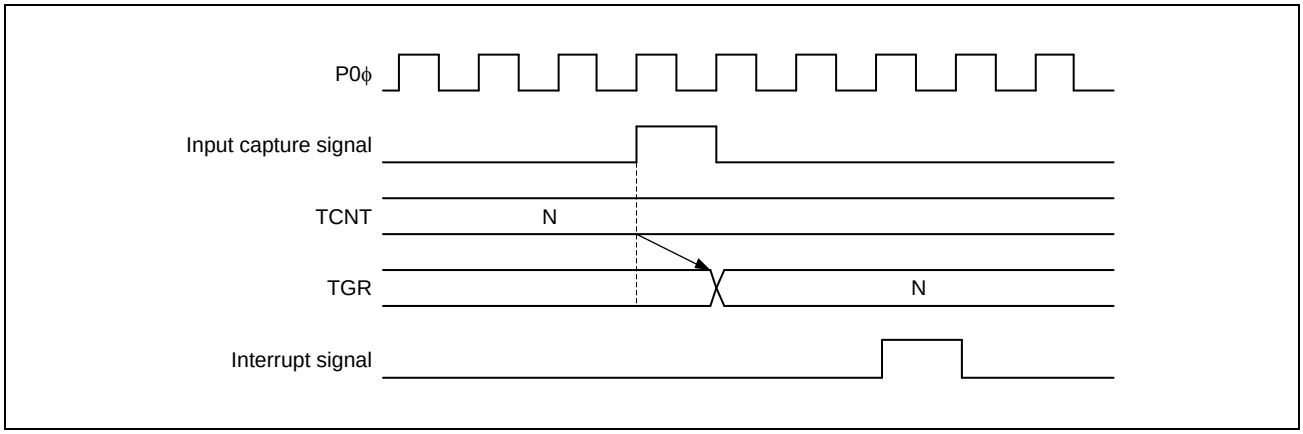


Figure 16.129 TGI Interrupt Timing (Input Capture) (MTU0 to MTU4 and MTU6 to MTU8)

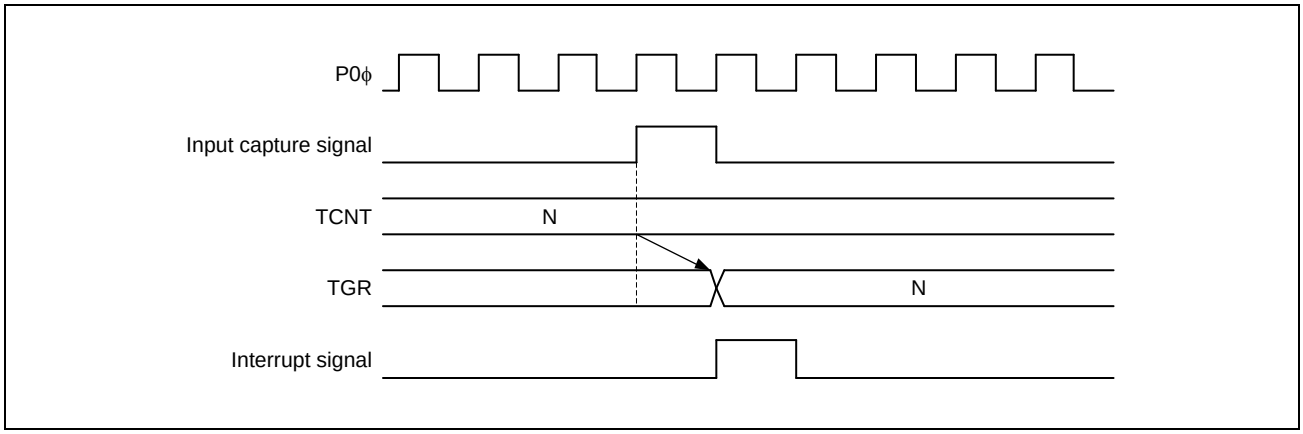


Figure 16.130 TGI Interrupt Timing (Input Capture) (MTU5)

(3) TCFV and TCFU Interrupt Timing

Figure 16.131 shows the TCIV interrupt request signal timing when an overflow is generated.

Figure 16.132 shows the TCIU interrupt request signal timing when an underflow is generated.

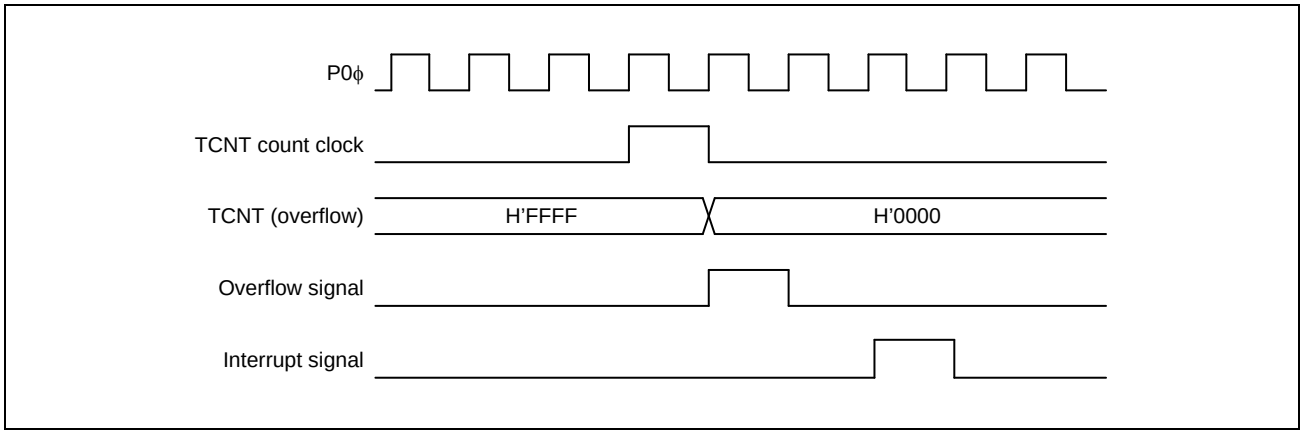


Figure 16.131 TCIV Interrupt Timing

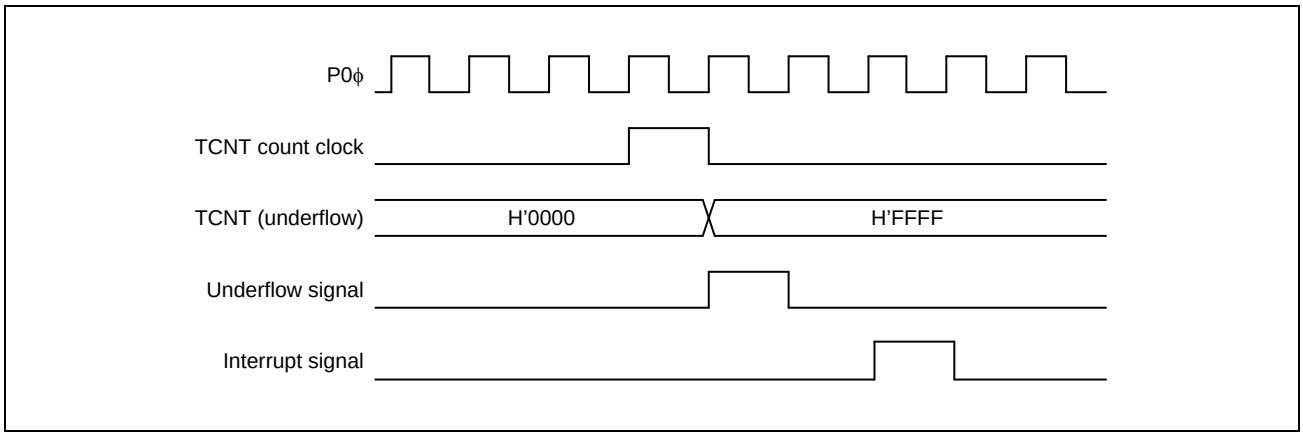


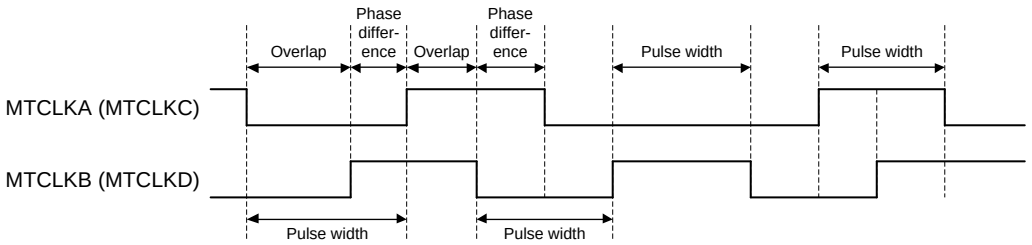
Figure 16.132 TCIU Interrupt Timing

16.6 Usage Note

16.6.1 Count Clock Restrictions

The count clock source pulse width must be at least three P0ϕ clock cycles for single-edge detection, and at least five P0ϕ clock cycles for both-edge detection. The MTU will not operate properly at narrower pulse widths.

In phase counting mode, the phase difference and overlap between the two input clocks must be at least three P0ϕ clock cycles, and the pulse width must be at least 5 P0ϕ clock cycles. **Figure 16.133** shows the input clock conditions in phase counting mode.



Note: Phase difference and overlap: 3 P0ϕ clock cycles or more

Note: Pulse width: 5 P0ϕ clock cycles or more

Figure 16.133 Phase Difference, Overlap, and Pulse Width in Phase Counting Mode

16.6.2 Note on Cycle Setting

When counter clearing on compare match is set, TCNT is cleared in the final state in which it matches the TGR value (the point at which TCNU updates the matched count value). Consequently, the actual counter frequency is given by the following formula:

- MTU0 to MTU4 and MTU6 to MTU8

$$f = \frac{CNTCLK}{N + 1}$$

- MTU5

$$f = \frac{CNTCLK}{N}$$

f: Counter frequency

CNTCLK: The count clock frequency set by TCR.TPSC[2:0] and TCR2.TPSC2[2:0]

N: TGR setting

16.6.3 Contention between TCNT Write and Clear Operations

If the counter clear signal is generated in the TCNT write cycle, TCNT clearing takes precedence and TCNT write operation is not performed.

Figure 16.134 shows the timing in this case.

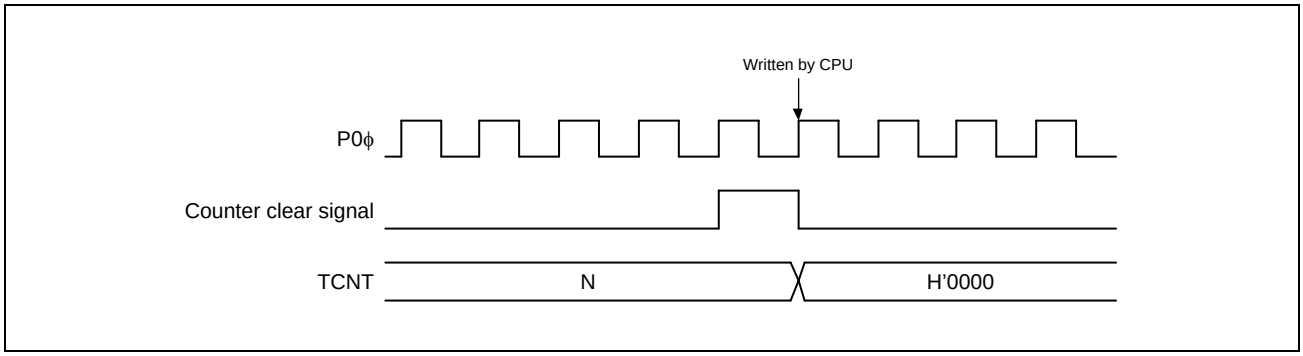


Figure 16.134 Contention between TCNT Write and Clear Operations

16.6.4 Contention between TCNT Write and Increment Operations

If incrementing occurs in a TCNT write cycle, TCNT write operation takes precedence and TCNT is not incremented.

Figure 16.135 shows the timing in this case.

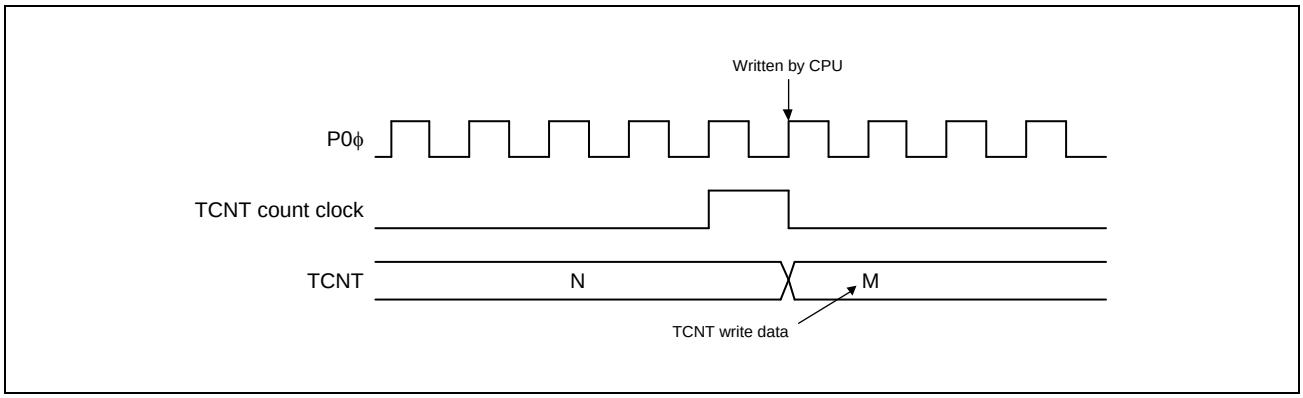


Figure 16.135 Contention between TCNT Write and Increment Operations

16.6.5 Contention between TGR Write Operation and Compare Match

If a compare match occurs in a TGR write cycle, TGR write operation is executed and the compare match signal is also generated.

Figure 16.136 shows the timing in this case.

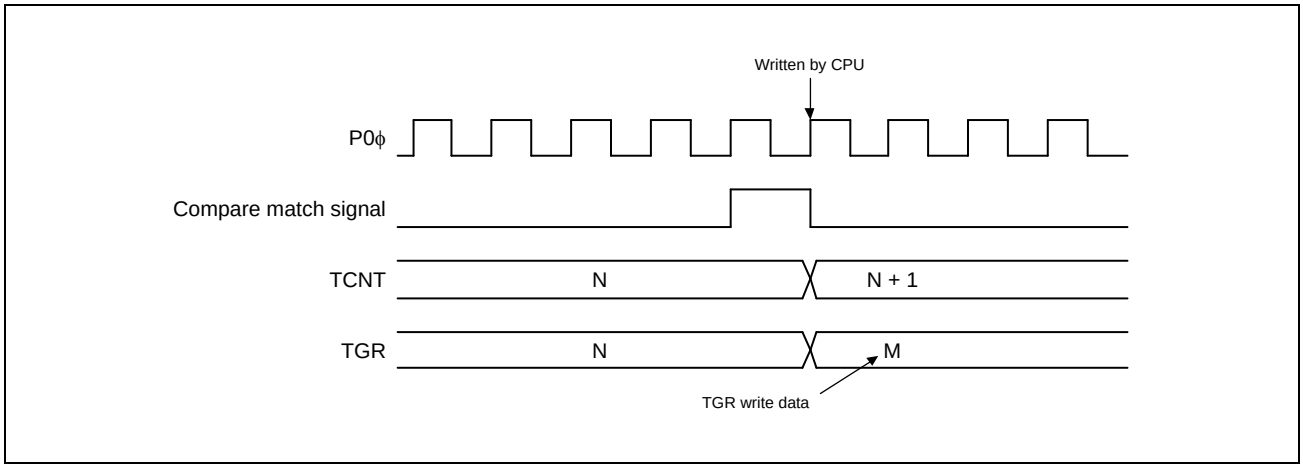


Figure 16.136 Contention between TGR Write Operation and Compare Match

16.6.6 Contention between Buffer Register Write Operation and Compare Match

If a compare match occurs in the T2 state in a TGR write cycle, the data before write operation is transferred to TGR by the buffer operation.

Figure 16.137 shows the timing in this case.

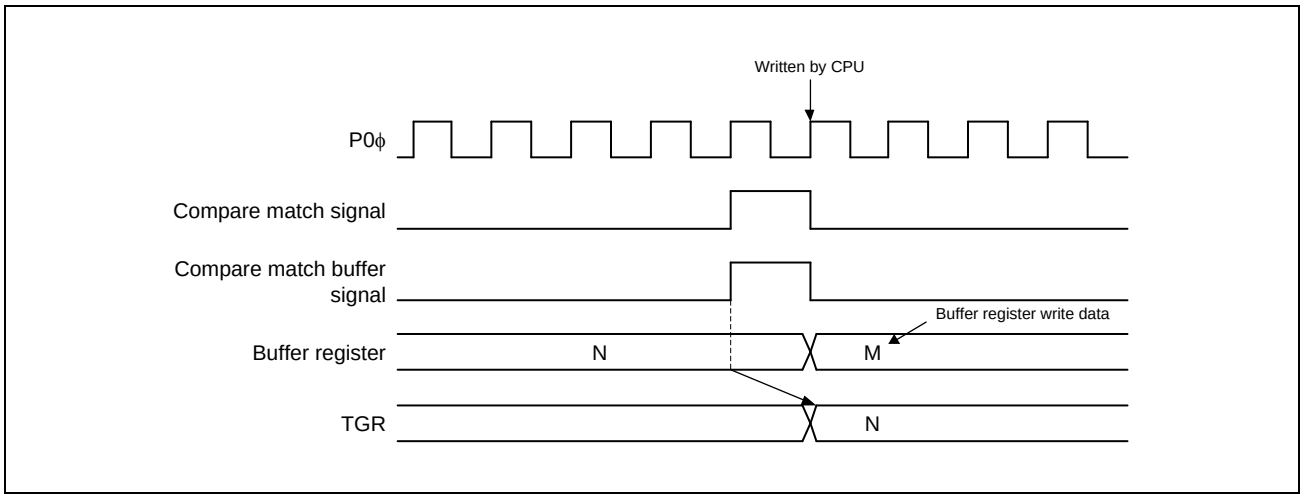


Figure 16.137 Contention between Buffer Register Write Operation and Compare Match

16.6.7 Contention between Buffer Register Write and TCNT Clear Operations

When the buffer transfer timing is set to the TCNT clear timing by the timer buffer transfer mode register (TBTM), if TCNT is cleared during the TGR write cycle, the data before write operation is transferred to TGR by the buffer operation.

Figure 16.138 shows the timing in this case.

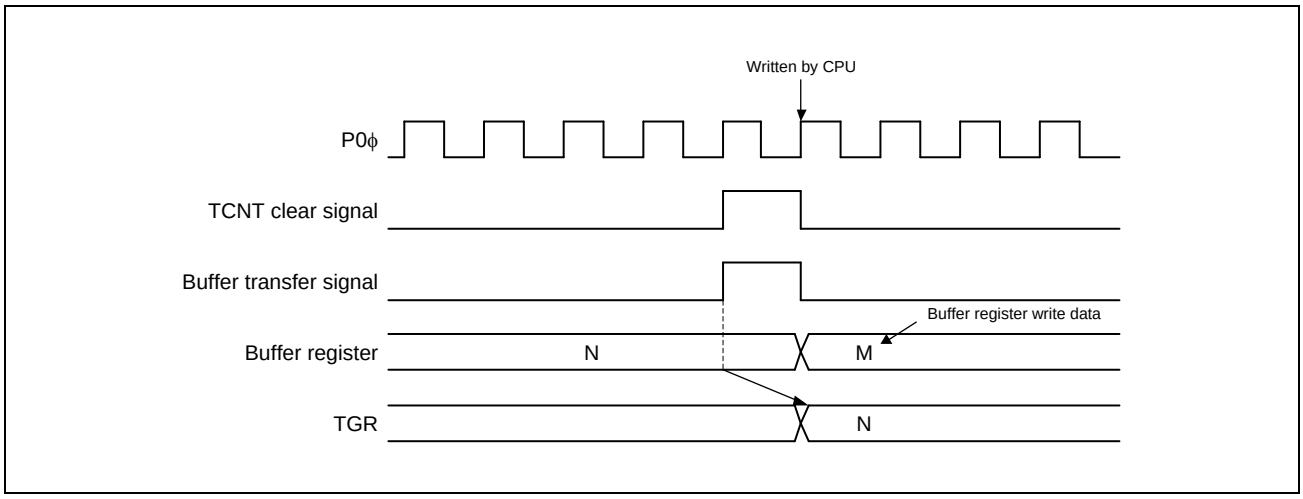


Figure 16.138 Contention between Buffer Register Write and TCNT Clear Operations

16.6.8 Contention between TGR Read Operation and Input Capture

If an input capture signal is generated in a TGR read cycle, the data before input capture transfer is read.

Figure 16.139 shows the timing in this case.

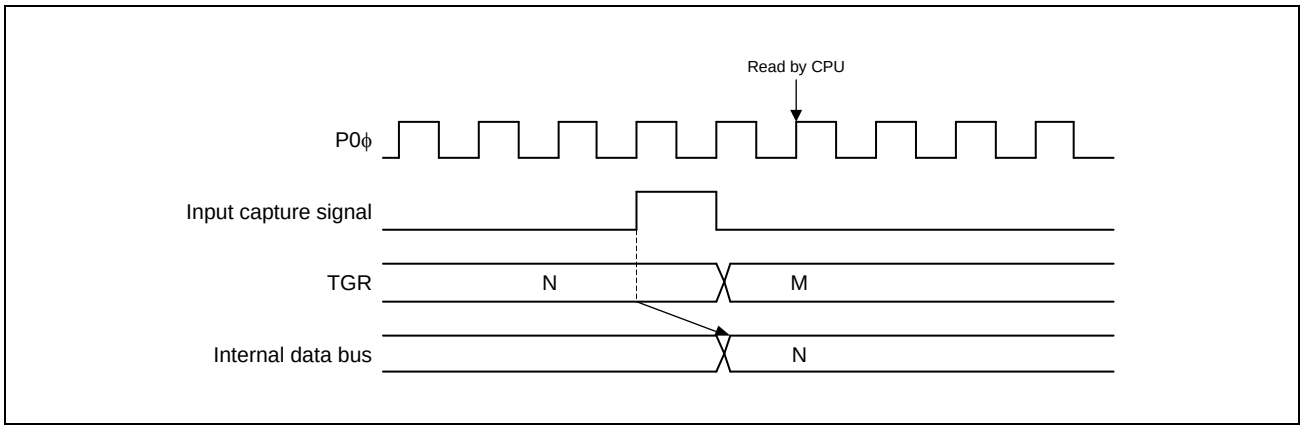


Figure 16.139 Contention between TGR Read Operation and Input Capture (MTU0 to MTU8)

16.6.9 Contention between TGR Write Operation and Input Capture

If an input capture signal is generated in the TGR write cycle, the input capture operation takes precedence and the TGR write operation is not performed in MTU0 to MTU4 and MTU6 to MTU8. In MTU5, the TGR write operation is performed and the input capture signal is generated.

Figure 16.140 and **Figure 16.141** show the timing in this case.

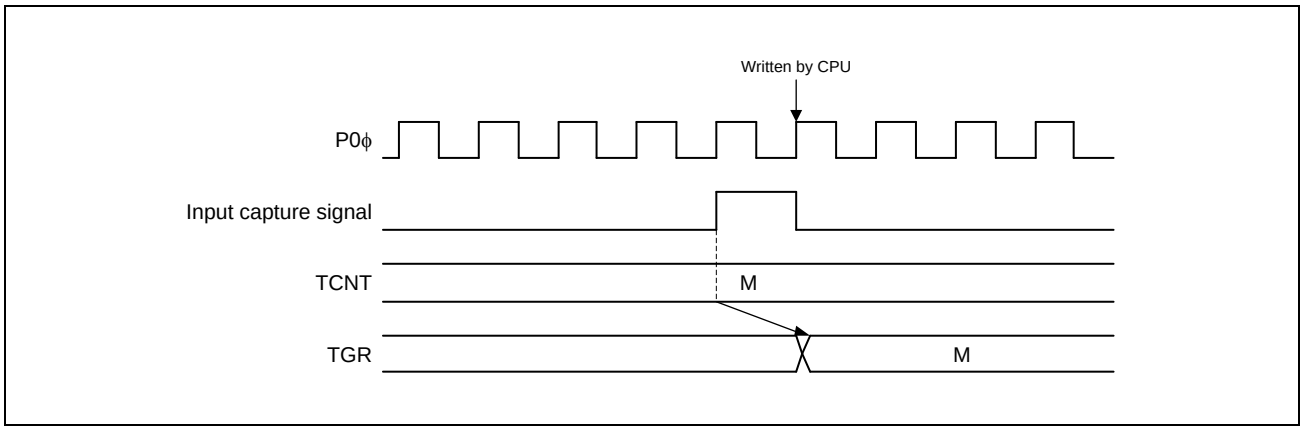


Figure 16.140 Contention between TGR Write Operation and Input Capture (MTU0 to MTU4 and MTU6 to MTU8)

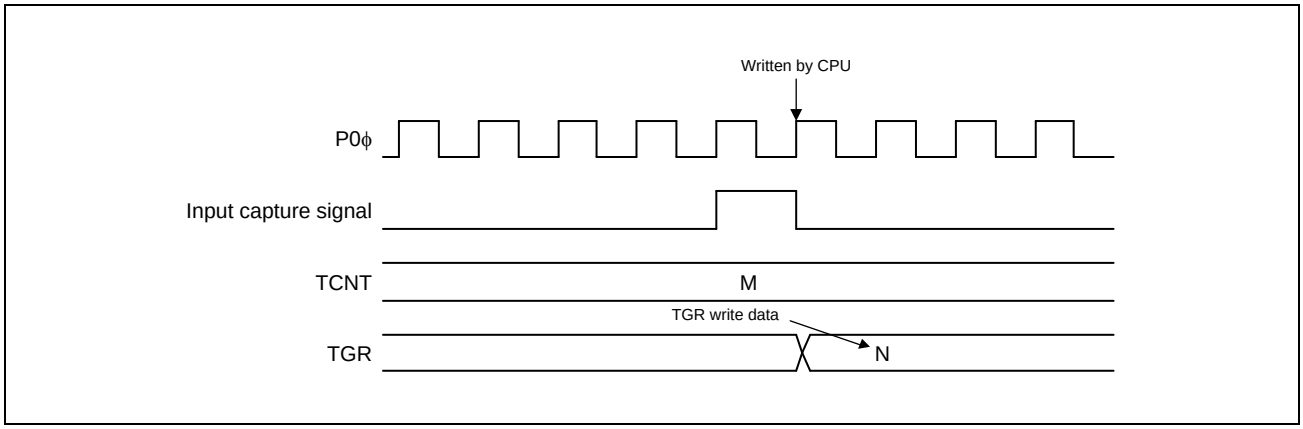


Figure 16.141 Contention between TGR Write Operation and Input Capture (MTU5)

16.6.10 Contention between Buffer Register Write Operation and Input Capture

If an input capture signal is generated in the buffer register write cycle, the buffer operation takes precedence and the buffer register write operation is not performed.

Figure 16.142 shows the timing in this case.

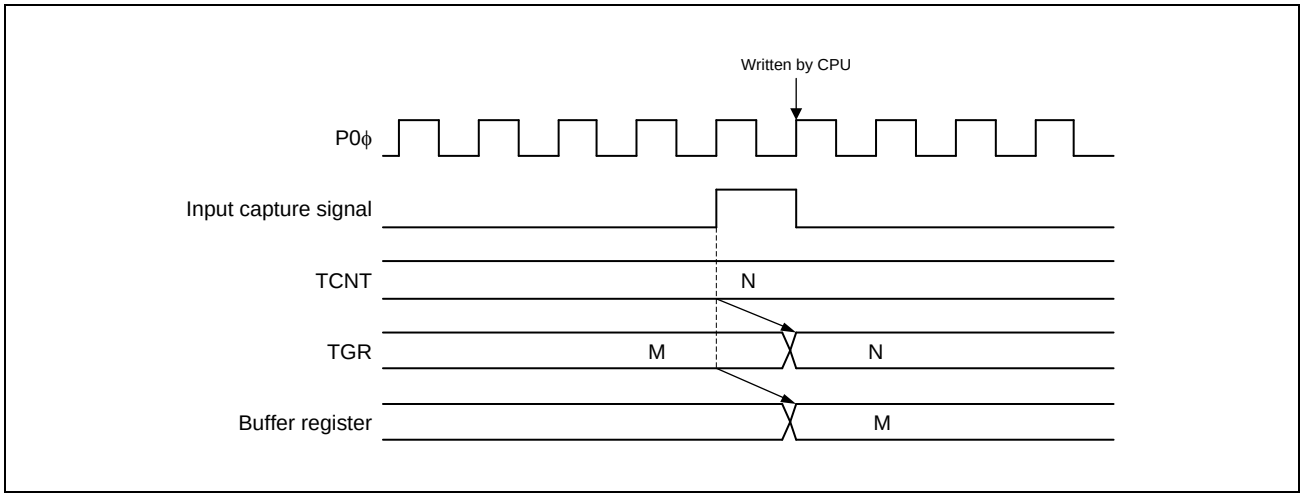


Figure 16.142 Contention between Buffer Register Write Operation and Input Capture

16.6.11 Contention between MTU2.TCNT Write Operation and Overflow/Underflow in Cascaded Operation

With timer counters TCNT in MTU1 and MTU2 cascaded, when a contention occurs between the counting by TCNT in MTU1 (a TCNT overflow/underflow in MTU2) and the writing to TCNT in MTU2, the TCNT write operation is performed in MTU2 and the TCNT count signal in MTU1 is disabled. In this case, if TGRA in MTU1 works as a compare match register and there is a match between the values of TGRA and TCNT in MTU1, a compare match signal is issued.

Furthermore, when the TCNT count clock in MTU1 is selected as the input capture source of MTU0, TGRA to TGRD in MTU0 work in input capture mode. In addition, when the TGRC compare match/input capture in MTU0 is selected as the input capture source of TGRB in MTU1, TGRB in MTU1 works in input capture mode.

Figure 16.143 shows the timing in this case.

When setting the TCNT clearing function in cascaded operation, be sure to synchronize MTU1 and MTU2.

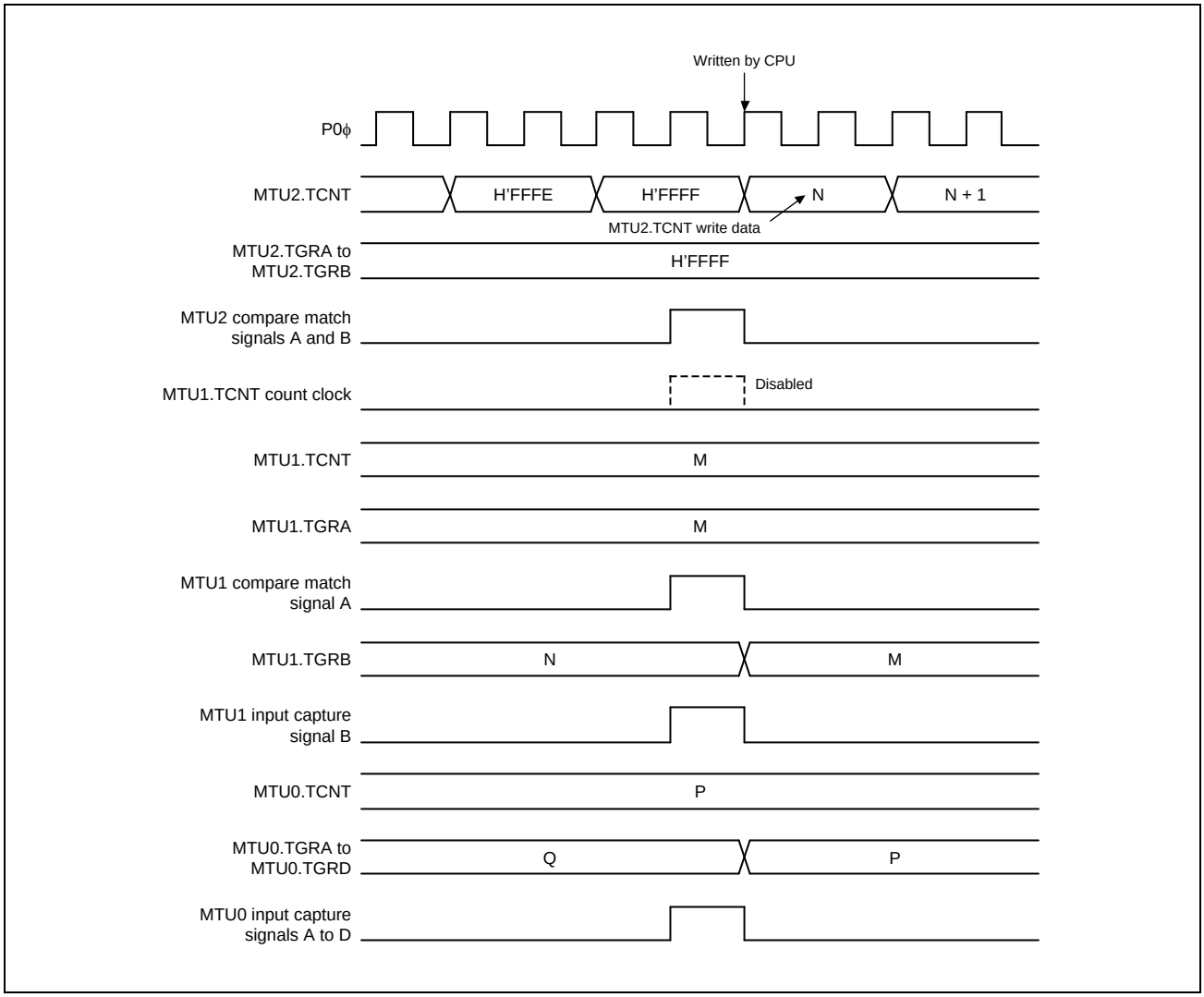


Figure 16.143 Contention between MTU2.TCNT Write Operation and Overflow/Underflow in Cascaded Operation

16.6.12 Counter Value When Count Operation is Stopped in Complementary PWM Mode

When counting operation in MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT) is stopped in complementary PWM mode, the MTU3.TCNT (MTU6.TCNT) value is set to the timer dead time register (TDDRA (TDDRB)) value and MTU4.TCNT (MTU7.TCNT) is set to H’0000.

When operation is restarted in complementary PWM mode, counting begins automatically from the initial setting state.

Figure 16.144 shows this operation.

When counting begins in another operating mode, be sure to make initial settings in MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT).

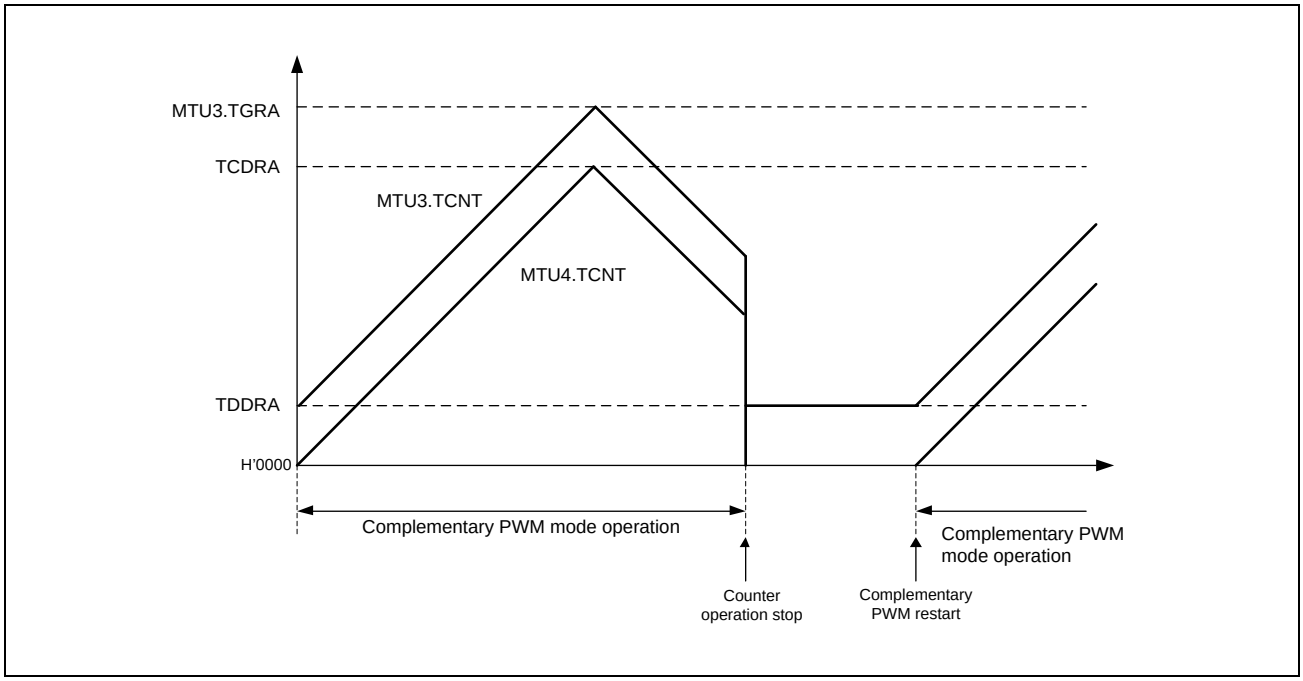


Figure 16.144 Counter Value When Stopped in Complementary PWM Mode

16.6.13 Buffer Operation Setting in Complementary PWM Mode

When modifying the PWM cycle set register (MTU3.TGRA or MTU6.TGRA), timer cycle data register (TCDRA or TCDRB), and duty set registers (MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB (MTU6.TGRB, MTU7.TGRA, and MTU7.TGRB)) in complementary PWM mode, be sure to use buffer operation. In addition, set the BFA and BFB bits in MTU4.TMDR1 (MTU7.TMDR1) to 0. The MTIOC4C (MTIOC7C) pin cannot output waveforms if the BFA bit in MTU4.TMDR1 (MTU7.TMDR1) is set to 1. Likewise, the MTIOC4D (MTIOC7D) pin cannot output waveforms if the BFB bit in MTU4.TMDR1 (MTU7.TMDR1) is set to 1.

In complementary PWM mode, buffer operation in MTU3 and MTU4 (or MTU6 and MTU7) depends on the settings in bits BFA and BFB of MTU3.TMDR1 (MTU6.TMDR1). When the BFA bit in MTU3.TMDR1 (MTU6.TMDR1) is set to 1, MTU3.TGRC (MTU6.TGRC) functions as a buffer register for MTU3.TGRA (MTU6.TGRA). At the same time, MTU4.TGRC (MTU7.TGRC) functions as a buffer register for MTU4.TGRA (MTU7.TGRA), and TCBRA (TCBRB) functions as a buffer register for TCDRA (TCDRB).

16.6.14 Buffer Operation and Compare Match in Reset-Synchronized PWM Mode

When setting buffer operation in reset-synchronized PWM mode, set the BFA and BFB bits in MTU4.TMDR1 (MTU7.TMDR1) to 0. The MTIOC4C (MTIOC7C) pin cannot output waveforms if the BFA bit in MTU4.TMDR1 (MTU7.TMDR1) is set to 1. Likewise, the MTIOC4D (MTIOC7D) pin cannot output waveforms if the BFB bit in MTU4.TMDR1 (MTU7.TMDR1) is set to 1.

In reset-synchronized PWM mode, buffer operation in MTU3 and MTU4 (or MTU 6 and MTU7) depends on the settings in the BFA and BFB bits of MTU3.TMDR1 (MTU6.TMDR1). For example, if the BFA bit in MTU3.TMDR1 (MTU6.TMDR1) is set to 1, MTU3.TGRC (MTU6.TGRC) functions as a buffer register for MTU3.TGRA (MTU6.TGRA). At the same time, MTU4.TGRC (MTU7.TGRC) functions as a buffer register for MTU4.TGRA (MTU7.TGRA).

While the MTU3.TGRC (MTU6.TGRC) and MTU3.TGRD (MTU6.TGRD) are operating as buffer registers, a TGImn interrupt (m = C or D; n = 3, 4, 6, or 7) is not generated.

Figure 16.145 shows an example of MTU3.TGR (MTU6.TGR), MTU4.TGR (MTU7.TGR), MTIOC3 (MTIOC6), and MTIOC4 (MTIOC7) operation with the BFA and BFB bits in MTU3.TMDR1 (MTU6.TMDR1) set to 1 and the BFA and BFB bits in MTU4.TMDR1 (MTU7.TMDR1) set to 0.

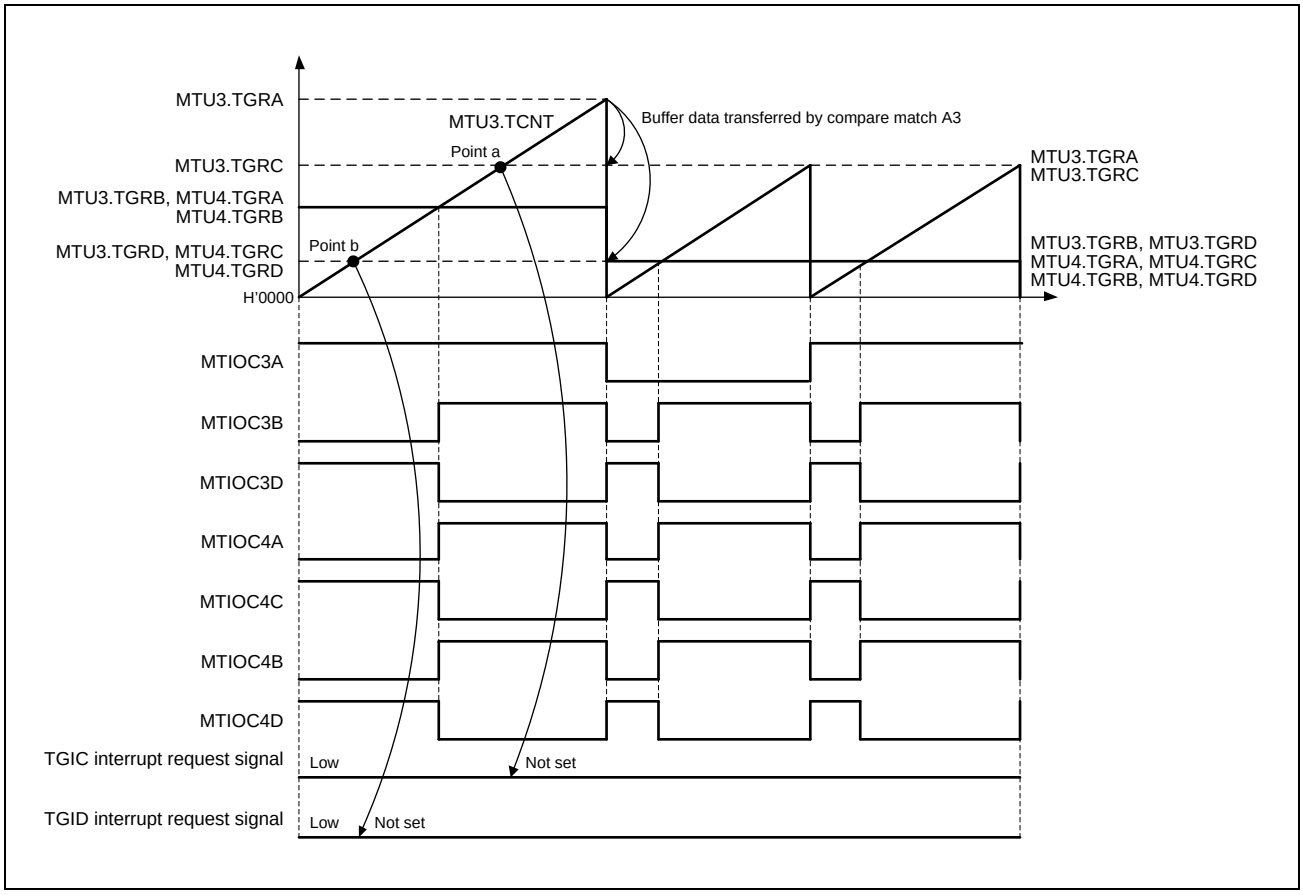


Figure 16.145 Buffer Operation and Compare Match in Reset-Synchronized PWM Mode

16.6.15 Overflow in Reset-Synchronized PWM Mode

After reset-synchronized PWM mode is selected, the TCNT counters in MTU3 and MTU4 (MTU6 and MTU7) start counting when the CST3 (CST6) bit of TSTRA (TSTRB) is set to 1. In this state, the TCNT count clock source and count edge in MTU4 (MTU7) are determined by the setting of TCR in MTU3 (MTU6).

In reset-synchronized PWM mode, with cycle register TGRA in MTU3 (MTU6) set to H'FFFF and the TGRA compare match in MTU3 (MTU6) selected as the counter clearing source, the TCNT counters in MTU3 and MTU4 (MTU6 and MTU7) count up to H'FFFF, then a compare match occurs with TGRA in MTU3 (MTU6), and the TCNT counters in MTU3 and MTU4 (MTU6 and MTU7) are both cleared. In this case, a TCIVn interrupt (n = 3, 4, 6, or 7) is not generated.

Figure 16.146 shows an example of the operation in reset-synchronized PWM mode with cycle register TGRA in MTU3 (MTU6) set to H'FFFF and the TGRA compare match in MTU3 (MTU6) specified for the counter clearing source.

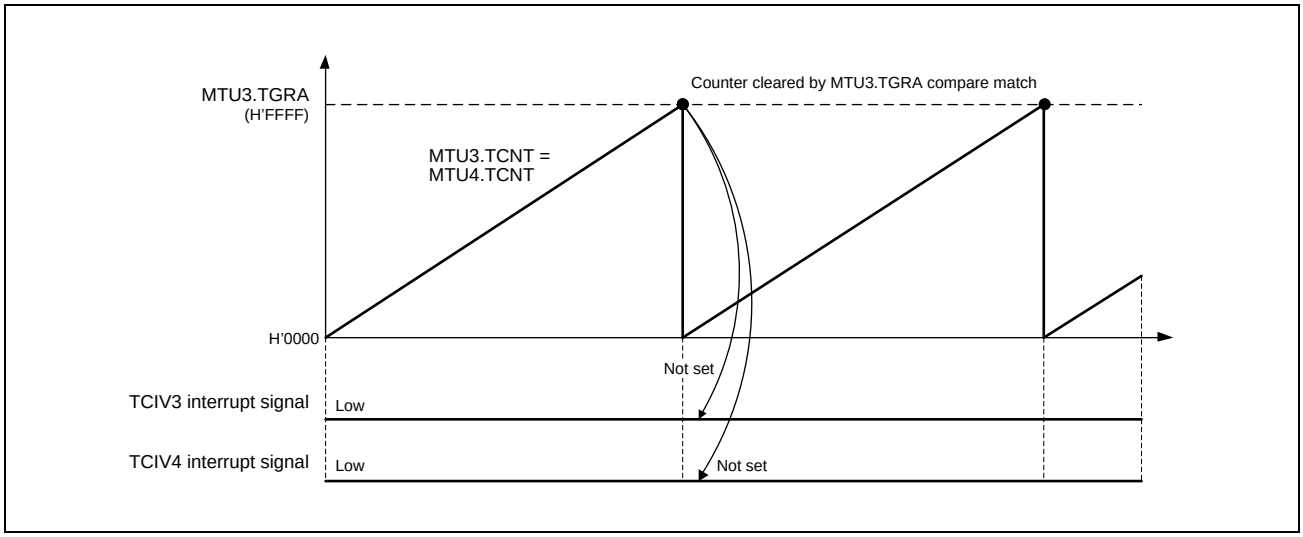


Figure 16.146 Overflow in Reset-Synchronized PWM Mode

16.6.16 Contention between Overflow/Underflow and Counter Clearing

If an overflow/underflow and counter clearing occur simultaneously, a TCIVn interrupt (n = 0 to 4 or 6 to 8) nor a TCIUn interrupt (n = 1 or 2) is not generated and TCNT clearing takes precedence.

Figure 16.147 shows the operation timing when a TGR compare match is specified as the clearing source and TGR is set to H'FFFF.

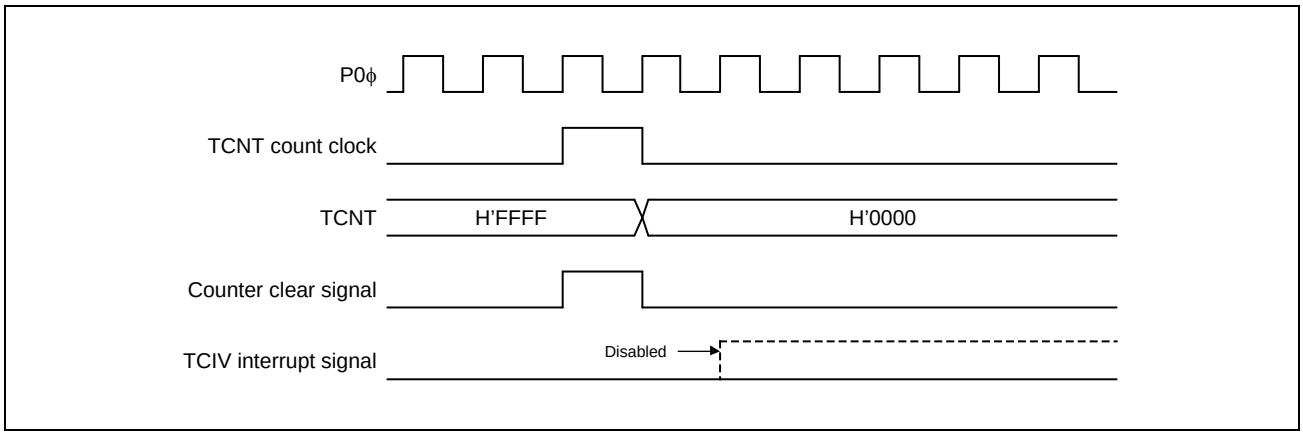


Figure 16.147 Contention between Overflow and Counter Clearing

16.6.17 Contention between TCNT Write Operation and Overflow/Underflow

If TCNT counts up or down in a TCNT write cycle and an overflow or an underflow occurs, the TCNT write operation takes precedence. A TCIVn interrupt (n = 0 to 4 or 6 to 8) nor a TCIUn interrupt (n = 1 or 2) is not generated.

Figure 16.148 shows the operation timing when there is contention between TCNT write operation and overflow.

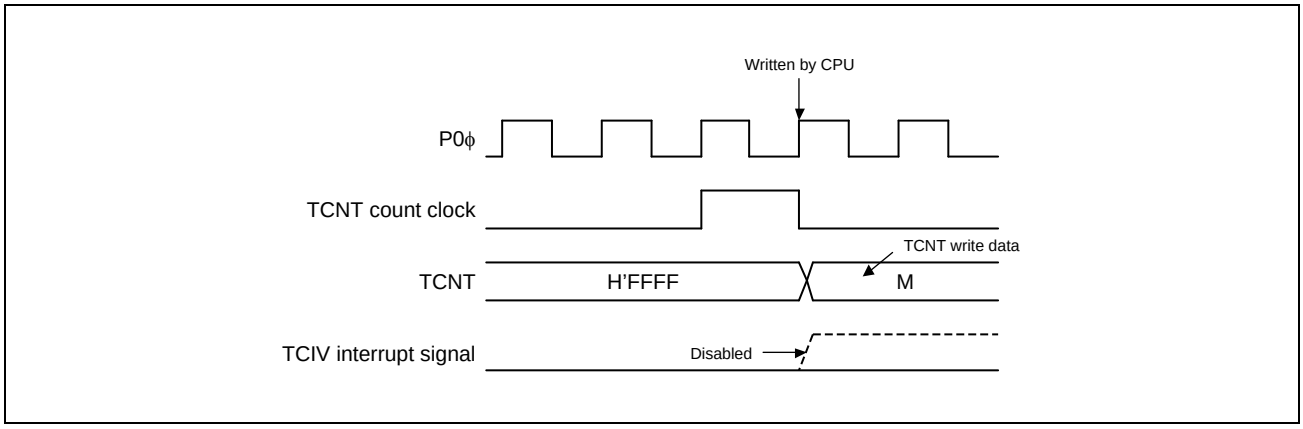


Figure 16.148 Contention between TCNT Write Operation and Overflow

16.6.18 Note on Transition from Normal Mode or PWM Mode 1 to Reset-Synchronized PWM Mode

When making a transition from normal mode or PWM mode 1 to reset-synchronized PWM mode in MTU3 and MTU4 (or MTU6 and MTU7), if the counter is stopped while the output pins (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4C, MTIOC4B, MTIOC4D, MTIOC6B, MTIOC6D, MTIOC7A, MTIOC7C, MTIOC7B, and MTIOC7D) are held at a high level and then operation is started after a transition to reset-synchronized PWM mode, the initial pin output will not be correct.

When making a transition from normal mode to reset-synchronized PWM mode, write H'11 to MTU3.TIORH, MTU3.TIORL, MTU4.TIORH, and MTU4.TIORL (MTU6.TIORH, MTU6.TIORL, MTU7.TIORH, and MTU7.TIORL) to initialize the output pin state to a low level, then set the registers to the initial value (H'00) before making the mode transition.

When making a transition from PWM mode 1 to reset-synchronized PWM mode, switch to normal mode, initialize the output pin state to a low level, and then set the registers to the initial value (H'00) before making the transition to reset-synchronized PWM mode.

16.6.19 Output Level in Complementary PWM Mode and Reset-Synchronized PWM Mode

When MTU3 and MTU4 (or MTU6 and MTU7) are in complementary PWM mode or reset-synchronized PWM mode, the PWM waveform output level is determined by the OLSP and OLSN bits in the timer output control register (TOCR1A or TOCR1B). In complementary PWM mode or reset-synchronized PWM mode, TIOR should be set to H'00. The output level in negative phase when the TDER bit in TDERA (TDERB) is set to 0 in complementary PWM mode (the dead time is not generated) does not depend on the setting of the OLSN bit in TOCR1A (TOCR1B). It is equivalent to the inverted level of positive phase output based on the setting of the OLSP bit in TOCR1A (TOCR1B).

16.6.20 Simultaneous Input Capture in MTU1.TCNT and MTU2.TCNT in Cascade Connection

When timer counters 1 and 2 (MTU1.TCNT and MTU2.TCNT) operate as a 32-bit counter in cascade connection, the cascaded counter value cannot be captured successfully in some cases even if input-capture input is simultaneously done to MTIOC1A and MTIOC2A or to MTIOC1B and MTIOC2B. This is because the input timing of MTIOC1A and MTIOC2A or to MTIOC1B and MTIOC2B may not be the same when external input-capture signals input into MTU1.TCNT and MTU2.TCNT are taken in synchronization with the internal clock.

For example, MTU1.TCNT (the counter for upper 16 bits) does not capture the count-up value by an overflow from MTU2.TCNT (the counter for lower 16 bits) but captures the count value before the up-counting. In this case, the values of MTU1.TCNT = H'FFF1 and MTU2.TCNT = H'0000 should be transferred to MTU1.TGRA and MTU2.TGRA or to MTU1.TGRB and MTU2.TGRB, but the values of MTU1.TCNT = H'FFF0 and MTU2.TCNT = H'0000 are erroneously transferred.

The MTU has a function that allows simultaneous capture of MTU1.TCNT and MTU2.TCNT with a single input capture input. This function can be used to read the 32-bit counter such that MTU1.TCNT and MTU2.TCNT are captured at the same time. For details, see **Section 16.2.11, Timer Input Capture Control Register (TICCR)**.

16.6.21 Interrupt-Skipping Function 2

When interrupt-skipping function 2 is in use and the difference between the values in MTU4.TADCORA and MTU4.TADCORB is small, correct counting of the number skipped may not be possible, in which case requests for A/D conversion will not be generated with the expected timing. The conditions listed below thus apply to these settings.

For MTU6 and MTU7, the same conditions apply to the settings of MTU7.TADCORA and MTU7.TADCORB.

(1) When the number skipped is zero for skipping function 2

- The difference between the values in MTU4.TADCORA and MTU4.TADCORB must be at least four.
- The interval of comparison for MTU4.TADCORA must be at least four cycles of P0ϕ clock (the updated value of MTU4.TADCORA is set to the previous value plus or minus at least four).
- The interval of comparison for MTU4.TADCORB must be at least four cycles of P0ϕ clock (the updated value of MTU4.TADCORB is set to the previous value plus or minus at least four).

(2) When the number skipped is one or more for skipping function 2

- The difference between the values in MTU4.TADCORA and MTU4.TADCORB must be at least two.
- The interval of comparison for MTU4.TADCORB must be at least two cycles of P0ϕ (the updated value of MTU4.TADCORB is set to the previous value plus or minus at least two).

16.6.22 Notes When Complementary PWM Mode Output Protection Function is Not Used

The complementary PWM mode output protection function is initially enabled. For details, refer to **Section 17, Port Output Enable 3 (POE3)**.

16.6.23 Notes Regarding Timer Counter (MTU5.TCNT) and Timer General Register (MTU5.TGR)

Do not set TGRj (j = U, V, or W) in MTU5 to the value of the corresponding TCNTj in MTU5 plus one while counting by TCNTj in MTU5 is stopped. If such a setting is made, a compare match will be generated even though counting is stopped.

In this case, if the value of the compare match enable bit (TGIE5j bit of TIER in MTU5) is 1 (enabled), a compare match interrupt will also be generated. If the value of the timer compare match clear register is 1 (enabled), the TCNTj counter in MTU5 is automatically cleared to H’0000 when the compare match is generated, regardless of whether compare match interrupts are enabled or disabled.

16.6.24 Notes to Prevent Malfunctions in Synchronous Clearing for Complementary PWM Mode

If control of the output waveform is enabled (the WRE bit of TWCRA or TWCRB is 1) at the time of synchronous counter clearing in complementary PWM mode, satisfaction of either condition 1 or 2 below has the following effects.

- Dead time on the PWM output pins is shortened (or disappears).
- The active level is output on the negative phase PWM output pins beyond the period for active-level output.

Condition 1:

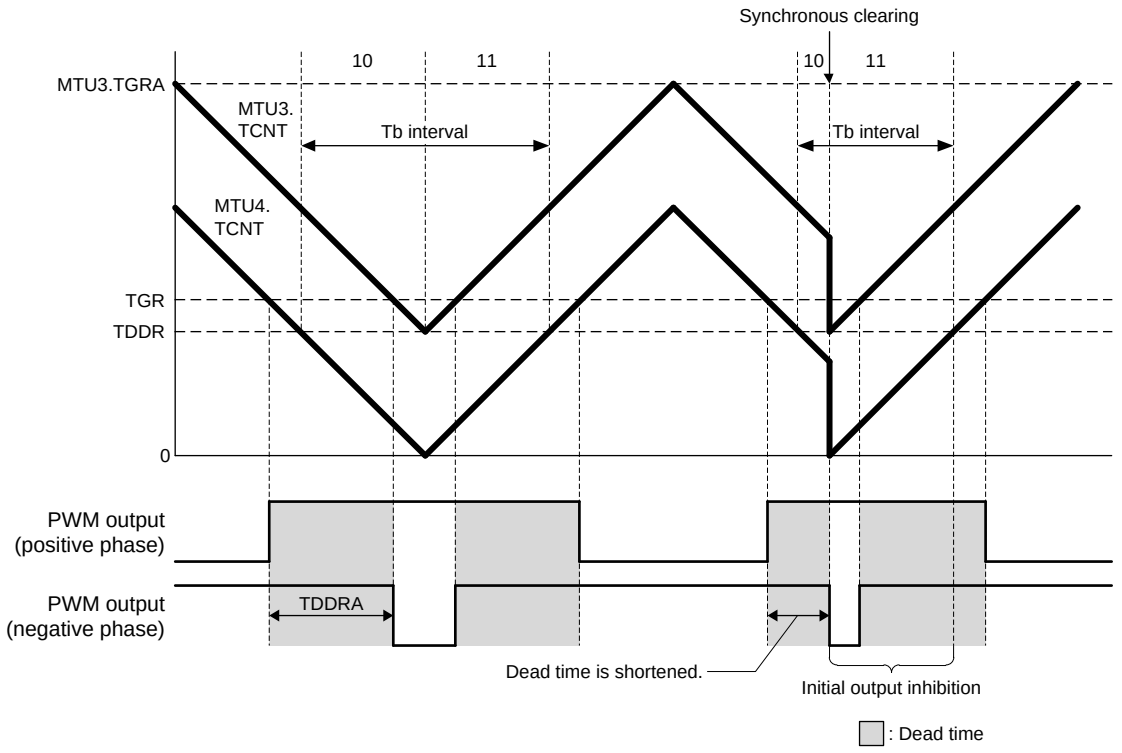
In portion (10) of the initial output inhibition period in **Figure 16.149**, synchronous clearing occurs within the dead-time period for PWM output.

Condition 2:

In portions (10) and (11) of the initial output inhibition period in **Figure 16.150**, synchronous clearing occurs when any condition from among TGRB in MTU3 (MTU6) ≤ TDDR (TDDRB), TGRA in MTU4 (MTU7) ≤ TDDR (TDDRB), and TGRB in MTU4 (MTU7) ≤ TDDR (TDDRB) is satisfied.

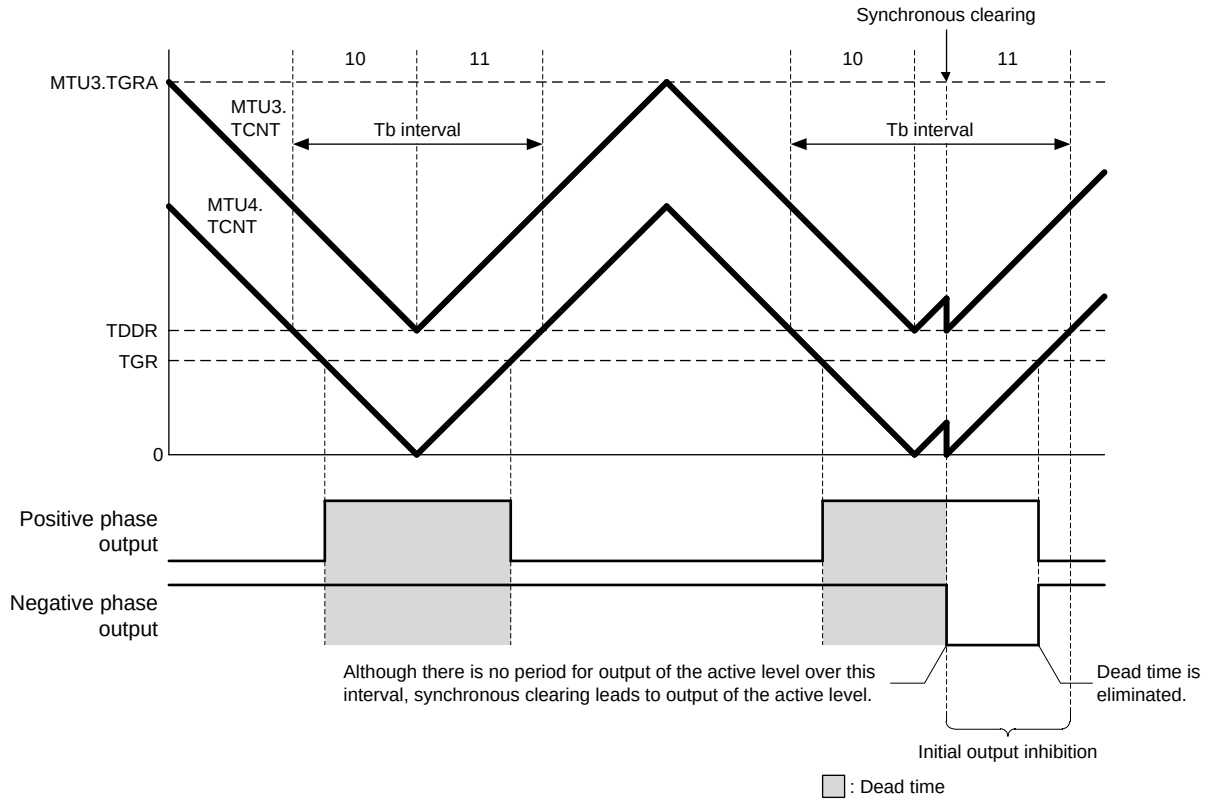
The following method avoids the above phenomena.

Ensure that synchronous clearing proceeds with the value of each comparison register (TGRB in MTU3 (MTU6), TGRA in MTU4 (MTU7), and TGRB in MTU4 (MTU7)) set to at least double the value of the TDDRA register (TDDRB register).



Note: PWM output is active low.

Figure 16.149 Example of Synchronous Clearing (When Condition 1 Applies)



Note: PWM output is active low.

Figure 16.150 Example of Synchronous Clearing (When Condition 2 Applies)

16.6.25 Continuous Output of Interrupt Signal in Response to a Compare Match

When the TGR register is set to H’0000, the P0ϕ/1 clock is set as the count clock, and compare match is set as the trigger for clearing of the count clock, the value of the TCNT counter remains H’0000, and the interrupt signal will be output continuously (i.e. its level will be flat) rather than output over a single cycle. Consequently, interrupts will not be detected in response to second and subsequent compare matches.

Figure 16.151 shows the timing for continuous output of the interrupt signal in response to a compare match.

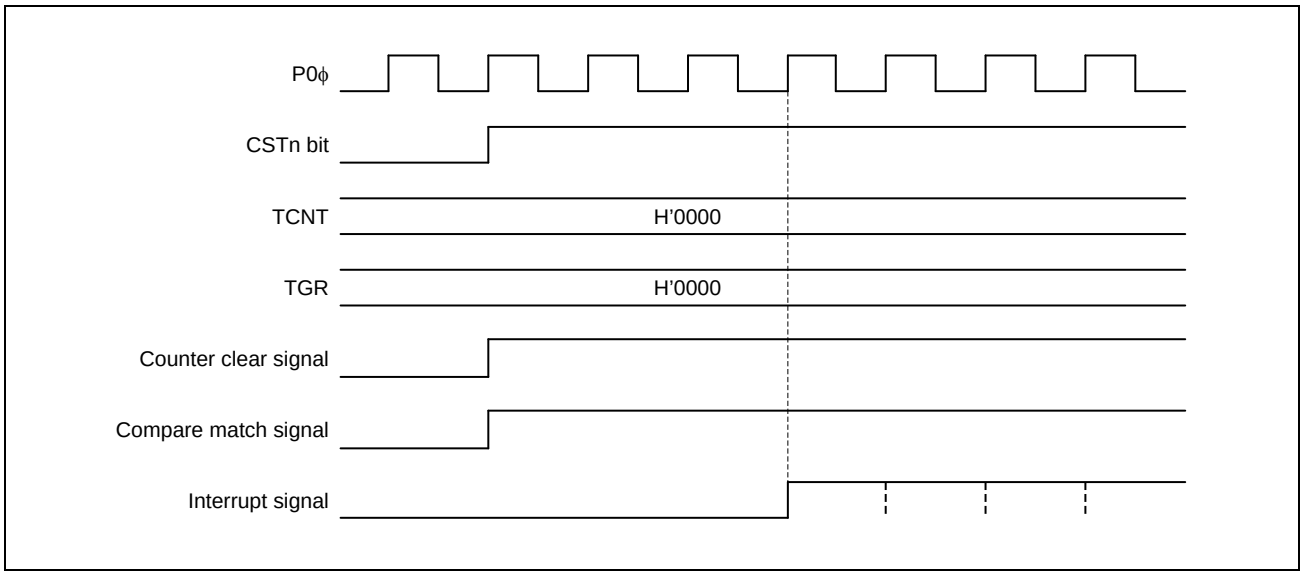
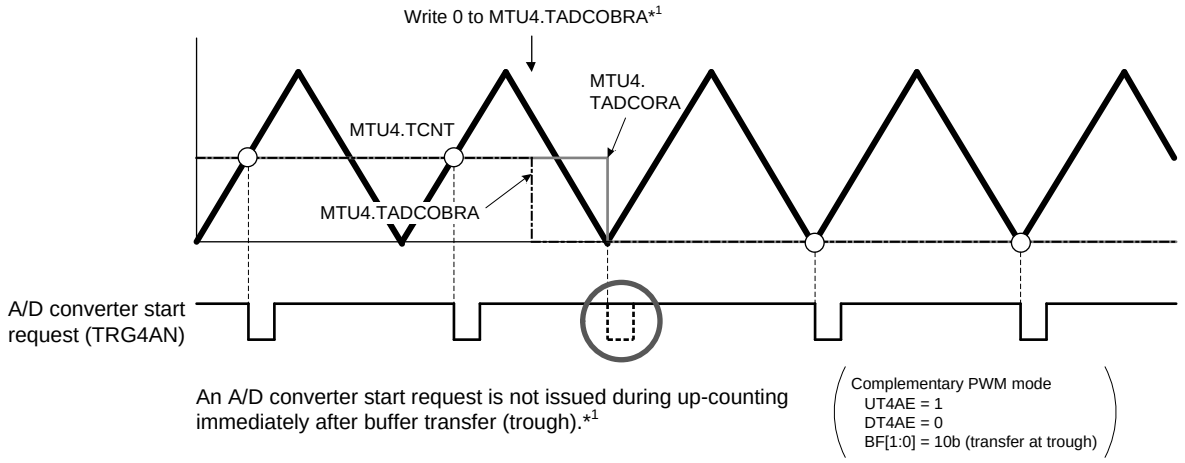


Figure 16.151 Continuous Output of Interrupt Signal in Response to a Compare Match

16.6.26 Notes on A/D Converter Start Request Delaying Function in Complementary PWM Mode

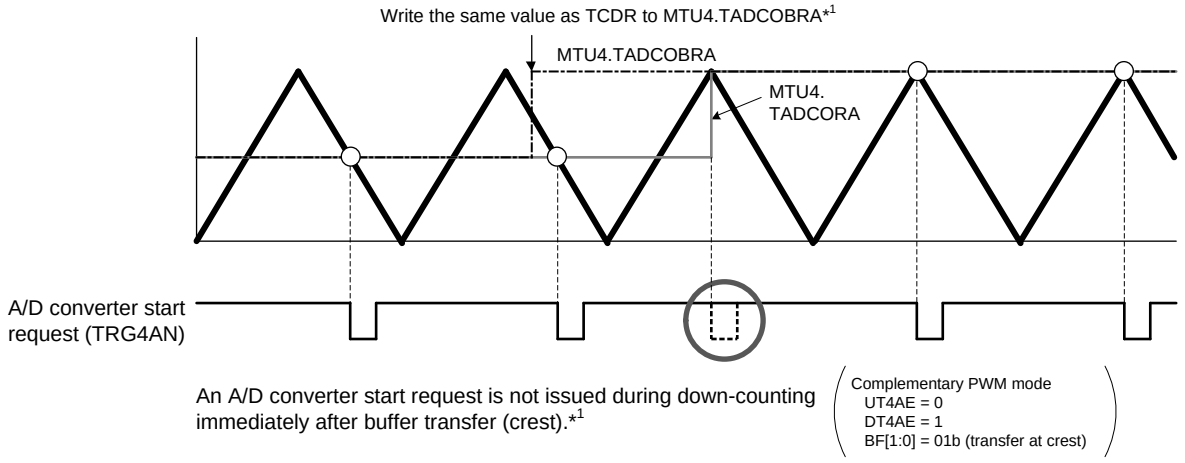
- When the TADCOBRA or TADCOBRB register in MTU4 (MTU7) is set to 0, the UT4AE or UT4BE (UT7AE or UT7BE) bit of TADCR in MTU4 (MTU7) is set to 1, and buffer transfer is performed at the trough of the value of the TCNT counter in MTU4 (MTU7), an A/D converter start request is not issued during up-counting immediately after the transfer (**Figure 16.152**).
- When the TADCOBRA or TADCOBRB register in MTU4 (MTU7) is set to the same value as that in the TCDR register, the DT4AE or DT4BE (DT7AE or DT7BE) bit of TADCR in MTU4 (MTU7) is set to 1, and buffer transfer is performed at the crest of the value of the TCNT counter in MTU4 (MTU7), an A/D converter start request is not issued during down-counting immediately after the transfer (**Figure 16.153**).
- When using A/D converter start requests in coordination with the interrupt skipping function, set up the TADCORA or TADCORB register in MTU4 (MTU7) so that the setting satisfies the condition "2 ≤ TADCORA or TADCORB in MTUn ≤ TCDR - 2" (n = 4 or 7).



Note: UT4AE, DT4AE, BF[1:0]: Bits in TADCR

Note 1. An A/D converter start request is issued when TCDR - 1 ≥ MTU4.TADCOBRA/TADCOBRB ≥ 1 is written.

Figure 16.152 A/D Converter Start Request when TADCOBRA is Set to 0 in MTU4



Note: UT4AE, DT4AE, BF[1:0]: Bits in TADCR

Note 1. An A/D converter start request is issued when TCDR - 1 ≥ MTU4.TADCOBRA/TADCOBRB ≥ 1 is written.

Figure 16.153 A/D Converter Start Request when TADCOBRA is Set to the Same Value as TCDR in MTU4

16.7 MTU Output Pin Initialization

16.7.1 Operating Modes

The MTU has the following six operating modes. Waveforms can be output in any of these modes.

- Normal mode (MTU0 to MTU4 and MTU6 to MTU8)
- PWM mode 1 (MTU0 to MTU4, MTU6, and MTU)
- PWM mode 2 (MTU0 to MTU2)
- Phase counting modes 1 to 5 (MTU1 and MTU2)
- Complementary PWM mode (MTU3, MTU4, MTU6, and MTU7)
- Reset-synchronized PWM mode (MTU3, MTU4, MTU6, and MTU7)

This section describes how to initialize the MTU output pins in each of these modes.

16.7.2 Operation in Case of Re-Setting Due to Error during Operation

If an error occurs during MTU operation, MTU output should be cut off by the system. The output can be cut off by allowing non-active level output from the pins by setting the pins as general output ports using the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports. MTU output can be disabled through TIOR settings. Complementary PWM output (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D, MTIOC6B, MTIOC6D, MTIOC7A, MTIOC7B, MTIOC7C, and MTIOC7D) should be specified through TOERA and TOERB settings. For PWM output pins, output can also be cut by hardware, using port output enable 3 (POE3). The pin initialization procedures for re-setting due to an error during operation and the procedures for restarting in a different mode after re-setting are described below.

The MTU has six operating modes, as stated above. There are thus 36 mode transition combinations, but some transitions are not available with certain channel and mode combinations. Available mode transition combinations are shown in **Table 16.80**.

Table 16.80 Mode Transition Combinations

	Normal	PWM1	PWM2	PCM	CPWM	RPWM
Normal	(1)	(2)	(3)	(4)	(5)	(6)
PWM1	(7)	(8)	(9)	(10)	(11)	(12)
PWM2	(13)	(14)	(15)	(16)	Not available	Not available
PCM	(17)	(18)	(19)	(20)	Not available	Not available
CPWM	(21)	(22)	Not available	Not available	(23) (24)	(25)
RPWM	(26)	(27)	Not available	Not available	(28)	(29)

Note: Normal: Normal mode
PWM1: PWM mode 1
PWM2: PWM mode 2
PCM: Phase counting modes 1 to 5
CPWM: Complementary PWM mode
RPWM: Reset-synchronized PWM mode

16.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation

- When making a transition to a mode (Normal, PWM1, PWM2, or PCM) in which the pin output level is selected by the timer I/O control register (TIOR) setting, initialize the pins by means of TIOR setting.
- In PWM mode 1, waveforms are not output to the MTIOCnB and MTIOCnD pins (n = 3, 4, 6, or 7). If the MTIOCnB or MTIOCnD output is selected as the function of the pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.
- In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers. If the MTIOCnm output (n = 0 to 2 and m = A to D) is selected as the function of the pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.
- In normal mode or PWM mode 2, if TGRC and TGRD operate as buffer registers, waveforms are not output to the corresponding MTIOCnC and MTIOCnD pins (n = 0, 3, 4, 6, or 7). If the MTIOCnC or MTIOCnD output is selected as the function of the pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.
- In PWM mode 1, if either TGRC or TGRD operates as a buffer register, waveforms are not output to the corresponding MTIOCnC and MTIOCnD pins (n = 0, 3, 4, 6, or 7). If the MTIOCnC or MTIOCnD output is selected as the function of the pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.
- When making a transition to a mode (CPWM or RPWM) in which the pin output level is selected by the timer output control register (TOCR1A, TOCR2A, TOCR1B, or TOCR2B) setting, temporarily disable output in MTU3 and MTU4 (or MTU6 and MTU7) with the timer output master enable register (TOERA or TOERB). If the MTIOCnm output (n =3, 4, 6, or 7 and m = A to D) is selected as the function of pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports. Switch to normal mode, perform initialization with TIOR, restore TIOR to its initial value, then operate the MTU in accordance with the mode setting procedure (TOCR1A setting, TOCR2A setting, TMDR1 setting, and TOERA setting (TOCR1B setting, TOCR2B setting, TMDR1 setting, and TOERB setting)).

NOTE

Channel number is substituted for “n” indicated in this section.

Pin initialization procedures are described below for the numbered combinations in **Table 16.80**. The active level is assumed to be low.

(1) Operation When Error Occurs in Normal Mode and Operation is Restarted in Normal Mode

Figure 16.154 shows a case in which an error occurs in normal mode and operation is restarted in normal mode after re- setting.

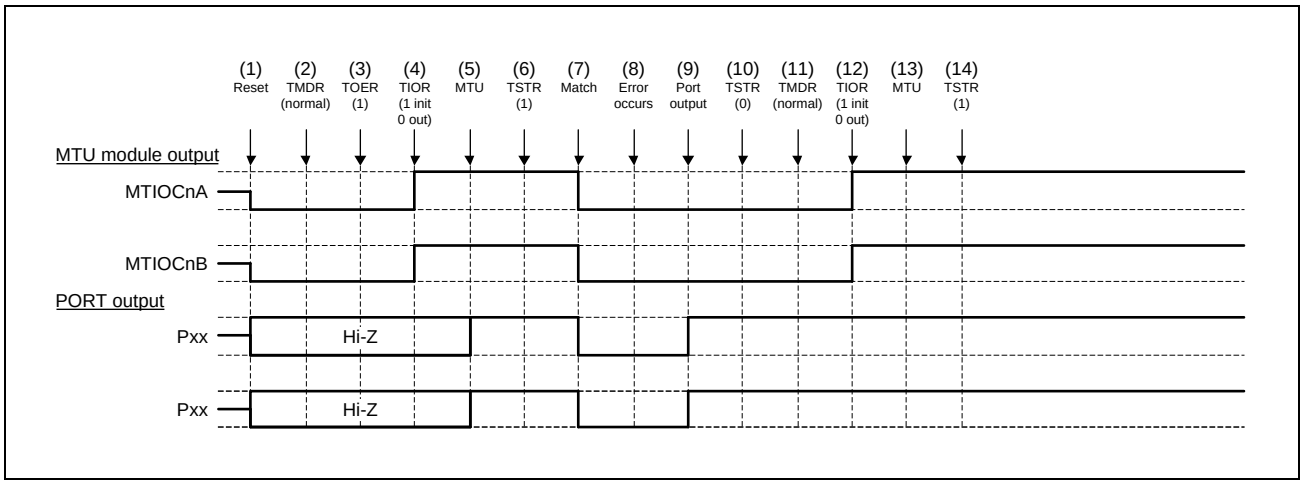


Figure 16.154 Error Occurrence in Normal Mode, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.

(2) After a reset, the TMDR1 setting is for normal mode.

(3) For MTU3 and MTU4 (MTU6 and MTU7), enable output with TOERA (TOERB) before initializing the pins with TIOR.

(4) Initialize the pins with TIOR. (In the example, the initial output is a high level, and a low level is output on compare match occurrence.)

(5) Set MTU output using the port mode control registers (PMCn) and port function control registers (PFCm) corresponding to the GPIO ports.

(6) Start count operation by setting TSTRA (TSTRB).

(7) Output goes low on compare match occurrence.

(8) An error occurs.

(9) Allow non-active level output by setting the pins as general output ports using the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.

(10) Stop count operation by setting TSTRA (TSTRB).

(11) This step is not necessary when restarting in normal mode.

(12) Initialize the pins with TIOR.

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(2) Operation When Error Occurs in Normal Mode and Operation is Restarted in PWM Mode 1

Figure 16.155 shows a case in which an error occurs in normal mode and operation is restarted in PWM mode 1 after re-setting.

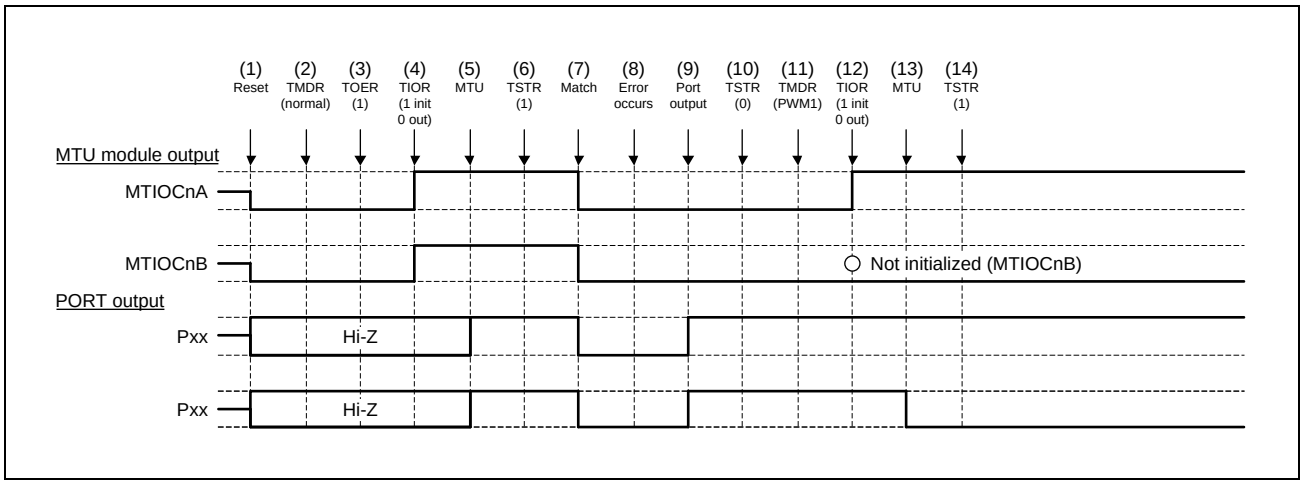


Figure 16.155 Error Occurrence in Normal Mode, Recovery in PWM Mode 1

(1) to (10) are the same as in **Figure 16.154**.

(11) Set PWM mode 1.

(12) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(3) Operation When Error Occurs in Normal Mode and Operation is Restarted in PWM mode 2

Figure 16.156 shows a case in which an error occurs in normal mode and operation is restarted in PWM mode 2 after re-setting.

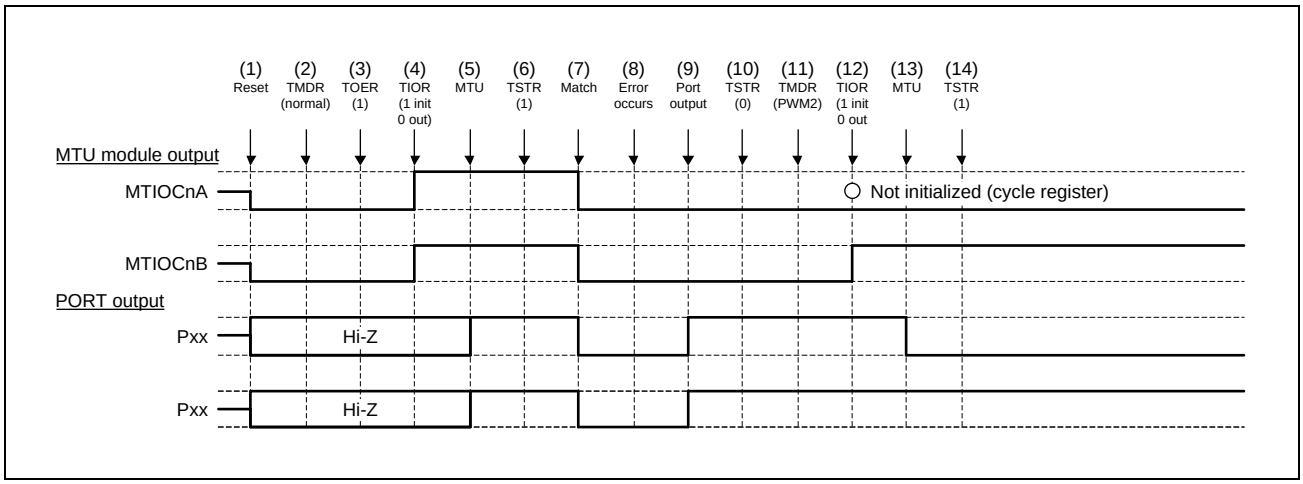


Figure 16.156 Error Occurrence in Normal Mode, Recovery in PWM Mode 2

(1) to (10) are the same as in **Figure 16.154**.

(11) Set PWM mode 2.

(12) Initialize the pins with TIOR. (In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers. To output a specified level, set up the general output ports with the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA.

NOTE

PWM mode 2 can only be selected for MTU0 to MTU2, and therefore TOERA setting is not necessary.

(4) Operation When Error Occurs in Normal Mode and Operation is Restarted in Phase Counting Mode

Figure 16.157 shows a case in which an error occurs in normal mode and operation is restarted in phase counting mode after re-setting.

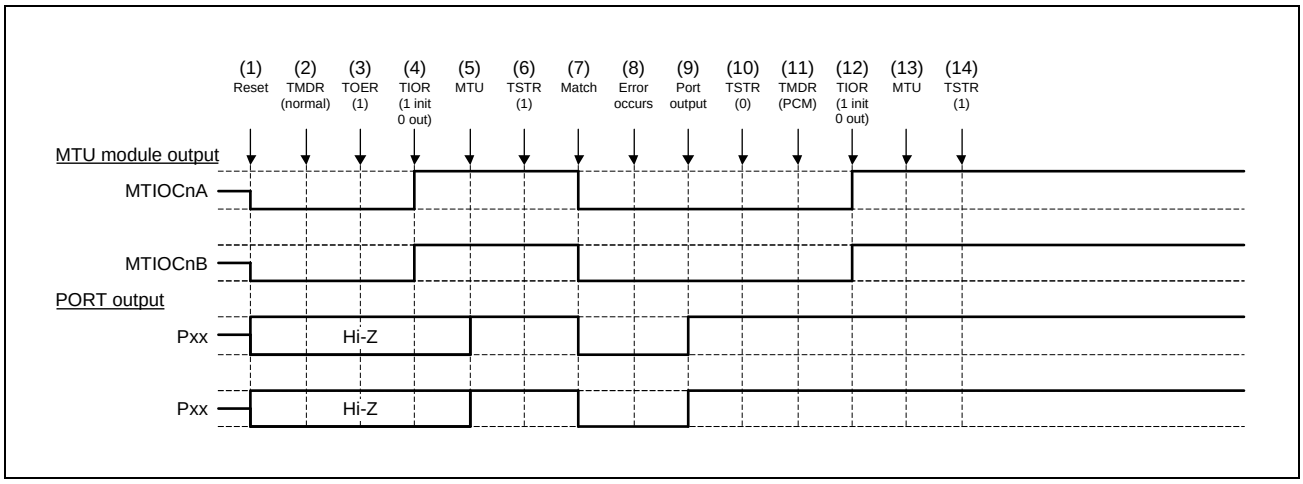


Figure 16.157 Error Occurrence in Normal Mode, Recovery in Phase Counting Mode

(1) to (10) are the same as in **Figure 16.154**.

(11) Set the phase counting mode.

(12) Initialize the pins with TIOR.

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA.

NOTE

The phase counting mode can only be selected for MTU1 and MTU2, and therefore TOERA setting is not necessary.

(5) Operation When Error Occurs in Normal Mode and Operation is Restarted in Complementary PWM Mode

Figure 16.158 shows a case in which an error occurs in normal mode and operation is restarted in complementary PWM mode after re-setting.

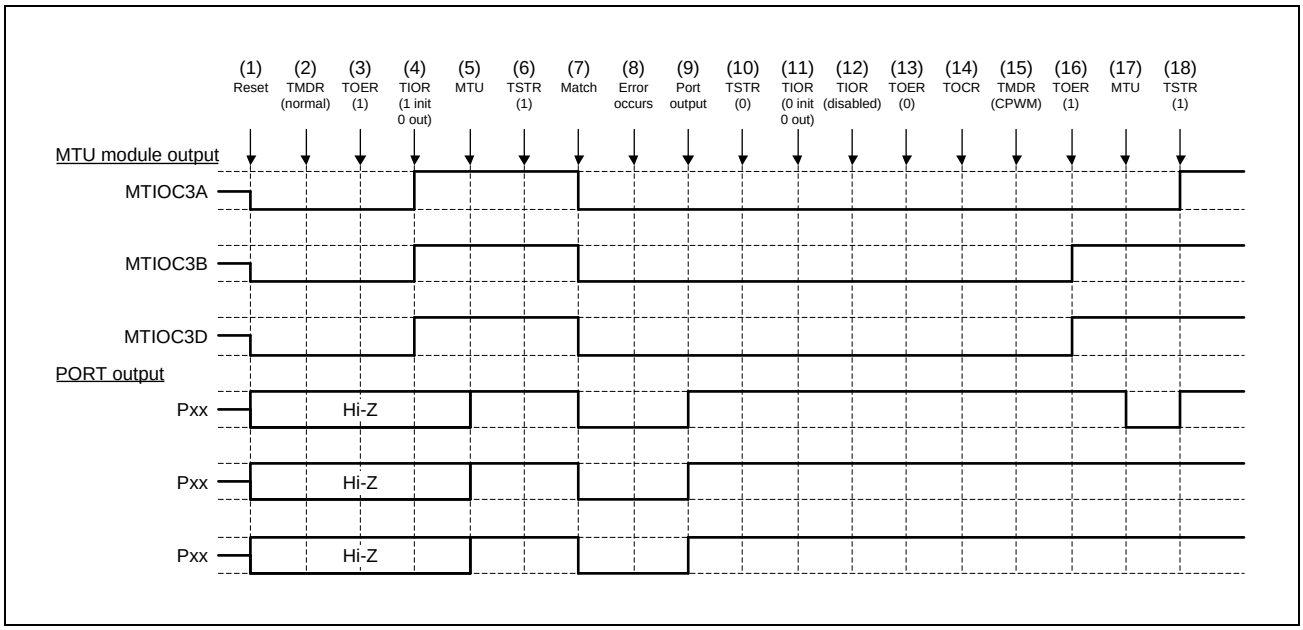


Figure 16.158 Error Occurrence in Normal Mode, Recovery in Complementary PWM Mode

(1) to (10) are the same as in **Figure 16.154**.

(11) Initialize the normal mode waveform generation section with TIOR.

(12) Disable operation of the normal mode waveform generation section with TIOR.

(13) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(14) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(15) Set complementary PWM mode.

(16) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(17) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(18) Restart operation by setting TSTRA (TSTRB).

(6) Operation When Error Occurs in Normal Mode and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 16.159 shows a case in which an error occurs in normal mode and operation is restarted in reset-synchronized PWM mode after re-setting.

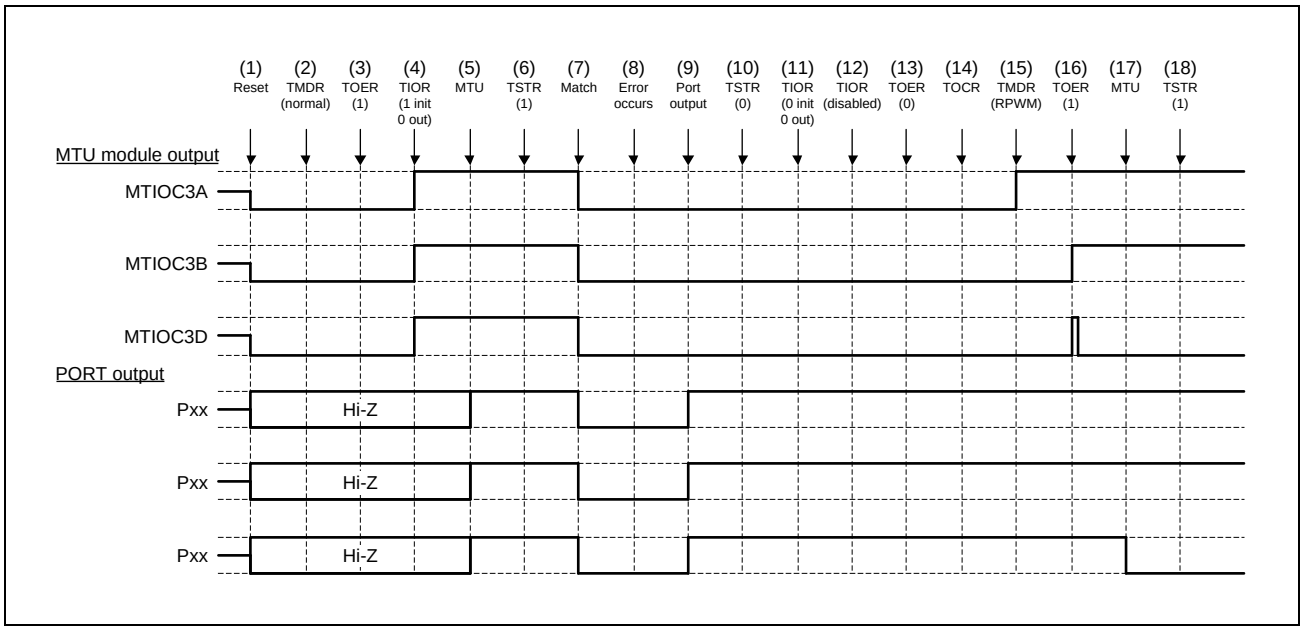


Figure 16.159 Error Occurrence in Normal Mode, Recovery in Reset-Synchronized PWM Mode

(1) to (13) are the same as in **Figure 16.156**.

(14) Select the reset-synchronized PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(15) Set reset-synchronized PWM mode.

(16) Enable output from MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(17) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(18) Restart operation by setting TSTRA (TSTRB).

(7) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Normal Mode

Figure 16.160 shows a case in which an error occurs in PWM mode 1 and operation is restarted in normal mode after re-setting.

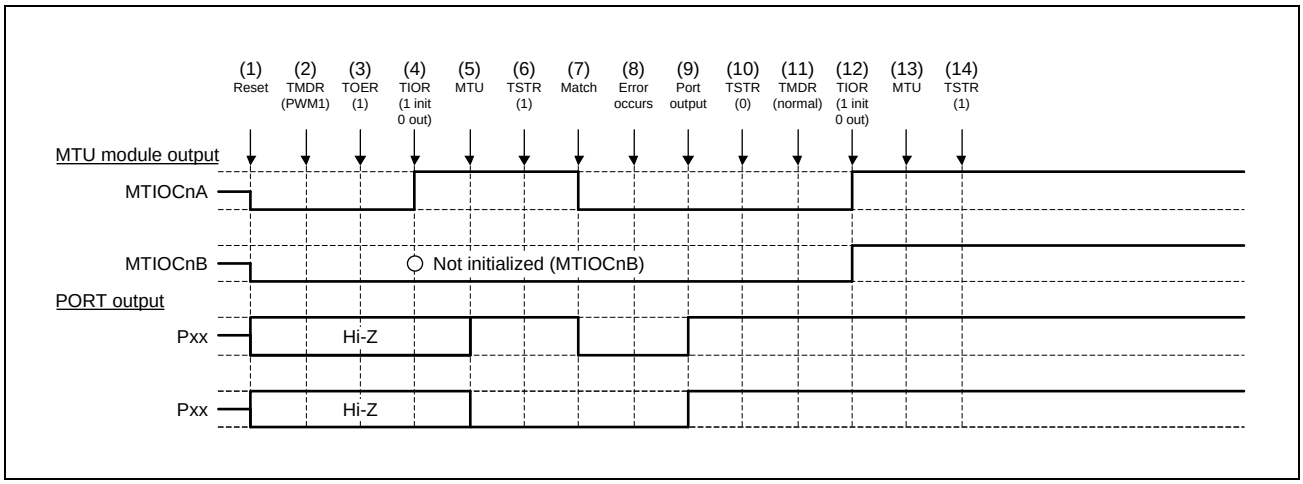


Figure 16.160 Error Occurrence in PWM Mode 1, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.

(2) Set PWM mode 1.

(3) For MTU3 and MTU4 (MTU6 and MTU7), enable output with TOERA (TOERB) before initializing the pins with TIOR.

(4) Initialize the pins with TIOR. (In the example, the initial output is a high level, and a low level is output on compare match occurrence. In PWM mode 1, the MTIOCnB side is not initialized.)

(5) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(6) Start count operation by setting TSTRA (TSTRB).

(7) Output goes low on compare match occurrence.

(8) An error occurs.

(9) Allow non-active level output by setting the pins as general output ports using the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.

(10) Stop count operation by setting TSTRA (TSTRB).

(11) Set normal mode.

(12) Initialize the pins with TIOR.

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(8) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in PWM mode 1

Figure 16.161 shows a case in which an error occurs in PWM mode 1 and operation is restarted in PWM mode 1 after re-setting.

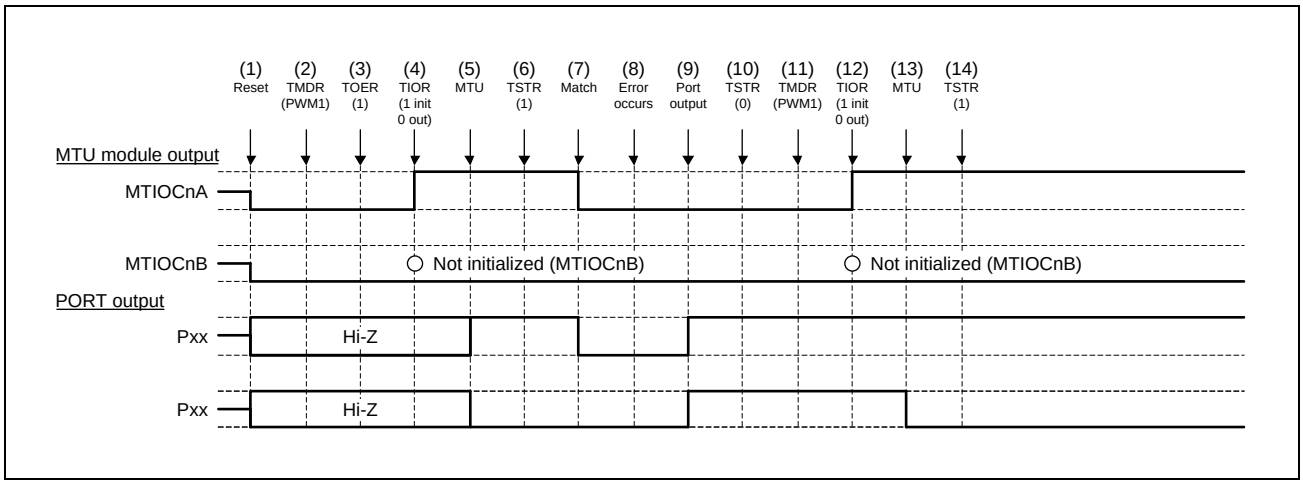


Figure 16.161 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 1

(1) to (10) are the same as in **Figure 16.160**.

(11) This step is not necessary when restarting in PWM mode 1.

(12) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(9) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in PWM mode 2

Figure 16.162 shows a case in which an error occurs in PWM mode 1 and operation is restarted in PWM mode 2 after re-setting.

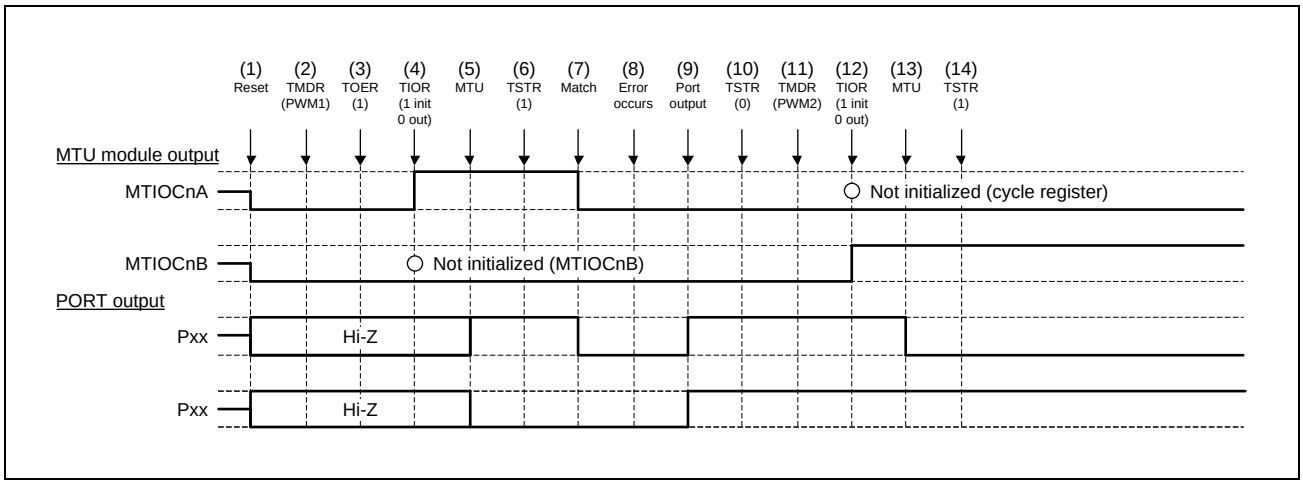


Figure 16.162 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 2

(1) to (10) are the same as in **Figure 16.160**.

(11) Set PWM mode 2.

(12) Initialize the pins with TIOR. (In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers.
To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA.

NOTE

PWM mode 2 can only be selected for MTU0 to MTU2, and therefore TOERA setting is not necessary.

(10) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Phase Counting Mode

Figure 16.163 shows a case in which an error occurs in PWM mode 1 and operation is restarted in phase counting mode after re-setting.

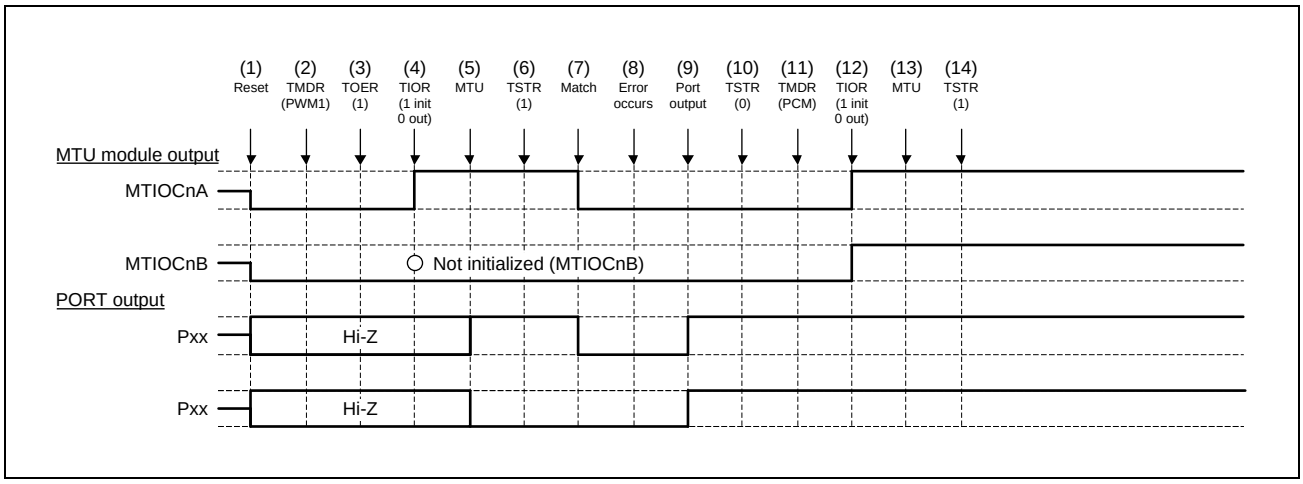


Figure 16.163 Error Occurrence in PWM Mode 1, Recovery in Phase Counting Mode

(1) to (10) are the same as in **Figure 16.160**.

(11) Set the phase counting mode.

(12) Initialize the pins with TIOR.

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA.

NOTE

The phase counting mode can only be selected for MTU1 and MTU2, and therefore TOERA setting is not necessary.

(11) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Complementary PWM Mode

Figure 16.164 shows a case in which an error occurs in PWM mode 1 and operation is restarted in complementary PWM mode after re-setting.

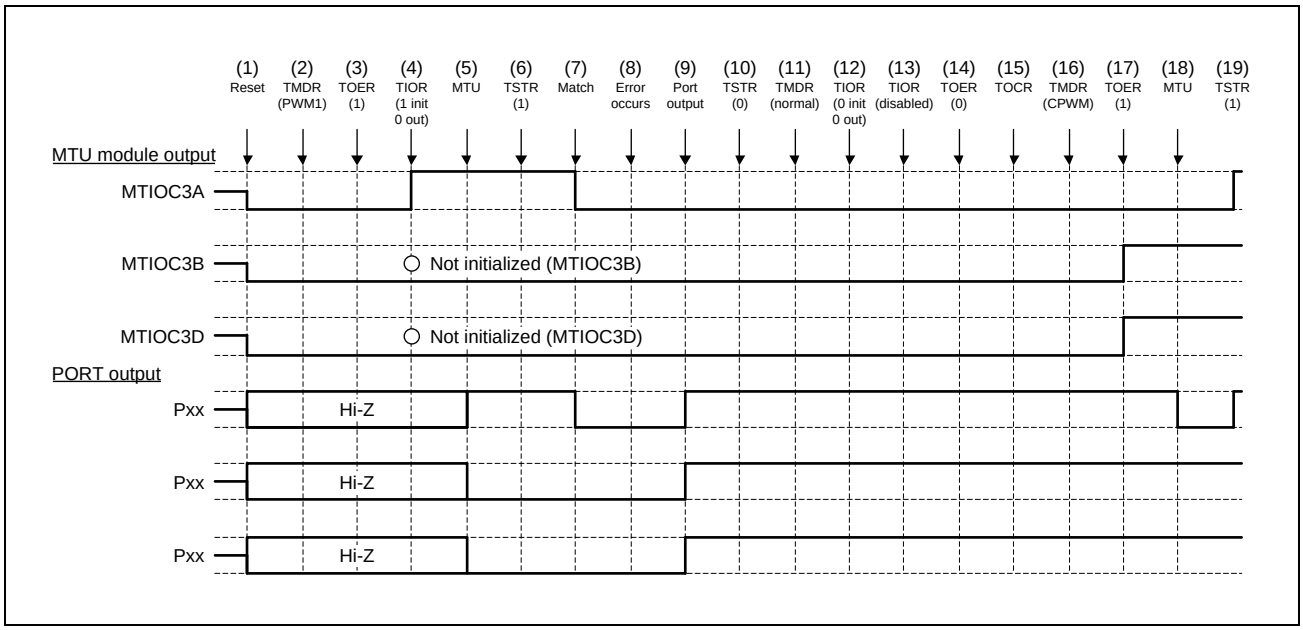


Figure 16.164 Error Occurrence in PWM Mode 1, Recovery in Complementary PWM Mode

(1) to (10) are the same as in **Figure 16.160**.

(11) Set normal mode to initialize the normal mode waveform generation section.

(12) Initialize the PWM mode 1 waveform generation section with TIOR.

(13) Disable operation of the PWM mode 1 waveform generation section with TIOR

(14) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(15) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(16) Set complementary PWM mode.

(17) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(18) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(19) Restart operation by setting TSTRA (TSTRB).

(12) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Reset- Synchronized PWM Mode

Figure 16.165 shows a case in which an error occurs in PWM mode 1 and operation is restarted in reset-synchronized PWM mode after re-setting.

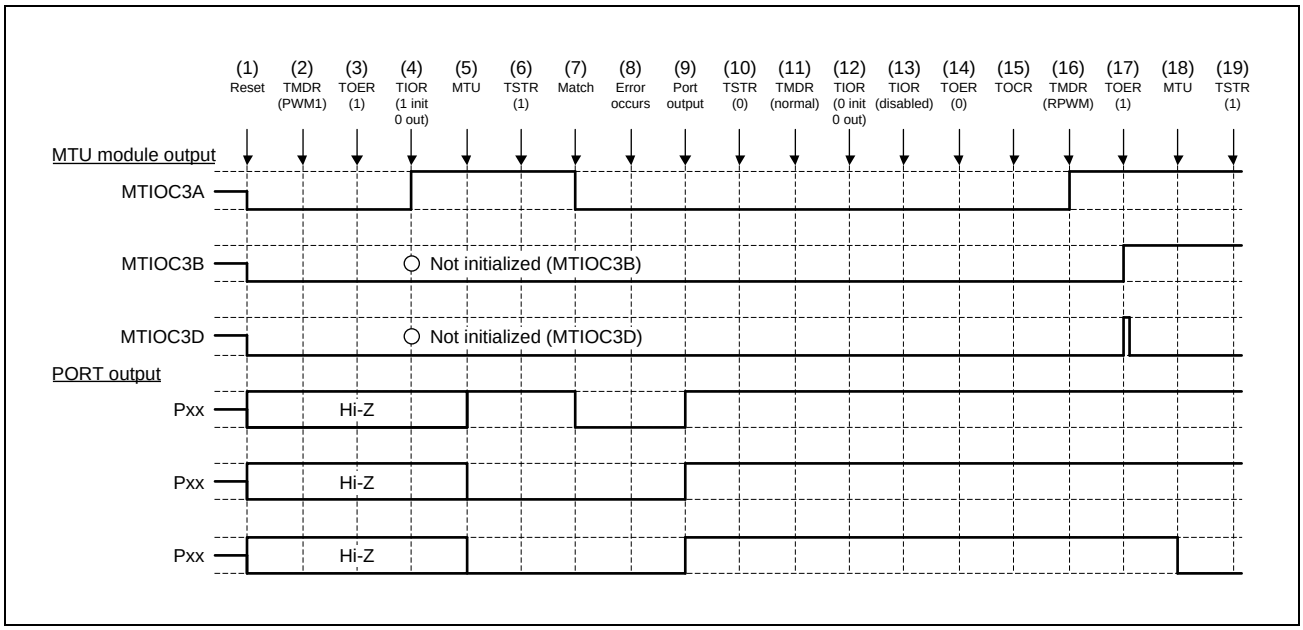


Figure 16.165 Error Occurrence in PWM Mode 1, Recovery in Reset-Synchronized PWM Mode

(1) to (14) are the same as in **Figure 16.164**.

(15) Select the reset-synchronized PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(16) Set reset-synchronized PWM mode.

(17) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(18) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(19) Restart operation by setting TSTRA (TSTRB).

(13) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in Normal Mode

Figure 16.166 shows a case in which an error occurs in PWM mode 2 and operation is restarted in normal mode after re-setting.

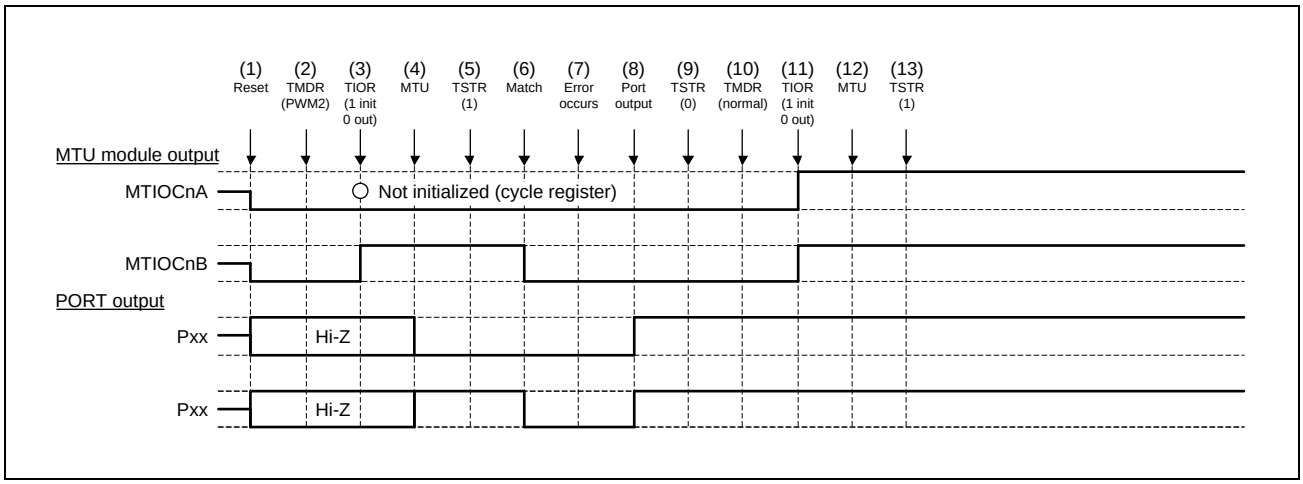


Figure 16.166 Error Occurrence in PWM Mode 2, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.

(2) Set PWM mode 2.

(3) Initialize the pins with TIOR. (In the example, the initial output is a high level, and a low level is output on compare match occurrence. In PWM mode 2, the pins corresponding to the TGR registers used as cycle registers are not initialized. In the example, TGRA in MTUn is used as a cycle register.)

(4) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(5) Start count operation by setting TSTRA.

(6) Output goes low on compare match occurrence.

(7) An error occurs.

(8) Allow non-active level output by setting the pins as general output ports using the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.

(9) Stop count operation by setting TSTRA.

(10) Set normal mode.

(11) Initialize the pins with TIOR.

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(14) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in PWM Mode 1

Figure 16.167 shows a case in which an error occurs in PWM mode 2 and operation is restarted in PWM mode 1 after re-setting.

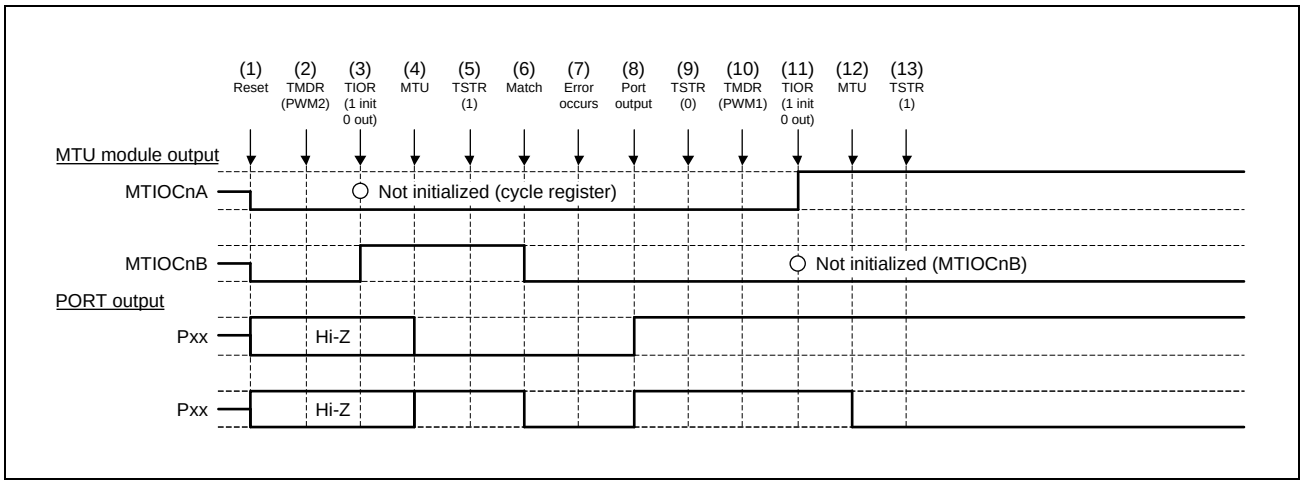


Figure 16.167 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 1

(1) to (9) are the same as in **Figure 16.166**.

(10) Set PWM mode 1.

(11) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(15) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in PWM Mode 2

Figure 16.168 shows a case in which an error occurs in PWM mode 2 and operation is restarted in PWM mode 2 after re-setting.

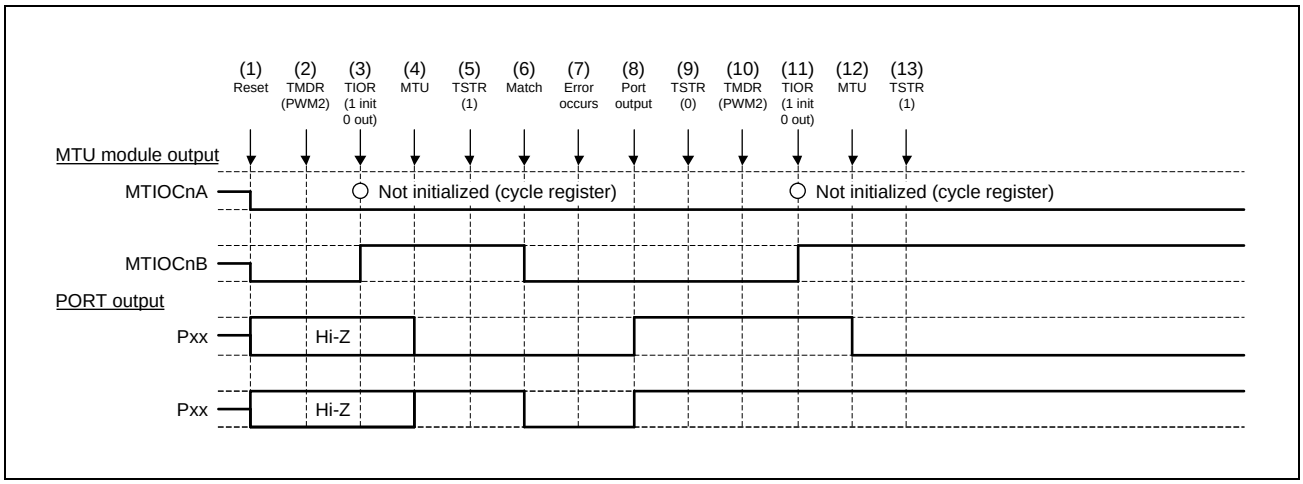


Figure 16.168 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 2

(1) to (9) are the same as in **Figure 16.166**.

(10) This step is not necessary when restarting in PWM mode 2.

(11) Initialize the pins with TIOR. (In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers.
To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(16) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in Phase Counting Mode

Figure 16.169 shows a case in which an error occurs in PWM mode 2 and operation is restarted in phase counting mode after re-setting.

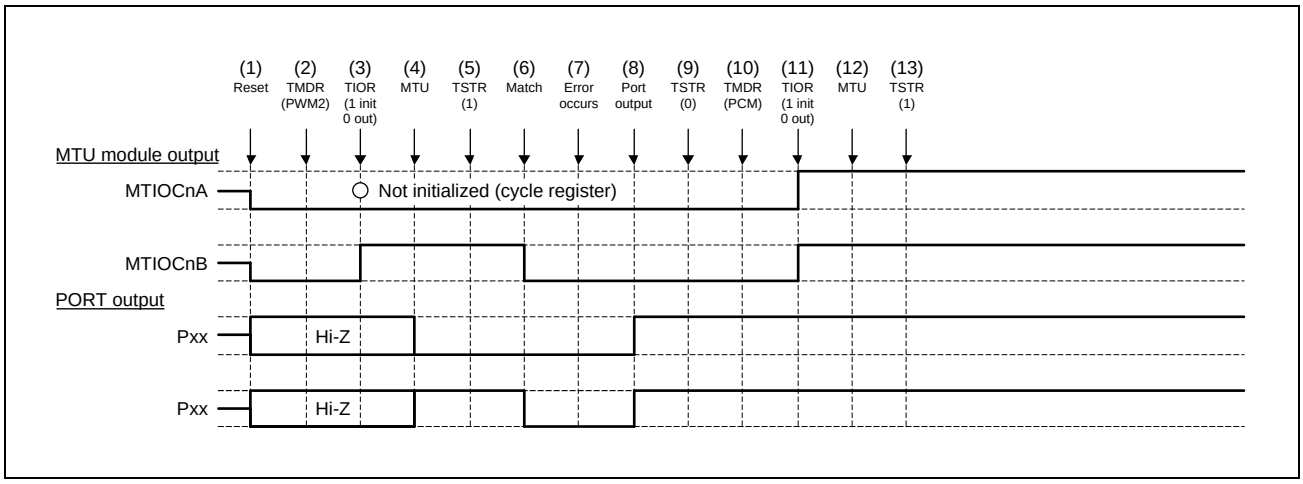


Figure 16.169 Error Occurrence in PWM Mode 2, Recovery in Phase Counting Mode

(1) to (9) are the same as in **Figure 16.166**.

(10) Set the phase counting mode.

(11) Initialize the pins with TIOR.

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(17) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in Normal Mode

Figure 16.170 shows a case in which an error occurs in phase counting mode and operation is restarted in normal mode after re-setting.

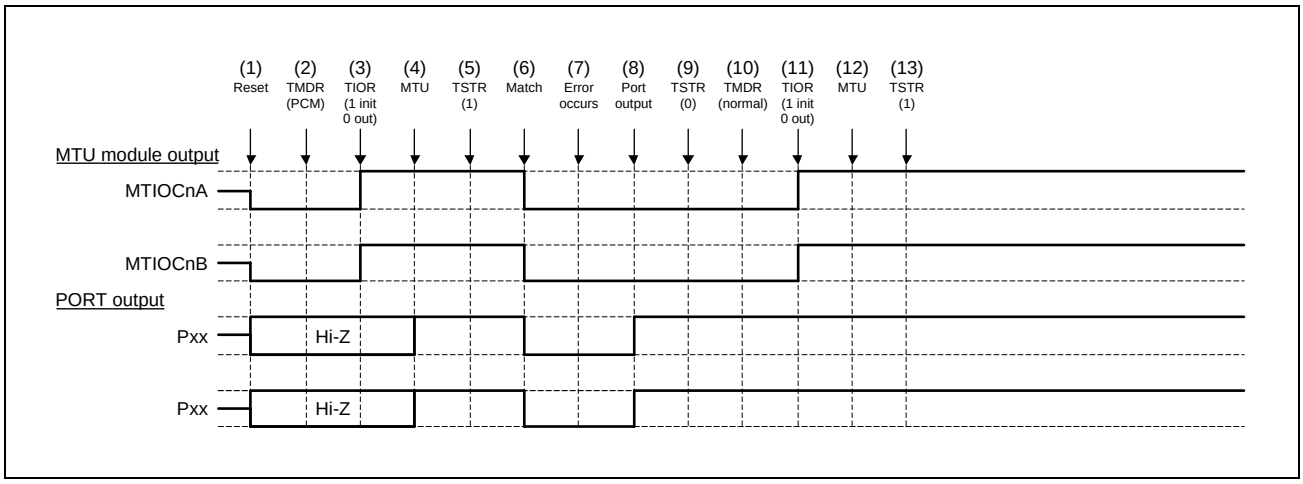


Figure 16.170 Error Occurrence in Phase Counting Mode, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.

(2) Set phase counting mode.

(3) Initialize the pins with TIOR. (In the example, the initial output is a high level, and a low level is output on compare match occurrence.)

(4) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(5) Start count operation by setting TSTRA.

(6) Output goes low on compare match occurrence.

(7) An error occurs.

(8) Allow non-active level output by setting the pins as general output ports using the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.

(9) Stop count operation by setting TSTRA.

(10) Set normal mode.

(11) Initialize the pins with TIOR.

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(18) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in PWM Mode 1

Figure 16.171 shows a case in which an error occurs in phase counting mode and operation is restarted in PWM mode 1 after re-setting.

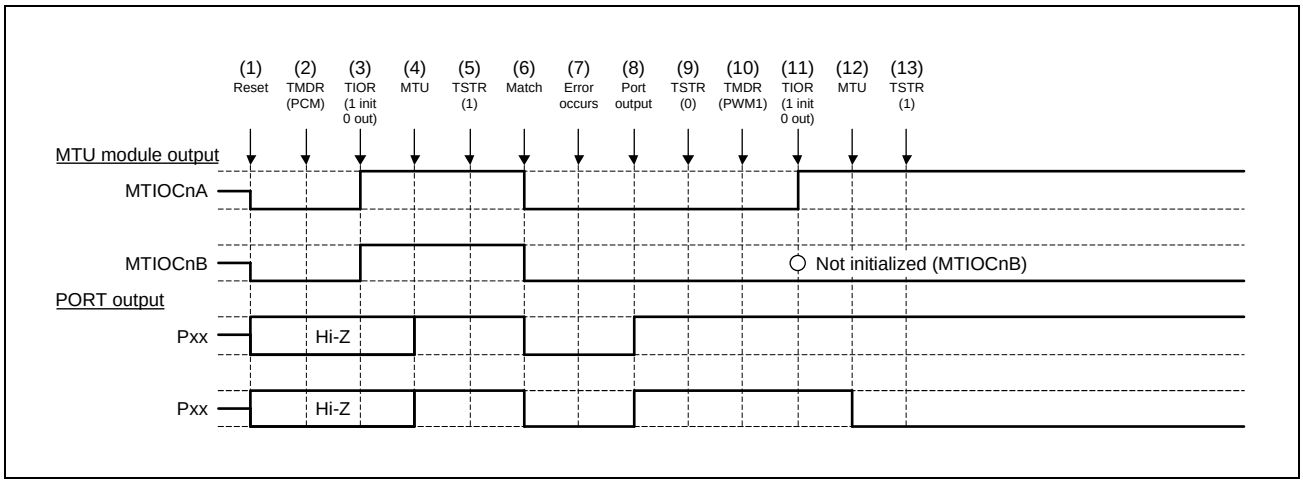


Figure 16.171 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 1

(1) to (9) are the same as in **Figure 16.170**.

(10) Set PWM mode 1.

(11) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(19) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in PWM Mode 2

Figure 16.172 shows a case in which an error occurs in phase counting mode and operation is restarted in PWM mode 2 after re-setting.

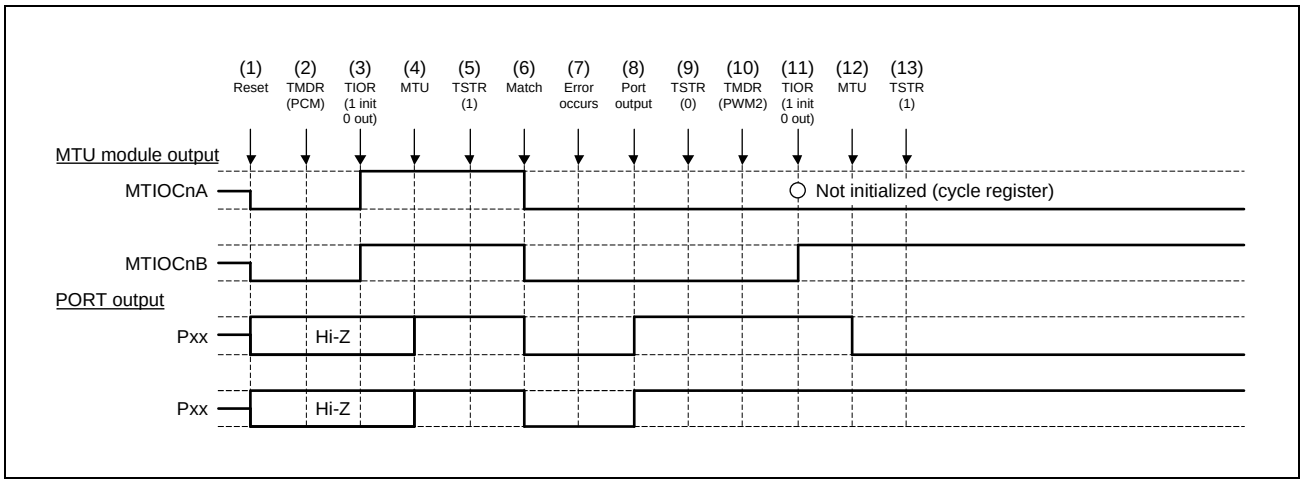


Figure 16.172 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 2

(1) to (9) are the same as in **Figure 16.170**.

(10) Set PWM mode 2.

(11) Initialize the pins with TIOR. (In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers.
To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(20) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in Phase Counting Mode

Figure 16.173 shows a case in which an error occurs in phase counting mode and operation is restarted in phase counting mode after re-setting.

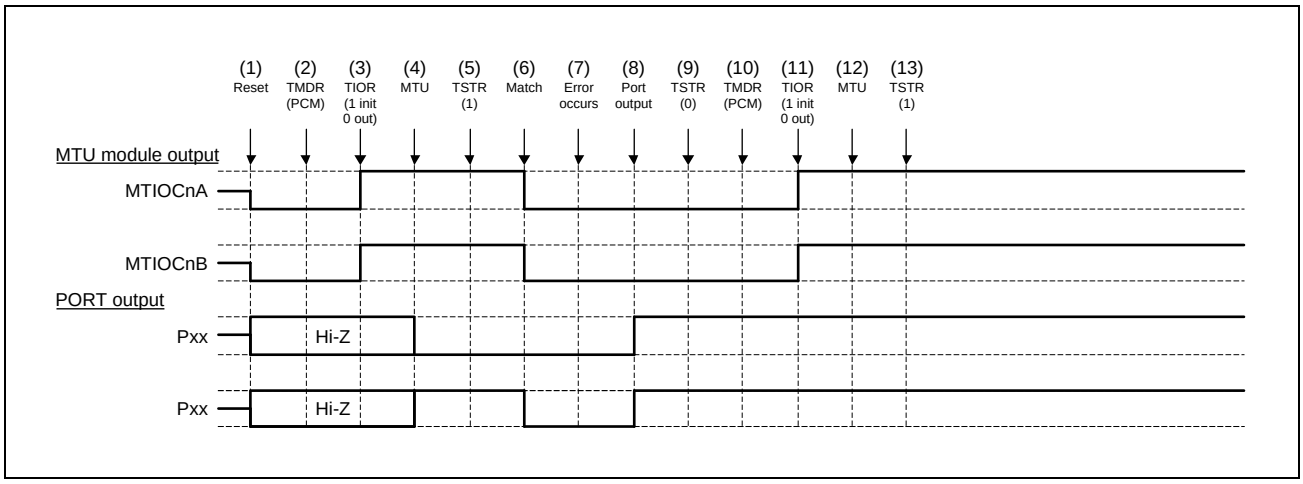


Figure 16.173 Error Occurrence in Phase Counting Mode, Recovery in Phase Counting Mode

(1) to (9) are the same as in **Figure 16.170**.

(10) This step is not necessary when restarting in phase counting mode.

(11) Initialize the pins with TIOR.

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(21) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Normal Mode

Figure 16.174 shows a case in which an error occurs in complementary PWM mode and operation is restarted in normal mode after re-setting.

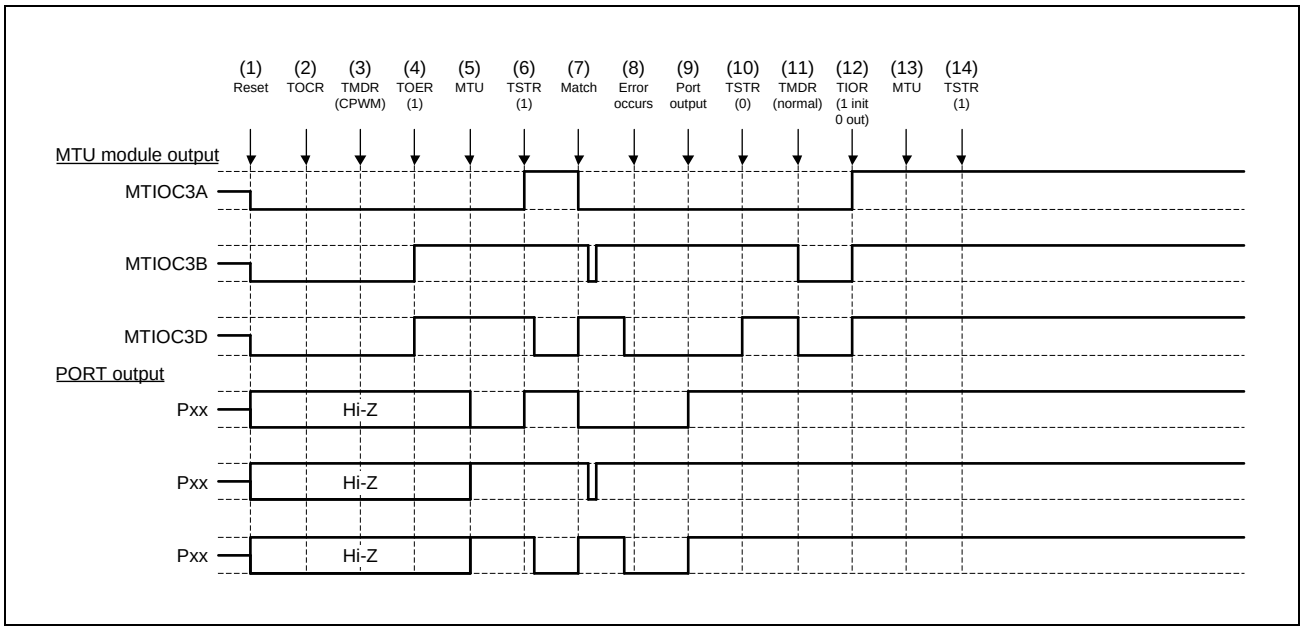


Figure 16.174 Error Occurrence in Complementary PWM Mode, Recovery in Normal Mode

(1) After a reset, the MTU3 output goes low and the ports enter high-impedance state.

(2) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(3) Set complementary PWM mode.

(4) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(5) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(6) Start count operation by setting TSTRA (TSTRB).

(7) The complementary PWM waveform is output on compare match occurrence.

(8) An error occurs.

(9) Allow non-active level output by setting the pins as general output ports using the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.

(10) Stop count operation by setting TSTRA (TSTRB). (MTU output becomes the initial complementary PWM output value).

(11) Set normal mode (MTU output goes low).

(12) Initialize the pins with TIOR.

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(22) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in PWM Mode 1

Figure 16.175 shows a case in which an error occurs in complementary PWM mode and operation is restarted in PWM mode 1 after re-setting.

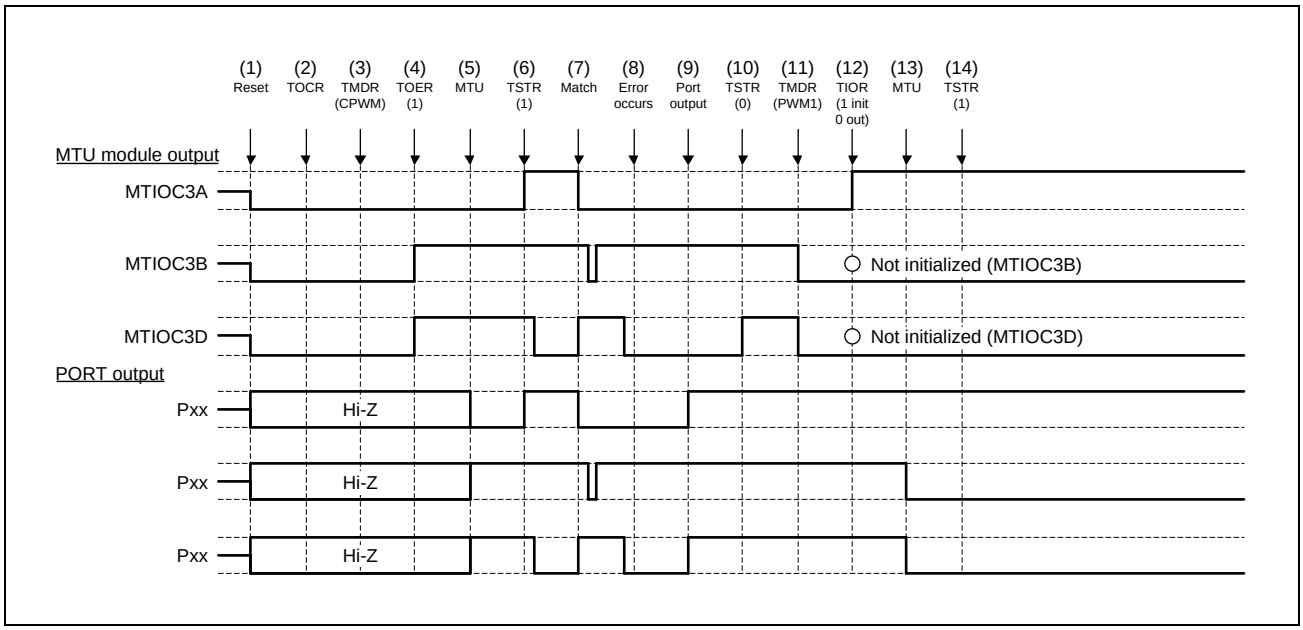


Figure 16.175 Error Occurrence in Complementary PWM Mode, Recovery in PWM Mode 1

(1) to (10) are the same as in **Figure 16.174**.

(11) Set PWM mode 1 (MTU output goes low).

(12) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(23) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Complementary PWM Mode

Figure 16.176 shows a case in which an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (when operation is restarted using the cycle and duty settings at the time of stopping the counter).

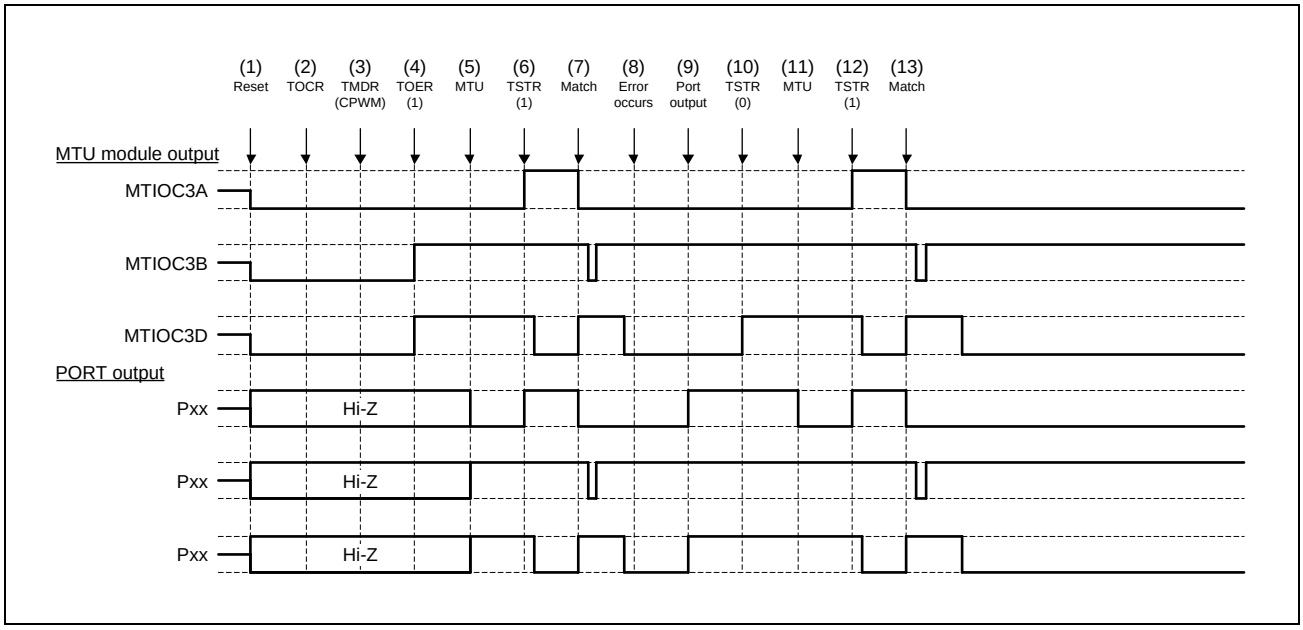


Figure 16.176 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode (1)

(1) to (10) are the same as in **Figure 16.174**.

(11) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(12) Restart operation by setting TSTRA (TSTRB).

(13) The complementary PWM waveform is output on compare match occurrence.

(24) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Complementary PWM Mode with New Settings

Figure 16.177 shows a case in which an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (operation is restarted using new cycle and duty ratio settings).

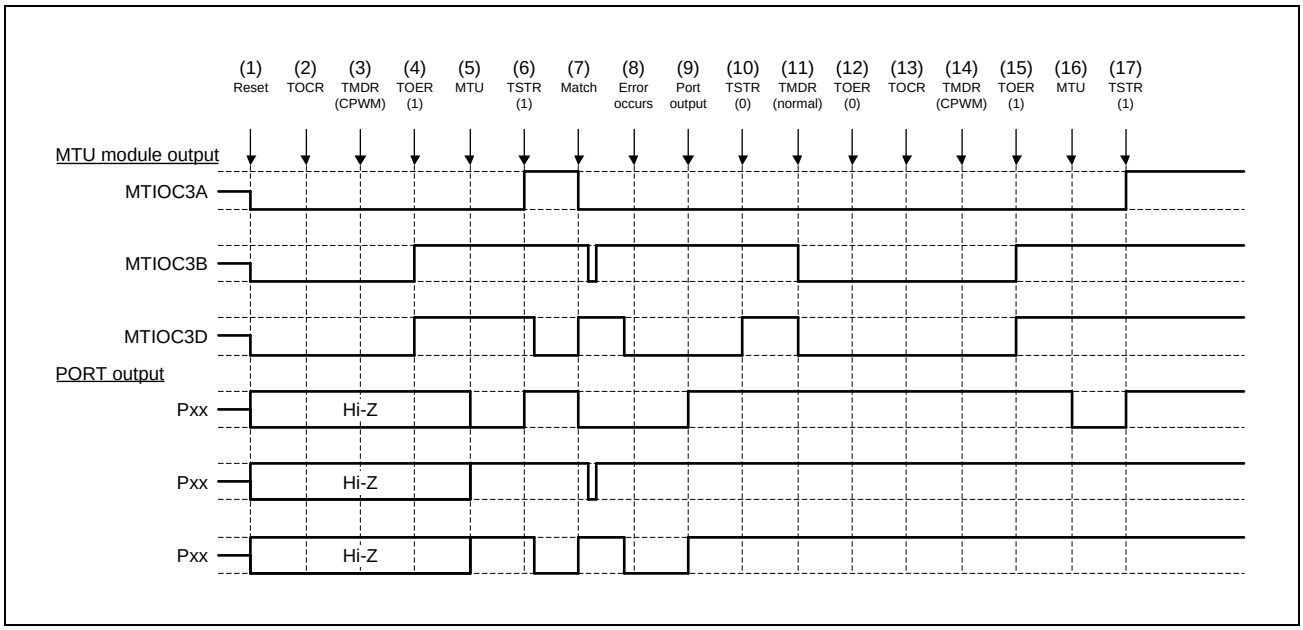


Figure 16.177 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode (2)

(1) to (10) are the same as in **Figure 16.174**.

(11) Set normal mode and make new settings (MTU output goes low).

(12) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(13) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(14) Set complementary PWM mode.

(15) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(16) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(17) Restart operation by setting TSTRA (TSTRB).

(25) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 16.178 shows a case in which an error occurs in complementary PWM mode and operation is restarted in reset- synchronized PWM mode after re-setting.

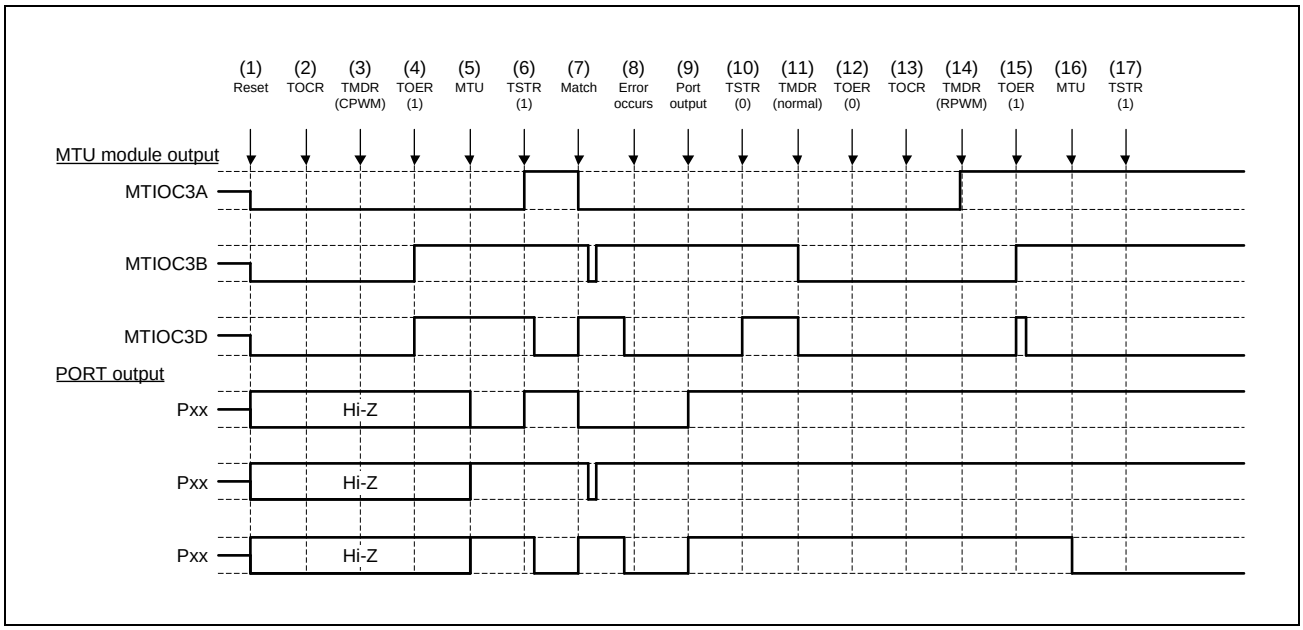


Figure 16.178 Error Occurrence in Complementary PWM Mode, Recovery in Reset-Synchronized PWM Mode

(1) to (10) are the same as in **Figure 16.174**.

(11) Set normal mode (MTU output goes low).

(12) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(13) Select the reset-synchronized PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(14) Set reset-synchronized PWM mode.

(15) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(16) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(17) Restart operation by setting TSTRA (TSTRB).

(26) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Normal Mode

Figure 16.179 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in normal mode after re-setting.

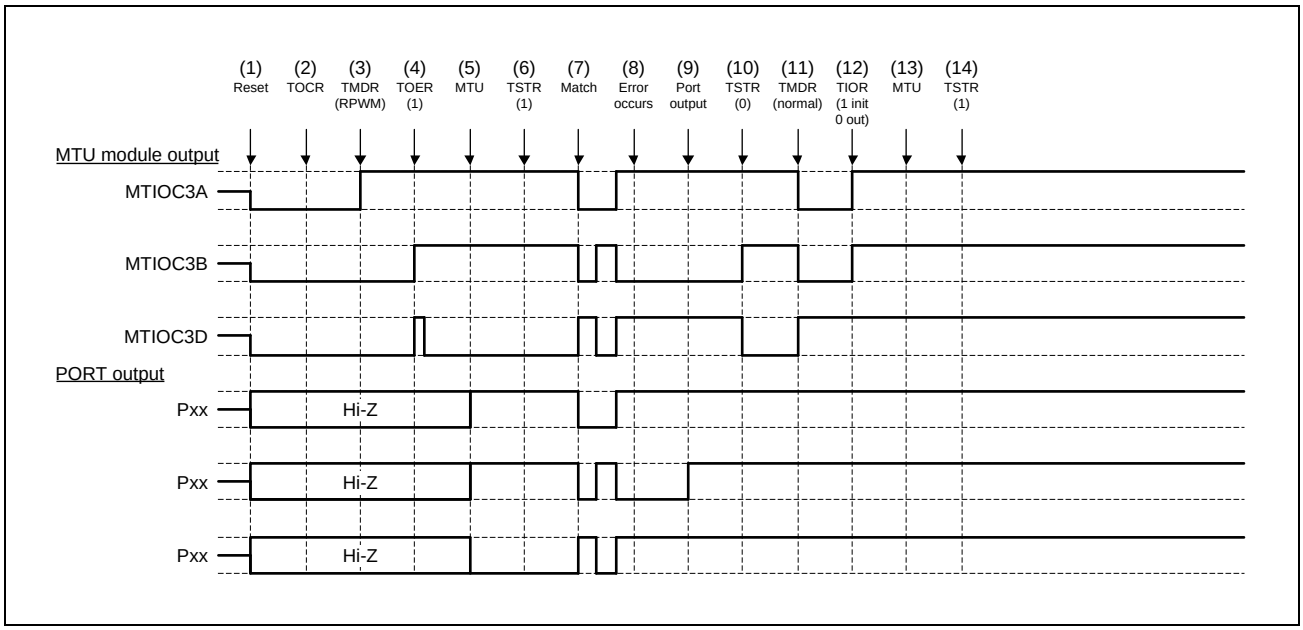


Figure 16.179 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.

(2) Select the reset-synchronized PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(3) Set reset-synchronized PWM mode.

(4) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(5) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(6) Start count operation by setting TSTRA (TSTRB).

(7) The reset-synchronized PWM waveform is output on compare match occurrence.

(8) An error occurs.

(9) Allow non-active level output by setting the pins as general output ports using the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.

(10) Stop count operation by setting TSTRA (TSTRB). (MTU output becomes the initial reset-synchronized PWM output value.)

(11) Set normal mode (positive-phase MTU output goes low, and negative-phase output goes high).

(12) Initialize the pins with TIOR.

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(27) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in PWM Mode 1

Figure 16.180 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in PWM mode 1 after re-setting.

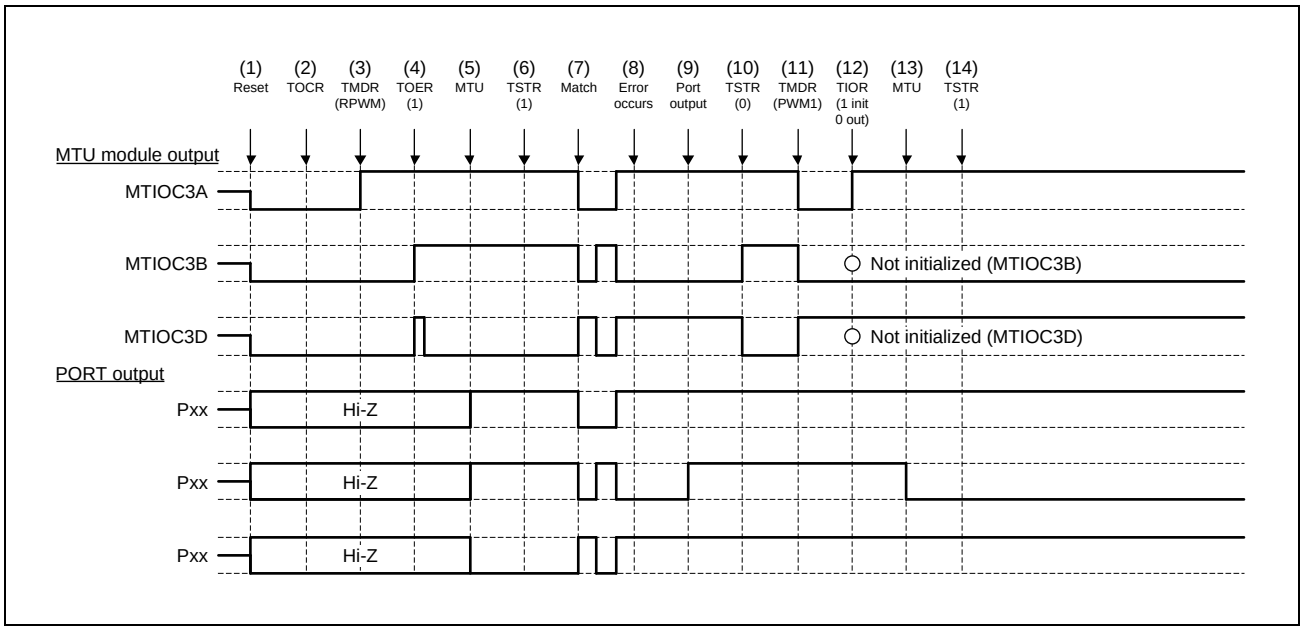


Figure 16.180 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in PWM Mode 1

(1) to (10) are the same as in **Figure 16.179**.

(11) Set PWM mode 1 (positive-phase MTU output goes low, and negative-phase output goes high).

(12) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(28) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Complementary PWM Mode

Figure 16.181 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in complementary PWM mode after re-setting.

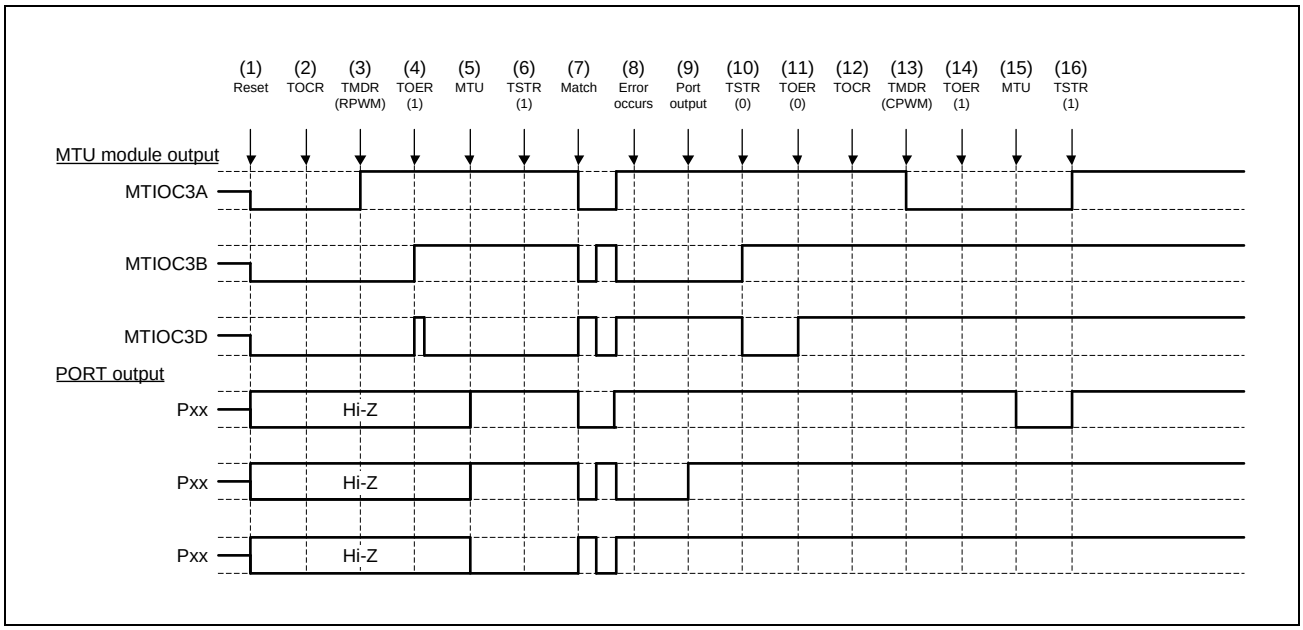


Figure 16.181 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Complementary PWM Mode

(1) to (10) are the same as in **Figure 16.179**.

(11) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(12) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(13) Set complementary PWM mode (MTU cyclic output pin goes low).

(14) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(15) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(16) Restart operation by setting TSTRA (TSTRB).

(29) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 16.182 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in reset-synchronized PWM mode after re-setting.

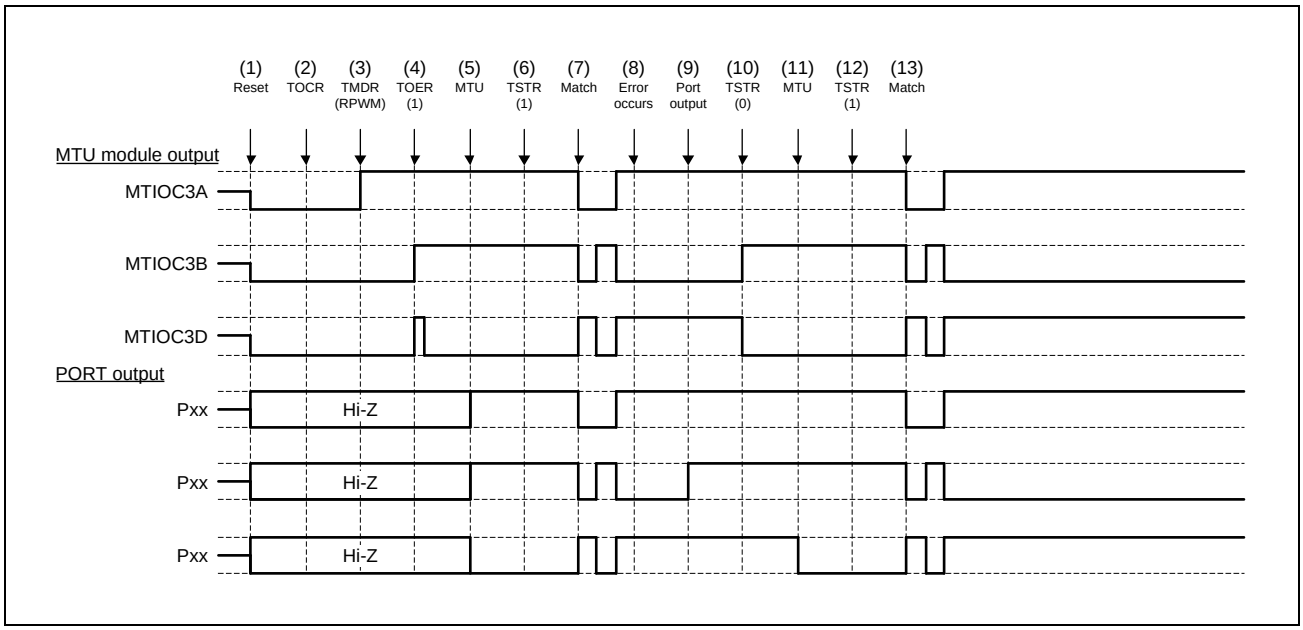


Figure 16.182 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Reset-Synchronized PWM Mode

(1) to (10) are the same as in **Figure 16.179**.

(11) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(12) Restart operation by setting TSTRA (TSTRB).

(13) The reset-synchronized PWM waveform is output on compare match occurrence.

17. Port Output Enable 3 (POE3)

The port output enable 3 (POE3) can be used to place output pins for the MTU3a in the high-impedance state in response to various conditions.

17.1 Overview

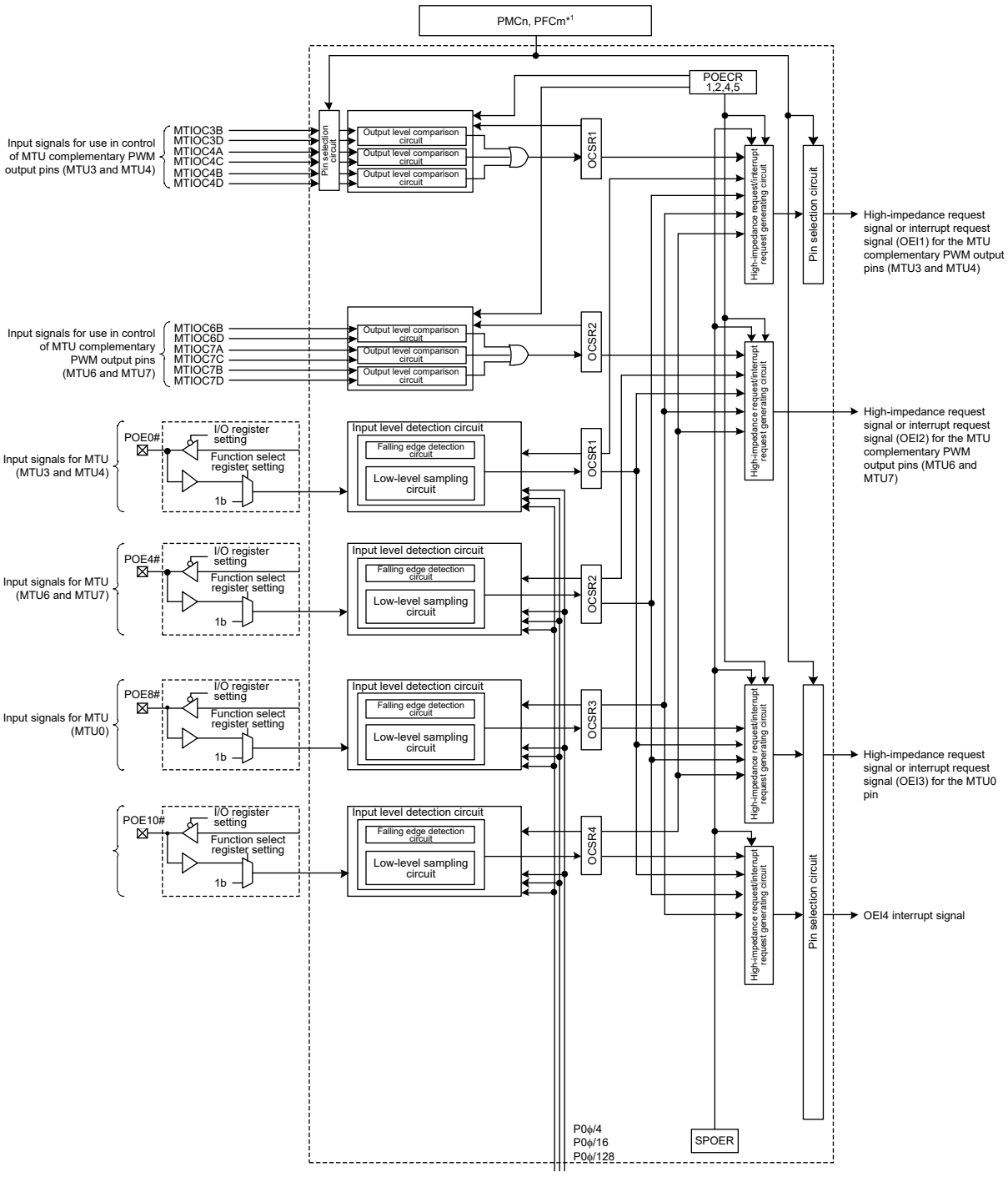
Table 17.1 lists the specifications of the POE3, and **Figure 17.1** shows a block diagram of the POE3.

Table 17.1 POE3 Specifications

Item	Description
Target pins to be placed in the high-impedance state	• MTU3a output pins MTU0 pin (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D) MTU3 pin (MTIOC3B, MTIOC3D) MTU4 pin (MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D) MTU6 pin (MTIOC6B, MTIOC6D) MTU7 pin (MTIOC7A, MTIOC7B, MTIOC7C, MTIOC7D)
Conditions for the high-impedance state	• Setting pins as inputs: setting the POE0#, POE4#, POE8#, or POE10# pins as inputs (falling edge or low- level sampling). • Short-circuits between output pins: A match (short circuit) between the output signal levels at the active level over one or more cycle on (the following combination of pins) • SPOER register setting being made
Function	• Each of the POE0#, POE4#, POE8#, and POE10# input pins can be set for falling edge, P0ϕ/4 × 16, P0ϕ/ 16 × 16, or P0ϕ/128 × 16 low-level sampling. • Pins for the MTU complementary PWM output, and MTU0 pin can be placed in high-impedance state by POE0#, POE4#, POE8#, and POE10# pin falling-edge or low-level sampling. • Pins for the MTU complementary PWM output can be placed in high-impedance state when output levels of the MTU complementary PWM output pins are compared and simultaneous active-level output continues for one cycle or more. • Pins for the MTU complementary PWM output, and MTU0 can be placed in the high-impedance state by modifying the settings of the SPOER register of POE3. • Interrupts can be generated by input-level sampling or output level comparison results.

	MTU complementary PWM output pins
1	MTIOC3B and MTIOC3D
2	MTIOC4A and MTIOC4C
3	MTIOC4B and MTIOC4D
4	MTIOC6B and MTIOC6D
5	MTIOC7A and MTIOC7C
6	MTIOC7B and MTIOC7D

The POE3 has input level detection circuits, pin selection circuits, output level comparison circuits, and a high-impedance request/interrupt request generating circuit as shown in **Figure 17.1**.



Note 1. Refer to **Section 45, General Purpose Input Output Port (GPIO)**.

Figure 17.1 POE3 Block Diagram

Table 17.2 shows input/output pins to be used by the POE3.

Table 17.2 POE3 Input/Output Pins

Pin Name	I/O	Description
POE0#	Input	Input pin for the request signal to place the MTU3 and MTU4 pins for MTU complementary PWM output in high-impedance state. In accord with register settings, this pin is also capable of placing the MTU0, MTU6, and MTU7 pins in the high-impedance state.
POE4#	Input	Input pin for the request signal to place the MTU6 and MTU7 pins for MTU complementary PWM output in high-impedance state. In accord with register settings, this pin is also capable of placing the MTU0, MTU3, and MTU4 pins in the high-impedance state.
POE8#	Input	Input pin for the request signal to place the pins for MTU0 in high-impedance state. In accord with register settings, this pin is also capable of placing the MTU3 and MTU4 pins or MTU6 and MTU7 pins for MTU complementary PWM output in high-impedance state.
POE10#	Input	In accord with register settings, this pin is also capable of placing the MTU3 and MTU4 pins or MTU6 and MTU7 pins for MTU complementary PWM output, and MTU0 pins in high-impedance state.

Table 17.3 shows output level comparisons with pin combinations.

Table 17.3 Pin Combinations

Pin Combination	I/O	Description
MTIOC3B and MTIOC3D	Output	The MTU3 and MTU4 pins for MTU complementary PWM output set in the M3SELR, M4SELR1, and M4SELR2 registers are placed in high-impedance state when both pins of a pair simultaneously output the active level*1 for one or more cycles of the POE3_CLKM_POE (P0ϕ). Pin combinations for output comparison and high-impedance control can be selected by registers of POE3. *Note 1.* The low level is output when the OLSP bit in TOCR1A of MTUn is 0 with the TOCS bit in TOCR1A of MTUn cleared to 0, or the high level is output when the OLSP bit is 1. Otherwise, the low level is output when the OLS3N, OLS3P, OLS2N, OLS2P, OLS1N, and OLS1P bits in TOCR2A of MTUn are 0 with the TOCS bit in TOCR1A of MTUn set to 1, or the high level is output when these bits are 1.
MTIOC4A and MTIOC4C	Output	
MTIOC4B and MTIOC4D	Output	
MTIOC6B and MTIOC6D	Output	The MTU6 and MTU7 pins for MTU complementary PWM output are placed in high-impedance state when both pins of a pair simultaneously output the active level*1 for one or more cycles of the POE3_CLKM_POE (P0ϕ). Pin combinations for output comparison and high-impedance control can be selected by registers of POE3. *Note 1.* The low level is output when the OLSP bit in TOCR1B of MTUn is 0 with the TOCS bit in TOCR1B of MTUn cleared to 0, or the high level is output when the OLSP bit is 1. Otherwise, the low level is output when the OLS3N, OLS3P, OLS2N, OLS2P, OLS1N, and OLS1P bits in TOCR2B of MTUn are 0 with the TOCS bit in TOCR1B of MTUn set to 1, or the high level is output when these bits are 1.
MTIOC7A and MTIOC7C	Output	
MTIOC7B and MTIOC7D	Output	

17.2 Register Descriptions

Table 17.4 shows the register configuration. The POE3 registers are initialized by a reset.

Base Address : H'0_1004_9800 (Cortex-A55 Address Space)
Base Address : H'4004_9800 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
Base Address : H'5004_9800 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 17.4 Register configuration

Register Name	Abbreviation	Offset Address	Access size
Input level control/status register 1	ICSR1	H'00	16
Output level control/status register 1	OCSR1	H'02	16
Input level control/status register 2	ICSR2	H'04	16
Output level control/status register 2	OCSR2	H'06	16
Input level control/status register 3	ICSR3	H'08	16
Software port output enable register	SPOER	H'0A	8
Port output enable control register 1	POECR1	H'0B	8
Port output enable control register 2	POECR2	H'0C	16
Port output enable control register 4	POECR4	H'10	16
Port output enable control register 5	POECR5	H'12	16
Input level control/status register 4	ICSR4	H'16	16

17.2.1 Input Level Control/Status Register 1 (ICSR1)

ICSR1 selects the input modes for the POE0# pins, controls the enable/disable of interrupts, and indicates status.

Address(es): H'1004_9800

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	POE0F	—	—	—	PIE1	—	—	—	—	—	—	POE0M[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/(W)*2	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	POE0M[1:0]	All 0	R/W*1	POE0 Mode Select b1 b0 0 0: Accepts a request on the falling edge of POE0# input. 0 1: Accepts a request when POE0# input has been sampled 16 times at P0ϕ/4 clock pulses and all are low level. 1 0: Accepts a request when POE0# input has been sampled 16 times at P0ϕ/16 clock pulses and all are low level. 1 1: Accepts a request when POE0# input has been sampled 16 times at P0ϕ/128 clock pulses and all are low level.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	PIE1	0	R/W	Port Interrupt Enable 1 0: Interrupt requests disabled 1: Interrupt requests enabled
b11 to b9	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b12	POE0F	0	R/(W)*2	POE0 Flag 0: Indicates that a high-impedance request has not been input to the POE0# pin. 1: Indicates that a high-impedance request has been input to the POE0# pin.
b15 to b13	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bits are modified once, writing these bits does not modify their values until the register is reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

POE0M[1:0] Bits (POE0 Mode Select)

These bits select the input mode of the POE0# pin.

PIE1 Bit (Port Interrupt Enable 1)

This bit enables or disables interrupt requests when any one of the POE0F bit of the ICSR1 is set to 1.

POE0F Flag (POE0 Flag)

This flag indicates that a high-impedance request has been input to the POE0# pin.

[Setting condition]

- When the input set by POE0M[1:0] occurs at the POE0# pin

[Clearing condition]

- By writing 0 to POE0F after reading POE0F = 1

17.2.2 Input Level Control/Status Register 2 (ICSR2)

ICSR2 selects the input mode for the POE4# pin, controls the enable/disable of interrupts, and indicates status.

Address(es): H'1004_9804

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	POE4F	—	—	—	PIE2	—	—	—	—	—	—	POE4M[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/(W)*2	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	POE4M[1:0]	All 0	R/W*1	POE4 Mode Select b1 b0 0 0: Accepts a request on the falling edge of POE4# input 0 1: Accepts a request when POE4# input has been sampled 16 times at P0ϕ/4 clock pulses and all are low level. 1 0: Accepts a request when POE4# input has been sampled 16 times at P0ϕ/16 clock pulses and all are low level. 1 1: Accepts a request when POE4# input has been sampled 16 times at P0ϕ/128 clock pulses and all are low level.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	PIE2	0	R/W	Port Interrupt Enable 2 0: Interrupt requests disabled 1: Interrupt requests enabled
b11 to b9	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b12	POE4F	0	R/(W)*2	POE4 Flag 0: Indicates that a high-impedance request has not been input to the POE4# pin. 1: Indicates that a high-impedance request has been input to the POE4# pin.
b15 to b13	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bits are modified once, writing these bits does not modify their values until the register is reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

POE4M[1:0] Bits (POE4 Mode Select)

These bits select the input mode of the POE4# pin.

PIE2 Bit (Port Interrupt Enable 2)

This bit enables or disables interrupt requests when the POE4F flag in ICSR2 is set to 1.

POE4F Flag (POE4 Flag)

This flag indicates that a high-impedance request has been input to the POE4# pin.

[Setting condition]

- When the input set by POE4M[1:0] occurs at the POE4# pin

[Clearing condition]

- By writing 0 to POE4F after reading POE4F = 1

17.2.3 Input Level Control/Status Register 3 (ICSR3)

ICSR3 selects the input mode for the POE8# pin, controls the enable/disable of interrupts, and indicates status.

Address(es): H'1004_9808

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	POE8F	—	—	POE8E	PIE3	—	—	—	—	—	—	POE8M[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/(W)*2	R/W	R/W	R/W*1	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	POE8M[1:0]	All 0	R/W*1	POE8 Mode Select b1 b0 0 0: Accepts a request on the falling edge of POE8# input 0 1: Accepts a request when POE8# input has been sampled 16 times at P0ϕ/4 clock pulses and all are low level. 1 0: Accepts a request when POE8# input has been sampled 16 times at P0ϕ/16 clock pulses and all are low level. 1 1: Accepts a request when POE8# input has been sampled 16 times at P0ϕ/128 clock pulses and all are low level.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	PIE3	0	R/W	Port Interrupt Enable 3 0: Interrupt requests disabled 1: Interrupt requests enabled
b9	POE8E	0	R/W*1	POE8 High- Impedance Enable 0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.
b11, b10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b12	POE8F	0	R/(W)*2	POE8 Flag 0: Indicates that a high-impedance request has not been input to the POE8# pin. 1: Indicates that a high-impedance request has been input to the POE8# pin.
b15 to b13	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bits are modified once, writing these bits does not modify their values until the register is reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

POE8M[1:0] Bits (POE8 Mode Select)

These bits select the input mode of the POE8# pin.

PIE3 Bit (Port Interrupt Enable 3)

This bit enables or disables interrupt requests when the POE8F bit in ICSR3 is set to 1.

POE8E Bit (POE8 High-Impedance Enable)

This bit specifies whether to place the corresponding pin in high-impedance state when the POE8F bit is set to 1.

POE8F Flag (POE8 Flag)

This flag indicates that a high-impedance request has been input to the POE8# pin.

[Setting condition]

- When the input set by POE8M[1:0] occurs at the POE8# pin

[Clearing condition]

- By writing 0 to POE8F after reading POE8F = 1

17.2.4 Input Level Control/Status Register 4 (ICSR4)

ICSR4 selects the POE10# pin input mode, controls the enable/disable of interrupts, and indicates status.

Address(es): H'1004_9816

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	POE10F	—	—	POE10E	PIE4	—	—	—	—	—	—	POE10M[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/(W)*2	R/W	R/W	R/W*1	R/W*1	R/W	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	POE10M[1:0]	All 0	R/W*1	POE10 Mode Select b1 b0 0 0: Accepts a request on the falling edge of POE10# input 0 1: Accepts a request when POE10# input has been sampled 16 times at P0ϕ/4 clock pulses and all are low level. 1 0: Accepts a request when POE10# input has been sampled 16 times at P0ϕ/16 clock pulses and all are low level. 1 1: Accepts a request when POE10# input has been sampled 16 times at P0ϕ/128 clock pulses and all are low level.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	PIE4	0	R/W	Port Interrupt Enable 4 0: Interrupt requests disabled 1: Interrupt requests enabled
b9	POE10E	0	R/W*1	POE10 High- Impedance Enable 0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.
b11, b10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b12	POE10F	0	R/(W)*2	POE10 Flag 0: Indicates that a high-impedance request has not been input to the POE10# pin. 1: Indicates that a high-impedance request has been input to the POE10# pin.
b15 to b13	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bits are modified once, writing these bits does not modify their values until the register is reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

POE10M[1:0] Bits (POE10 Mode Select)

These bits select the input mode of the POE10# pin.

PIE4 Bit (Port Interrupt Enable 4)

This bit enables or disables interrupt requests when the POE10F bit is set to 1.

POE10E Bit (POE10 High-Impedance Enable)

This bit specifies whether to place the corresponding pin in high-impedance state when the POE10F bit is set to 1.

POE10F Bit (POE10 Flag)

This flag indicates that a request for the high-impedance state has been input to the POE10# pin.

[Setting condition]

- When the input set by POE10M[1:0] occurs at the POE10# pin

[Clearing condition]

- By writing 0 to POE10F after reading POE10F = 1

17.2.5 Output Level Control/Status Register 1 (OCSR1)

OCSR1 controls the enable/disable of output level comparison and interrupts, and indicates status.

Address(es): H'1004_9802

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	OSF1	—	—	—	—	—	OCE1	OIE1	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/(W)*2	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b7 to b0	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	OIE1	0	R/W	Output Short Interrupt Enable 1 0: Interrupt requests disabled 1: Interrupt requests enabled
b9	OCE1	0	R/W*1	Output Short High-Impedance Enable 1 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b14 to b10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b15	OSF1	0	R/(W)*2	Output Short Flag 1 0: Indicates that outputs have not simultaneously become an active level. 1: Indicates that outputs have simultaneously become an active level.

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

OIE1 Bit (Output Short Interrupt Enable 1)

This bit enables or disables interrupt requests when the OSF1 bit in OCSR1 is set to 1.

OCE1 Bit (Output Short High-Impedance Enable 1)

This bit specifies whether to place the pins in high-impedance state when the OSF1 bit in OCSR1 is set to 1.

OSF1 Flag (Output Short Flag 1)

This flag indicates that any one of the three pairs of two-phase MTU3 and MTU4 pins for MTU complementary PWM output to be compared has simultaneously become an active level.

[Setting condition]

- When any one of the three pairs of two-phase outputs has simultaneously become an active level

[Clearing condition]

- By writing 0 to OSF1 after reading OSF1 = 1

17.2.6 Output Level Control/Status Register 2 (OCSR2)

OCSR2 controls the enable/disable of output level comparison and interrupts, and indicates status.

Address(es): H'1004_9806

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	OSF2	—	—	—	—	—	OCE2	OIE2	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/(W)*2	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b7 to b0	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	OIE2	0	R/W	Output Short Interrupt Enable 2 0: Interrupt requests disabled 1: Interrupt requests enabled
b9	OCE2	0	R/W*1	Output Short High-Impedance Enable 2 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b14 to b10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b15	OSF2	0	R/(W)*2	Output Short Flag 2 0: Indicates that outputs have not simultaneously become an active level. 1: Indicates that outputs have simultaneously become an active level.

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

OIE2 Bit (Output Short Interrupt Enable 2)

This bit enables or disables interrupt requests when the OSF2 bit in OCSR2 is set to 1.

OCE2 Bit (Output Short High-Impedance Enable 2)

This bit specifies whether to place the pins in high-impedance state when the OSF2 bit in OCSR2 is set to 1.

OSF2 Flag (Output Short Flag 2)

This flag indicates that any one of the three pairs of two-phase MTU6 and MTU7 pins for MTU complementary PWM output to be compared has simultaneously become an active level.

[Setting condition]

- When any one of the three pairs of two-phase outputs has simultaneously become an active level

[Clearing condition]

- By writing 0 to OSF2 after reading OSF2 = 1

17.2.7 Software Port Output Enable Register (SPOER)

SPOER controls high-impedance state of the pins.

Address(es): H'1004_980A

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	MTUCH0HIZ	MTUCH67HIZ	MTUCH34HIZ
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	MTUCH34HIZ	0	R/W	MTU3 or MTU4 Output High-Impedance Enable 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b1	MTUCH67HIZ	0	R/W	MTU6 and MTU7 Output High- Impedance Enable 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b2	MTUCH0HIZ	0	R/W	MTU0 Output High-Impedance Enable 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

MTUCH34HIZ Bit (MTU3 or MTU4 Output High-Impedance Enable)

This bit specifies whether to place the MTU complementary PWM output pins (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D) in high-impedance state.

[Setting condition]

- By writing 1 to MTUCH34HIZ

[Clearing conditions]

- Reset
- By writing 0 to MTUCH34HIZ after reading MTUCH34HIZ = 1*1

Note 1. To write 0 to this bit, be sure to read 1 and then write 0.

MTUCH67HIZ Bit (MTU6 and MTU7 Output High-Impedance Enable)

This bit specifies whether to place the MTU complementary PWM output pins (MTIOC6B, MTIOC6D, MTIOC7A, MTIOC7B, MTIOC7C, and MTIOC7D) in high-impedance state.

[Setting condition]

- By writing 1 to MTUCH67HIZ

[Clearing conditions]

- Reset
- By writing 0 to MTUCH67HIZ after reading MTUCH67HIZ = 1*1

Note 1. To write 0 to this bit, be sure to read 1 and then write 0.

MTUCH0HIZ Bit (MTU0 Output High-Impedance Enable)

This bit specifies whether to place the MTU0 pins in high-impedance state.

[Setting condition]

- By writing 1 to MTUCH0HIZ

[Clearing conditions]

- Reset
- By writing 0 to MTUCH0HIZ after reading MTUCH0HIZ = 1*1

Note 1. To write 0 to this bit, be sure to read 1 and then write 0.

17.2.8 Port Output Enable Control Register 1 (POECR1)

POECR1 controls high-impedance state of the MTU0 pins.

Address(es): H'1004_980B

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	MTU0DZE	MTU0CZE	MTU0BZE	MTU0AZE
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b0	MTU0AZE	0	R/W*1	MTIOC0A High-Impedance Enable 0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.
b1	MTU0BZE	0	R/W*1	MTIOC0B High-Impedance Enable 0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.
b2	MTU0CZE	0	R/W*1	MTIOC0C High-Impedance Enable 0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.
b3	MTU0DZE	0	R/W*1	MTIOC0D High-Impedance Enable 0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.
b7 to b4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

MTU0AZE Bit (MTIOC0A High-Impedance Enable)

This bit specifies whether to place the MTIOC0A output for the MTU0 pin in high-impedance state when any of the POE8F flag in ICSR3, MTUCH0HIZ bit in SPOER or, as additionally specified in POECR5, the POEmF (n = 1, 2, 4; m = 0, 4, 10) flag in ICSRn, is set to 1.

MTU0BZE Bit (MTIOC0B High-Impedance Enable)

This bit specifies whether to place the MTIOC0B output for the MTU0 pin in high-impedance state when any of the POE8F flag in ICSR3, MTUCH0HIZ bit in SPOER or, as additionally specified in POECR5, the POEmF (n = 1, 2, 4; m = 0, 4, 10) flag in ICSRn, is set to 1.

MTU0CZE Bit (MTIOC0C High-Impedance Enable)

This bit specifies whether to place the MTIOC0C output for the MTU0 pin in high-impedance state when any of the POE8F flag in ICSR3, MTUCH0HIZ bit in SPOER or, as additionally specified in POECR5, the POEmF (n = 1, 2, 4; m = 0, 4, 10) flag in ICSRn, is set to 1.

MTU0DZE Bit (MTIOC0D High-Impedance Enable)

This bit specifies whether to place the MTIOC0D output for the MTU0 pin in high-impedance state when any of the POE8F flag in ICSR3, MTUCH0HIZ bit in SPOER or, as additionally specified in POECR5, the POEmF (n = 1, 2, 4; m = 0, 4, 10) flag in ICSRn, is set to 1.

17.2.9 Port Output Enable Control Register 2 (POECR2)

POECR2 controls high-impedance state of the MTU complementary PWM output pins (MTU3, MTU4, MTU6, and MTU7 pins).

Address(es): H'1004_980C

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	MTU3BDZE	MTU4ACZE	MTU4BDZE	—	—	—	—	—	MTU6BDZE	MTU7ACZE	MTU7BDZE
Initial Value	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1	R/W*1	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b0	MTU7BDZE	1	R/W*1	MTIOC7B/7D High-Impedance Enable*2 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b1	MTU7ACZE	1	R/W*1	MTIOC7A/7C High-Impedance Enable*2 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b2	MTU6BDZE	1	R/W*1	MTIOC6B/6D High-Impedance Enable*2 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	MTU4BDZE	1	R/W*1	MTIOC4B/4D High-Impedance Enable 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b9	MTU4ACZE	1	R/W*1	MTIOC4A/4C High-Impedance Enable 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b10	MTU3BDZE	1	R/W*1	MTIOC3B/3D High-Impedance Enable 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b15 to b11	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

Note 2. Set this bit to 0 when MTU6 or MTU7 is not used.

MTU7BDZE Bit (MTIOC7B/7D High-Impedance Enable)

This bit specifies whether to place the MTIOC7B output and MTIOC7D output for the MTU3 pin in high-impedance state when any one of the OSF2 flag in OCSR2, POE4F flag in ICSR2, MTUCH67HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 1, 3, 4; m = 0, 8, 10) flag in ICSRn, is set to 1.

MTU7ACZE Bit (MTIOC7A/7C High-Impedance Enable)

This bit specifies whether to place the MTIOC7A output and MTIOC7C output for the MTU3 pin in high-impedance state when any one of the OSF2 flag in OCSR2, POE4F flag in ICSR2, MTUCH67HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 1, 3, 4; m = 0, 8, 10) flag in ICSRn, is set to 1.

MTU6BDZE Bit (MTIOC6B/6D High-Impedance Enable)

This bit specifies whether to place the MTIOC6B output and MTIOC6D output for the MTU3 pin in high-impedance state when any one of the OSF2 flag in OCSR2, POE4F flag in ICSR2, MTUCH67HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 1, 3, 4; m = 0, 8, 10) flag in ICSRn, is set to 1.

MTU4BDZE Bit (MTIOC4B/4D High-Impedance Enable)

This bit specifies whether to place the MTIOC4B output and MTIOC4D output for the MTU4 pin in high-impedance state when any one of the OSF1 flag in OCSR1, POE0F flag in ICSR1, MTUCH34HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 2 to 4; m = 4, 8, 10) flag in ICSRn, is set to 1.

MTU4ACZE Bit (MTIOC4A/4C High-Impedance Enable)

This bit specifies whether to place the MTIOC4A output and MTIOC4C output for the MTU4 pin in high-impedance state when any one of the OSF1 flag in OCSR1, POE0F flag in ICSR1, MTUCH34HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 2 to 4; m = 4, 8, 10) flag in ICSRn, is set to 1.

MTU3BDZE Bit (MTIOC3B/3D High-Impedance Enable)

This bit specifies whether to place the MTIOC3B output and MTIOC3D output for the MTU3 pin in high-impedance state when any one of the OSF1 flag in OCSR1, POE0F flag in ICSR1, MTUCH34HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 2 to 4; m = 4, 8, 10) flag in ICSRn, is set to 1.

17.2.10 Port Output Enable Control Register 4 (POECR4)

The POECR4 is used to extend the control conditions of the high-impedance state for the MTU3, MTU4, MTU6, and MTU7 pins for the MTU complementary PWM output.

For details about the targets and conditions of high-impedance control, see **Figure 17.2**.

Address(es): H'1004_9810

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	IC4ADDMT67	IC3ADDMT67	—	IC1ADDMT67	—	—	—	—	IC4ADDMT34	IC3ADDMT34	IC2ADDMT34	—	—
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0
R/W	R/W	R/W	R/W	R/W*1	R/W*1	R/W	R/W*1	R/W	R/W	R/W	R/W	R/W*1	R/W*1	R/W*1	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b1	—	1	R/W	Reserved This bit is read as 1. The write value should be 1.
b2	IC2ADDMT34ZE	0	R/W*1	MTU3 and MTU4 High- Impedance POE4F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b3	IC3ADDMT34ZE	0	R/W*1	MTU3 and MTU4 High- Impedance POE8F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b4	IC4ADDMT34ZE	0	R/W*1	MTU3 and MTU4 High- Impedance POE10F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b8 to b5	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0
b9	IC1ADDMT67ZE	0	R/W*1	MTU6 and MTU7 High- Impedance POE0F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b10	—	1	R/W	Reserved This bit is read as 1. The write value should be 1.
b11	IC3ADDMT67ZE	0	R/W*1	MTU6 and MTU7 High- Impedance POE8F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b12	IC4ADDMT67ZE	0	R/W*1	MTU6 and MTU7 High- Impedance POE10F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b15 to b13	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

IC2ADDMT34ZE Bit (MTU3 and MTU4 High-Impedance POE4F Add)

Adds the POE4F flag in ICSR2 (POE4#) to the high-impedance control conditions for the MTU3 and MTU4 pins (MTIOC3B/MTIOC3D/MTIOC4A/MTIOC4C/MTIOC4B/MTIOC4D).

IC3ADDMT34ZE Bit (MTU3 and MTU4 High-Impedance POE8F Add)

Adds the POE8F flag in ICSR3 (POE8#) to the high-impedance control conditions for the MTU3 and MTU4 pins (MTIOC3B/MTIOC3D/MTIOC4A/MTIOC4C/MTIOC4B/MTIOC4D).

IC4ADDMT34ZE Bit (MTU3 and MTU4 High-Impedance POE10F Add)

Adds the POE10F flag in ICSR4 (POE10#) to the high-impedance control conditions for the MTU3 and MTU4 pins (MTIOC3B/MTIOC3D/MTIOC4A/MTIOC4C/MTIOC4B/MTIOC4D).

IC1ADDMT67ZE Bit (MTU6 and MTU7 High-Impedance POE0F Add)

Adds the POE0F flag in ICSR1 (POE0#) to the high-impedance control conditions for the MTU6, and MTU7 pins (MTIOC6B/MTIOC6D/MTIOC7A/MTIOC7C/MTIOC7B/MTIOC7D).

IC3ADDMT67ZE Bit (MTU6 and MTU7 High-Impedance POE8F Add)

Adds the POE8F flag in ICSR3 (POE8#) to the high-impedance control conditions for the MTU6, and MTU7 pins (MTIOC6B/MTIOC6D/MTIOC7A/MTIOC7C/MTIOC7B/MTIOC7D).

IC4ADDMT67ZE Bit (MTU6 and MTU7 High-Impedance POE10F Add)

Adds the POE10F flag in ICSR4 (POE10#) to the high-impedance control conditions for the MTU6, and MTU7 pins (MTIOC6B/MTIOC6D/MTIOC7A/MTIOC7C/MTIOC7B/MTIOC7D).

17.2.11 Port Output Enable Control Register 5 (POECR5)

The POECR5 is used to extend the control conditions of the high-impedance for the MTU0 pin.

For details about the targets and conditions of high-impedance control, see **Figure 17.2**.

Address(es): H'1004_9812

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	IC4ADDMT0ZE	—	IC2ADDMT0ZE	IC1ADDMT0ZE	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W	R/W*1	R/W*1	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b1	IC1ADDMT0ZE	0	R/W*1	MTU0 High-Impedance POE0F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b2	IC2ADDMT0ZE	0	R/W*1	MTU0 High-Impedance POE4F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b3	—	1	R/W	Reserved This bit is read as 1. The write value should be 1.
b4	IC4ADDMT0ZE	0	R/W*1	MTU0 High-Impedance POE10F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b15 to b5	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

IC1ADDMT0ZE Bit (MTU0 High-Impedance POE0F Add)

Adds the POE0F flag in ICSR1 (POE0#) to the high-impedance control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

IC2ADDMT0ZE Bit (MTU0 High-Impedance POE4F Add)

Adds the POE4F flag in ICSR2 (POE4#) to the high-impedance control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

IC4ADDMT0ZE Bit (MTU0 High-Impedance POE10F Add)

Adds the POE10F flag in ICSR4 (POE10#) to the high-impedance control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

17.3 Operation

Table 17.5 shows the target pins for high-impedance control and conditions to place the pins in high-impedance state.

Table 17.5 Target Pins and Conditions for High-Impedance Control (1/2)

Pins	Conditions	Detailed Conditions
MTU3 pins (MTIOC3B and MTIOC3D)	• Operation for detection of the POE0# input level • Operation for comparison of the output levels on the MTIOC3B and MTIOC3D pins • SPOER setting • Additional conditions of the POECR4	MTU3BDZE• ((POE0F) +(OSF1•OCE1) +(MTUCH34HIZ) +(IC2ADDMT34ZE•POE4F) +(IC3ADDMT34ZE•POE8E•POE8F) +(IC4ADDMT34ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU4 pins (MTIOC4A and MTIOC4C)	• Operation for detection of the POE0# input level • Operation for comparison of the output levels on the MTIOC4A and MTIOC4C pins • SPOER setting • Additional conditions of the POECR4	MTU4ACZE• ((POE0F) +(OSF1•OCE1) + (MTUCH34HIZ) +(IC2ADDMT34ZE•POE4F) +(IC3ADDMT34ZE•POE8E•POE8F) +(IC4ADDMT34ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU4 pins (MTIOC4B and MTIOC4D)	• Operation for detection of the POE0# input level • Operation for comparison of the output levels on the MTIOC4B and MTIOC4D pins • SPOER setting • Additional conditions of the POECR4	MTU4BDZE• ((POE0F) +(OSF1•OCE1) +(MTUCH34HIZ) +(IC2ADDMT34ZE•POE4F) +(IC3ADDMT34ZE•POE8E•POE8F) +(IC4ADDMT34ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU6 pins (MTIOC6B and MTIOC6D)	• Operation for detection of the POE4# input level • Operation for comparison of the output levels on the MTIOC6B and MTIOC6D pins • SPOER setting • Additional conditions of the POECR4	MTU6BDZE• ((POE4F) +(OSF2•OCE2) +(MTUCH67HIZ) +(IC1ADDMT67ZE•POE0F) +(IC3ADDMT67ZE•POE8E•POE8F) +(IC4ADDMT67ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU7 pins (MTIOC7A and MTIOC7C)	• Operation for detection of the POE4# input level • Operation for comparison of the output levels on the MTIOC7A and MTIOC7C pins • SPOER setting • Additional conditions of the POECR4	MTU7ACZE• ((POE4F) +(OSF2•OCE2) +(MTUCH67HIZ) +(IC1ADDMT67ZE•POE0F) +(IC3ADDMT67ZE•POE8E•POE8F) +(IC4ADDMT67ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU7 pins (MTIOC7B and MTIOC7D)	• Operation for detection of the POE4# input level • Operation for comparison of the output levels on the MTIOC7B and MTIOC7D pins • SPOER setting • Additional conditions of the POECR4	MTU7BDZE• ((POE4F) +(OSF2•OCE2) +(MTUCH67HIZ) +(IC1ADDMT67ZE•POE0F) +(IC3ADDMT67ZE•POE8E•POE8F) +(IC4ADDMT67ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU0 pin (MTIOC0A)	• Operation for detection of the POE8# input level • SPOER setting • Additional conditions of the POECR5	MTU0AZE• ((POE8F•POE8E) +(MTUCH0HIZ) +(IC1ADDMT0ZE•POE0F) +(IC2ADDMT0ZE•POE4F) +(IC4ADDMT0ZE•ICSR4.POE10E•ICSR4.POE10F)

Table 17.5 Target Pins and Conditions for High-Impedance Control (2/2)

Pins	Conditions	Detailed Conditions
MTU0 pin (MTIOC0B)	• Operation for detection of the POE8# input level • SPOER setting • Additional conditions of the POECR5	MTU0BZE• ((POE8F•POE8E) +(MTUCH0HIZ) +(IC1ADDMT0ZE•POE0F) +(IC2ADDMT0ZE•POE4F) +(IC4ADDMT0ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU0 pin (MTIOC0C)	• Operation for detection of the POE8# input level • SPOER setting • Additional conditions of the POECR5	MTU0CZE• ((POE8F•POE8E) +(MTUCH0HIZ) +(IC1ADDMT0ZE•POE0F) +(IC2ADDMT0ZE•POE4F) +(IC4ADDMT0ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU0 pin (MTIOC0D)	• Operation for detection of the POE8# input level • SPOER setting • Additional conditions of the POECR5	MTU0DZE• ((POE8F•POE8E) +(MTUCH0HIZ) +(IC1ADDMT0ZE•POE0F) +(IC2ADDMT0ZE•POE4F) +(IC4ADDMT0ZE•ICSR4.POE10E•ICSR4.POE10F)

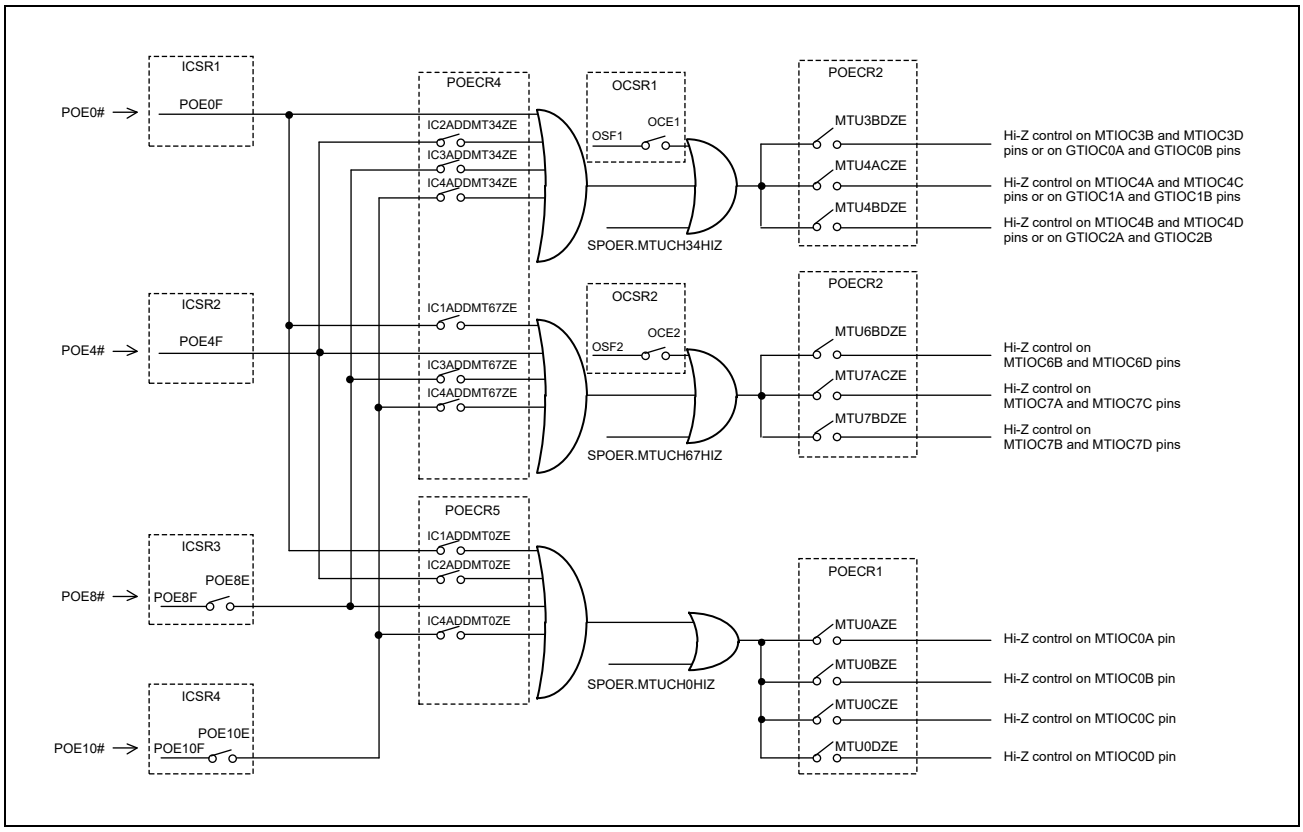


Figure 17.2 Target Pins and Conditions for High-Impedance Control

High-impedance requests to individual pins can be controlled by the settings in the POECR1 to POECR2 registers.

The following input pins can be added to high-impedance control conditions by the settings in the POECR4 to POECR5 registers: input pins other than the POE0# pin for the MTU3 and MTU4 pins; input pins other than the POE4# pin for the MTU6 and MTU7 pins; input pins other than the POE8# pin for the MTU0 pins.

For example, setting the IC2ADDMT34ZE bit in POECR4 to 1 outputs high-impedance requests to the MTU3 and MTU4 pins even when the POE4# input is detected.

High-impedance requests due to the detection of POE8# input or POE10# input can be controlled by the settings in the ICSR3 to ICSR4 registers. (The ICSR1 and ICSR2 registers do not control enabling or disabling of output of high-impedance requests.)

High-impedance requests to the MTU3, MTU4, MTU6, and MTU7 pins as the results of output level comparison can be controlled by the settings in the OCSR1 and OCSR2 registers.

17.3.1 MTU Pin Selection

In this LSI, each terminal function for MTU is assigned to multiple ports. **Table 17.6** shows the correspondence table of MTU terminal functions and compatible ports. The pin used as the MTU must be set separately in the general-purpose I/O port register. (Set by PMCn and PFCm registers of IO_TOP)

Table 17.6 Correspondence between MTU Pins (1/2)

MTU Pin Functions	Corresponding Ports
MTCLKA	P2_0
MTCLKB	P2_1
MTCLKC	P2_2
MTCLKD	P2_3
MTIOC0A	P3_0
	P7_0
	P12_0
MTIOC0B	P3_1
	P7_1
	P12_1
MTIOC0C	P3_2
	P7_2
MTIOC0D	P3_3
	P7_3
MTIOC1A	P0_2
	P1_0
MTIOC1B	P0_3
	P1_1
MTIOC2A	P0_0
	P4_0
MTIOC2B	P0_1
	P4_1
MTIOC3A	P4_2
	P17_0
MTIOC3B	P4_3
	P17_1
MTIOC3C	P4_4
	P17_2
MTIOC3D	P4_5
	P17_3
MTIOC4A	P9_0
	P13_0
MTIOC4B	P9_1
	P13_1
MTIOC4C	P9_2
	P13_2
MTIOC4D	P9_3
	P13_3

Table 17.6 Correspondence between MTU Pins (2/2)

MTU Pin Functions	Corresponding Ports
MTIC5U	P1_2
	P14_0
MTIC5V	P1_3
	P14_1
MTIC5W	P1_4
	P14_2
MTIOC6A	P10_0
MTIOC6B	P10_1
MTIOC6C	P10_2
MTIOC6D	P10_3
MTIOC7A	P5_0
	P6_0
	P8_0
MTIOC7B	P5_1
	P6_1
	P8_1
MTIOC7C	P5_2
	P6_2
	P8_2
MTIOC7D	P5_3
	P6_3
	P8_3
MTIOC8A	P4_0
	P15_0
MTIOC8B	P4_1
	P15_1
MTIOC8C	P4_2
	P15_2
MTIOC8D	P4_3
	P15_3

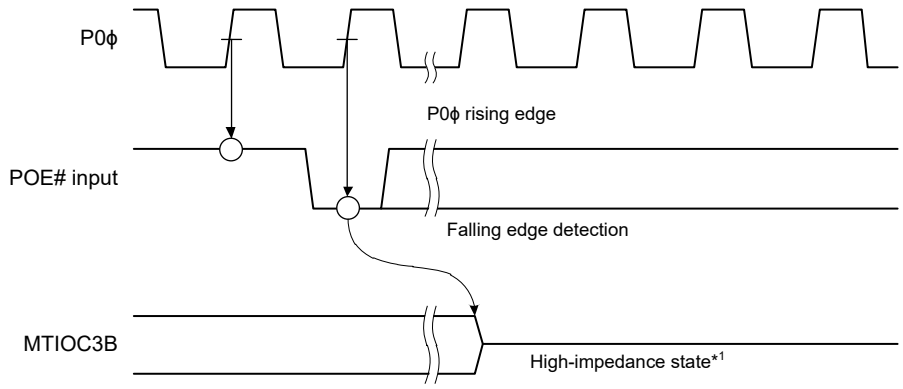
17.3.2 Input Level Detection Operation

If the input conditions set by ICSR1 to ICSR4 occur on the POE0#, POE4#, POE8#, and POE10# pins, the MTU3 and MTU4 or MTU6 and MTU7 pins for the MTU complementary PWM output, and MTU0 pin are placed in high-impedance state. Note however, that these pins are still placed in the high-impedance state even when the MTU functions are not selected for the pins.

(1) Falling Edge Detection

When a change from a high to low level is input to the POE0#, POE4#, POE8#, and POE10# pins, the pins for the MTU complementary PWM output, and pin functions multiplexed with the MTU0 pins are placed in high-impedance state.

Figure 17.3 shows a sample timing after the level changes in input to the POE0#, POE4#, POE8#, and POE10# pins until the respective pins enter high-impedance state.

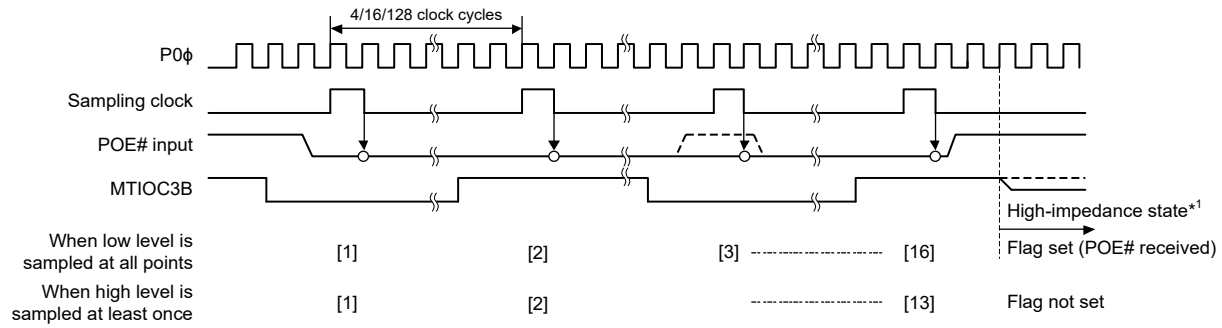


Note 1. Other pins also enter the high-impedance state at the same timing.

Figure 17.3 Falling Edge Detection

(2) Low-Level Detection

Figure 17.4 shows the low-level detection operation. Sixteen continuous low levels are sampled with the sampling clock selected by ICSR1 to ICSR4. If even one high level is detected during this interval, the low level is not accepted. The timing when pins for the MTU complementary PWM output, and MTU0 pins enter the high-impedance state after the sampling clock is input is the same in both falling-edge detection and in low-level detection.



Note 1. Other pins also enter the high-impedance state at the same timing.

Figure 17.4 Low-Level Detection Operation

17.3.3 Output Level Compare Operation

Figure 17.5 shows an example of the output level compare operation for the combination of MTIOC3B and MTIOC3D. The operation is the same for the other pin combinations.

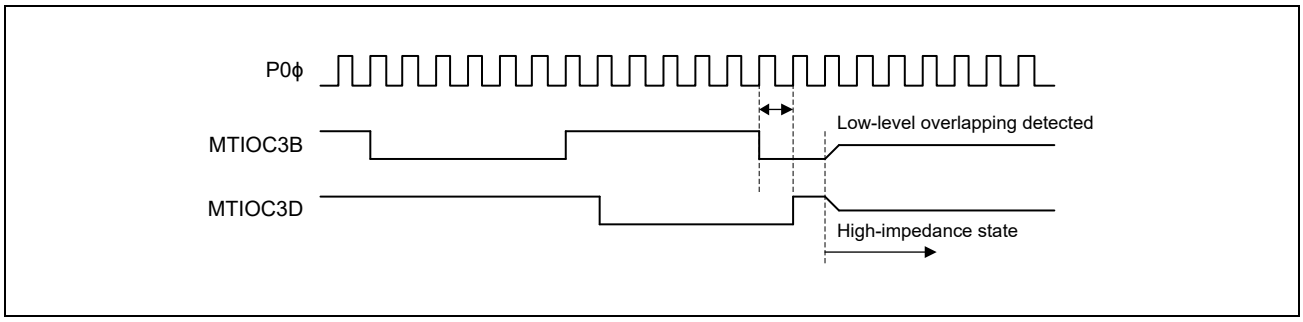


Figure 17.5 Output Level Compare Operation

17.3.4 High-Impedance Control Using Registers

The high-impedance state of the MTU pins (MTU0, MTU3, MTU4, MTU6, and MTU7) can be directly controlled by using the software port output enable register (SPOER).

For instance, setting the MTUCH34HIZ bit in SPOER to 1 places the MTU3 and MTU4 pins specified by the port output enable control register 2 (POECR2) in the high-impedance state.

The high-impedance state of other pins can also be controlled by setting the appropriate bits in SPOER.

17.3.5 Additional Functions for Controlling High-Impedance States

Settings in port enable registers 4 to 5 (POECR4 and POECR5) can add further high-impedance control conditions for the MTU complementary PWM output and MTU0 pins.

For instance, the settings listed below can be added as high-impedance control conditions for the MTU3 and MTU4 pins.

- Setting the IC2ADDMT34ZE bit in POECR4 to 1 adds the input level detection by the POE4#
- Setting the IC3ADDMT34ZE bit in POECR4 to 1 and adds the input level detection by the POE8#
- Setting the IC4ADDMT34ZE bit in POECR4 to 1 and adds the input level detection by the POE10# (ICSR4.POE10F)

The high-impedance state of other pins can also be controlled by setting the appropriate bits in the POECR4 and POECR5.

17.3.6 Release from High-Impedance State

MTU pins which have entered high-impedance state due to input-level detection can be released from the state either by returning them to their initial state with a reset, or by clearing all of the ICSR1.POE0F, ICSR2.POE4F, ICSR3.POE8F, and ICSR4.POE10F flags. However, note that when low-level sampling is selected with the ICSR1.POE0M[1:0], ICSR2.POE4M[1:0], ICSR3.POE8M[1:0], ICSR4.POE10M[1:0], and ICSR5.POE10M[1:0] bits, just writing 0 to a flag is ignored (the flag is not cleared); flags can be cleared by writing 0 to it only after a high level is input to the POE0#, POE4#, POE8#, and POE10# pins and is sampled.

MTU pins which have entered high-impedance state due to output level detection can be released from the state either by returning them to their initial state with a reset, or by clearing the OCSR1.OSF1 flag or the OCSR2.OSF2 flag.

17.4 POE3 Setting Procedure

Figure 17.6 shows the POE3 configuration procedure. As an example, high impedance control by comparing the output levels of the MTU 3 terminals (MTIOC3B / MTIOC3D) is shown. In **Figure 17.6**, select P32_1 for the MTIOC3B terminal and P36_1 for the MTIOC3D terminal.

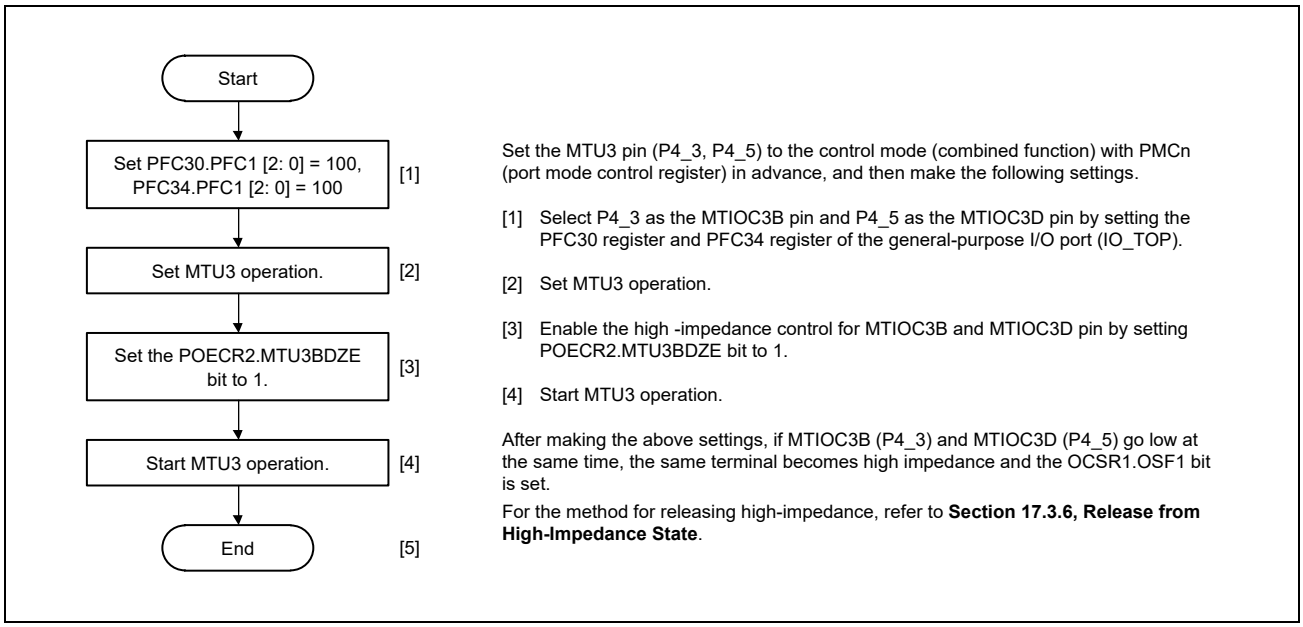


Figure 17.6 Procedure for Setting POE3

17.5 Interrupts

The POE3 issues a request to generate an interrupt when the specified condition is satisfied during input level detection or output level comparison. **Table 17.7** shows the interrupt sources and their conditions.

Table 17.7 Interrupt Sources and Conditions

Name	Interrupt Source	Interrupt Flag	Condition
OEI1	Output enable interrupt 1	POE0F and OSF1	PIE1•POE0F+OIE1•OSF1
OEI2	Output enable interrupt 2	POE4F and OSF2	PIE2•POE4F+OIE2•OSF2
OEI3	Output enable interrupt 3	POE8F	PIE3•POE8F
OEI4	Output enable interrupt 4	ICSR4.POE10F	PIE4•ICSR4.POE10F

17.6 Usage Notes

17.6.1 High-Impedance Control when MTU6 and MTU7 are not Used

When MTU6 and MTU7 are not to be used, set the POECR2.MTU6BDZE, MTU7ACZE, and MTU7BDZE bits to 0 to disable high-impedance control.

18. General PWM Timer (GPT)

18.1 Overview

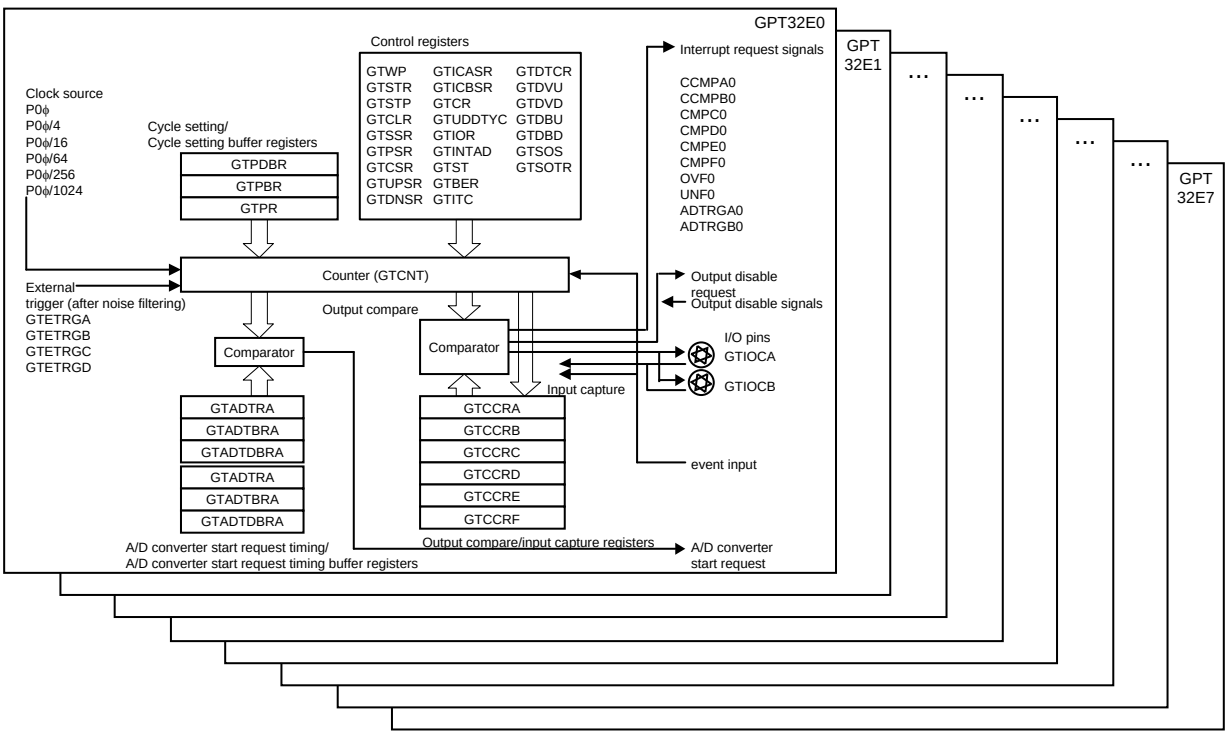
This LSI has a general purpose PWM timer (GPT) composed of 8 channels of 32-bit timer (GPT32E). **Table 18.1** lists the GPT specifications, **Table 18.2** shows the GPT functions, **Figure 18.1** shows the block diagram, and **Table 18.3** lists the I/O pins.

Table 18.1 GPT specifications

Parameter	Specifications
Functions	• 32 bits × 8 channels • Up-counting or down-counting (saw waves) or up/down-counting (triangle waves) for each counter. • Clock sources independently selectable for each channel • Two I/O pins per channel • Two output compare/input capture registers per channel • For the two output compare/input capture registers of each channel, four registers are provided as buffer registers and are capable of operating as comparison registers when buffering is not in use. • In output compare operation, buffer switching can be at crests or troughs, enabling the generation of laterally asymmetric PWM waveforms. • Registers for setting up frame cycles in each channel (with capability for generating interrupts at overflow or underflow) • Generation of dead times in PWM operation • Synchronous starting, stopping and clearing counters for arbitrary channels • Starting, stopping, clearing and up/down counters in response to input level comparison • Starting, clearing, stopping and up/down counters in response to a maximum of four external triggers • Output pin disable function by dead time error and detected short-circuits between output pins • A/D converter start triggers can be generated (GPT32E0 to GPT32E3) • Enables the noise filter for input capture and external trigger operation

Table 18.2 GPT functions

Parameter		GPT32E
Count clock		Pϕ Pϕ/4 Pϕ/16 Pϕ/64 Pϕ/256 Pϕ/1024
Output compare/input capture registers (GTCCR)		GTCCRA GTCCRB
Compare/buffer registers		GTCCRC GTCCRD GTCCRE GTCCRF
Cycle setting register		GTPR
Cycle setting buffer registers		GTPBR GTPDBR
I/O pins		GTIOCA GTIOCB
External trigger input pin		GTETRGA GTETRGB GTETRGC GTETRGD
Counter clear sources		GTPR register compare match, input capture, input pin status, or input on the GTETRGA, GTETRGB, GTETRGC, or GTETRGD pins
Compare match output	Low output	Available
	High output	Available
	Toggle output	Available
Input capture function		Available
Automatic addition of dead time		Available
PWM mode		Available
Phase count function		Available
Buffer operation		Double buffer
One-shot operation		Available
DMA activation		All the interrupt sources
A/D converter start trigger		Compare match of GTADTRA or GTADTRB (Channel 0 to 3)
Interrupt sources		10 sources • GTCCRA compare match/input capture (CCMPAn) • GTCCRB compare match/input capture (CCMPBn) • GTCCRC compare match (CMPCn) • GTCCRD compare match (CMPDn) • GTCCRE compare match (CMPEn) • GTCCRF compare match (CMPFn) • GTADTRA compare match (ADTRGAn) • GTADTRB compare match (ADTRGBn) • GTCNT overflow (GTPR compare match) (OVFn) • GTCNT underflow (UNFn)
Interrupt skipping function		Skips GTCNT overflows (GTPR compare match) (OVFn)/GTCNT underflow (UNFn) interrupts (with interlocking function for other interrupts or A/D conversion requests).
Noise filtering function		Available



GTWP: General PWM Timer Write-Protection Register
GTSTR: General PWM Timer Software Start Register
GTSTP: General PWM Timer Software Stop Register
GTCLR: General PWM Timer Software Clear Register
GTSSR: General PWM Timer Start Source Select Register
GTPSR: General PWM Timer Stop Source Select Register
GTCSR: General PWM Timer Clear Source Select Register
GTUPSR: General PWM Timer Up Count Source Select Register
GTDNSR: General PWM Timer Down Count Source Select Register
GTICASR: General PWM Timer Input Capture Source Select Register A
GTICBSR: General PWM Timer Input Capture Source Select Register B
GTCR: General PWM Timer Control Register
GTUDDTYC: General PWM Timer Count Direction and Duty Setting Register
GTIOR: General PWM Timer I/O Control Register
GTINTAD: General PWM Timer Interrupt Output Setting Register
GTST: General PWM Timer Status Register
GTBER: General PWM Timer Buffer Enable Register
GTITC: General PWM Timer Interrupt and A/D Converter Start Request Skipping Setting Register
GTCNT: General PWM Timer Counter
GTCCRA: General PWM Timer Compare Capture Register A
GTCCRB: General PWM Timer Compare Capture Register B
GTCCRC: General PWM Timer Compare Capture Register C
GTCCRD: General PWM Timer Compare Capture Register D
GTCCRE: General PWM Timer Compare Capture Register E
GTCCRF: General PWM Timer Compare Capture Register F
GTPR: General PWM Timer Cycle Setting Register
GTPBR: General PWM Timer Cycle Setting Buffer Register
GTPDBR: General PWM Timer Cycle Setting Double-Buffer Register
GTADTRA: General PWM Timer A/D Converter Start Request Timing Register A
GTADTBRA: General PWM Timer A/D Converter Start Request Timing Buffer Register A
GTADTDBRA: General PWM Timer A/D Converter Start Request Timing Double-Buffer Register A
GTADTRB: General PWM Timer A/D Converter Start Request Timing Register B
GTADTBRB: General PWM Timer A/D Converter Start Request Timing Buffer Register B
GTADTDBRB: General PWM Timer A/D Converter Start Request Timing Double-Buffer Register B
GTDTCR: General PWM Timer Dead Time Control Register
GTDVU: General PWM Timer Dead Time Value Register U
GTDVD: General PWM Timer Dead Time Value Register D
GTDBU: General PWM Timer Dead Time Buffer Register U
GTDBD: General PWM Timer Dead Time Buffer Register D
GTSOS: General PWM Timer Output Protection Function Status Register
GTSOTR: General PWM Timer Output Protection Function Temporary Release Register

Figure 18.1 GPT Block Diagram

Table 18.3 GPT I/O pins

Channel	Pin name	I/O	Function
Shared	GTETRGA	Input	External trigger input pin A (after noise filtering)
	GTETRGB	Input	External trigger input pin B (after noise filtering)
	GTETRGC	Input	External trigger input pin C (after noise filtering)
	GTETRGD	Input	External trigger input pin D (after noise filtering)
GPT32E0	GTIOC0A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC0B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E1	GTIOC1A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC1B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E2	GTIOC2A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC2B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E3	GTIOC3A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC3B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E4	GTIOC4A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC4B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E5	GTIOC5A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC5B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E6	GTIOC6A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC6B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E7	GTIOC7A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC7B	I/O	GTCCRB register input capture input/output compare output/PWM output pin

18.2 Register Descriptions

Table 18.4 lists the registers in the GPT.

Base Address: H'0_1004_8000 (Cortex-A55 Address Space)
Base Address: H'4004_8000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
Base Address: H'5004_8000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 18.4 GPT registers (1/2)

Module symbol	Register name	Register symbol	Reset value	Offset Address (m = 0 to 7)	Access size
GPT32Em (m = 0 to 7)	General PWM Timer Write-Protection Register	GTWP	H'0000_0000	H'00 + H'0100 × m	32
	General PWM Timer Software Start Register	GTSTR	H'0000_0000	H'04 + H'0100 × m	32
	General PWM Timer Software Stop Register	GTSTP	H'FFFF_FFFF	H'08 + H'0100 × m	32
	General PWM Timer Software Clear Register	GTCLR	H'0000_0000	H'0C + H'0100 × m	32
	General PWM Timer Start Source Select Register	GTSSR	H'0000_0000	H'10 + H'0100 × m	32
	General PWM Timer Stop Source Select Register	GTPSR	H'0000_0000	H'14 + H'0100 × m	32
	General PWM Timer Clear Source Select Register	GTCSR	H'0000_0000	H'18 + H'0100 × m	32
	General PWM Timer Up Count Source Select Register	GTUPSR	H'0000_0000	H'1C + H'0100 × m	32
	General PWM Timer Down Count Source Select Register	GTDNSR	H'0000_0000	H'20 + H'0100 × m	32
	General PWM Timer Input Capture Source Select Register A	GTICASR	H'0000_0000	H'24 + H'0100 × m	32
	General PWM Timer Input Capture Source Select Register B	GTICBSR	H'0000_0000	H'28 + H'0100 × m	32
	General PWM Timer Control Register	GTCR	H'0000_0000	H'2C + H'0100 × m	32
	General PWM Timer Count Direction and Duty Setting Register	GTUDDTYC	H'0000_0001	H'30 + H'0100 × m	32
	General PWM Timer I/O Control Register	GTIOR	H'0000_0000	H'34 + H'0100 × m	32
	General PWM Timer Interrupt Output Setting Register	GTINTAD	H'000_00000	H'38 + H'0100 × m	32
	General PWM Timer Status Register	GTST	H'0000_8000	H'3C + H'0100 × m	32
	General PWM Timer Buffer Enable Register	GTBER	H'0000_0000	H'40 + H'0100 × m	32
	General PWM Timer Interrupt and A/D Converter Start Request Skipping Setting Register	GTITC	H'0000_0000	H'44 + H'0100 × m	32
	General PWM Timer Counter	GTCNT	H'0000_0000	H'48 + H'0100 × m	32
	General PWM Timer Compare Capture Register A	GTCCRA	H'FFFF_FFFF	H'4C + H'0100 × m	32
	General PWM Timer Compare Capture Register B	GTCCRB	H'FFFF_FFFF	H'50 + H'0100 × m	32
	General PWM Timer Compare Capture Register C	GTCCRC	H'FFFF_FFFF	H'54 + H'0100 × m	32
	General PWM Timer Compare Capture Register E	GTCCRE	H'FFFF_FFFF	H'58 + H'0100 × m	32
	General PWM Timer Compare Capture Register D	GTCCRD	H'FFFF_FFFF	H'5C + H'0100 × m	32
	General PWM Timer Compare Capture Register F	GTCCRF	H'FFFF_FFFF	H'60 + H'0100 × m	32
	General PWM Timer Cycle Setting Register	GTPR	H'FFFF_FFFF	H'64 + H'0100 × m	32

Table 18.4 GPT registers (2/2)

Module symbol	Register name	Register symbol	Reset value	Offset Address (m = 0 to 7)	Access size
GPT32Em (m = 0 to 7)	General PWM Timer Cycle Setting Buffer Register	GTPBR	H'FFFF_FFFF	H'68 + H'0100 × m	32
	General PWM Timer Cycle Setting Double-Buffer Register	GTPDBR	H'FFFF_FFFF	H'6C + H'0100 × m	32
	A/D Converter Start Request Timing Register A	GTADTRA	H'FFFF_FFFF	H'70 + H'0100 × m	32
	A/D Converter Start Request Timing Buffer Register A	GTADTBRA	H'FFFF_FFFF	H'74 + H'0100 × m	32
	A/D Converter Start Request Timing Double-Buffer Register A	GTADTDBRA	H'FFFF_FFFF	H'78 + H'0100 × m	32
	A/D Converter Start Request Timing Register B	GTADTRB	H'FFFF_FFFF	H'7C + H'0100 × m	32
	A/D Converter Start Request Timing Buffer Register B	GTADTBRB	H'FFFF_FFFF	H'80 + H'0100 × m	32
	A/D Converter Start Request Timing Double-Buffer Register B	GTADTDBRB	H'FFFF_FFFF	H'84 + H'0100 × m	32
	General PWM Timer Dead Time Control Register	GTDTCR	H'0000_0000	H'88 + H'0100 × m	32
	General PWM Timer Dead Time Value Register U	GTDVU	H'FFFF_FFFF	H'8C + H'0100 × m	32
	General PWM Timer Dead Time Value Register D	GTDVD	H'FFFF_FFFF	H'90 + H'0100 × m	32
	General PWM Timer Dead Time Buffer Register U	GTDBU	H'FFFF_FFFF	H'94 + H'0100 × m	32
	General PWM Timer Dead Time Buffer Register D	GTDBD	H'FFFF_FFFF	H'98 + H'0100 × m	32
	General PWM Timer Output Protection Function Status Register	GTSOS	H'0000_0000	H'9C + H'0100 × m	32
	General PWM Timer Output Protection Function Temporary Release Register	GTSOTR	H'0000_0000	H'A0 + H'0100 × m	32

18.2.1 General PWM Timer Write-Protection Register (GTWP)

Address(es): GPT32Em.GTWP H'1004_8000 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRKEY[7:0]								—	—	—	—	—	—	—	WP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	WP	0	R/W	Register Write Disable 0: Enable writes to the register 1: Disable writes to the register.
b7 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b15 to b8	PRKEY[7:0]	All 0	R/W	GTWP Key Code When H'A5 is written to these bits, the writes to the WP bit are permitted. These bits are read as 0.
b31 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

To prevent accidental changes, the GTWP enables or disables writing to following registers:

- GTSSR
- GTPSR
- GTCSR
- GTUPSR
- GTDNSR
- GTICASR
- GTICBSR
- GTCR
- GTUDDTYC
- GTIOR
- GTINTAD
- GTST
- GTBER
- GTITC
- GTCNT
- GTCCRA
- GTCCRB
- GTCCRC
- GTCCRD
- GTCCRE
- GTCCRF
- GTPR
- GTPBR
- GTPDBR
- GTADTRA
- GTADTBRA
- GTADTDBRA
- GTADTRB
- GTADTBRB
- GTADTDBRB
- GTDTCR
- GTDVU
- GTDVD
- GTDBU
- GTDBD
- GTSOS
- GTSOTR.

18.2.2 General PWM Timer Software Start Register (GTSTR)

Address(es): GPT32Em.GTSTR H'1004_8004 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CSTRT 7	CSTRT 6	CSTRT 5	CSTRT 4	CSTRT 3	CSTRT 2	CSTRT 1	CSTRT 0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The GTSTR starts the GTCNT counter operation for each channel x, where x = 0 to 7.

The GTSTR bit number represents the channel number. The GTSTR register is shared by all of the channels. The GTCNT counter starts for the channel associated with the GTSTR bit number where 1 is written. Writing 0 has no effect on the status of the GTCNT counter and the value of GTSTR register.

CSTRTx bit (channel x GTCNT Count Start) (x = 0 to 7)

The CSTRTx bit starts channel x of the GTCNT counter operation. Writing to GTSTR.CSTRTx bit has no effect unless GPTx.GTSSR.CSTRT bit is set to 1.

Read data shows the counter status of each channel (GTCR.CST bit). Zero means the counter stops and 1 means the counter is running.

18.2.3 General PWM Timer Software Stop Register (GTSTP)

Address(es): GPT32Em.GTSTP H'1004_8008 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CSTOP 7	CSTOP 6	CSTOP 5	CSTOP 4	CSTOP 3	CSTOP 2	CSTOP 1	CSTOP 0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The GTSTP stops the GTCNT counter operation for each channel x, where x = 0 to 7.

The GTSTP bit number represents the channel number. The GTSTP register is shared by all of the channels. The GTCNT counter stops for the channel associated with the GTSTP bit number where 1 is written. Writing 0 has no effect on the status of GTCNT counter and the value of GTSTP register.

CSTOPx bit (channel x GTCNT Count Stop) (x = 0 to 7)

The CSTOPx bit stops channel x of the GTCNT counter operation. Writing to GTSTP.CSTOPx bit has no effect unless GPTx.GTPSR.CSTOP bit is set to 1. Read data shows the counter status of each channel (invert of GTCR.CST bit). Zero means the counter is running and 1 means the counter stops.

18.2.4 General PWM Timer Software Clear Register (GTCLR)

Address(es): GPT32Em.GTCLR H'1004_800C + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CCLR7	CCLR6	CCLR5	CCLR4	CCLR3	CCLR2	CCLR1	CCLR0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTCLR is a write-only register that clears the GTCNT counter operation for each channel x, where x = 0 to 7. The GTCLR bit number represents the channel number. The GTCLR register is shared by all of the channels. The

GTCNT counter is cleared for the channel associated with the GTCLR bit number where 1 is written. Writing 0 has no effect on the status of the GTCNT counter.

CCLRx bit (channel x GTCNT Count Clear) (x = 0 to 7).

Channel x of the GTCNT counter value is cleared on writing 1 to the CCLRx bit. This bit is read as 0.

18.2.5 General PWM Timer Start Source Select Register (GTSSR)

Address(es): GPT32Em.GTSSR H'1004_8010 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CSTRT	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SSCBFAH	SSCBFAL	SSCBRAH	SSCBRAL	SSCAFBH	SSCAFBL	SSCARBH	SSCARBL	SSGTRGDF	SSGTRGDR	SSGTRGCF	SSGTRGCR	SSGTRGBF	SSGTRGBR	SSGTRGAF	SSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	SSGTRGAR	0	R/W	GTETRGA Pin Rising Input Source Counter Start Enable 0: Disable counter start on the rising edge of GTETRGA input 1: Enable counter start on the rising edge of GTETRGA input
b1	SSGTRGAF	0	R/W	GTETRGA Pin Falling Input Source Counter Start Enable 0: Disable counter start on the falling edge of GTETRGA input 1: Enable counter start on the falling edge of GTETRGA input
b2	SSGTRGBR	0	R/W	GTETRGB Pin Rising Input Source Counter Start Enable 0: Disable counter start on the rising edge of GTETRGB input 1: Enable counter start on the rising edge of GTETRGB input
b3	SSGTRGBF	0	R/W	GTETRGB Pin Falling Input Source Counter Start Enable 0: Disable counter start on the falling edge of GTETRGB input 1: Enable counter start on the falling edge of GTETRGB input
b4	SSGTRGCR	0	R/W	GTETRGC Pin Rising Input Source Counter Start Enable 0: Disable counter start on the rising edge of GTETRGC input 1: Enable counter start on the rising edge of GTETRGC input
b5	SSGTRGCF	0	R/W	GTETRGC Pin Falling Input Source Counter Start Enable 0: Disable counter start on the falling edge of GTETRGC input 1: Enable counter start on the falling edge of GTETRGC input
b6	SSGTRGDR	0	R/W	GTETRGD Pin Rising Input Source Counter Start Enable 0: Disable counter start on the rising edge of GTETRGD input 1: Enable counter start on the rising edge of GTETRGD input
b7	SSGTRGDF	0	R/W	GTETRGD Pin Falling Input Source Counter Start Enable 0: Disable counter start on the falling edge of GTETRGD input 1: Enable counter start on the falling edge of GTETRGD input
b8	SSCARBL	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Start Enable 0: Disable counter start on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter start on the rising edge of GTIOCA input when GTIOCB input is 0
b9	SSCARBH	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Start Enable 0: Disable counter start on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter start on the rising edge of GTIOCA input when GTIOCB input is 1

Bit	Bit Name	Initial Value	R/W	Description
b10	SSCAFBL	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Start Enable 0: Disable counter start on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter start on the falling edge of GTIOCA input when GTIOCB input is 0
b11	SSCAFBH	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Start Enable 0: Disable counter start on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter start on the falling edge of GTIOCA input when GTIOCB input is 1
b12	SSCBRAL	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Start Enable 0: Disable counter start on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter start on the rising edge of GTIOCB input when GTIOCA input is 0
b13	SSCBRAH	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Start Enable 0: Disable counter start on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter start on the rising edge of GTIOCB input when GTIOCA input is 1
b14	SSCBFAL	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Start Enable 0: Disable counter start on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter start on the falling edge of GTIOCB input when GTIOCA input is 0
b15	SSCBFAH	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Start Enable 0: Disable counter start on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter start on the falling edge of GTIOCB input when GTIOCA input is 1
b30 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b31	CSTRT	0	R/W	Software Source Counter Start Enable 0: Disable counter start by the GTSTR register 1: Enable counter start by the GTSTR register

GTSSR sets the source to start the GTCNT counter.

SSGTRGAR bit (GTETRGA Pin Rising Input Source Counter Start Enable)

The SSGTRGAR bit enables or disables GTCNT counter start on the rising edge of GTETRGA pin input.

SSGTRGAF bit (GTETRGA Pin Falling Input Source Counter Start Enable)

The SSGTRGAF bit enables or disables GTCNT counter start on the falling edge of GTETRGA pin input.

SSGTRGBR bit (GTETRGB Pin Rising Input Source Counter Start Enable)

The SSGTRGBR bit enables or disables GTCNT counter start on the rising edge of GTETRGB pin input.

SSGTRGBF bit (GTETRGB Pin Falling Input Source Counter Start Enable)

The SSGTRGBF bit enables or disables GTCNT counter start on the falling edge of GTETRGB pin input.

SSGTRGCR bit (GTETRGC Pin Rising Input Source Counter Start Enable)

The SSGTRGCR bit enables or disables GTCNT counter start on the rising edge of GTETRGC pin input.

SSGTRGCF bit (GTETRGC Pin Falling Input Source Counter Start Enable)

The SSGTRGCF bit enables or disables GTCNT counter start on the falling edge of GTETRGC pin input.

SSGTRGDR bit (GTETRGD Pin Rising Input Source Counter Start Enable)

The SSGTRGDR bit enables or disables GTCNT counter start on the rising edge of GTETRGD pin input.

SSGTRGDF bit (GTETRGD Pin Falling Input Source Counter Start Enable)

The SSGTRGDF bit enables or disables GTCNT counter start on the falling edge of GTETRGD pin input.

SSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Start Enable)

The SSCARBL bit enables or disables GTCNT counter start on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

SSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Start Enable)

The SSCARBH bit enables or disables GTCNT counter start on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

SSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Start Enable)

The SSCAFBL bit enables or disables GTCNT counter start on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

SSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Start Enable)

The SSCAFBH bit enables or disables GTCNT counter start on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

SSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Start Enable)

The SSCBRAL bit enables or disables GTCNT counter start on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

SSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Start Enable)

The SSCBRAH bit enables or disables GTCNT counter start on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

SSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Start Enable)

The SSCBFAL bit enables or disables GTCNT counter start on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

SSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Start Enable)

The SSCBFAH bit enables or disables GTCNT counter start on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

CSTRT bit (Software Source Counter Start Enable)

The CSTRT bit enables or disables GTCNT counter start by GTSTR register.

18.2.6 General PWM Timer Stop Source Select Register (GTPSR)

Address(es): GPT32Em.GTPSR H'1004_8014 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CSTOP	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PSCBFAH	PSCBFAL	PSCBRAH	PSCBRAL	PSCAFBH	PSCAFBL	PSCARBH	PSCARBL	PSGTRGDF	PSGTRGDR	PSGTRGCF	PSGTRGCR	PSGTRGBF	PSGTRGBR	PSGTRGAF	PSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	PSGTRGAR	0	R/W	GTETRGA Pin Rising Input Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTETRGA input 1: Enable counter stop on the rising edge of GTETRGA input
b1	PSGTRGAF	0	R/W	GTETRGA Pin Falling Input Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTETRGA input 1: Enable counter stop on the falling edge of GTETRGA input
b2	PSGTRGBR	0	R/W	GTETRGB Pin Rising Input Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTETRGB input 1: Enable counter stop on the rising edge of GTETRGB input
b3	PSGTRGBF	0	R/W	GTETRGB Pin Falling Input Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTETRGB input 1: Enable counter stop on the falling edge of GTETRGB input
b4	PSGTRGCR	0	R/W	GTETRGC Pin Rising Input Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTETRGC input 1: Enable counter stop on the rising edge of GTETRGC input
b5	PSGTRGCF	0	R/W	GTETRGC Pin Falling Input Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTETRGC input 1: Enable counter stop on the falling edge of GTETRGC input
b6	PSGTRGDR	0	R/W	GTETRGD Pin Rising Input Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTETRGD input 1: Enable counter stop on the rising edge of GTETRGD input
b7	PSGTRGDF	0	R/W	GTETRGD Pin Falling Input Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTETRGD input 1: Enable counter stop on the falling edge of GTETRGD input
b8	PSCARBL	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter stop on the rising edge of GTIOCA input when GTIOCB input is 0
b9	PSCARBH	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter stop on the rising edge of GTIOCA input when GTIOCB input is 1

Bit	Bit Name	Initial Value	R/W	Description
b10	PSCAFBL	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter stop on the falling edge of GTIOCA input when GTIOCB input is 0
b11	PSCAFBH	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter stop on the falling edge of GTIOCA input when GTIOCB input is 1
b12	PSCBRAL	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter stop on the rising edge of GTIOCB input when GTIOCA input is 0
b13	PSCBRAH	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter stop on the rising edge of GTIOCB input when GTIOCA input is 1
b14	PSCBFAL	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter stop on the falling edge of GTIOCB input when GTIOCA input is 0
b15	PSCBFAH	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter stop on the falling edge of GTIOCB input when GTIOCA input is 1
b30 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b31	CSTOP	0	R/W	Software Source Counter Stop Enable 0: Disable counter stop by the GTSTP register 1: Enable counter stop by the GTSTP register

GTPSR sets the source to stop the GTCNT counter.

PSGTRGAR bit (GTETRGA Pin Rising Input Source Counter Stop Enable)

The PSGTRGAR bit enables or disables GTCNT counter stop on the rising edge of GTETRGA pin input.

PSGTRGAF bit (GTETRGA Pin Falling Input Source Counter Stop Enable)

The PSGTRGAF bit enables or disables GTCNT counter stop on the falling edge of GTETRGA pin input.

PSGTRGBR bit (GTETRGB Pin Rising Input Source Counter Stop Enable)

The PSGTRGBR bit enables or disables GTCNT counter stop on the rising edge of GTETRGB pin input.

PSGTRGBF bit (GTETRGB Pin Falling Input Source Counter Stop Enable)

The PSGTRGBF bit enables or disables GTCNT counter stop on the falling edge of GTETRGB pin input.

PSGTRGCR bit (GTETRGC Pin Rising Input Source Counter Stop Enable)

The PSGTRGCR bit enables or disables GTCNT counter stop on the rising edge of GTETRGC pin input.

PSGTRGCF bit (GTETRGC Pin Falling Input Source Counter Stop Enable)

The PSGTRGCF bit enables or disables GTCNT counter stop on the falling edge of GTETRGC pin input.

PSGTRGDR bit (GTETRGD Pin Rising Input Source Counter Stop Enable)

The PSGTRGDR bit enables or disables GTCNT counter stop on the rising edge of GTETRGD pin input.

PSGTRGDF bit (GTETRGD Pin Falling Input Source Counter Stop Enable)

The PSGTRGDF bit enables or disables GTCNT counter stop on the falling edge of GTETRGD pin input.

PSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Stop Enable)

The PSCARBL bit enables or disables GTCNT counter stop on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

PSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Stop Enable)

The PSCARBH bit enables or disables GTCNT counter stop on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

PSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Stop Enable)

The PSCAFBL bit enables or disables GTCNT counter stop on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

PSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Stop Enable)

The PSCAFBH bit enables or disables GTCNT counter stop on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

PSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Stop Enable)

The PSCBRAL bit enables or disables GTCNT counter stop on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

PSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Stop Enable)

The PSCBRAH bit enables or disables GTCNT counter stop on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

PSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Stop Enable)

The PSCBFAL bit enables or disables GTCNT counter stop on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

PSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Stop Enable)

The PSCBFAH bit enables or disables GTCNT counter stop on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

CSTOP bit (Software Source Counter Stop Enable)

The CSTOP bit enables or disables GTCNT counter stop by GTSTP register.

18.2.7 General PWM Timer Clear Source Select Register (GTCSR)

Address(es): GPT32Em.GTCSR H'1004_8018 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CCLR	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CSCBFAH	CSCBFAL	CSCBRAH	CSCBRAL	CSCAFBH	CSCAFBL	CSCARBH	CSCARBL	CSGTRGDF	CSGTRGDR	CSGTRGCF	CSGTRGCR	CSGTRGBF	CSGTRGBR	CSGTRGAF	CSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	CSGTRGAR	0	R/W	GTETRGA Pin Rising Input Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTETRGA input 1: Enable counter clear on the rising edge of GTETRGA input
b1	CSGTRGAF	0	R/W	GTETRGA Pin Falling Input Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTETRGA input 1: Enable counter clear on the falling edge of GTETRGA input
b2	CSGTRGBR	0	R/W	GTETRGB Pin Rising Input Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTETRGB input 1: Enable counter clear on the rising edge of GTETRGB input
b3	CSGTRGBF	0	R/W	GTETRGB Pin Falling Input Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTETRGB input 1: Enable counter clear on the falling edge of GTETRGB input
b4	CSGTRGCR	0	R/W	GTETRGC Pin Rising Input Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTETRGC input 1: Enable counter clear on the rising edge of GTETRGC input
b5	CSGTRGCF	0	R/W	GTETRGC Pin Falling Input Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTETRGC input 1: Enable counter clear on the falling edge of GTETRGC input
b6	CSGTRGDR	0	R/W	GTETRGD Pin Rising Input Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTETRGD input 1: Enable counter clear on the rising edge of GTETRGD input
b7	CSGTRGDF	0	R/W	GTETRGD Pin Falling Input Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTETRGD input 1: Enable counter clear on the falling edge of GTETRGD input
b8	CSCARBL	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter clear on the rising edge of GTIOCA input when GTIOCB input is 0
b9	CSCARBH	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter clear on the rising edge of GTIOCA input when GTIOCB input is 1

Bit	Bit Name	Initial Value	R/W	Description
b10	CSCAFBL	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter clear on the falling edge of GTIOCA input when GTIOCB input is 0
b11	CSCAFBH	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter clear on the falling edge of GTIOCA input when GTIOCB input is 1
b12	CSCBRAL	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter clear on the rising edge of GTIOCB input when GTIOCA input is 0
b13	CSCBRAH	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter clear on the rising edge of GTIOCB input when GTIOCA input is 1
b14	CSCBFAL	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter clear on the falling edge of GTIOCB input when GTIOCA input is 0
b15	CSCBFAH	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter clear on the falling edge of GTIOCB input when GTIOCA input is 1
b30 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b31	CCLR	0	R/W	Software Source Counter Clear Enable 0: Disable counter clear by the GTCLR register 1: Enable counter clear by the GTCLR register

GTCSR sets the source to clear the GTCNT counter.

CSGTRGAR bit (GTETRGA Pin Rising Input Source Counter Clear Enable)

The CSGTRGAR bit enables or disables GTCNT counter clear on the rising edge of GTETRGA pin input.

CSGTRGAF bit (GTETRGA Pin Falling Input Source Counter Clear Enable)

The CSGTRGAF bit enables or disables GTCNT counter clear on the falling edge of GTETRGA pin input.

CSGTRGBR bit (GTETRGB Pin Rising Input Source Counter Clear Enable)

The CSGTRGBR bit enables or disables GTCNT counter clear on the rising edge of GTETRGB pin input.

CSGTRGBF bit (GTETRGB Pin Falling Input Source Counter Clear Enable)

The CSGTRGBF bit enables or disables GTCNT counter clear on the falling edge of GTETRGB pin input.

CSGTRGCR bit (GTETRGC Pin Rising Input Source Counter Clear Enable)

The CSGTRGCR bit enables or disables GTCNT counter clear on the rising edge of GTETRGC pin input.

CSGTRGCF bit (GTETRGC Pin Falling Input Source Counter Clear Enable)

The CSGTRGCF bit enables or disables GTCNT counter clear on the falling edge of GTETRGC pin input.

CSGTRGDR bit (GTETRGD Pin Rising Input Source Counter Clear Enable)

The CSGTRGDR bit enables or disables GTCNT counter clear on the rising edge of GTETRGD pin input.

CSGTRGDF bit (GTETRGD Pin Falling Input Source Counter Clear Enable)

The CSGTRGDF bit enables or disables GTCNT counter clear on the falling edge of GTETRGD pin input.

CSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Clear Enable)

The CSCARBL bit enables or disables GTCNT counter clear on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

CSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Clear Enable)

The CSCARBH bit enables or disables GTCNT counter clear on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

CSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Clear Enable)

The CSCAFBL bit enables or disables GTCNT counter clear on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

CSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Clear Enable)

The CSCAFBH bit enables or disables GTCNT counter clear on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

CSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Clear Enable)

The CSCBRAL bit enables or disables GTCNT counter clear on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

CSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Clear Enable)

The CSCBRAH bit enables or disables GTCNT counter clear on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

CSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Clear Enable)

The CSCBFAL bit enables or disables GTCNT counter clear on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

CSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Clear Enable)

The CSCBFAH bit enables or disables GTCNT counter clear on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

CCLR bit (Software Source Counter Clear Enable)

The CCLR bit enables or disables GTCNT counter clear by GTCLR register.

18.2.8 General PWM Timer Up Count Source Select Register (GTUPSR)

Address(es): GPT32Em.GTUPSR H'1004_801C + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	USCBFAH	USCBFAL	USCBRAH	USCBRAL	USCAFBH	USCAFBL	USCARBH	USCARBL	USGTRGDF	USGTRGDR	USGTRGCF	USGTRGCR	USGTRGBF	USGTRGBR	USGTRGAF	USGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	USGTRGAR	0	R/W	GTETRGA Pin Rising Input Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTETRGA input 1: Enable counter count up on the rising edge of GTETRGA input.
b1	USGTRGAF	0	R/W	GTETRGA Pin Falling Input Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTETRGA input 1: Enable counter count up on the falling edge of GTETRGA input.
b2	USGTRGBR	0	R/W	GTETRGB Pin Rising Input Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTETRGB input 1: Enable counter count up on the rising edge of GTETRGB input.
b3	USGTRGBF	0	R/W	GTETRGB Pin Falling Input Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTETRGB input 1: Enable counter count up on the falling edge of GTETRGB input.
b4	USGTRGCR	0	R/W	GTETRGC Pin Rising Input Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTETRGC input 1: Enable counter count up on the rising edge of GTETRGC input.
b5	USGTRGCF	0	R/W	GTETRGC Pin Falling Input Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTETRGC input 1: Enable counter count up on the falling edge of GTETRGC input.
b6	USGTRGDR	0	R/W	GTETRGD Pin Rising Input Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTETRGD input 1: Enable counter count up on the rising edge of GTETRGD input.
b7	USGTRGDF	0	R/W	GTETRGD Pin Falling Input Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTETRGD input 1: Enable counter count up on the falling edge of GTETRGD input.
b8	USCARBL	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter count up on the rising edge of GTIOCA input when GTIOCB input is 0.
b9	USCARBH	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter count up on the rising edge of GTIOCA input when GTIOCB input is 1.

Bit	Bit Name	Initial Value	R/W	Description
b10	USCAFBL	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter count up on the falling edge of GTIOCA input when GTIOCB input is 0.
b11	USCAFBH	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter count up on the falling edge of GTIOCA input when GTIOCB input is 1.
b12	USCBRAL	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter count up on the rising edge of GTIOCB input when GTIOCA input is 0.
b13	USCBRAH	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter count up on the rising edge of GTIOCB input when GTIOCA input is 1.
b14	USCBFAL	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter count up on the falling edge of GTIOCB input when GTIOCA input is 0.
b15	USCBFAH	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter count up on the falling edge of GTIOCB input when GTIOCA input is 1.
b31 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTUPSR sets the source to count up the GTCNT counter.

When at least 1 bit in the GTUPSR register is set to 1, the GTCNT counter is counted up by the source that is set to 1 in this register. In such cases, the setting of GTCR.TPCS has no effect.

USGTRGAR bit (GTETRGA Pin Rising Input Source Counter Count Up Enable)

The USGTRGAR bit enables or disables GTCNT counter count up on the rising edge of GTETRGA pin input.

USGTRGAF bit (GTETRGA Pin Falling Input Source Counter Count Up Enable)

The USGTRGAF bit enables or disables GTCNT counter count up on the falling edge of GTETRGA pin input.

USGTRGBR bit (GTETRGB Pin Rising Input Source Counter Count Up Enable)

The USGTRGBR bit enables or disables GTCNT counter count up on the rising edge of GTETRGB pin input.

USGTRGBF bit (GTETRGB Pin Falling Input Source Counter Count Up Enable)

The USGTRGBF bit enables or disables GTCNT counter count up on the falling edge of GTETRGB pin input.

USGTRGCR bit (GTETRGC Pin Rising Input Source Counter Count Up Enable)

The USGTRGCR bit enables or disables GTCNT counter count up on the rising edge of GTETRGC pin input.

USGTRGCF bit (GTETRGC Pin Falling Input Source Counter Count Up Enable)

The USGTRGCF bit enables or disables GTCNT counter count up on the falling edge of GTETRGC pin input.

USGTRGDR bit (GTETRGD Pin Rising Input Source Counter Count Up Enable)

The USGTRGDR bit enables or disables GTCNT counter count up on the rising edge of GTETRGD pin input.

USGTRGDF bit (GTETRGD Pin Falling Input Source Counter Count Up Enable)

The USGTRGDF bit enables or disables GTCNT counter count up on the falling edge of GTETRGD pin input.

USCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Count Up Enable)

The USCARBL bit enables or disables GTCNT counter count up on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

USCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Count Up Enable)

The USCARBH bit enables or disables GTCNT counter count up on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

USCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Count Up Enable)

The USCAFBL bit enables or disables GTCNT counter count up on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

USCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Count Up Enable)

The USCAFBH bit enables or disables GTCNT counter count up on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

USCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Count Up Enable)

The USCBRAL bit enables or disables GTCNT counter count up on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

USCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Count Up Enable)

The USCBRAH bit enables or disables GTCNT counter count up on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

USCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Count Up Enable)

The USCBFAL bit enables or disables GTCNT counter count up on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

USCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Count Up Enable)

The USCBFAH bit enables or disables GTCNT counter count up on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

18.2.9 General PWM Timer Down Count Source Select Register (GTDNSR)

Address(es): GPT32Em.GTDNSR H'1004_8020 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DSCBFAH	DSCBFAL	DSCBRAH	DSCBRAL	DSCAFBH	DSCAFBL	DSCARBH	DSCARBL	DSGTRGDF	DSGTRGDR	DSGTRGCF	DSGTRGCR	DSGTRGBF	DSGTRGBR	DSGTRGAF	DSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	DSGTRGAR	0	R/W	GTETRGA Pin Rising Input Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTETRGA input 1: Enable counter count down on the rising edge of GTETRGA input.
b1	DSGTRGAF	0	R/W	GTETRGA Pin Falling Input Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTETRGA input 1: Enable counter count down on the falling edge of GTETRGA input.
b2	DSGTRGBR	0	R/W	GTETRGB Pin Rising Input Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTETRGB input 1: Enable counter count down on the rising edge of GTETRGB input.
b3	DSGTRGBF	0	R/W	GTETRGB Pin Falling Input Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTETRGB input 1: Enable counter count down on the falling edge of GTETRGB input.
b4	DSGTRGCR	0	R/W	GTETRGC Pin Rising Input Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTETRGC input 1: Enable counter count down on the rising edge of GTETRGC input.
b5	DSGTRGCF	0	R/W	GTETRGC Pin Falling Input Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTETRGC input 1: Enable counter count down on the falling edge of GTETRGC input.
b6	DSGTRGDR	0	R/W	GTETRGD Pin Rising Input Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTETRGD input 1: Enable counter count down on the rising edge of GTETRGD input.
b7	DSGTRGDF	0	R/W	GTETRGD Pin Falling Input Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTETRGD input 1: Enable counter count down on the falling edge of GTETRGD input.
b8	DSCARBL	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter count down on the rising edge of GTIOCA input when GTIOCB input is 0.
b9	DSCARBH	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter count down on the rising edge of GTIOCA input when GTIOCB input is 1.

Bit	Bit Name	Initial Value	R/W	Description
b10	DSCAFBL	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter count down on the falling edge of GTIOCA input when GTIOCB input is 0.
b11	DSCAFBH	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter count down on the falling edge of GTIOCA input when GTIOCB input is 1.
b12	DSCBRAL	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter count down on the rising edge of GTIOCB input when GTIOCA input is 0.
b13	DSCBRAH	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter count down on the rising edge of GTIOCB input when GTIOCA input is 1.
b14	DSCBFAL	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter count down on the falling edge of GTIOCB input when GTIOCA input is 0.
b15	DSCBFAH	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter count down on the falling edge of GTIOCB input when GTIOCA input is 1.
b31 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTDNSR sets the source to count down the GTCNT counter.

When at least 1 bit in the GTDNSR register is set to 1, the GTCNT counter is counted down by the source that is set to 1 in this register. In such cases, the setting of GTCR.TPCS has no effect.

DSGTRGAR bit (GTETRGA Pin Rising Input Source Counter Count Down Enable)

The DSGTRGAR bit enables or disables GTCNT counter count down on the rising edge of GTETRGA pin input.

DSGTRGAF bit (GTETRGA Pin Falling Input Source Counter Count Down Enable)

The DSGTRGAF bit enables or disables GTCNT counter count down on the falling edge of GTETRGA pin input.

DSGTRGBR bit (GTETRGB Pin Rising Input Source Counter Count Down Enable)

The DSGTRGBR bit enables or disables GTCNT counter count down on the rising edge of GTETRGB pin input.

DSGTRGBF bit (GTETRGB Pin Falling Input Source Counter Count Down Enable)

The DSGTRGBF bit enables or disables GTCNT counter count down on the falling edge of GTETRGB pin input.

DSGTRGCR bit (GTETRGC Pin Rising Input Source Counter Count Down Enable)

The DSGTRGCR bit enables or disables GTCNT counter count down on the rising edge of GTETRGC pin input.

DSGTRGCF bit (GTETRGC Pin Falling Input Source Counter Count Down Enable)

The DSGTRGCF bit enables or disables GTCNT counter count down on the falling edge of GTETRGC pin input.

DSGTRGDR bit (GTETRGD Pin Rising Input Source Counter Count Down Enable)

The DSGTRGDR bit enables or disables GTCNT counter count down on the rising edge of GTETRGD pin input.

DSGTRGDF bit (GTETRGD Pin Falling Input Source Counter Count Down Enable)

The DSGTRGDF bit enables or disables GTCNT counter count down on the falling edge of GTETRGD pin input.

DSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Count Down Enable)

The DSCARBL bit enables or disables GTCNT counter count down on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

DSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Count Down Enable)

The DSCARBH bit enables or disables GTCNT counter count down on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

DSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Count Down Enable)

The DSCAFBL bit enables or disables GTCNT counter count down on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

DSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Count Down Enable)

The DSCAFBH bit enables or disables GTCNT counter count down on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

DSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Count Down Enable)

The DSCBRAL bit enables or disables GTCNT counter count down on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

DSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Count Down Enable)

The DSCBRAH bit enables or disables GTCNT counter count down on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

DSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Count Down Enable)

The DSCBFAL bit enables or disables GTCNT counter count down on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

DSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Count Down Enable)

The DSCBFAH bit enables or disables GTCNT counter count down on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

18.2.10 General PWM Timer Input Capture Source Select Register A (GTICASR)

Address(es): GPT32Em.GTICASR H'1004_8024 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ASCBFAH	ASCBFAL	ASCBRAH	ASCBRAL	ASCAFBH	ASCAFBL	ASCARBH	ASCARBL	ASGTRGDF	ASGTRGDR	ASGTRGCF	ASGTRGCR	ASGTRGBF	ASGTRGBR	ASGTRGAF	ASGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	ASGTRGAR	0	R/W	GTETRGA Pin Rising Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTETRGA input 1: Enable GTCCRA input capture on the rising edge of GTETRGA input.
b1	ASGTRGAF	0	R/W	GTETRGA Pin Falling Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTETRGA input 1: Enable GTCCRA input capture on the falling edge of GTETRGA input.
b2	ASGTRGBR	0	R/W	GTETRGB Pin Rising Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTETRGB input 1: Enable GTCCRA input capture on the rising edge of GTETRGB input.
b3	ASGTRGBF	0	R/W	GTETRGB Pin Falling Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTETRGB input 1: Enable GTCCRA input capture on the falling edge of GTETRGB input.
b4	ASGTRGCR	0	R/W	GTETRGC Pin Rising Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTETRGC input 1: Enable GTCCRA input capture on the rising edge of GTETRGC input.
b5	ASGTRGCF	0	R/W	GTETRGC Pin Falling Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTETRGC input 1: Enable GTCCRA input capture on the falling edge of GTETRGC input.
b6	ASGTRGDR	0	R/W	GTETRGD Pin Rising Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTETRGD input 1: Enable GTCCRA input capture on the rising edge of GTETRGD input.
b7	ASGTRGDF	0	R/W	GTETRGD Pin Falling Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTETRGD input 1: Enable GTCCRA input capture on the falling edge of GTETRGD input.
b8	ASCARBL	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value Low Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable GTCCRA input capture on the rising edge of GTIOCA input when GTIOCB input is 0.
b9	ASCARBH	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value High Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable GTCCRA input capture on the rising edge of GTIOCA input when GTIOCB input is 1.

Bit	Bit Name	Initial Value	R/W	Description
b10	ASCAFBL	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value Low Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable GTCCRA input capture on the falling edge of GTIOCA input when GTIOCB input is 0.
b11	ASCAFBH	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value High Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable GTCCRA input capture on the falling edge of GTIOCA input when GTIOCB input is 1.
b12	ASCBRAL	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value Low Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable GTCCRA input capture on the rising edge of GTIOCB input when GTIOCA input is 0.
b13	ASCBRAH	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value High Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable GTCCRA input capture on the rising edge of GTIOCB input when GTIOCA input is 1.
b14	ASCBFAL	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value Low Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable GTCCRA input capture on the falling edge of GTIOCB input when GTIOCA input is 0.
b15	ASCBFAH	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value High Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable GTCCRA input capture on the falling edge of GTIOCB input when GTIOCA input is 1.
b31 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTICASR sets the source of input capture for GTCCRA.

ASGTRGAR bit (GTETRGA Pin Rising Input Source GTCCRA Input Capture Enable)

The ASGTRGAR bit enables or disables input capture for GTCCRA on the rising edge of GTETRGA pin input.

ASGTRGAF bit (GTETRGA Pin Falling Input Source GTCCRA Input Capture Enable)

The ASGTRGAF bit enables or disables input capture for GTCCRA on the falling edge of GTETRGA pin input.

ASGTRGBR bit (GTETRGB Pin Rising Input Source GTCCRA Input Capture Enable)

The ASGTRGBR bit enables or disables input capture for GTCCRA on the rising edge of GTETRGB pin input.

ASGTRGBF bit (GTETRGB Pin Falling Input Source GTCCRA Input Capture Enable)

The ASGTRGBF bit enables or disables input capture for GTCCRA on the falling edge of GTETRGB pin input.

ASGTRGCR bit (GTETRGC Pin Rising Input Source GTCCRA Input Capture Enable)

The ASGTRGCR bit enables or disables input capture for GTCCRA on the rising edge of GTETRGC pin input.

ASGTRGCF bit (GTETRGC Pin Falling Input Source GTCCRA Input Capture Enable)

The ASGTRGCF bit enables or disables input capture for GTCCRA on the falling edge of GTETRGC pin input.

ASGTRGDR bit (GTETRGD Pin Rising Input Source GTCCRA Input Capture Enable)

The ASGTRGDR bit enables or disables input capture for GTCCRA on the rising edge of GTETRGD pin input.

ASGTRGDF bit (GTETRGD Pin Falling Input Source GTCCRA Input Capture Enable)

The ASGTRGDF bit enables or disables input capture for GTCCRA on the falling edge of GTETRGD pin input.

ASCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source GTCCRA Input Capture Enable)

The ASCARBL bit enables or disables input capture for GTCCRA on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

ASCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source GTCCRA Input Capture Enable)

The ASCARBH bit enables or disables input capture for GTCCRA on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

ASCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source GTCCRA Input Capture Enable)

The ASCAFBL bit enables or disables input capture for GTCCRA on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

ASCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source GTCCRA Input Capture Enable)

The ASCAFBH bit enables or disables input capture for GTCCRA on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

ASCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source GTCCRA Input Capture Enable)

The ASCBRAL bit enables or disables input capture for GTCCRA on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

ASCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source GTCCRA Input Capture Enable)

The ASCBRAH bit enables or disables input capture for GTCCRA on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

ASCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source GTCCRA Input Capture Enable)

The ASCBFAL bit enables or disables input capture for GTCCRA on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

ASCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source GTCCRA Input Capture Enable)

The ASCBFAH bit enables or disables input capture for GTCCRA on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

18.2.11 General PWM Timer Input Capture Source Select Register B (GTICBSR)

Address(es): GPT32Em.GTICBSR H'1004_8028 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BSCBFAH	BSCBFAL	BSCBRAH	BSCBRAL	BSCAFBH	BSCAFBL	BSCARBH	BSCARBL	BSGTRGDF	BSGTRGDR	BSGTRGCF	BSGTRGCR	BSGTRGBF	BSGTRGBR	BSGTRGAF	BSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	BSGTRGAR	0	R/W	GTETRGA Pin Rising Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTETRGA input 1: Enable GTCCRB input capture on the rising edge of GTETRGA input.
b1	BSGTRGAF	0	R/W	GTETRGA Pin Falling Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTETRGA input 1: Enable GTCCRB input capture on the falling edge of GTETRGA input.
b2	BSGTRGBR	0	R/W	GTETRGB Pin Rising Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTETRGB input 1: Enable GTCCRB input capture on the rising edge of GTETRGB input.
b3	BSGTRGBF	0	R/W	GTETRGB Pin Falling Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTETRGB input 1: Enable GTCCRB input capture on the falling edge of GTETRGB input.
b4	BSGTRGCR	0	R/W	GTETRGC Pin Rising Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTETRGC input 1: Enable GTCCRB input capture on the rising edge of GTETRGC input.
b5	BSGTRGCF	0	R/W	GTETRGC Pin Falling Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTETRGC input 1: Enable GTCCRB input capture on the falling edge of GTETRGC input.
b6	BSGTRGDR	0	R/W	GTETRGD Pin Rising Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTETRGD input 1: Enable GTCCRB input capture on the rising edge of GTETRGD input.
b7	BSGTRGDF	0	R/W	GTETRGD Pin Falling Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTETRGD input 1: Enable GTCCRB input capture on the falling edge of GTETRGD input.
b8	BSCARBL	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value Low Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable GTCCRB input capture on the rising edge of GTIOCA input when GTIOCB input is 0.
b9	BSCARBH	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value High Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable GTCCRB input capture on the rising edge of GTIOCA input when GTIOCB input is 1.

Bit	Bit Name	Initial Value	R/W	Description
b10	BSCAFBL	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value Low Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable GTCCRB input capture on the falling edge of GTIOCA input when GTIOCB input is 0.
b11	BSCAFBH	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value High Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable GTCCRB input capture on the falling edge of GTIOCA input when GTIOCB input is 1.
b12	BSCBRAL	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value Low Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable GTCCRB input capture on the rising edge of GTIOCB input when GTIOCA input is 0.
b13	BSCBRAH	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value High Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable GTCCRB input capture on the rising edge of GTIOCB input when GTIOCA input is 1.
b14	BSCBFAL	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value Low Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable GTCCRB input capture on the falling edge of GTIOCB input when GTIOCA input is 0.
b15	BSCBFAH	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value High Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable GTCCRB input capture on the falling edge of GTIOCB input when GTIOCA input is 1.
b31 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTICBSR sets the source of input capture for GTCCRB.

BSGTRGAR bit (GTETRGA Pin Rising Input Source GTCCRB Input Capture Enable)

The BSGTRGAR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGA pin input.

BSGTRGAF bit (GTETRGA Pin Falling Input Source GTCCRB Input Capture Enable)

The BSGTRGAF bit enables or disables input capture for GTCCRB on the falling edge of GTETRGA pin input.

BSGTRGBR bit (GTETRGB Pin Rising Input Source GTCCRB Input Capture Enable)

The BSGTRGBR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGB pin input.

BSGTRGBF bit (GTETRGB Pin Falling Input Source GTCCRB Input Capture Enable)

The BSGTRGBF bit enables or disables input capture for GTCCRB on the falling edge of GTETRGB pin input.

BSGTRGCR bit (GTETRGC Pin Rising Input Source GTCCRB Input Capture Enable)

The BSGTRGCR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGC pin input.

BSGTRGCF bit (GTETRGC Pin Falling Input Source GTCCRB Input Capture Enable)

The BSGTRGCF bit enables or disables input capture for GTCCRB on the falling edge of GTETRGC pin input.

BSGTRGDR bit (GTETRGD Pin Rising Input Source GTCCRB Input Capture Enable)

The BSGTRGDR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGD pin input.

BSGTRGDF bit (GTETRGD Pin Falling Input Source GTCCRB Input Capture Enable)

The BSGTRGDF bit enables or disables input capture for GTCCRB on the falling edge of GTETRGD pin input.

BSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source GTCCRB Input Capture Enable)

The BSCARBL bit enables or disables input capture for GTCCRB on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

BSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source GTCCRB Input Capture Enable)

The BSCARBH bit enables or disables input capture for GTCCRB on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

BSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source GTCCRB Input Capture Enable)

The BSCAFBL bit enables or disables input capture for GTCCRB on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

BSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source GTCCRB Input Capture Enable)

The BSCAFBH bit enables or disables input capture for GTCCRB on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

BSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source GTCCRB Input Capture Enable)

The BSCBRAL bit enables or disables input capture for GTCCRB on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

BSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source GTCCRB Input Capture Enable)

The BSCBRAH bit enables or disables input capture for GTCCRB on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

BSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source GTCCRB Input Capture Enable)

The BSCBFAL bit enables or disables input capture for GTCCRB on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

BSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source GTCCRB Input Capture Enable)

The BSCBFAH bit enables or disables input capture for GTCCRB on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

18.2.12 General PWM Timer Control Register (GTCR)

Address(es): GPT32Em.GTCR H'1004_802C + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	TPCS[2:0]			—	—	—	—	—	MD[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	CST	0	R/W	Count Start 0: Stop count operation 1: Perform count operation.
b15 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b18 to b16	MD[2:0]	All 0	R/W	Mode Select b18 b16 0 0 0: Saw-wave PWM mode (single buffer or double buffer possible) 0 0 1: Saw-wave one-shot pulse mode (fixed buffer operation) 0 1 0: Setting prohibited 0 1 1: Setting prohibited 1 0 0: Triangle-wave PWM mode 1 (32-bit transfer at trough) (single buffer or double buffer possible) 1 0 1: Triangle-wave PWM mode 2 (32-bit transfer at crest and trough) (single buffer or double buffer possible) 1 1 0: Triangle-wave PWM mode 3 (64-bit transfer at trough) fixed buffer operation) 1 1 1: Setting prohibited.
b23 to b19	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b26 to b24	TPCS[2:0]	All 0	R/W	Timer Prescaler Select b26 b24 0 0 0: P0ϕ 0 0 1: P0ϕ/4 0 1 0: P0ϕ/16 0 1 1: P0ϕ/64 1 0 0: P0ϕ/256 1 0 1: P0ϕ/1024.
b31 to b27	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTCR controls GTCNT.

CST bit (Count Start)

The CST bit controls the GTCNT counter start and stop.

[Setting conditions]

- GTSTR value where the channel number associated with the bit number is set to 1 with the GTSSR.CSTRT bit being 1.
- among input on the GTIOCA, GTIOCB, or GTETRGA to GTETRGD pins that is enabled in the GTSSR register as a source for the start of counting
- 1 is written by software directly.

[Clearing conditions]

- GTSTP value where the channel number associated with the bit number is set to 1 with the GTPSR.CSTOP bit being 1.
- among input on the GTIOCA, GTIOCB, or GTETRGA to GTETRGD pins that is enabled in the GTPSR register as a source for the stop of counting
- 0 is written by software directly.

MD[2:0] bits (Mode Select)

The MD[2:0] bits select the GPT operating mode. The MD[2:0] bits must be set while the GTCNT operation is stopped.

TPCS[2:0] bits (Timer Prescaler Select)

The TPCS[2:0] bits select the clock for GTCNT. A clock prescaler can be selected independently for each channel. The TPCS[2:0] bits must be set while the GTCNT operation is stopped.

18.2.13 General PWM Timer Count Direction and Duty Setting Register (GTUDDTYC)

Address(es): GPT32Em.GTUDDTYC H'1004_8030 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	OBDTYR	OBDTYF	OBDTY[1:0]		—	—	—	—	OADTYR	OADTYF	OADTY[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UDF	UD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	UD	1	R/W	Count Direction Setting 0: Count down on GTCNT 1: Counts up on GTCNT.
b1	UDF	0	R/W	Forcible Count Direction Setting 0: Do not force setting 1: Force setting.
b15 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b17, b16	OADTY[1:0]	All 0	R/W	GTIOCA Output Duty Setting b17 b16 0 x: GTIOCA pin duty depends on compare match 1 0: GTIOCA pin duty 0% 1 1: GTIOCA pin duty 100%.
b18	OADTYF	0	R/W	Forcible GTIOCA Output Duty Setting 0: Do not force setting 1: Force setting.
b19	OADTYR	0	R/W	GTIOCA Output Value Selecting after Releasing 0%/100% Duty Setting 0: Apply output value set in 0%/100% duty to GTIOA[3:2] function after releasing 0%/100% duty setting 1: Apply masked compare match output value to GTIOA[3:2] function after releasing 0%/100% duty setting.
b23 to b20	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b25, b24	OBDTY[1:0]	All 0	R/W	GTIOCB Output Duty Setting b25 b24 0 x: GTIOCB pin duty depends on compare match 1 0: GTIOCB pin duty = 0% 1 1: GTIOCB pin duty = 100%.
b26	OBDTYF	0	R/W	Forcible GTIOCB Output Duty Setting 0: Do not force setting 1: Force setting.
b27	OBDTYR	0	R/W	GTIOCB Output Value Selecting after Releasing 0%/100% Duty Setting 0: Apply output value set in 0%/100% duty to GTIOB[3:2] function after releasing 0%/100% duty setting 1: Apply masked compare match output value to GTIOB[3:2] function after releasing 0%/100% duty setting.

Bit	Bit Name	Initial Value	R/W	Description
b31 to b28	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Remarks: x: Don't care

GTUDDTYC sets the direction in which GTCNT counts (up-counting or down-counting) and sets the duty of GTIOCA/ GTIOCB pin output.

Count direction:

- In saw-wave mode

  When the UD value is set to 0 during up-counting, the count direction changes at an overflow (the timing synchronous with count clock after the GTCNT value becomes the GTPR value). When the UD value is set to 1 during down- counting, the count direction changes at an underflow (the timing synchronous with count clock after the GTCNT value becomes 0).

  When the UD value changes from 1 to 0 with the UDF bit being 0 and while counting is stopped, the counter starts up- counting and the count direction changes at an overflow (the timing synchronous with count clock after the GTCNT value becomes GTPR value). When the UD value changes from 0 to 1 with the UDF bit being 0 and while counting stops, the counter starts down-counting and the count direction changes at an underflow (the timing synchronous with count clock after the GTCNT value becomes 0).

  When the UDF bit is set to 1 while counting stops, the UD bit value is reflected in the count direction when counting starts.

- In triangle-wave mode

  When the UD value changes during counting, the count direction does not change. When the UD value changes while the UDF bit is 0 and counting stops, the change is not reflected in the count direction when counting starts.

  When the UDF bit is set to 1 while counting stops, the UD value is reflected in the count direction when counting starts.

UD bit (Count Direction Setting)

The UD bit sets the count direction for up-counting or down-counting, for GTCNT.

UDF bit (Forcible Count Direction Setting)

The UDF bit forcibly sets the count direction when GTCNT starts operation as the UD value. Only write 0 to this bit during counter operation. When 1 is written to this bit while counting stops, return this bit to 0 before counting starts.

Output duty:

- In saw-wave mode

  When the OADTY/OBDTY value changes during up-counting, the duty is reflected at an overflow (GTCNT = GTPR). When the OADTY/OBDTY value changes during down-counting, the duty is reflected at an underflow (GTCNT = 0). When the OADTY/OBDTY value changes to 1 with the OADTYF/OBDTYF bit being 0 and while counting is stopped, the output duty is not reflected at starting counter operation. When the count direction is up, the output duty is reflected at an overflow (GTCNT = GTPR). When the count direction is down, the output duty is reflected at an underflow (GTCNT = 0). When the OADTY/OBDTY value changes to 0 with the OADTYF/OBDTYF bit being 1 and while counting stops, the output duty is reflected at the starting counter operation.

- In triangle-wave mode

  When the OADTY/OBDTY value changes during counting, the duty is reflected at an underflow. When the OADTY/OBDTY value changes to 1 with the OADTYF/OBDTYF bit being 0 and while counting stops, the output duty is not reflected at the starting counter operation, however the output duty is reflected at an underflow.

  When the OADTY/OBDTY value changes to 0 with the OADTYF/OBDTYF bit being 1 and while counting stops, the output duty is reflected at starting counter operation.

OmDTY[1:0] bits (GTIOCm Output Duty Setting) (m = A, B)

The OmDTY[1:0] bits set the output duty (0%, 100% or compare match control) of the GTIOCm pin.

OmDTYF bit (Forcible GTIOCm Output Duty Setting) (m = A, B)

The OmDTYF bit forcibly sets the output duty cycle to the OmDTY setting. Set this bit to 0 during counter operation. When this bit is set to 1 while counting is stopped, return this bit to 0 until the first period ends after the counter starts.

OmDTYR bit (GTIOCm Output Value Selecting after Releasing 0%/100% Duty Setting) (m = A, B)

The OmDTYR bits select the value that is the object of output retained or toggled at cycle end, when the control changes from 0%/100% duty setting to compare match for GTIOCm pin and GTIOR.GTIOm[3:2] are set to 00b (output retained at cycle end) or GTIOR.GTIOm[3:2] are set to 11b (output toggled at cycle end).

GPT32 internally continues the compare match operation in performing 0%/100% duty operation. When the OmDTYR bit is set to 1, the GTCIOm pin outputs the value specified in the GTIOR.GTIOm[3:2] bits at the end of compare match cycle.

18.2.14 General PWM Timer I/O Control Register (GTIOR)

Address(es): GPT32Em.GTIOR H'1004_8034 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	NFCSB[1:0]		NFBEN	—	—	OBDF[1:0]		OBE	OBHLD	OBDFLT	—	GTIOB[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	NFCSA[1:0]		NFAEN	—	—	OADF[1:0]		OAE	OAHLD	OADFLT	—	GTIOA[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b4 to b0	GTIOA[4:0]	All 0	R/W	GTIOCA Pin Function Select See **Table 18.5**.
b5	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b6	OADFLT	0	R/W	GTIOCA Pin Output Value Setting at the Count Stop 0: Output low on GTIOCA pin when counting stops 1: Output high on GTIOCA pin when counting stops.
b7	OAHLD	0	R/W	GTIOCA Pin Output Setting at the Start/Stop Count 0: Set GTIOCA pin output level on counting start and stop based on the register setting. 1: Retain GTIOCA pin output level on counting start and stop.
b8	OAE	0	R/W	GTIOCA Pin Output Enable 0: Disable output 1: Enable output.
b10, b9	OADF[1:0]	All 0	R/W	GTIOCA Pin Disable Value Setting b10 b9 0 0: Prohibit output disable 0 1: Set GTIOCA pin to Hi-Z on output disable 1 0: Set GTIOCA pin to 0 on output disable 1 1: Set GTIOCA pin to 1 on output disable.
b12, b11	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b13	NFAEN	0	R/W	Noise Filter A Enable 0: Disable noise filter for GTIOCA pin 1: Enable noise filter for GTIOCA pin.
b15, b14	NFCSA[1:0]	All 0	R/W	Noise Filter A Sampling Clock Select b15 b14 0 0: P0ϕ 0 1: P0ϕ/4 1 0: P0ϕ/16 1 1: P0ϕ/64.
b20 to b16	GTIOB[4:0]	All 0	R/W	GTIOCB Pin Function Select See **Table 18.5**.
b21	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.

Bit	Bit Name	Initial Value	R/W	Description
b22	OBDFLT	0	R/W	GTIOCB Pin Output Value Setting at the Count Stop 0: Output low on GTIOCB pin when counting stops 1: Output high on GTIOCB pin when counting stops.
b23	OBHLD	0	R/W	GTIOCB Pin Output Setting at the Start/Stop Count 0: Set GTIOCB pin output level on counting start and stop based on the register setting 1: Retain GTIOCB pin output level on counting start and stop.
b24	OBE	0	R/W	GTIOCB Pin Output Enable 0: Disable output 1: Enable output.
b26, b25	OBDF[1:0]	All 0	R/W	GTIOCB Pin Disable Value Setting b26 b25 0 0: Prohibit output disable 0 1: Set GTIOCB pin to Hi-Z on output disable 1 0: Set GTIOCB pin to 0 on output disable 1 1: Set GTIOCB pin to 1 on output disable.
b28, b27	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b29	NFBEN	0	R/W	Noise Filter B Enable 0: Disable noise filter for GTIOCB pin 1: Enable noise filter for GTIOCB pin.
b31, b30	NFCSB[1:0]	All 0	R/W	Noise Filter B Sampling Clock Select b31 b30 0 0: P0ϕ 0 1: P0ϕ/4 1 0: P0ϕ/16 1 1: P0ϕ/64.

GTIOR sets the functions of the GTIOCA and GTIOCB pins.

GTIOA[4:0] bits (GTIOCA Pin Function Select)

The GTIOA[4:0] bits select the GTIOCA pin function. For details, see **Table 18.5**

OADFLT bit (GTIOCA Pin Output Value Setting at the Count Stop)

The OADFLT bit sets whether the GTIOCA pin outputs high or low when counting stops.

OAHLD bit (GTIOCA Pin Output Setting at the Start/Stop Count)

The OAHLD bit specifies whether the GTIOCA pin output level is retained or the level at the start/stop of counting depends on the register setting.

[When the OAHLD bit is set to 0]

- The value specified in bit [4] of the GTIOA[4:0] bits is output when counting starts
- The value specified in the OADFLT bit is output when counting stops
- If the OADFLT bit is modified while counting stops, it is immediately reflected in the output.

[When the OAHLD bit is set to 1]

- The output is retained when counting starts or stops.

OAE bit (GTIOCA Pin Output Enable)

The OAE bit disables or enables the GTIOCA pin output.

When the GTCCRA register is in use as an input capture register (at least one bit in the GTICASR register is set to 1), signals will not be output from the GTIOCA pin regardless of the setting of this bit.

OADF[1:0] bits (GTIOCA Pin Disable Value Setting)

The OADF bits select a value to be output from the GTIOCA pin in response to the output disable requests.

NFAEN bit (Noise Filter A Enable)

The NFAEN bit disables or enables the noise filter for input from the GTIOCA pin. Because changing the value of the bit might lead to internal generation of an unexpected edge, select the output compare function for the relevant pin in the GTIOR register before doing so.

NFCSA[1:0] bits (Noise Filter A Sampling Clock Select)

The NFCSA[1:0] bits set the sampling interval for the noise filter of the GTIOCA pin. When setting these bits, wait for two cycles of the selected sampling interval before setting the input-capture function.

GTIOB[4:0] bits (GTIOCB Pin Function Select)

The GTIOB[4:0] bits select the GTIOCB pin function. For details, see **Table 18.5**

OBDFLT bit (GTIOCB Pin Output Value Setting at the Count Stop)

The OBDFLT bit sets whether the GTIOCB pin outputs high or low when counting stops.

OBHLD bit (GTIOCB Pin Output Setting at the Start/Stop Count)

The OBHLD bit specifies whether the GTIOCB pin output level is retained or the level at the start/stop of counting depends on the register setting.

[When the OBHLD bit is set to 0]

- The value specified in bit [4] of the GTIOB[4:0] bits is output when counting starts
- The value specified in the OBDFLT bit is output when counting stops
- If the OBDFLT bit is modified while counting stops, it is immediately reflected in the output.

[When the OBHLD bit is set to 1]

- The output is retained when counting starts or stops.

OBE bit (GTIOCB Pin Output Enable)

The OBE bit disables or enables the GTIOCB pin output.

When the GTCCRB register is in use as an input capture register (at least one bit in the GTICBSR register is set to 1), signals will not be output from the GTIOCB pin regardless of the setting of this bit.

OBDF[1:0] bits (GTIOCB Pin Disable Value Setting)

The OBDF bits select a value to be output from the GTIOCB pin in response to the output disable requests.

NFBEN bit (Noise Filter B Enable)

The NFBEN bit disables or enables the noise filter for input from the GTIOCB pin. Because changing the value of the bit might lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the GTIOR register before doing so.

NFCSB[1:0] bits (Noise Filter B Sampling Clock Select)

The NFCSB[1:0] bits set the sampling interval for the noise filter of the GTIOCB pin. When setting these bits, wait for two cycles of the selected sampling interval before setting the input-capture function.

Table 18.5 Settings of GTIOA[4:0] and GTIOB[4:0] bits

GTIOA/GTIOB[4:0] bits					Function		
b4	b3	b2	b1	b0	b4	b3, b2	b1, b0
0	0	0	0	0	Set initial output low	Retain output at cycle end	Retain output at GTCCRA/GTCCRB compare match
0	0	0	0	1			Output low at GTCCRA/GTCCRB compare match
0	0	0	1	0			Output high at GTCCRA/GTCCRB compare match
0	0	0	1	1			Toggle output at GTCCRA/GTCCRB compare match
0	0	1	0	0		Output low at cycle end	Retain output at GTCCRA/GTCCRB compare match
0	0	1	0	1			Output low at GTCCRA/GTCCRB compare match
0	0	1	1	0			Output high at GTCCRA/GTCCRB compare match
0	0	1	1	1			Toggle output at GTCCRA/GTCCRB compare match
0	1	0	0	0		Output high at cycle end	Retain output at GTCCRA/GTCCRB compare match
0	1	0	0	1			Output low at GTCCRA/GTCCRB compare match
0	1	0	1	0			Output high at GTCCRA/GTCCRB compare match
0	1	0	1	1			Toggle output at GTCCRA/GTCCRB compare match
0	1	1	0	0		Toggle output at cycle end	Retain output at GTCCRA/GTCCRB compare match
0	1	1	0	1			Output low at GTCCRA/GTCCRB compare match
0	1	1	1	0			Output high at GTCCRA/GTCCRB compare match
0	1	1	1	1			Toggle output at GTCCRA/GTCCRB compare match
1	0	0	0	0	Set initial output high	Retain output at cycle end	Retain output at GTCCRA/GTCCRB compare match
1	0	0	0	1			Output low at GTCCRA/GTCCRB compare match
1	0	0	1	0			Output high at GTCCRA/GTCCRB compare match
1	0	0	1	1			Toggle output at GTCCRA/GTCCRB compare match
1	0	1	0	0		Output low at cycle end	Retain output at GTCCRA/GTCCRB compare match
1	0	1	0	1			Output low at GTCCRA/GTCCRB compare match
1	0	1	1	0			Output high at GTCCRA/GTCCRB compare match
1	0	1	1	1			Toggle output at GTCCRA/GTCCRB compare match
1	1	0	0	0		Output high at cycle end	Retain output at GTCCRA/GTCCRB compare match
1	1	0	0	1			Output low at GTCCRA/GTCCRB compare match
1	1	0	1	0			Output high at GTCCRA/GTCCRB compare match
1	1	0	1	1			Toggle output at GTCCRA/GTCCRB compare match
1	1	1	0	0		Toggle output at cycle end	Retain output at GTCCRA/GTCCRB compare match
1	1	1	0	1			Output low at GTCCRA/GTCCRB compare match
1	1	1	1	0			Output high at GTCCRA/GTCCRB compare match
1	1	1	1	1			Toggle output at GTCCRA/GTCCRB compare match

Note 1. The cycle end means an overflow (GTCNT is changed from GTPR to 0 in up-counting) or underflow (GTCNT is changed from 0 to GTPR in down-counting). In this case, the GTCNT counter is cleared for saw waves and for the trough (GTCNT is changed from 0 to 1) for triangle waves.

Note 2. When the timing of a cycle end and the timing of a GTCCRA/GTCCRB compare match are the same in a compare-match operation, the b3 and b2 settings are given priority in saw-wave PWM mode, and the b1 and b0 settings are given priority in any other mode.

Note 3. In event count operation where at least 1 bit in GTUPSR or GTDNSR is set to 1, the setting of b3 and b2 is ignored.

18.2.15 General PWM Timer Interrupt Output Setting Register (GTINTAD)

Address(es): GPT32Em.GTINTAD H'1004_8038 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	GRPABL	GRPABH	GRPDTE	—	—	GRP[1:0]		—	—	—	—	ADTRBDEN	ADTRBUEN	ADTRADEN	ADTRAUEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	GTINTPR[1:0]		GTINTF	GTINTE	GTINTD	GTINTC	GTINTB	GTINTA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	GTINTA	0	R/W	GTCCRA Compare Match/Input Capture Interrupt Enable 0: Disable Interrupt request 1: Enable Interrupt request.
b1	GTINTB	0	R/W	GTCCRB Compare Match/Input Capture Interrupt Enable 0: Disable Interrupt request 1: Enable Interrupt request.
b2	GTINTC	0	R/W	GTCCRC Compare Match Interrupt Enable 0: Disable Interrupt request 1: Enable Interrupt request.
b3	GTINTD	0	R/W	GTCCRD Compare Match Interrupt Enable 0: Disable Interrupt request 1: Enable Interrupt request.
b4	GTINTE	0	R/W	GTCCRE Compare Match Interrupt Enable 0: Disable Interrupt request 1: Enable Interrupt request.
b5	GTINTF	0	R/W	GTCCRF Compare Match Interrupt Enable 0: Disable Interrupt request 1: Enable Interrupt request.
b7, b6	GTINTPR[1:0]	All 0	R/W	GTPR Compare Match Interrupt Enable b7 b6 0 0: Disable Interrupt request 0 1: Interrupt requests in response to overflows for saw waves and crests for triangle waves are enabled. 1 0: Interrupt requests in response to underflows for saw waves and troughs for triangle waves are enabled. 1 1: Interrupt requests in response to overflows and underflows for saw waves and crests and troughs for triangle waves are enabled.
b15 to b8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b16	ADTRAUEN	0	R/W	GTADTRA Compare Match (Up-Counting) A/D Converter Start Request Enable 0: Disable A/D converter start request 1: Enable A/D converter start request.
b17	ADTRADEN	0	R/W	GTADTRA Compare Match (Down-Counting) A/D Converter Start Request Enable 0: Disable A/D converter start request 1: Enable A/D converter start request.

Bit	Bit Name	Initial Value	R/W	Description
b18	ADTRBUEN	0	R/W	GTADTRB Compare Match (Up-Counting) A/D Converter Start Request Enable 0: Disable A/D converter start request 1: Enable A/D converter start request.
b19	ADTRBDEN	0	R/W	GTADTRB Compare Match (Down-Counting) A/D Converter Start Request Enable 0: Disable A/D converter start request 1: Enable A/D converter start request.
b23 to b20	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b25, b24	GRP[1:0]	All 0	R/W	Output Disable Source Select b25 b24 0 0: Select Group A output disable request 0 1: Select Group B output disable request 1 0: Select Group C output disable request 1 1: Select Group D output disable request.
b27, b26	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b28	GRPDTE	0	R/W	Dead Time Error Output Disable Request Enable 0: Disable dead time error output disable request 1: Enable dead time error output disable request.
b29	GRPABH	0	R/W	Same Time Output Level High Disable Request Enable 0: Disable same time output level high disable request 1: Enable same time output level high disable request.
b30	GRPABL	0	R/W	Same Time Output Level Low Disable Request Enable 0: Disable same time output level low disable request 1: Enable same time output level low disable request.
b31	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.

GTINTAD enables or disables interrupt requests, A/D converter start requests and output disable requests.

GTINTA bit (GTCCRA Compare Match/Input Capture Interrupt Enable)

The GTINTA bit enables and disables the interrupt request (CCMPAn) in response to compare matches with or input capture by the GTCCRA register.

GTINTB bit (GTCCRB Compare Match/Input Capture Interrupt Enable)

The GTINTB bit enables and disables the interrupt request (CCMPBn) in response to compare matches with or input capture by the GTCCRB register.

GTINTC bit (GTCCRC Compare Match Interrupt Enable)

The GTINTC bit enables and disables the interrupt request (CMPCn) in response to compare matches with the GTCCRC register.

GTINTD bit (GTCCRD Compare Match Interrupt Enable)

The GTINTD bit enables and disables the interrupt request (CMPDn) in response to compare matches with the GTCCRD register.

GTINTE bit (GTCCRE Compare Match Interrupt Enable)

The GTINTE bit enables and disables the interrupt request (CMPEn) in response to compare matches with the GTCCRE register.

GTINTF bit (GTCCRF Compare Match Interrupt Enable)

The GTINTF bit enables and disables the interrupt request (CMPFn) in response to compare matches with the GTCCRF register.

GTINTPR[1:0] bits (GTPR Compare Match Interrupt Enable)

The GTINTPR[1:0] bits enable and disable the interrupt requests (OVFn and UNFn) in response to compare matches with the GTPR register (and overflows of the GTCNT counter) and underflows of the GTCNT counter.

ADTRAUEN bit (GTADTRA Compare Match (Up-Counting) A/D Converter Start Request Enable)

The ADTRAUEN bit enables or disables A/D converter start requests generated by GTADTRA compare matches during GTCNT up-counting.

ADTRADEN bit (GTADTRA Compare Match (Down-Counting) A/D Converter Start Request Enable)

The ADTRADEN bit enables or disables A/D converter start requests generated by GTADTRA compare matches during GTCNT down-counting.

ADTRBUEN bit (GTADTRB Compare Match (Up-Counting) A/D Converter Start Request Enable)

The ADTRBUEN bit enables or disables A/D converter start requests generated by GTADTRB compare matches during GTCNT up-counting.

ADTRBDEN bit (GTADTRB Compare Match (Down-Counting) A/D Converter Start Request Enable)

The ADTRBDEN bit enables or disables A/D converter start requests generated by GTADTRB compare matches during GTCNT down-counting.

GRP[1:0] bits (Output Disable Source Select)

The GRP[1:0] bits select GTIOCA pin and GTIOCB pin output disable source. The output disable requests to the POEG are output to the group selected by the GRP[1:0] bits. Each of the output disable sources, a dead time error, simultaneous output of high-level, and low-level signal, is also output in accord with the setting of its enabling bit. GTST.ODF shows the request of output disable source group that is selected with the GRP[1:0] bits. Set the GRP[1:0] bits when GTIOR.OAE and OBE bits are both 0.

GRPDTE bit (Dead Time Error Output Disable Request Enable)

The GRPDTE bit enables or disables dead time error output disable request.

GRPABH bit (Same Time Output Level High Disable Request Enable)

The GRPABH bit enables or disables output disable request when GTIOCA pin and GTIOCB pin output 1 at the same time.

GRPABL bit (Same Time Output Level Low Disable Request Enable)

The GRPABL bit enables or disables output disable request when GTIOCA pin and GTIOCB pin output 0 at the same time.

18.2.16 General PWM Timer Status Register (GTST)

Address(es): GPT32Em.GTST H'1004_803C + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	OABLF	OABHF	DTEF	—	—	—	ODF	—	—	—	—	ADTRBDF	ADTRBUF	ADTRADF	ADTRAUF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TUCF	—	—	—	—	ITCNT[2:0]			TCFPU	TCFPO	TCFF	TCFE	TCFD	TCFC	TCFB	TCFA
Initial Value	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R	R	R	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1

Bit	Bit Name	Initial Value	R/W	Description
b0	TCFA	0	R/(W)*1	Input Capture/Compare Match Flag A 0: No input capture/compare match of GTCCRA occurred 1: Input capture/compare match of GTCCRA occurred.
b1	TCFB	0	R/(W)*1	Input Capture/Compare Match Flag B 0: No input capture/compare match of GTCCRB occurred 1: Input capture/compare match of GTCCRB occurred.
b2	TCFC	0	R/(W)*1	Compare Match Flag C 0: No compare match of GTCCRC occurred 1: Compare match of GTCCRC occurred.
b3	TCFD	0	R/(W)*1	Compare Match Flag D 0: No compare match of GTCCRD occurred 1: Compare match of GTCCRD occurred.
b4	TCFE	0	R/(W)*1	Compare Match Flag E 0: No compare match of GTCCRE occurred 1: Compare match of GTCCRE occurred.
b5	TCFF	0	R/(W)*1	Compare Match Flag F 0: No compare match of GTCCRF occurred 1: Compare match of GTCCRF occurred.
b6	TCFPO	0	R/(W)*1	Overflow Flag 0: No overflow (crest) occurred 1: Overflow (crest) occurred.
b7	TCFPU	0	R/(W)*1	Underflow Flag 0: No underflow (trough) occurred 1: Underflow (trough) occurred.
b10 to b8	ITCNT[2:0]	All 0	R	OVFn/UNFn Interrupt Skipping Count Counter Counter for counting the number of times a timer interrupt is skipped.
b14 to b11	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b15	TUCF	1	R	Count Direction Flag 0: GTCNT counter is counting down 1: GTCNT counter is counting up.
b16	ADTRAUF	0	R/(W)*1	GTADTRA Compare Match (Counting Up) A/D Converter Start Request Flag 0: A compare match with the GTADTRA register has not occurred during counting up. 1: A compare match with the GTADTRA register has occurred during counting up.

Bit	Bit Name	Initial Value	R/W	Description
b17	ADTRADF	0	R/(W) *1	GTADTRA Compare Match (Counting Down) A/D Converter Start Request Flag 0: A compare match with the GTADTRA register has not occurred during counting down. 1: A compare match with the GTADTRA register has occurred during counting down.
b18	ADTRBUF	0	R/(W) *1	GTADTRB Compare Match (Counting Up) A/D Converter Start Request Flag 0: A compare match with the GTADTRB register has not occurred during counting up. 1: A compare match with the GTADTRB register has occurred during counting up.
b19	ADTRBDF	0	R/(W) *1	GTADTRB Compare Match (Counting Down) A/D Converter Start Request Flag 0: A compare match with the GTADTRB register has not occurred during counting down. 1: A compare match with the GTADTRB register has occurred during counting down.
b23 to b20	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b24	ODF	0	R	Output Disable Flag 0: No output disable request occurred 1: Output disable request occurred.
b27 to b25	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b28	DTEF	0	R	Dead Time Error Flag 0: No dead time error occurred 1: Dead time error occurred.
b29	OABHF	0	R	Same Time Output Level High Flag 0: GTIOCA pin and GTIOCB pin did not output 1 at the same time 1: GTIOCA pin and GTIOCB pin output 1 at the same time.
b30	OABLF	0	R	Same Time Output Level Low Flag 0: GTIOCA pin and GTIOCB pin did not output 0 at the same time 1: GTIOCA pin and GTIOCB pin output 0 at the same time.
b31	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.

Note 1. Only 0 can be written to this bit. Do not write 1.

GTST indicates the status of the GPT.

TCFA flag (Input Capture/Compare Match Flag A)

The TCFS flag indicates the status for the input capture or compare match of GTCCRA.

[Setting conditions]

- GTCNT = GTCCRA, when the GTCCRA register functions as a compare match register
- GTCNT counter value is transferred to GTCCRA by the input capture signal when the GTCCRA register functions as an input capture register.

[Clearing condition]

- 0 is written to this flag.

TCFB flag (Input Capture/Compare Match Flag B)

The TCFB flag indicates the status for the input capture or compare match of GTCCRB.

[Setting conditions]

- GTCNT = GTCCRB, when the GTCCRB register functions as a compare match register
- GTCNT counter value is transferred to GTCCRB by the input capture signal when the GTCCRB register function as an input capture register.

[Clearing condition]

- 0 is written to this flag.

TCFC flag (Compare Match Flag C)

The TCFC flag indicates the status for the compare match of GTCCRC.

[Setting condition]

- GTCNT = GTCCRC

[Clearing condition]

- 0 is written to this flag. [Not comparing condition]
- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 01b, 10b, 11b (GTCCRC performs buffer operation).

TCFD flag (Compare Match Flag D)

The TCFD flag indicates the status for the compare match of GTCCRD.

[Setting condition]

- GTCNT = GTCCRD.

[Clearing condition]

- 0 is written to this flag. [Not comparing condition]
- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (Triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 10b, 11b (GTCCRD performs buffer operation).

TCFE flag (Compare Match Flag E)

The TCFE flag indicates the status for the compare match of GTCCRE.

[Setting condition]

- GTCNT = GTCCRE.

[Clearing condition]

- 0 is written to this flag. [Not comparing condition]
- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (Triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 01b, 10b, 11b (GTCCRE performs buffer operation).

TCFF flag (Compare Match Flag F)

The TCFF flag indicates the status for the compare match of GTCCRF.

[Setting condition]

- GTCNT = GTCCRF.

[Clearing condition]

- 0 is written to this flag. [Not comparing condition]
- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (Triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 10b, 11b (GTCCRF performs buffer operation).

TCFPO flag (Overflow Flag)

The TCFPO flag indicates when an overflow or a crest has occurred.

[Setting conditions]

- In saw-wave mode, an overflow (GTCNT changes from GTPR to 0 in up-counting) has occurred
- In triangle-wave mode, a crest (GTCNT changes from GTPR to GTPR-1) has occurred
- In counting by hardware sources, an overflow (GTCNT changes from GTPR to 0 in up-counting) has occurred.

[Clearing condition]

- 0 is written to this flag.

TCFPU flag (Underflow Flag)

The TCFPU flag indicates when an underflow or a trough has occurred.

[Setting conditions]

- In saw-wave mode, an underflow (GTCNT changes from 0 to GTPR in down-counting) has occurred
- In triangle-wave mode, a crest (GTCNT changes from 0 to 1) has occurred
- In counting by hardware sources, an underflow (GTCNT changes from 0 to GTPR in down-counting) has occurred.

[Clearing condition]

- 0 is written to this flag.

ITCNT[2:0] bits (OVFn/UNFn Interrupt Skipping Count Counter)

When the OVFn/UNFn (n=0 to 7) interrupt skipping function is used (the GTITC.IVTC[1:0] bits are set to a value other than 00b), the counter in the ITCNT[2:0] bits increments by 1 every time the OVFn/UNFn interrupt source that is selected in GTITC.IVTC[1:0] is generated.

[Clearing conditions]

- The OVFn/UNFn interrupt skipping function is not used (GTITC.IVTT[2:0] is 000b when GTITC.IVTC[1:0] is 00b)
- The OVFn/UNFn interrupt skipping count matches the specified count (ITCNT[2:0] matches the skipping count specified in GTITC.IVTT[2:0]).

TUCF flag (Count Direction Flag)

The TUCF flag indicates the count direction of GTCNT. In event count operation, this flag is set to 1 in up-counting and is set to 0 in down-counting.

ADTRAUF flag (GTADTRA Compare Match (Counting Up) A/D Converter Start Request Flag)

The ADTRAUF flag indicates whether or not a compare match with the GTADTRA register has occurred during counting up.

[Setting condition]

- GTCNT = GTADTRA during counting up

[Clearing condition]

- 0 is written to this flag.

ADTRADF flag (GTADTRA Compare Match (Counting Down) A/D Converter Start Request Flag)

The ADTRADF flag indicates whether or not a compare match with the GTADTRA register has occurred during counting down.

[Setting condition]

- GTCNT = GTADTRA during counting down

[Clearing condition]

- 0 is written to this flag.

ADTRBUF flag (GTADTRB Compare Match (Counting Up) A/D Converter Start Request Flag)

The ADTRBUF flag indicates whether or not a compare match with the GTADTRB register has occurred during counting up.

[Setting condition]

- GTCNT = GTADTRB during counting up

[Clearing condition]

- 0 is written to this flag.

ADTRBDF flag (GTADTRB Compare Match (Counting Down) A/D Converter Start Request Flag)

The ADTRBDF flag indicates whether or not a compare match with the GTADTRB register has occurred during counting down.

[Setting condition]

- GTCNT = GTADTRB during counting down

[Clearing condition]

- 0 is written to this flag.

ODF flag (Output Disable Flag)

The ODF flag shows the request of the output disable source group that is selected by GRP[1:0] bits. When output is disabled, an output disable control is not released within the same cycle in which an output disable request is negated. It is released in the next cycle.

DTEF flag (Dead Time Error Flag)

The DTEF flag indicates that the timer output toggle point after the automatic addition of dead time has exceeded the timer cycle.

This flag returns to 0 when the timer output toggle point after the automatic addition of dead time is returned to the cycle. This flag is read only. Writing 0 to clear the flag is not allowed.

When an interrupt by the DTEF flag is enabled (GTINTAD.GRPDTE = 1), the DTEF flag is output to POEG as an output disable request each time the DTEF flag changes from 0 to 1.

[Setting condition]

- The timer output toggle point after the automatic addition of dead time has exceeded the timer cycle.
  For triangle wave in up-counting: GTCCRA – GTDVU ≤ 0
  For triangle wave in down-counting: GTCCRA – GTDVD < 0
  For saw wave 1 shot pulse mode in up-counting:
  GTCCRA – GTDVU < 0 or GTCCRA + GTDVD > GTPR
  For saw wave 1 shot pulse mode in down-counting:
  GTCCRA + GTDVU > GTPR or GTCCRA - GTDVD < 0

[Clearing condition]

- The timer output toggle point after the automatic addition of dead time is within the timer cycle.

OABHF flag (Same Time Output Level High Flag)

The OABHF flag indicates that GTIOCA pin and GTIOCB pin output 1 at the same time.

When GTIOCA pin or GTIOCB pin output 0, this flag returns to 0. This flag is read only. Writing 0 to clear the flag is not allowed. When an interrupt by the OABHF flag is enabled (GTINTAD.GRPABH = 1), the OABHF flag is output to POEG as an output disable request.

[Setting condition]

- GTIOCA pin and GTIOCB pin output 1 at the same time when both the OAE and OBE bits are set to 1.

[Clearing conditions]

- GTIOCA pin output value is different from GTIOCB pin output value when both the OAE and OBE bits are set to 1
- GTIOCA pin and GTIOCB pin output 0 at the same time when both the OAE and OBE bits are set to 1
- Either OAE bit or OBE bit is set to 0.

OABLF flag (Same Time Output Level Low Flag)

The OABLF flag indicates that GTIOCA pin and GTIOCB pin output 0 at the same time.

When GTIOCA pin or GTIOCB pin output 1, this flag returns to 0. This flag is read only. Writing 0 to clear the flag is not allowed. When an interrupt by the OABLF flag is enabled (GTINTAD.GRPABL = 1), the OABLF flag is output to POEG as an output disable request.

[Setting condition]

- GTIOCA pin and GTIOCB pin output 0 at the same time when both the OAE and OBE bits are set to 1.

[Clearing conditions]

- GTIOCA pin output value is different from GTIOCB pin output value when both the OAE and OBE bits are set to 1
- GTIOCA pin and GTIOCB pin output 1 at the same time when both the OAE and OBE bits are set to 1
- At least either OAE bit or OBE bit is set to 0.

The compare-target signals to generate the OABHF/OABLF flag are the compare match outputs (PWM outputs) signals before masked by the output disable function. When the output disable state is performed, a compare match also performs continuously in the GPT and the OABHF/OABLF flag is updated in association with the result of the compared values.

18.2.17 General PWM Timer Buffer Enable Register (GTBER)

Address(es): GPT32Em.GTBER H'1004_8040 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	ADTDB	ADTTB[1:0]		—	ADTDA	ADTTA[1:0]		—	CCRSWT	PR[1:0]		CCRB[1:0]		CCRA[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	BD[3]	BD[2]	BD[1]	BD[0]
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	BD[0]	0	R/W	GTCCR Buffer Operation Disable 0: Enable buffer operation 1: Disable buffer operation.
b1	BD[1]	0	R/W	GTPR Buffer Operation Disable 0: Enable buffer operation 1: Disable buffer operation.
b2	BD[2]	0	R/W	GTADTR Buffer Operation Disable 0: Enable buffer operation 1: Disable buffer operation.
b3	BD[3]	0	R/W	GTDV Buffer Operation Disable 0: Enable buffer operation 1: Disable buffer operation.
b15 to b4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b17, b16	CCRA[1:0]	All 0	R/W	GTCCRA Buffer Operation b17 b16 0 0: No buffer operation 0 1: Single buffer operation (GTCCRA ⇔ GTCCRC) 1 x: Double buffer operation (GTCCRA ⇔ GTCCRC ⇔ GTCCRD).
b19, b18	CCRB[1:0]	All 0	R/W	GTCCRB Buffer Operation b19 b18 0 0: No buffer operation 0 1: Single buffer operation (GTCCRB ⇔ GTCCRE) 1 x: Double buffer operation (GTCCRB ⇔ GTCCRE ⇔ GTCCRF).
b21, b20	PR[1:0]	All 0	R/W	GTPR Buffer Operation b21 b20 0 0: No buffer operation 0 1: Single buffer operation (GTPBR ⇔ GTPR) 1 x: Double buffer operation (GTPDBR ⇔ GTPBR ⇔ GTPR).
b22	CCRSWT	0	R/W	GTCCRA and GTCCRB Forcible Buffer Operation Writing 1 to this bit forces a buffer transfer of GTCCRA and GTCCRB. This bit automatically returns to 0 after 1 is written. This bit is read as 0.
b23	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.

Bit	Bit Name	Initial Value	R/W	Description
b25, b24	ADTTA[1:0]	All 0	R/W	GTADTRA Buffer Transfer Timing Select • Triangle waves b25 b24 0 0: No transfer 0 1: Transfer at crest 1 0: Transfer at trough 1 1: Transfer at both crest and trough. • Saw waves b25 b24 0 0: No transfer Values other than 0 0: Transfer is triggered when the counter underflows (in down-counting), overflows (in up- counting), or is cleared.
b26	ADTDA	0	R/W	GTADTRA Double Buffer Operation 0: Single buffer operation (GTADTBRA ⇔ GTADTRA) 1: Double buffer operation (GTADTDBRA ⇔ GTADTBRA ⇔ GTADTDRA).
b27	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b29, b28	ADTTB[1:0]	All 0	R/W	GTADTRB Buffer Transfer Timing Select • Triangle waves b29 b28 0 0: No transfer 0 1: Transfer at crest 1 0: Transfer at trough 1 1: Transfer at both crest and trough. • Saw waves b29 b28 0 0: No transfer Values other than 0 0: Transfer is triggered when the counter underflows (in down-counting), overflows (in up- counting), or is cleared.
b30	ADTDB	0	R/W	GTADTRB Double Buffer Operation 0: Single buffer operation (GTADTBRB ⇔ GTADTRB) 1: Double buffer operation (GTADTDBRB ⇔ GTADTBRB ⇔ GTADTDRB).
b31	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.

GTBER provides settings for the buffer operation and must be set while the GTCNT operation stops.

BD[0] bit (GTCCR Buffer Operation Disable)

The BD[0] bit disables buffer operation using GTCCRA, GTCCRB, GTCCRC, GTCCRD, GTCCRE, and GTCCRF of the combined GPT.

When GTDTCR.TDE is 1 and when BD[0] is set to 0, GTCCRB does not perform buffer operation and the GTCCRB register is automatically set to a compare match value for a negative-phase waveform with dead time.

BD[1] bit (GTPR Buffer Operation Disable)

The BD[1] bit disables buffer operation using GTPR, GTPBR, and GTPDBR of the combined GPT.

BD[2] bit (GTADTR Buffer Operation Disable)

The BD[2] bit disables buffer operation using GTADTRA, GTADTRB, GTADTBRA, GTADTBRB, GTADTDBRA, and GTADTDBRB of the combined GPT. In event count operation, this bit is not available and the GTADTR buffer operation is not performed.

BD[3] bit (GTDV Buffer Operation Disable)

The BD[3] bit disables buffer operation using GTDVU, GTDVD, GTDBD, and GTDBU of the combined GPT.

When the GTDTCR.TDFER bit is set to 1 and when BD[3] is set to 0, buffer operation is not performed and the GTDVD value is set as a value of GTDVU automatically. In event count operation, this bit is not available and the GTDV buffer operation is not performed.

CCRA[1:0] bits (GTCCRA Buffer Operation)

The CCRA[1:0] bits set buffer operation using GTCCRA, GTCCRC, and GTCCRD of the combined GPT. When buffer operation is restricted by the operating mode set in GTCR, the GTCR setting is given priority.*[1]

CCRB[1:0] bits (GTCCRB Buffer Operation)

The CCRB[1:0] bits set buffer operation using GTCCRB, GTCCRE, and GTCCRF of the combined GPT. When buffer operation is restricted by the operating mode set in GTCR, the GTCR setting is given priority.* [1]

PR[1:0] bits (GTPR Buffer Operation)

The PR[1:0] bits set buffer operation using GTPR, GTPBR, and GTPDBR of the combined GPT.

CCRSWT bit (GTCCRA and GTCCRB Forcible Buffer Operation)

Writing 1 to the CCRSWT bit forcibly performs a buffer transfer of GTCCRA and GTCCRB. This bit automatically returns to 0 after 1 is written. This bit is read as 0 and is only valid when counting is stopped with a specified compare match operation.

ADTTA[1:0] bits (GTADTRA Buffer Transfer Timing Select)

The ADTTA[1:0] bits set the transfer timing for buffer operation of GTADTRA, GTADTBRA, and GTADTDBRA. These bits are not available in event count operation.

ADTDA bit (GTADTRA Double Buffer Operation)

The ADTDA bit sets buffer operation with GTADTRA, GTADTBRA, and GTADTDBRA combined. This bit is not available in event count operation.

ADTTB[1:0] bits (GTADTRB Buffer Transfer Timing Select)

The ADTTB[1:0] bits set the transfer timing for buffer operation of GTADTRB, GTADTBRB, and GTADTDBRB. This bit is not available in event count operation.

ADTDB bit (GTADTRB Double Buffer Operation)

The ADTDB bit sets buffer operation using GTADTRB, GTADTBRB, and GTADTDBRB of the combined GPT. This bit is not available in event count operation.

Note 1. The buffer operation mode is fixed in saw-wave one-shot pulse mode, or triangle-wave PWM mode 3 (64-bit transfer at trough).

18.2.18 General PWM Timer Interrupt and A/D Converter Start Request Skipping Setting Register (GTITC)

Address(es): GPT32Em.GTITC H'1004_8044 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	ADTBL	—	ADTAL	—	IVTT[2:0]			IVTC[1:0]		ITLF	ITLE	ITLD	ITLC	ITLB	ITLA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	ITLA	0	R/W	GTCCRA Compare Match/Input Capture Interrupt Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b1	ITLB	0	R/W	GTCCRB Compare Match/Input Capture Interrupt Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b2	ITLC	0	R/W	GTCCRC Compare Match Interrupt Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b3	ITLD	0	R/W	GTCCRD Compare Match Interrupt Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b4	ITLE	0	R/W	GTCCRE Compare Match Interrupt Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b5	ITLF	0	R/W	GTCCRF Compare Match Interrupt Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b7, b6	IVTC[1:0]	All 0	R/W	OVFn/UNFn Interrupt Skipping Function Select b7 b6 0 0: Do not perform skipping 0 1: Count and skip both overflow and underflow for saw waves and crest for triangle waves 1 0: Count and skip both overflow and underflow for saw waves and trough for triangle waves 1 1: Count and skip both overflow and underflow for saw waves and both crest and trough for triangle waves.

Bit	Bit Name	Initial Value	R/W	Description
b10 to b8	IVTT[2:0]	All 0	R/W	OVFn/UNFn Interrupt Skipping Count Select b10 b8 0 0 0: No skipping 0 0 1: Skipping count of 1 0 1 0: Skipping count of 2 0 1 1: Skipping count of 3 1 0 0: Skipping count of 4 1 0 1: Skipping count of 5 1 1 0: Skipping count of 6 1 1 1: Skipping count of 7.
b11	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b12	ADTAL	0	R/W	GTADTRA A/D Converter Start Request Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b13	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b14	ADTBL	0	R/W	GTADTRB A/D Converter Start Request Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b31 to b15	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTITC sets the skipping function for the GTCNT counter overflow (GTPR compare match) interrupt (OVFn) and underflow interrupt (UNFn). It also specifies whether to link other interrupts and A/D converter start requests with the OVFn/UNFn interrupt skipping function. An output disable request to the POEG cannot be linked with the OVFn/UNFn interrupt skipping function. This bit is not available in event count operation.

ITLA bit (GTCCRA Compare Match/Input Capture Interrupt Link)

The ITLA bit specifies whether to link the GTCCRA compare match/input capture interrupt (GTCIA) with the OVFn/UNFn interrupt skipping function.

ITLB bit (GTCCRB Compare Match/Input Capture Interrupt Link)

The ITLB bit specifies whether to link the GTCCRB compare match/input capture interrupt (GTCIB) with the OVFn/UNFn interrupt skipping function.

ITLC bit (GTCCRC Compare Match Interrupt Link)

The ITLC bit specifies whether to link the GTCCRC compare match interrupt (GTCIC) with the OVFn/UNFn interrupt skipping function.

ITLD bit (GTCCRD Compare Match Interrupt Link)

The ITLD bit specifies whether to link the GTCCRD compare match interrupt (GTCID) with the OVFn/UNFn interrupt skipping function.

ITLE bit (GTCCRE Compare Match Interrupt Link)

The ITLE bit specifies whether to link the GTCCRE compare match interrupt (GTCIE) with the OVFn/UNFn interrupt skipping function.

ITLF bit (GTCCRF Compare Match Interrupt Link)

The ITLF bit specifies whether to link the GTCCRF compare match interrupt (GTCIF) with the OVFn/UNFn interrupt skipping function.

IVTC[1:0] bits (OVFn/UNFn Interrupt Skipping Function Select)

The IVTC[1:0] bits set the skipping function for the GTPR compare match (GTCNT overflow) interrupt (OVFn) and GTCNT counter underflow interrupt (UNFn).

IVTT[2:0] bits (OVFn/UNFn Interrupt Skipping Count Select)

The IVTT[2:0] bits set the skipping count for the GTPR compare match (GTCNT overflow) interrupt (OVFn) and GTCNT counter underflow interrupt (UNFn). When modifying the IVTT[2:0] bits, first set the IVTC[1:0] bits to 00b.

ADTAL bit (GTADTRA A/D Converter Start Request Link)

The ADTAL bit specifies whether to link the GTADTRA A/D converter start request with OVFn/UNFn interrupt skipping function.

ADTBL bit (GTADTRB A/D Converter Start Request Link)

The ADTBL bit specifies whether to link the GTADTRB A/D converter start request with OVFn/UNFn interrupt skipping function.

18.2.19 General PWM Timer Counter (GTCNT)

Address(es): GPT32Em.GTCNT H'1004_8048 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTCNT is a 32-bit read/write counter and can only be written to after counting stops. GTCNT must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited. GTCNT must be set within the range of $0 \leq GTCNT \leq GTPR$.

18.2.20 General PWM Timer Compare Capture Register n (GTCCRn) (n = A to F)

Address(es): GPT32Em.GTCCRA H'1004_804C + H'0100 × m (m = 0 to 7)
GPT32Em.GTCCRB H'1004_8050 + H'0100 × m (m = 0 to 7)
GPT32Em.GTCCRC H'1004_8054 + H'0100 × m (m = 0 to 7)
GPT32Em.GTCCRE H'1004_8058 + H'0100 × m (m = 0 to 7)
GPT32Em.GTCCRD H'1004_805C + H'0100 × m (m = 0 to 7)
GPT32Em.GTCCRF H'1004_8060 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTCCRn registers are read/write registers.

GTCCRA and GTCCRB are registers used for both output compare and input capture.

GTCCRC and GTCCRE are compare match registers that can also function as buffer registers for GTCCRA and GTCCRB.

GTCCRD and GTCCRF are compare match registers that can also function as buffer registers for GTCCRC and GTCCRE (double-buffer registers for GTCCRA and GTCCRB).

18.2.21 General PWM Timer Cycle Setting Register (GTPR)

Address(es): GPT32Em.GTPR H'1004_8064 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTPR is a read/write register that sets the maximum count value of GTCNT. For saw waves, the value of (GTPR + 1) is the cycle. For triangle waves, the value of (GTPR value × 2) is the cycle.

18.2.22 General PWM Timer Cycle Setting Buffer Register (GTPBR)

Address(es): GPT32Em.GTPBR H'1004_8068 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTPBR is a read/write register that functions as a buffer register for GTPR.

18.2.23 General PWM Timer Cycle Setting Double-Buffer Register (GTPDBR)

Address(es): GPT32Em.GTPDBR H'1004_806C + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTPDBR is a 32-bit read/write register that functions as a buffer register for GTPBR (double-buffer register for GTPR).

18.2.24 A/D Converter Start Request Timing Register n (GTADTRn) (n = A, B)

Address(es): GPT32Em.GTADTRA H'1004_8070 + H'0100 × m (m = 0 to 7)
GPT32Em.GTADTRB H'1004_807C + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTADTRn registers are 32-bit read/write registers that set the timing of A/D converter start request generation. When the GTADTRn value matches the GTCNT counter value, an A/D converter start request is generated. GTADTRn must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited.

18.2.25 A/D Converter Start Request Timing Buffer Register n (GTADTBRn) (n = A, B)

Address(es): GPT32Em.GTADTBRA H'1004_8074 + H'0100 × m (m = 0 to 7)
GPT32Em.GTADTBRB H'1004_8080 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTADTBRn registers are 32-bit read/write registers that function as buffer registers for GTADTRn. GTADTBRn must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited.

18.2.26 A/D Converter Start Request Timing Double-Buffer Register n (GTADTDBRn) (n = A, B)

Address(es): GPT32Em.GTADTDBRA H'1004_8078 + H'0100 × m (m = 0 to 7)
GPT32Em.GTADTDBRB H'1004_8084 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTADTDBRn registers are 32-bit read/write registers that function as buffer registers for GTADTBRn (double-buffer registers for GTADTR). GTADTDBRn must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited.

18.2.27 General PWM Timer Dead Time Control Register (GTDTCR)

Address(es): GPT32Em.GTDTCR H'1004_8088 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	TDFER	—	—	TDBDE	TDBUE	—	—	—	TDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	TDE	0	R/W	Negative-Phase Waveform Setting 0: Set GTCCRB without using GTDVU and GTDVD. 1: Use GTDVU and GTDVD to set the compare match value for negative-phase waveform with automatic dead time in GTCCRB.
b3 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b4	TDBUE	0	R/W	GTDVU Buffer Operation Enable 0: Disable GTDVU buffer operation 1: Enable GTDVU buffer operation.
b5	TDBDE	0	R/W	GTDVD Buffer Operation Enable 0: Disable GTDVD buffer operation 1: Enable GTDVD buffer operation.
b7, b6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	TDFER	0	R/W	GTDVD Setting 0: Set GTDVU and GTDVD separately 1: Automatically set the value written to GTDVU to GTDVD.
b31 to b9	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTDTCR enables automatic setting of a compare match value for negative-phase waveform with dead time.

TDE bit (Negative-Phase Waveform Setting)

The TDE bit specifies whether to use GTDVU and GTDVD. When GTDVU and GTDVD are used, the compare match value for a negative-phase waveform with dead time obtained by the compare match value of a positive-phase waveform (GTCCRA) and the dead time value (GTDVU and GTDVD), is automatically set in GTCCRB.

The TDE bit setting is ignored in saw-wave PWM mode, and automatic setting does not take place.

The GTCCRB value is automatically set and has the following upper and lower limit values. If the obtained GTCCRB value is not within the upper or lower limit, the following limit value is set in GTCCRB and the GTST.DTEF flag is set to 1. However, in triangle waves, when the obtained GTCCRB value exceeds the upper limit value, the GTST.DTEF flag is set to 0.

- Triangle waves
  Upper limit value: GTPR – 1
  Lower limit value: 1 in up-counting, 0 in down-counting
- Saw-wave one-shot pulse mode
  Upper limit value: GTPR
  Lower limit value: 0

TDBUE bit (GTDVU Buffer Operation Enable)

The TDBUE bit enables buffer operation with GTDVU and GTDBU combined. The buffer transfer timing is the trough for triangle waves, and an overflow or underflow for saw waves.

TDBDE bit (GTDVD Buffer Operation Enable)

The TDBDE bit enables buffer operation with GTDVD and GTDBD combined. The buffer transfer timing is the trough for triangle waves, and an overflow or underflow for saw waves. When this bit and the TDFER bit are set to 1 simultaneously, the TDFER bit setting is given priority.

TDFER bit (GTDVD Setting)

The TDFER bits sets whether or not the value written to GTDVU is also set to GTDVD automatically.

18.2.28 General PWM Timer Dead Time Value Register n (GTDVn) (n = U, D)

Address(es): GPT32Em.GTDVU H'1004_808C + H'0100 x m (m = 0 to 7)
GPT32Em.GTDVD H'1004_8090 + H'0100 x m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTDVn is a 32-bit read/write register that sets the dead time for generating PWM waveforms with dead time. GTDVU is used for up-counting and GTDVD is used for down-counting.

It is prohibited to set the GTDVn register with a value larger than or equal to that of the GTPR register.

Do not set the change point of waveform beyond the count cycle period, when the automatic dead time setting function is used. By reading the GTCCRB register, can be seen the change point for the reverse-phase waveform after the dead time added, which is set in the automatic dead time setting function.

When GTDVn is used, writing to GTCCRB is not allowed. When this register is set to 0, waveforms without dead time are output. GTDVn must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited.

When GTDVn buffer operation is enabled, GTDBn can be written anytime. The value of GTDBn is transferred to GTDVn at the end of the count cycle period. When the GTDVn buffer operation is disabled, set the CST bit in the GTCR register to stop the GPT with before changing GTDVn to a new value. When GTDVn buffer operation is disabled, to change GTDVn to a new value, stop the GPT with the CST bit in the GTCR register.

18.2.29 General PWM Timer Dead Time Buffer Register n (GTDBn) (n = U, D)

Address(es): GPT32Em.GTDBU H'1004_8094 + H'0100 × m (m = 0 to 7)
GPT32Em.GTDBD H'1004_8098 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTDBn is a 32-bit read/write register that functions as a buffer register for GTDVn.

18.2.30 General PWM Timer Output Protection Function Status Register (GTSOS)

Address(es): GPT32Em.GTSOS H'1004_809C + H'0100 x m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOS[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	SOS[1:0]	All 0	R	Output Protection Function Status b1 b0 0 0: Normal operation 0 1: Protected state (set GTCCRA = 0 during transfer at trough or crest) 1 0: Protected state (set GTCCRA ≥ GTPR during transfer at trough) 1 1: Protected state (set GTCCRA ≥ GTPR during transfer at crest).
b7 to b2	—	All 0	R	Reserved These bits are read as 0.
b9, b8	—	All 0	R	Reserved The read value is undefined.
b31 to b10	—	All 0	R	Reserved These bits are read as 0.

GTSOS is a status register that indicates the status of the output protection function. The output protection function is enabled only when the dead time is automatically set (GTDTCR.TDE bit = 1) in triangle-wave mode.

SOS[1:0] bits (Output Protection Function Status)

The SOS[1:0] bits indicate the status of the output protection function in triangle-wave PWM mode.

18.2.31 General PWM Timer Output Protection Function Temporary Release Register (GTSOTR)

Address(es): GPT32Em.GTSOTR H'1004_80A0 + H'0100 x m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOTR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	SOTR	0	R/W	Output Protection Function Temporary Release 0: Do not release protected state 1: Release protected state.
b31 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTSOTR temporarily releases the protected state of GTIOCB pin output when output protection is set. The protected state can be released only when GTSOS.SOS[1:0] bits = 10b (protected state in which GTCCRA ≥ GTPR has occurred during transfer at trough). The protected state cannot be released in any other case.

SOTR bit (Output Protection Function Temporary Release)

The SOTR bit specifies whether to temporarily release the protected state of the GTIOCB pin output in an output protected state. When the SOTR bit is set to 1, the output protection function is canceled from the first trough. When the SOTR bit is set to 0, output protection resumes from the first trough.

18.3 Operation

18.3.1 Basic Operation

Each channel has a 32-bit timer that performs a periodic count operation using the count clock and hardware sources. The count function provides both up-counting and down-counting. The GTPR controls the count cycle. When the GTCNT counter value matches the value in GTCCRA or GTCCRB, the output from the associated pin GTIOCA or GTIOCB can be changed. GTCCRA or GTCCRB can be used as an input capture register with hardware resources. GTCCRC and GTCCRD can function as buffer registers for GTCCRA. GTCCRE and GTCCRF can function as buffer registers for GTCCRB.

18.3.1.1 Counter Operation

(1) Counter start and stop

The counter of each channel starts the count operation by setting GTCR.CST to 1. The GTCR.CST bit value is changed by following sources.

- Writing to GTCR register
- Writing 1 to the bit in GTSTR associated with the GPT channel number when the GTSSR.CSTRT bit set to 1
- Writing 1 to the bit in GTSTP associated with the GPT channel number when the GTPSR.CSTOP bit set to 1
- The hardware source selected in the GTSSR register
- The hardware source selected in the GTPSR register.

(2) Periodic count operation in up-counting by count clock

The GTCNT counter in each channel starts up-counting when the associated GTCR.CST bit is set to 1 with GTUPSR and GTDNSR registers set to H’0000_0000. When the GTCNT value changes from the GTPR value to 0 (overflow), the GTST.TCFPO flag is set to 1. When GTCNT overflows, up-counting resumes from H’0000_0000.

Figure 18.2 shows an example of a periodic count operation in up-counting.

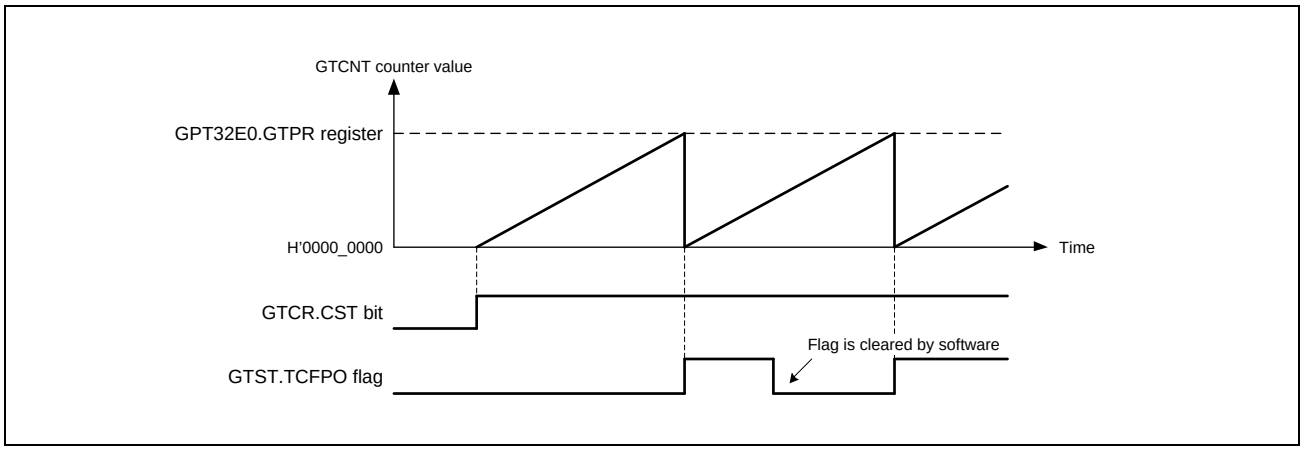


Figure 18.2 Example of periodic count operation in up-counting by the count clock

Figure 18.3 shows an example setting for periodic count operation in up-counting.

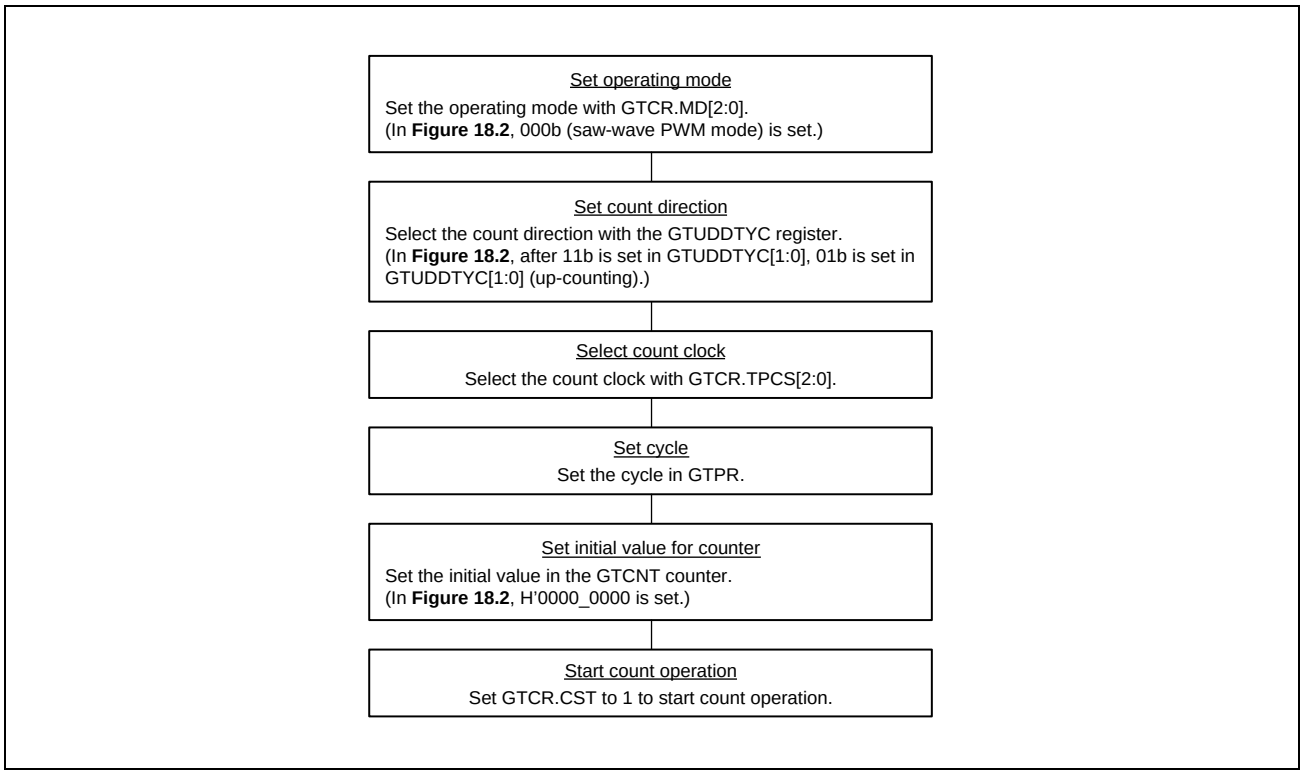


Figure 18.3 Example setting for a periodic count operation in up-counting by the count clock

(3) Periodic count operation in down-counting by count clock

The GTCNT counter in each channel can perform down-counting by setting GTUDDTYC.UD with GTUPSR and GTDNSR registers set to H'0000_0000. When GTCNT changes from 0 to the GTPR value (underflow), GTST.TCFPU is set to 1. When the GTCNT counter underflows, down-counting resumes from the GTPR value.

Figure 18.4 shows an example of periodic count operation in down-counting by the count clock.

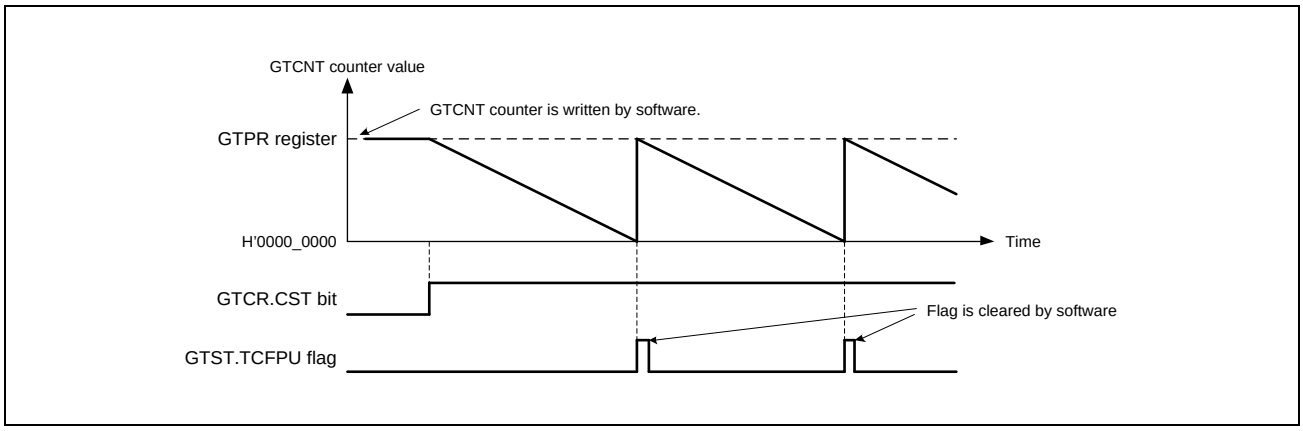


Figure 18.4 Example of periodic count operation in down-counting by the count clock

Figure 18.5 shows an example setting for periodic count operation in down-counting by the count clock.

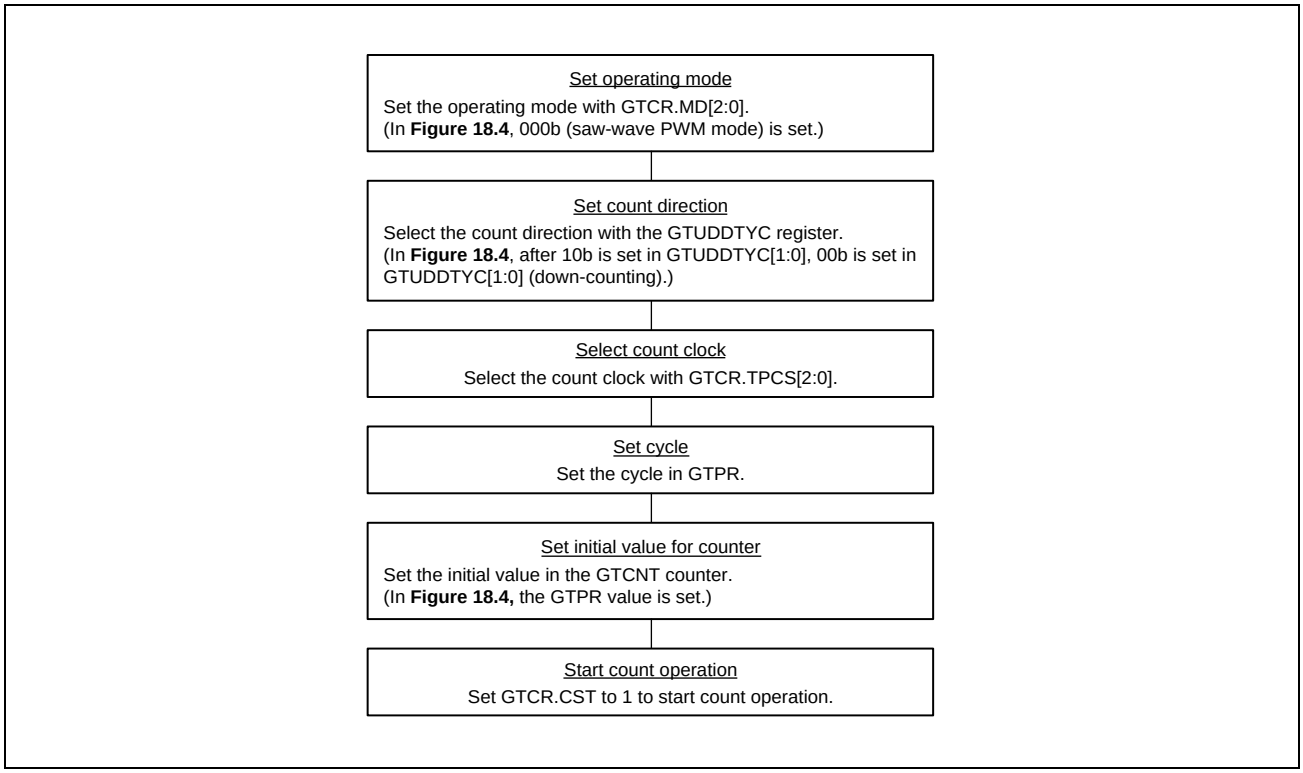


Figure 18.5 Example setting for periodic count operation in down-counting by count clock

(4) Event count operation in up-counting using hardware sources

The GTCNT counter in each channel can perform up-counting using hardware sources as set in GTUPSR.

When GTUPSR is set to enable, the count clock selected in GTCR.TPCS[2:0] and the count direction selected in GTUDDTYC.UD are ignored. If up-counting and down-counting using hardware sources occur at the same time, the GTCNT counter value does not change. The overflow behavior for up-counting using hardware sources is the same as for up-counting by the count clock.

When GTCR.CST bit is set to 1 to count up using hardware sources, the count operation is enabled. After GTCR.CST is set to 1, the counter cannot count up for 1 clock cycle as specified in GTCR.TPCS[2:0] because the count operation is synchronized by the count clock selected by GTCR.TPCS[2:0]. Set the GTCR.TPCS[2:0] bits to 000b to select counting up of single cycles of the P0ϕ clock with a delay of one clock cycle after the GTCR.CST bit has been set to 1.

Figure 18.6 shows an example of a periodic count operation in up-counting by a hardware resource (rising edge of GTETRGA pin).

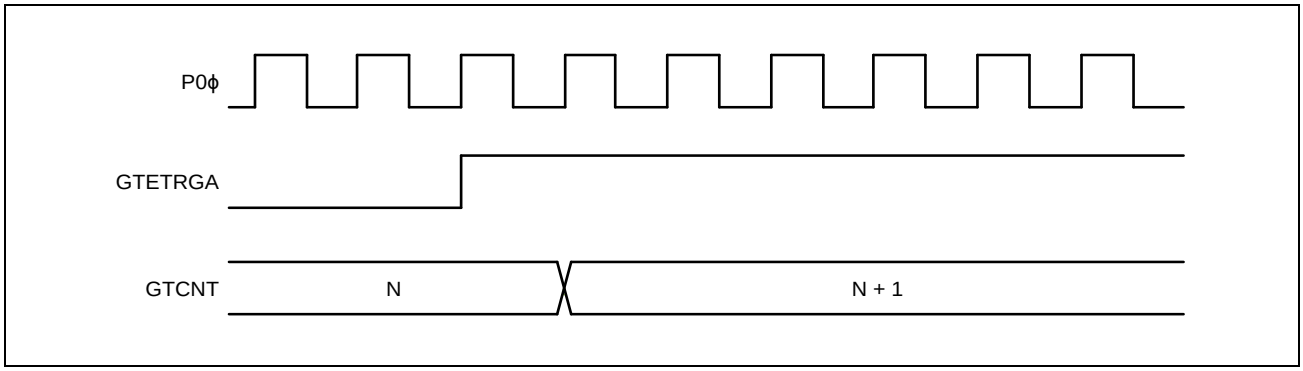


Figure 18.6 Example of periodic count operation in up-counting using hardware sources

Figure 18.7 shows an example setting for periodic count operation in up-counting by the count clock.

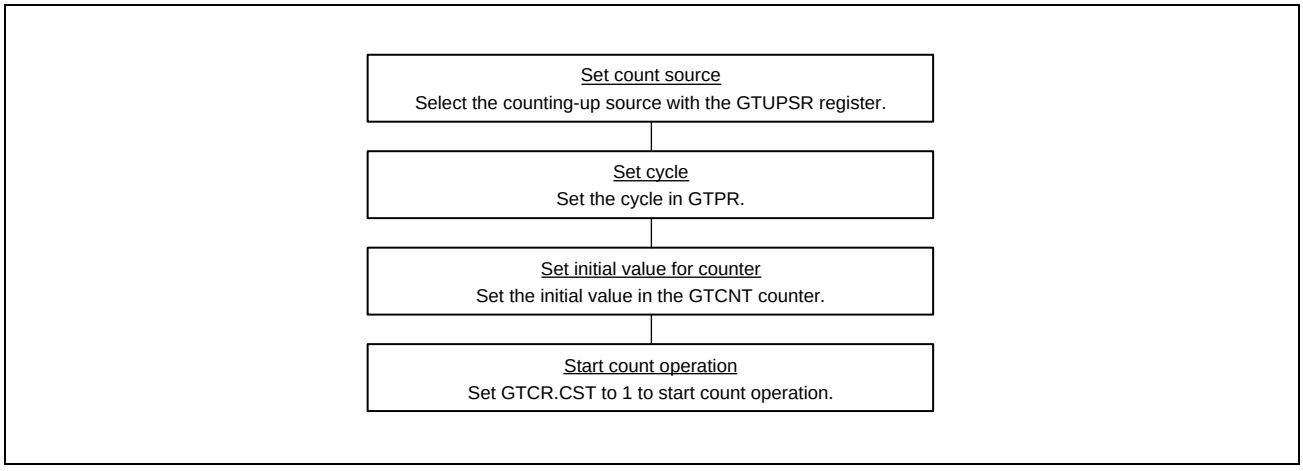


Figure 18.7 Example setting for an event count operation in up-counting using hardware sources

(5) Event count operation in down-counting using hardware sources

The GTCNT counter in each channel can perform down-counting using hardware sources set in the GTDNSR. When GTDNSR is set to enable, the count clock selected in GTCR.TPCS[2:0] and the count direction selected in GTUDDTYC.UD are ignored. If up-counting and down-counting using hardware sources occur at the same time,

GTCNT counter value does not change. The underflow behavior for down-counting using hardware sources is the same as for down-counting by the count clock.

When GTCR.CST bit is set to 1 to count down using hardware sources, the count operation is enabled. After GTCR.CST is set to 1, the counter cannot count down for 1 clock cycle as specified in GTCR.TPCS[2:0] because the count operation is synchronized with the count clock selected by GTCR.TPCS[2:0]. Set the GTCR.TPCS[2:0] bits to 000b to select counting up of single cycles of the P0ϕ clock with a delay of one clock cycle after the GTCR.CST bit has been set to 1.

Figure 18.8 shows an example of a periodic count operation in down-counting by a hardware resource (rising edge of GTETRGA pin).

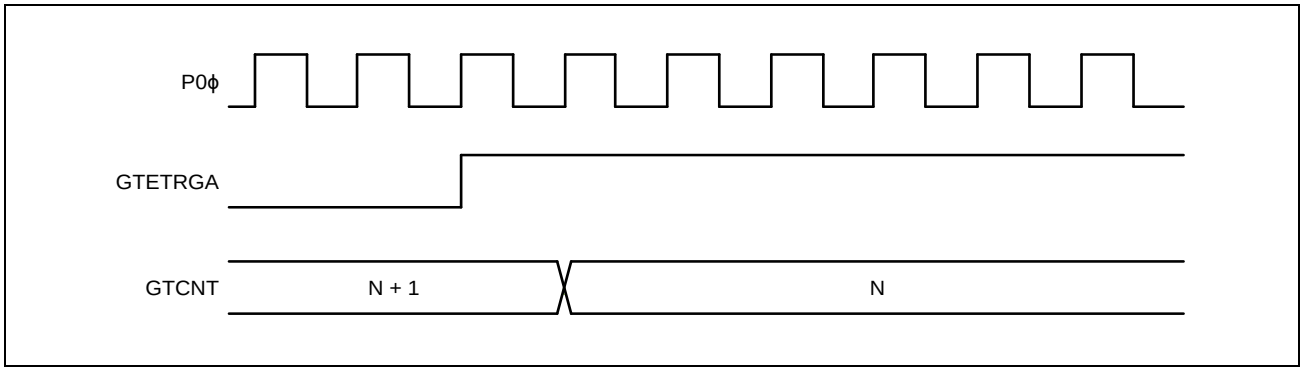


Figure 18.8 Example of event count operation in down-counting using hardware sources

Figure 18.9 shows an example setting for a periodic count operation in down-counting using a hardware resource.

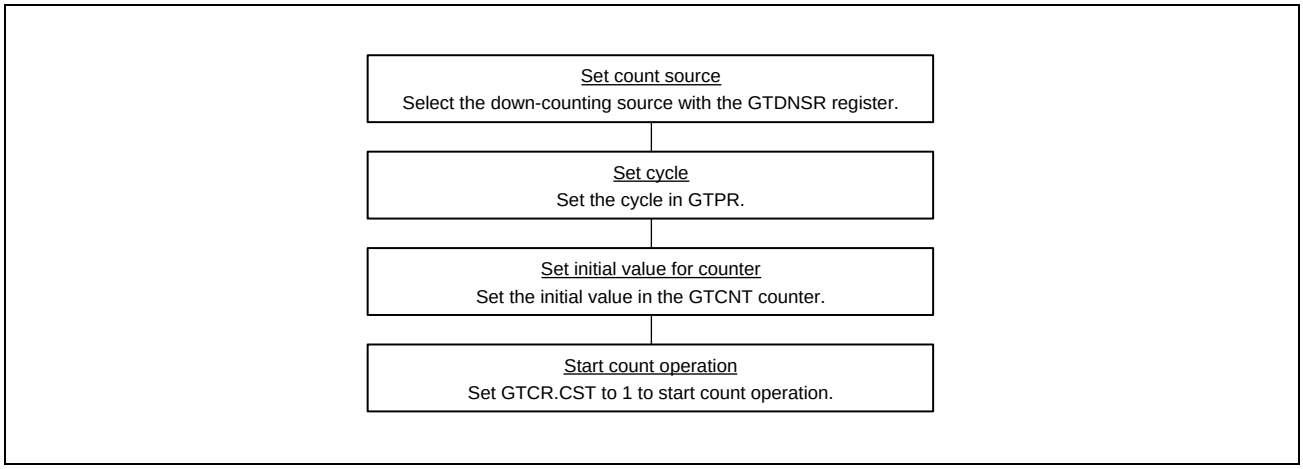


Figure 18.9 Example setting for an event count operation in down-counting using hardware sources

(6) Counter clear operation

The counter of each channel is cleared by either of the following sources.

- Writing 0 to GTCNT register
- Writing 1 to the bit in GTCLR associated with the GPT channel number when the GTCSR.CCLR bit set to 1
- The hardware source selected in GTCSR register.

Writing to the GTCNT register is prohibited during count operation. The GTCNT counter can be cleared both by writing 1 to the GTCLR and by the clear request of hardware sources, whether GTCNT is counting (GTCR.CST = 1) or not (GTCR.CST = 0).

For saw waves selected by setting GTCR.MD[2:0] and the count direction flag showing down-counting (GTST.TUCF = 0), the GTCNT register is set to the value of the GTPR register when the counter is cleared by writing 1 to the GTCLR register or by the hardware source specified in the GTCSR register. When not in saw waves mode and down-counting, the GTCNT register is set to 0 when writing 1 to the GTCLR register and when clearing by hardware sources are performed.

In event count operation when at least 1 bit in GTUPSR or GTDNSR is set to 1, after clear sources occur, both writing to GTCLR register and clearing by hardware sources are performed immediately to synchronize with P0ϕ. If other settings are used, clear is synchronized with the counter clock selected by GTCR.TPCS[2:0].

18.3.1.2 Waveform Output by Compare Match

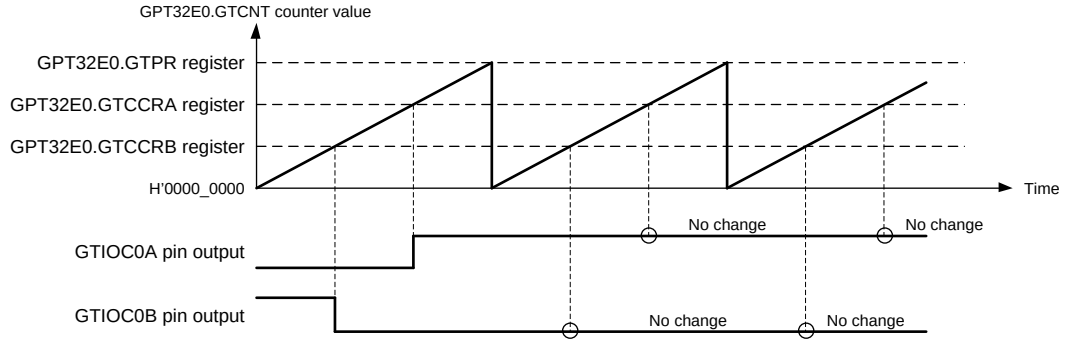
Compare match means that the GTCNT counter value matches the value of GTCCRA or GTCCRB. When a compare match occurs, the compare match flag is generated synchronously with the count clock including the event count. At the same time the GPT can output low, high, or toggle output from the associated GTIOCA or GTIOCB output pin. In addition, the GTIOCA or GTIOCB pin output can be low, high, or toggle at the cycle end which is determined by GTPR. The cycle end is:

- For saw waves in up-counting – when GTCNT changes from the GTPR value to 0 (overflow)
- For saw waves in down-counting – when GTCNT changes from 0 to GTPR value (underflow)
- For saw waves – when the GTCNT counter is cleared
- For triangle waves – when the GTCNT changes from 0 to 1 (trough).

(1) Low and high output

Figure 18.10 shows an example of low and high output operation by a compare match of GTCCRA and GTCCRB.

In this example, the GPT32E0.GTCNT counter performs up-counting, and settings are made so that high is output from the GTIOC0A pin by a GPT32E0.GTCCRA compare match, and low is output from the GTOC0B pin by a GPT32E0.GTCCRB compare match. The pin level does not change when the specified level and pin level match.



[Setting examples]

GPT32E0.GTIOR.GTIOA[4:0] bits: Initial output is low, high output at compare match, output retained at cycle end

GPT32E0.GTIOR.GTIOB[4:0] bits: Initial output is high, low output at compare match, output retained at cycle end

Figure 18.10 Example of low and high output operation

Figure 18.11 shows an example setting for low output and high output operation.

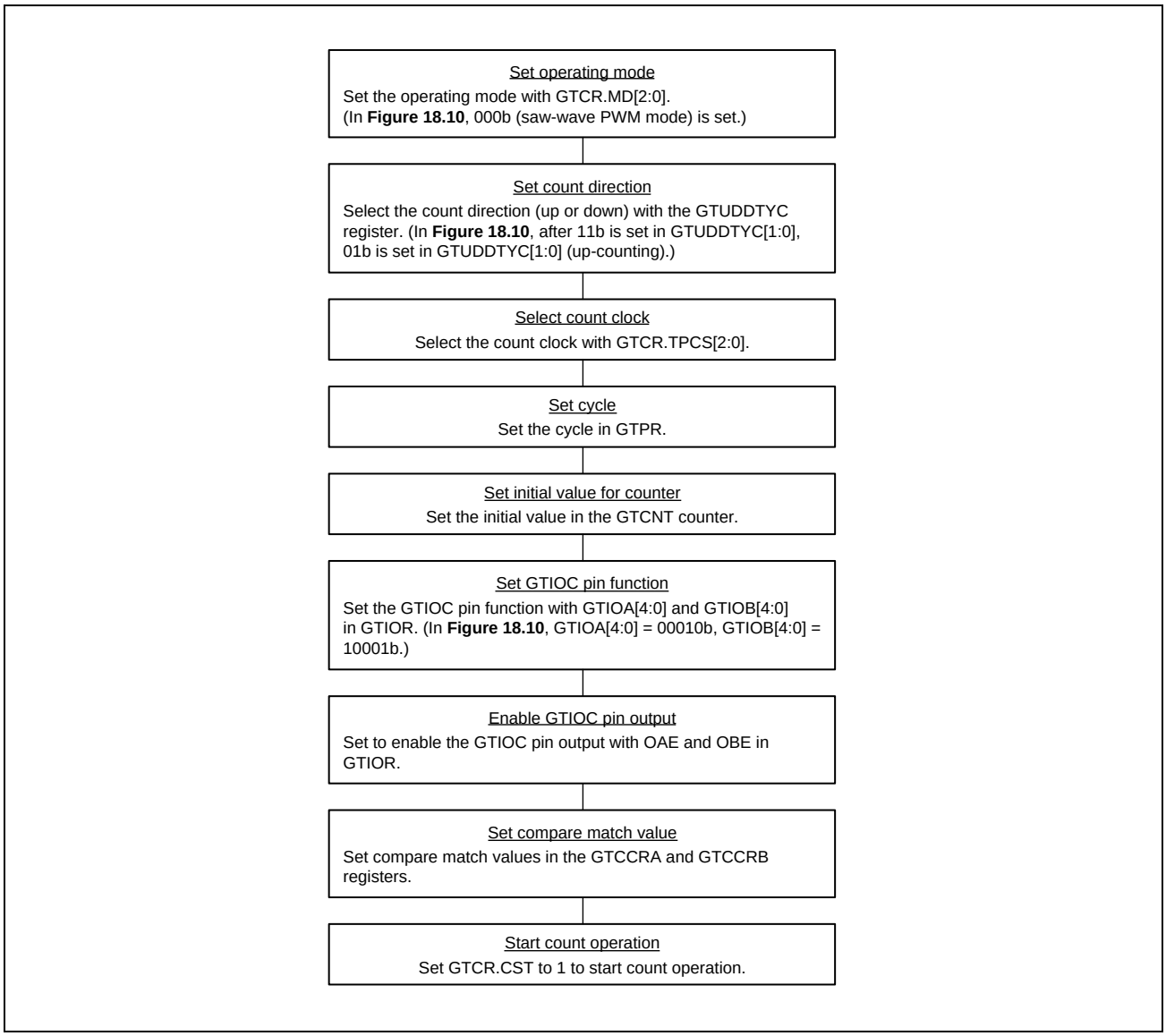
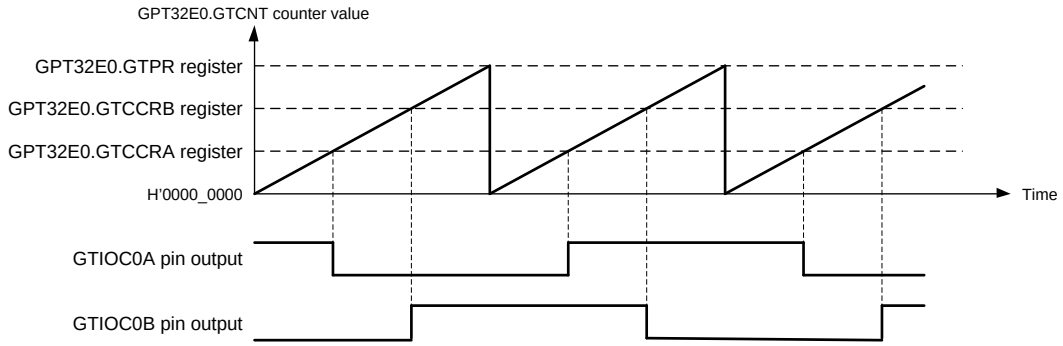


Figure 18.11 Example setting for low output and high output operation

(2) Toggled output

Figure 18.12 and **Figure 18.13** show examples of toggled output operation by compare matches of GTCCRA and GTCCRB. In **Figure 18.12**, the GPT32E0.GTCNT counter performs up-counting, and settings are made so that the GTIOC0A pin output by a GPT32E0.GTCCRA compare match and GTIOC0B pin output by a GPT32E0.GTCCRB compare match are toggled.

Figure 18.13, the GPT32E0.GTCNT counter performs up-counting, and settings are made so that the GTIOC0A output is toggled by a compare match of GPT32E0.GTCCRA and the GTIOC0B output is toggled at the cycle end.

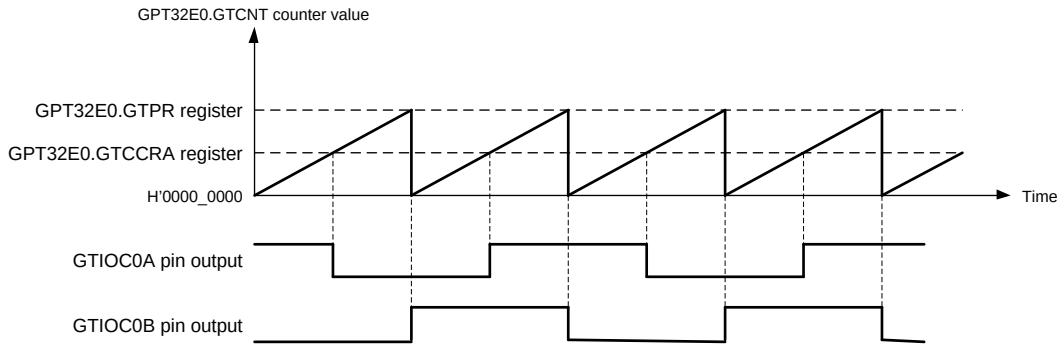


[Setting examples]

GPT32E0.GTIOR.GTIOA[4:0] bits: Initial output is high, output toggled at compare match, output retained at cycle end

GPT32E0.GTIOR.GTIOB[4:0] bits: Initial output is low, output toggled at compare match, output retained at cycle end

Figure 18.12 Example of toggled output operation (1)



[Setting examples]

GPT32E0.GTIOR.GTIOA[4:0] bits: Initial output is high, output toggled at compare match, output retained at cycle end

GPT32E0.GTIOR.GTIOB[4:0] bits: Initial output is low, output retained at compare match, output toggled at cycle end

Figure 18.13 Example of toggled output operation (2)

Figure 18.14 shows an example setting for toggled output operation.

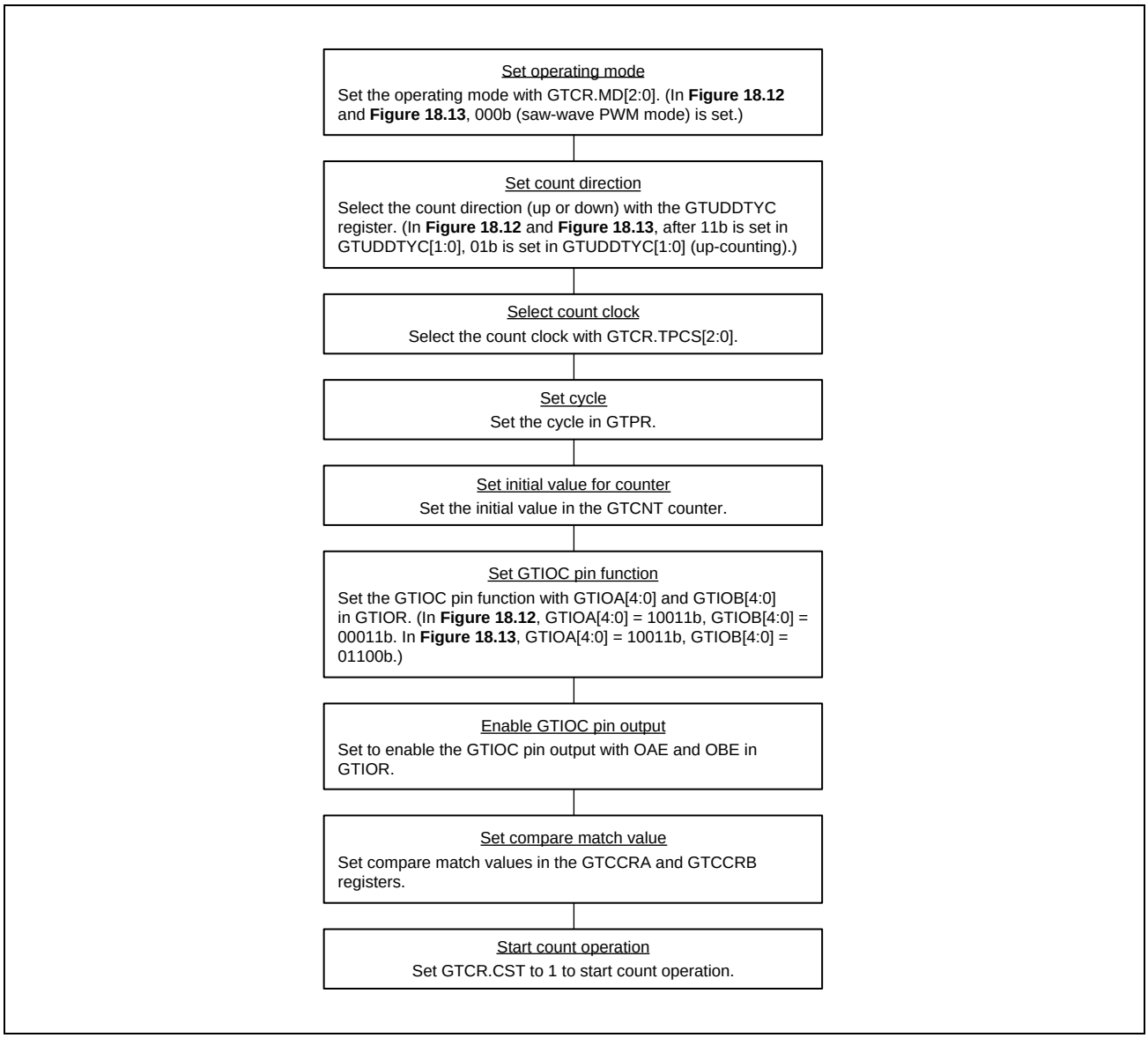


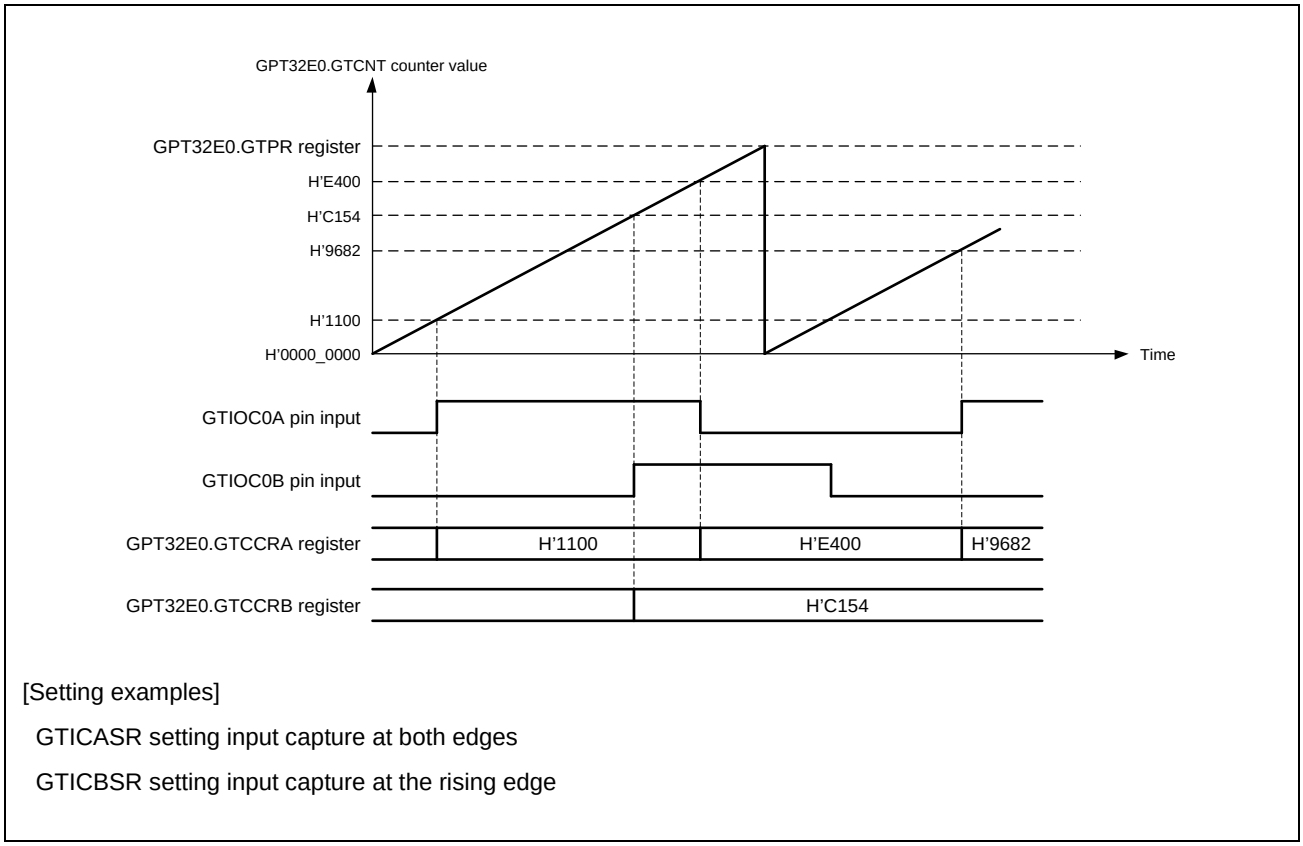
Figure 18.14 Example setting for toggled output operation

18.3.1.3 Input Capture Function

The GTCNT counter value can be transferred to either GTCCRA or GTCCRB on detection of the hardware source that is set in GTICASR and GTICBSR.

Figure 18.15 shows an example of the input capture function.

In this example, the GPT32E0.GTCNT counter performs up-counting by the count clock, and settings are made so that an input capture is performed to GTICCRA at both edges of the GTIOC0A input pin and to GTICCRB on the rising edge of the GTIOC0B input pin.



[Setting examples]

GTICASR setting input capture at both edges

GTICBSR setting input capture at the rising edge

Figure 18.15 Example of input capture operation

Figure 18.16 shows an example setting for an input capture operation with count operation by the count clock.

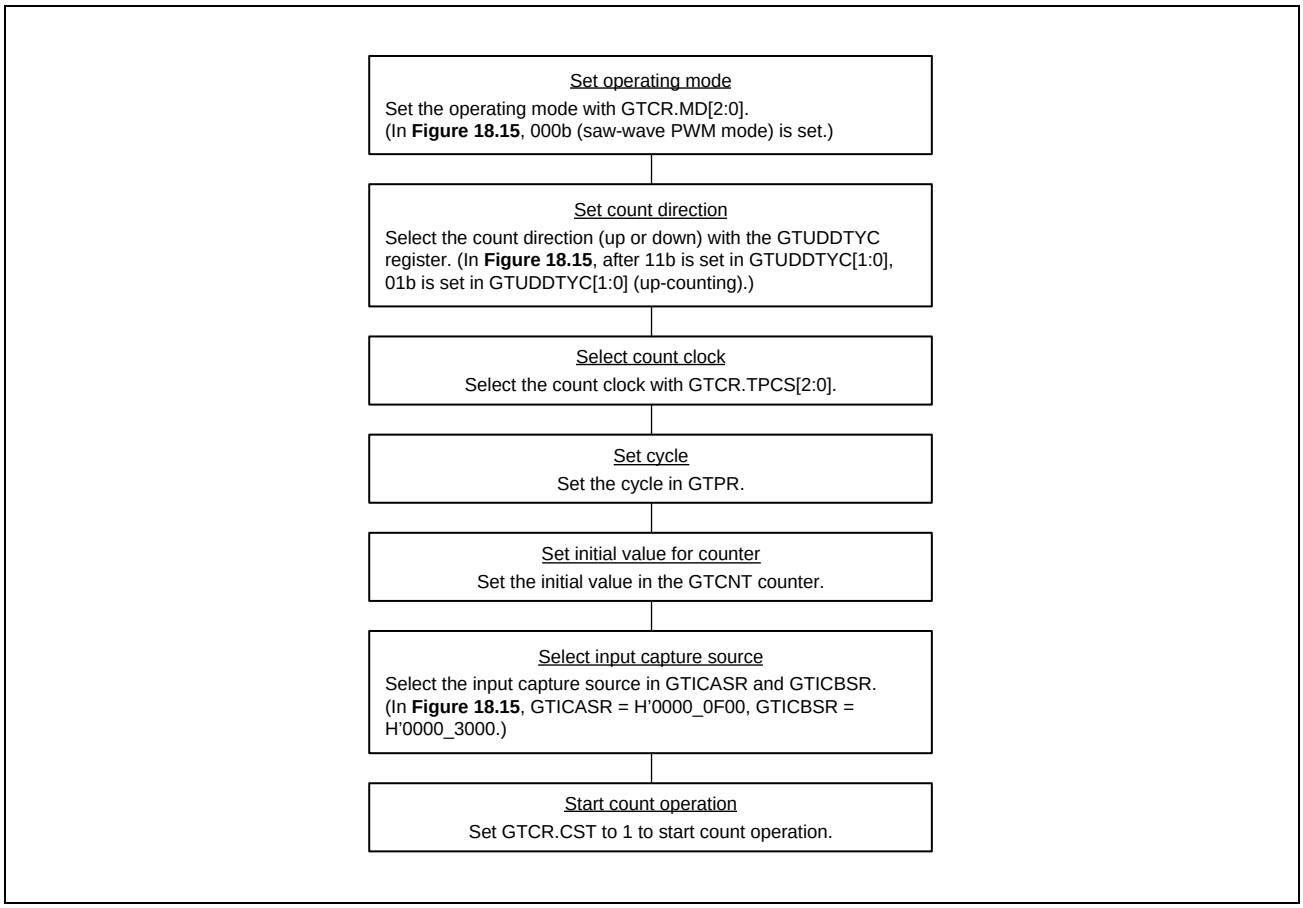


Figure 18.16 Example setting for input capture operation

18.3.2 Buffer Operation

The following buffer operations can be set with GTBER:

- GTPR, GTPBR, and GTPDBR
- GTCCRA, GTCCRC, and GTCCRD
- GTCCRB, GTCCRE, and GTCCRF
- GTADTRA, GTADTBRA, and GTADTDBRA
- GTADTRB, GTADTBRB, and GTADTDBRB.

The following buffer operations can be set with GTDTCR:

- GTDVU and GTDBU
- GTDVD and GTDBD.

18.3.2.1 GTPR Register Buffer Operation

GTPBR can function as a buffer register for GTPR, and GTPDBR can function as a buffer register for GTPBR (double-buffer register for GTPR). The buffer transfer is performed at an overflow during up-counting or an underflow during down-counting in saw-wave mode or in event count, and at a trough in triangle-wave mode.

In saw-wave mode or in event count, the buffer transfer is performed when the following counter clear operations occur during counting:

- Clear by hardware sources (the clear source is selected in GTCSR[23:0])
- Clear by software (when GTCSR.CCLR bit is 1 and GTCLR[n] bit is set to 1, n = channel number).

To set GTPR to function as double buffer, set GTBER.PR[1:0] to 10b or 11b. To set GTPR to not function as a buffer, set GTBER.PR[1:0] to 00b.

Figure 18.17 to **Figure 18.19** show examples of GTPR buffer operation and **Figure 18.20** shows an example setting for GTPR buffer operation.

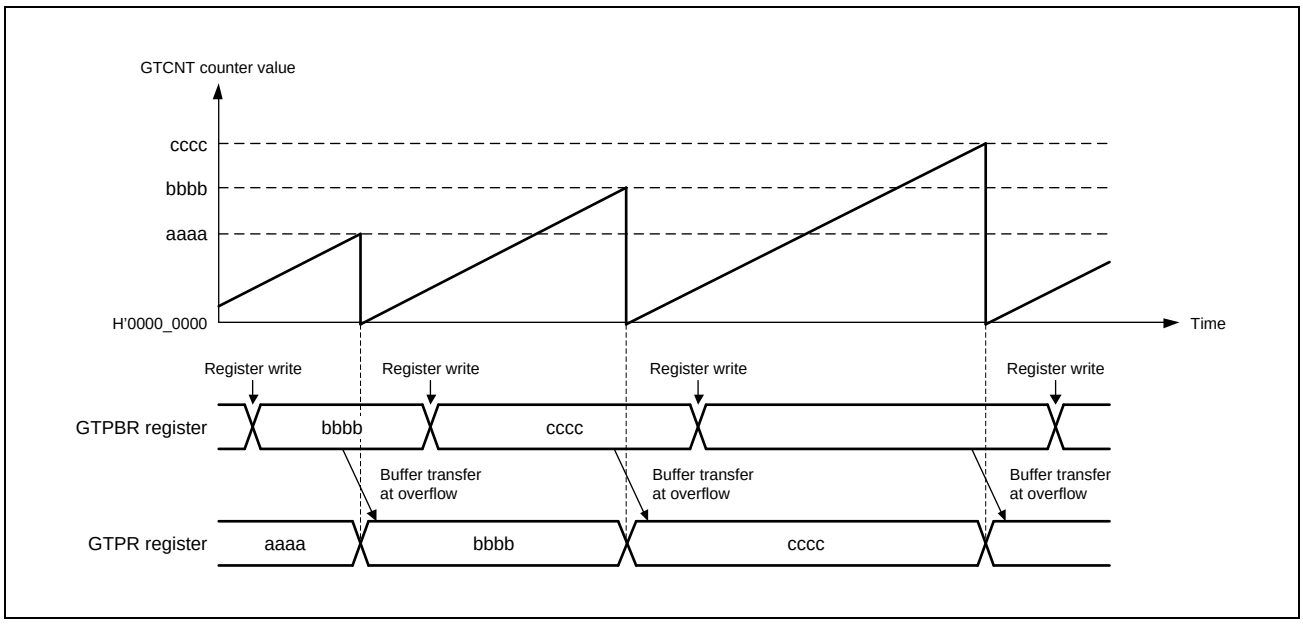


Figure 18.17 Example of GTPR buffer operation with saw waves in up-counting

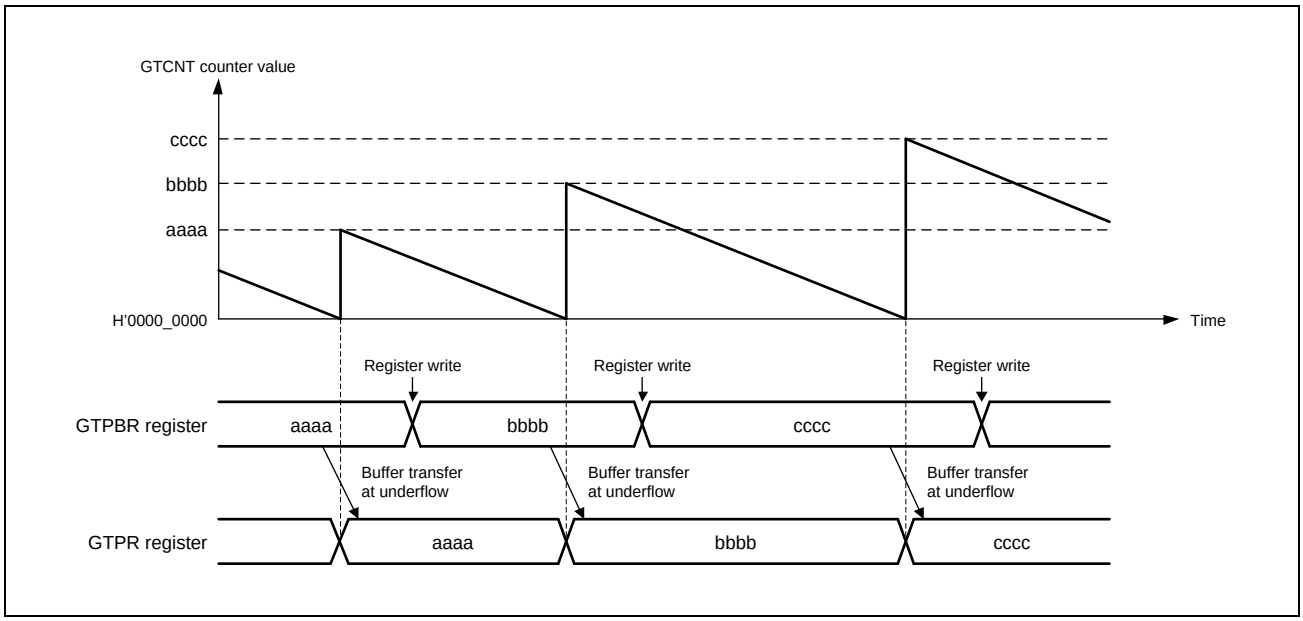


Figure 18.18 Example of GTPR buffer operation with saw waves in down-counting

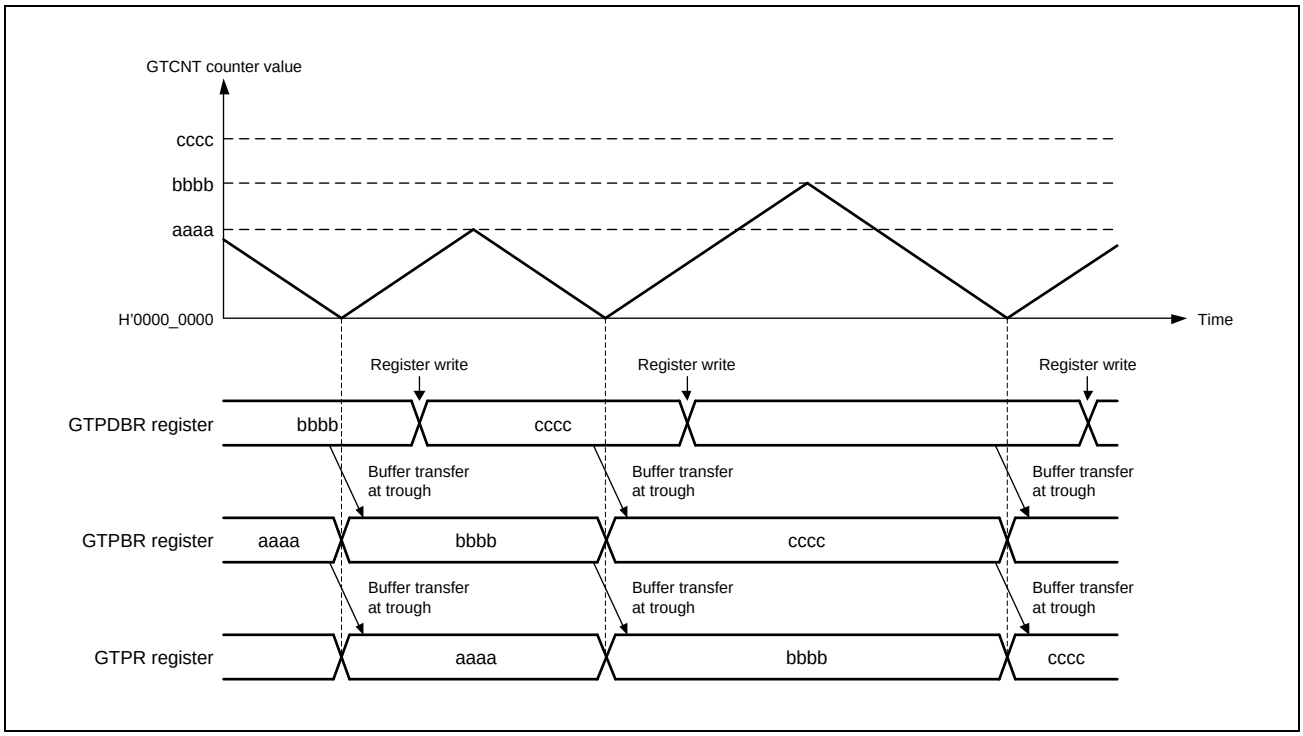


Figure 18.19 Example of GTPR double buffer operation with triangle waves

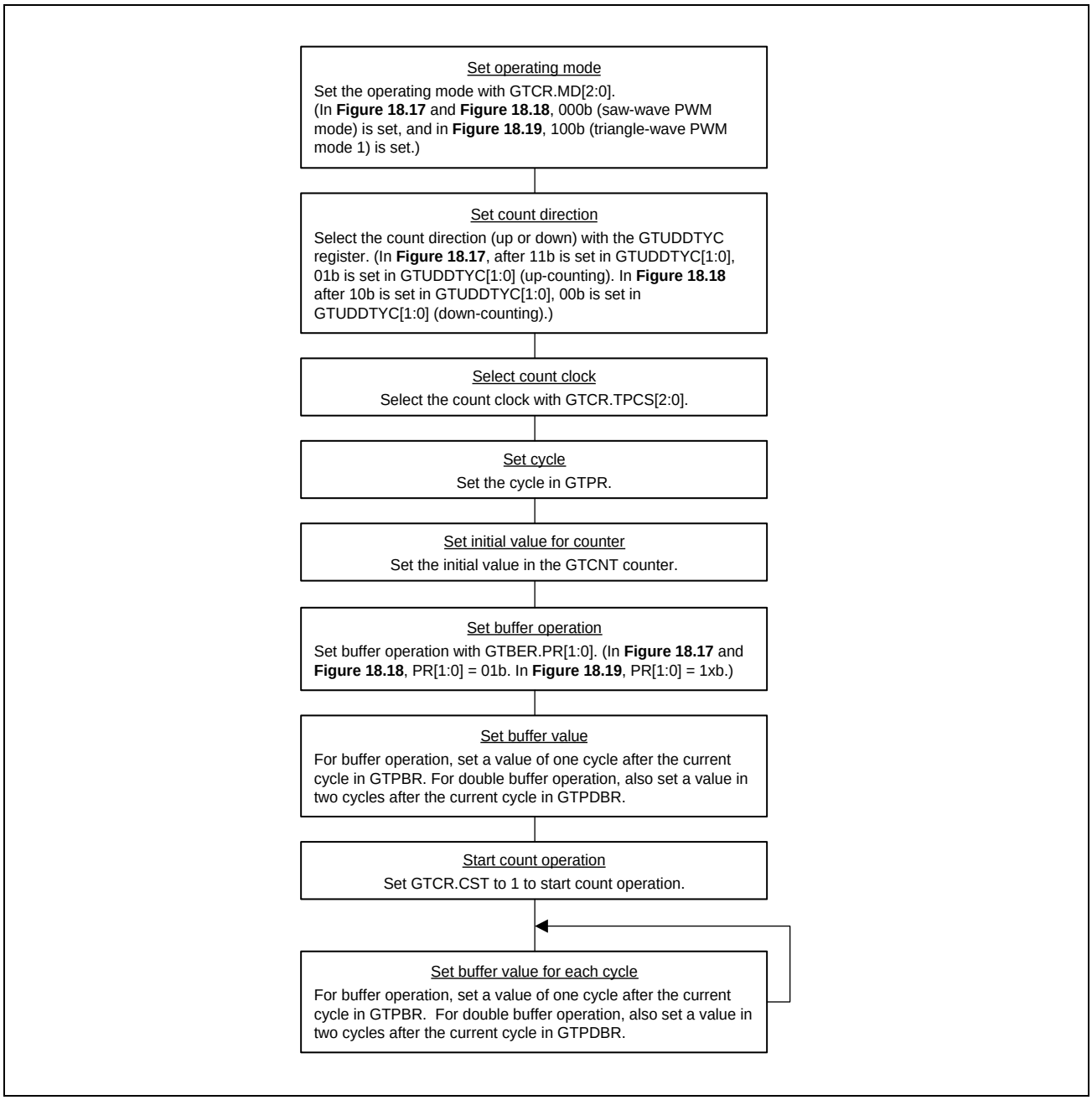


Figure 18.20 Example setting for GTPR buffer operation

18.3.2.2 Buffer Operation for GTCCRA and GTCCRB

GTCCRC can function as the GTCCRA buffer register and GTCCRD can function as the GTCCRC buffer register (double-buffer register for GTCCRA). Similarly, GTCCRE can function as the GTCCRB buffer register and GTCCRF can function as the GTCCRE buffer register (double-buffer register for GTCCRB).

To set GTCCRA or GTCCRB to function as a double buffer, set GTBER.CCRA[1:0] or GTBER.CCRB[1:0] to 10b or 11b. For single buffer operation, set GTBER.CCRA[1:0] or GTBER.CCRB[1:0] to 01b. To set GTCCRA or GTCCRB to not function as a buffer, set GTBER.CCRA[1:0] or GTBER.CCRB[1:0] to 00b.

(1) When GTCCRA or GTCCRB functions as an output compare register

Buffer transfer occurs in the following situations:

- Buffer transfer by overflow or underflow
  Buffer transfer is performed at an overflow (during up-counting) or an underflow (during down-counting) in saw-wave mode or in event count operation. In triangle-wave mode, buffer transfer is performed at a trough (triangle-wave PWM mode 1) or a crest and trough (triangle-wave PWM mode 2).
- Buffer transfer by counter clear
  In saw-wave mode or in event count operation, during counting, buffer transfer (which is the same as an overflow during up-counting or an underflow during down-counting) is performed by the counter clear sources the same as shown in **Section 18.3.2.1, GTPR Register Buffer Operation**. In triangle-wave mode, buffer transfer is not performed by the counter clear.
- Forcible buffer transfer
  When 1 is written to the GTBER.CCRSWT bit while counting is stopped, the GTCCRA and the GTCCRB register buffer transfer are performed forcibly in saw-wave mode, in event count operation and in triangle-wave mode. Additionally, buffer transfer from the GTCCRD register to temporary register A and from the GTCCRF register to temporary register B are performed in saw-wave 1 shot pulse mode or triangle-wave PWM mode 3.

Figure 18.21 to **Figure 18.23** show examples of GTCCRA and GTCCRB buffer operation and **Figure 18.24** shows an example setting for GTCCRA and GTCCRB buffer operation.

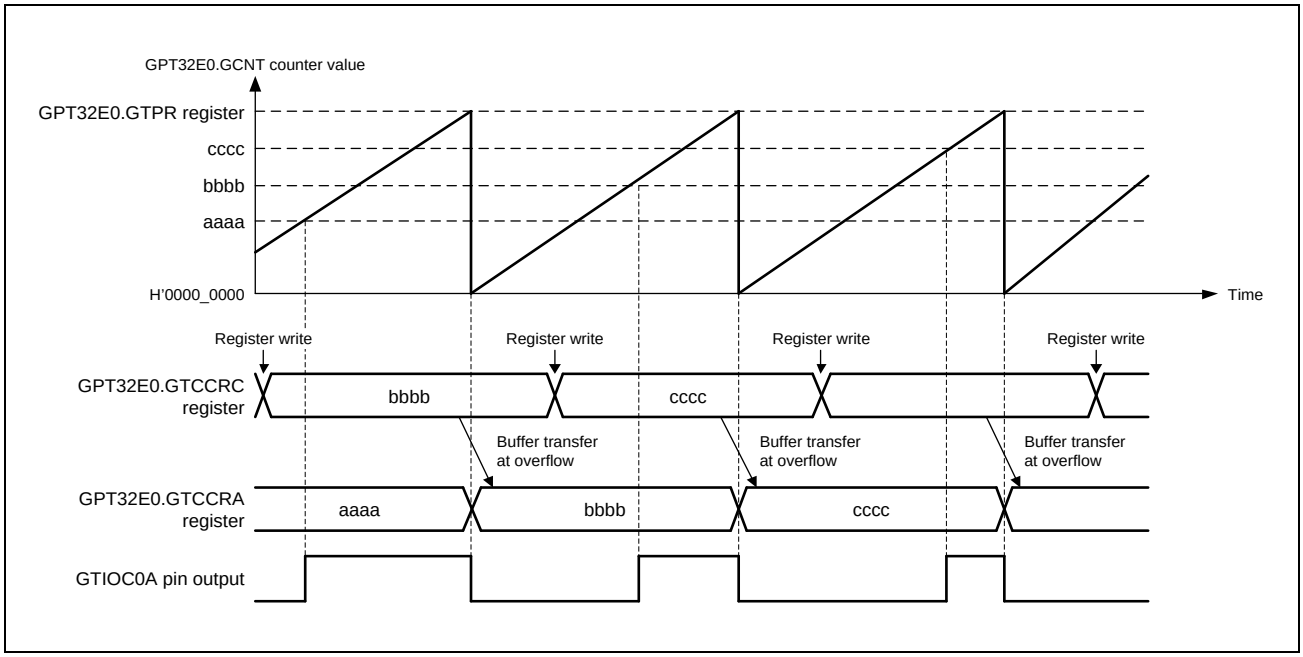


Figure 18.21 Example of GTCCRA and GTCCRB buffer operation with output compare, saw waves in up-counting, high output at GTCCRA compare match, and low output at cycle end

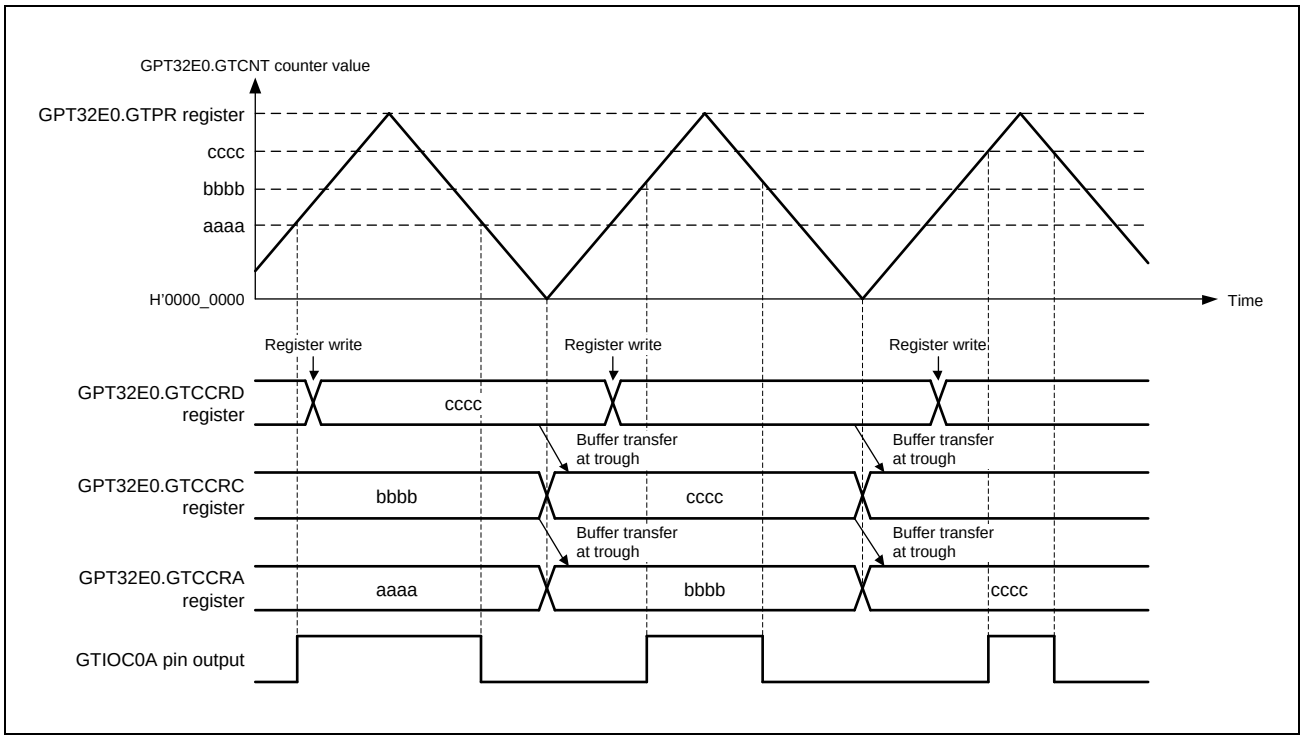


Figure 18.22 Example of GTCCRA and GTCCRB double buffer operation with output compare, triangle waves, buffer operation at trough, output toggled at GTCCRA compare match, and output retained at cycle end

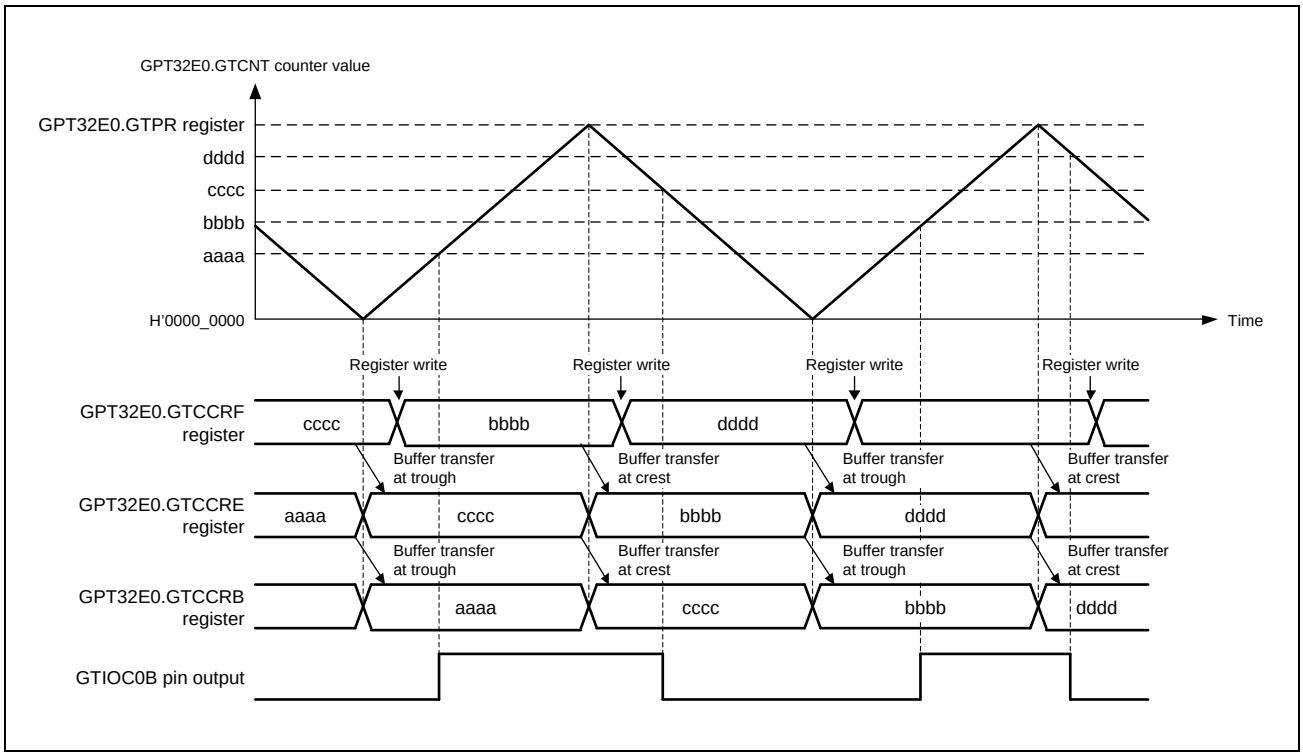


Figure 18.23 Example of GTCCRA and GTCCRB double buffer operation with output compare, triangle waves, buffer operation at both troughs and crests, output toggled at GTCCRB compare match, and output retained at cycle end

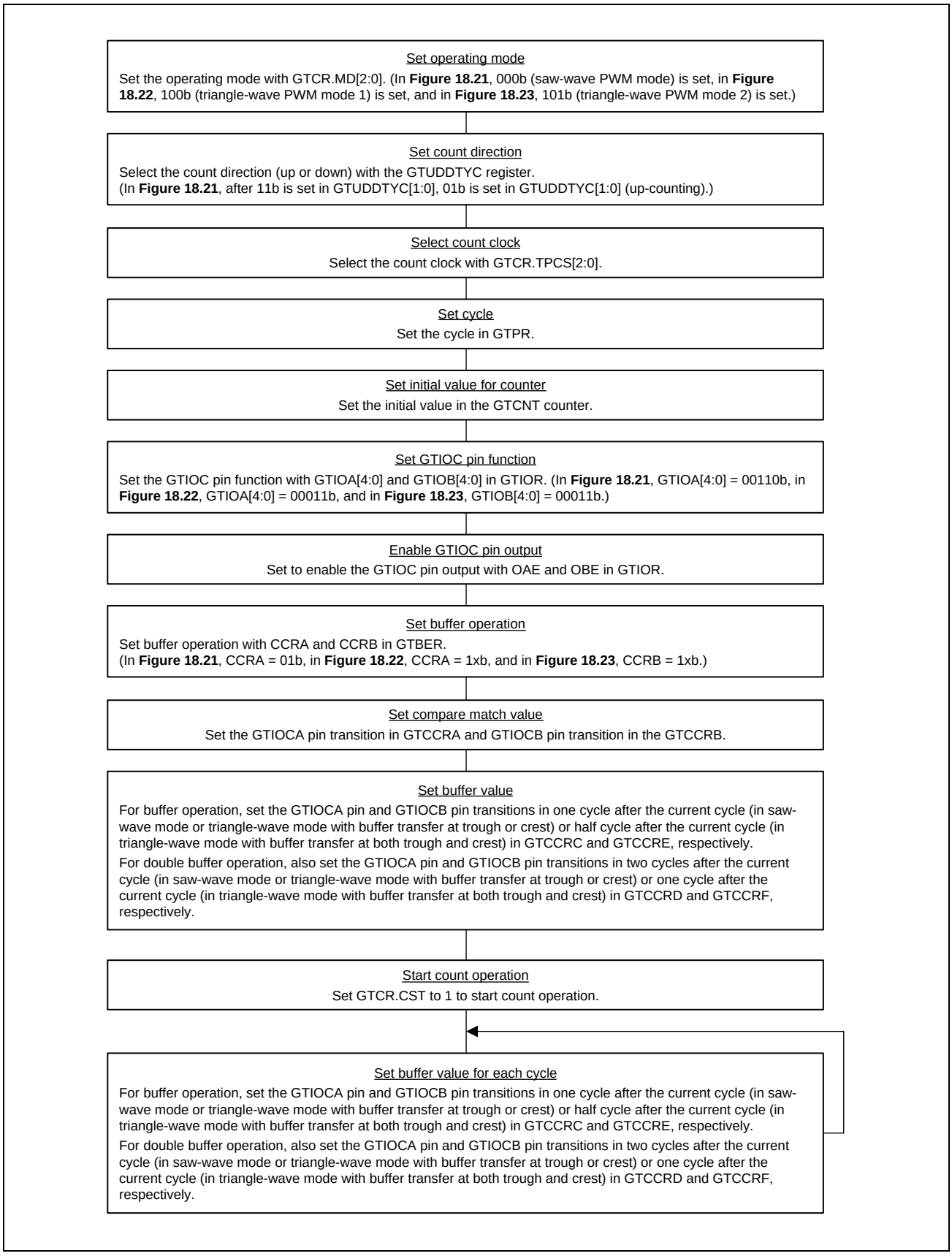


Figure 18.24 Example setting for GTCCRA and GTCCRB buffer operation with output compare

(2) When GTCCRA or GTCCRB functions as an input capture register

When an input capture is generated, the GTCNT counter value is transferred to GTCCRA and GTCCRB and the stored GTCCRA and GTCCRB register values are transferred to the buffer registers. In input capture operation, the buffer transfer is not performed by the counter clear.

Figure 18.25 and **Figure 18.26** show examples of GTCCRA and GTCCRB buffer operation and **Figure 18.27** shows an example setting for GTCCRA and GTCCRB buffer operation.

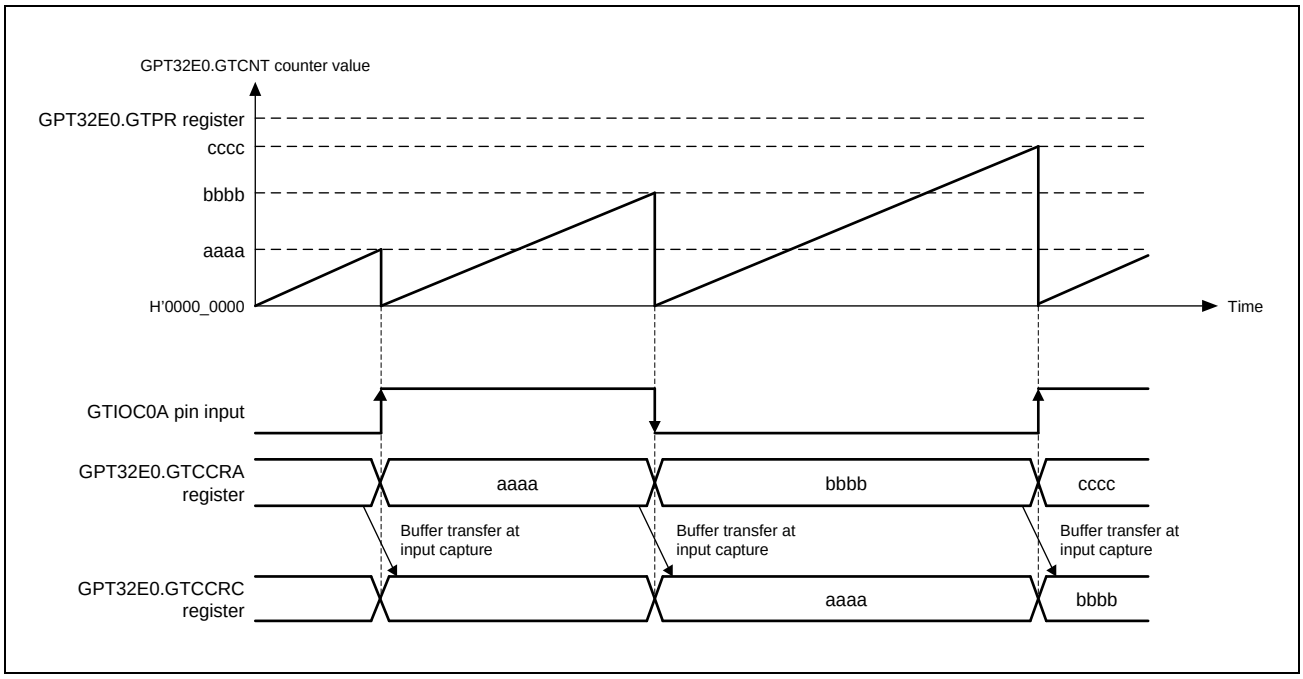


Figure 18.25 Example of GTCCRA and GTCCRB buffer operation with input capture at both edges of GTIOC0A input, saw waves in up-counting, and GTCNT counter cleared at both edges of GTIOC0A input

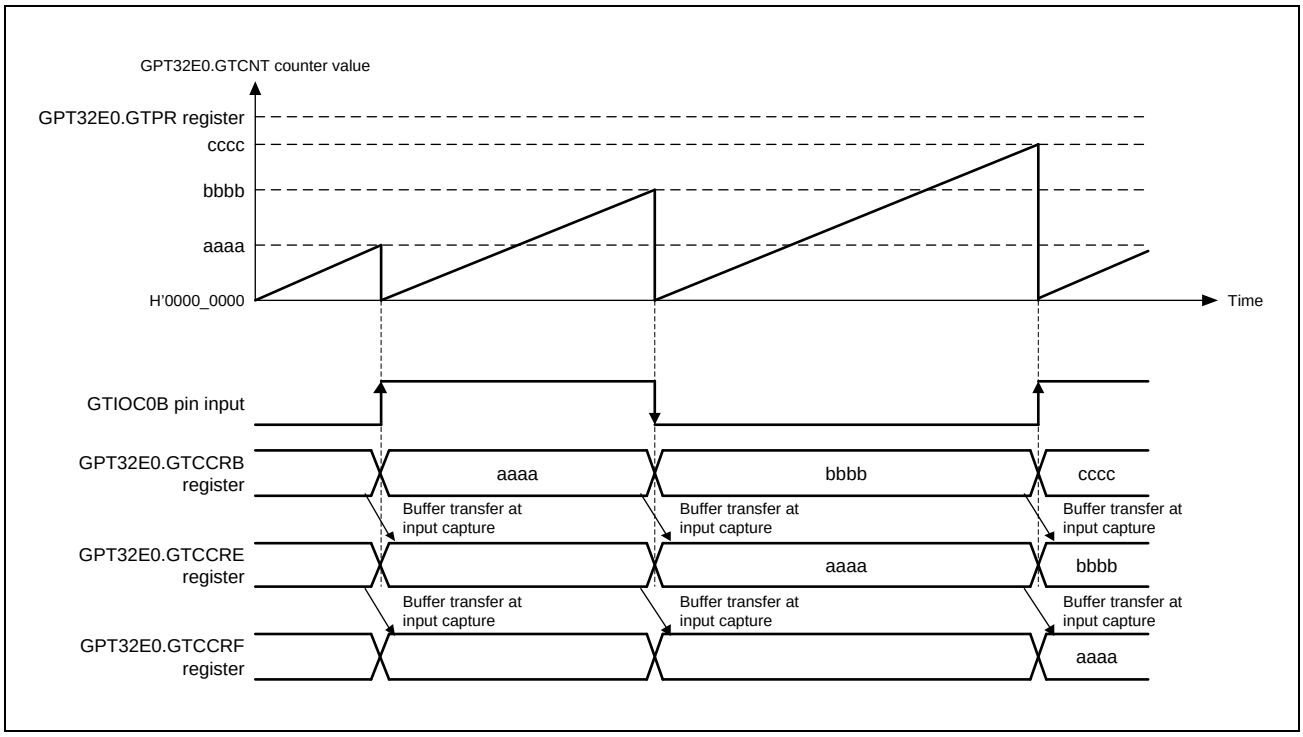


Figure 18.26 Example of GTCCRA and GTCCRB double buffer operation with input capture at both edges of GTIOC0B input, saw waves in up-counting, and GTCNT counter cleared at both edges of GTIOC0B input

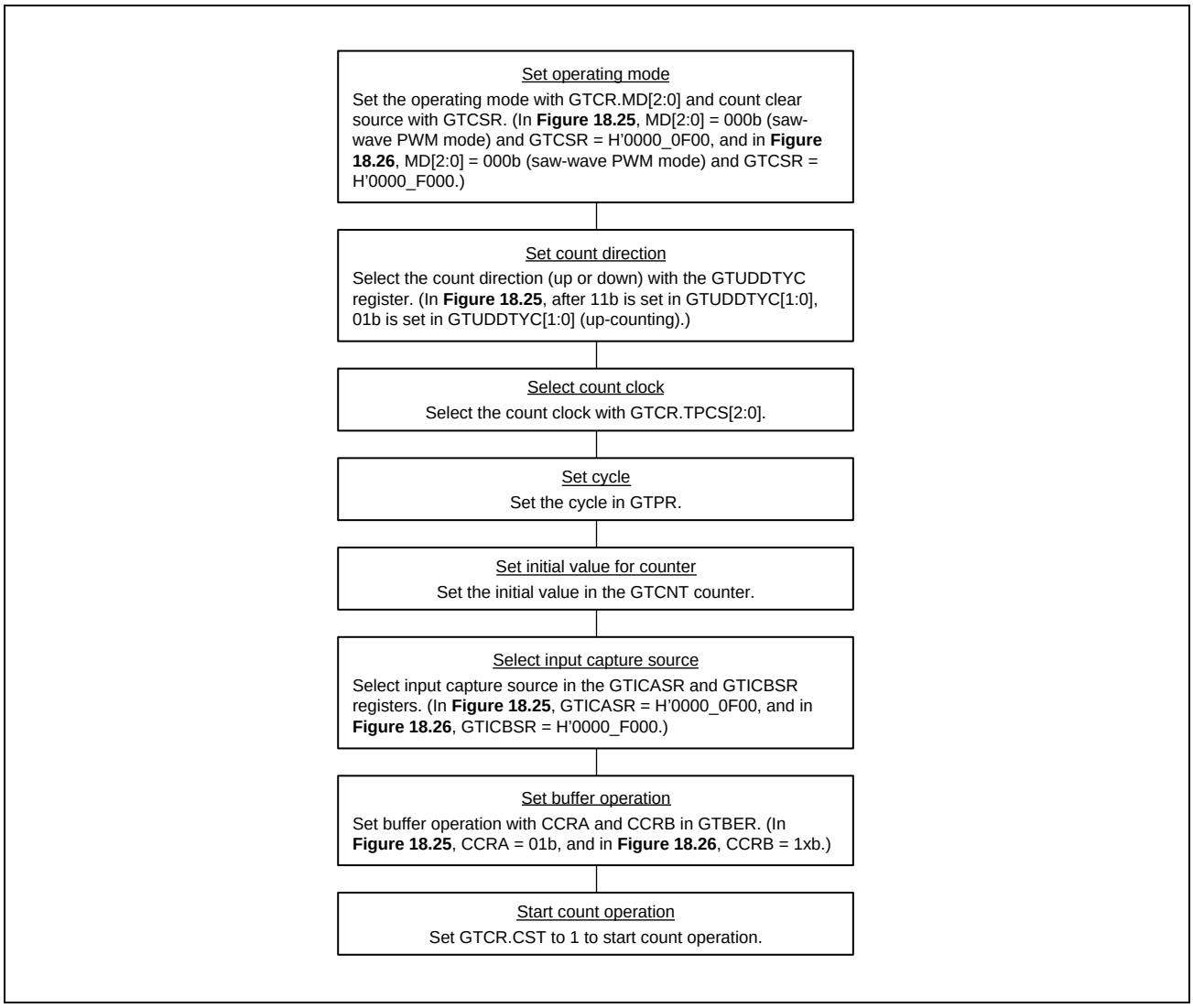


Figure 18.27 Example setting for GTCCRA and GTCCRB buffer operation with input capture

18.3.2.3 Buffer Operation for GTADTRA and GTADTRB

GTADTBRA can function as the GTADTRA buffer register and GTADTDBRA can function as the GTADTBRA buffer register (double-buffer register for GTADTRA). Similarly, GTADTBRB can function as the GTADTRB buffer register and GTADTDBRB can function as the GTADTBRB buffer register (double-buffer register for GTADTRB).

To set GTADTRA or GTADTRB to function as a double buffer, set GTBER.ADTDA or GTBER.ADTDB to 1. For single buffer operation, set GTBER.ADTDA or GTBER.ADTDB to 0. To set GTADTRA or GTADTRB to not function as a buffer, set GTBER.ADTTA[1:0] or GTBER.ADTTB[1:0] to 00b.

The buffer transfer timing can be set with the GTBER.ADTTA[1:0] bits. For saw waves, overflows (during up-counting) or underflows (during down-counting) can be selected. For triangle waves, crests are selected when GTBER.ADTTA[1:0] = 01b, troughs are selected when GTBER.ADTTA[1:0] = 10b, and both crests and troughs are selected when GTBER.ADTTA[1:0] = 11b.

Figure 18.28 to **Figure 18.30** show examples of GTADTRA and GTADTRB buffer operation and **Figure 18.31** shows an example setting for GTDTRA and GTADTRB buffer operation.

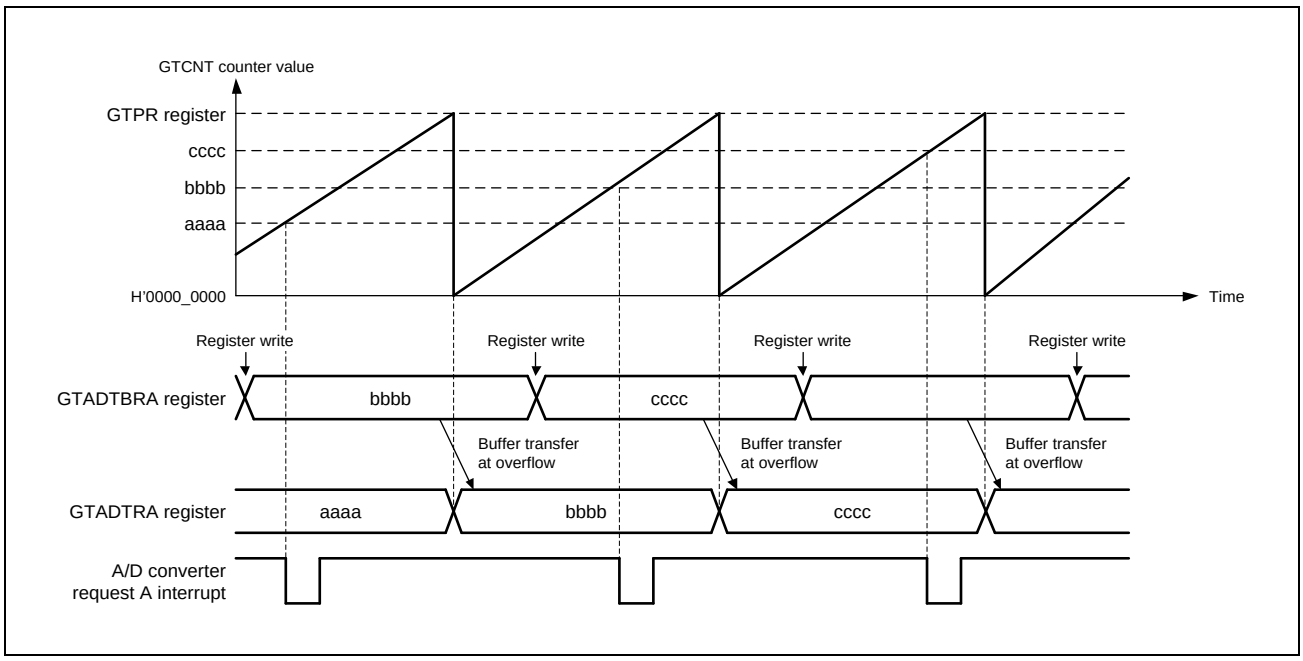


Figure 18.28 Example of GTADTRA and GTADTRB buffer operation with saw waves in up-counting and A/D converter start request interrupt generated by up-counting

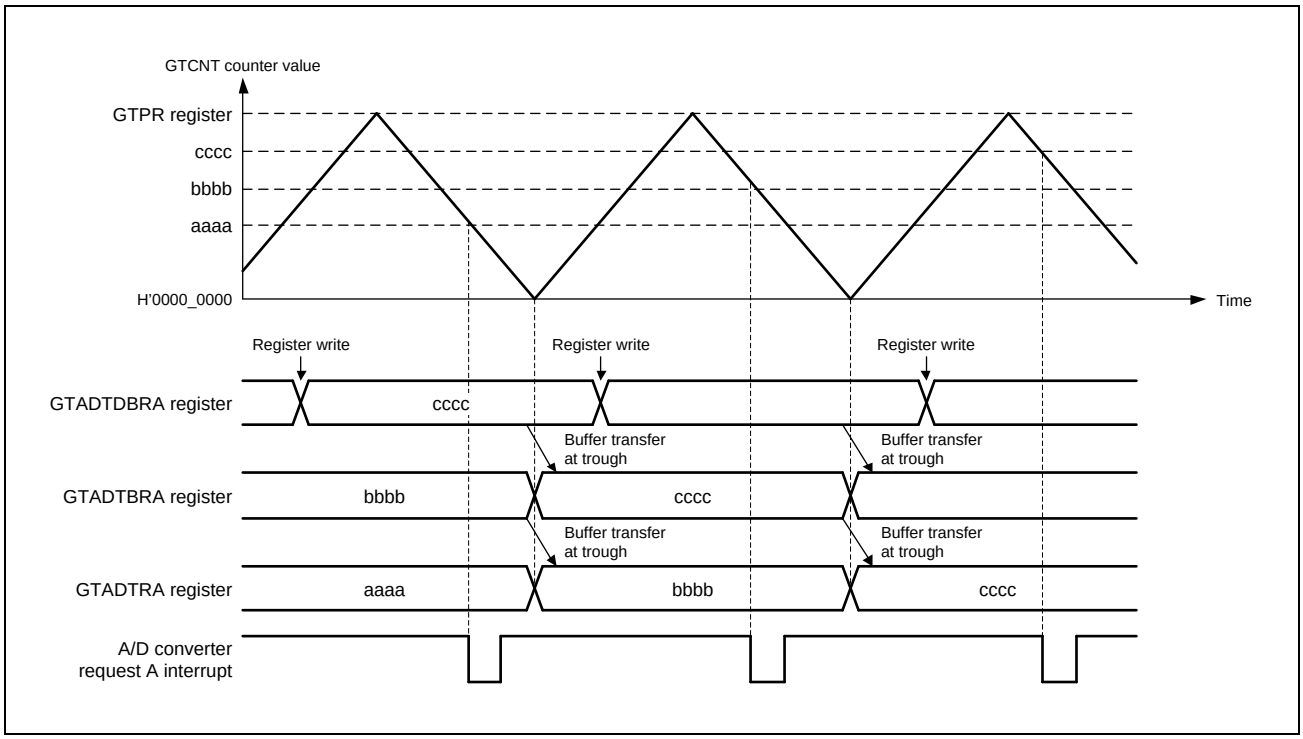


Figure 18.29 Example of GTADTRA and GTADTRB double buffer operation with triangle waves, buffer transfer at troughs, and A/D converter start request interrupt generated by down-counting

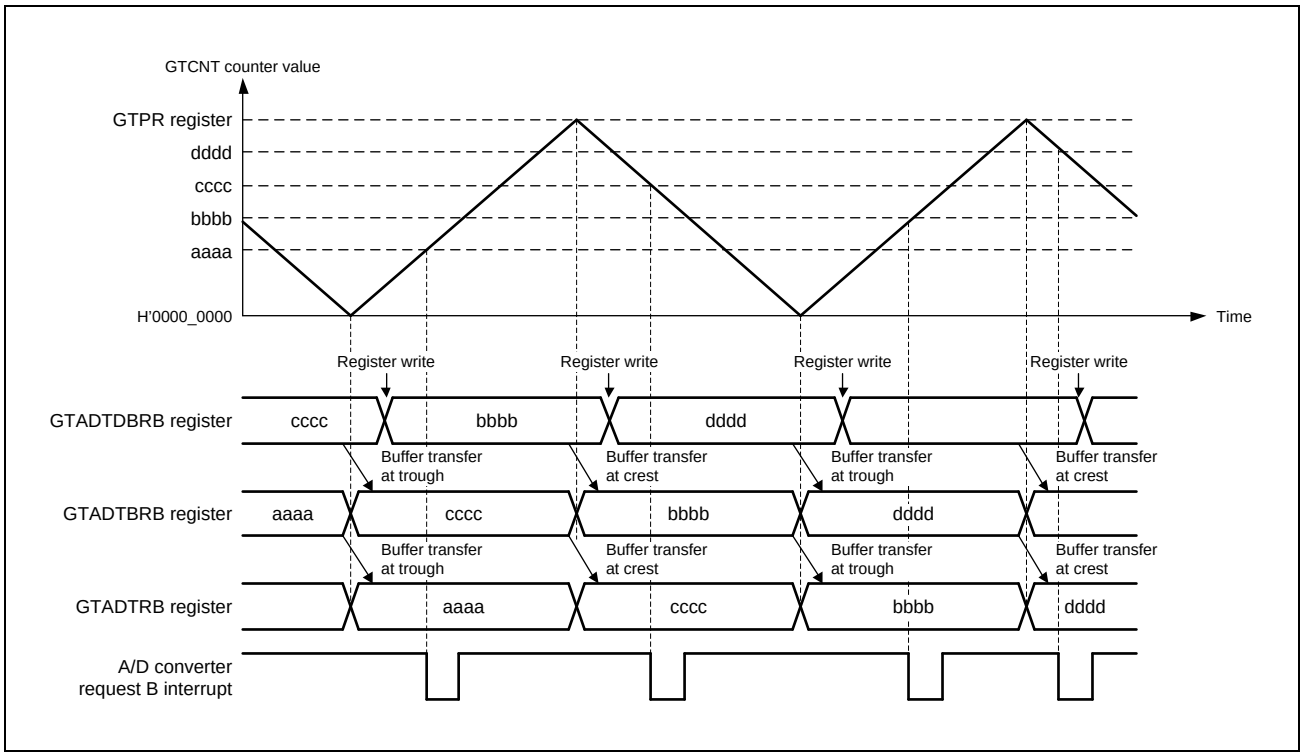


Figure 18.30 Example of GTADTRA and GTADTRB double buffer operation with triangle waves, buffer transfer at both troughs and crests, and A/D converter start request interrupt generated by both up- and down- counting

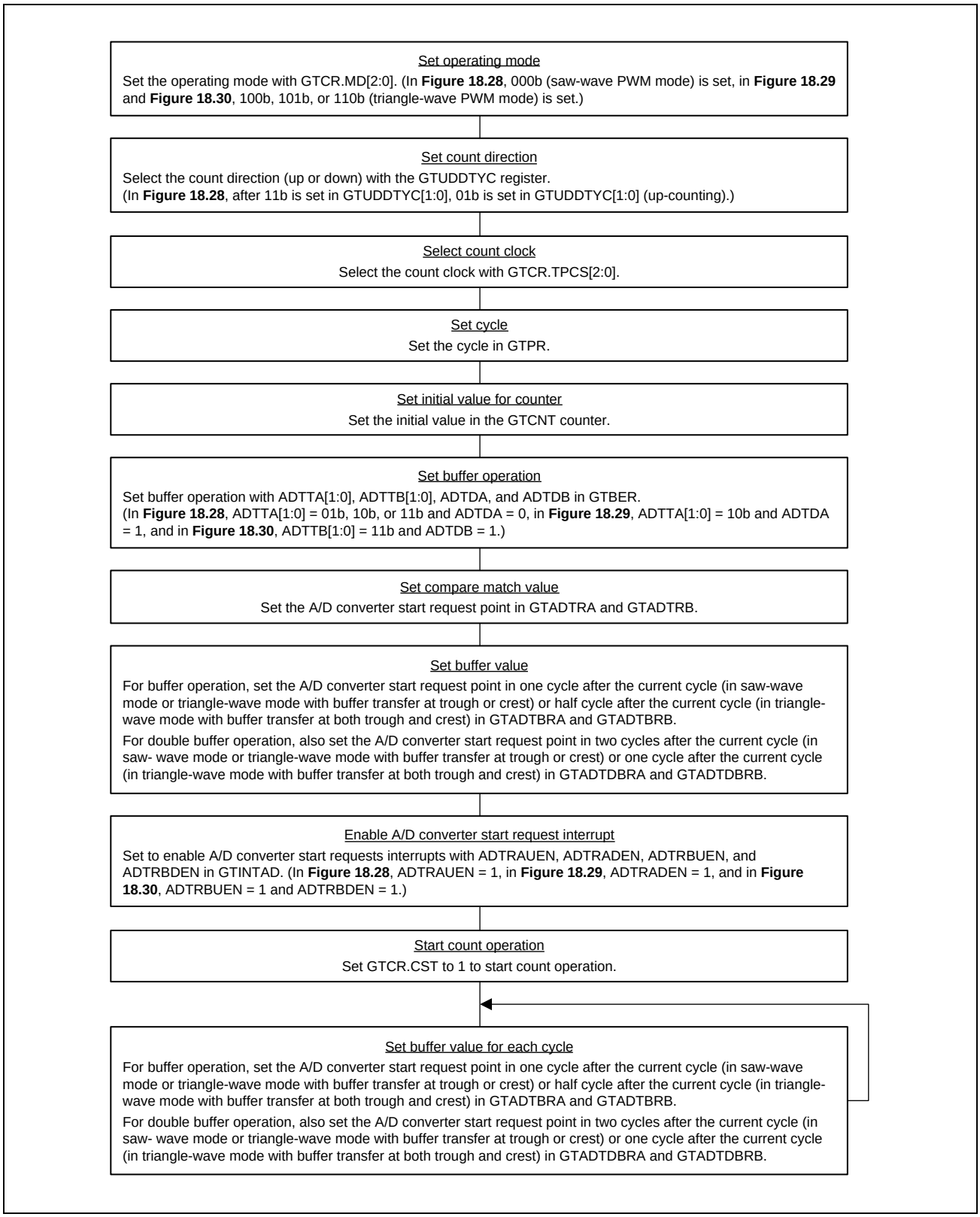


Figure 18.31 Example setting for GTADTRA and GTADTRB buffer operation

18.3.3 PWM Output Operating Mode

The GPT can output PWM waveforms to the GTIOCA or GTIOCB pin by a compare match between the GTCNT counter and GTCCRA or GTCCRB.

By setting GTDTCR, GTDVU, and GTDVD, the compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

18.3.3.1 Saw-Wave PWM Mode

In saw-wave PWM mode, GTCNT performs saw-wave (half-wave) operation by setting the cycle in GTPR.A PWM waveform is output to the GTIOCA or GTIOCB pin when a GTCCRA or GTCCRB compare match occurs. The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the cycle end based on the GTIOR setting.

Figure 18.32 shows an example of saw-wave PWM mode operation, and **Figure 18.33** shows an example setting for saw-wave PWM mode.

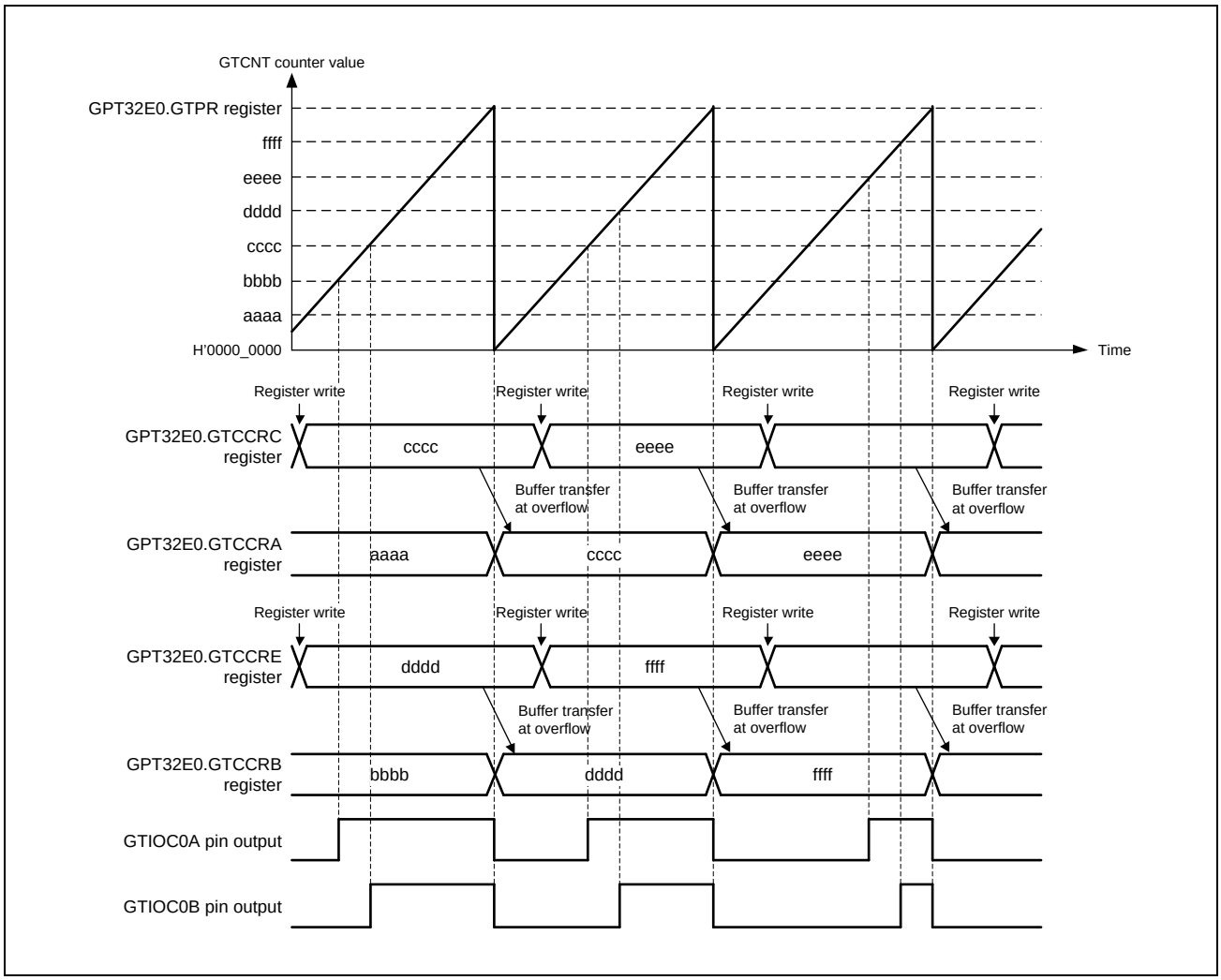


Figure 18.32 Example of saw-wave PWM mode operation with up-counting, buffer operation, high output at GTCCRA/GTCCRB compare match, and low output at cycle end

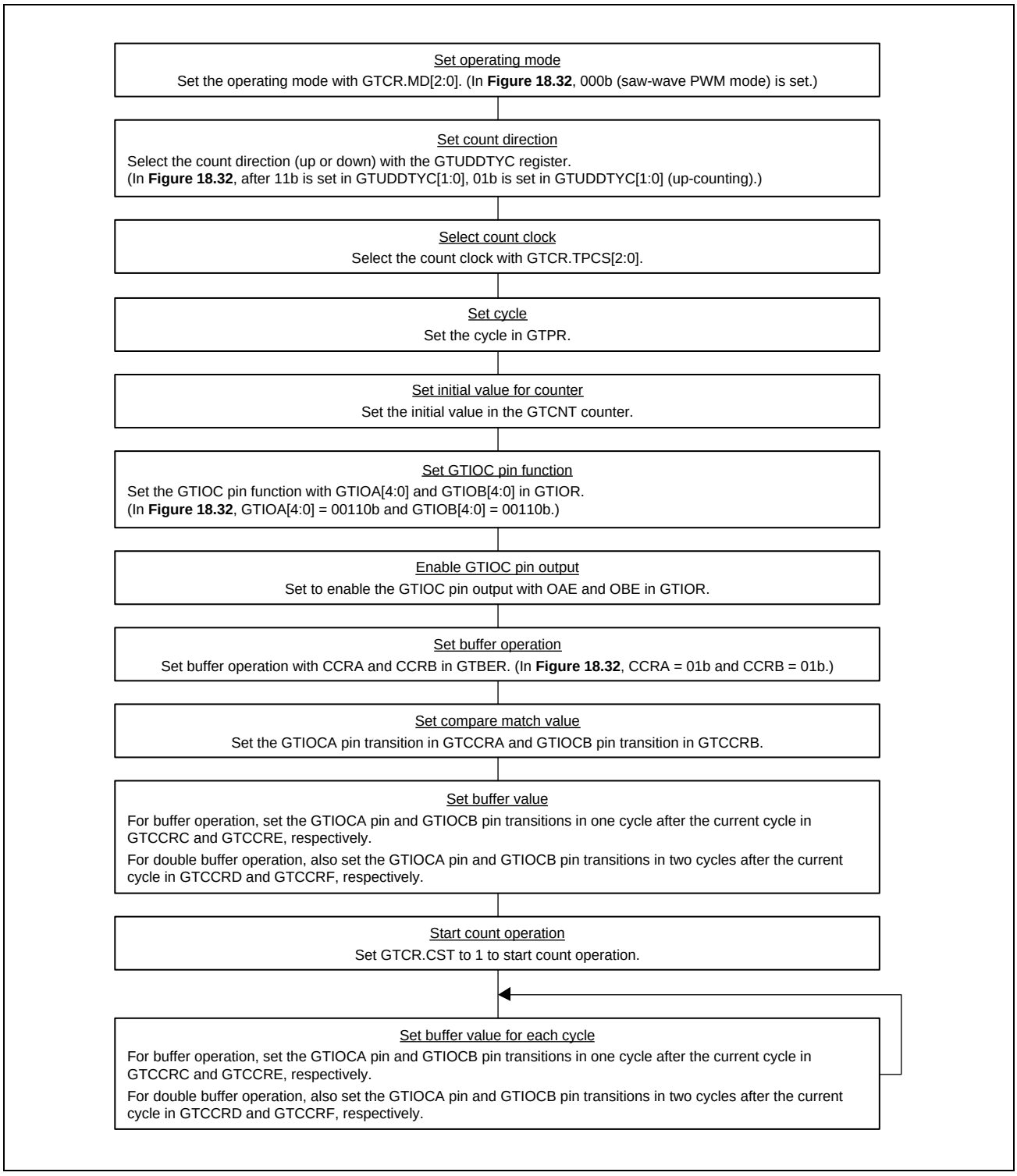


Figure 18.33 Example setting for saw-wave PWM mode

18.3.3.2 Saw-Wave One-Shot Pulse Mode

The saw-wave one-shot pulse mode is a mode in which the cycle is set in GTPR. The GTCNT counter performs saw-wave (half-wave) operation and a PWM waveform is output to the GTIOCA or GTIOCB pin at a compare match of GTCCRA or GTCCRB with buffer operation fixed.

Buffer operation in saw-wave one-shot pulse mode is different from the usual buffer operation. Buffer transfer is performed from the following:

- GTCCRC to GTCCRA at the cycle end
- GTCCRE to GTCCRB at the cycle end
- GTCCRD to temporary register A at the cycle end
- GTCCRF to temporary register B at the cycle end
- Temporary register A to GTCCRA at a GTCCRA compare match
- Temporary register B to GTCCRB at a GTCCRB compare match.

The pin output value can be selected from low output, high output, or toggle output separately for a compare match and the cycle end based on the GTIOR setting.

When 1 is written to the GTBER.CCRSWT bit while counting is stopped, the values of the GTCCRD and GTCCRF registers are forcibly transferred to the temporary registers A and B, respectively. By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

Figure 18.34 shows an example of saw-wave one-shot pulse mode operation, and **Figure 18.35** shows an example setting for saw-wave one-shot pulse mode.

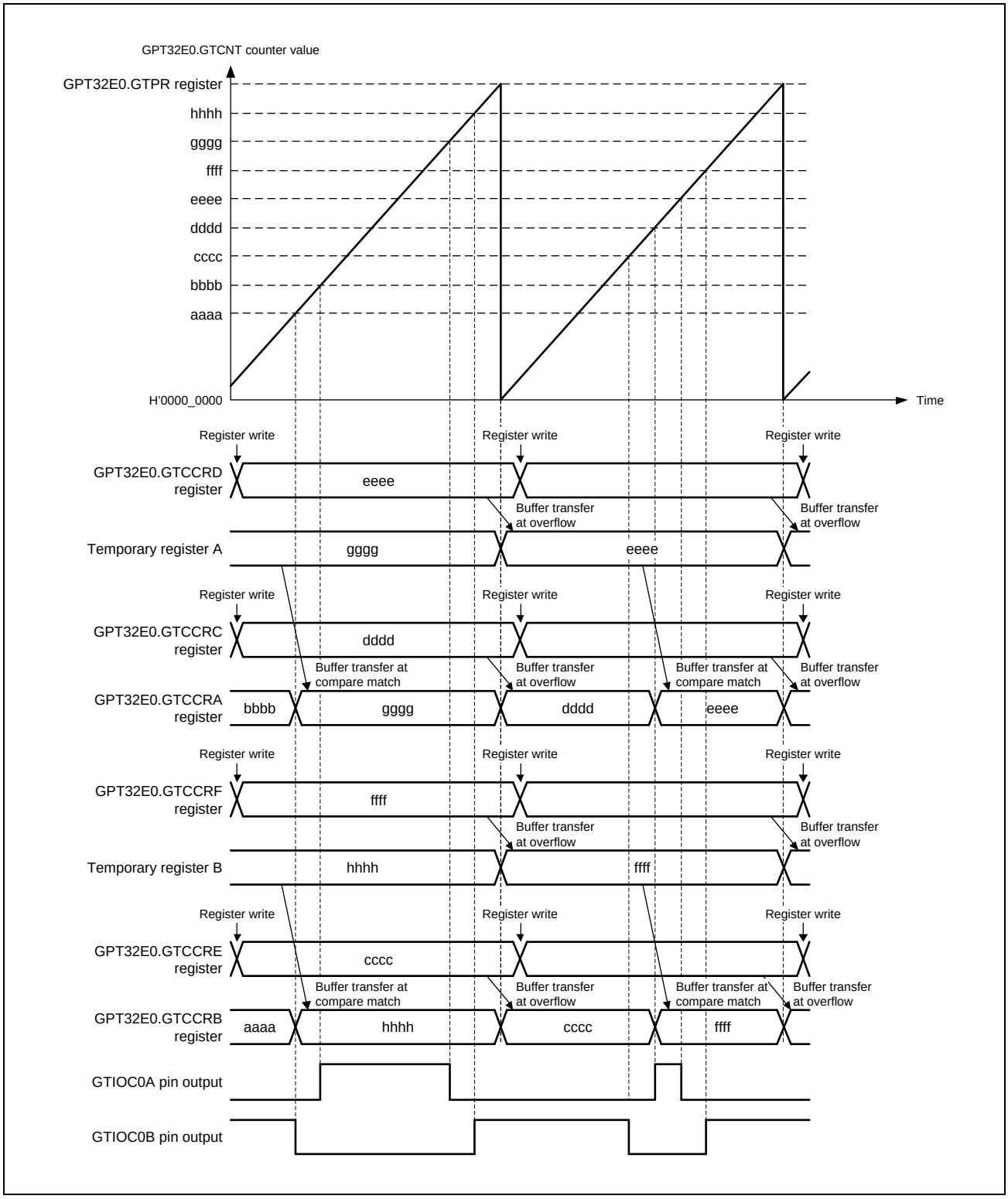


Figure 18.34 Example of saw-wave one-shot pulse mode operation with up-counting, low output from the GTIOC0A pin and high output from the GTIOC0B pin at count start, output toggled at GTCCRA/GTCCRB compare match, and output retained at cycle end

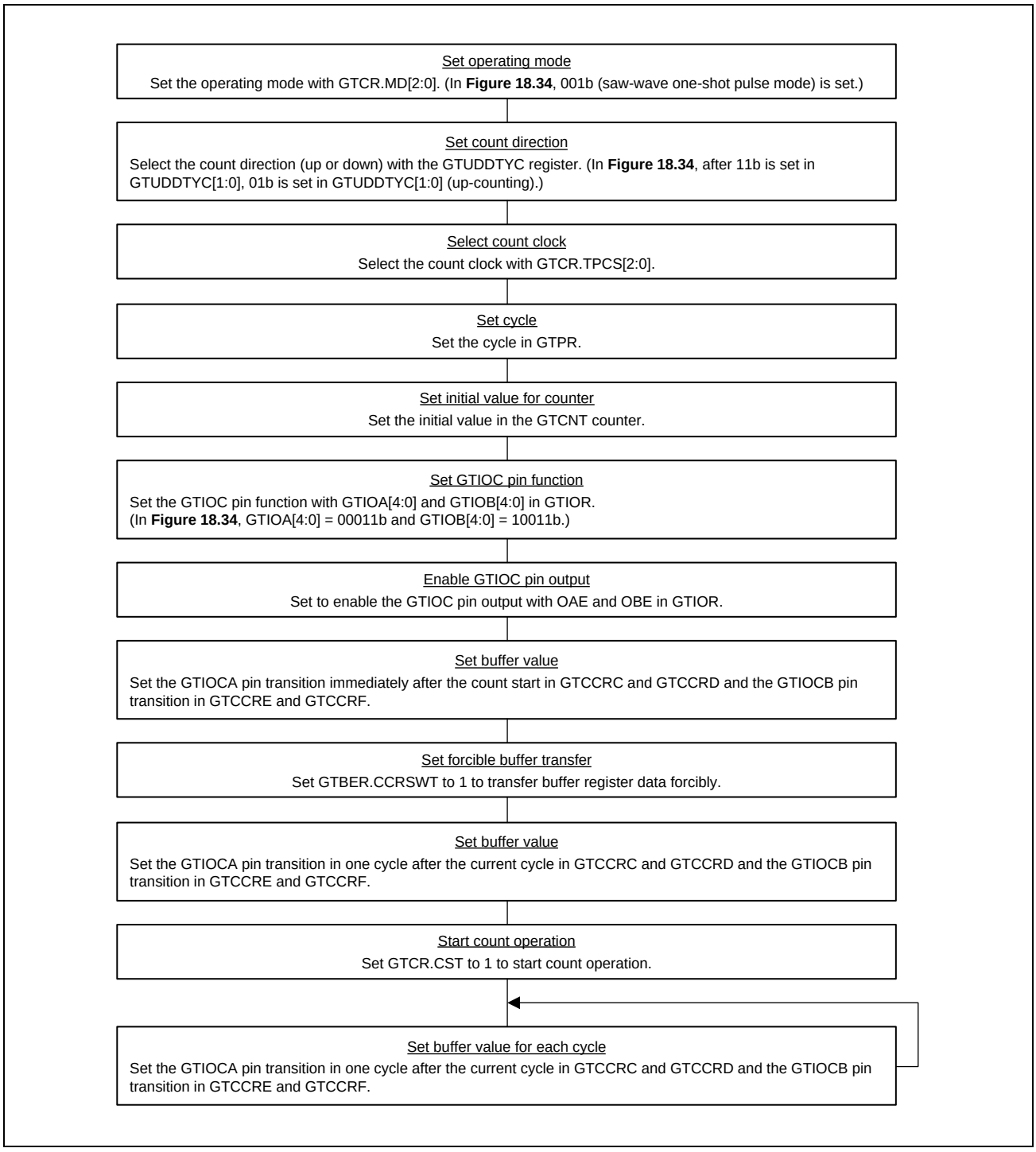


Figure 18.35 Example setting for saw-wave one-shot pulse mode

18.3.3.3 Triangle-Wave PWM Mode 1 (32-bit transfer at trough)

The triangle-wave PWM mode 1 is a mode in which the cycle is set in GTPR. The GTCNT counter performs triangle-wave (full-wave) operation, and a PWM waveform is output to the GTIOCA or GTIOCB pin when a GTCCRA or GTCCRB compare match occurs. Buffer transfer is performed at the trough. The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the cycle end based on the GTIOR setting.

By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

Figure 18.36 shows an example of a triangle-wave PWM mode 1 operation, and **Figure 18.37** shows an example setting for a triangle-wave PWM mode 1.

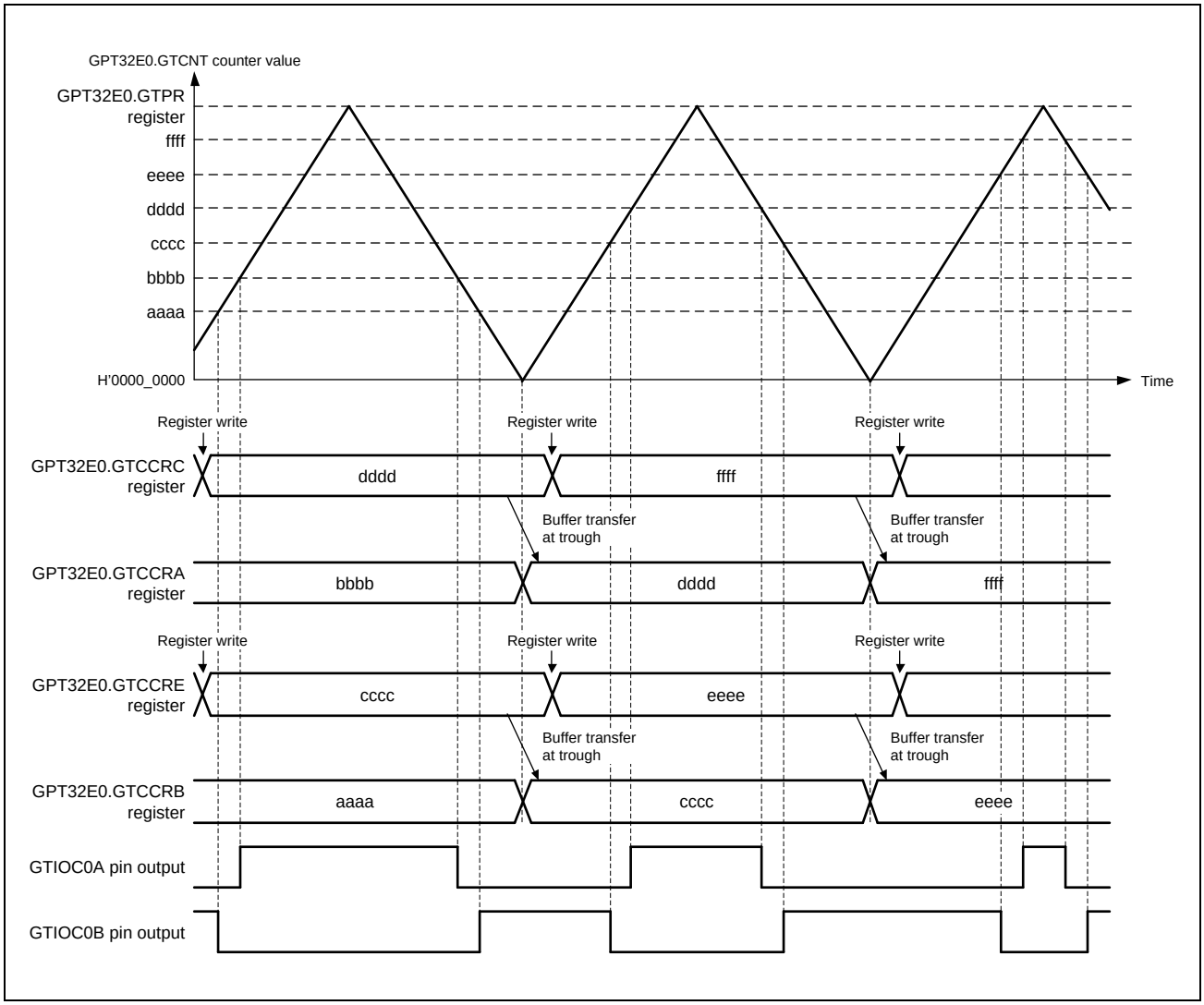


Figure 18.36 Example of triangle-wave PWM mode 1 operation with buffer operation, low output from the GTIOC0A pin and high output from the GTIOC0B pin at count start, output toggled at GTCCRA/GTCCRB register compare match, and output retained at cycle end

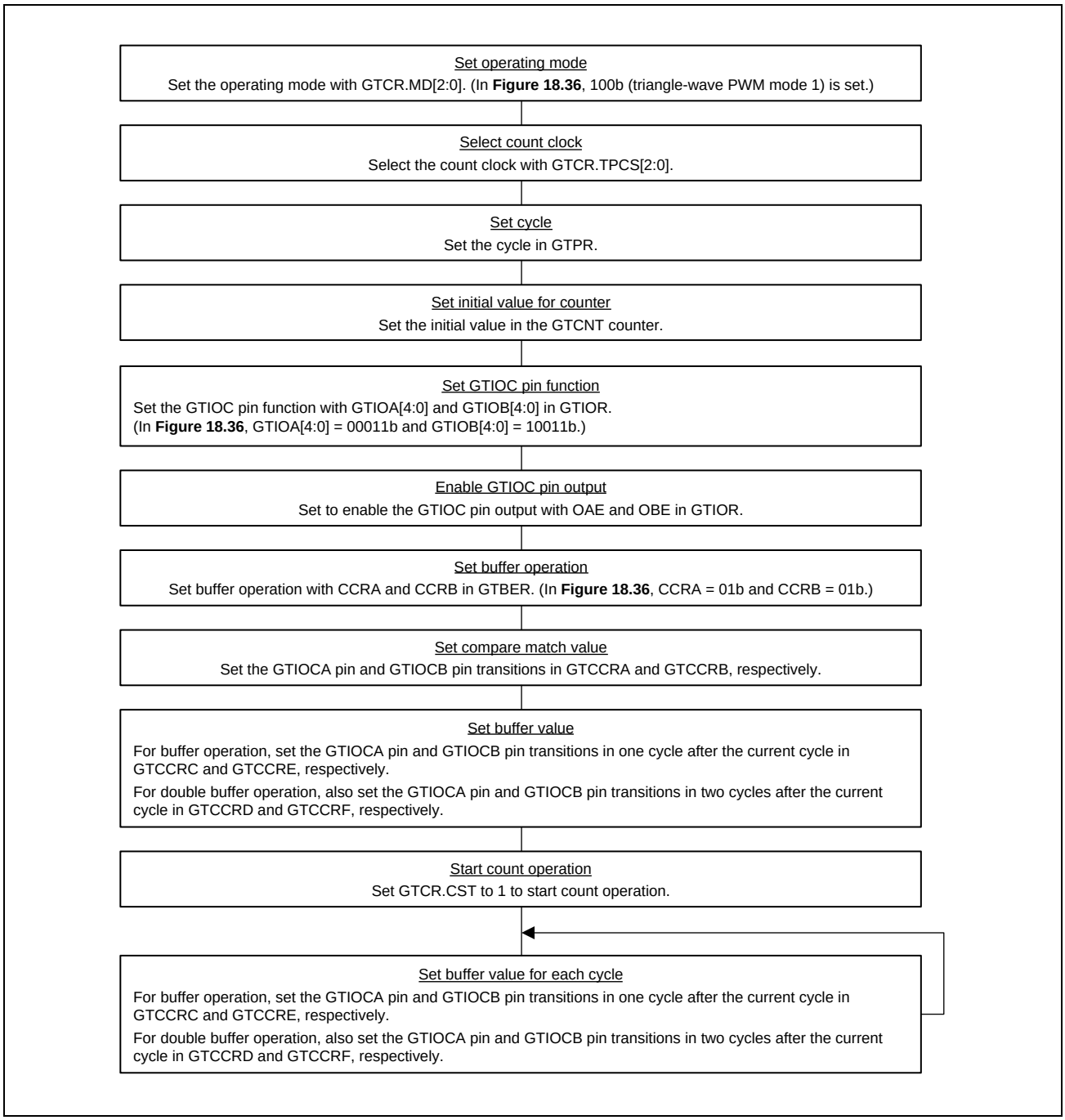


Figure 18.37 Example setting for triangle-wave PWM mode 1

18.3.3.4 Triangle-Wave PWM Mode 2 (32-bit transfer at crest and trough)

Similarly to triangle-wave PWM mode 1, in triangle-wave PWM mode 2 the cycle is set in GTPR. The GTCNT counter performs triangle-wave (full-wave) operation, and a PWM waveform is output to the GTIOCA or GTIOCB pin when a GTCCRA or GTCCRB compare match occurs. The buffer transfer is performed at both crests and troughs. The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the cycle end based on the GTIOR setting.

By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

Figure 18.38 shows an example of triangle-wave PWM mode 2 operation, and **Figure 18.39** shows an example setting for triangle-wave PWM mode 2.

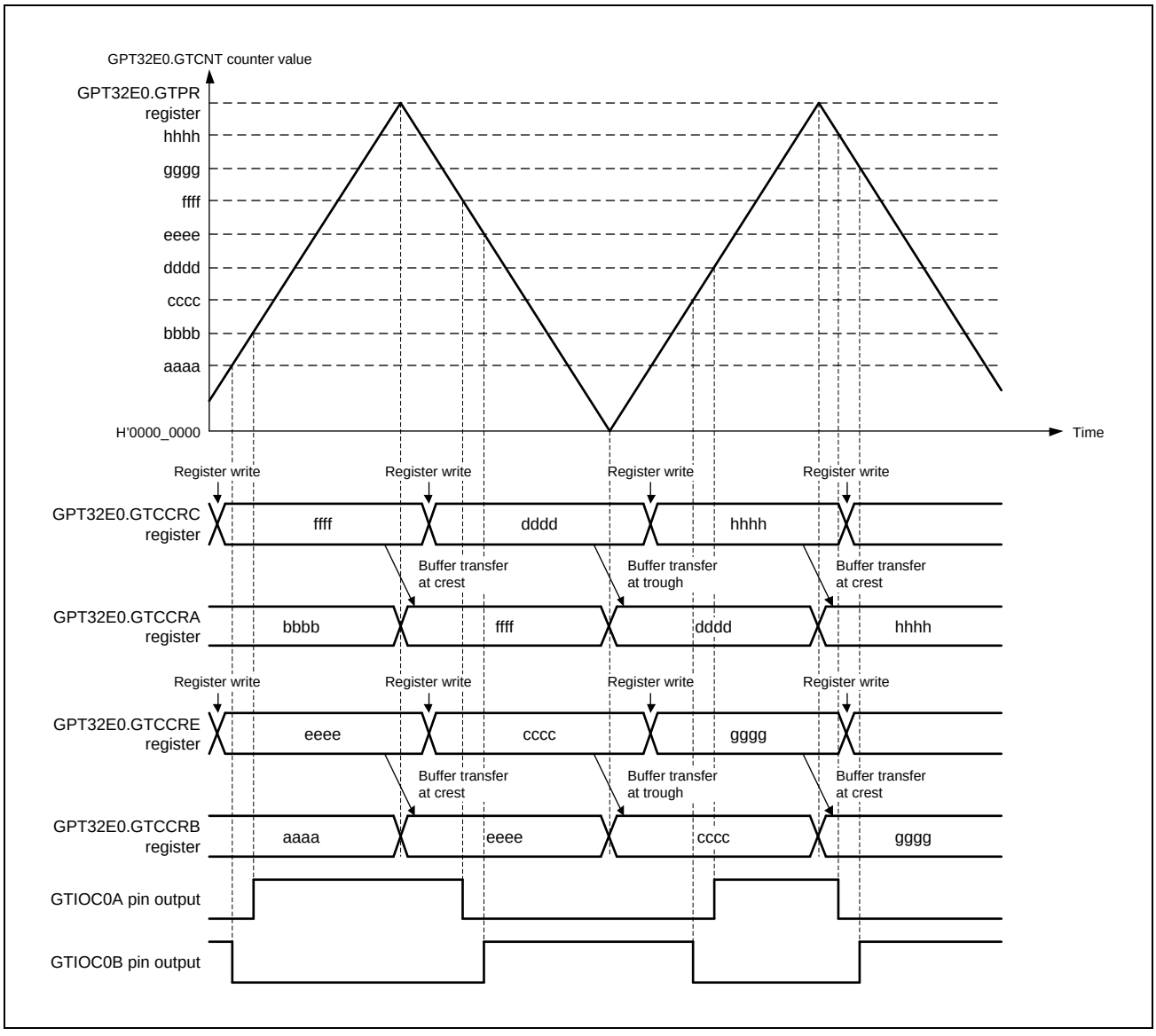


Figure 18.38 Example of triangle-wave PWM mode 2 operation with buffer operation, low output from the GTIOC0A pin and high output from the GTIOC0B pin at count start, output toggled at GTCCRA/GTCCRB compare match, and output retained at cycle end

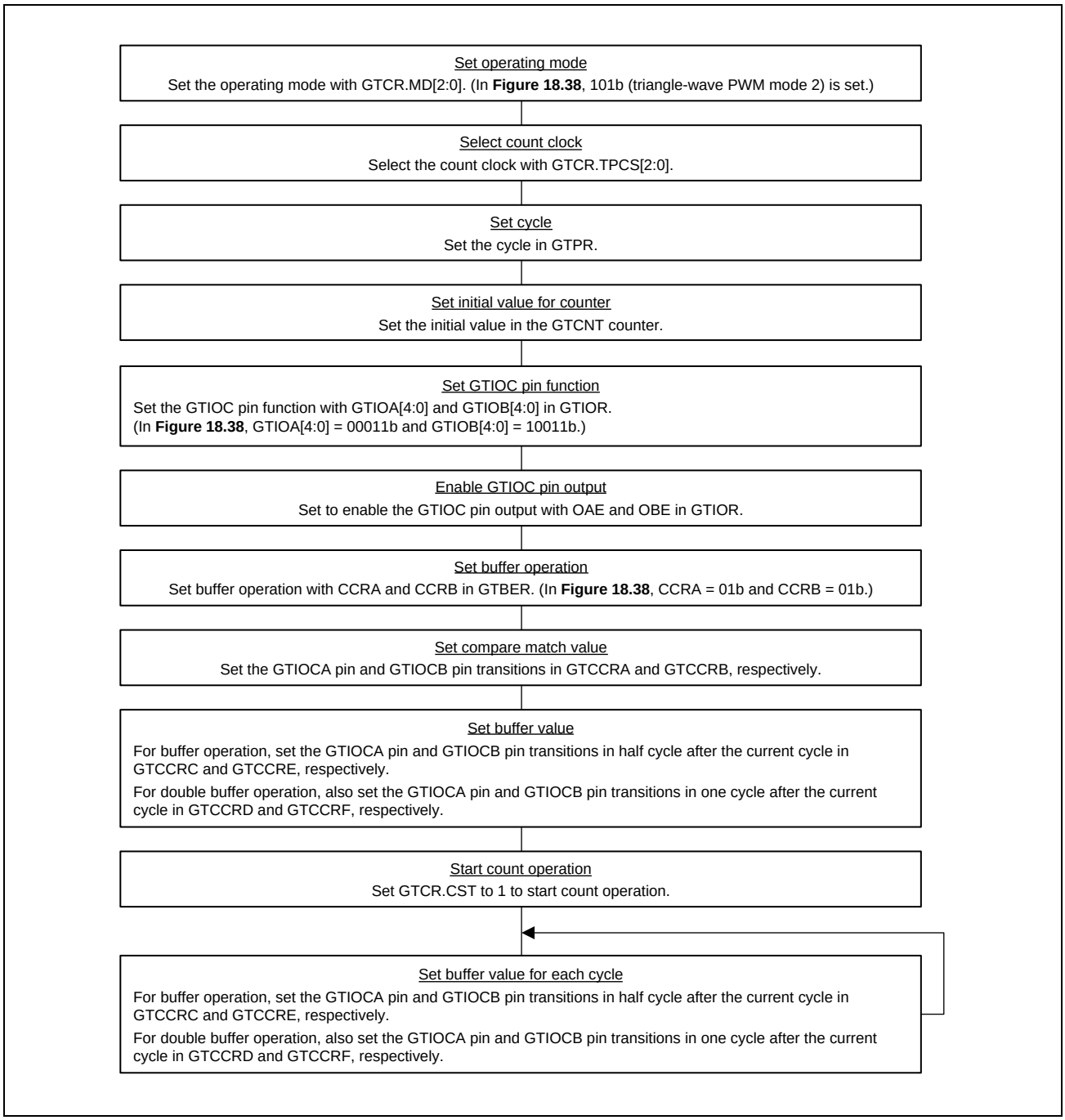


Figure 18.39 Example setting for triangle-wave PWM mode 2

18.3.3.5 Triangle-Wave PWM Mode 3 (64-bit transfer at trough)

The triangle-wave PWM mode 3 is a mode in which the cycle is set in GTPR. The GTCNT counter performs triangle-wave (full-wave) operation and a PWM waveform is output to the GTIOCA or GTIOCB pin at a compare match of GTCCRA or GTCCRB with buffer operation fixed. Buffer operation in triangle-wave PWM mode 3 is different from the usual buffer operation. Buffer transfer is performed from the following:

- GTCCRC to GTCCRA at the trough
- GTCCRE to GTCCRB at the trough
- GTCCRD to temporary register A at the trough
- GTCCRF to temporary register B at the trough
- Temporary register A to GTCCRA at the crest
- Temporary register B to GTCCRB at the crest

The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the cycle end based on the GTIOR setting. By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

Figure 18.40 shows an example of triangle-wave PWM mode 3 operation, and **Figure 18.41** shows an example setting for triangle-wave PWM mode 3.

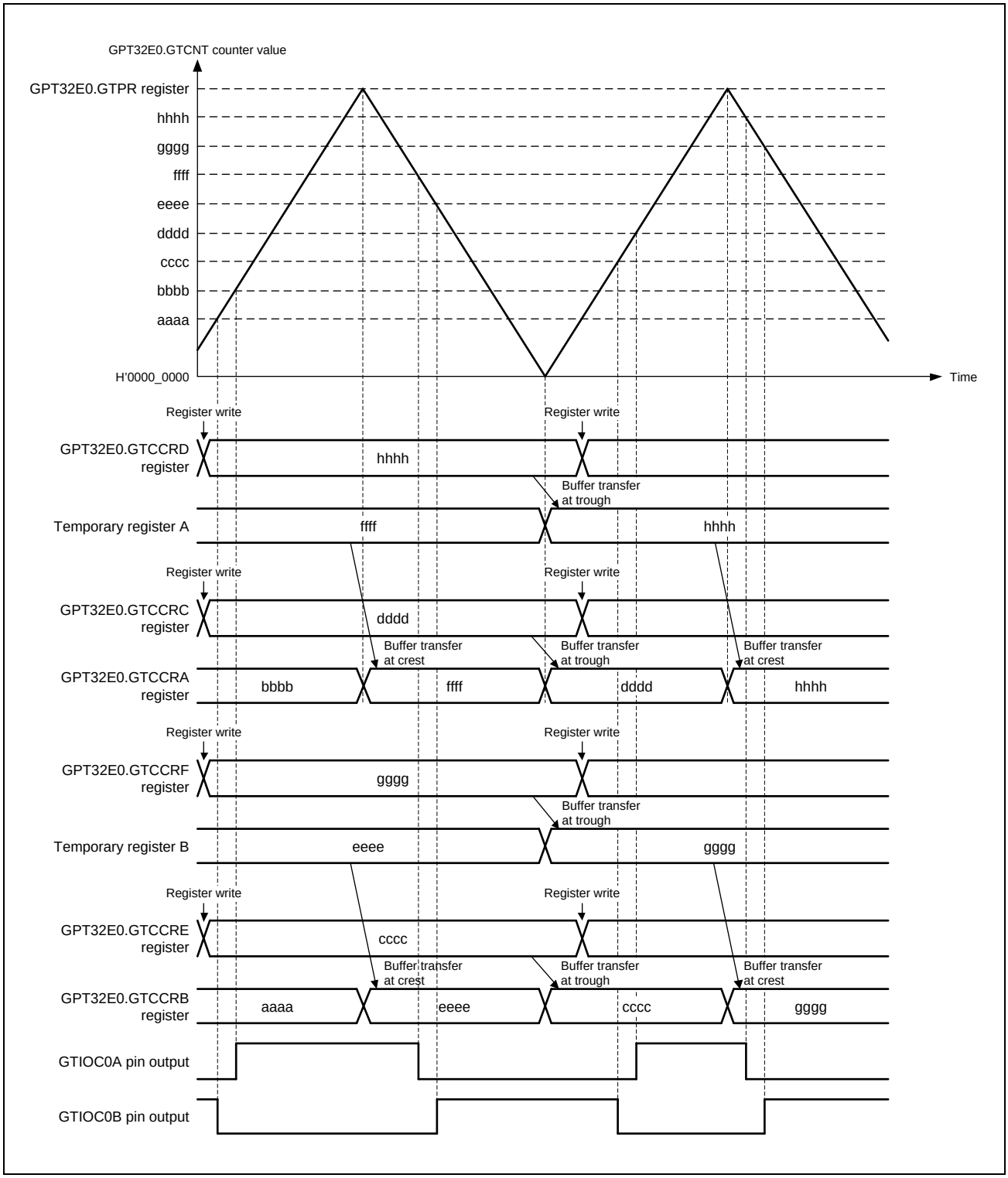


Figure 18.40 Example of triangle-wave PWM mode 3 operation with low output from the GTIOC0A pin and high output from the GTIOC0B pin at count start, output toggled at GTCCRA/GTCCRB compare match, and output retained at cycle end

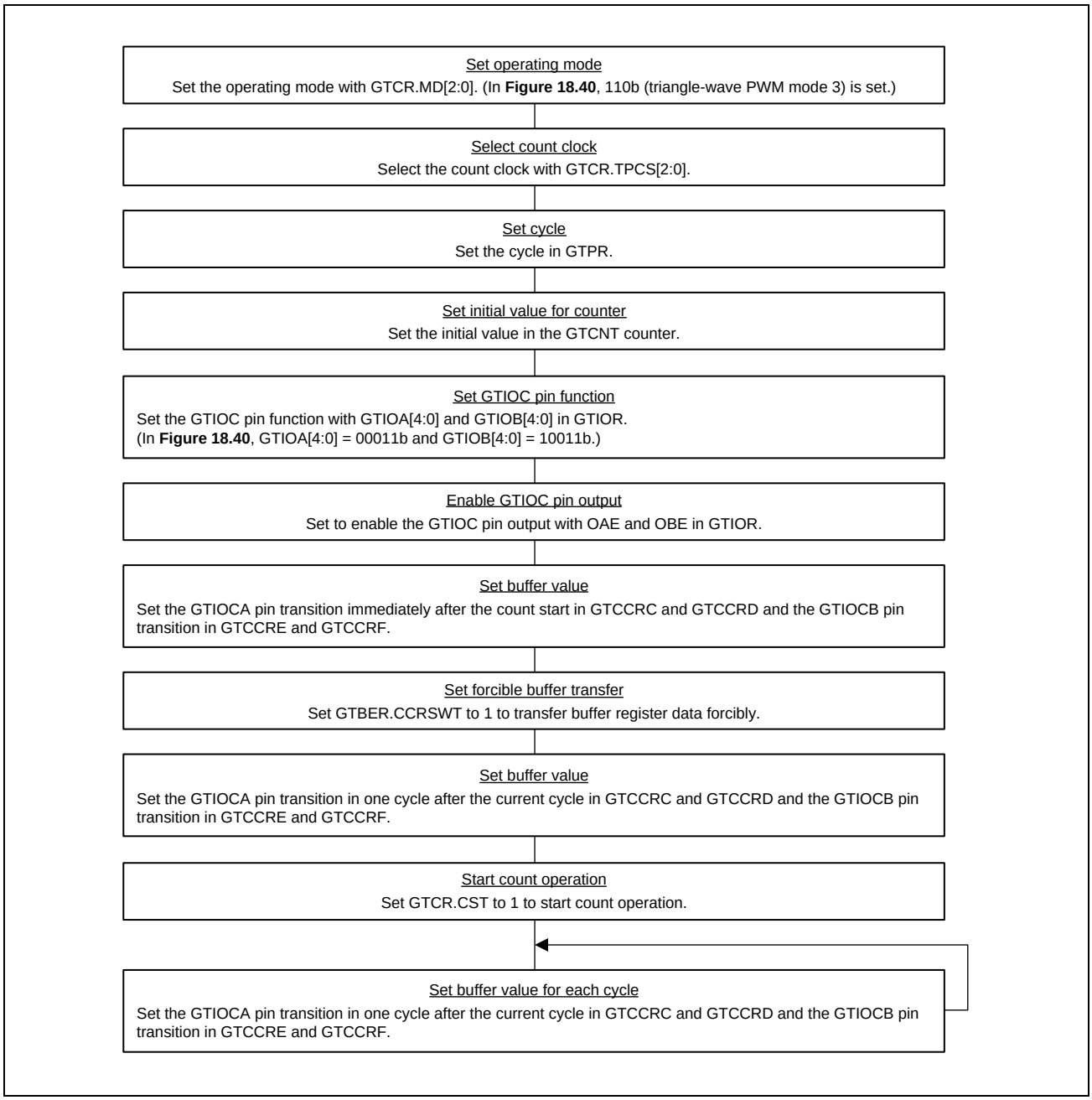


Figure 18.41 Example setting for triangle-wave PWM mode 3

18.3.4 Automatic Dead Time Setting Function

By setting GTDTCR, a compare match value for a negative waveform with dead time obtained by a compare match value for a positive waveform (GTCCRA value) and specified dead time values (GTDVU and GTDVD values) can automatically be set to GTCCRB. The automatic dead time setting function can be used in saw-wave one-shot pulse mode and all the triangle PWM modes.

Dead time can be separately set for the first half and second half of a waveform. Dead time for the transition in the first half of a negative waveform is set in GTDVU and that in the second half is set in GTDVD. The same dead time can also be set for the first and second halves by setting the GTDTCR.TDFER bit to 1.

GTDBU can be used as a buffer register of GTDVU, and GTDBD can be used as a buffer register of GTDVD.

In the saw-wave mode, buffer transfer is performed when the GTCND counter overflows (up-counting), underflows (down-counting), or is cleared. In the triangle-wave mode, transfer proceeds at troughs.

The change point of the negative phase waveform with the automatic dead time setting can be confirmed by reading the GTCCRB register value.

Writing to the GTCCRB register is prohibited when using the automatic dead time setting function. It is prohibited to set dead time such that the change point of the waveform exceeds the count cycle.

If there is an error condition on the dead time setting, be able to generate waveforms that secured dead time by correcting the transition points of the positive and negative phase waveforms, as shown in **Table 18.6**.

The transition point of the corrected negative-phase waveform is automatically set in the GTCCRB register. Since the internal signal is used to judge the transition point of the positive phase waveform, the GTCCRA register is not updated with the corrected value.

By correcting the waveform transition point due to a dead time error on the saw-wave one-shot pulse mode, if the order of waveform transition points is disturbed or it exceeds the count cycle period after correction, the complementary relationship between the positive-phase waveform and the negative-phase waveform is not guaranteed.

By setting the GTCCRA equal 0 or greater than or equal to the GTPR of the GTCCRA register in the triangle-wave PWM mode, if the dead time setting exceeds the count cycle period, output transition is suppressed by the output protection function. When GTCCRA ≥ GTPR + GTDVn, GTPR - 1 is set as the upper limit value in the GTCCRB register (Refer to **Section 18.7.4, Output Protection Function for GTIOC Pin Output**).

When GTCCRA ≥ GTPR + GTDVn, GTPR - 1 is set as the upper limit value in the GTCCRB register.

Table 18.6 Correction of waveform transition point at dead time error occurrence

Wave mode	Count direction	Interval	Dead time error condition	Positive-Phase waveform transition point with corrected	Negative-phase waveform transition point with corrected
saw-wave one-shot pulse mode	Up counting	first half	GTCCRA-GTDVU < 0	GTDVU	0
		second half	GTCCRA-GTDVD > GTPR	GTPR-GTDVD	GTPR
	Down counting	first half	GTCCRA-GTDVU > GTPR	GTPR-GTDVU	GTPR
		second half	GTCCRA-GTDVD < 0	GTDVD	0
triangle PWM mode 1/2/3	Up counting	(first half)	GTCCRA-GTDVU ≤ 0	GTDVU+1	1
	Down counting	(second half)	GTCCRA-GTDVD < 0	GTDVD	0

Values for automatic dead time setting can be read from GTCCRB. The automatic dead time value setting to GTCCRB is performed at the next count clock cycle when registers that are used for calculating the automatic dead time value are updated.

The method of writing a new value to a GTDVn register depends on whether buffer operation is enabled or disabled. When GTDVn buffer operation is enabled: The GTDBn register can be written at any time. The value in the GTDBn register is transferred to the GTDVn register at the cycle end.

When GTDVn buffer operation is disabled: Set the GTCR.CST bit to stop the GPT before changing the value of the GTDVn register.

Figure 18.42 to **Figure 18.45** show examples of automatic dead time setting function operation. **Figure 18.46** and **Figure 18.47** show the setting examples.

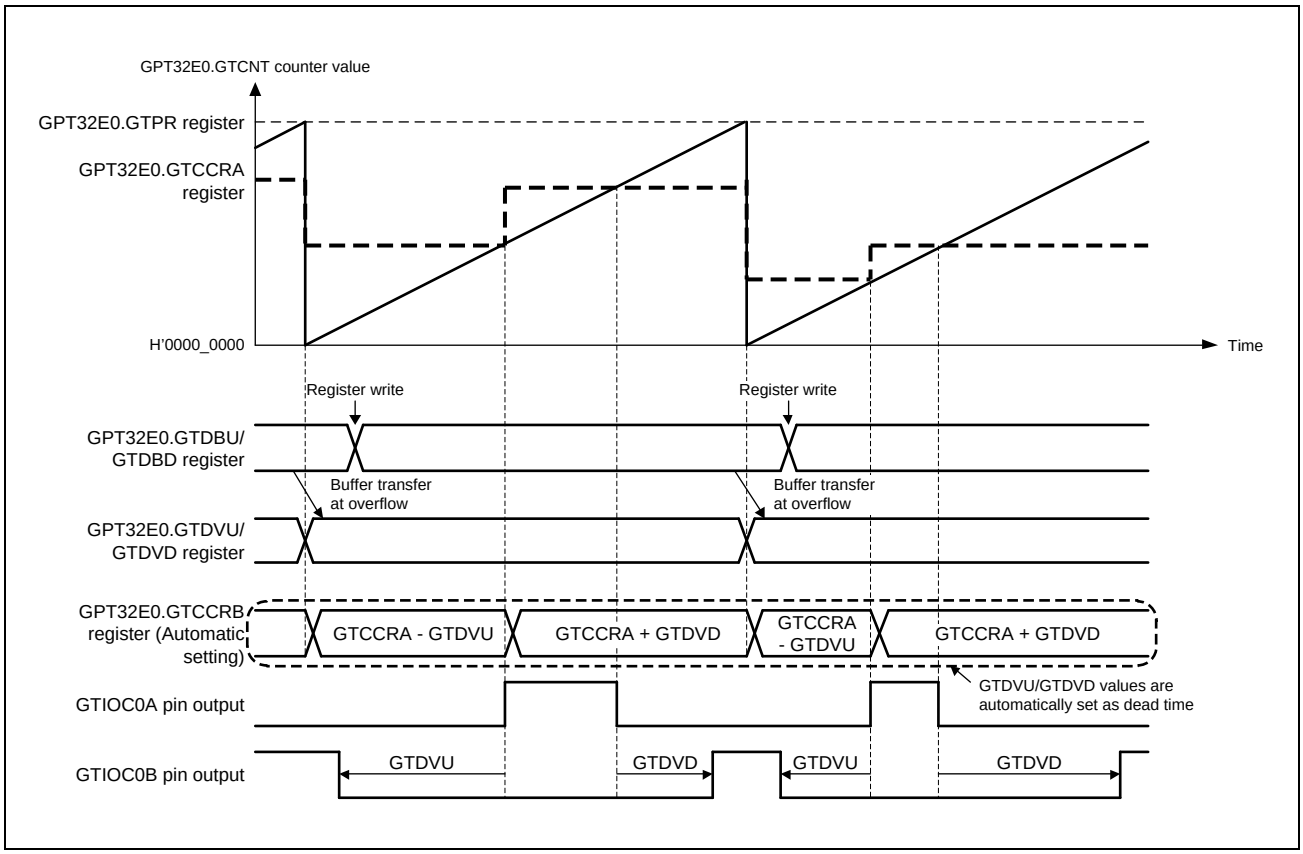


Figure 18.42 Example of automatic dead time setting function operation with saw-wave one-shot pulse mode, up-counting, GTDVU and GTDVD set to buffer operation, and active-high

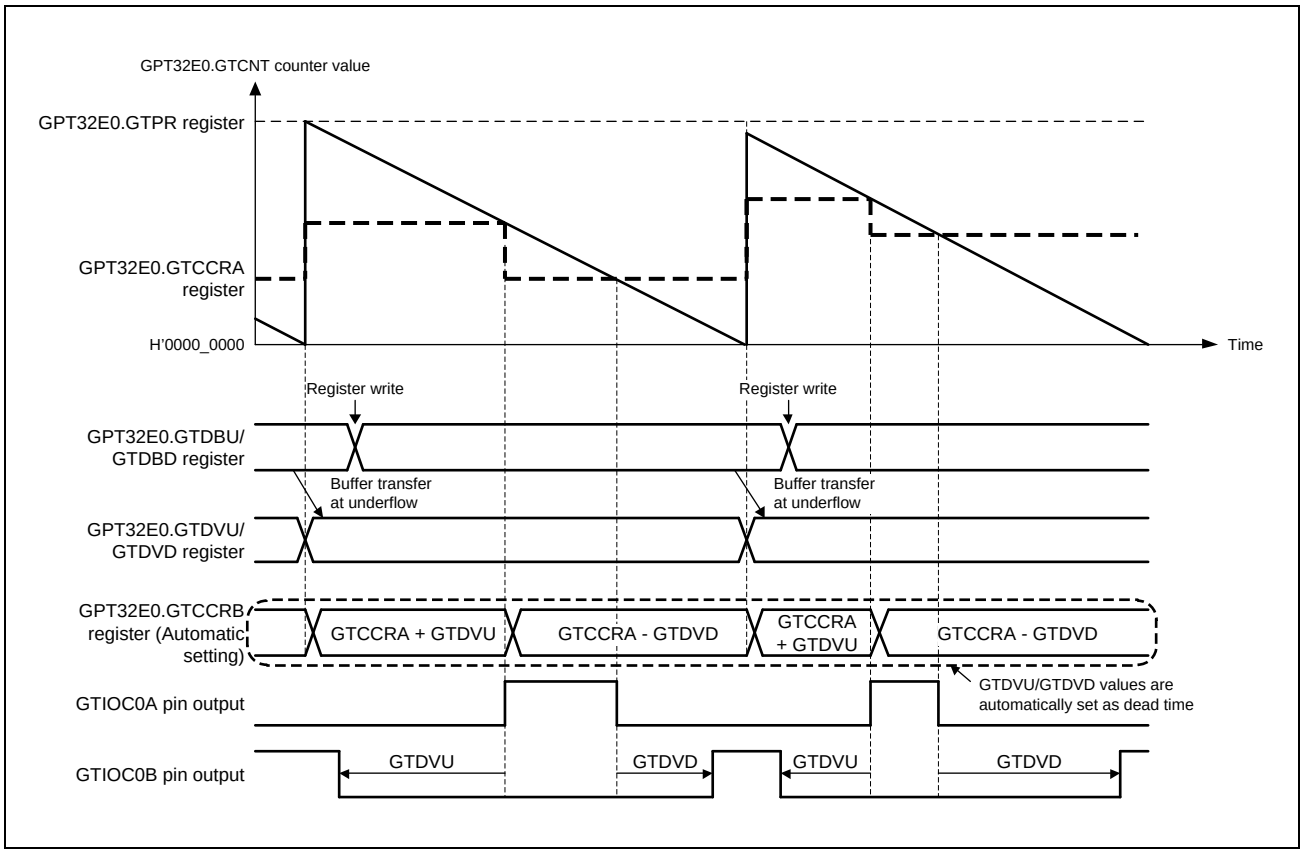


Figure 18.43 Example of automatic dead time setting function operation with saw-wave one-shot pulse mode, down-counting, GTDVU and GTDVD set to buffer operation, and active-high

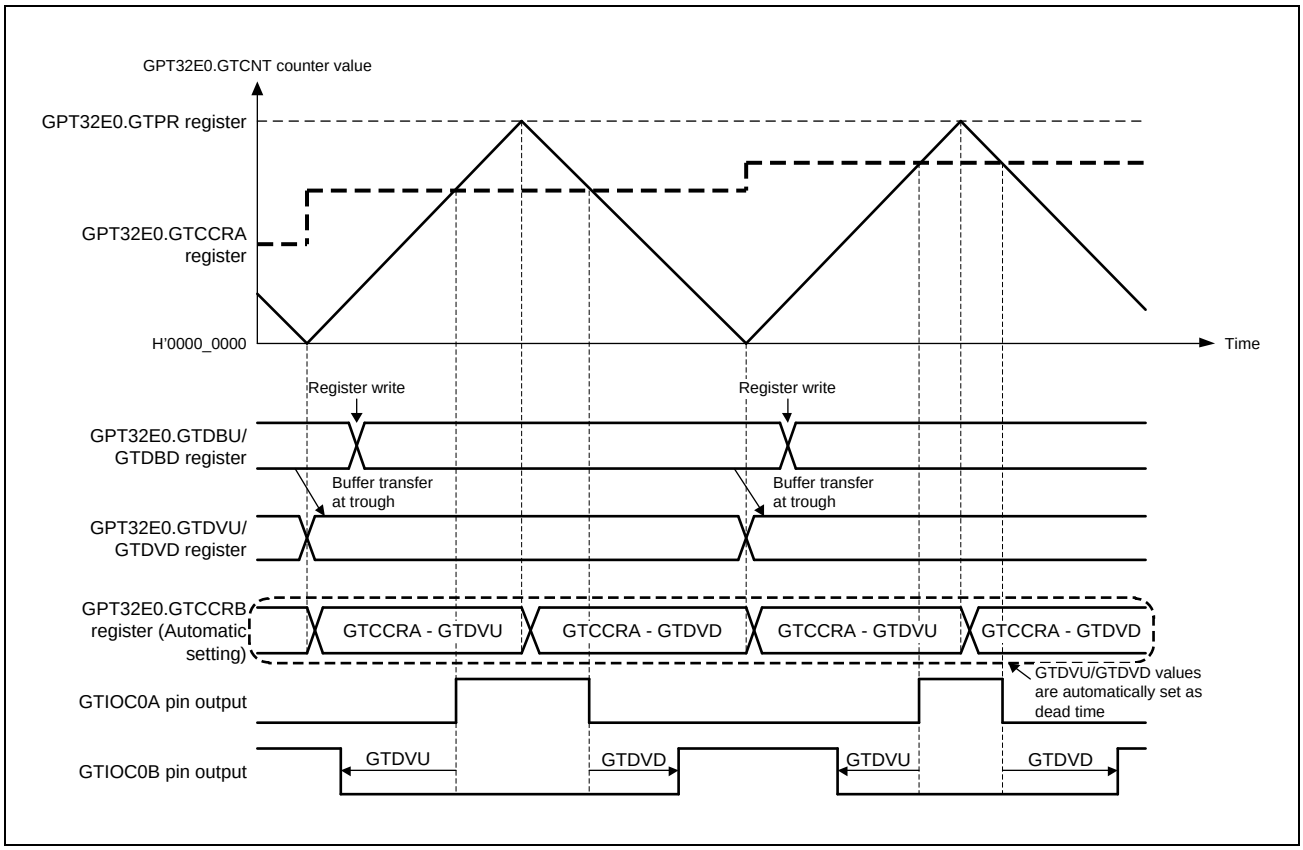


Figure 18.44 Example of automatic compare-match value setting function with dead time with triangle-wave PWM mode 1, GTDVU and GTDVD set to buffer operation, active-high

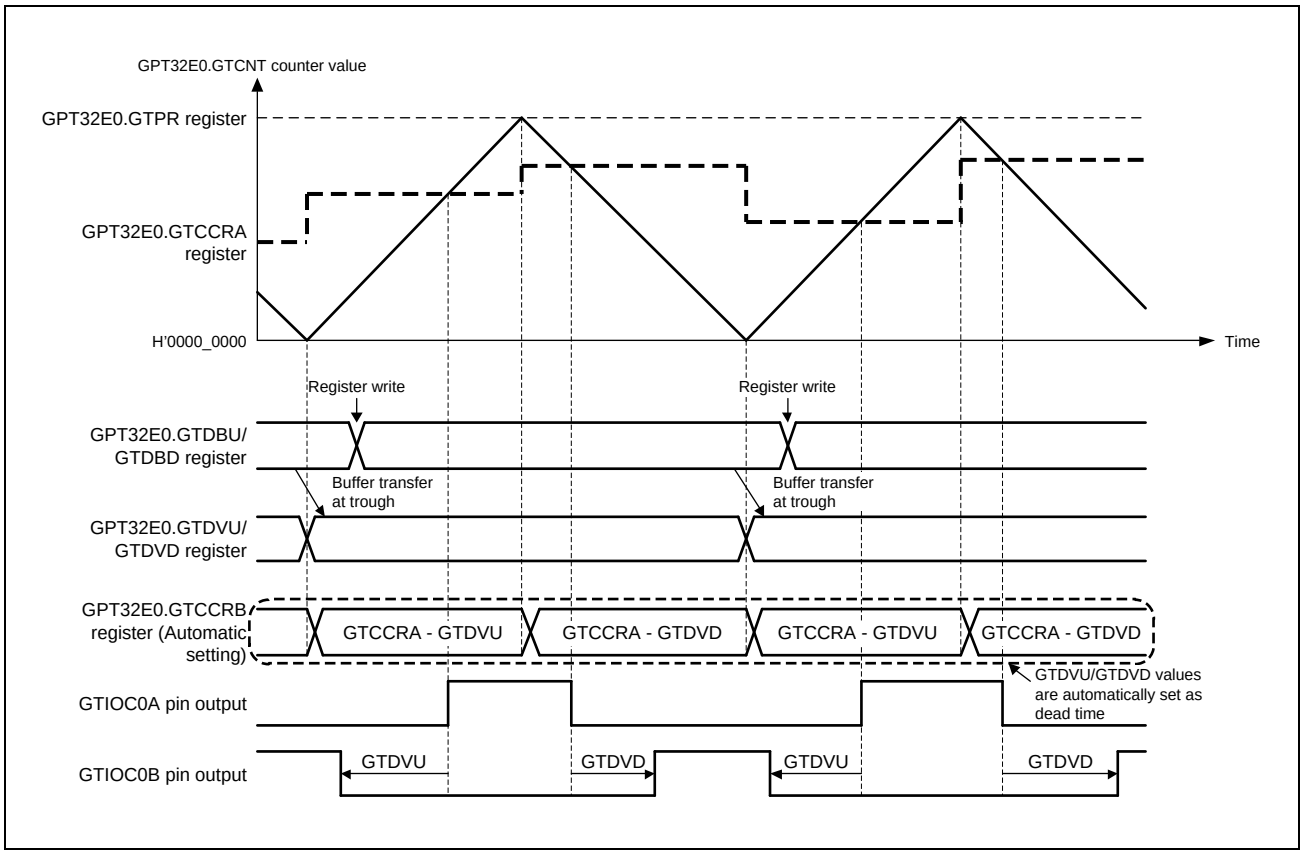


Figure 18.45 Example of automatic compare-match value setting function with dead time, with triangle-wave PWM mode 2 or 3, GTDVU and GTDVD set to buffer operation, and active-high

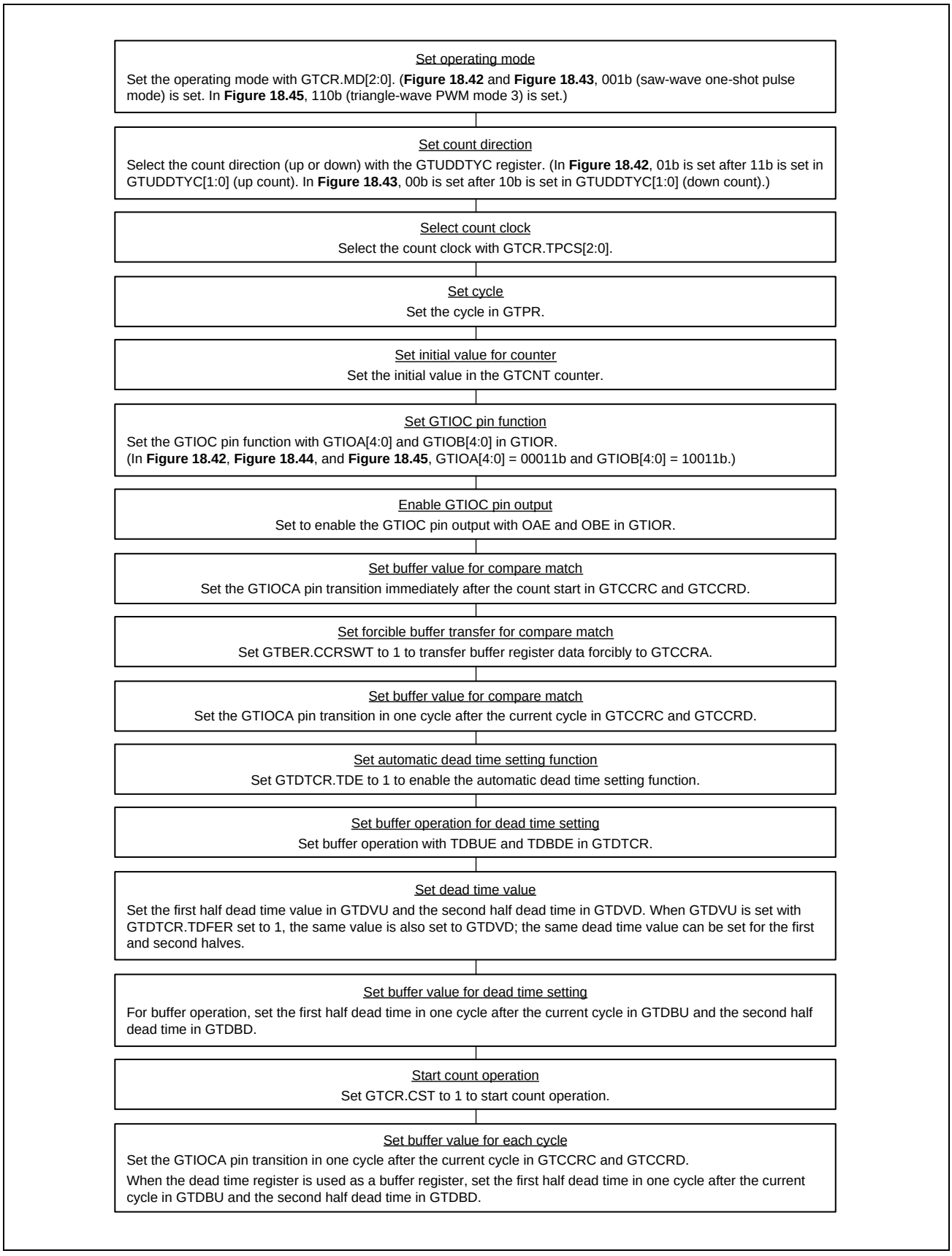


Figure 18.46 Example setting for automatic dead time setting function with saw-wave one-shot pulse mode, and triangle-wave PWM mode 3

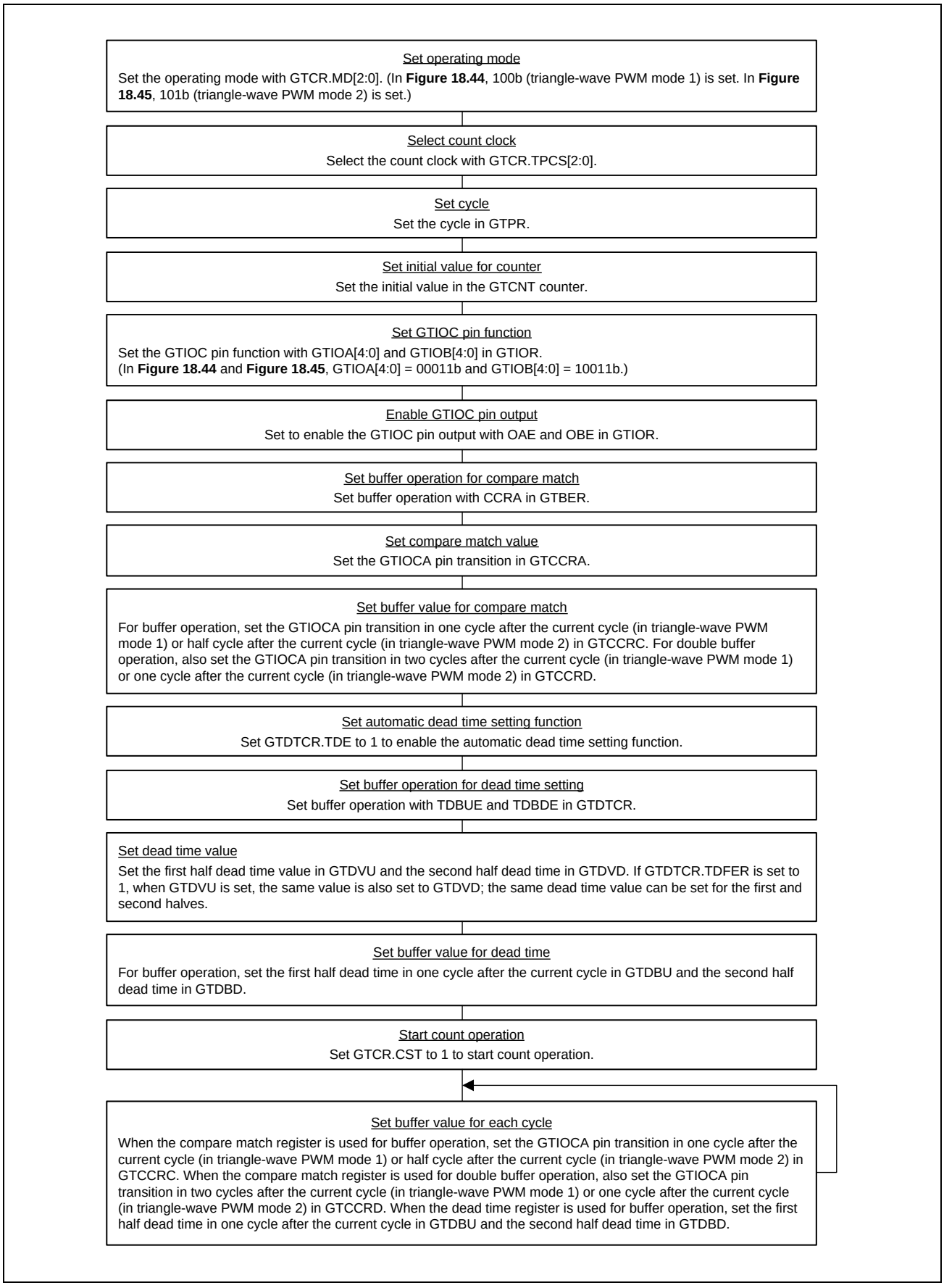


Figure 18.47 Example setting for automatic dead time setting function with triangle-wave PWM mode 1 or 2

18.3.5 Count Direction Changing Function

The count direction of the GTCNT counter can be changed by modifying the UD bit in GTUDDTYC.

In saw-wave mode, if the UD bit in GTUDDTYC is modified during count operation, the count direction is changed at an overflow (when modified during up-counting) or an underflow (when modified during down-counting). If the GTUDDTYC.UD bit is modified while the count operation stops and the GTUDDTYC.UDF bit is 0, the GTUDDTYC.UD bit modification is not reflected at the start of counting and the count direction is changed at an overflow or an underflow. If the UDF bit is set to 1 while the count operation stops, the GTUDDTYC.UD bit value at that time is reflected at the start of counting.

In triangle-wave mode, the count direction does not change even though the UD bit in GTUDDTYC is modified during the count operation. Similarly, even though the GTUDDTYC.UD bit is modified while the count operation stops and GTUDDTYC.UDF bit is 0, the GTUDDTYC.UD bit value is not reflected to the count operation. If the GTUDDTYC.UDF bit is set to 1 while the count operation stops, the GTUDDTYC.UD bit value at that time is reflected at the start of counting.

If the count direction changes during a saw-wave count operation, the GTPR value after the start of up-counting is reflected to the count cycle during up-counting and the GTPR value before the start of down-counting is reflected during down-counting.

Figure 18.48 shows an example of count direction changing function operation.

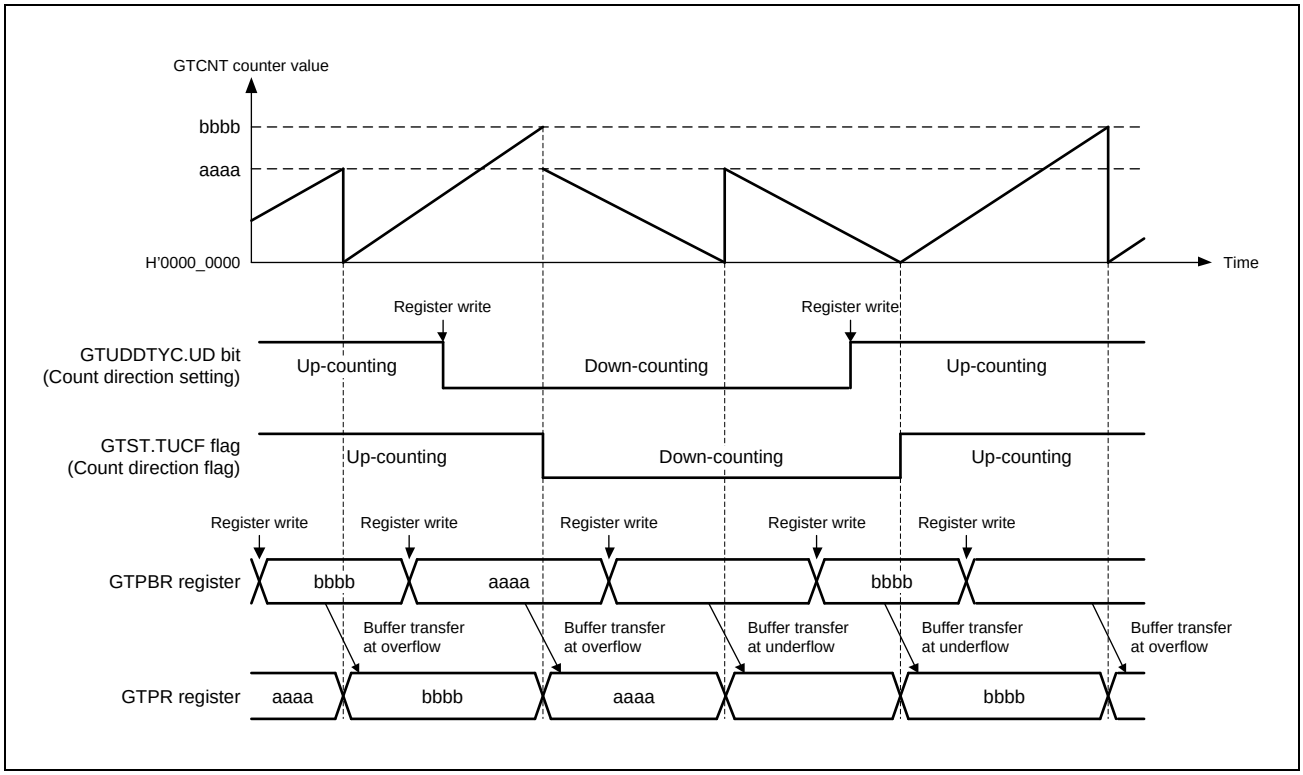


Figure 18.48 Example of count direction changing function operation during buffer operation

18.3.6 Function of Output Duty 0% and 100%

The output duty of the GTIOCA pin and the GTIOCB pin are set to 0% or 100% by changing the GTUDDTYC.OADTY bit or GTUDDTYC.OBDTY bit.

In saw-wave mode, if the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified during the count operation, the output duty setting is reflected at an overflow (when modified during up-counting) or an underflow (when modified during down-counting). If the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified while the count operation stops and the GTUDDTYC.OADTYF or the GTUDDTYC.OBDTYF bit is 0, the output duty modification is not reflected at the start of counting. The output duty changes at an overflow or an underflow. If the GTUDDTYC.OADTYF or the GTUDDTYC.OBDTYF bit is set to 1 while the count operation stops, the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit value at that time is reflected at the start of counting.

In triangle-wave mode, if the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified during the count operation, the output duty setting is reflected an underflow.

If the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified while the count operation stops and the GTUDDTYC.OADTYF or the GTUDDTYC.OBDTYF bit is 0, the output duty modification is not reflected at the start of counting. The output duty changes at an underflow. If the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified while the count operation stops and the GTUDDTYC.OADTYF or the GTUDDTYC.OBDTYF bit is 1, the output duty modification is reflected at the start of counting.

In performing 0%/100% duty operation, GPT internally continues to:

- Perform compare match operation
- Set compare match flag
- Output interrupt
- Perform buffer operation.

When the control is changed from 0% or 100% duty setting to compare match, the output value of GTIOCA pin at cycle end is decided by GTIOR.GTIOA[3:2] and GTUDDTYC.OADTYR. The output value of GTIOCB pin at cycle end is decided by GTIOR.GTIOB[3:2] and GTUDDTYC.OBDTYR.

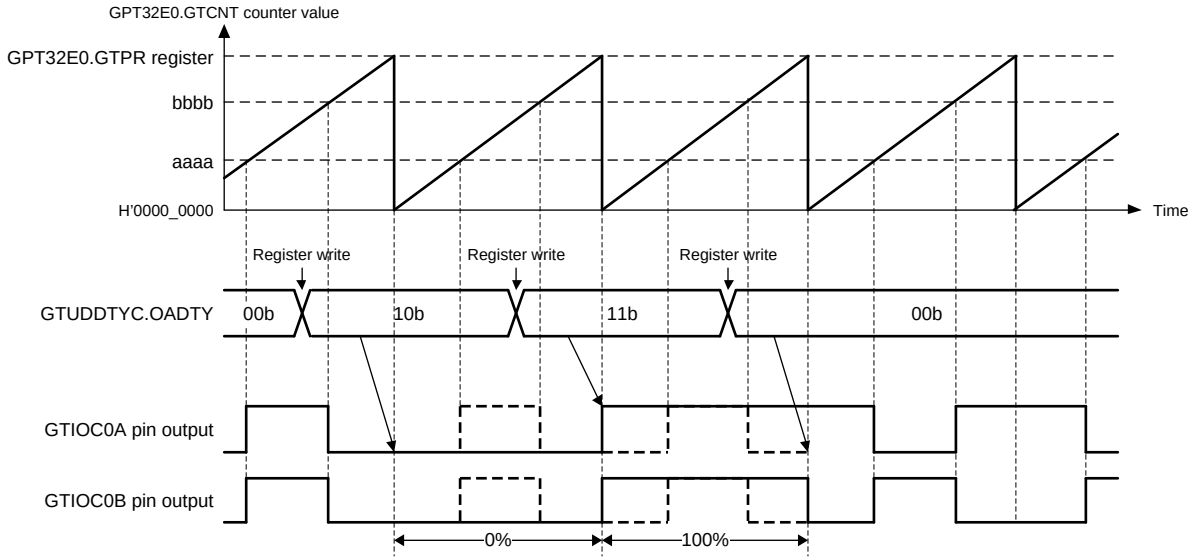
When GTIOR.GTIOA[3:2] and GTIOR.GTIOB[3:2] are set to 01b, the output pins output low at cycle end. When GTIOR.GTIOA[3:2] and GTIOR.GTIOB[3:2] are set to 10b, the output pins output high at cycle end.

GTUDDTYC.OADTYR selects the value that is the object of output retained/toggled at cycle end, when GTIOR.GTIOm[3:2] are set to 00b (output retained at cycle end) or when GTIOR.GTIOm[3:2] are set to 11b (output toggled at cycle end). **Table 18.7** shows the values of GTIOCA/GTIOCB pin output at cycle end.

Table 18.7 Output values after releasing 0% or 100% duty setting (m = A, B)

GTIOR.GTIOm[3:2]	Compare match value at cycle end masked by 0% or 100% duty setting	GTUDDTYC.OmDTYR in duty 0% setting		GTUDDTYC.OmDTYR in duty 100% setting	
		0	1	0	1
00 (Output retained at cycle end)	0	0	0	1	0
	1	0	1	1	1
01 (low output at cycle end)	—	0	0	0	0
10 (high output at cycle end)	—	1	1	1	1
11 (Output toggled at cycle end)	0	1	1	0	1
	1	1	0	0	0

Figure 18.49 shows the example of output duty 0% and 100% functions.



[Setting examples]

GPT32E0.GTIOR.GTIOA[4:0] bits: 00011b
Initial output of low, output toggled at compare match, output retained at cycle end

GPT32E0.GTUDDTYC.OADTYR bit: 0
Applied the value of duty 0%/100% output to GTIOA[3:2] bits function after 0%/100% duty setting is released.

GPT32E0.GTIOR.GTIOB[4:0] bits: 00011b
Initial output of low, output toggled at compare match, output retained at cycle end

GPT32E0.GTUDDTYC.OBDTYR bit: 1
Applied the value of masked compare match output to GTIOB[3:2] bits function after 0%/100% duty setting is released

Figure 18.49 Example of output duty 0% and 100% functions

18.3.7 Hardware Count Start/Count Stop and Clear Operation

The GTCNT counter can be started, stopped, or cleared by the following hardware sources:

- External trigger input
- GTIOCA/GTIOCB pin input.

18.3.7.1 Hardware Start Operation

The GTCNT counter can be started by selecting a hardware source using GTSSR.

Figure 18.50 shows an example of a count start operation by a hardware source. **Figure 18.51** shows the setting example.

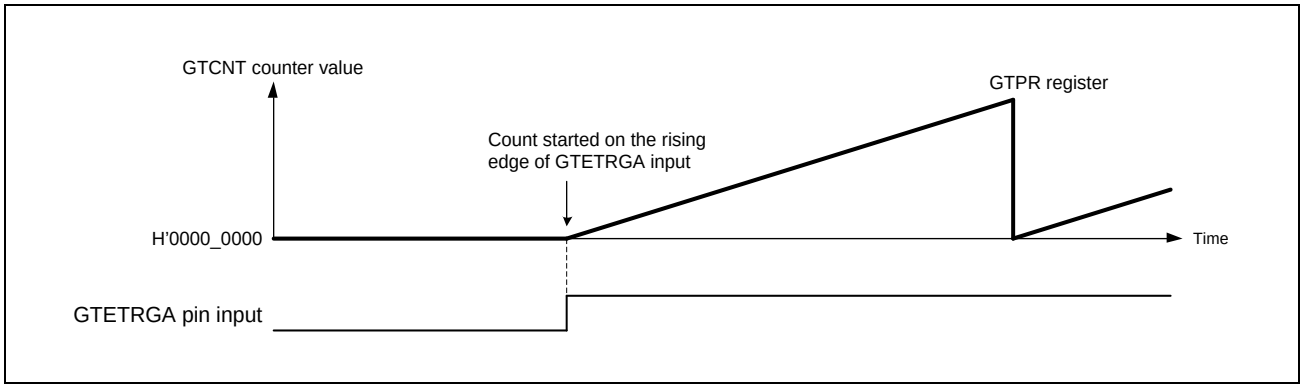


Figure 18.50 Example of count start operation by hardware source, started on the rising edge of GTETRGA pin input

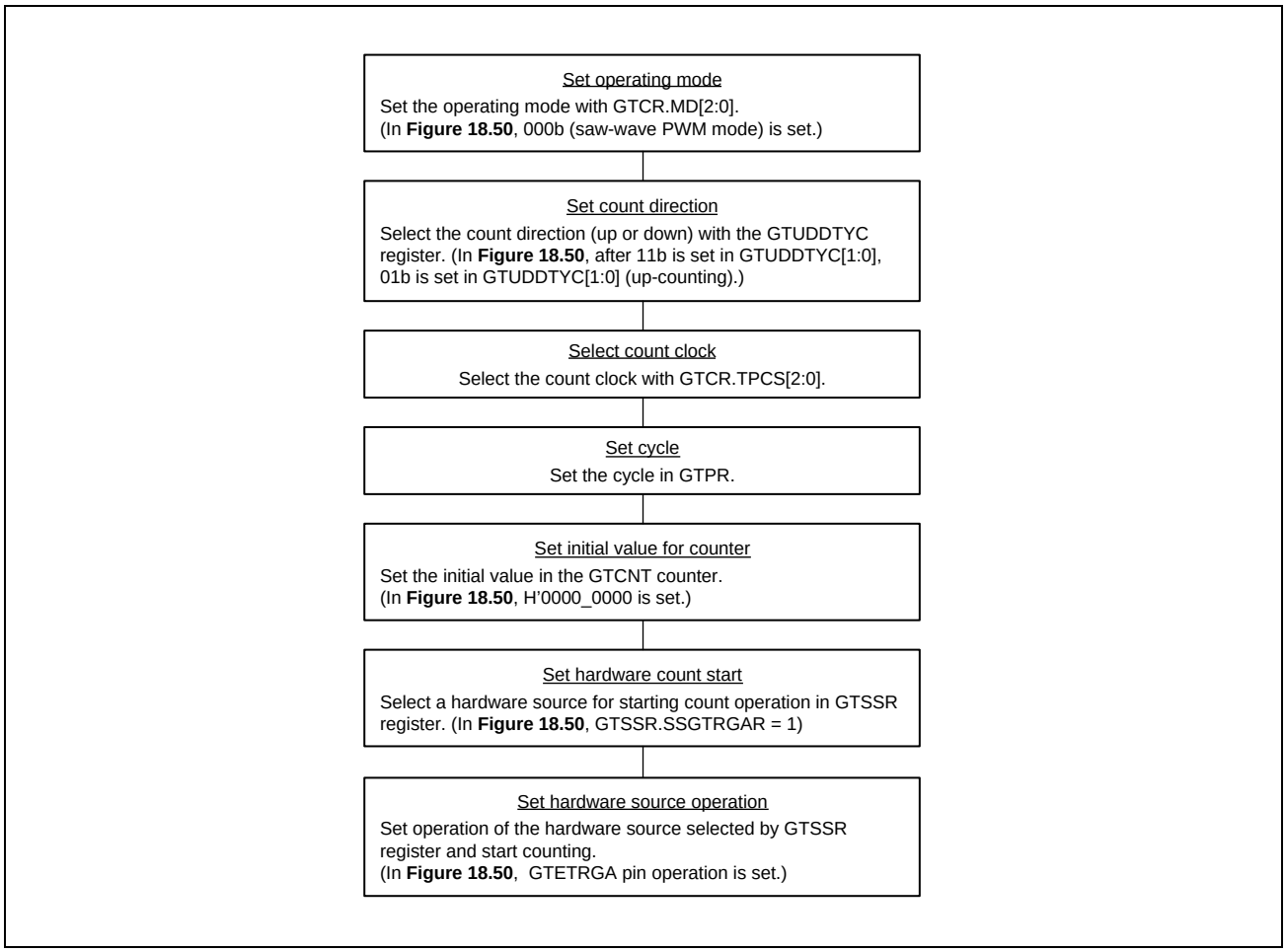


Figure 18.51 Example setting for count start operation by hardware source

18.3.7.2 Hardware Stop Operation

The GTCNT counter can be stopped by selecting a hardware source using GTPSR. **Figure 18.52** shows an example of a count stop operation by a hardware source. **Figure 18.53** shows the setting example. In this example, the count operation stops at the rising edge of GTETRGA pin input and restarts at the rising edge of GTETRGB pin input.

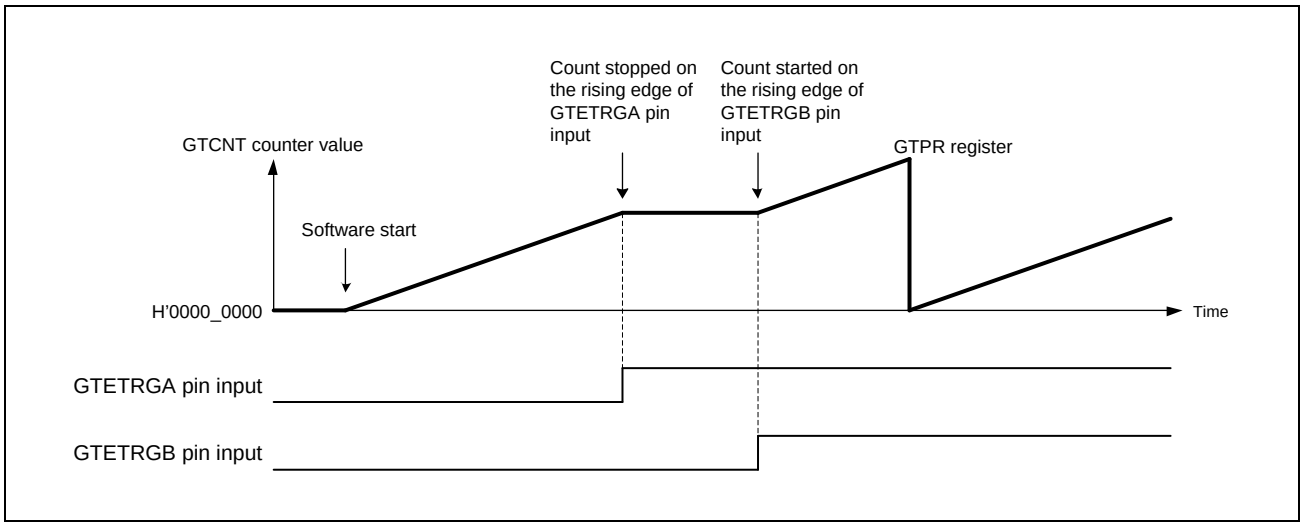


Figure 18.52 Example of count start operation by hardware source, started by software, stopped on the rising edge of GTETRGA pin input, and restarted on the rising edge of GTETRGB pin input

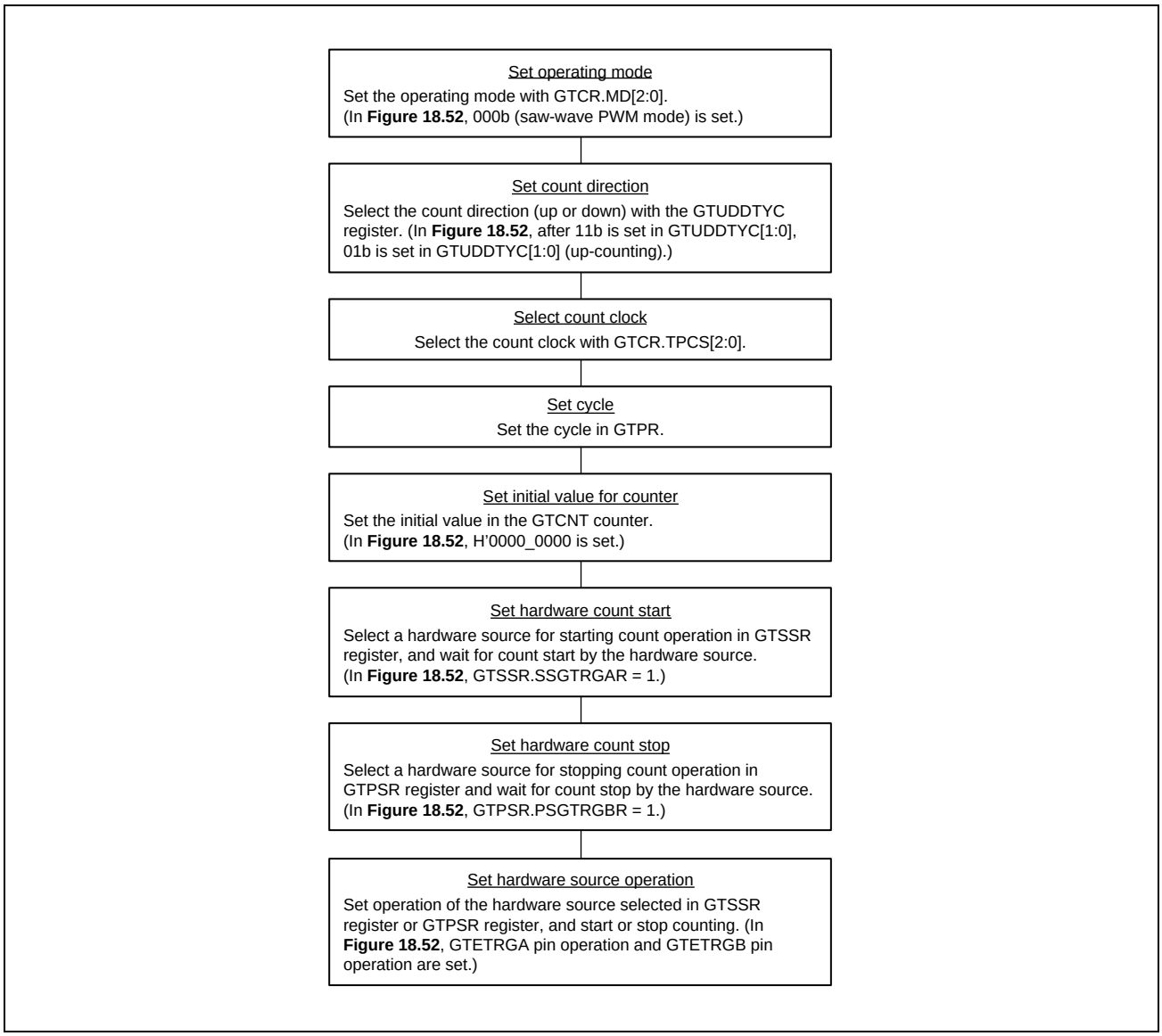


Figure 18.53 Example setting for count stop operation by hardware source

Figure 18.54 shows an example of a count start/stop operation by a hardware source. **Figure 18.55** shows the setting example. In this example, the counter operates during the high-level periods of the external trigger input GTETRGA.

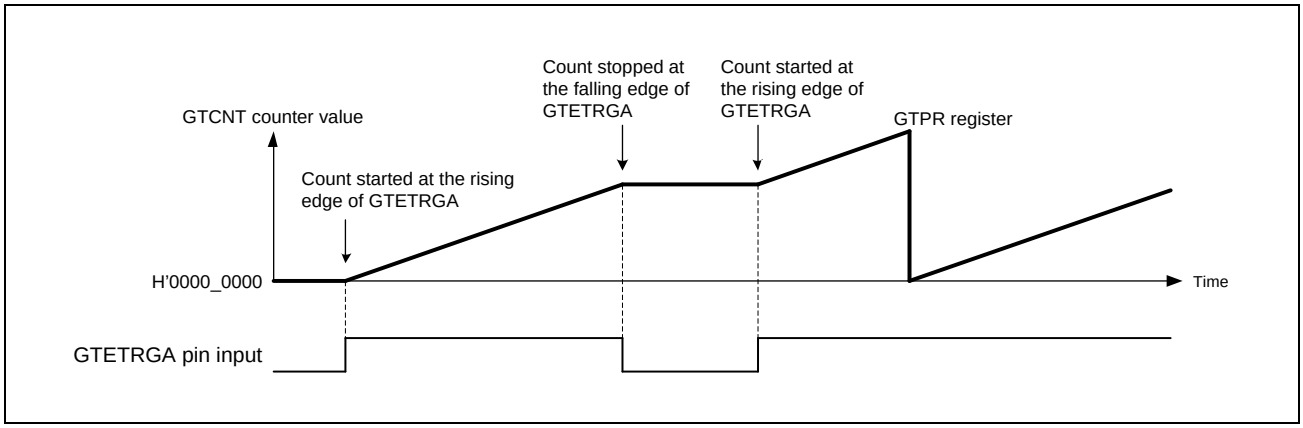


Figure 18.54 Example of count start/stop operation by hardware source, started on the rising edge of the GTETRGA pin input and stopped on the falling edge of the GTETRGA pin input

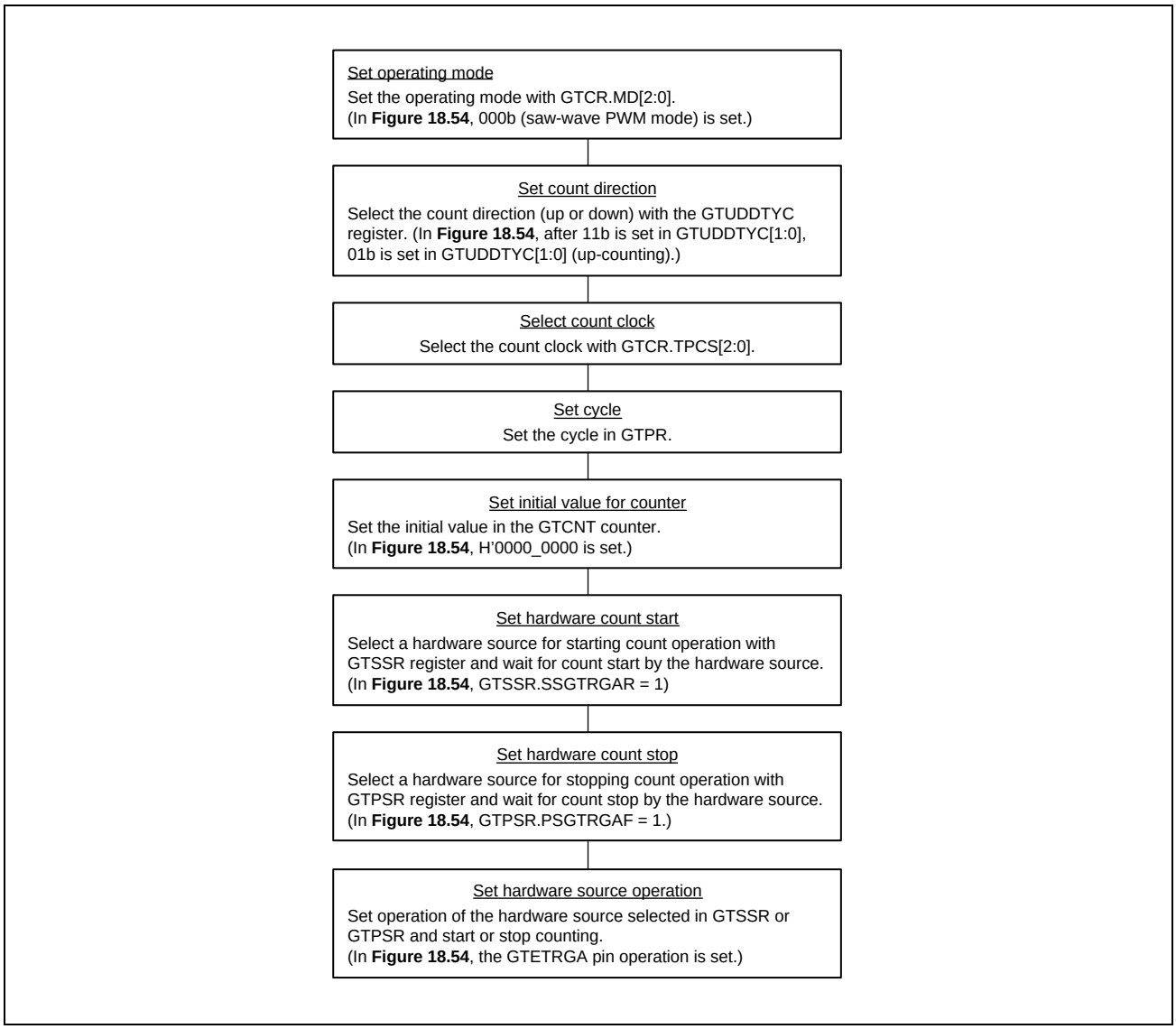


Figure 18.55 Example setting for count start/stop operation by hardware source

18.3.7.3 Hardware Clear Operation

The GTCNT counter can be cleared by selecting a hardware source using GTCSR. The OVFn/UNFn (n=0 to 7) interrupt (overflow/underflow interrupt) is not generated when the GTCNT counter is cleared by a hardware source or by software.

Figure 18.56 and **Figure 18.57** show examples of the GTCNT counter clearing operation by a hardware source. **Figure 18.58** shows the setting example. In this example, the GTCNT counter starts at the rising edge of GTETRGA pin input, and the counter stops/clears at the falling edge of GTETRGA pin input.

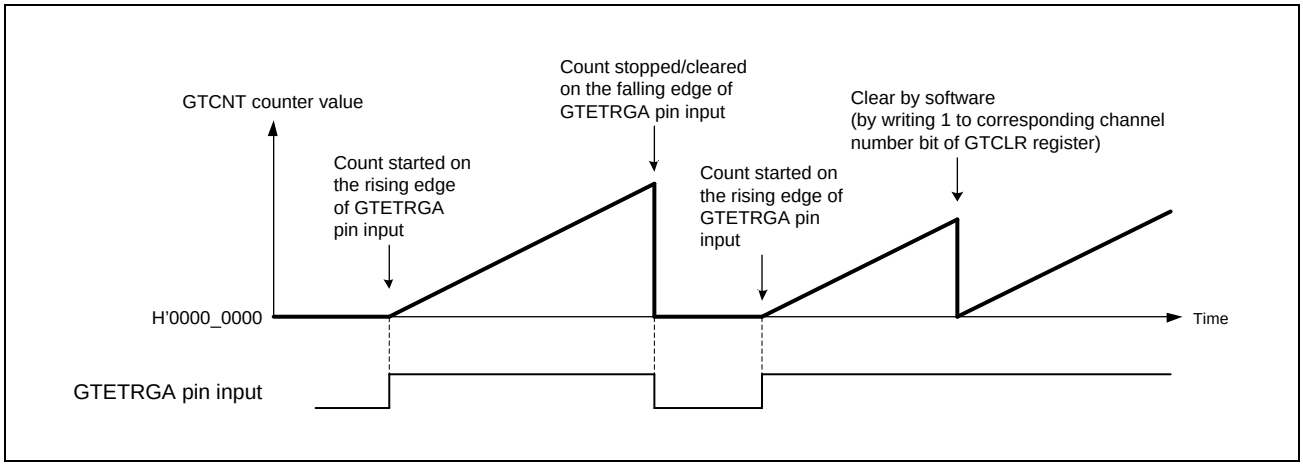


Figure 18.56 Example of count clearing operation by hardware source with saw wave up-counting, started on the rising edge of GTETRGA pin input, and stopped/cleared on the falling edge of GTETRGA pin input

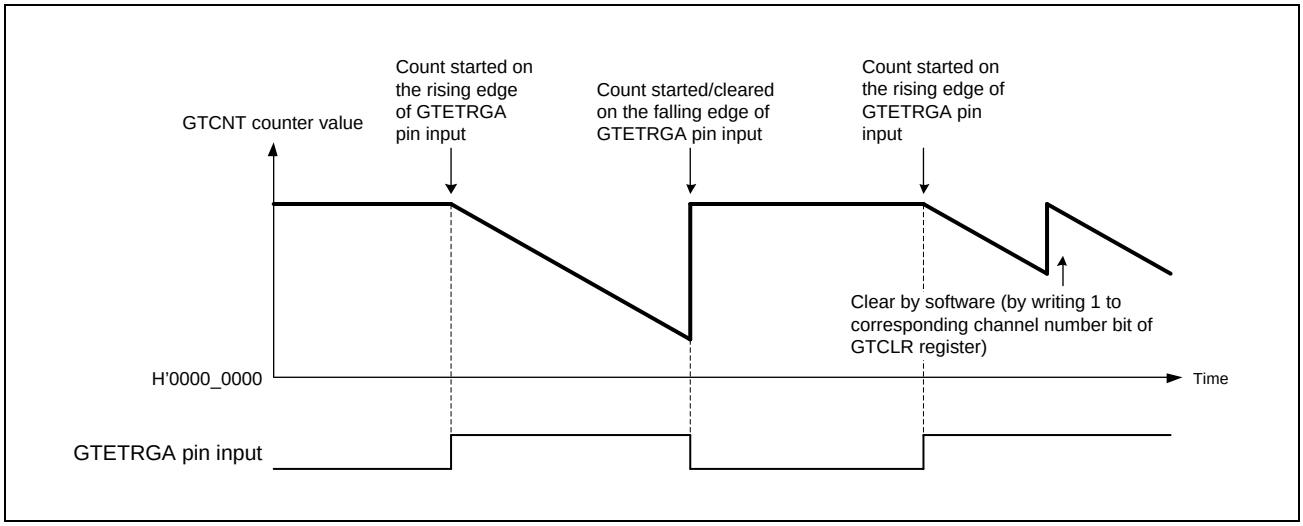


Figure 18.57 Examples of count clearing operation by hardware source with saw wave down-counting, started at event input A, and stopped/cleared at event input B

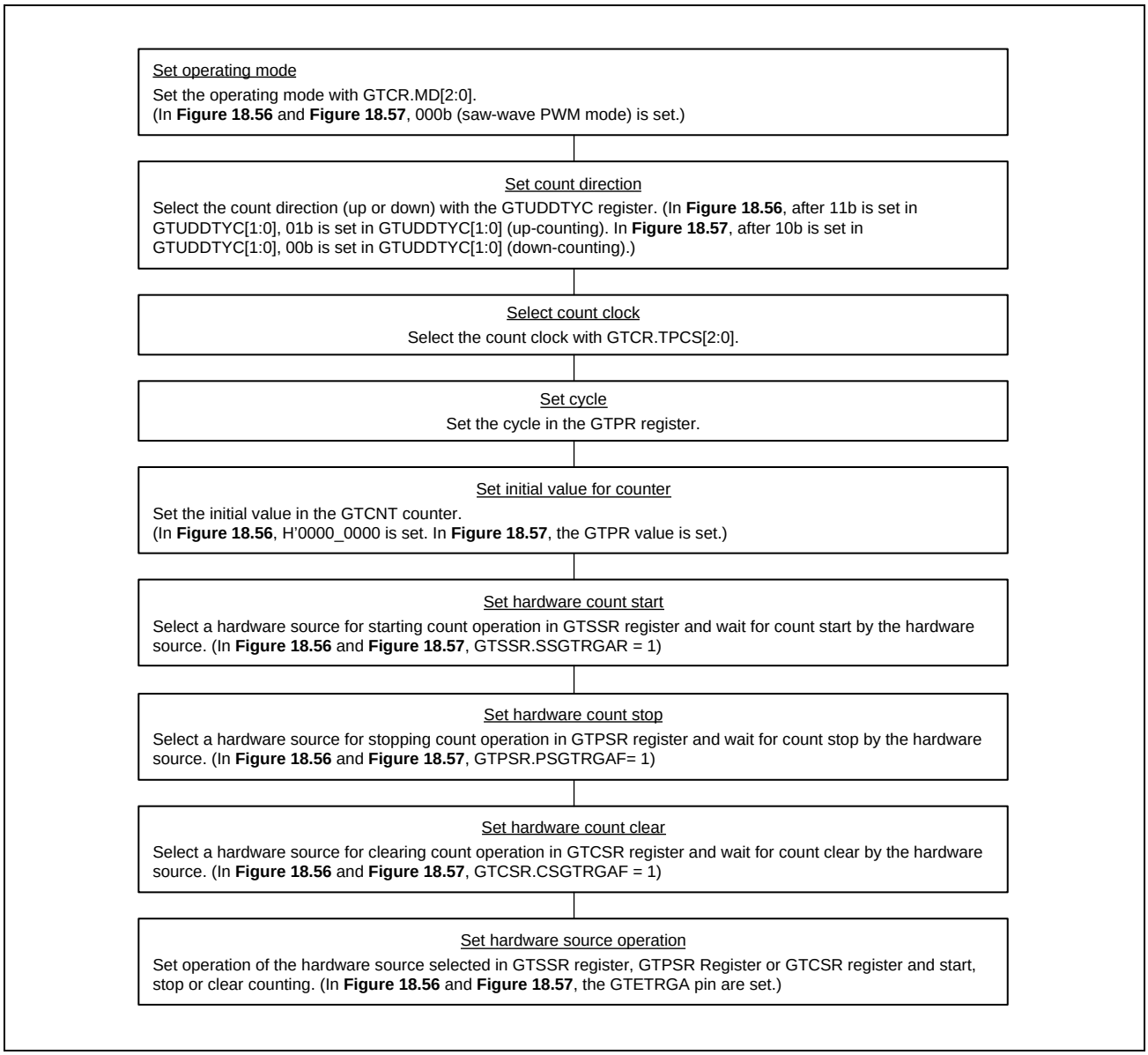


Figure 18.58 Example setting for count clearing operation by hardware source

The OVFn/UNFn (n=0 to 7) interrupt (overflow/underflow interrupt) is not generated when the counter is cleared by a hardware source or by software.

Figure 18.59 shows the relationship between the counter clearing by a hardware source and the OVFn (n=0 to 7) interrupt.

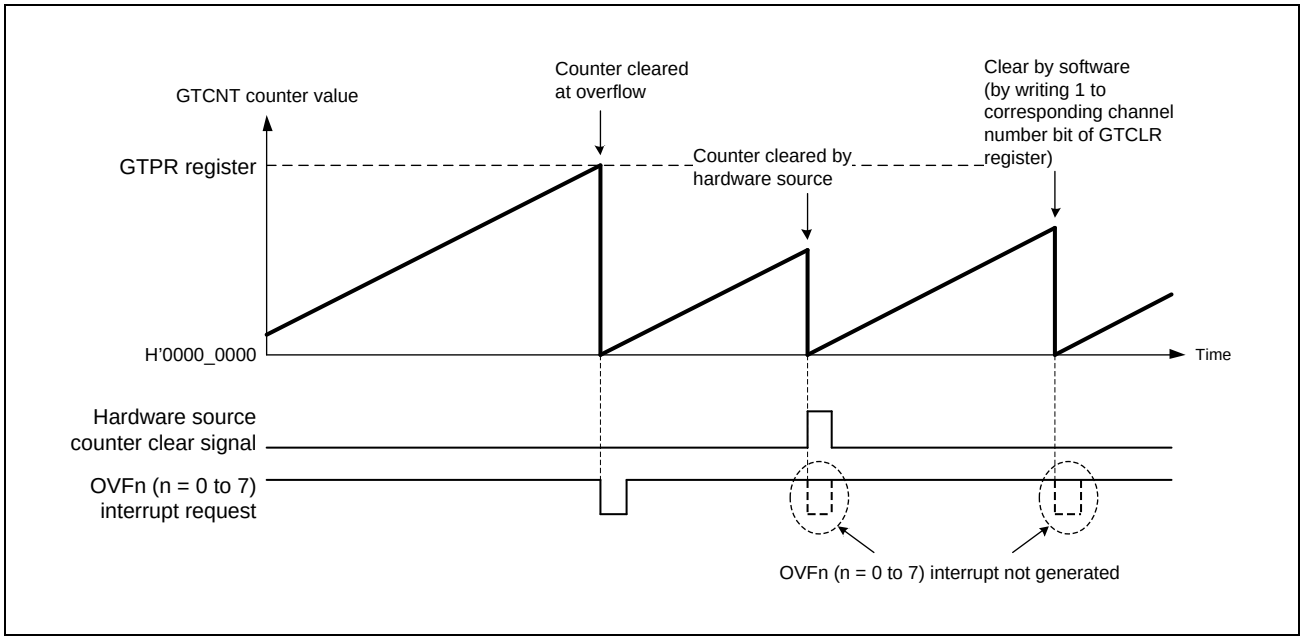


Figure 18.59 Relationship between counter clearing by hardware source and OVFn (n = 0 to 7) interrupt

18.3.8 Synchronized Operation

Synchronized operation on channels such as a synchronized start, stop and clear operation can be performed.

18.3.8.1 Synchronized Operation by Software

The GTCNT counters can be started, stopped, and cleared on multiple channels by setting the associated GTSTR, GTSTP or GTCLR bits simultaneously to 1.

Count start with a phase difference is possible by setting the initial value in the GTCNT counter and setting the associated GTSTR bits simultaneously to 1.

Figure 18.60 shows an example of a simultaneous start, stop and clear by software. **Figure 18.61** shows an example of phase start operation by software.

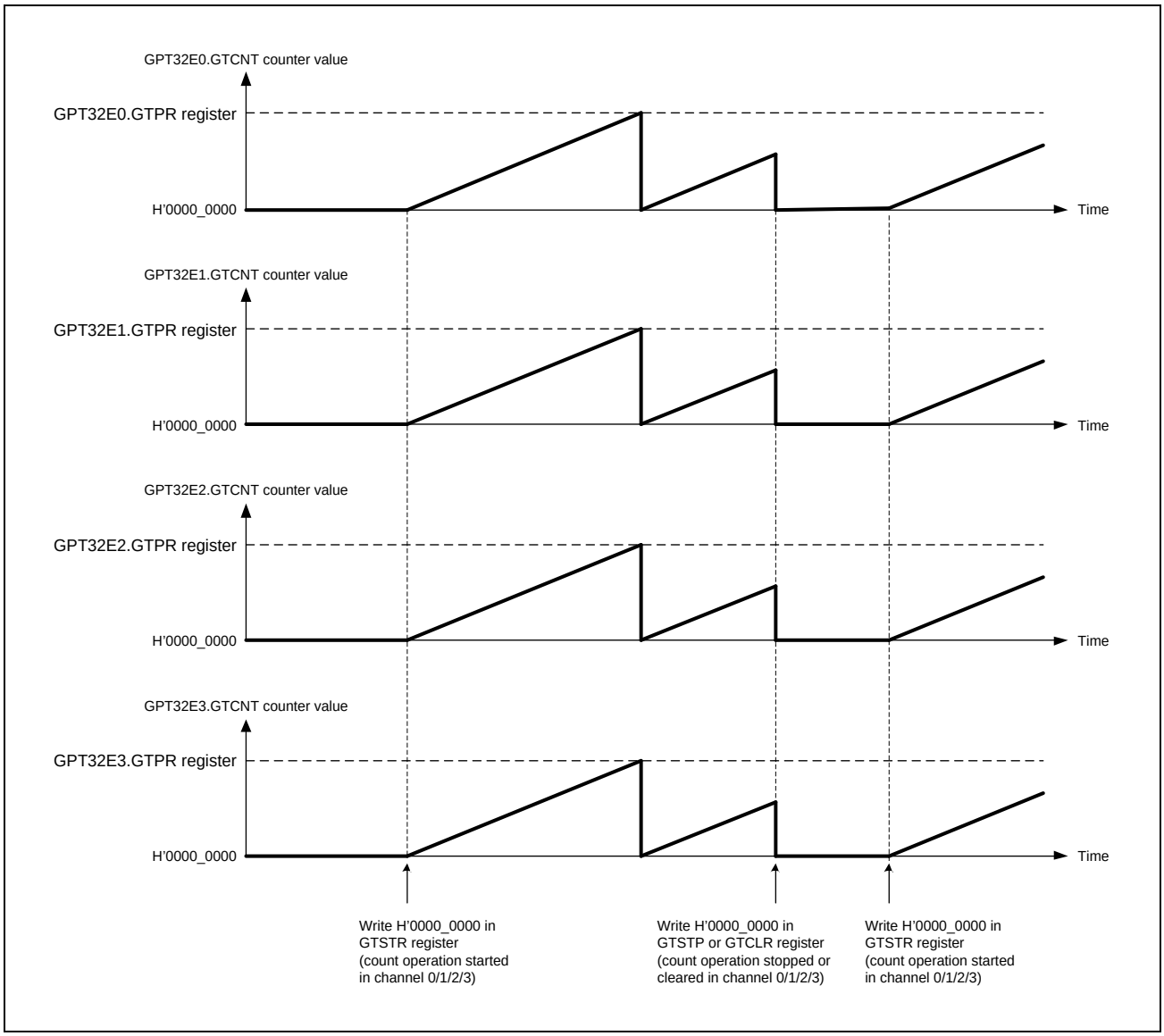


Figure 18.60 Example of a simultaneous start, stop, and clear by software, with the same count cycle (GTPR register value)

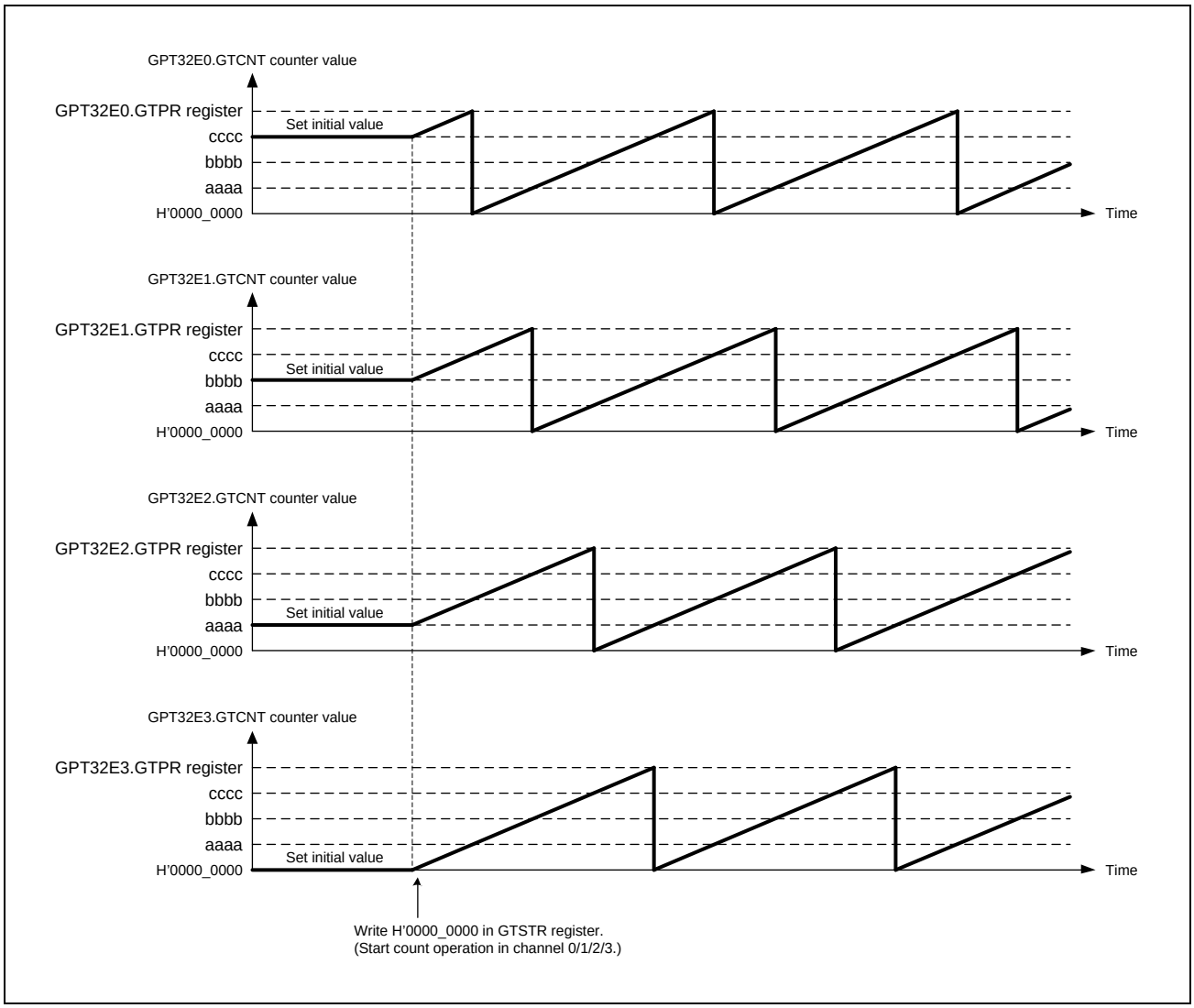


Figure 18.61 Example of software phase start with the same count cycle (GTPR register value)

18.3.8.2 Synchronized Operation by Hardware

The GTCNT counters can be started simultaneously by the following hardware sources:

- External trigger input

Figure 18.62 shows an example of a simultaneous start, stop and clear operation by a hardware source. **Figure 18.63** shows the setting example.

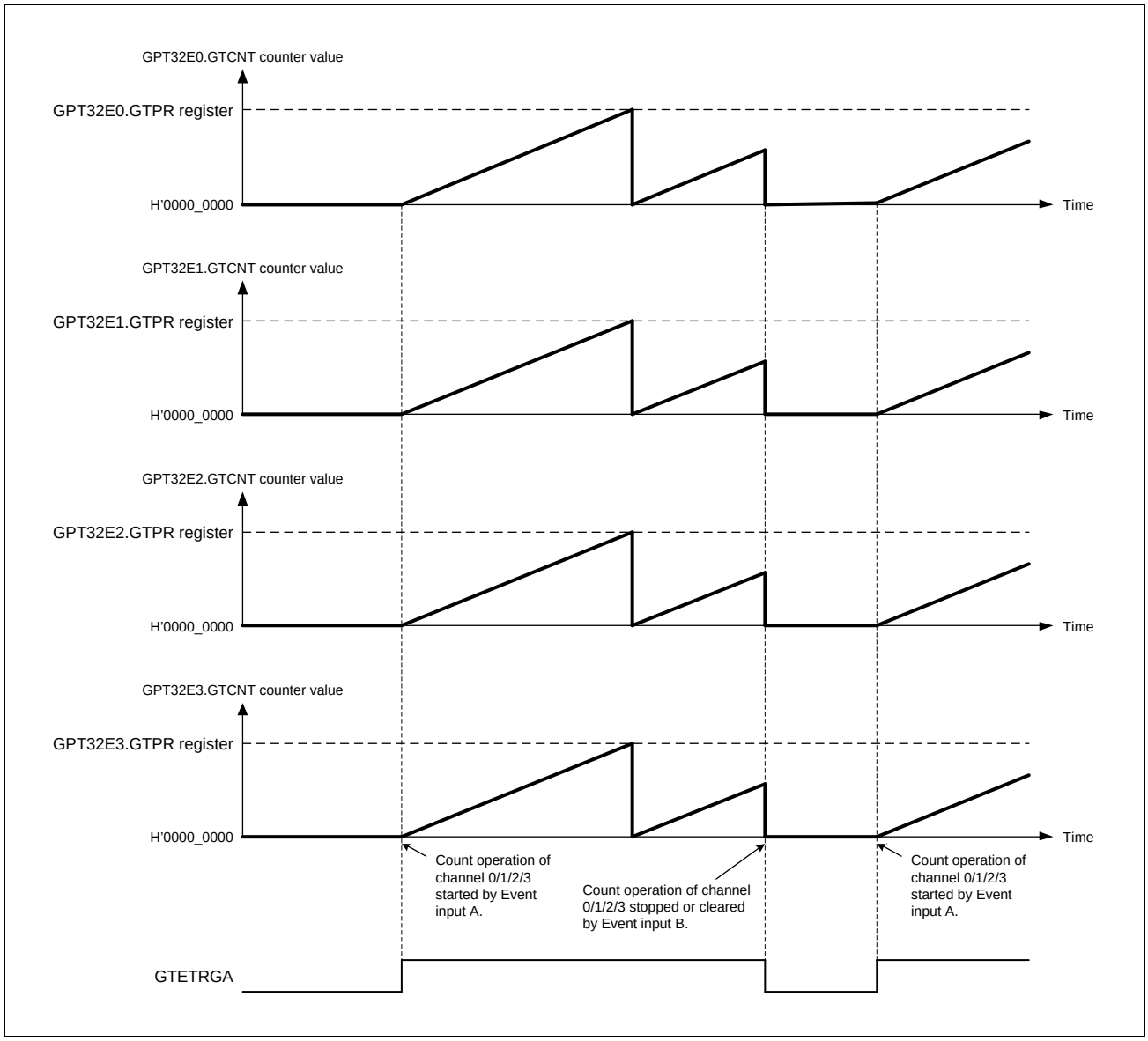


Figure 18.62 Example of a simultaneous start, stop, and clear by the hardware sources, with the same count cycle (GTPR register value)

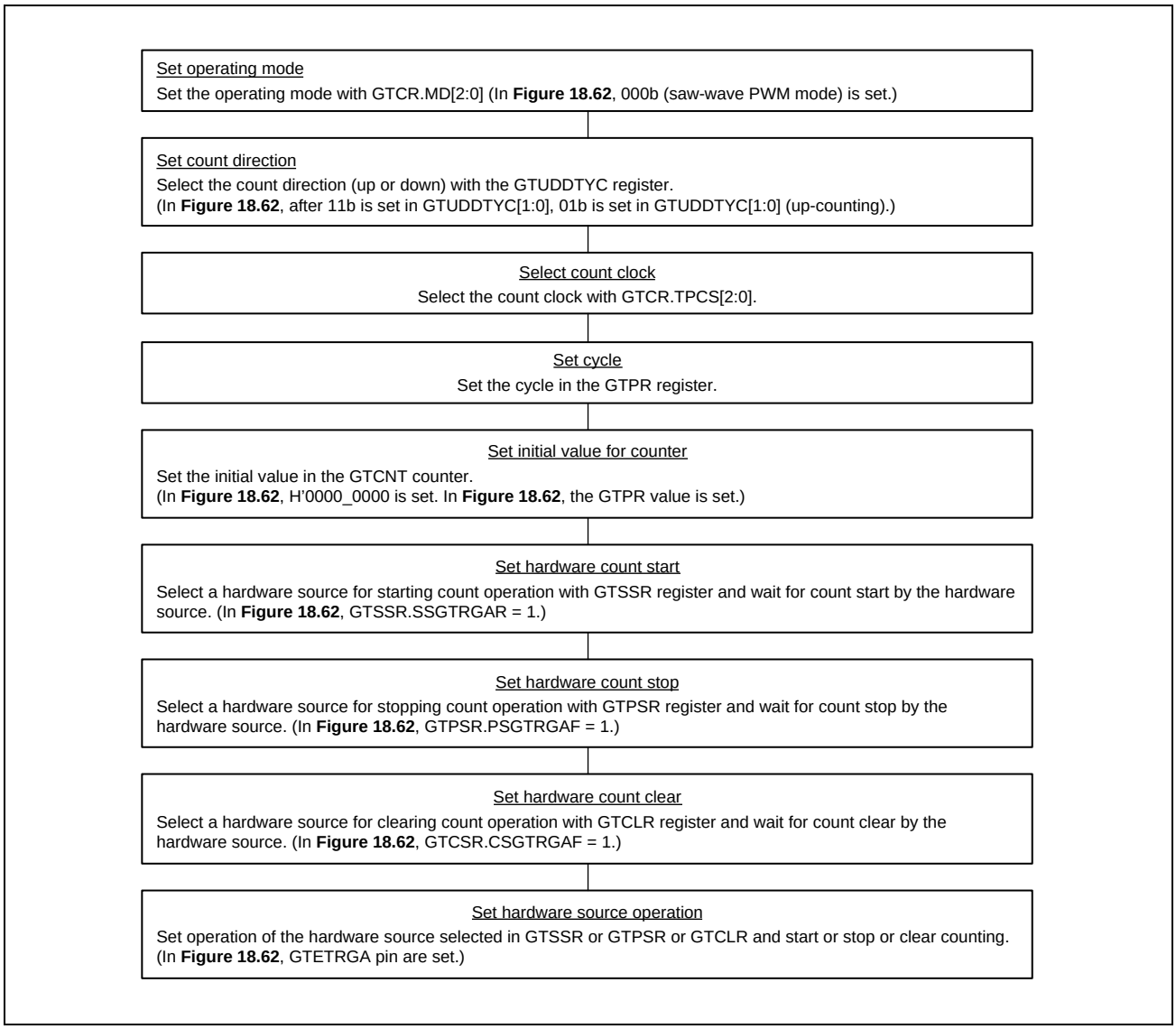


Figure 18.63 Example setting for simultaneous start by hardware source

18.3.9 PWM Output Operation Examples

18.3.9.1 Synchronized PWM Output

The GPT output 16 phases of linked PWM waveforms for a maximum of 8 channels by synchronizing operation on channels.

Figure 18.64 shows an example in which four channels perform synchronized operation in saw-wave PWM mode and eight phases of PWM waveforms are output. The GTIOCA is set so that it outputs low as the initial value, high at a GTCCRA compare match, and low at the cycle end. The GTIOCB is set so that it outputs low as the initial value, high at a GTCCRB compare match, and low at the cycle end.

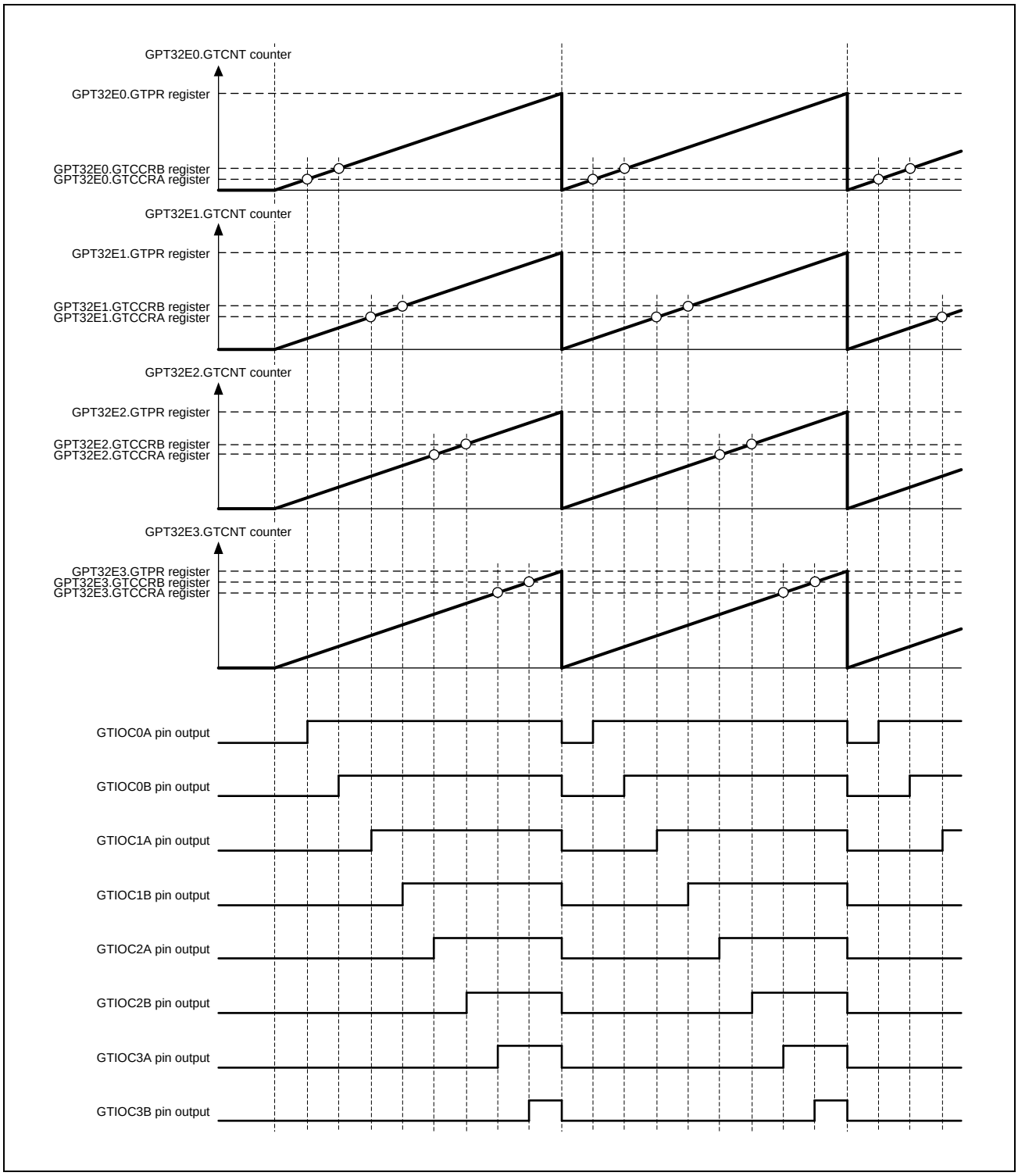


Figure 18.64 Example of synchronized PWM output

18.3.9.2 Three-Phase Saw-Wave Complementary PWM Output

Figure 18.65 shows an example in which three channels perform synchronized operation in saw-wave PWM mode and 3-phase complementary PWM waveforms are output. The GTIOCA pin is set so that it outputs low as the initial value, high at a GTCCRA compare match, and low at the cycle end. The GTIOCB pin is set so that it outputs high as the initial value, low at a GTCCRB compare match, and high at the cycle end.

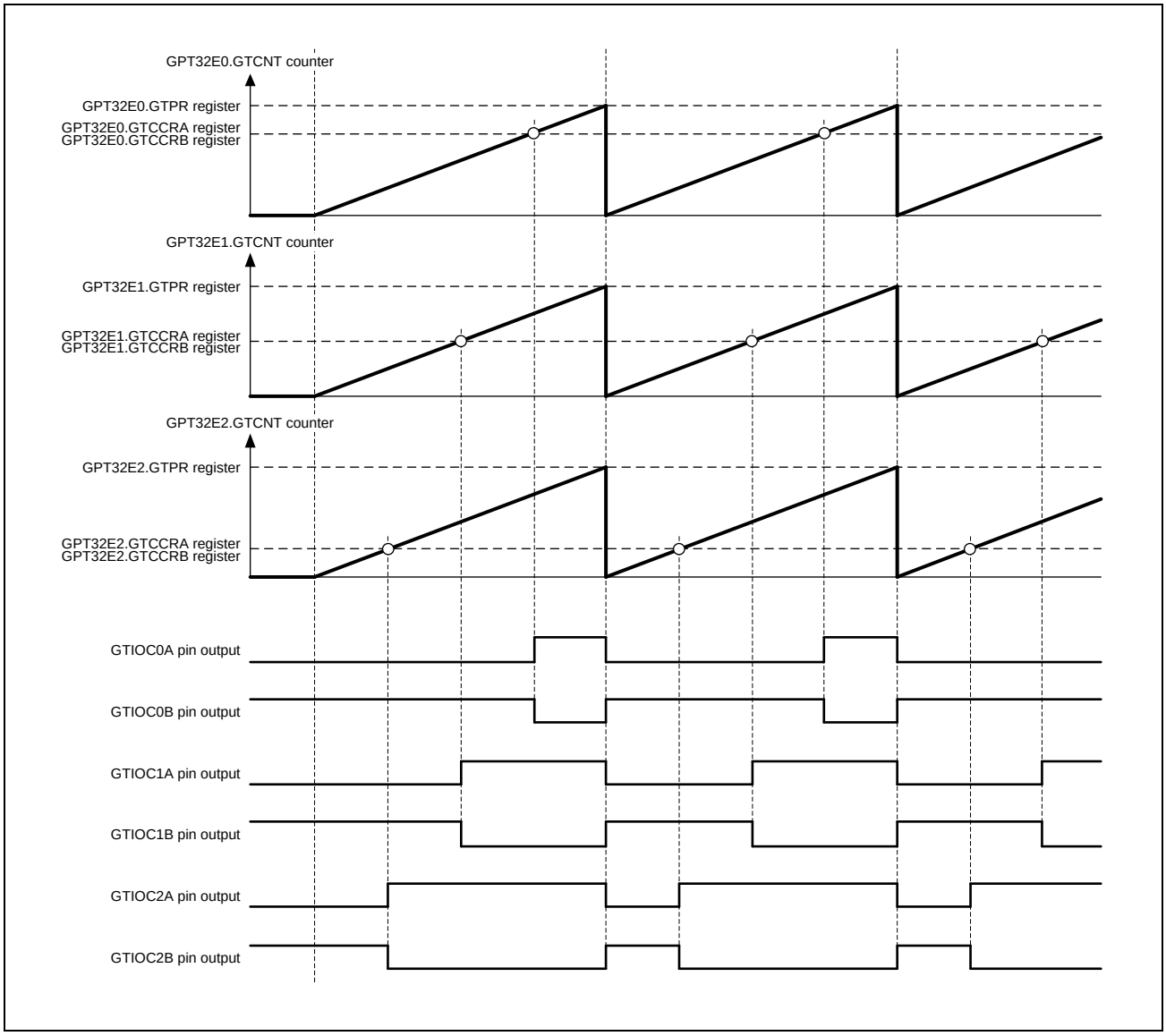


Figure 18.65 Example of 3-phase saw-wave complementary PWM output

18.3.9.3 3-Phase Saw-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 18.66 shows an example in which three channels perform synchronized operation in saw-wave one-shot pulse mode with automatic dead time setting and 3-phase complementary PWM waveforms are output. The GTIOCA pin is set so that it outputs low as the initial value, toggle the output at a GTCCRA compare match, and retain the output at the cycle end. The GTIOCB pin is set so that it outputs high as the initial value, toggle the output at a GTCCRB compare match, and retain the output at the cycle end.

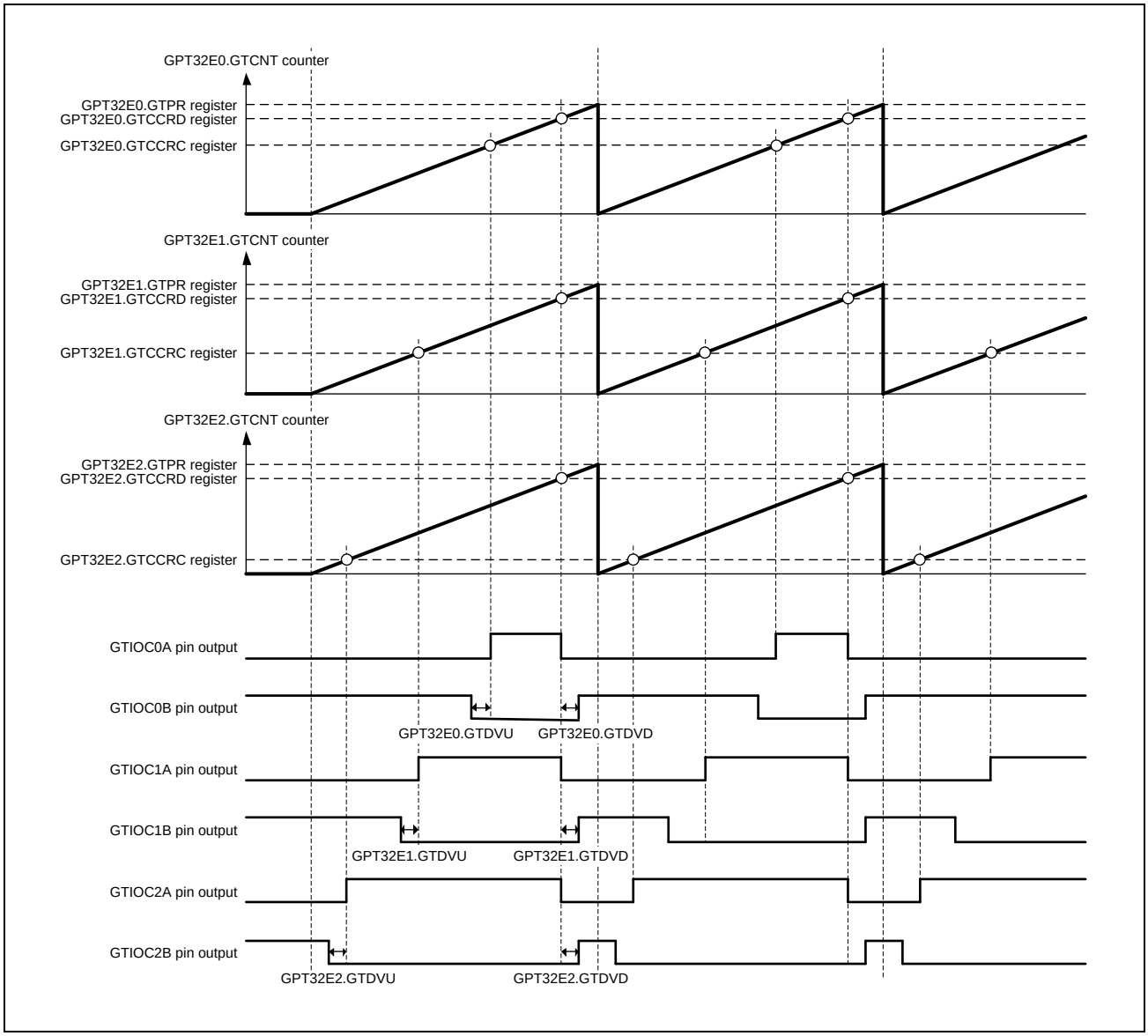


Figure 18.66 Example of 3-phase saw-wave complementary PWM output with automatic dead time setting

18.3.9.4 3-Phase Triangle-Wave Complementary PWM Output

Figure 18.67 shows an example in which three channels perform synchronized operation in triangle-wave PWM mode 1 and 3-phase complementary PWM waveforms are output. The GTIOCA pin is set so that it outputs low as the initial value, toggles the output at a GTCCRA compare match, and retains the output at the cycle end. The GTIOCB pin is set so that it outputs high as the initial value, toggles the output at a GTCCRB compare match, and retains the output at the cycle end.

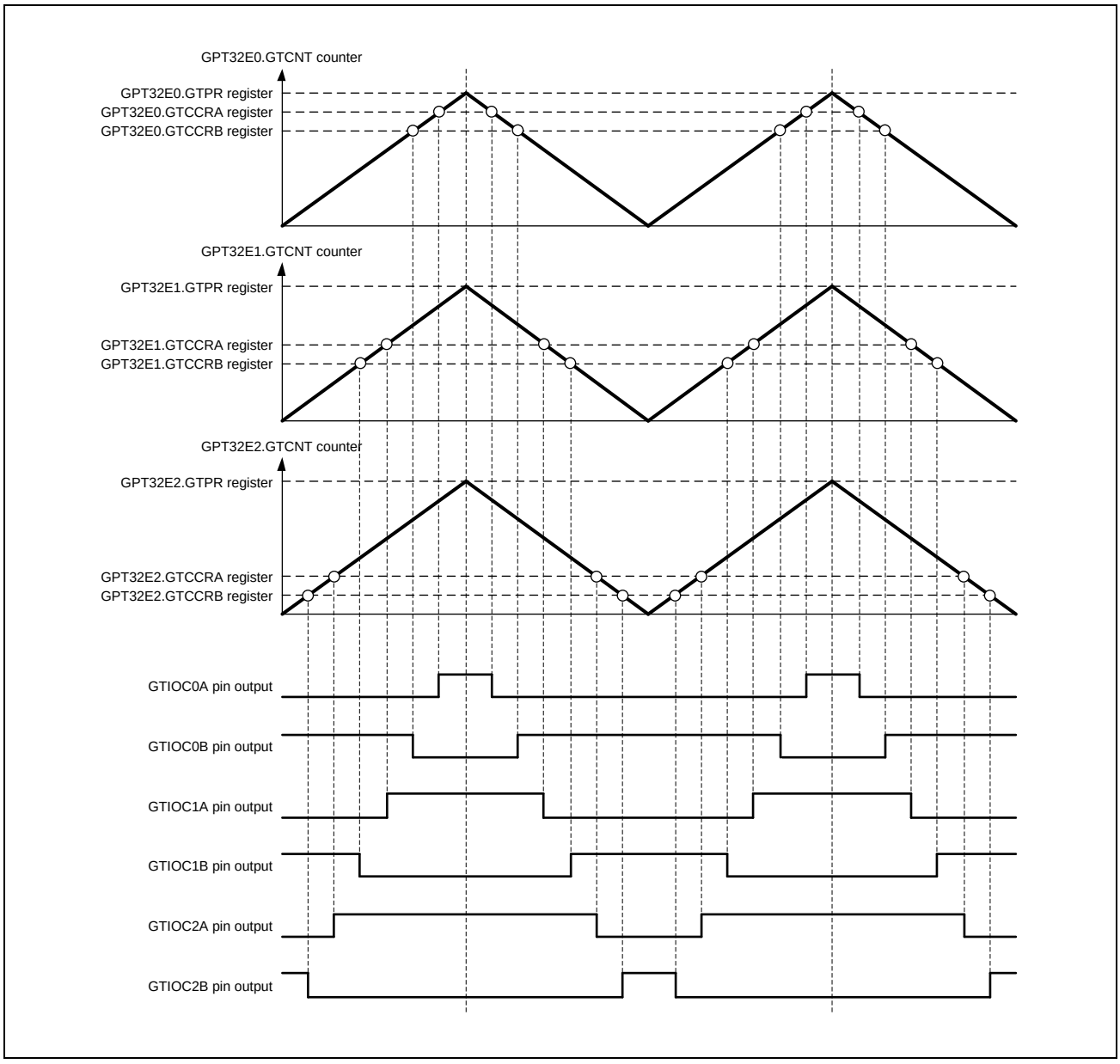


Figure 18.67 Example of 3-phase triangle-wave complementary PWM output

18.3.9.5 3-Phase Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 18.68 shows an example in which three channels perform synchronized operation in triangle-wave PWM mode 1 with automatic dead time setting and 3-phase complementary PWM waveforms are output. The GTIOCA pin is set so that it outputs low as the initial value, toggles the output at a GTCCRA compare match, and retains the output at the cycle end. The GTIOCB pin is set so that it outputs high as the initial value, toggles the output at a GTCCRB compare match, and retains the output at the cycle end.

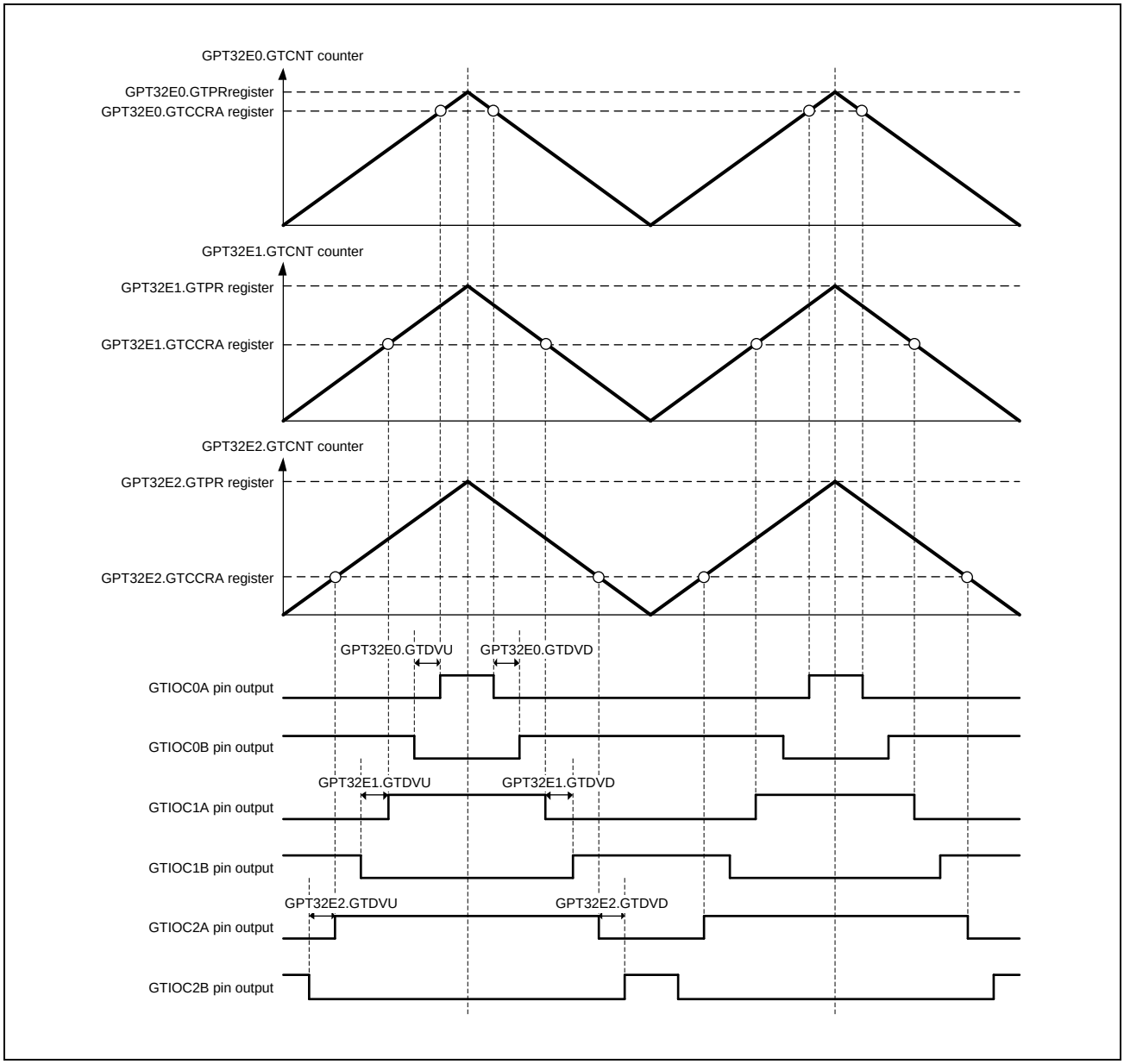


Figure 18.68 Example of 3-phase triangle-wave complementary PWM output with automatic dead time setting

18.3.9.6 3-Phase Asymmetric Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 18.69 shows an example in which three channels perform synchronized operation in triangle-wave PWM mode 3 with automatic dead time setting and 3-phase complementary PWM waveforms are output. The GTIOCA is set so that it outputs low as the initial value, toggles the output at a GTCCRA compare match, and retains the output at the cycle end. The GTIOCB is set so that it outputs high as the initial value, toggles the output at a GTCCRB compare match, and retains the output at the cycle end.

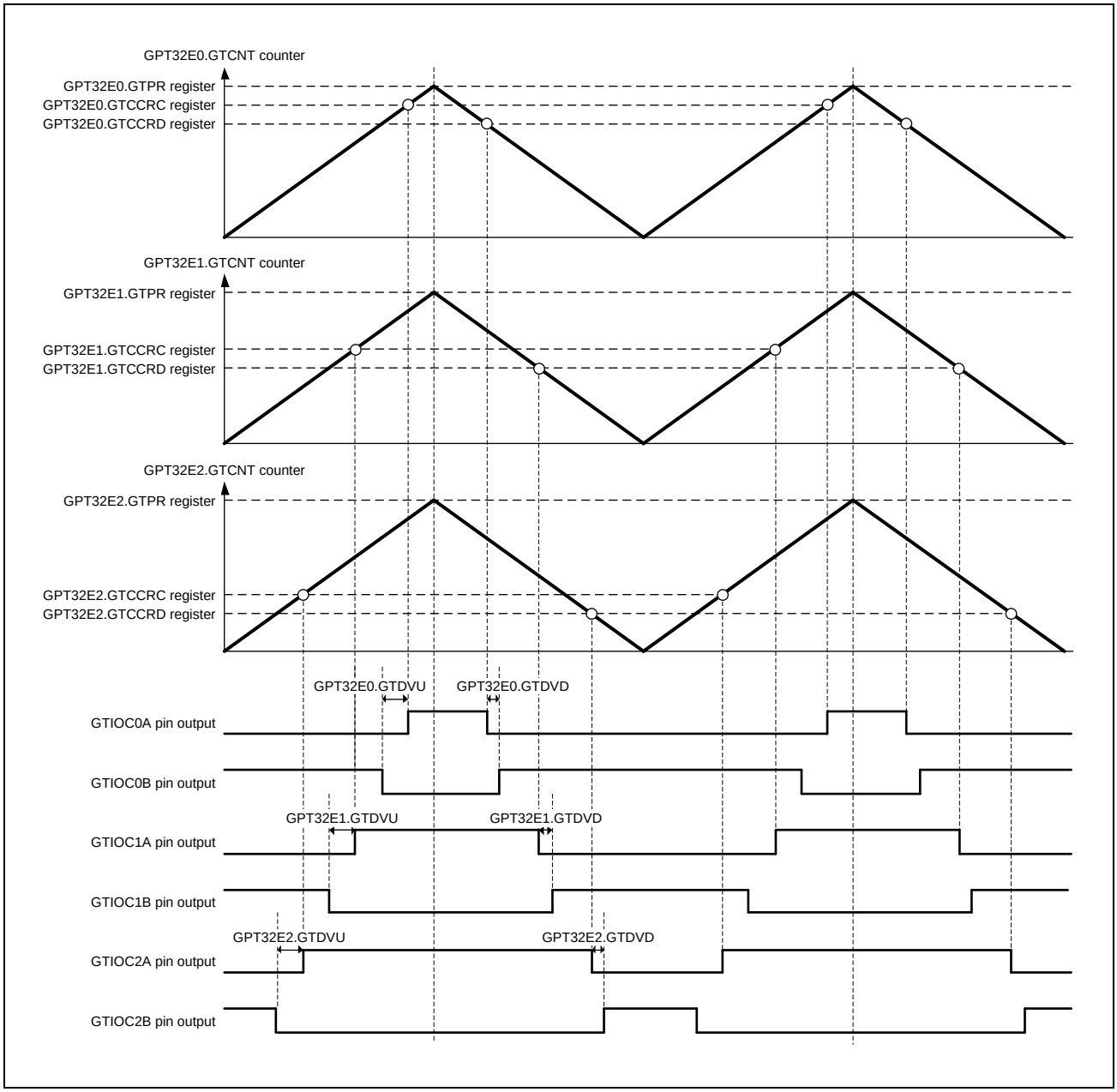


Figure 18.69 Example of 3-phase asymmetric triangle-wave complementary PWM output with automatic dead time setting

18.3.10 Phase Counting Function

The phase difference between the GTIOCA pin input and GTIOCB pin input is detected and the associated GTCNT counts up or counts down. The detectable phase difference is available in any combination with the relationship between the edge and the level of GTIOCA pin and GTIOCB pin input being set in the GTUPSR and GTDNSR registers. For details on count operation, see **Section 18.3.1.1, Counter Operation**.

Figure 18.70 to **Figure 18.79** show phase counting modes 1 to 5. **Table 18.8** to **Table 18.17** show conditions of up- counting or down-counting and lists settings for the GTUPSR and GTDNSR registers.

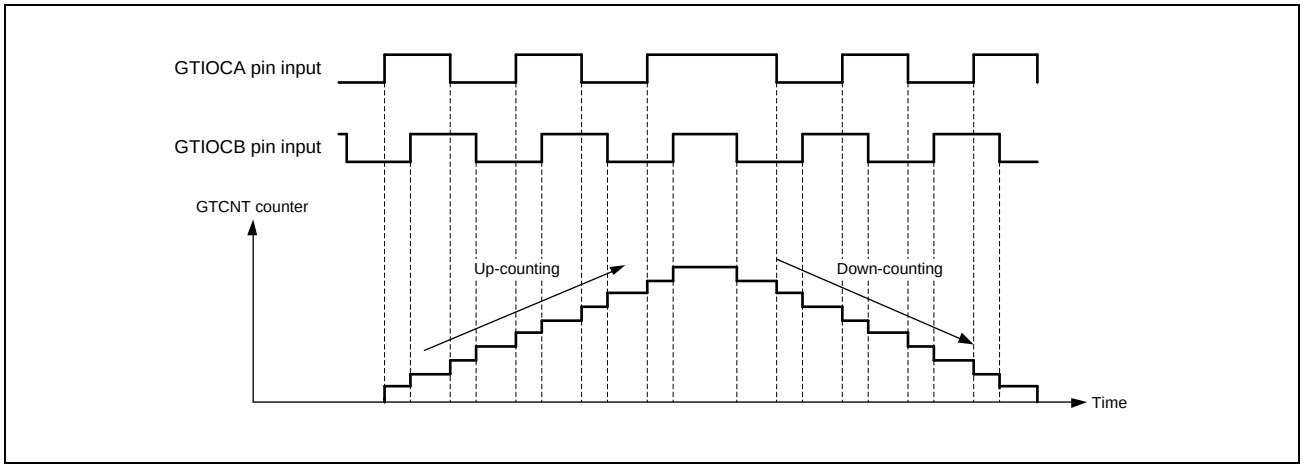


Figure 18.70 Example of phase counting mode 1

Table 18.8 Conditions of up-counting and down-counting in phase counting mode 1

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Up-counting	GTUPSR = H'0000_6900 GTDNSR = H'0000_9600
Low	↓		
↑	Low		
↓	High		
High	↓	Down-counting	
Low	↑		
↑	High		
↓	Low		

Remarks: ↑ : Rising edge
↓ : Falling edge

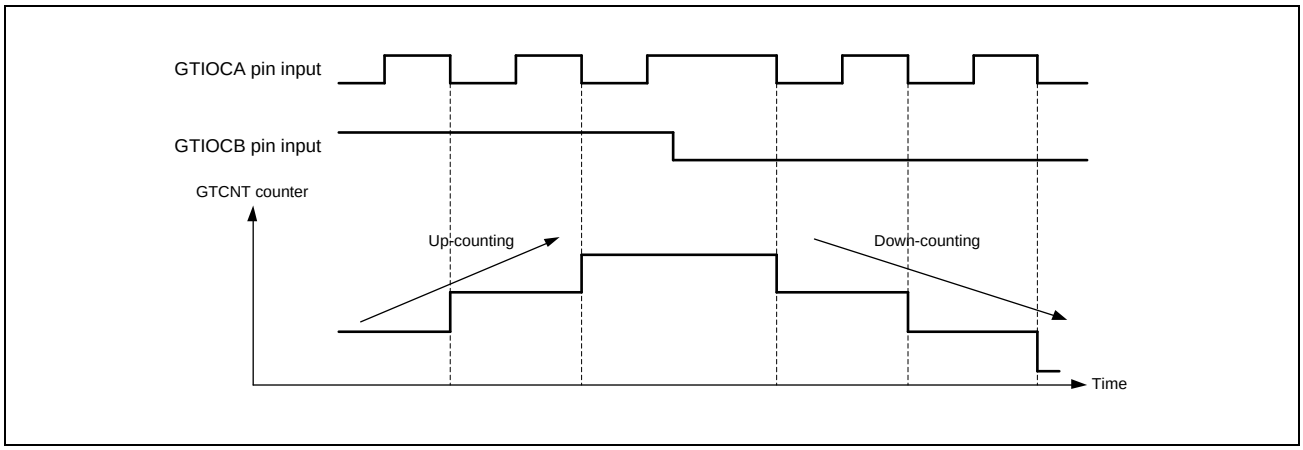


Figure 18.71 Example of phase counting mode 2 (A)

Table 18.9 Conditions of up-counting and down-counting in phase counting mode 2 (A)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Don't care	GTUPSR = H'0000_0800
Low	↓		GTDNSR = H'0000_0400
↑	Low		
↓	High	Up-counting	
High	↓	Don't care	
Low	↑		
↑	High		
↓	Low	Down-counting	

Remarks: ↑ : Rising edge
↓ : Falling edge

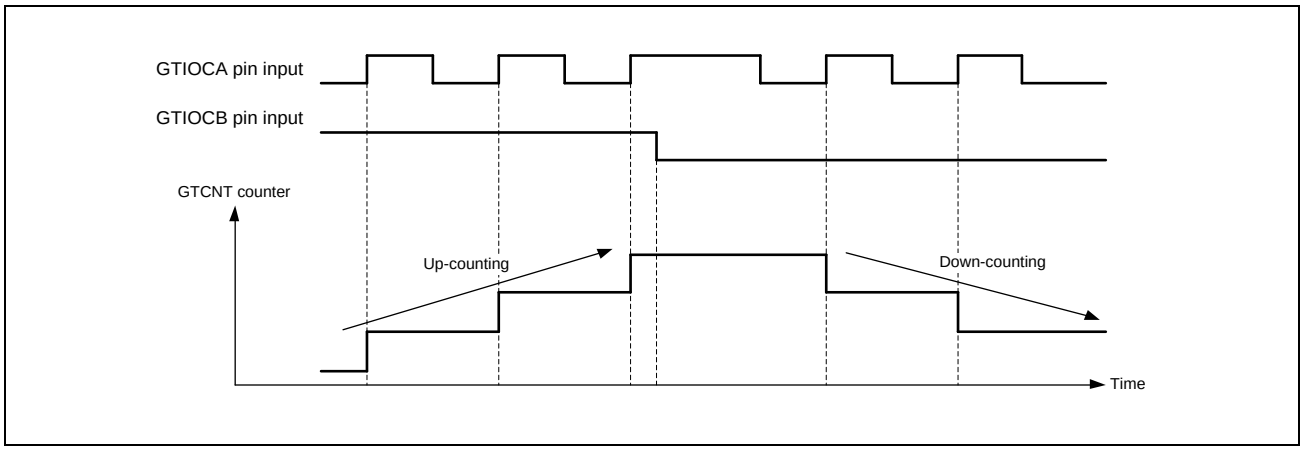


Figure 18.72 Example of phase counting mode 2 (B)

Table 18.10 Conditions of up-counting and down-counting in phase counting mode 2 (B)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Don't care	GTUPSR = H'0000_0200 GTDNSR = H'0000_0100
Low	↓		
↑	Low	Down-counting	
↓	High	Don't care	
High	↓		
Low	↑		
↑	High	Up-counting	
↓	Low	Don't care	

Remarks: ↑ : Rising edge
↓ : Falling edge

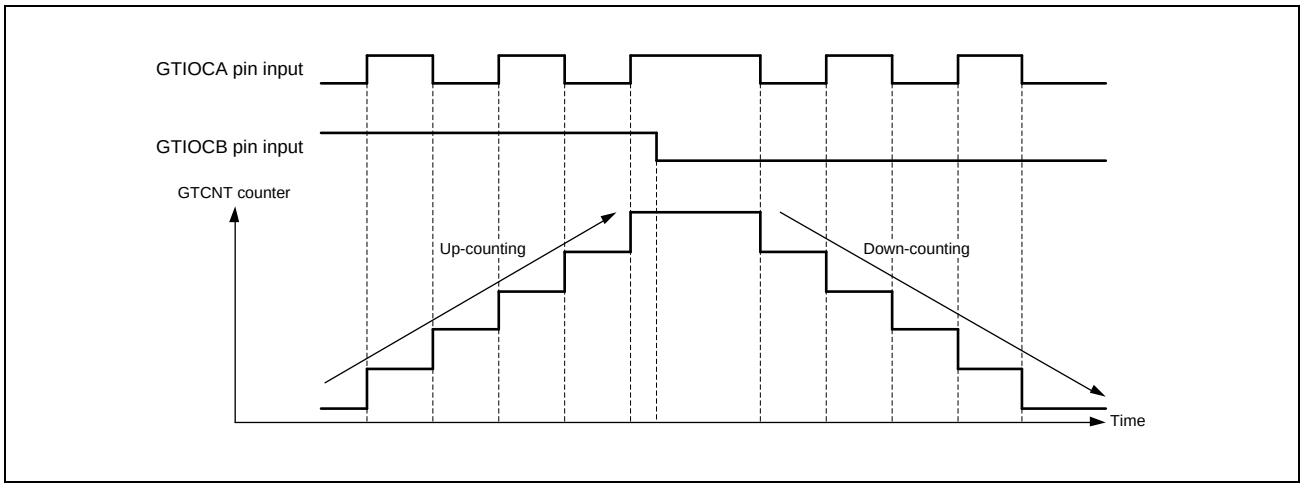


Figure 18.73 Example of phase counting mode 2 (C)

Table 18.11 Conditions of up-counting and down-counting in phase counting mode 2 (C)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Don't care	GTUPSR = H'0000_0A00 GTDNSR = H'0000_0500
Low	↓		
↑	Low	Down-counting	
↓	High	Up-counting	
High	↓	Don't care	
Low	↑		
↑	High	Up-counting	
↓	Low	Down-counting	

Remarks: ↑ : Rising edge
↓ : Falling edge

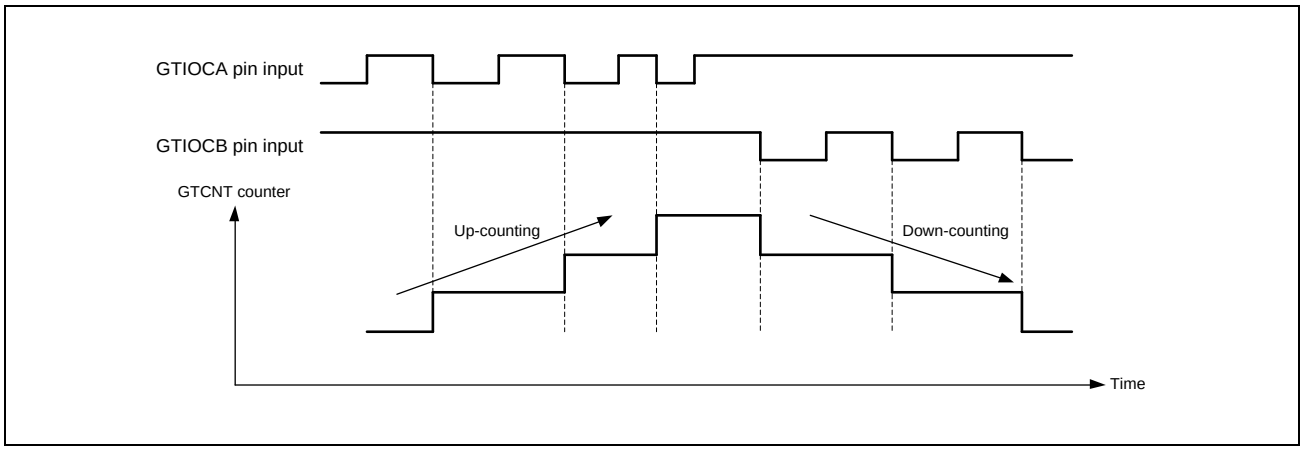


Figure 18.74 Example of phase counting mode 3 (A)

Table 18.12 Conditions of up-counting and down-counting in phase counting mode 3 (A)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Don't care	GTUPSR = H'0000_0800
Low	↓		GTDNSR = H'0000_8000
↑	Low		
↓	High	Up-counting	
High	↓	Down-counting	
Low	↑	Don't care	
↑	High		
↓	Low		

Remarks: ↑ : Rising edge
↓ : Falling edge

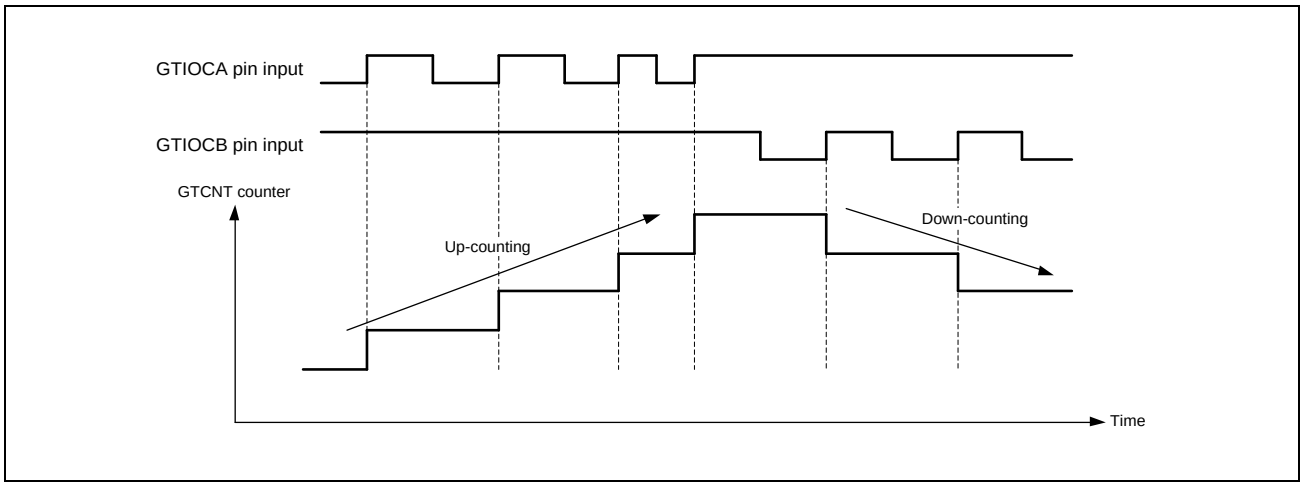


Figure 18.75 Example of phase counting mode 3 (B)

Table 18.13 Conditions of up-counting and down-counting in phase counting mode 3 (B)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Down-counting	GTUPSR = H'0000_0200 GTDNSR = H'0000_2000
Low	↓	Don't care	
↑	Low		
↓	High		
High	↓		
Low	↑		
↑	High	Up-counting	
↓	Low	Don't care	

Remarks: ↑ : Rising edge
↓ : Falling edge

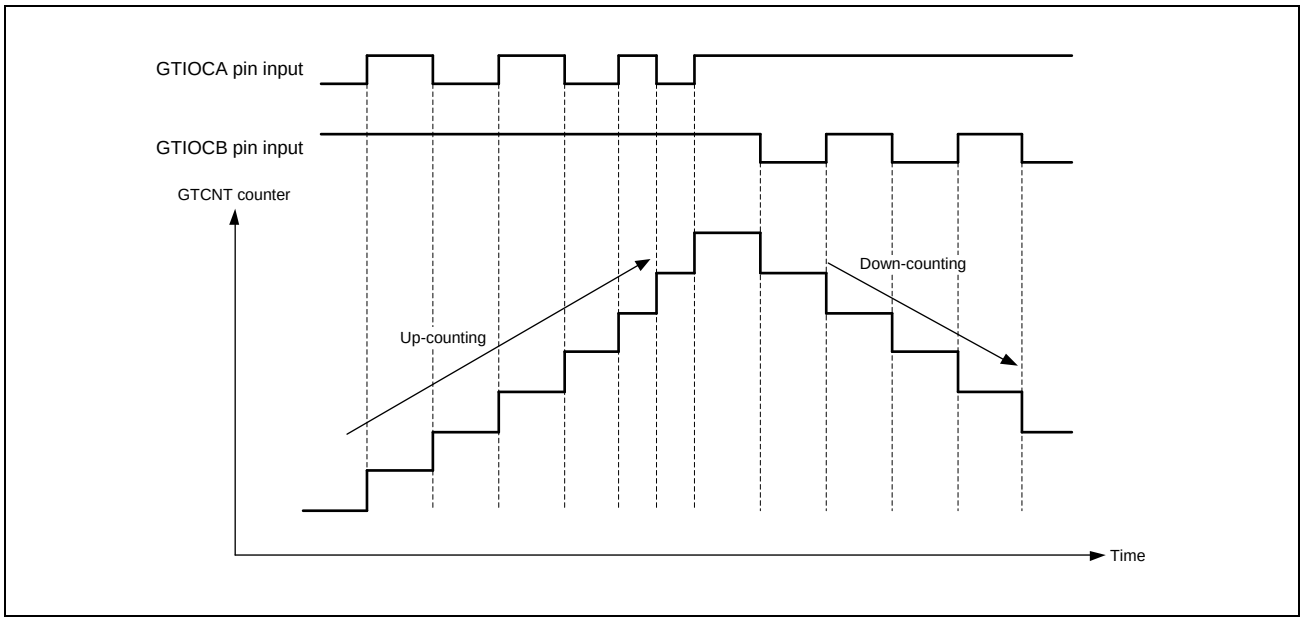


Figure 18.76 Example of phase counting mode 3 (C)

Table 18.14 Conditions of up-counting and down-counting in phase counting mode 3 (C)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Down-counting	GTUPSR = H'0000_0A00 GTDNSR = H'0000_A000
Low	↓	Don't care	
↑	Low		
↓	High	Up-counting	
High	↓	Down-counting	
Low	↑	Don't care	
↑	High	Up-counting	
↓	Low	Don't care	

Remarks: ↑ : Rising edge
↓ : Falling edge

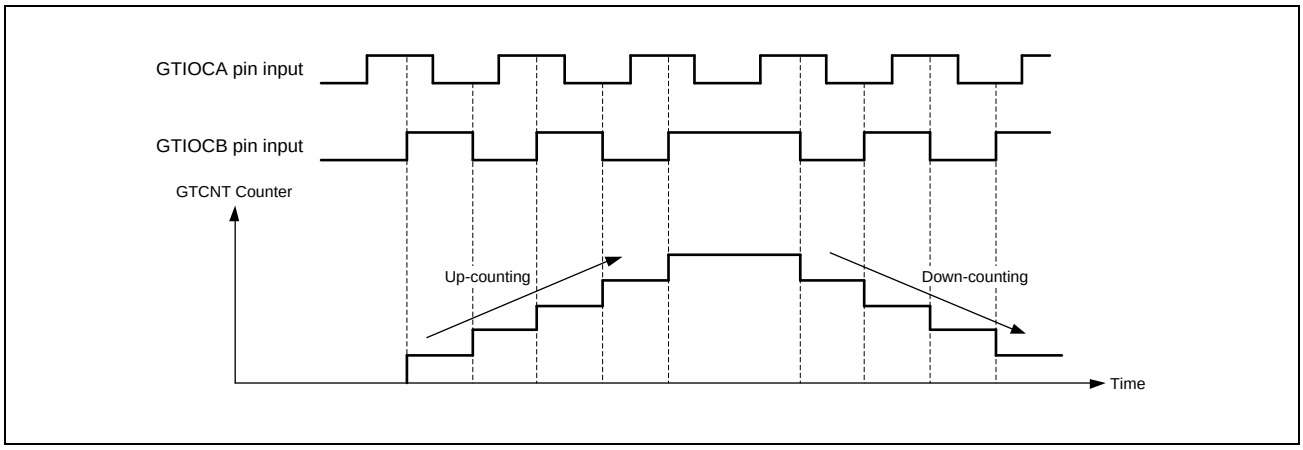


Figure 18.77 Example of phase counting mode 4

Table 18.15 Conditions of up-counting and down-counting in phase counting mode 4

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Up-counting	GTUPSR = H'0000_6000 GTDNSR = H'0000_9000
Low	↓		
↑	Low	Don't care	
↓	High		
High	↓	Down-counting	
Low	↑		
↑	High	Don't care	
↓	Low		

Remarks: ↑ : Rising edge
↓ : Falling edge

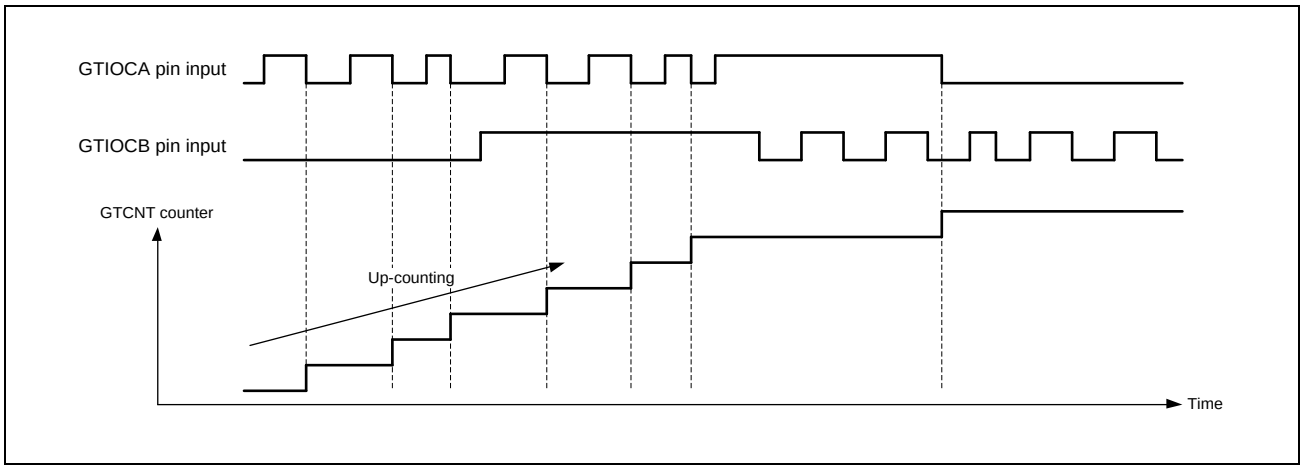


Figure 18.78 Example of phase counting mode 5 (A)

Table 18.16 Conditions of up-counting and down-counting in phase counting mode 5 (A)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Don't care	GTUPSR = H'0000_0C00 GTDNSR = H'0000_0000
Low	↓		
↑	Low		
↓	High	Up-counting	
High	↓	Don't care	
Low	↑		
↑	High		
↓	Low	Up-counting	

Remarks: ↑ : Rising edge
↓ : Falling edge

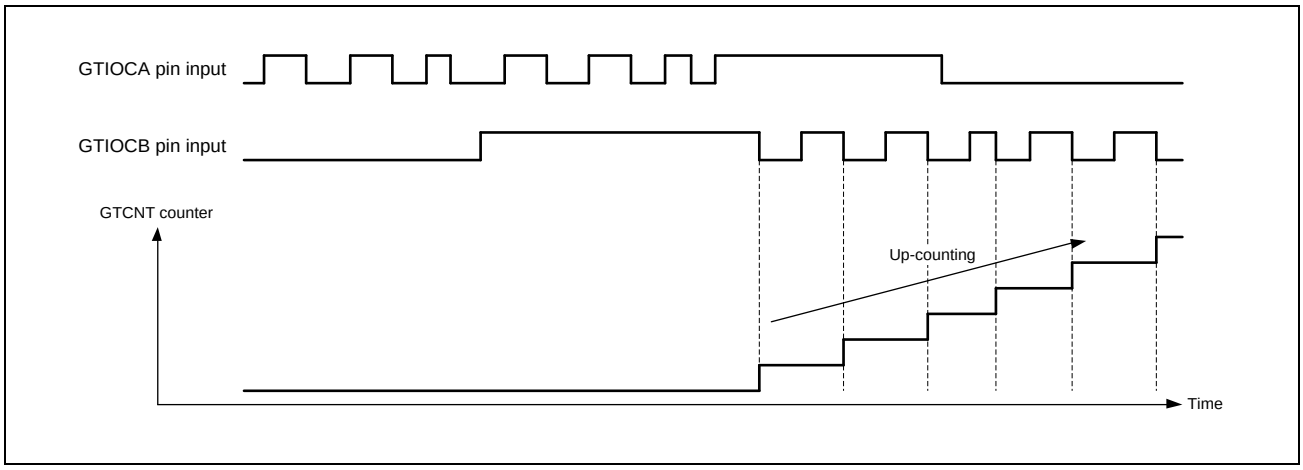


Figure 18.79 Example of phase counting mode 5 (B)

Table 18.17 Conditions of up-counting and down-counting in phase counting mode 5 (B)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Don't care	GTUPSR = H'0000_C000 GTDNSR = H'0000_0000
Low	↓	Up-counting	
↑	Low	Don't care	
↓	High		
High	↓	Up-counting	
Low	↑	Don't care	
↑	High		
↓	Low		

Remarks: ↑ : Rising edge
↓ : Falling edge

18.4 Interrupt Sources

18.4.1 Interrupt Sources and Priorities

The GPT provides the following interrupt sources:

- GTCCR input capture/compare match
- GTADTR compare match
- GTCNT counter overflow (GTPR compare match)/underflow.

Each interrupt source has its own status flag. When an interrupt source signal is generated, the associated status flag in GTST is set to 1. The associated status flag in GTST can be cleared by writing 0. If flag set and flag clear happen at the same time, flag clear takes priority over flag set. These flags are automatically updated by internal state. Relative channel priorities can be changed by the Interrupt Controller. However the priority within a channel is fixed. For details, see **Section 8, Interrupt Controller**. **Table 18.18** lists the GPT interrupt sources.

Table 18.18 Interrupt sources (1/3)

Channel	Name	Interrupt source	Interrupt flag	DMAC activation
0	CCMPA0	GPT32E0.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB0	GPT32E0.GTCCRB input capture/compare match	TCFB	Possible
	CMPC0	GPT32E0.GTCCRC compare match	TCFC	Possible
	CMPD0	GPT32E0.GTCCRD compare match	TCFD	Possible
	CMPE0	GPT32E0.GTCCRE compare match	TCFE	Possible
	CMPF0	GPT32E0.GTCCRF compare match	TCFF	Possible
	ADTRGA0	GPT32E0.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB0	GPT32E0.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF0	GPT32E0.GTCNT overflow (GPT32E0.GTPR compare match)	TCFPO	Possible
	UNF0	GPT32E0.GTCNT underflow	TCFPU	Possible
1	CCMPA1	GPT32E1.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB1	GPT32E1.GTCCRB input capture/compare match	TCFB	Possible
	CMPC1	GPT32E1.GTCCRC compare match	TCFC	Possible
	CMPD1	GPT32E1.GTCCRD compare match	TCFD	Possible
	CMPE1	GPT32E1.GTCCRE compare match	TCFE	Possible
	CMPF1	GPT32E1.GTCCRF compare match	TCFF	Possible
	ADTRGA1	GPT32E1.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB1	GPT32E1.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF1	GPT32E1.GTCNT overflow (GPT32E1.GTPR compare match)	TCFPO	Possible
	UNF1	GPT32E1.GTCNT underflow	TCFPU	Possible

Table 18.18 Interrupt sources (2/3)

Channel	Name	Interrupt source	Interrupt flag	DMAC activation
2	CCMPA2	GPT32E2.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB2	GPT32E2.GTCCRB input capture/compare match	TCFB	Possible
	CMPC2	GPT32E2.GTCCRC compare match	TCFC	Possible
	CMPD2	GPT32E2.GTCCRD compare match	TCFD	Possible
	CMPE2	GPT32E2.GTCCRE compare match	TCFE	Possible
	CMPF2	GPT32E2.GTCCRF compare match	TCFF	Possible
	ADTRGA2	GPT32E2.GTCCRE compare match	ADTRAUF ADTRADF	Possible
	ADTRGB2	GPT32E2.GTCCRF compare match	ADTRBUF ADTRBDF	Possible
	OVF2	GPT32E2.GTCNT overflow (GPT32E2.GTPR compare match)	TCFPO	Possible
	UNF2	GPT32E2.GTCNT underflow	TCFPU	Possible
3	CCMPA3	GPT32E3.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB3	GPT32E3.GTCCRB input capture/compare match	TCFB	Possible
	CMPC3	GPT32E3.GTCCRC compare match	TCFC	Possible
	CMPD3	GPT32E3.GTCCRD compare match	TCFD	Possible
	CMPE3	GPT32E3.GTCCRE compare match	TCFE	Possible
	CMPF3	GPT32E3.GTCCRF compare match	TCFF	Possible
	ADTRGA3	GPT32E3.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB3	GPT32E3.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF3	GPT32E3.GTCNT overflow (GPT32E3.GTPR compare match)	TCFPO	Possible
	UNF3	GPT32E3.GTCNT underflow	TCFPU	Possible
4	CCMPA4	GPT32E4.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB4	GPT32E4.GTCCRB input capture/compare match	TCFB	Possible
	CMPC4	GPT32E4.GTCCRC compare match	TCFC	Possible
	CMPD4	GPT32E4.GTCCRD compare match	TCFD	Possible
	CMPE4	GPT32E4.GTCCRE compare match	TCFE	Possible
	CMPF4	GPT32E4.GTCCRF compare match	TCFF	Possible
	ADTRGA4	GPT32E4.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB4	GPT32E4.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF4	GPT32E4.GTCNT overflow (GPT32E4.GTPR compare match)	TCFPO	Possible
	UNF4	GPT32E4.GTCNT underflow	TCFPU	Possible

Table 18.18 Interrupt sources (3/3)

Channel	Name	Interrupt source	Interrupt flag	DMAC activation
5	CCMPA5	GPT32E5.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB5	GPT32E5.GTCCRB input capture/compare match	TCFB	Possible
	CMPC5	GPT32E5.GTCCRC compare match	TCFC	Possible
	CMPD5	GPT32E5.GTCCRD compare match	TCFD	Possible
	CMPE5	GPT32E5.GTCCRE compare match	TCFE	Possible
	CMPF5	GPT32E5.GTCCRF compare match	TCFF	Possible
	ADTRGA5	GPT32E5.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB5	GPT32E5.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF5	GPT32E5.GTCNT overflow (GPT32E5.GTPR compare match)	TCFPO	Possible
	UNF5	GPT32E5.GTCNT underflow	TCFPU	Possible
6	CCMPA6	GPT32E6.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB6	GPT32E6.GTCCRB input capture/compare match	TCFB	Possible
	CMPC6	GPT32E6.GTCCRC compare match	TCFC	Possible
	CMPD6	GPT32E6.GTCCRD compare match	TCFD	Possible
	CMPE6	GPT32E6.GTCCRE compare match	TCFE	Possible
	CMPF6	GPT32E6.GTCCRF compare match	TCFF	Possible
	ADTRGA6	GPT32E6.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB6	GPT32E6.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF6	GPT32E6.GTCNT overflow (GPT32E6.GTPR compare match)	TCFPO	Possible
	UNF6	GPT32E6.GTCNT underflow	TCFPU	Possible
7	CCMPA7	GPT32E7.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB7	GPT32E7.GTCCRB input capture/compare match	TCFB	Possible
	CMPC7	GPT32E7.GTCCRC compare match	TCFC	Possible
	CMPD7	GPT32E7.GTCCRD compare match	TCFD	Possible
	CMPE7	GPT32E7.GTCCRE compare match	TCFE	Possible
	CMPF7	GPT32E7.GTCCRF compare match	TCFF	Possible
	ADTRGA7	GPT32E7.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB7	GPT32E7.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF7	GPT32E7.GTCNT overflow (GPT32E7.GTPR compare match)	TCFPO	Possible
	UNF7	GPT32E7.GTCNT underflow	TCFPU	Possible

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

(1) ADTRGAn interrupt (n = 0 to 7)

When the GTCNT counter value matches with the GTADTRA register, an interrupt request is generated under the following condition:

- In up-counting, the interrupt enable bit (ADTRAUEN) in the GTINTAD register is 1
- In down-counting, the interrupt enable bit (ADTRADEN) in the GTINTAD register is 1.

In event count operation performing, this interrupt request is not generated.

(2) ADTRGBn interrupt (n = 0 to 7)

When the GTCNT counter value matches with the GTADTRB register, an interrupt request is generated under the following condition:

- In up-counting, the interrupt enable bit (ADTRBUEN) in the GTINTAD register is 1
- In down-counting, the interrupt enable bit (ADTRBDEN) in the GTINTAD register is 1.

In event count operation performing, this interrupt request is not generated.

(3) CCMPAn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRA register functions as a compare match register, the GTCNT counter value matches with the GTCCRA register.
- When the GTCCRA register functions as an input capture register, the GTCNT counter value is transferred to the GTCCRA register by an input capture signal.

(4) CCMPBn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRB register functions as a compare match register, the GTCNT counter value matches with the GTCCRB register.
- When the GTCCRB register functions as an input capture register, the GTCNT counter value is transferred to the GTCCRB register by an input capture signal.

(5) CMPCn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRC register functions as a compare match register, the GTCNT counter value matches with the GTCCRC register.

A compare match is not performed and an interrupt is not requested under the following conditions:

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 01b, 10b, 11b (buffer operation with the GTCCRC register).

(6) CMPDn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRD register functions as a compare match register, the GTCNT counter value matches with the GTCCRD register.

A compare match is not performed and therefore interrupt is not requested under the following conditions:

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 10b, 11b (buffer operation with the GTCCRD register).

(7) CMPEn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition.

- When the GTCCRE register functions as a compare match register, the GTCNT counter value matches with the GTCCRE register.

A compare match is not performed and an interrupt is not requested under the following conditions.

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 01b, 10b, 11b (buffer operation with the GTCCRE register)

(8) CMPFn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRF register functions as a compare match register, the GTCNT counter value matches with the GTCCRF register.

A compare match is not performed and therefore interrupt is not requested under the following conditions:

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 10b, 11b (buffer operation with the GTCCRF register).

(9) OVFn interrupt (n = 0 to 7)

An interrupt request is generated under the following conditions:

- In saw-wave mode, interrupt requests are enabled at overflows (when the GTCNT counter value changes from GTPR to 0 during up-counting)
- In triangle-wave mode, interrupt requests are enabled at crests (GTCNT changes from GTPR to GTPR-1)
- In counting by hardware sources, overflow (GTCNT changes from GTPR to 0 in up count) has occurred.

(10) UNFn interrupt (n = 0 to 7)

An interrupt request is generated under the following conditions:

- In saw-wave mode, interrupt requests are enabled at underflows (when the GTCNT counter value changes from 0 to GTPR during down-counting).
- In triangle-wave mode, interrupt requests are enabled at troughs (GTCNT changes from 0 to 1).
- In counting by hardware sources, underflow (GTCNT changes from 0 to GTPR in down count) has occurred.

Table 18.19 Interrupt signals, interrupt permission bits, and interrupt status flags

Interrupt signal	Interrupt permission bit	Interrupt status flag
UNFn	GTINTAD[7:6] (GTINTPR[1:0])	GTST[7] (TCFPU)
OVFn		GTST[6] (TCFPO)
ADTRGBn	GTINTAD[19] (ADTRBDEN) GTINTAD[18] (ADTRBUEN)	GTST[19] (ADTRBDF) GTST[18] (ADTRBUF)
ADTRGAn	GTINTAD[17] (ADTRADEN) GTINTAD[16] (ADTRAUEN)	GTST[17] (ADTRADF) GTST[16] (ADTRAUF)
CMPFn	GTINTAD[5] (GTINTF)	GTST[5] (TCFF)
CMPEn	GTINTAD[4] (GTINTE)	GTST[4] (TCFE)
CMPDn	GTINTAD[3] (GTINTD)	GTST[3] (TCFD)
CMPCn	GTINTAD[2] (GTINTC)	GTST[2] (TCFC)
CCMPBn	GTINTAD[1] (GTINTB)	GTST[1] (TCFB)
CCMPAn	GTINTAD[0] (GTINTA)	GTST[0] (TCFA)

18.4.2 DMAC Activation

The DMAC can be activated by the interrupt in each channel. For details, see **Section 14, Direct Memory Access Controller**.

18.4.3 Interrupt and A/D Conversion Request Skipping Function

By setting the GTITC register, the GTCNT counter overflow (GTPR compare match) interrupt (OVFn) and underflow interrupt (UNFn) can be skipped. Other interrupts and A/D converter start request signals can be skipped in coordination with the OVFn/UNFn skipping function.

The interrupt request skipping function only depends on the setting of GTITC register and is independent of the setting of interrupt permission bits in the GTINTAD register.

When both troughs and crests are counted and skipped in triangle-wave mode, if the number of times of skipping is odd, OVFn/UNFn interrupt requests cannot be generated at troughs only or at crests only depending on the skipping counter start timing. To count both troughs and crests and generate the OVFn/UNFn interrupts at troughs only or crests only in triangle-wave mode, you must set an even number of skips.

Similarly, in saw-wave mode, when both overflows and underflows are counted and skipped with the count direction changed, OVFn interrupt requests cannot be generated on either overflows or underflows only. To count both overflows and underflows with the count direction changed and generate the OVFn/UNFn interrupts on either overflows or underflows only in saw wave mode, first check the skipping state carefully.

Before changing the skipping count, you must release the skipping count setting (GTITC.IVTC[1:0] bits = 00b).

Figure 18.80 to **Figure 18.85** show examples of skipping function operation.

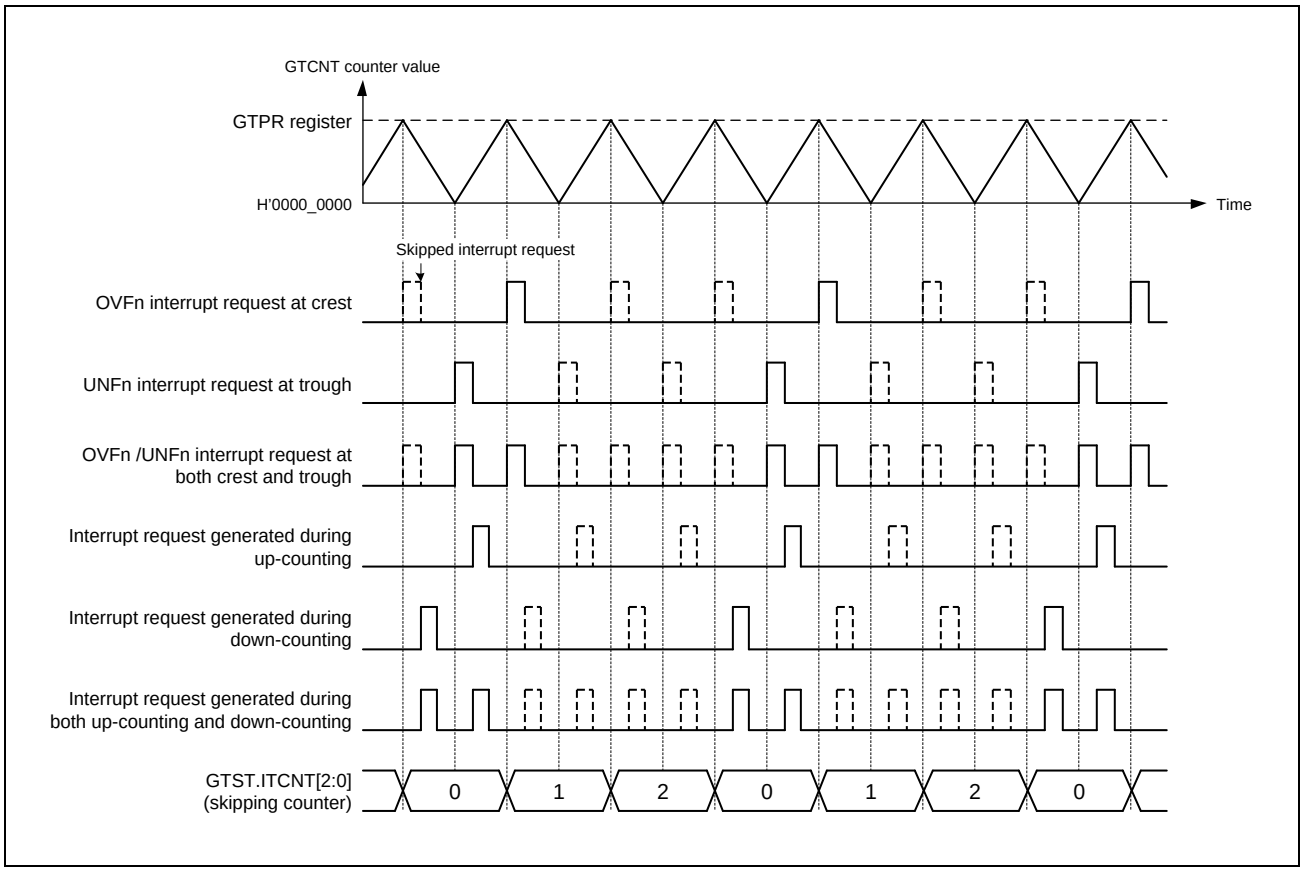


Figure 18.80 Example of interrupt skipping function operation with triangle waves, counting and skipping crests, and skipping count = 2

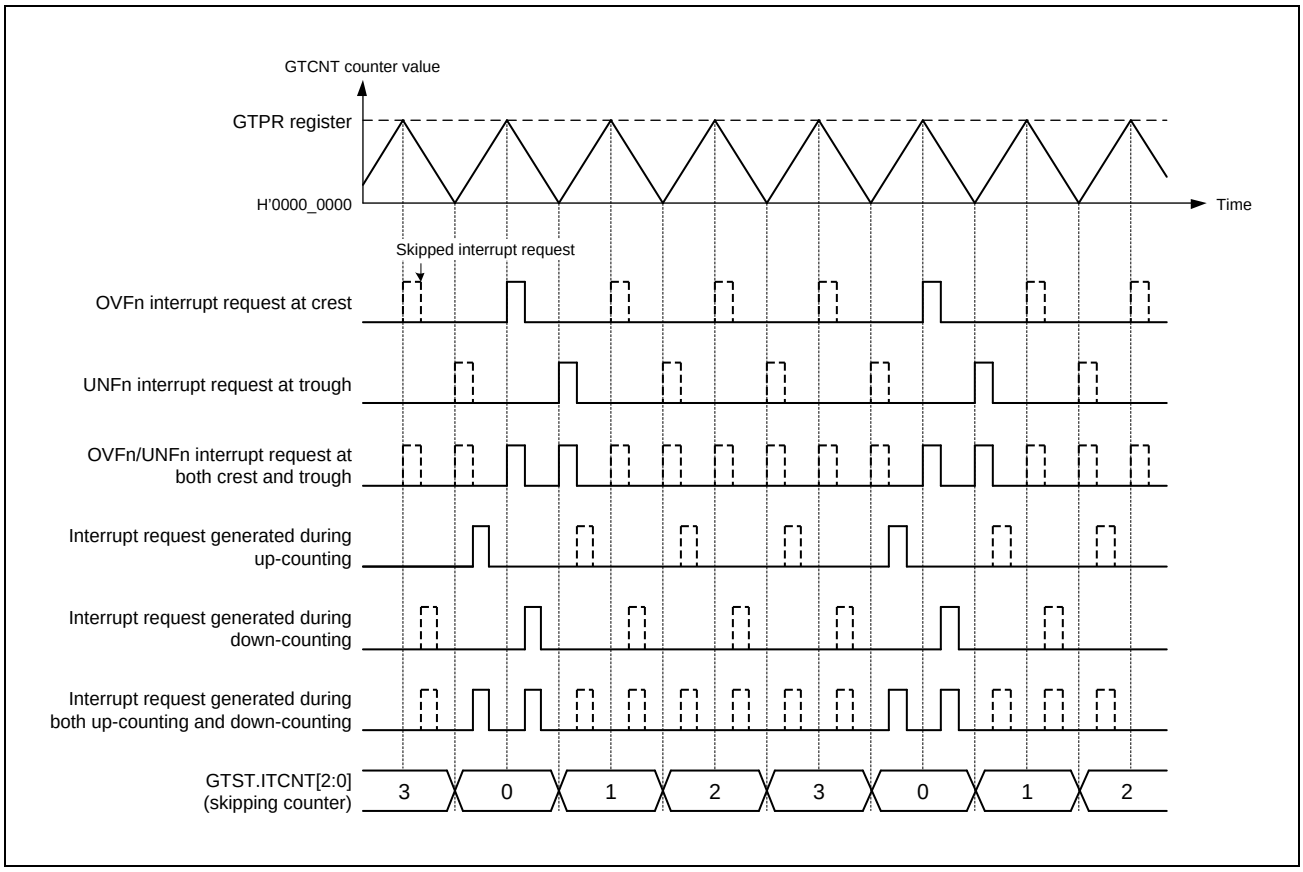


Figure 18.81 Example of interrupt skipping function operation with triangle waves, counting and skipping troughs, and skipping count = 3

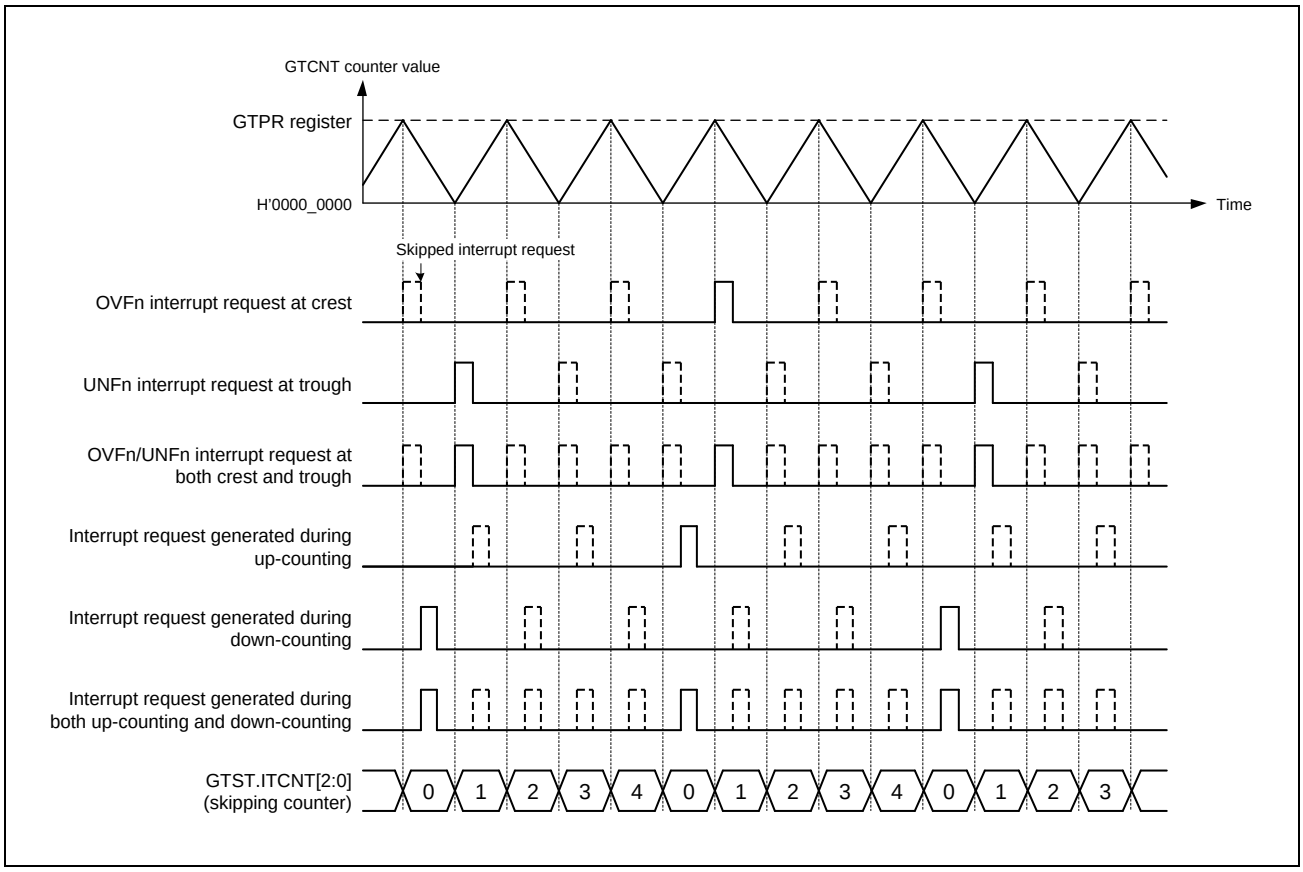


Figure 18.82 Example of interrupt skipping function operation with triangle waves, counting and skipping both troughs and crests, and skipping count = 4

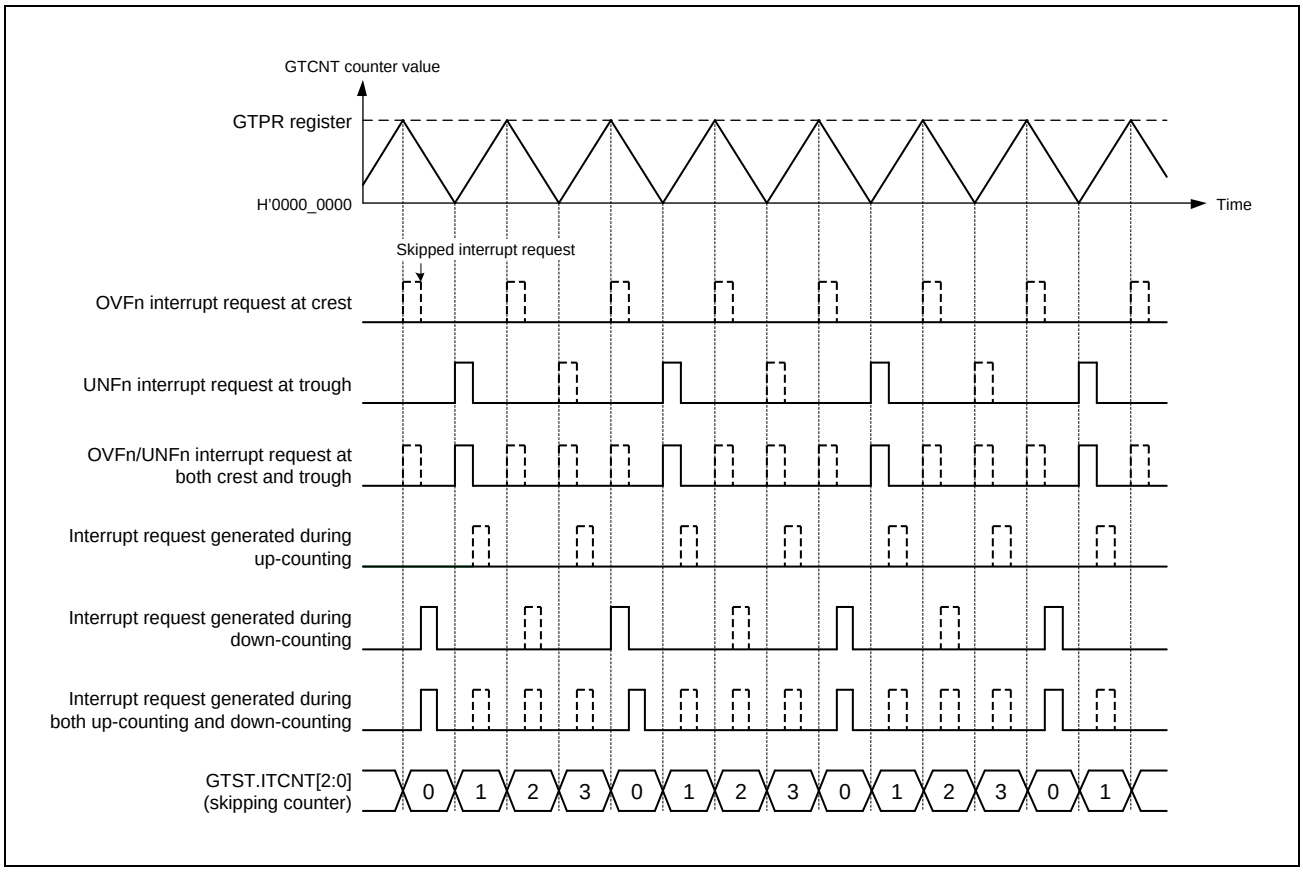


Figure 18.83 Example of interrupt skipping function operation with triangle waves, counting and skipping both troughs and crests, skipping count = 3, and skipping started at up-counting

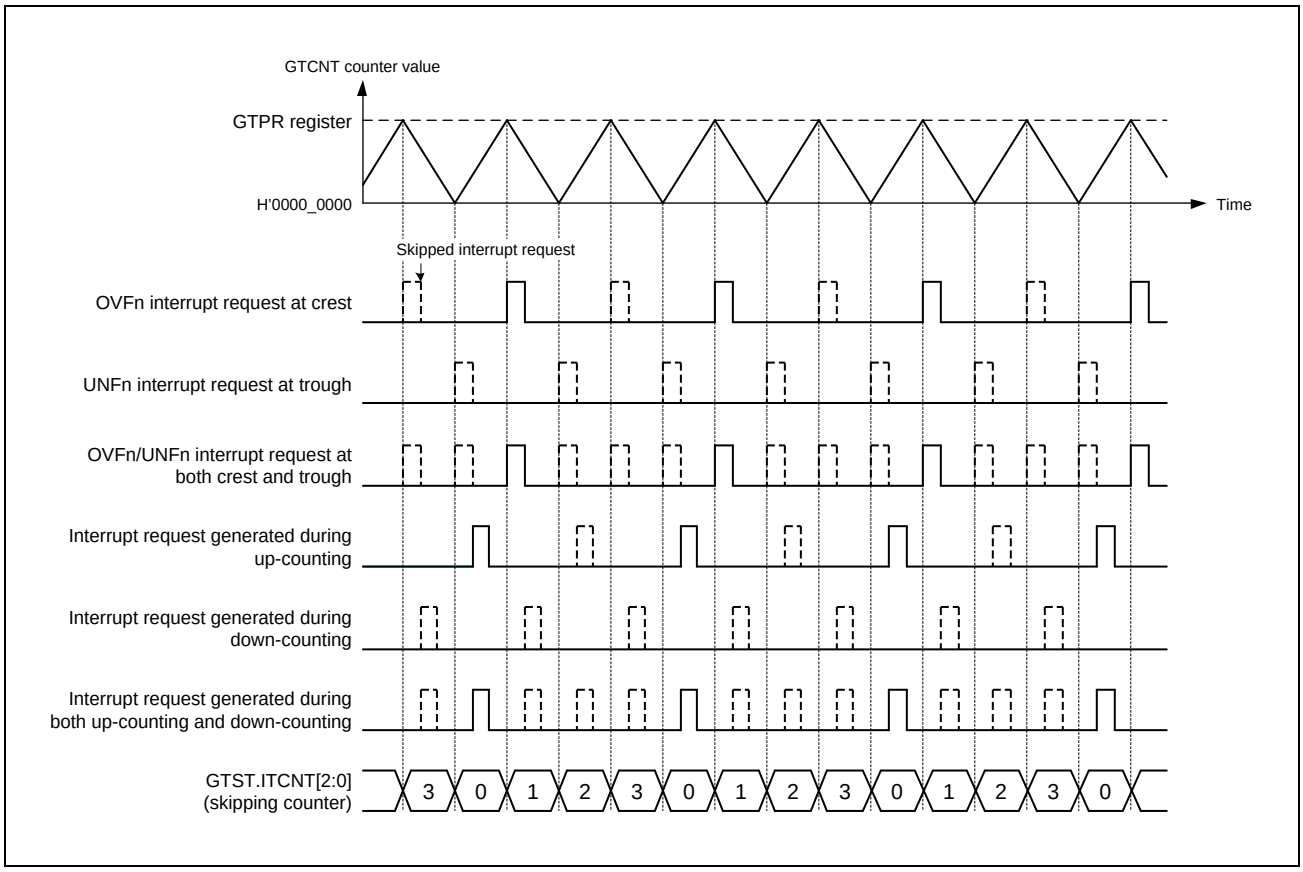


Figure 18.84 Example of interrupt skipping function operation with triangle waves, counting and skipping both troughs and crests, skipping count = 3, and skipping started at down-counting

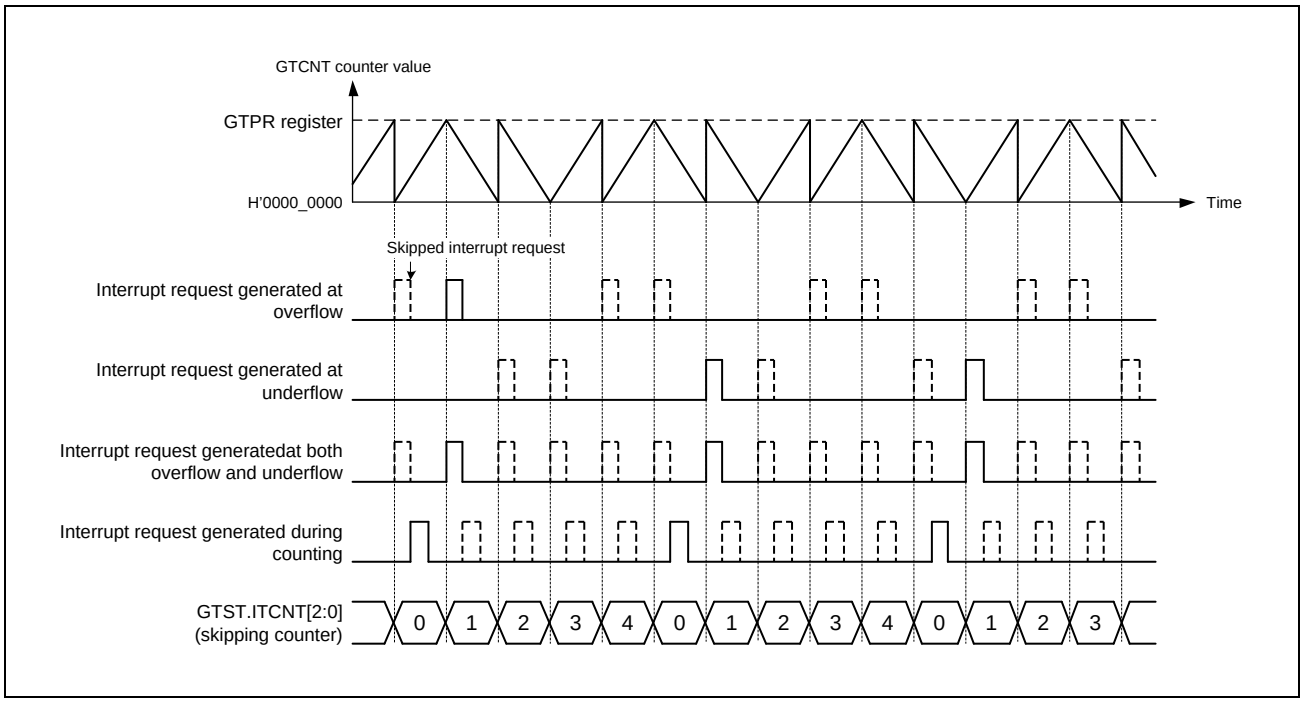


Figure 18.85 Example of interrupt skipping function operation with saw waves, operation with count direction changed, counting and skipping both overflows and underflows, and skipping count = 4

18.5 A/D Converter Start Request

An A/D converter start request can be issued at a compare match between the GTCNT counter and GTADTRA or GTADTRB, and up-counting only, down-counting only, or both up-counting and down-counting can be specified. In event count operation performing, A/D converter start requests interrupt cannot be generated.

GTADTRA and GTADTRB each has two buffer registers. Buffer operation with GTADTRA combined with GTADTBRA and GTADTDBRA, and buffer operation with GTADTRB combined with GTADTBRB and GTADTDBRB can be performed.

Figure 18.86 shows an example of A/D converter start request operation, and **Figure 18.87** shows an example setting for A/D converter start request operation.

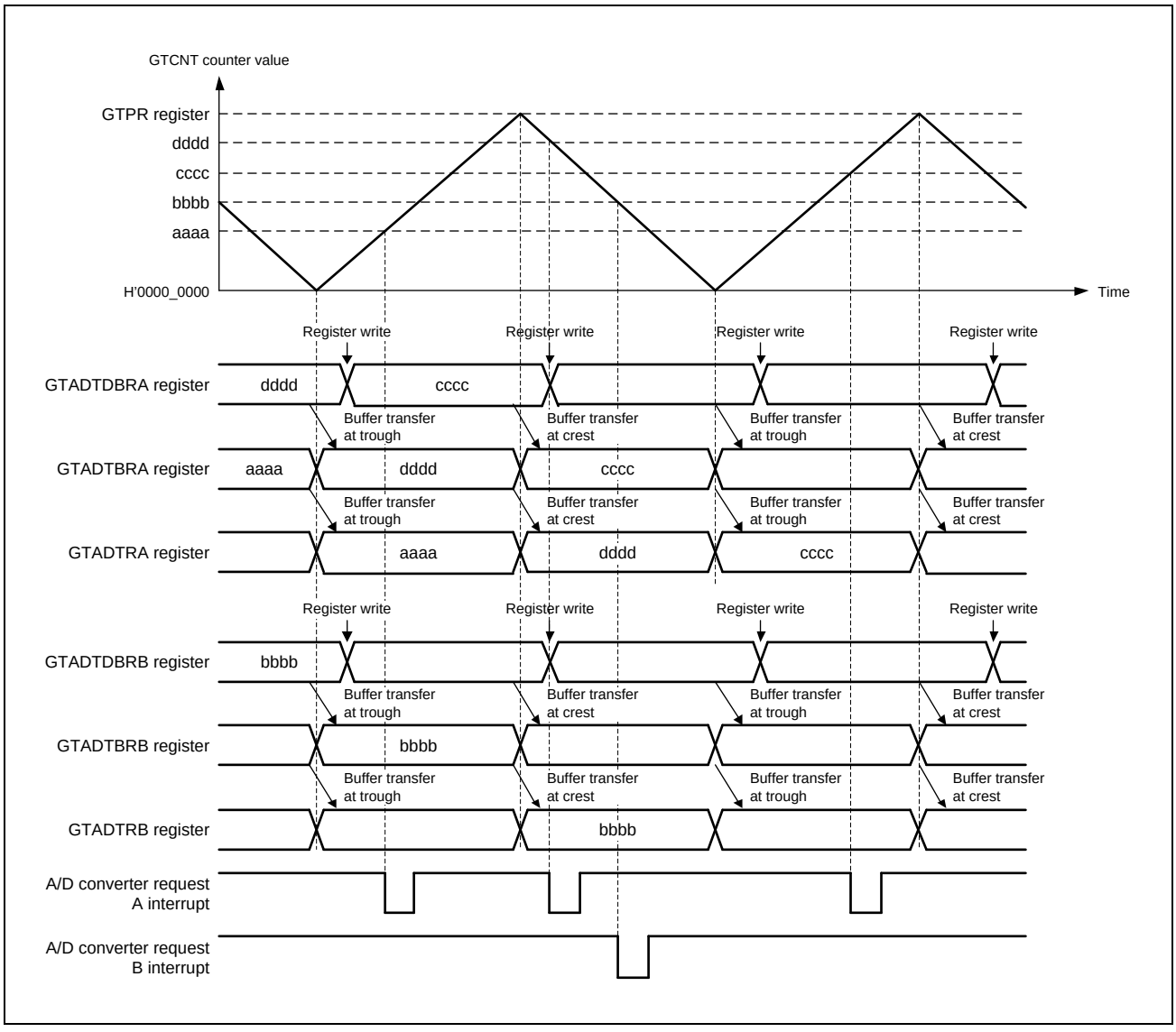


Figure 18.86 Example of A/D converter start request timing operation with triangle waves, double buffer operation, buffer transfer at both troughs and crests, A/D converter start request interrupt by GTADTRA at both up-counting and down-counting, and A/D converter start request interrupt by GTADTRB at down-counting

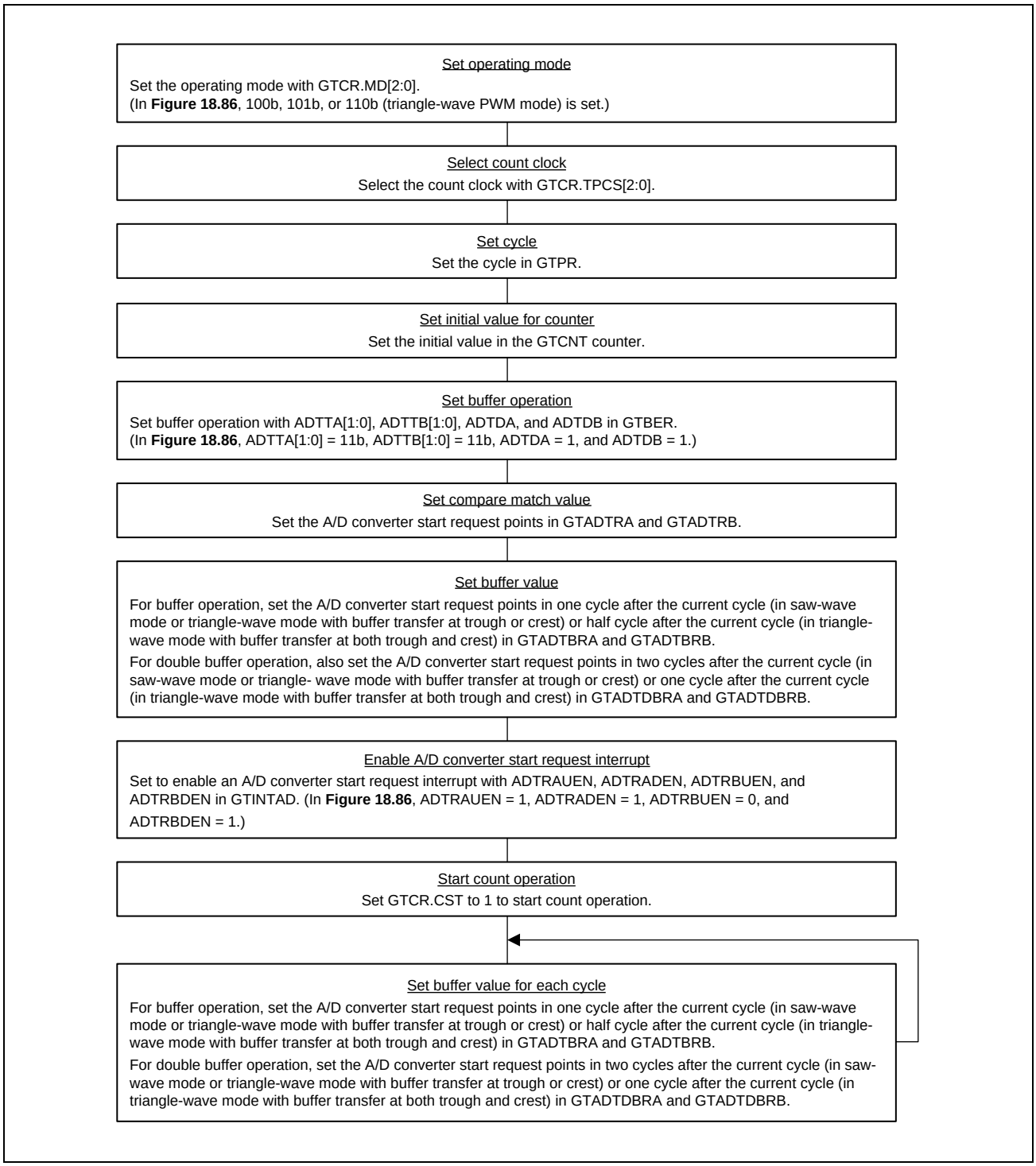


Figure 18.87 Example setting for A/D converter start request timing operation

18.6 Noise Filter Function

Each pin for use in input capture to the GPT is equipped with a noise filter. The noise filter samples input signals at the sampling clock and removes the pulses whose length is less than three sampling cycles.

The noise filter functionality includes enabling and disabling the noise filter for each pin and setting of the sampling clock for each channel.

Figure 18.88 shows the timing of noise filtering.

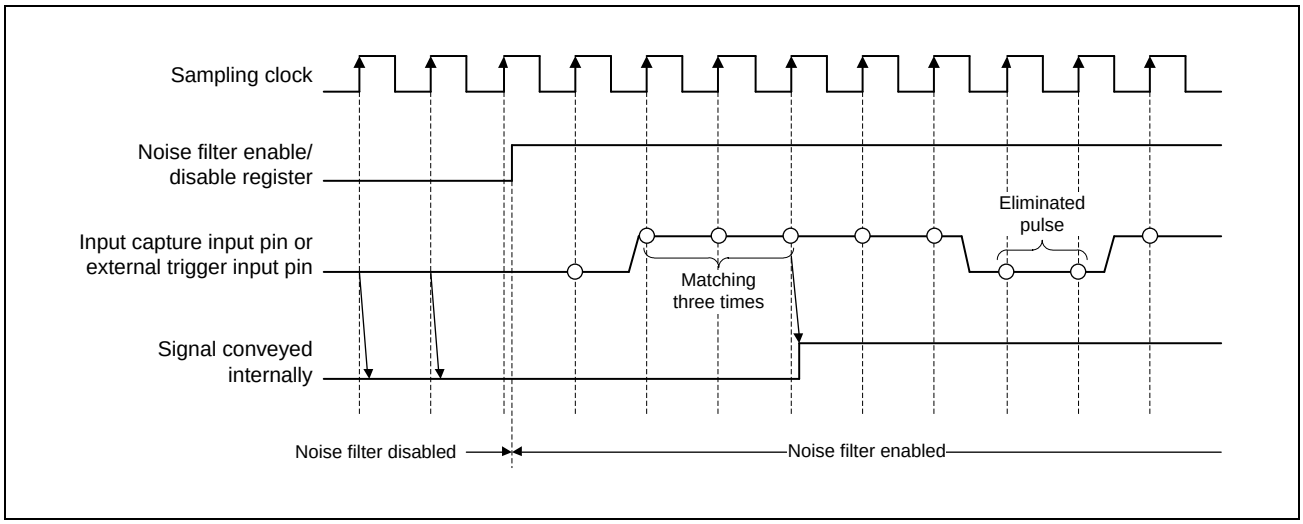


Figure 18.88 Timing of noise filtering

If noise filtering is enabled, the input capture operation or external trigger operation performs on the edges of the noise filtered signal after a delay of a sampling interval × 3 + Pϕ. This is caused by the noise filtering for the input capture input or external trigger operation.

18.7 Protection Function

18.7.1 Write-Protection for Registers

To prevent registers from being accidentally modified, registers can be write-protected in channel units by setting GTWP.WP. Write-protection can be set for the following registers:

GTSSR, GTPSR, GTCSR, GTUPSR, GTDNSR, GTICASR, GTICBSR, GTCR, GTUDDTYC, GTIOR, GTINTAD,GTST, GTBER, GTITC, GTCNT, GTCCRA, GTCCRB, GTCCRC, GTCCRD, GTCCRE, GTCCRF, GTPR, GTPBR, GTPDBR, GTADTRA, GTADTBRA, GTADTDBRA, GTADTRB, GTADTBRB, GTADTDBRB, GTDTCR, GTDVU, GTDVD, GTDBU, GTDBD, GTSOS, GTSOTR.

18.7.2 Disabling of Buffer Operation

If the timing of buffer register write is delayed in relative to the timing for the buffer transfer, buffer operation can be suspended with the GTBER.BD setting. Buffer transfer can be temporarily disabled even when a buffer transfer condition is generated during a buffer register write. This can be done by setting the associated GTBER.BD bit to 1 (buffer operation disabled) before a buffer register write and clearing the bit to 0 (buffer operation enabled) after completion of writing to all buffer registers. **Figure 18.89** shows an example of operation for disabling buffer operation.

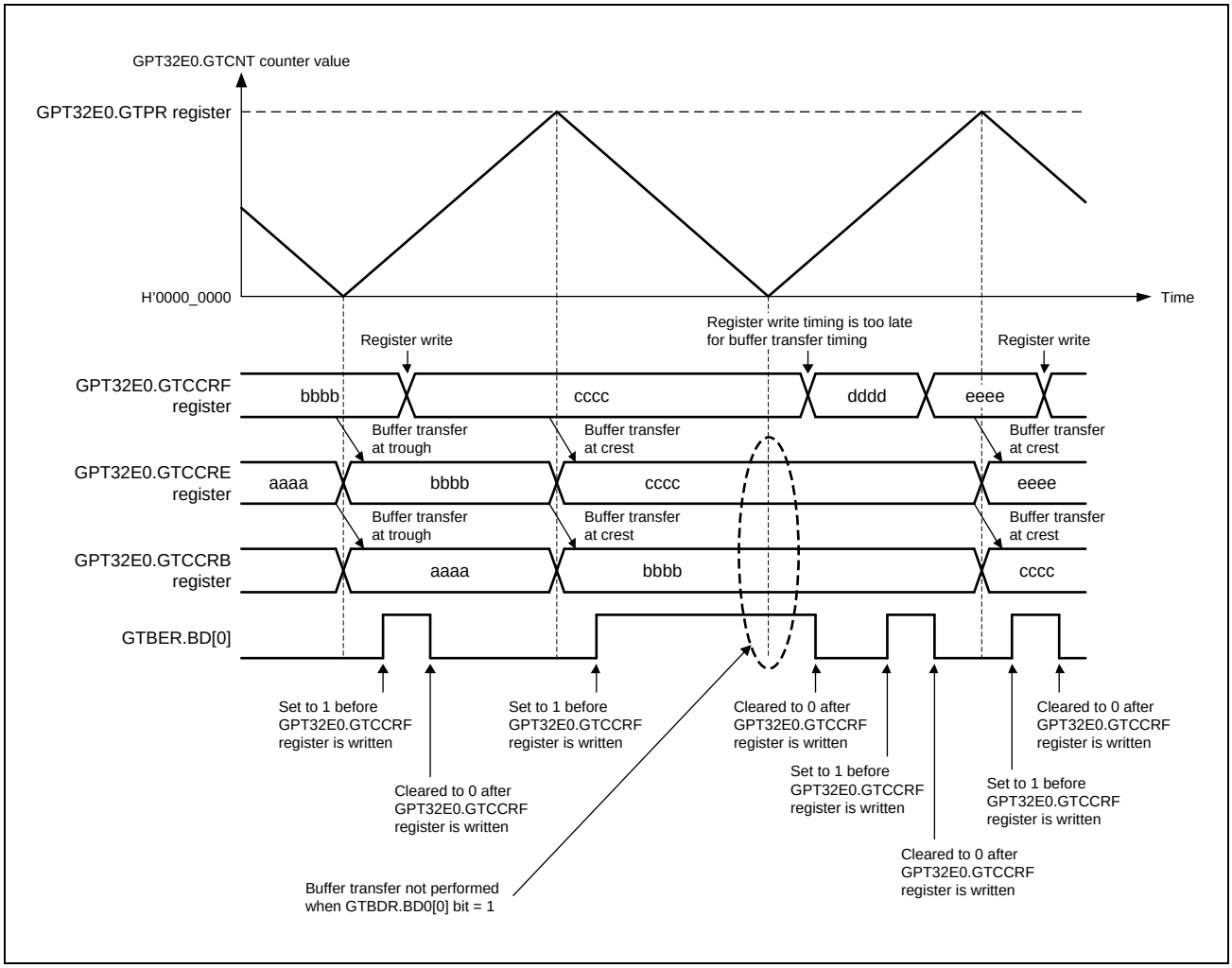


Figure 18.89 Example of operation for disabling buffer operation with triangle waves, double buffer operation, and buffer transfer at both troughs and crests

18.7.3 GTIOC Pin Output Negate Control

For protection from system failure, the output disable control that changes the GTIOC pin output value forcibly is provided for GTIOC pin output by the request of output disable from POEG.

When dead time error occurs or the GTIOCA pin output value is the same as the GTIOCB pin output value, output protection is required. GPT detects this condition and generates output disable requests to POEG based on the settings in the output disable request permission bits, such as GTINTAD.GRPDTE, GTINTAD.GRPABH, GTINTAD.GRPABL. After the POEG receives output disable requests from each channel and calculates external input using an OR operation, the POEG generates output disable requests to GPT.

One output disable signal (representing the shared output disable request signal of the GTIOCA pin and the GTIOCB pin) out of four* output disable requests generated by the POEG is selected by setting GTINTAD.GRP[1:0]. The status of the selected disabled output request is monitored by reading the GTST.ODF bit. The output level during output disable is based on the GTIOR.OADF[1:0] setting for the GTIOCA pin and the GTIOR.OBDF[1:0] setting for the GTIOCB pin. The change to the output disable state is performed asynchronously by generating the output disable request from the POEG. The release of the output disable state is performed at end of cycle by terminating the output disable request. The timing of release of the output disable state is a minimum of 3 P0ϕ cycles after terminating the output disable request. To perform output disable control reliably, allow at least 4 P0ϕ cycles after generating the output disable request (by clearing the output disable request flag in POEG) until the output disable request is terminated.

When event count is performed or when the output disable state is to be released immediately without waiting for an end of cycle, GTIOR.OADF[1:0] must be set to 00b (for GTIOCA pin) or GTIOR.OBDF[1:0] must be set to 00b (for GTIOCB pin).

Figure 18.90 shows an example of the GTIOC pin output disable control operation.

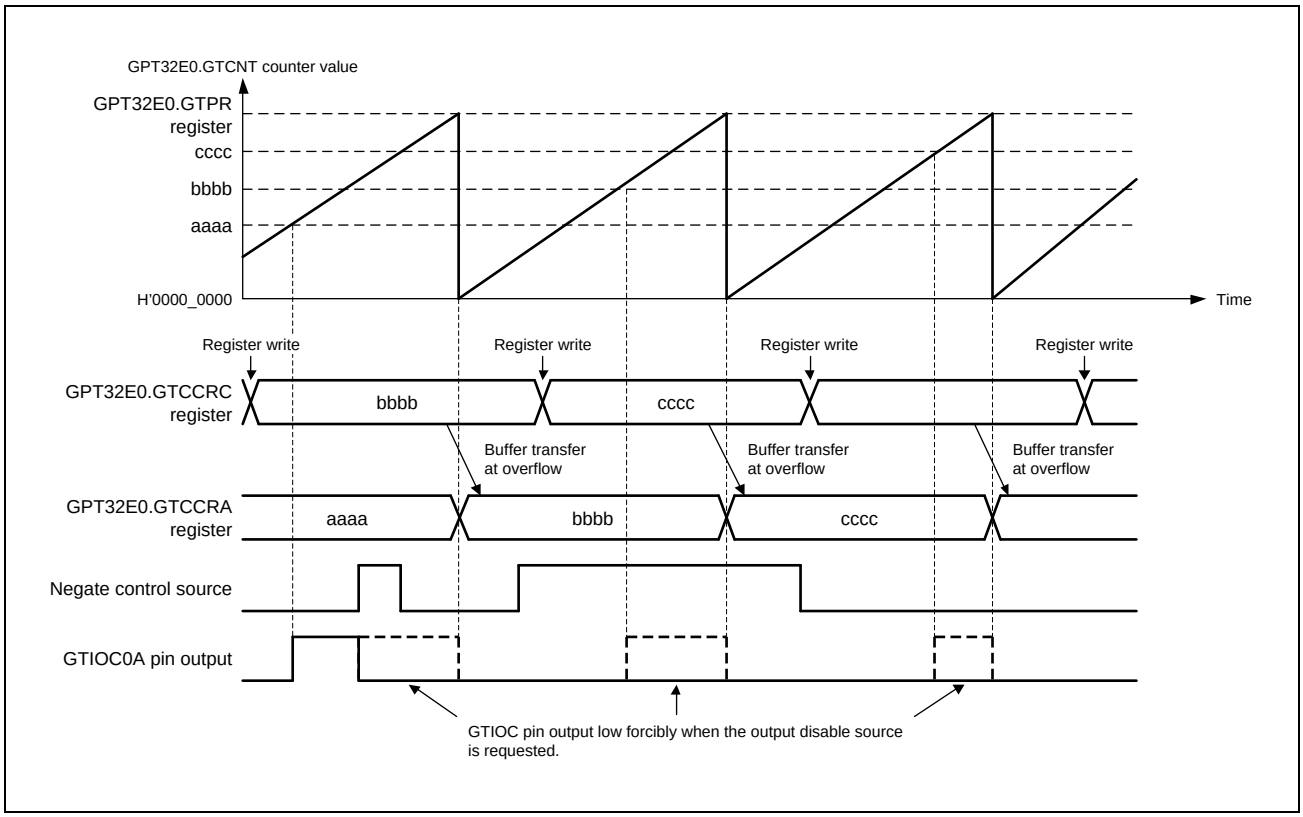


Figure 18.90 Example of GTIOC pin output disable control operation with saw-wave up-counting, buffer operation, active level 1, high output at GTCCRA compare match, low output at cycle end, and low output at output disable

18.7.4 Output Protection Function for GTIOC Pin Output

In preparation for incorrect settings of the GTCCRA register (settings outside the range of 0 < GTCCRA < GTPR), the output protection function for the GTIOC pin output (disabling function) is activated when the automatic dead time setting (GTDTCR.TDE = 1) is made in triangle-wave mode. The status of the output protection function can be read from GTSOS.SOS[1:0].

Figure 18.91 shows the output protection function state transition.

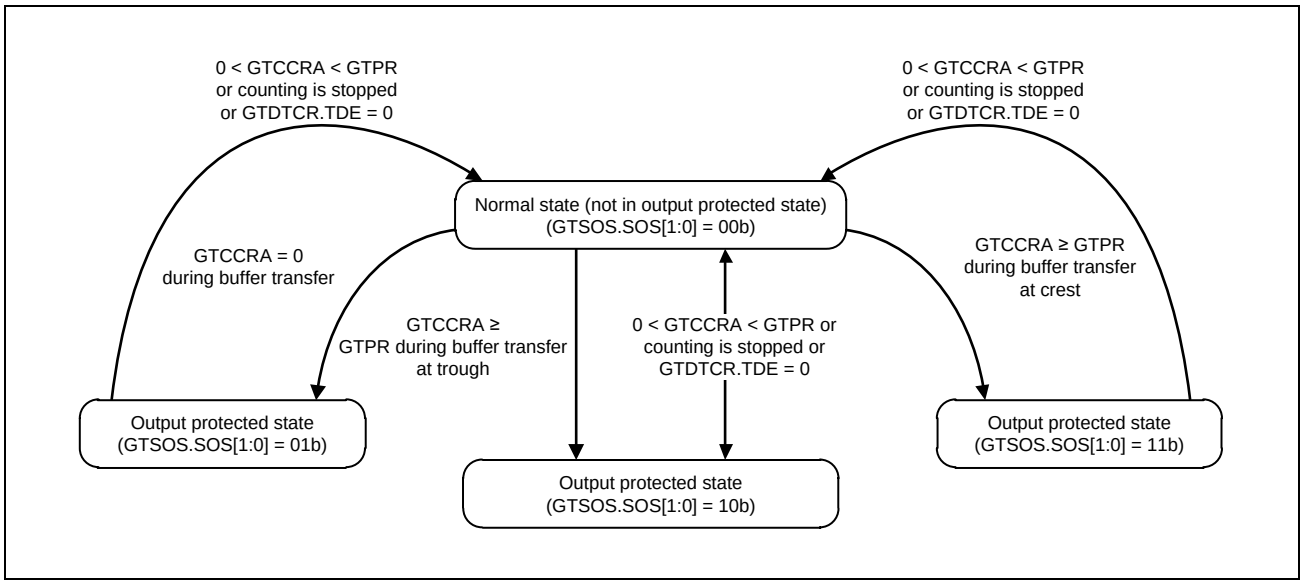
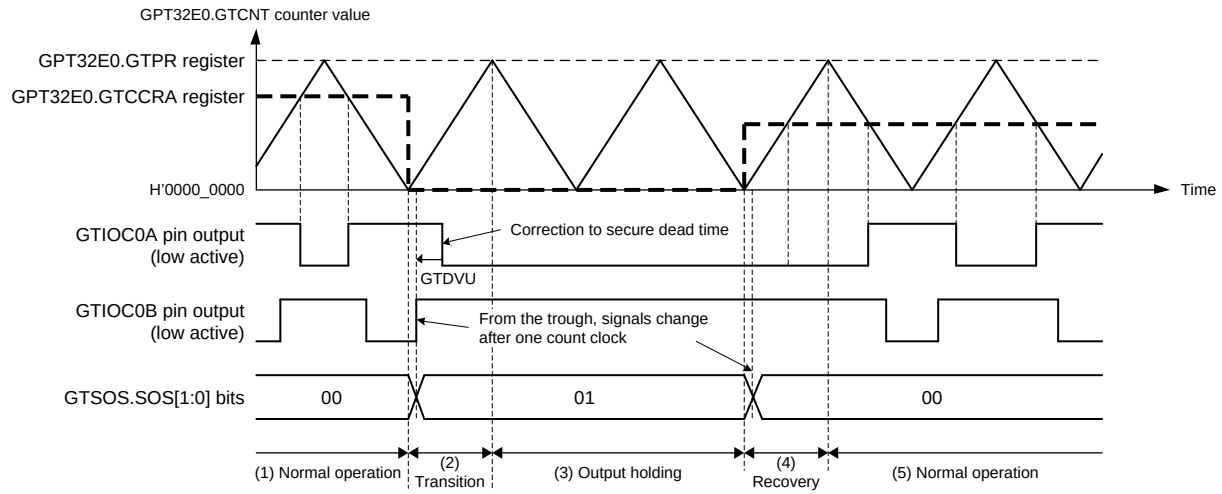


Figure 18.91 Output protection function

18.7.4.1 Output Protection Function When the GTCCRA Register is Set to 0 during Buffer Transfer

Figure 18.92 and **Figure 18.93** show examples of output protection function operation when the GTCCRA register is set to 0 during buffer transfer at troughs, and **Figure 18.94** and **Figure 18.95** show examples when the GTCCRA register is set to 0 during buffer transfer at crests.



(1) Normal operation period: Toggle output due to a compare match is performed normally.

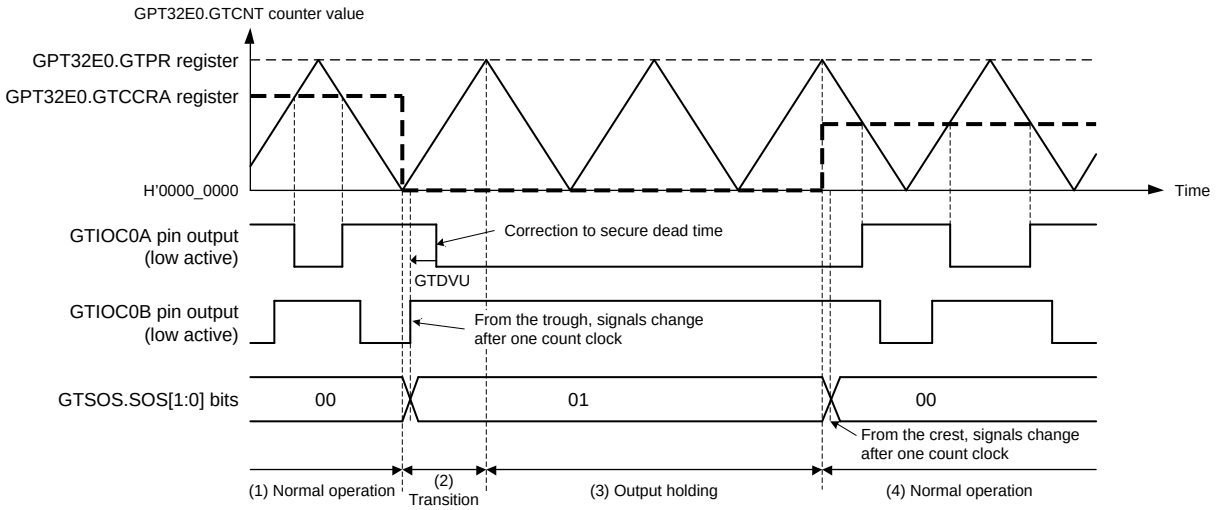
(2) Transition period: When GTCCRA register = 0 is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 01b, and toggle output is continued up to the next crest.

(3) Output holding period: The output is held while GTCCRA register = 0.

(4) Recovery period: When 0 < GTCCRA register < GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and holding output is maintained up to the next crest.

(5) Normal operation period: Toggle output due to a compare match isperformed normally.

Figure 18.92 Example of output protection operation when GTCCRA is set to 0 during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low



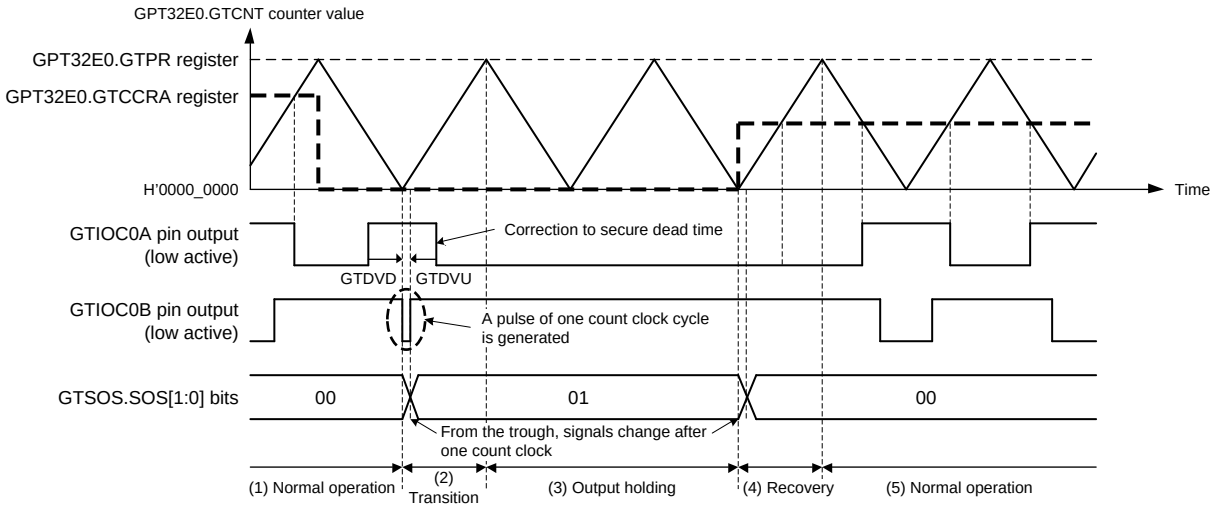
(1) Normal operation period: Toggle output due to a compare match is performed normally.

(2) Transition period: When GTCCRA register = 0 is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 01b, and toggle output is continued up to the next crest.

(3) Output holding period: The output is held while GTCCRA register = 0.

(4) Normal operation period: When 0 < GTCCRA register < GTPR register is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and toggle output due to a compare match is performed normally.

Figure 18.93 Example of output protection operation when GTCCRA is set to 0 during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at crests, and active-low



(1) Normal operation period: Toggle output due to a compare match is performed normally.

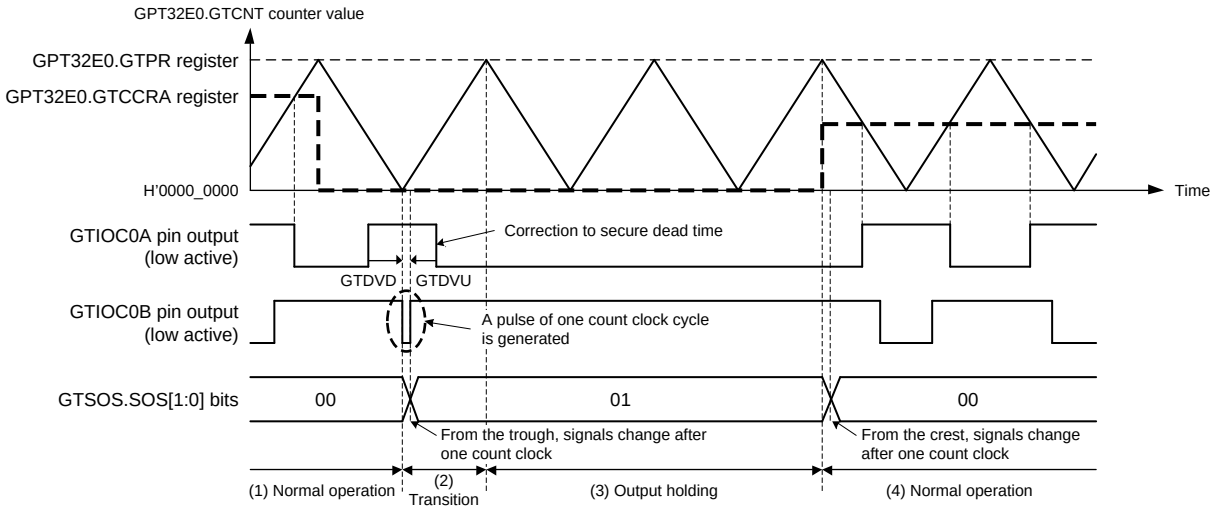
(2) Transition period: When GTCCRA register = 0 is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 01b, and toggle output is continued up to the next crest.

(3) Output holding period: The output is held while GTCCRA register = 0.

(4) Recovery period: When 0 < GTCCRA register < GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and holding output is maintained up to the next crest.

(5) Normal operation period: Toggle output due to a compare match is performed normally.

Figure 18.94 Example of output protection operation when GTCCRA is set to 0 during buffer transfer at crests, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low



(1) Normal operation period: Toggle output due to a compare match is performed normally.

(2) Transition period: When GTCCRA register = 0 is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 01b, and toggle output is continued up to the next crest.

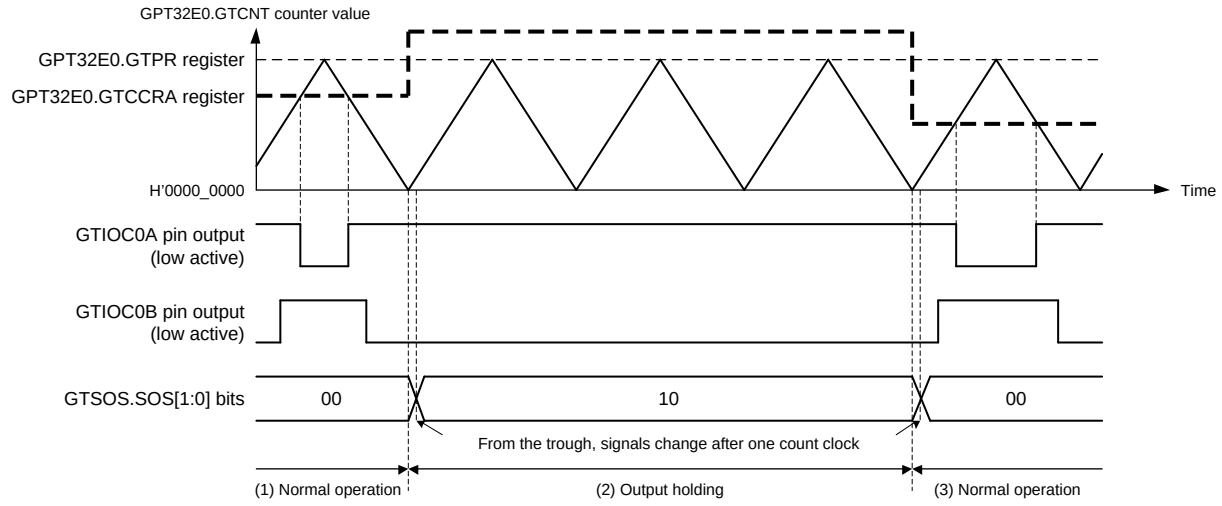
(3) Output holding period: The output is held while GTCCRA register = 0.

(4) Normal operation period: When 0 < GTCCRA register < GTPR register is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and toggle output due to a compare match is performed normally.

Figure 18.95 Example of output protection operation when GTCCRA is set to 0 during buffer transfer at crests, with 0 < GTCCRA < GTPR restored during buffer transfer at crests, and active-low

18.7.4.2 Output Protection Function When GTCCRA ≥ GTPR is Set during Buffer Transfer at Troughs

Figure 18.96 and **Figure 18.97** show examples of output protection function operation when GTCCRA ≥ GTPR is set during buffer transfer at troughs.

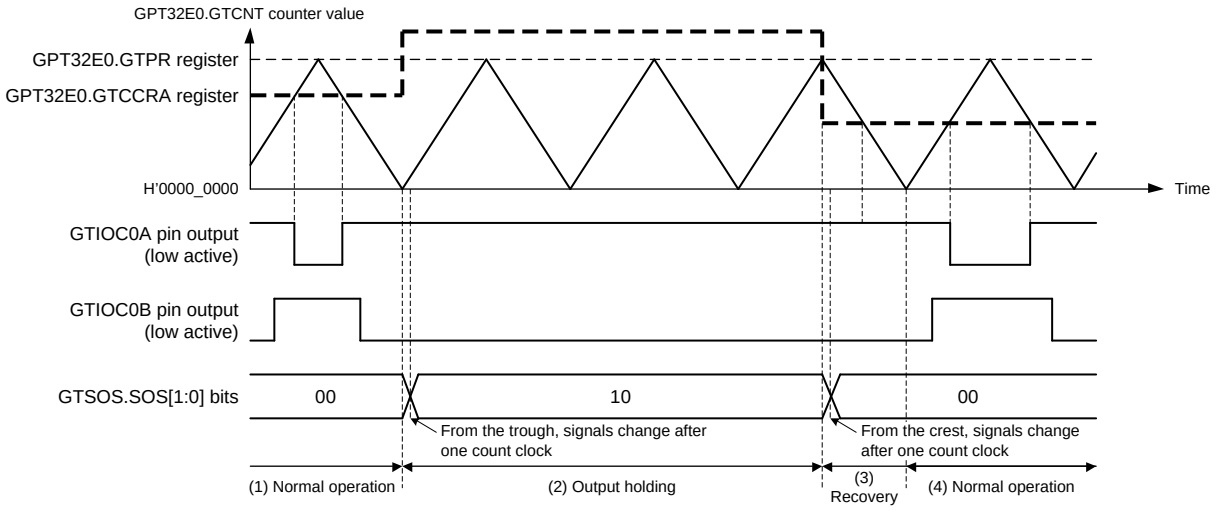


(1) Normal operation period: Toggle output due to a compare match is performed normally.

(2) Output holding: When GTCCRA register ≥ GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 10b, and the output is held while GTCCRA register ≥ GTPR register.

(3) Normal operation period: When 0 < GTCCRA register < GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and toggle output due to a compare match is performed normally.

Figure 18.96 Example of output protection operation when GTCCRA ≥ GTPR is set during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low



(1) Normal operation period: Toggle output due to a compare match is performed normally.

(2) Output holding period: When GTCCRA register ≥ GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 10b, and the output is held while GTCCRA register ≥ GTPR register.

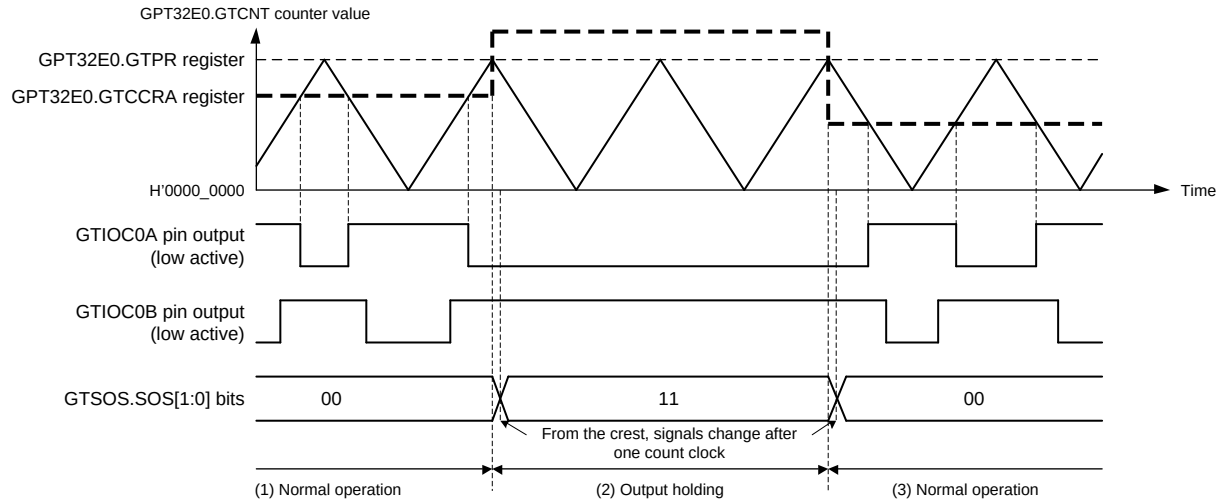
(3) Recovery period: When 0 < GTCCRA register < GTPR register is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and holding output is maintained up to the next trough.

(4) Normal operation period: Toggle output due to a compare match is performed normally.

Figure 18.97 Example of output protection operation when GTCCRA ≥ GTPR is set during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at crests, and active-low

18.7.4.3 Output Protection Function When GTCCRA ≥ GTPR is Set during Buffer Transfer at Crests

Figure 18.98 and **Figure 18.99** show examples of output protection function operation when GTCCRA ≥ GTPR is set during buffer transfer at crests.

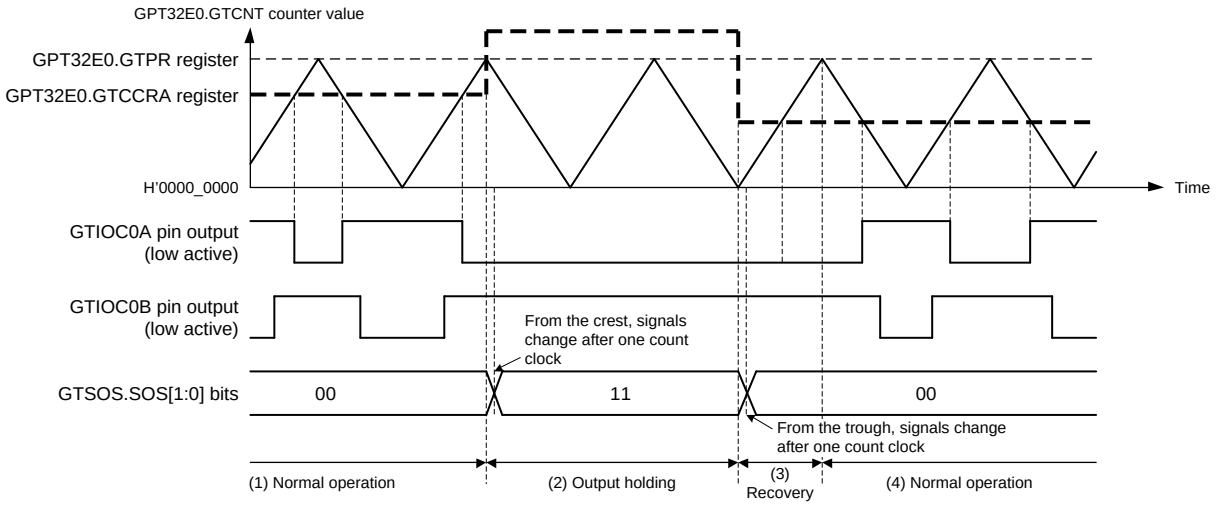


(1) Normal operation period: Toggle output due to a compare match is performed normally.

(2) Output holding period:: When GTCCRA register ≥ GTPR register is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] bits = 11b, and the output is held while GTCCRA register ≥ GTPR register.

(3) Normal operation period: When 0 < GTCCRA register < GTPR register is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and toggle output due to a compare match is performed normally.

Figure 18.98 Example of output protection operation when GTCCRA ≥ GTPR is set during buffer transfer at crests, with 0 < GTCCRA < GTPR restored during buffer transfer at crests, and active- low



(1) Normal operation period: Toggle output due to a compare match is performed normally.

(2) Output holding period: When GTCCRA ≥ GTPR is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] = 11b, and the output is held while GTCCRA ≥ GTPR.

(3) Recovery period: When 0 < GTCCRA < GTPR is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] = 00b, and holding output is maintained up to the next crest.

(4) Normal operation period: Toggle output due to a compare match is performed normally.

Figure 18.99 Example of output protection function operation when GTCCRA ≥ GTPR is set during buffer transfer at crests, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low

18.7.4.4 Restricted Specification of Output Protection Function

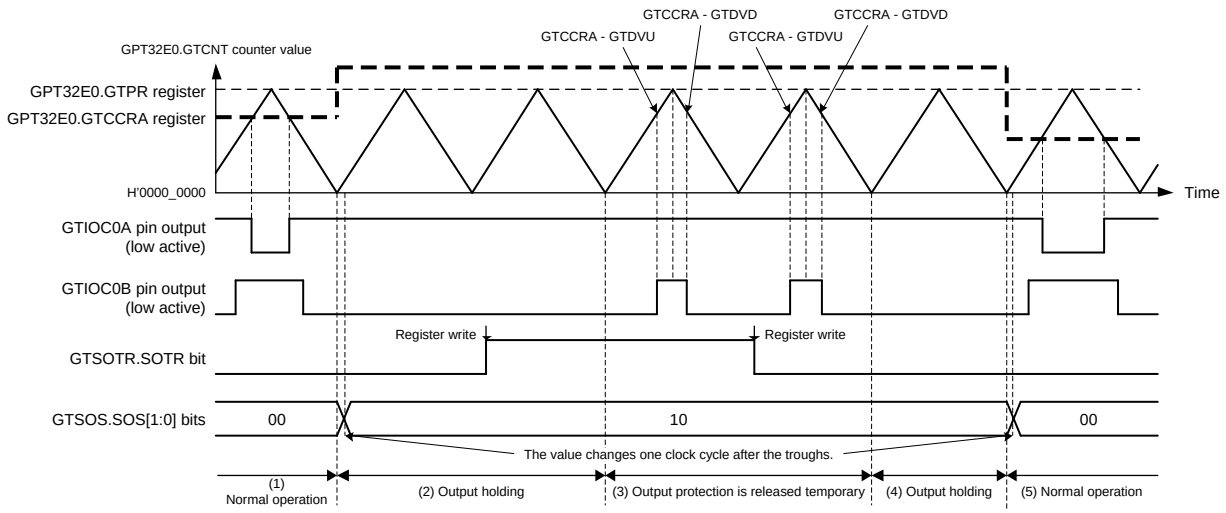
The output protection function deactivates the level of one of the positive and negative outputs, even if an incorrect value is set in the GTCCRA register during counting (a setting outside the range of 0 < GTCCRA < GTPR). However, it will not work properly unless the following conditions are satisfied.

- At count start: The value of the GTCCRA register must be set within the range of (0 < GTCCRA < GTPR).
- During buffer transfer at crest: GTCCRA - GTDVD < GTPR - 1 has to be satisfied..
- During buffer transfer at trough: If GTCCRA ≥ GTPR, GTCCRA - GTDVU > 1 has to be satisfied.

18.7.4.5 Temporary cancellation of Output Protection Function

When the GTSOTR.SOTR bit is set to 1 with GTSOS.SOS[1:0] bits equal to 10b (showing output protection state by GTCCRA ≥ GTPR during buffer transfer at troughs), the output protection function for GTIOCB pin is temporarily canceled. GTSOS.SOS[1:0] bits retain the value of 10b even when the output protection function is canceled. When the SOTR bit is set to 0, the output protection function for GTIOCB pin resumes.

Figure 18.100 shows examples of temporary cancellation of output protection function operation when the GTCCRA ≥ GTPR is set during buffer transfer at troughs.



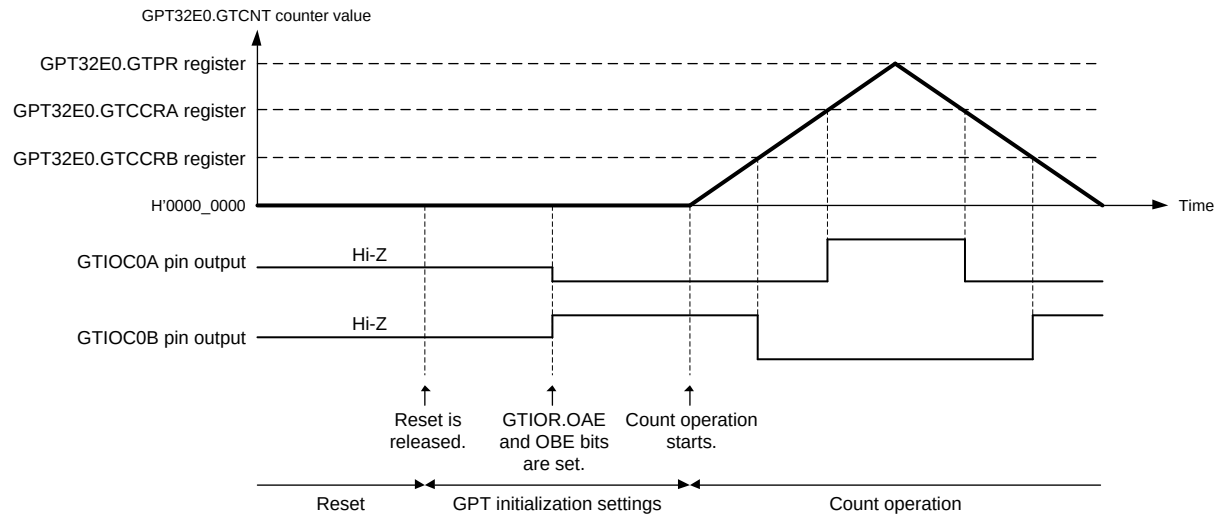
(1) Normal operation period:	Toggle output due to a compare match is performed normally.
(2) Output holding period:	When GTCCRA register ≥ GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 10b, and the output is held while GTCCRA register ≥ GTPR register. If GTSOTR.SOTR is set to 1 during the output holding period, the GTIOCB output is holding until the first trough after the setting.
(3) Output protection releasing period:	After GTSOTR.SOTR is set to 1, the GTIOCB output is performed normally at the first trough after the setting. If GTSOTR.SOTR is set to 0 during the output protection releasing period, the output is holding at the first trough after the setting.
(4) Output holding period:	Until 0 < GTCCRA register < GTPR register is detected in the trough, the output is holding.
(5) Normal operation period:	When 0 < GTCCRA register < GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and toggle output due to a compare match is performed normally.

Figure 18.100 Example of temporary cancellation of output protection function operation when GTCCRA ≥ GTPR is set during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low

18.8 Initialization Method of Output Pins

18.8.1 Pin Settings after Reset

The GPT registers are initialized at a reset. Start counting after selecting the port mode, setting GTIOR.OAE and GTIOR.OBE bits, and outputting the GPT function to external pins.



[Setting examples]

GTIOR.GTIOA[4:0] bits: Initial output of low, output retained at cycle end, output toggled at compare match

GTIOR.GTIOB[4:0] bits: Initial output of high, output retained at cycle end, output toggled at compare match

Figure 18.101 Example of pin settings after reset

18.8.2 Pin Initialization Caused by Error during Operation

If an error occurs during GPT operation, the following four types of pin processing can be performed before pin initialization:

- Set the OAHLD and OBHLD bits in GTIOR to 1 and retain the outputs at count stop.
- Set the OAHLD and OBHLD bits in GTIOR to 0, specify arbitrary output values in OADFLT and OBDFLT in GTIOR, and output the arbitrary values on count stop.
- Set the pin to output an arbitrary value as a general output port by setting the PMn, Pn, and PMCn registers of the I/O port in advance. Set the OAE and OBE bits in GTIOR to 0 and the control bit associated with the pin in the PMCn to 0, to allow arbitrary values to be output from the pin set as a general output port when an error occurs.
- Drive the output to a high impedance state using the POEG function.

When the automatic dead time setting is made, clear the GTDTCR.TDE bit to 0 after counting stops. When counting stops, only the values of registers that are changed by a GPT external source change. If counting resumes, operation continues from where it stops. If counting stops, registers must be initialized before counting starts.

18.9 Usage Notes

18.9.1 GTCCRn Settings during Compare Match Operation (n = A to F)

(1) When automatic dead time setting is made in triangle-wave PWM mode

The GTCCRA register must satisfy the following conditions: GTDVU < GTCCRA, GTDVD < GTCCRA, and GTCCRA < GTPR.

When the setting of GTCCRA = 0 or GTCCRA ≥ GTPR is made during count operation, the output protection function is activated.

However, it will not work properly unless the following conditions are satisfied.

- At count start: The value of the GTCCRA register must be set within the range of (0 < GTCCRA < GTPR).
- During buffer transfer at crest: GTCCRA - GTDVD < GTPR - 1 has to be satisfied..
- During buffer transfer at through: If GTCCRA ≥ GTPR, GTCCRA - GTDVU > 1 has to be satisfied.

For details, see **Section 18.7.4, Output Protection Function for GTIOC Pin Output**.

(2) When automatic dead time setting is not made in triangle-wave PWM mode

The GTCCRA register must be set within the range of 0 < GTCCRA < GTPR. If GTCCRA = 0 or GTCCRA = GTPR is set, a compare match occurs within the cycle only when GTCCRA = 0 or GTCCRA = GTPR is satisfied. When GTCCRA > GTPR, no compare match occurs.

Similarly, GTCCRB must be set within the range of 0 < GTCCRB < GTPR. If GTCCRB = 0 or GTCCRA = GTPR is set, a compare match occurs within the cycle only when GTCCRB = 0 or GTCCRB = GTPR is satisfied. When GTCCRB > GTPR, no compare match occurs.

(3) When automatic dead time setting is made in saw-wave one-shot pulse mode

The GTCCRC and GTCCRD registers must be set to satisfy the following constraints. If the constraints are not satisfied, correct output waveforms with secured dead time might not be obtained.

- In up-counting: GTCCRC < GTCCRD, GTCCRC > GTDVU, GTCCRD < GTPR - GTDVD
- In down-counting: GTCCRC > GTCCRD, GTCCRC < GTPR – GTDVU, GTCCRD > GTDVD

(4) When automatic dead time setting is not made in saw-wave one-shot pulse mode

The GTCCRC and GTCCRD registers must be set to satisfy the following constraints. If the constraints are not satisfied, two compare matches do not occur and pulse output cannot be performed.

- In up-counting: 0 < GTCCRC < GTCCRD < GTPR
- In down-counting: GTPR > GTCCRC > GTCCRD > 0

Similarly, GTCCRE and GTCCRF must be set to satisfy the following constraints. If the constraints are not satisfied, two compare matches do not occur and pulse output cannot be performed.

- In up-counting: 0 < GTCCRE < GTCCRF < GTPR
- In down-counting: GTPR > GTCCRE > GTCCRF > 0

(5) In saw-wave PWM mode

The GTCCRA register must be set with the range of 0 < GTCCRA < GTPR. If GTCCRA = 0 or GTCCRA = GTPR is set, a compare match occurs within the cycle only when GTCCRA = 0 or GTCCRA = GTPR is satisfied. If GTCCRA > GTPR is set, no compare match occurs.

Similarly, GTCCRB must be set with the range of 0 < GTCCRB < GTPR. If GTCCRB = 0 or GTCCRB = GTPR is set, a compare match occurs within the cycle only when GTCCRB = 0 or GTCCRB = GTPR is satisfied. If GTCCRB > GTPR is set, no compare match occurs.

18.9.2 Setting Range for the GTCNT Counter

The GTCNT counter register must be set with the range of 0 ≤ GTCNT ≤ GTPR.

18.9.3 Starting and Stopping the GTCNT Counter

The control timing of starting and stopping the GTCNT counter by the GTCR.CST bit synchronizes the count clock that is selected in GTCR.TPCS[2:0]. When GTCR.CST is updated, the GTCNT counter starts/stops after a count clock that is selected in GTCR.TPCS[2:0]. Therefore, an event generated before the GTCNT counter actually starts is ignored.

Note, however, that an event might be accepted or an interrupt might occur after the GTCR.CST bit is set to 0.

18.9.4 Priority On Conflicts

(1) GTCNT register

Table 18.20 shows a priority order of events updating GTCNT register.

Table 18.20 Priority order of sources updating GTCNT

Source updating GTCNT	Priority order
Writing by CPU (Writing to GTCNT/GTCLR)	High
Clear by hardware sources set in GTCSR	↑
Count up or down by hardware sources set in GTUPSR/GTDNSR	↑
Count operation	Low

If up-counting and down-counting by hardware sources occur at the same time, the GTCNT counter value does not change. When there is a conflict between updating the GTCNT register and reading by the CPU, pre-update data is read.

(2) GTCR.CST bit

When there is a conflict between starting/stopping by hardware sources set in the GTSSR/GTPSR registers and writing by the CPU (writing to GTCR/GTSTR/GTSTP registers), writing by CPU has a priority over starting/stopping by hardware sources.

When there is a conflict between starting by hardware sources set in the GTSSR register and stopping by hardware sources set in GTPSR register, the GTCR.CST bit value does not change. Where there is a conflict between updating the GTCR.CST bit and reading by the CPU, pre-update data is read.

(3) GTCCRn registers (n = A to F)

When there is a conflict between input capture/buffer transfer operation and writing to GTCCRn registers, writing to GTCCRn registers has a priority over input capture/buffer transfer operation. When there is a conflict between input capture and writing to the counter register by the CPU or updating the counter register by hardware sources, the pre-update counter value is captured. Where there is a conflict between updating the GTCCRn registers and reading by the CPU, pre-update data is read.

(4) GTPR registers

When there is a conflict between buffer transfer operation and writing to the GTPR register, writing to GTPR register has a priority over buffer transfer operation. When there is a conflict between updating GTPR register and reading by the CPU, pre-update data is read.

(5) GTADTRn registers (n = A, B)

When there is a conflict between buffer transfer operation and writing to the GTADTRn registers, writing to the GTADTRn registers has priority over buffer transfer operation. Where there is a conflict between updating GTADTRn registers and reading by the CPU, pre-update data is read.

(6) GTDVn registers (n = U, D)

When there is a conflict between buffer transfer operation and writing to GTDVn registers, writing to GTDVn registers has priority over buffer transfer operation. When there is a conflict between updating GTDVn registers and reading by the CPU, pre-update data is read.

19. Port Output Enable for GPT (POEG)

19.1 Overview

The output pins of the general PWM timer (GPT) can be disabled by using the port output enabling function for the GPT (POEG). Specifically, either of the following ways can be used.

- Input level detection of the GTETRGA to GTETRGD pins
- Output-disable request from the GPT
- Register settings

The GTETRGA to GTETRGD pins can also be used as GPT external trigger input pins.

Table 19.1 lists the POEG specifications, **Figure 19.1** shows the block diagram, and **Table 19.2** lists the input pins.

Table 19.1 POEG Specifications

Parameter	Specifications
Output-disable control through input level detection	GPT output pins can be disabled when a GTETRGA to GTETRGD rising edge or high level is sampled after polarity and filter selection
Output-disable request from the GPT	• When the GTIOCA pin and the GTIOCB pin are driven to an active level simultaneously, the GPT generates an output-disable request to the POEG. Through reception of these requests, the POEG can control whether the GTIOCA and GTIOCB pins are output-disabled • GPT output pins can be set to be disabled when the GPT output pins detect a dead time error or short circuit detection between the output terminals
Output-disable control by software (registers)	GPT output pins can be disabled by modifying the register settings
Interrupt	• Allows output-disable control by input level detection • Allows output-disable requests from the GPT
External trigger output to the GPT (count start, count stop, count clear, up-count, down-count, or input capture function)	GTETRGA to GTETRGD signals can be output to the GPT after polarity and filter selection
Noise filtering	• Three times sampling for every Pϕ/1, Pϕ/8, Pϕ/32, or Pϕ/128 can be set for any of the input pins GTETRGA to GTETRGD • Positive or negative polarity can be selected for any of the input pins, GTETRGA to GTETRGD • Signal state after polarity and filter selection can be monitored

GTETRGn (n = A to D) in the subsequent descriptions indicates the GTETRGA to GTETRGD pins. The signal input from each of the pins corresponds to the POEGGn register for which n matches the n of the pin.

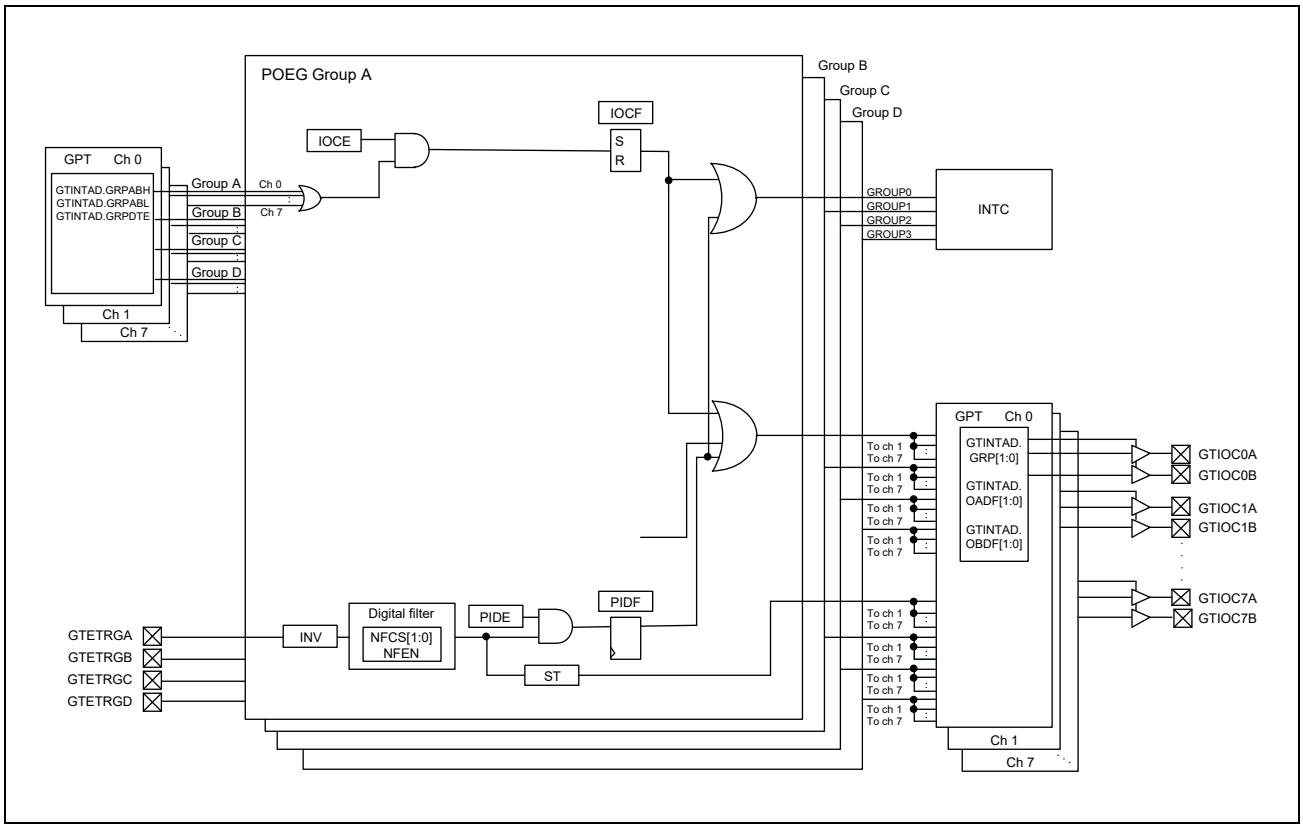


Figure 19.1 POEG Block Diagram

Table 19.2 POEG Input Pins

Pin Name	I/O	Description
GTETRGA	Input	GPT output pin output-disable request signal and GPT external trigger input pin A
GTETRGB	Input	GPT output pin output-disable request signal and GPT external trigger input pin B
GTETRGC	Input	GPT output pin output-disable request signal and GPT external trigger input pin C
GTETRGD	Input	GPT output pin output-disable request signal and GPT external trigger input pin D

19.2 Register Descriptions

Table 19.3 to **Table 19.5** shows the register configuration.

Table 19.3 Register Configuration 1 (Cortex-A55 Address Space)

Register Name	Abbreviation	Address	Access Size
POEG group A setting register	POEGGA	H'0_1004_8800	32
POEG group B setting register	POEGGB	H'0_1004_8C00	32
POEG group C setting register	POEGGC	H'0_1004_9000	32
POEG group D setting register	POEGGD	H'0_1004_9400	32

Table 19.4 Register Configuration 2 (Cortex-M33/Cortex-M33_FPU Address Space Secure)

Register Name	Abbreviation	Address	Access Size
POEG group A setting register	POEGGA	H'4004_8800	32
POEG group B setting register	POEGGB	H'4004_8C00	32
POEG group C setting register	POEGGC	H'4004_9000	32
POEG group D setting register	POEGGD	H'4004_9400	32

Note: Base address of Secure is exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above is in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 19.5 Register Configuration 3 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Register Name	Abbreviation	Address	Access Size
POEG group A setting register	POEGGA	H'5004_8800	32
POEG group B setting register	POEGGB	H'5004_8C00	32
POEG group C setting register	POEGGC	H'5004_9000	32
POEG group D setting register	POEGGD	H'5004_9400	32

Note: Base address of Non-Secure is exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above is in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

19.2.1 POEG Group n Setting Register (POEGGn) (n = A to D)

The POEGGA to POEGGD registers control the output-disable state of the GPT pins, interrupts, and the external trigger input to the GPT. In the descriptions, POEGGn represents all of the POEGGA to POEGGD registers.

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	NFCS[1:0]		NFEN	INV	—	—	—	—	—	—	—	—	—	—	—	ST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	IOCE	PIDE	SSF	—	IOCF	PIDF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W*2	R/W*2	R/W	R/W	R(/W)*1	R(/W)*1

Bit	Bit Name	Initial Value	R/W	Description
b31, b30	NFCS[1:0]	All 0	R/W	Noise Filter Clock Select b1 b0 0 0: Sample GTETRGn pin input level three times every P0ϕ/1 0 1: Sample GTETRGn pin input level three times every P0ϕ/8 1 0: Sample GTETRGn pin input level three times every P0ϕ/32 1 1: Sample GTETRGn pin input level three times every P0ϕ/128.
b29	NFEN	0	R/W	Noise Filter Enable 0: Disable noise filtering 1: Enable noise filtering.
b28	INV	0	R/W	GTETRGn Input Reverse 0: Input GTETRGn as-is 1: Input GTETRGn in reverse.
b27 to b17	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b16	ST	0	R	GTETRGn Input Status Flag 0: GTETRGn input after filtering was 0 1: GTETRGn input after filtering was 1.
b15 to b6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b5	IOCE	0	R/W*2	Enable for GPT Output-Disable Request 0: Disable output-disable requests from GPT disable request 1: Enable output-disable requests from GPT disable request.
b4	PIDE	0	R/W*2	Port Input Detection Enable 0: Disable output-disable requests from the GTETRGn pins 1: Enable output-disable requests from the GTETRGn pins.
b3	SSF	0	R/W	Software Stop Flag 0: No output-disable request from software occurred 1: Output-disable request from software occurred.
b2	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b1	IOCF	0	R(/W)*1	Detection Flag for GPT Output-Disable Request 0: No output-disable request from GPT disable request occurred 1: Output-disable request from GPT disable request occurred.

Bit	Bit Name	Initial Value	R/W	Description
b0	PIDF	0	R(/W) *1	Port Input Detection Flag 0: No output-disable request from the GTETRGn pin occurred 1: Output-disable request from the GTETRGn pin occurred.

Note 1. Only 0 can be written, to clear the flag.

Note 2. Can be modified only once after a reset.

19.3 Output-Disable Control Operation

The output of the GTIOCxA and GTIOCxB pins can be disabled when any of the following conditions are satisfied.

- Input level or edge detection of the GTETRGn pins

  When POEGGn.PIDE is 1, the POEGGn.PIDF flag is set to 1.

- Output-disable request from the GPT

  When POEGGn.IOCE is 1, the POEGGn.IOCF flag is set to 1.
  The output-disable requests enabled by GRPDTE, GRPABH, and GRPABL bits of the GTINTAD register in the GPT are applied to the group selected by GRP[1:0] bits of the GTINTAD register.

- SSF bit setting

  When POEGGn.SSF is set to 1.

The state of the GTIOCxA and the GTIOCxB pins when the output is disabled is controlled by the GPT module.

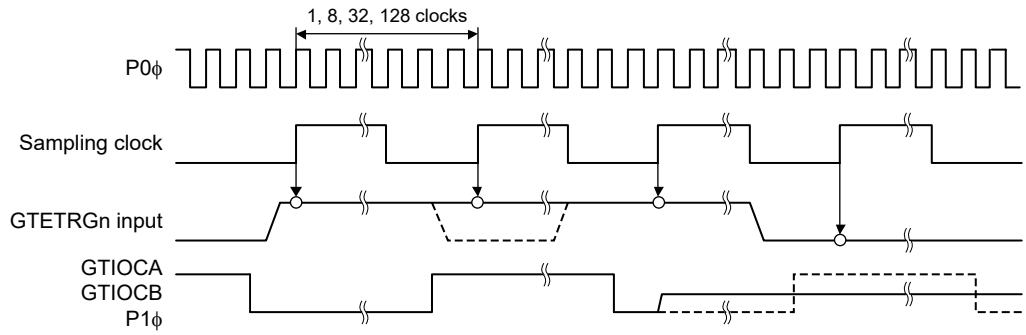
For details, see the descriptions of the GTINTAD.GRP[1:0], GTIOR.OADF[1:0], and GTIOR.OBDF[1:0] bits of the general PWM timer (GPT).

19.3.1 Pin Input Level Detection Operation

If the input conditions set in POEGGn.PIDE, POEGGn.NFCS[1:0], POEGGn.NFEN, and POEGGn.INV occur on the GTETRGn pins, the GPT output pins are output-disabled.

Digital Filter

Figure 19.2 shows high-level detection by the digital filter. When a high level associated with the POEGGn.INV polarity setting is detected three times consecutively with the sampling clock selected in POEGGn.NFCS[1:0] and POEGGn.NFEN, the detected level is recognized as high, and the GPT output pins are output-disabled. If even one low level is detected during this interval, the detected level is not recognized as high. In addition, in an interval where the sampling clock is not being output, changes of the levels on the GTETRGn pins are ignored.



When high level is sampled at all points	[1]	[2]	[3]	Flag set (GTETRGn received)
When low level is sampled at least once	[1]	[0]	[1]	Flag not set

Note: Individual channel output can be set in the GPT settings.
Low level sampling can be selected in POEGGn.INV.

Figure 19.2 Example of Digital Filter Operation

19.3.2 Output-Disable Requests from the GPT

For details on this operation, see the description of **18.7.3, GTIOC Pin Output Negate Control** in **Section 18, General PWM Timer (GPT)**.

19.3.3 Output-Disable Control Using Registers

The GPT output pins can be directly controlled by writing to the software stop flag, POEGGn.SSF.

19.3.4 Release from Output-Disable

To release the GPT output pins placed in the output-disable state, either return them to their initial state with a reset or clear all of the following:

- POEGGn.PIDF flag
- POEGGn.IOCF flag
- POEGGn.SSF flag.

Writing 0 to the POEGGn.PIDF flag is ignored (the flag is not cleared) if the external input pins, GTETRGn, are not disabled and the POEGGn.ST bit is not set to 0.

Writing 0 to the POEGGn.IOCF flag is valid (the flag is cleared) only if all of the GTST.DTEF, GTST.OABHF, and GTST.OABLF flags in the GPT are set to 0.

Figure 19.3 shows the release timing for output-disable. The output-disable is released at the beginning of the next count cycle of the GPT after the flag is cleared.

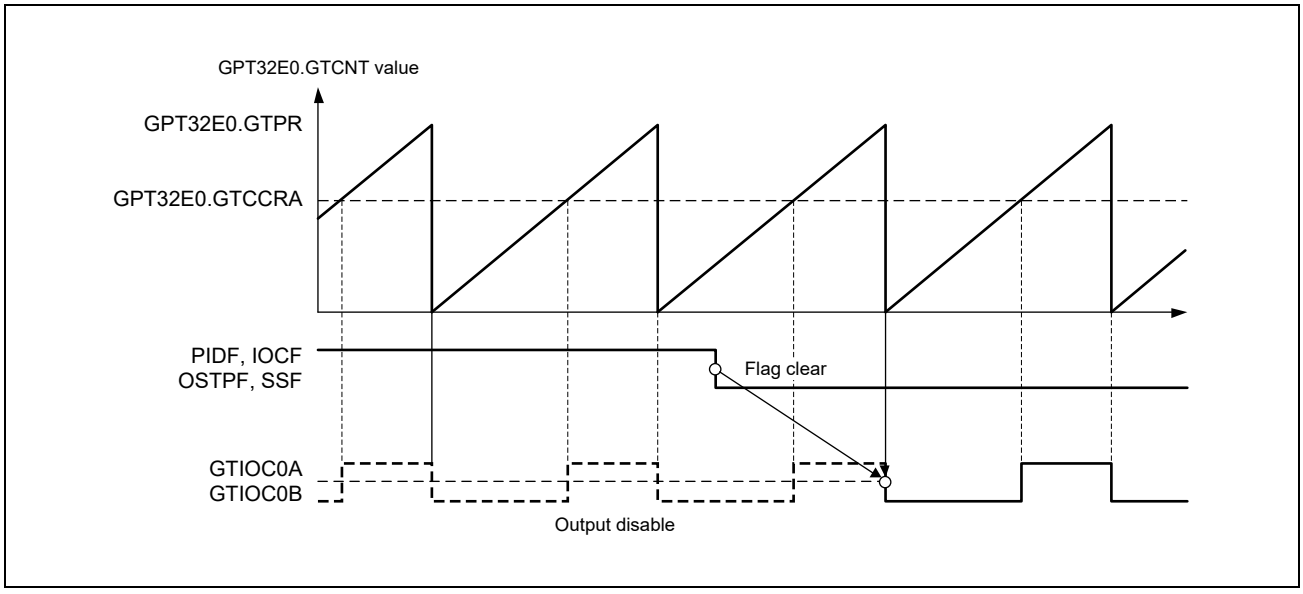


Figure 19.3 Output-Disable Release Timing for GPT Pin Outputs

19.4 Interrupt Sources

The POEG generates an interrupt request when triggered by these sources:

- Output-disable control by input level detection
- Output-disable request from the GPT
- Output-disable control by the registers.

Table 19.6 lists the conditions for interrupt requests.

Table 19.6 Interrupt Sources and Conditions

Interrupt Source	Symbol	Associated Flag	Trigger Conditions
POEG group A interrupt	GROUP0	POEGGA.IOCF	Output-disable request from a GPT disable request occurred
		POEGGA.PIDF	Output-disable request from the GTETRGA pin occurred
POEG group B interrupt	GROUP1	POEGGB.IOCF	Output-disable request from a GPT disable request occurred
		POEGGB.PIDF	Output-disable request from the GTETRGB pin occurred
POEG group C interrupt	GROUP2	POEGGC.IOCF	Output-disable request from a GPT disable request occurred
		POEGGC.PIDF	Output-disable request from the GTETRGC pin occurred
POEG group D interrupt	GROUP3	POEGGD.IOCF	Output-disable request from a GPT disable request occurred
		POEGGD.PIDF	Output-disable request from the GTETRGD pin occurred

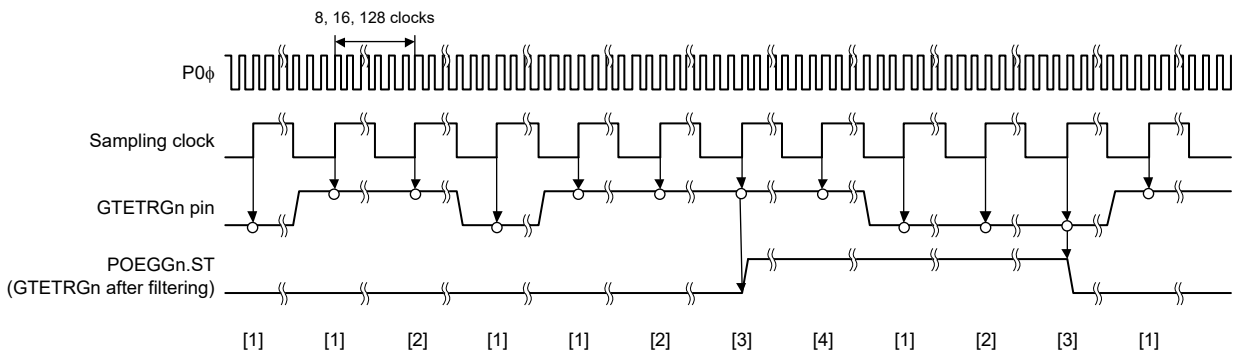
19.5 External Trigger Output to the GPT

The POEG outputs the GTETRGn signals to the GPT as the GPT operation trigger signal for the following:

- Count start
- Count stop
- Count clear
- Up-count
- Down-count
- Input capture.

For the POEGGn.INV polarity setting signal, when the same level is input three times continuously with the sampling clock selected in POEGGn.NFCS[1:0] and POEGGn.NFEN, that value is output. Set the control registers the same as for the input level detection operation described in **Section 19.3.1, Pin Input Level Detection Operation**. The state after filtering can be monitored in POEGGn.ST.

Figure 19.4 shows the output timing of an external trigger to the GPT.



Note: Each channel output can be set in the GPT settings.
Polarity can be reversed in POEGGn.INV.

Figure 19.4 Output Timing of External Trigger to GPT

19.6 Usage Notes

19.6.1 Specifying Pins Associated with the GPT

The POEG controls output-disable only when a pin is associated with the GPT in the PMCn and PFCm register settings of IO_TOP. When the pin is specified as a general I/O pin, the POEG does not perform output-disable control.

20. General Timer (GTM)

20.1 Functional Overview

The General timer has the following features.

- Two operating modes
  - Interval timer mode
  - Free-running comparison mode
- Choice between startup of DMA by compare match and generation of interrupt

20.1.1 Features of GTM

Channels

This product has the following number of channels of the General timer.

Table 20.1 Channels of General Timer

General Timer	
Number of Channels	8
Name	OSTMn

Note: n = 0 to 7

Meaning of n

Throughout this section, the individual channels of the General timer are identified by the index “n” (n = 0 to 7), for example OSTMnTO for the General timer n output register.

Register address

The register addresses of the General timer are given as offsets from the individual base addresses <OSTMn_base>. The register base addresses of each OSTMn are listed in the following table.

Table 20.2 Register Base Addresses

Base Address Name	Base Address
<OSTM0_base>	H’0_1280_1000 (H’4280_1000*1, H’5280_1000*2)
<OSTM1_base>	H’0_1280_1400 (H’4280_1400*1, H’5280_1400*2)
<OSTM2_base>	H’0_1280_1800 (H’4280_1800*1, H’5280_1800*2)
<OTSM3_base>	H’0_1280_1C00 (H’4280_1C00*1, H’5280_1C00*2)
<OSTM4_base>	H’0_1280_2000 (H’4280_2000*1, H’5280_2000*2)
<OSTM5_base>	H’0_1280_2400 (H’4280_2400*1, H’5280_2400*2)
<OSTM6_base>	H’0_1280_2800 (H’4280_2800*1, H’5280_2800*2)
<OTSM7_base>	H’0_1280_2C00 (H’4280_2C00*1, H’5280_2C00*2)

Note: Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Note 1. Cortex-M33/Cortex-M33_FPU Address Space Secure

Note 2. Cortex-M33/Cortex-M33_FPU Address Space Non-Secure

Interrupts

The General timers can generate the following interrupt requests.

Table 20.3 OSTMn Interrupt Requests

OSTMn Signal	Function	Startup of Direct Memory Access Controller
OSTM0TINT	OSTM0 interrupt	✓
OSTM1TINT	OSTM1 interrupt	✓
OSTM2TINT	OSTM2 interrupt	✓
OSTM3TINT	OSTM3 interrupt	✓
OSTM4TINT	OSTM4 interrupt	✓
OSTM5TINT	OSTM5 interrupt	✓
OSTM6TINT	OSTM6 interrupt	✓
OSTM7TINT	OSTM7 interrupt	✓

20.2 Registers

The General timer is controlled by the following registers.

20.2.1 Registers Overview

The list of OSTMn (n = 0 to 7) registers and the memory addresses are as follows.

For the base addresses, see **Table 20.2**.

For the actual addresses, the offset values indicated in the following table are added to the base addresses.

Register Name	Function	R/W	Reset Value	Access Unit (bit)			Address
				8	16	32	
OSTMnCMP	OSTM compare register	R/W	H'0000_0000	—	—	✓	<OSTMn_base> + H'00
OSTMnCNT	OSTM counter register	R	H'FFFF_FFFF	—	—	✓	<OSTMn_base> + H'04
OSTMnTE	OSTM count enable status register	R	H'00	✓	—	—	<OSTMn_base> + H'10
OSTMnTS	OSTM count start trigger register	W	H'00	✓	—	—	<OSTMn_base> + H'14
OSTMnTT	OSTM count stop trigger register	W	H'00	✓	—	—	<OSTMn_base> + H'18
OSTMnCTL	OSTM control register	R/W	H'00	✓	—	—	<OSTMn_base> + H'20

20.2.2 Details of OSTM Registers

20.2.2.1 OSTM Compare Register (OSTMnCMP)

Depending on the mode of operation, this register holds the start value for the down-counter or the value for comparison with that of the counter.

Access Size: This register is readable/writable in 32-bit units.

Address(es): <OSTMn_base>

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OSTMnCMP[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTMnCMP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	OSTMnCMP [31:0]	All 0	R/W	• In interval timer mode: start value of the down-counter • In free-running comparison mode: value for comparison

20.2.2.2 OSTM Counter Register (OSTMnCNT)

This register indicates the counter value of the timer.

Access Size: This register is readable in 32-bit units.

Address(es): <OSTMn_base> + H'4

Initial Value: The initial value depends on the operating mode of the General timer. Refer to **Table 20.4**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OSTMnCNT[31:16]															
Initial Value	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTMnCNT[15:0]															
Initial Value	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	OSTMnCNT [31:0]	*1	R	32-bit counter value

Note 1. The initial value depends on the operating mode of the General timer. Refer to **Table 20.4**.

The following table shows the correspondence between operating mode, counting direction and initial value. The initial value is the value read from the counter after a change to the operating mode.

Table 20.4 Correspondence between Operating Mode, Counting Direction and Initial Value

Timer Operating Mode	OSTMnCTL.OSTMnMD1	Counting Direction	Initial value
Interval timer mode	0*1	Down	H'FFFF_FFFF
Free-running comparison mode	1	Up	H'0000_0000

Note 1. Value after reset

20.2.2.3 OSTM Count Enable Status Register (OSTMnTE)

This register indicates whether the counter is enabled or disabled.

Access Size: This register is readable in 8-bit units.

Address(es): <OSTMn_base>+ H'10

Initial Value: H'00

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTE
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OSTMnTE	0	R	This bit indicates whether the counter is enabled or disabled. 0: Counter disabled 1: Counter enabled This bit is set to 1 in response to OSTMnTS.OSTMnTS being set to 1. This bit is reset to 0 in response to OSTMnTT.OSTMnTT being set to 1.

Note 1. When OSTMnTE = 0, the counter retains its value.
If the counter is restarted, it
• restarts counting down from the value in the OSTMnCMP register if it is in interval timer mode or
• restarts counting up from the counter value H'0000_0000 if it is in free running comparison mode.

Note 2. For debugging, the GTM counter can be paused by the counter stop request.

20.2.2.4 OSTM Count Start Trigger Register (OSTMnTS)

This register starts the counter.

Access Size: This register is writable in 8-bit units. It is always read as H'00.

Address(es): <OSTMn_base>+ H'14

Initial Value: H'00

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTS
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OSTMnTS	0	W	This bit starts the counter. 0: This setting has no effect. 1: Starts the counter and sets OSTMnTE.OSTMnTE = 1. • In interval timer mode, a forced restart is executed if this bit is set while OSTMnTE.OSTMnTE = 1. • In free-running comparison mode, setting this bit is ignored as long as OSTMnTE.OSTMnTE = 1.

20.2.2.5 OSTM Count Stop Trigger Register (OSTMnTT)

This register stops the counter.

Access Size: This register is writable in 8-bit units. It is always read as H'00.

Address(es): <OSTMn_base>+ H'18

Initial Value: H'00

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTT
Initial Value								
R/W	R	R	R	R	R	R	R	W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OSTMnTT	0	W	Stops the counter. 0: This setting has no effect. 1: Stops the counter and clears the OSTMnTE.OSTMnTE bit.

20.2.2.6 OSTM Control Register (OSTMnCTL)

This register specifies the operating mode for the counter and controls enabling/disabling of OSTMnTINT interrupt requests when counting starts.

Access Size: This register is readable/writable in 8-bit units. Writing to this register is only possible if the counter is disabled (OSTMnTOE.OSTMnTOE = 0).

Address(es): <OSTMn_base>+ H'20

Initial Value: H'00

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OSTMnMD1	OSTMnMD0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	OSTMnMD1	0	R/W	Specifies the operating mode for the counter. 0: Interval timer mode 1: Free-running comparison mode
0	OSTMnMD0	0	R/W	Controls enabling/disabling of OSTMnTINT interrupt requests when counting starts. 0: Disables the interrupts when counting starts. 1: Enables the interrupts when counting starts.

20.3 Functional Description

Each General timer is a 32-bit timer/counter.

The settings for operating mode specify the direction of counting (up or down) and the generation of interrupt requests.

20.3.1 Block Diagram

The following block diagram shows the main components of GTM.

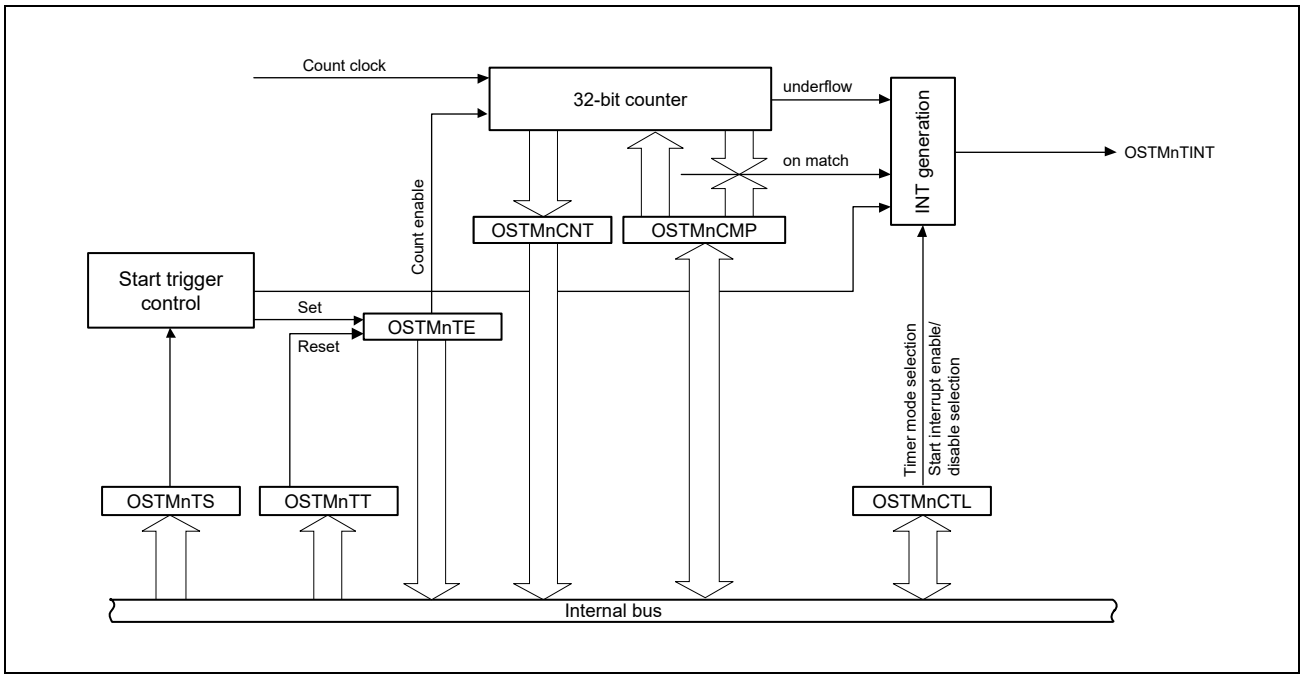


Figure 20.1 Block Diagram of GTM

20.3.2 Count Clock

The count clock of OSTMn is P0ϕ (OSTMn_PCLK).

20.3.3 Generation of Interrupt Request

An OSTMnTINT interrupt request is generated whenever the counter reaches H’0000_0000 (in interval timer mode) or matches the comparison value (in free-running comparison mode).

An interrupt request can also be generated on starting and restarting of the counter. This is controlled by the OSTMnCTL.OSTMnMD0 bit.

This operation is shown in the following figure.

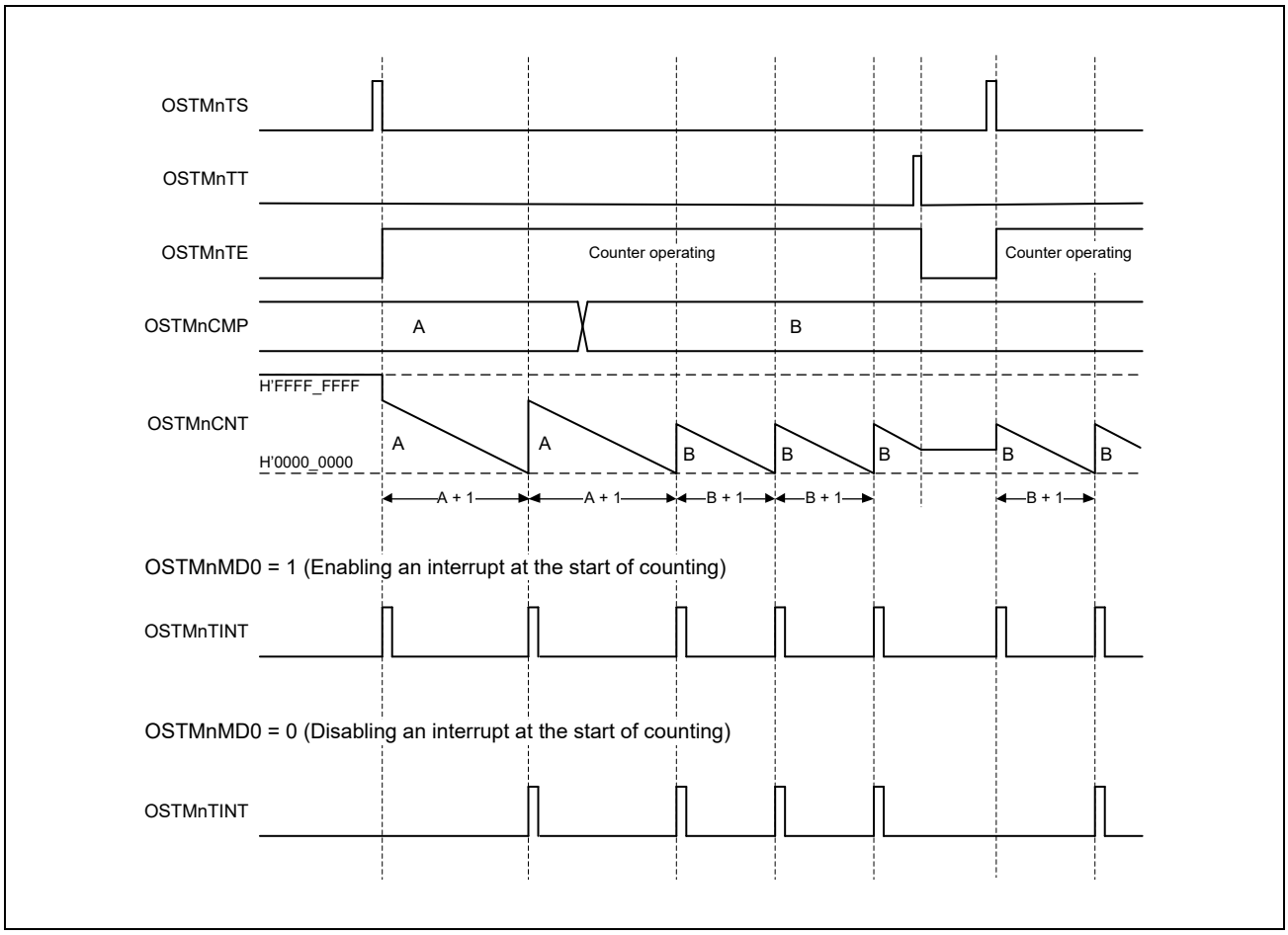


Figure 20.2 Generating an Interrupt when Counting Starts (in Interval Timer Mode)

20.3.4 Starting and Stopping the Timer

The General timer is started and stopped as follows.

Starting the timer

The timer is started in either of the following way:

- setting the OSTMnTS.OSTMnTSF bit to 1

Status bit OSTMnTE.OSTMnTE is set to 1.

The counter starts to count up or down in accord with the settings for operating mode.

Stopping the timer

Setting the OSTMnTT.OSTMnTT bit to 1 stops the timer.

This also clears the OSTMnTE.OSTMnTE status flag.

20.3.5 Interval Timer Mode

Select the interval timer mode when an General timer is to be used as a reference timer for generating interrupt requests at a fixed interval.

20.3.5.1 Basic Operation in Interval Timer Mode

In interval timer mode, the timer counts down from the value specified in the OSTMnCMP register. An OSTMnTINT interrupt request is generated when the counter reaches H'0000_0000.

Select interval timer mode by setting OSTMnCTL.OSTMnMD1 = 0.

New values can be written to the OSTMnCMP register at any time. If it is rewritten during count operation, the counter loads the new OSTMnCMP value when the next H'0000_0000 is reached.

Cycles of OSTMnTINT output

The cycle of OSTMnTINT output is as follows.

- OSTMnTINT generation cycle = counter-clock cycle × (OSTMnCMP + 1)

The following figure shows the basic operation of OSTM when counter-start interrupts is enabled in interval timer mode.

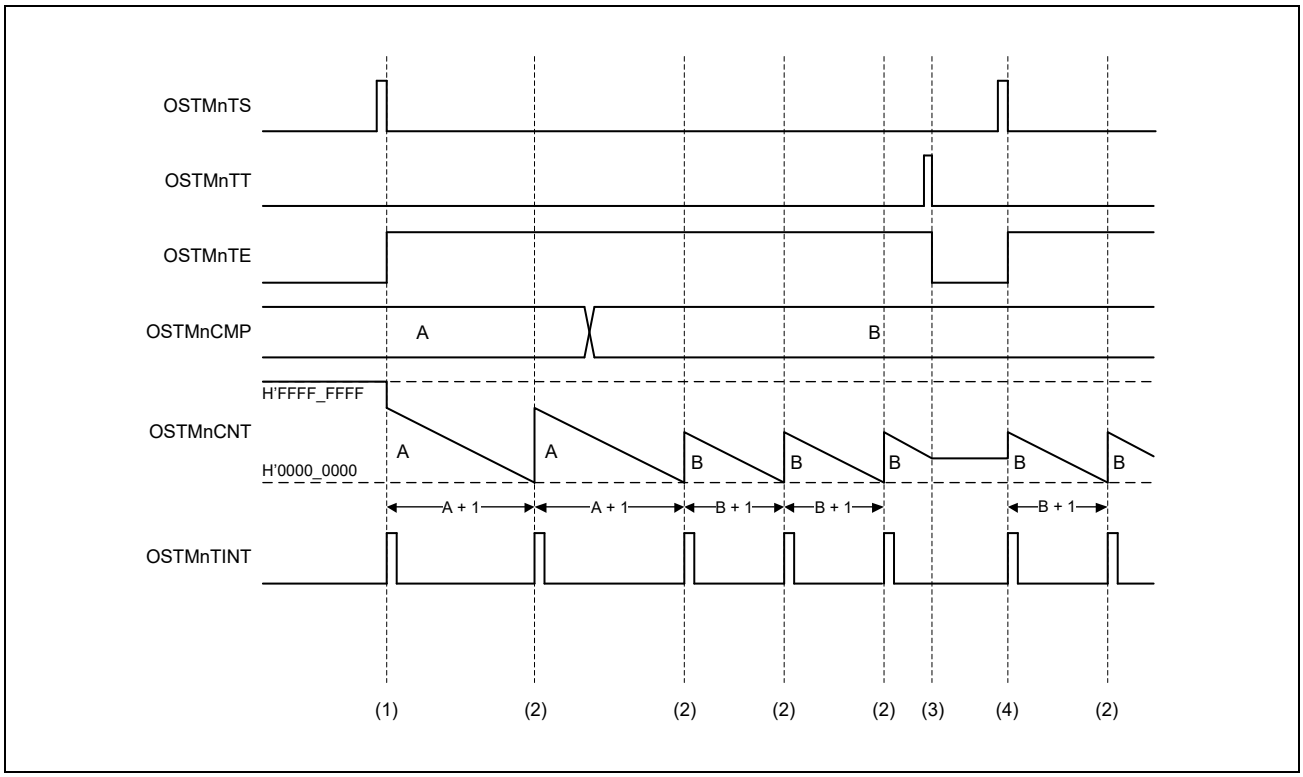


Figure 20.3 Timing Diagram of OSTM in Interval Timer Mode

The timing diagram above shows the following:

(1) The counter starts counting when OSTMnTS.OSTMnTS = 1. The OSTMnTE.OSTMnTE bit is set to indicate enabling of the counter.
The counter starts counting down from the value of OSTMnCMP.
If OSTMnCTL.OSTMnMD0 is 1, OSTMTINT interrupt requests are generated at the start of counting. The OSTMnCNT register contains the current value as the counter.

(2) When the counter reaches H'0000_0000, an OSTMTINT interrupt request is generated. The counter loads the new start value from OSTMnCMP and continues counting down.

(3) When the counter is stopped (OSTMnTT.OSTMnTT = 1), the OSTMnTE.OSTMnTE bit is cleared to indicate disabling of the counter. The counter retains its current value until it is restarted.

(4) When counting is restarted (OSTMnTS.OSTMnTS = 1), the counter loads the new start value from OSTMnCMP and starts counting down.

Forced restart

The counter is forcibly restarted by setting OSTMnTS.OSTMnTS = 1 during counting.

The counter loads the start value from the OSTMnCMP register and continues to count down.

The following figure shows the forced restart of the General Timer in interval timer mode, with counter-start interrupts enabled (OSTMnCTL.OSTMnMD0 = 1).

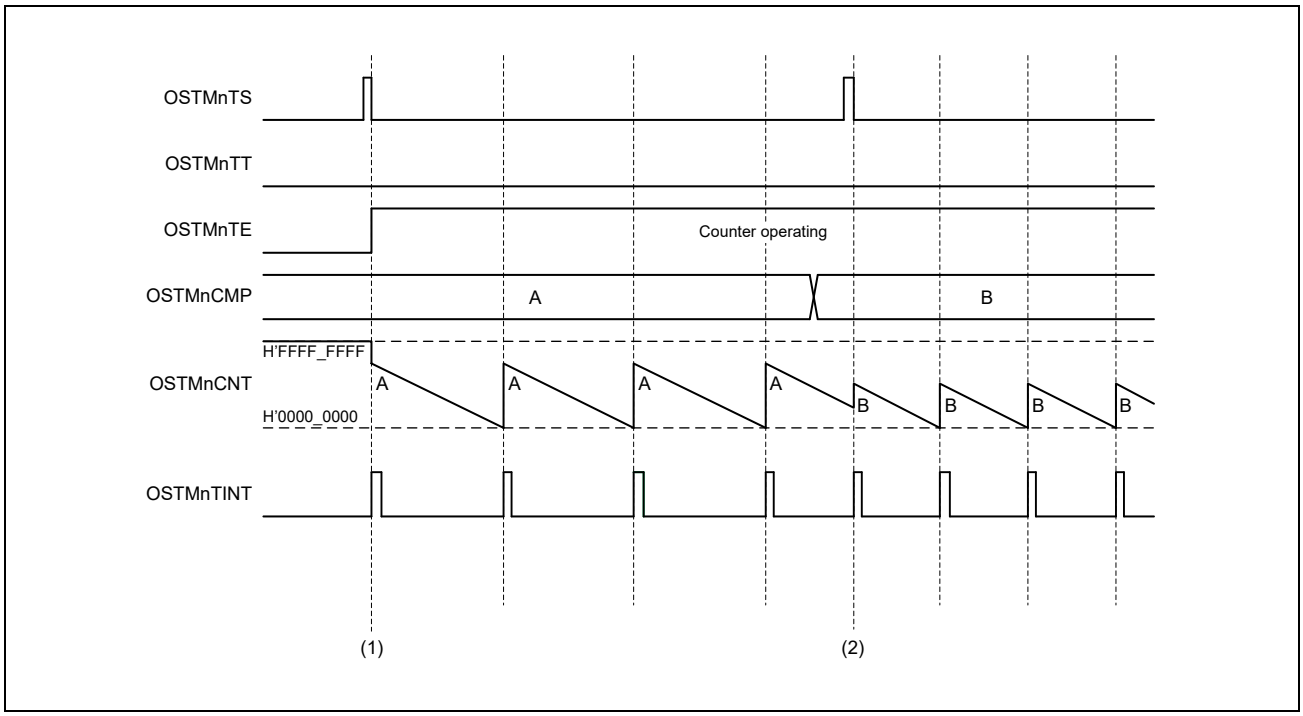


Figure 20.4 Timing Diagram of Forced Restart in Interval Timer Mode

Operations shown in the above timing diagram are as follows.

(1) The counter is started and stopped as described under **Figure 20.3, Timing Diagram of OSTM in Interval Timer Mode**.

(2) Setting OSTMnTS.OSTMnTS = 1 restarts the counter while counting is in progress (i.e. while OSTMnTE.OSTMnTE = 1).
The counter immediately restarts counting down, starting with the current value of OSTMnCMP.
When OSTMnCTL.OSTMnMD0 = 1, an OSTMTINT interrupt request is generated when counting starts.

20.3.5.2 Operation when OSTMnCMP = H’0000_0000

When OSTMnCMP = H’0000_0000, OSTM behaves as follows.

- When the counter is enabled, the OSTMTINT interrupt request is always set to 1.

The following figure shows operations of OSTM when OSTMnCMP = H’0000_0000, and counter-start interrupts are enabled.

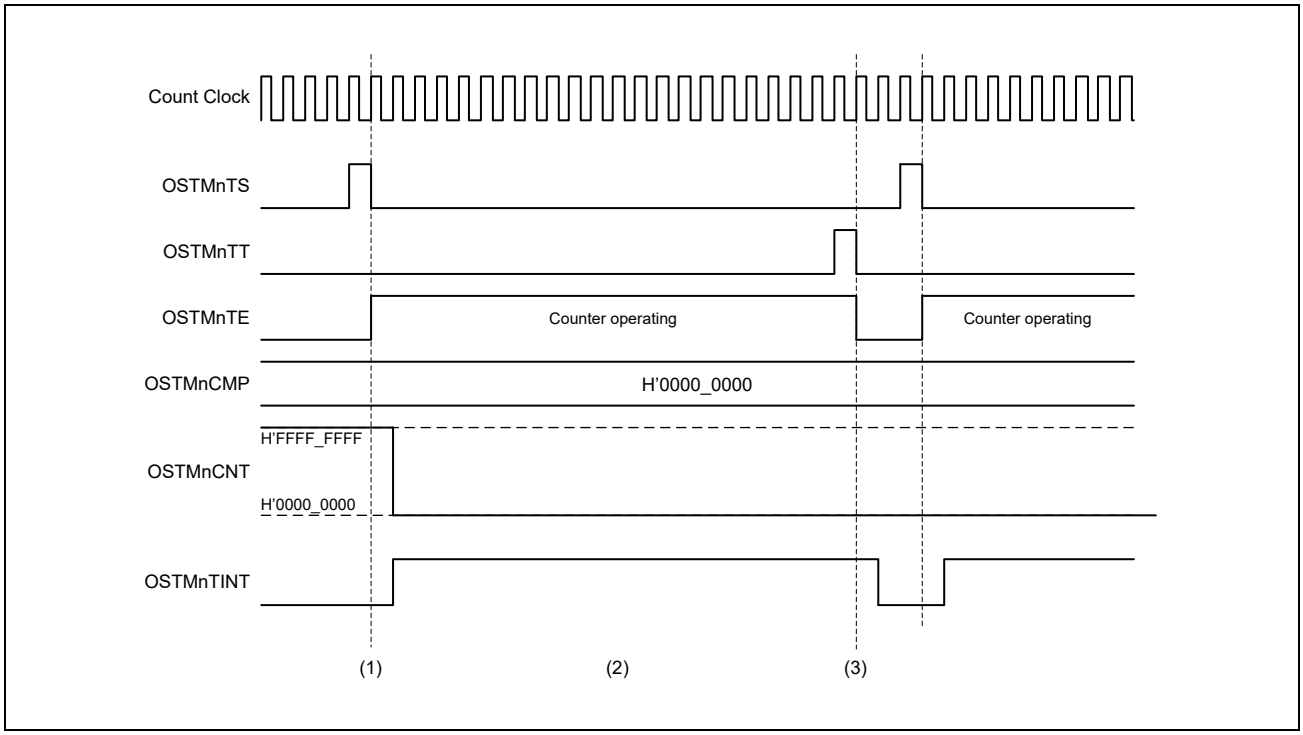


Figure 20.5 Timing Diagram when OSTMnCMP = H’0000_0000 in Interval Timer Mode

The timing diagram above shows the following operations:

(1) The counter is reloaded with the value in OSTMnCMP as soon as it starts counting, so the value H’0000_0000 is retained in OSTMnCMP.

(2) The OSTMTINT interrupt request is continuously asserted.

(3) After the counter stops, the OSTMTINT interrupt request signal is deasserted.

(4) When interrupts on starting of the counter are disabled, no interrupt is generated when counting starts.

20.3.6 Free-Running Comparison Mode

20.3.6.1 Basic Operation in Free-Running Comparison Mode

In free-running comparison mode, the counter counts up from H'0000_0000 to H'FFFF_FFFF. An OSTMnTINT interrupt request is output when the current value of the counter matches the value of the OSTMnCMP register. The free-running comparison mode is selected by setting the OSTMnCTL.OSTMnMD1 bit to 1.

New values can be written to the OSTMnCMP register at any time.

The following figure shows the basic operation of OSTM in free-run compare mode with the start of counting enabled (OSTMnCTL.OSTMnMD0 = 1).

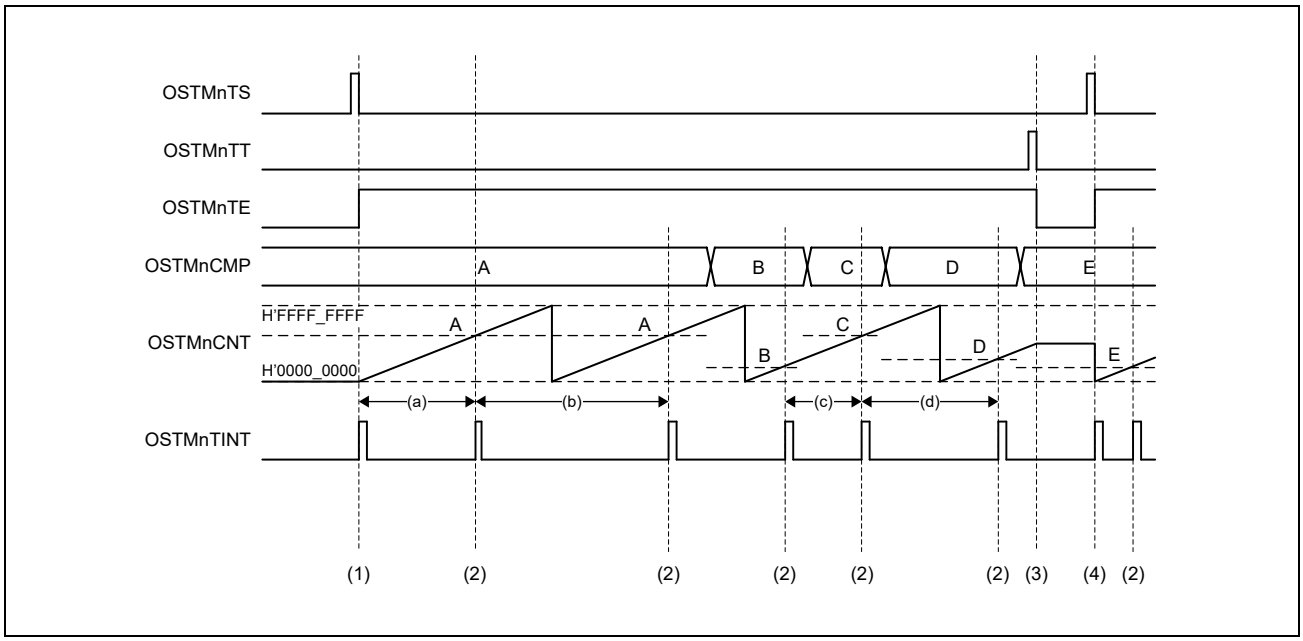


Figure 20.6 Timing Diagram of OSTM in Free-Run Compare Mode

The timing diagram above shows the following:

(1) The counter starts counting when OSTMnTS.OSTMnTS = 1.
The OSTMnTE.OSTMnTE bit is set to indicate enabling of the counter. The counter counts up from H'0000_0000 to H'FFFF_FFFF. The OSTMnCNT register is the counter, so it contains the current value.
When OSTMnCTL.OSTMnMD0 = 1, an OSTMTINT interrupt request is generated at the start of counting.

(2) When the current counter value matches the value in the OSTMnCMP register, an OSTMTINT interrupt request is generated.

(3) When the counter is stopped (OSTMnTT.OSTMnTT = 1), the OSTMnTE.OSTMnTE bit is cleared to indicate disabling of the counter.
The counter retains its current value until it is restarted.

(4) Counting by the counter restarts from H'0000_0000 when OSTMnTS.OSTMnTS = 1.

OSTMTINT period

The OSTMTINT generation period is different at the start of counting and depends on the old and new compare values if OSTMnCMP is rewritten during operation.

Table 20.5 OSTMTINT Generation Timing

Old Value for Comparison	New Value for Comparison	Counter Value at Time of Rewriting	Period of OSTMTINT Generation	Label in Timing Diagram
Counter starts			(A + 1) × counter clock period	(a)
A	A	No rewriting	(H'FFFF_FFFF + 1) × counter clock period	(b)
B	C > B	B < counter value < C	(C – B) × counter clock period	(c)
C	D < C	Counter value > D,C	(H'FFFF_FFFF – C + D + 1) × counter clock period	(d)

Forced restart

Forced restarting does not proceed during counting even if the OSTMnTS.OSTMnTS bit is set. The counter ignores the attempted setting and continues counting.

20.3.6.2 Operation when OSTMnCMP = H’0000_0000

The following figure shows the operation of OSTM when OSTMnCMP = H’0000_0000, and counter-start interrupts are enabled (OSTMnCTL.OSTMnMD0 = 1).

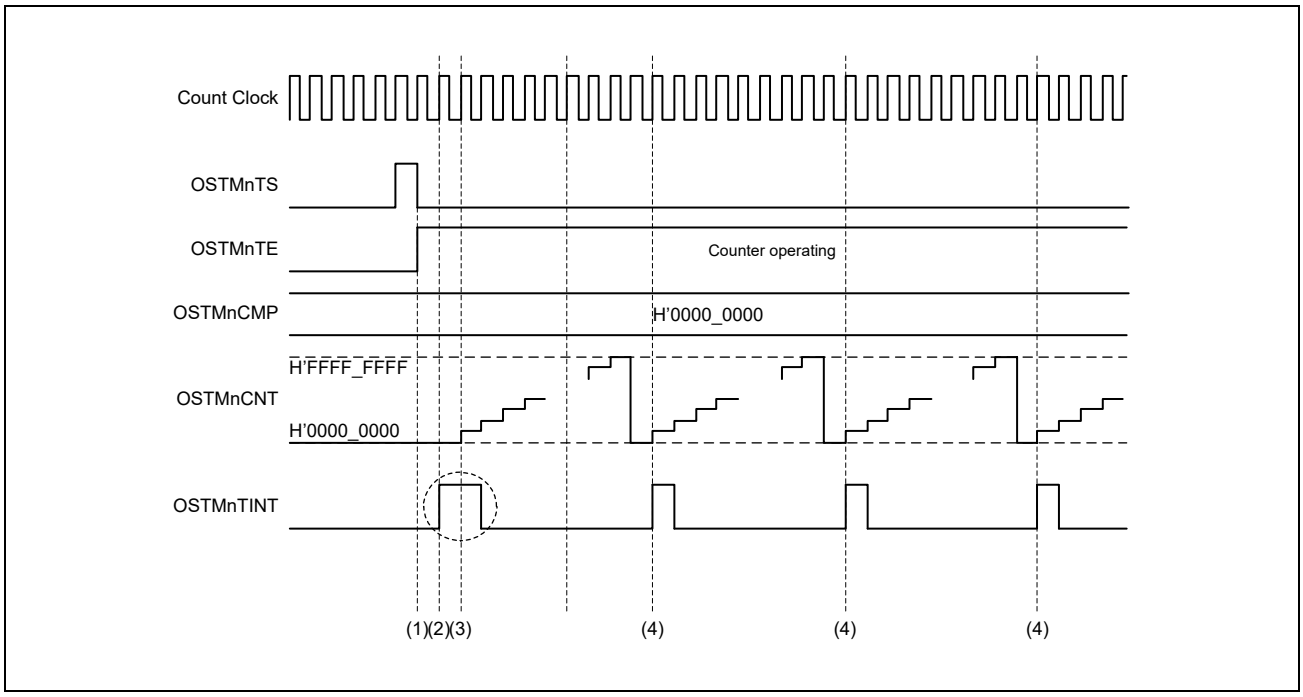


Figure 20.7 Timing Diagram when OSTMnCMP = H’0000_0000 in Free-Run Compare Mode

The timing diagram above shows the following operations.

(1) Once the counter starts, it counts up from H’0000_0000 to H’FFFF_FFFF.

(2) An OSTMTINT interrupt request is generated when counting starts.

(3) If the current counter value matches OSTMnCMP, an OSTMTINT interrupt request is generated. If OSTMnCMP = H’0000_0000 in the above case, OSTMTINT is generated over two clock cycles.

(4) Every (H’FFFF_FFFF + 1) clock cycles the OSTMTINT interrupt request is asserted.

When interrupts on starting of the counter are disabled, no interrupt is generated when counting starts.

21. Watchdog Timer (WDT)

This LSI has 3 channels of watchdog timer as **Table 21.1** and generates a reset request signal when the counter value is not rewritten and overflows due to system runaway.

Table 21.1 WDT Channels

Channel	Functions
WDT CH0	WDT to check the operation of Cortex-A55-CPU
WDT CH1	WDT to check the operation of Cortex-M33 CPU
WDT CH2	WDT to check the operation of Cortex-M33 CPU with FPU

Also, a reset request signal or interrupt signal can be generated by inputting a parity error signal due to a parity error in the CPU of this LSI.

21.1 Features

All WDT modules have the following functions implemented in common as basic WDT functions.

21.1.1 Normal Watchdog Timer Function

- Bus interface: Compatible with AMBA® 2.0 APB.
- Operating frequency: Bus interface 100 [MHz], counter control 24 [MHz]
- The counter operation and the bus interface are asynchronous and can operate without depending on the size relationship of the clock cycles of each other.
- A 32-bit counter that operates with the clock (WDTn_CLK (n = 0, 1, 2)) input to this module.
- An interrupt request signal is generated every cycle set in the counter of this module.
- When WDTn_CLK (n = 0, 1, 2) = 24 MHz, the counter cycle can be set in 43.69 msec unit from 43.69 msec to 178956.97 msec.
- After the interrupt request signal is generated, if the software does not clear the watchdog timer until the next counter overflow, the reset request signal is generated.
- The set value of each register and the Elapsed time of the watchdog timer can be read.

21.1.2 Reset Request Function due to CPU Parity Error

- A reset request signal (WDTRSTB) or interrupt signal (PERROUT) is generated by inputting a parity error signal.
- The reset request signal is generated as the OR of the reset request of this module and the reset request of the parity error circuit.
- The polarity of the parity error signal can be set with this module.
- Function can be turned ON/OFF for each 32-bit parity error signal.
- The output of the reset request signal and the interrupt signal can be switched by inputting the parity error signal.
- It is possible to force a parity error to occur by register setting.

21.1.3 Internal Block Diagram

Figure 21.1 shows the internal block of WDT. WDT consists of a WDT-CORE part that implements the function of a normal watchdog timer and a part that detects a parity error.

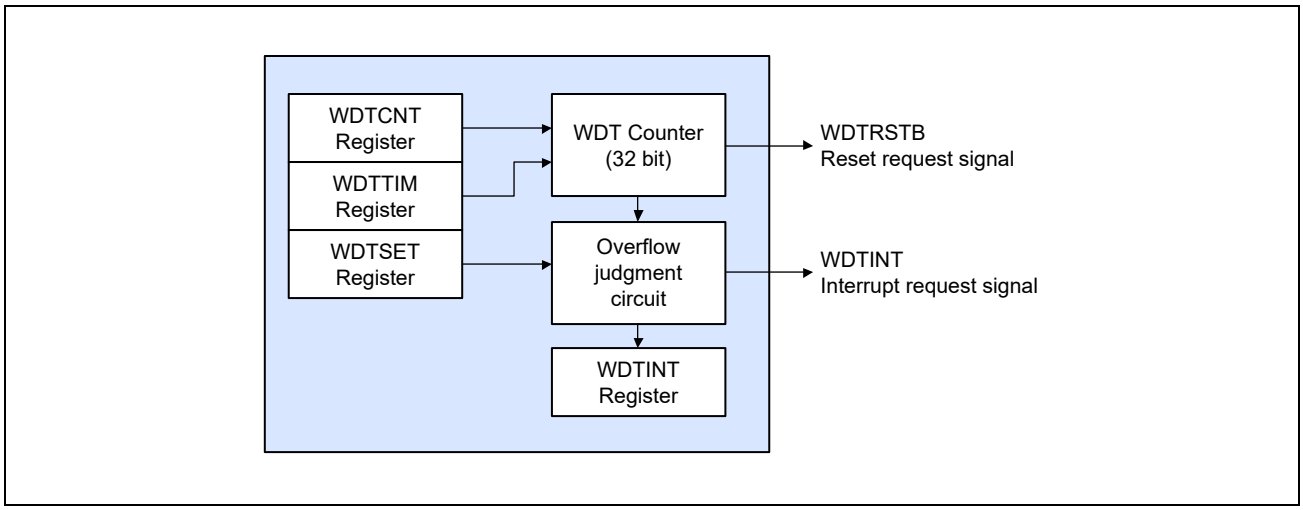


Figure 21.1 WDT_CORE Internal Block

In addition to the above WDT-Core function, WDT receives a parity error interrupt signal generated by the CPU when a Parity Error occurs in the CPU monitored by WDT and generates a reset request signal or interrupt request signal.

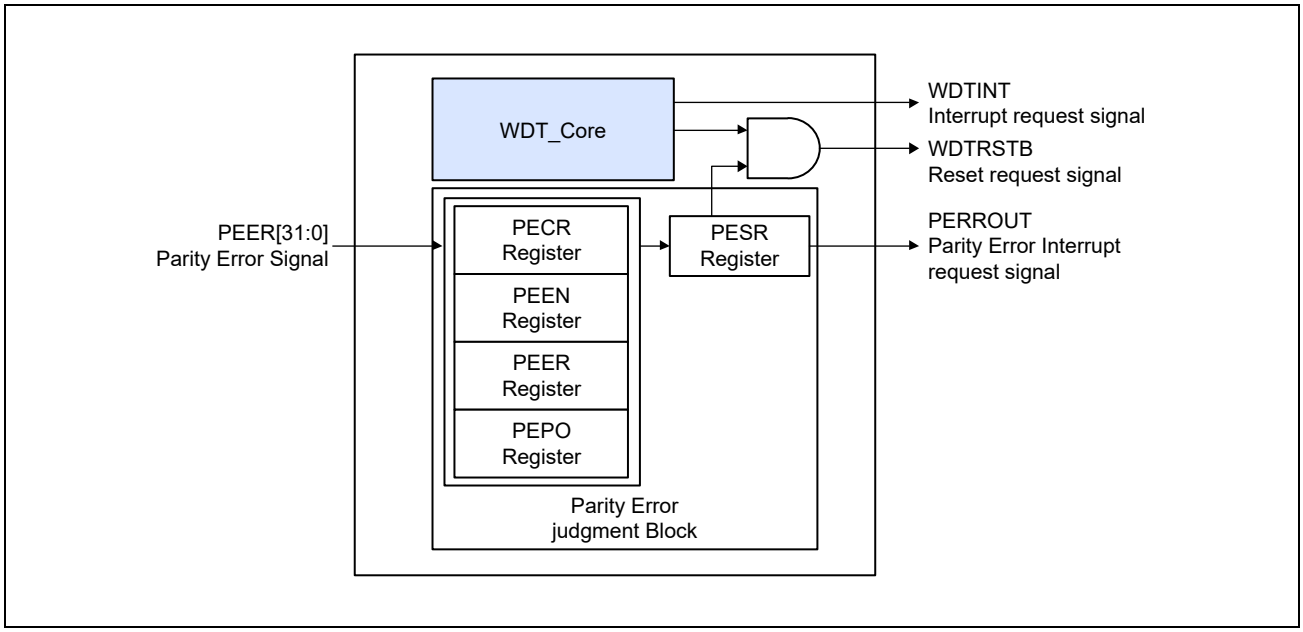


Figure 21.2 WDT_CORE and Parity Error Handling Block

21.1.4 Input/Output Pin

Table 21.2 shows the pin configuration.

Table 21.2 Pin Configuration

Item	Pin Name	I/O	Function
Watchdog timer overflow/ CPU parity error	WDTOVF_PERROUT#	Output	Watchdog timer counter overflow signal output/ Interrupt signal output when CPU parity error occurs

21.2 Register Configuration

Table 21.3 shows the base address for each WDT channel.

Table 21.3 Base Address for Each WDT Channel

	Channel	WDT	Base Address
Cortex-A55 Address Space	WDT CH0	WDT to check the operation of Cortex-A55-CPU	H'0_1280_0800
	WDT CH1	WDT to check the operation of Cortex-M33 CPU	H'0_1280_0400
	WDT CH2	WDT to check the operation of Cortex-M33 CPU with FPU	H'0_1280_0000
Cortex-M33 Address Space Secure	WDT CH0	WDT to check the operation of Cortex-A55-CPU	H'4280_0800
	WDT CH1	WDT to check the operation of Cortex-M33 CPU	H'4280_0400
	WDT CH2	WDT to check the operation of Cortex-M33 CPU with FPU	H'4280_0000
Cortex-M33 Address Space Non-Secure	WDT CH0	WDT to check the operation of Cortex-A55-CPU	H'5280_0800
	WDT CH1	WDT to check the operation of Cortex-M33 CPU	H'5280_0400
	WDT CH2	WDT to check the operation of Cortex-M33 CPU with FPU	H'5280_0000

Note: Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 21.4 shows WDT Register Configuration on all WDT Channels.

Table 21.4 WDT Register Configuration

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
WDT CH0	WDT Control Register_0	WDTCNT_0	R/W	H'0000_0000	H'0000	32
	WDT Period Setting Register_0	WDTSET_0	R/W	H'FFF0_0000	H'0004	32
	WDT Elapsed Time Register_0	WDTTIM_0	R/W	H'0000_0000	H'0008	32
	WDT Interrupt control Register_0	WDTINT_0	R/W	H'0000_0000	H'000C	32
	Parity Error Control Register_0	PECR_0	R/W	H'0000_0000	H'0010	32
	Parity Error forced Enable Register_0	PEEN_0	R/W	H'0000_0000	H'0014	32
	Party Status Register_0	PESR_0	R/W	H'0000_0000	H'0018	32
	Parity Error Enable Register_0	PEER_0	R/W	H'0000_0000	H'001C	32
	Parity Error Polarity setting Register_0	PEPO_0	R/W	H'0000_0000	H'0020	32
WDT CH1	WDT Control Register_1	WDTCNT_1	R/W	H'0000_0000	H'0000	32
	WDT Period Setting Register_1	WDTSET_1	R/W	H'FFF0_0000	H'0004	32
	WDT Elapsed Time Register_1	WDTTIM_1	R/W	H'0000_0000	H'0008	32
	WDT Interrupt control Register_1	WDTINT_1	R/W	H'0000_0000	H'000C	32
	Parity Error Control Register_1	PECR_1	R/W	H'0000_0000	H'0010	32
	Parity Error forced Enable Register_1	PEEN_1	R/W	H'0000_0000	H'0014	32
	Party Status Register_1	PESR_1	R/W	H'0000_0000	H'0018	32
	Parity Error Enable Register_1	PEER_1	R/W	H'0000_0000	H'001C	32
	Parity Error Polarity setting Register_1	PEPO_1	R/W	H'0000_0000	H'0020	32
WDT CH2	WDT Control Register_2	WDTCNT_2	R/W	H'0000_0000	H'0000	32
	WDT Period Setting Register_2	WDTSET_2	R/W	H'FFF0_0000	H'0004	32
	WDT Elapsed Time Register_2	WDTTIM_2	R/W	H'0000_0000	H'0008	32
	WDT Interrupt control Register_2	WDTINT_2	R/W	H'0000_0000	H'000C	32
	Parity Error Control Register_2	PECR_2	R/W	H'0000_0000	H'0010	32
	Parity Error forced Enable Register_2	PEEN_2	R/W	H'0000_0000	H'0014	32
	Party Status Register_2	PESR_2	R/W	H'0000_0000	H'0018	32
	Parity Error Enable Register_2	PEER_2	R/W	H'0000_0000	H'001C	32
	Parity Error Polarity setting Register_2	PEPO_2	R/W	H'0000_0000	H'0020	32

21.3 Register Descriptions

21.3.1 WDT Control Register_n (WDTCNT_n) (n = 0, 1, 2)

This register sets the operation enable of the watchdog timer function. Once the software activates the watchdog timer, the watchdog timer does not stop*1 until it is reset. Also, the setting register*2 of the watchdog timer cannot be changed. The setting register change must be completed*2 before enabling the watchdog timer operation.

Note 1. For debugging, the internal counter can be paused with the CNTSTOP signal.

Note 2. WDT cycle setting register (WDTSET) and WDT Elapsed time register (WDTTIM)

Note 3. When changing the setting register, wait until the write data value of the setting register is reflected before enabling the watchdog timer operation. The reflection time of the setting register is 6 × P0ϕ + 9 × OSCCLK or more. If this reflection time is not observed, the setting register value will not be reflected and will be ignored.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
0	WDTEN	0	R/W	Watchdog timer enable register 1: Watchdog timer operation enabled 0: Disabled

21.3.2 WDT Period Setting Register_n (WDTSET_n) (n = 0, 1, 2)

This register is used to set the cycle of the 32-bit counter that composes the watchdog timer. Writing is not possible while the watchdog timer is running. If you write continuously, the data written later will be valid. Calculate the watchdog timer cycle using the following formula.

$$\text{Watchdog timer cycle} = \text{WDTn\_CLK (n} = 0, 1, 2\text{) cycle} \times 1024 \times 1024 \times (\text{WDTTIME setting value} + 1)$$

$$\text{WDTTIME setting value} = \frac{\text{WDT cycle}}{(\text{WDTn\_CLK (n} = 0,1,2\text{) cycle} \times 1024 \times 1024) - 1}$$

For example, WDTn_CLK (n = 0, 1, 2) = 24 MHz, the counter cycle can be set in 43.69 msec unit from 43.69 msec to 178956.97 msec.

It can be set in units. The setting value of WDTTIME is the value set in WDTSET[31:20], and the value from H'000 to H'FFF can be set.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WDTTIME11	WDTTIME10	WDTTIME9	WDTTIME8	WDTTIME7	WDTTIME6	WDTTIME5	WDTTIME4	WDTTIME3	WDTTIME2	WDTTIME1	WDTTIME0	—	—	—	—
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	WDTTIME [11:0]	All 1	R/W	Watchdog timer period cycle setting It cannot be written while the watchdog timer is running.
19 to 0	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.

CAUTION

The value that the watchdog timer compares when an interrupt occurs is "WDTSET[31:0] + H'000F_FFFF". Therefore, when the setting value of this register is H'0010_0000, the interrupt occurrence count is "H'001F_FFFF" (not H'0010_0000).

21.3.3 WDT Elapsed Time Register_n (WDTTIM_n (n = 0, 1, 2))

This register indicates the 32-bit count value that composes the watchdog timer. If the watchdog timer is read during operation, the Elapsed time from the time it is cleared to the time of reading can be read.

By writing to this register when the watchdog timer is stopped, the value at the start of counting can be set in the 32-bit counter that configures the watchdog timer. When a value larger than the counter cycle (value when clearing) set in the WDT cycle setting register is written and the watchdog timer operation is started, the 32-bit counter once overflows and returns to 0. It operates at the set correct cycle.

You cannot write while the watchdog timer is running. If you write continuously, the data written later will be valid.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRTTIME31	CRTTIME30	CRTTIME29	CRTTIME28	CRTTIME27	CRTTIME26	CRTTIME25	CRTTIME24	CRTTIME23	CRTTIME22	CRTTIME21	CRTTIME20	CRTTIME19	CRTTIME18	CRTTIME17	CRTTIME16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRTTIME15	CRTTIME14	CRTTIME13	CRTTIME12	CRTTIME11	CRTTIME10	CRTTIME9	CRTTIME8	CRTTIME7	CRTTIME6	CRTTIME5	CRTTIME4	CRTTIME3	CRTTIME2	CRTTIME1	CRTTIME0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRTTIME [31:0]	All 0	R/W	Watchdog timer count value. It cannot be written while the watchdog timer is running.

CAUTION

When setting a value in the WDTTIM register, do not set a value that is WDTSET[31:0] register + H'000F_FFFF.

21.3.4 WDT Interrupt Control Register_n (WDTINT_n (n = 0, 1, 2))

This register reads the interrupt status of the watchdog timer, clears interrupts, and clears the counters that compose the watchdog timer. Set the INTDISP bit of this register to “1” within the time set by WDTSET. If not set, the WDTINT pin is asserted. After that, within the time set by WDTSET

When the INTDISP bit is not set to “1”, the WDTRSTB pin is asserted.

When performing continuous write access to the WDTINT register, leave a write interval of 5 × P0ϕ + 5 × OSCCLK or more. If the write interval is not observed, the register will not be written.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INTDISP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
0	INTDISP	0	R/W	Register for interrupt status read and clear Read: Interrupt status 1: With interrupt 0: Without interrupt Write: Interrupt clear 1: Interrupt clear 0: Invalid (no change)

21.3.5 Parity Error Control Register_n (PECR_n (n = 0, 1, 2))

This register controls whether a reset (WDTRSTB) or an interrupt (PERROUT) is generated due to a CPU parity error.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PECR31	PECR30	PECR29	PECR28	PECR27	PECR26	PECR25	PECR24	PECR23	PECR22	PECR21	PECR20	PECR19	PECR18	PECR17	PECR16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PECR15	PECR14	PECR13	PECR12	PECR11	PECR10	PECR9	PECR8	PECR7	PECR6	PECR5	PECR4	PECR3	PECR2	PECR1	PECR0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PECR [31:0]	All 0	R/W	Parity error control Controls the following operations when the value is 1 for each bit of PESR[31:0] 0: Reset (WDTRSTB) is generated 1: Generate interrupt (PEEROUT)

21.3.6 Parity Error Forced Enable Register_n (PEEN_n (n = 0, 1, 2))

This register forcibly asserts reset (WDTRSTB) or PERROUT even when no CPU parity error has occurred.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PEEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
0	PEEN	0	R/W	Parity error forced enable 0: Reset (WDTRSTB) or PERROUT generation follows the operation of the parity error register 1: Force reset (WDTRSTB) or assert PERROUT

21.3.7 Parity Error Status Register_n (PESR_n (n = 0, 1, 2))

This register holds the cause of CPU parity error.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PESR31	PESR30	PESR29	PESR28	PESR27	PESR26	PESR25	PESR24	PESR23	PESR22	PESR21	PESR20	PESR19	PESR18	PESR17	PESR16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PESR15	PESR14	PESR13	PESR12	PESR11	PESR10	PESR9	PESR8	PESR7	PESR6	PESR5	PESR4	PESR3	PESR2	PESR1	PESR0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PESR[31:0]	All 0	R/W	Parity error flag 0 to 31 Indicates that a parity error has occurred. Read: Interrupt status 1: With interrupt 0: Without interrupt Write: Interrupt clear 1: Interrupt clear 0: Invalid (nothing changed)

21.3.8 Parity Error Enable Register_n (PEER_n (n = 0, 1, 2))

This register controls the assertion of the WDTRSTB signal and PERROUT signal due to a CPU parity error. Each bit controls the behavior of parity error.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PEER31	PEER30	PEER29	PEER28	PEER27	PEER26	PEER25	PEER24	PEER23	PEER22	PEER21	PEER20	PEER19	PEER18	PEER17	PEER16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PEER15	PEER14	PEER13	PEER12	PEER11	PEER10	PEER9	PEER8	PEER7	PEER6	PEER5	PEER4	PEER3	PEER2	PEER1	PEER0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PEER[31:0]	All 0	R/W	Parity error enable 31-0 Controls the operation of PESR. 0: Disables reset (WDTRSTB) and PERROUT generation operations 1: Permits reset (WDTRSTB) and PERROUT generation operation

21.3.9 Parity Error Polarity Setting Register_n (PEPO_n (n = 0, 1, 2))

This register sets the polarity of CPU parity error.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PEPO31	PEPO30	PEPO29	PEPO28	PEPO27	PEPO26	PEPO25	PEPO24	PEPO23	PEPO22	PEPO21	PEPO20	PEPO19	PEPO18	PEPO17	PEPO16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PEPO15	PEPO14	PEPO13	PEPO12	PEPO11	PEPO10	PEPO9	PEPO8	PEPO7	PEPO6	PEPO5	PEPO4	PEPO3	PEPO2	PEPO1	PEPO0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PEPO [31:0]	All 0	R/W	Active polarity of parity 31-0 Set the Active polarity (H/L) of each parity error signal connected to PEERR. 0: Active High 1: Active Low

21.3.10 Register Setting Order for Each WDT Channel

The following shows the register setting sequence for controlling the watchdog timer and parity error circuit of each channel. The control of the original watchdog timer and the parity error circuit do not affect each other, so there is no restriction on the setting order in the following diagrams (1) and (2).

The parity error circuit cannot accept the parity error that occurred before setting PEER = 1. When rewriting PEEN from 0 to 1, it is possible to rewrite at any timing. However, when rewriting PEEN from 1 to 0, once PEER = 0 and (2) flow execution is required.

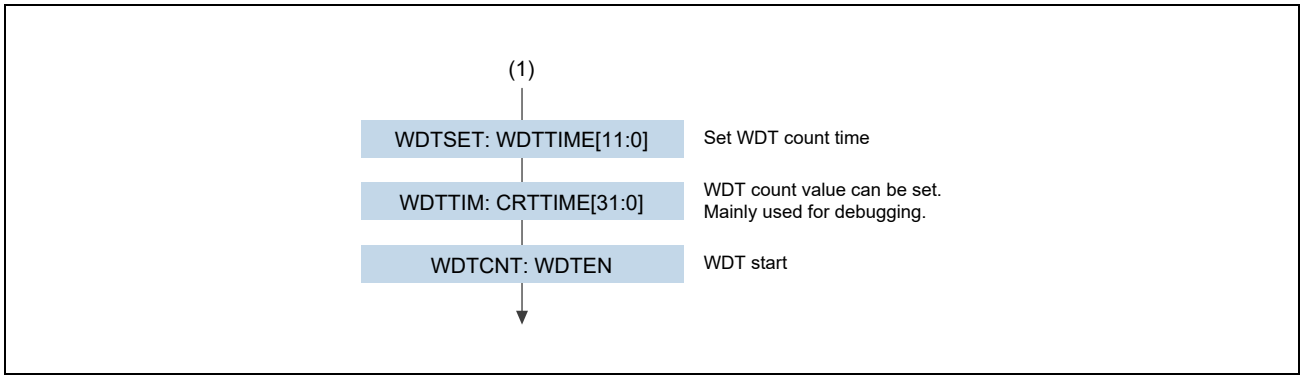


Figure 21.3 WDT Register Setting Order

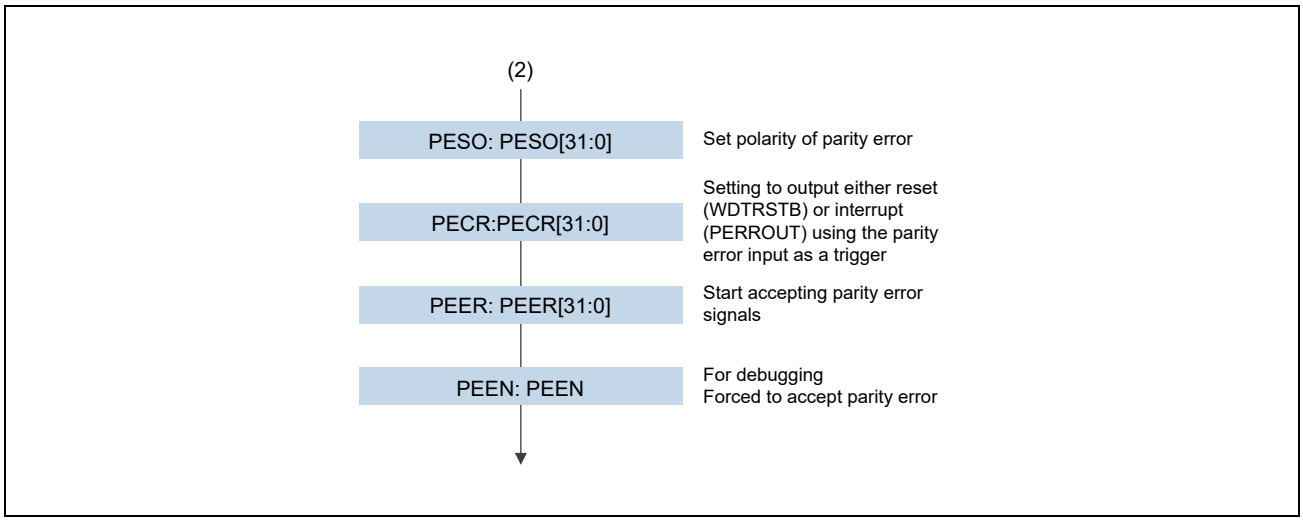


Figure 21.4 Parity Error Circuit Register Setting Order

21.4 Operation

21.4.1 WDT_CORE Operaton Timing

Figure 21.5 shows the operation diagram of the WDT-Core block. When 1 is set in Bit [0] of the WDTCNT register, the 32-bit internal counter WDT-Counter starts counting. If the counter value matches the value set in the WDTSET register, the counter becomes overflow, clears the counter to 0, and then generates an interrupt request signal.

An interrupt signal is output at the first counter overflow, and a reset request signal is output if the counter is not cleared and the second counter overflow occurs.

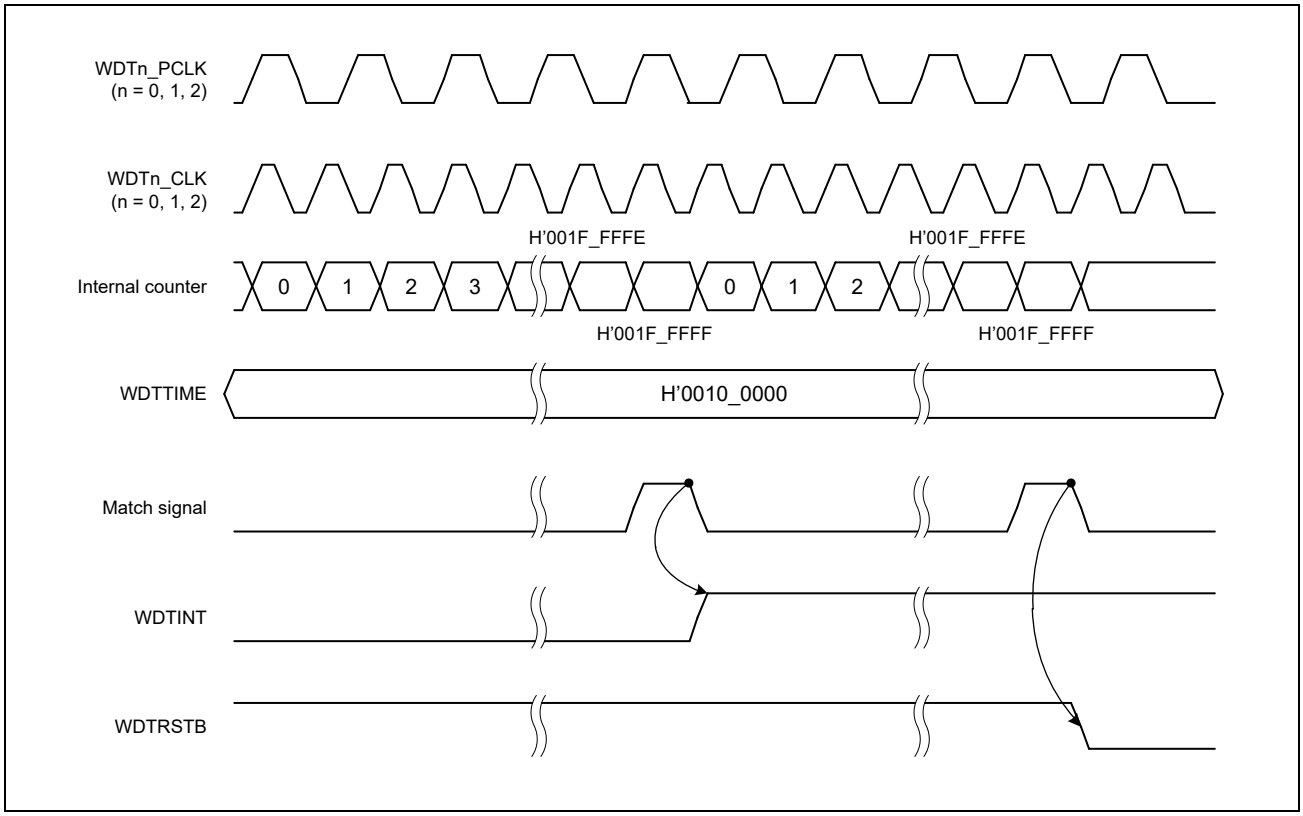


Figure 21.5 WDT Output Signal Operation Timing

21.4.2 WDT Interrupt Control Register (WDTINT) Specification for Write Access Interval

After the write access to the WDT interrupt control register is completed, the written data is synchronized with WDTn_CLK (n = 0, 1, 2), so a synchronization period is required. When performing write access to the WDTINT register, it is necessary to leave a write interval of 5 × P0ϕ + 5 × OSCCLK or more between the next write access.*1

Note 1. If the write interval is not observed, the interrupt request may not be cleared.
In this case, after making a write access and confirming that the interrupt factor has been cleared, perform the following write access.

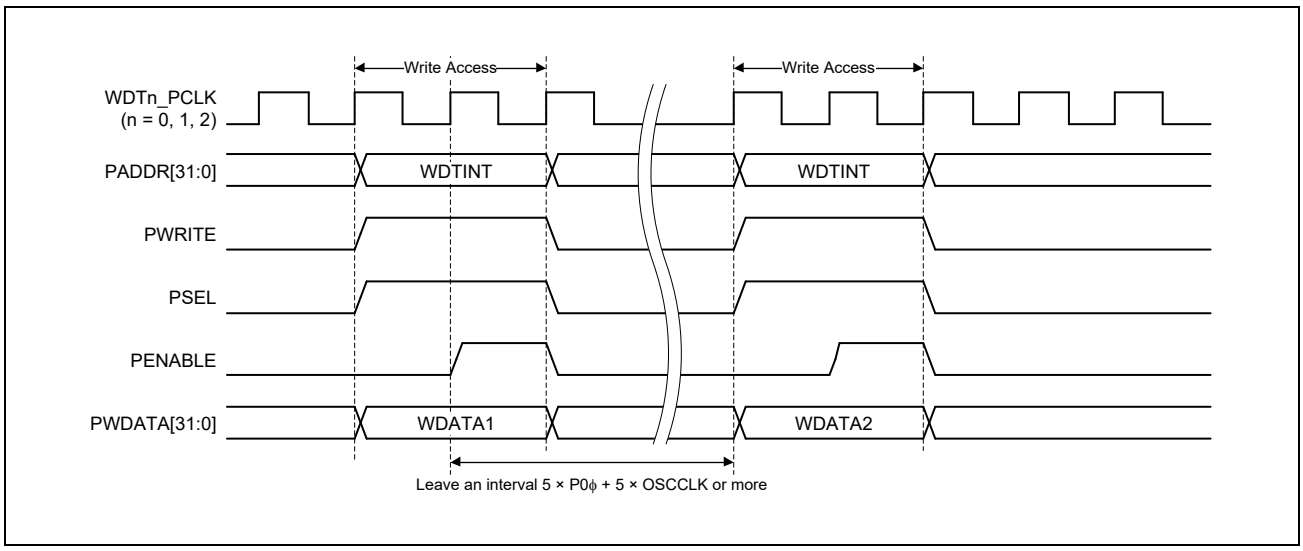


Figure 21.6 WDTINT Write Access Interval Regulation

21.4.3 Watchdog Timer Operation Reflection Timing of Register Setting Value

Each register other than the WDT interrupt control register (WDTINT) is always synchronized with WDTn_CLK (n = 0, 1, 2) at regular intervals. The timing at which the written data is reflected in the watchdog timer operation is in the range of small 2 × P0ϕ + 4 × OSCCLK) to large value (6 × P0ϕ +9 × OSCCLK).

Figure 21.7 (Watchdog timer operation reflection timing 1 of register setting value) shows how write data is small and synchronized.

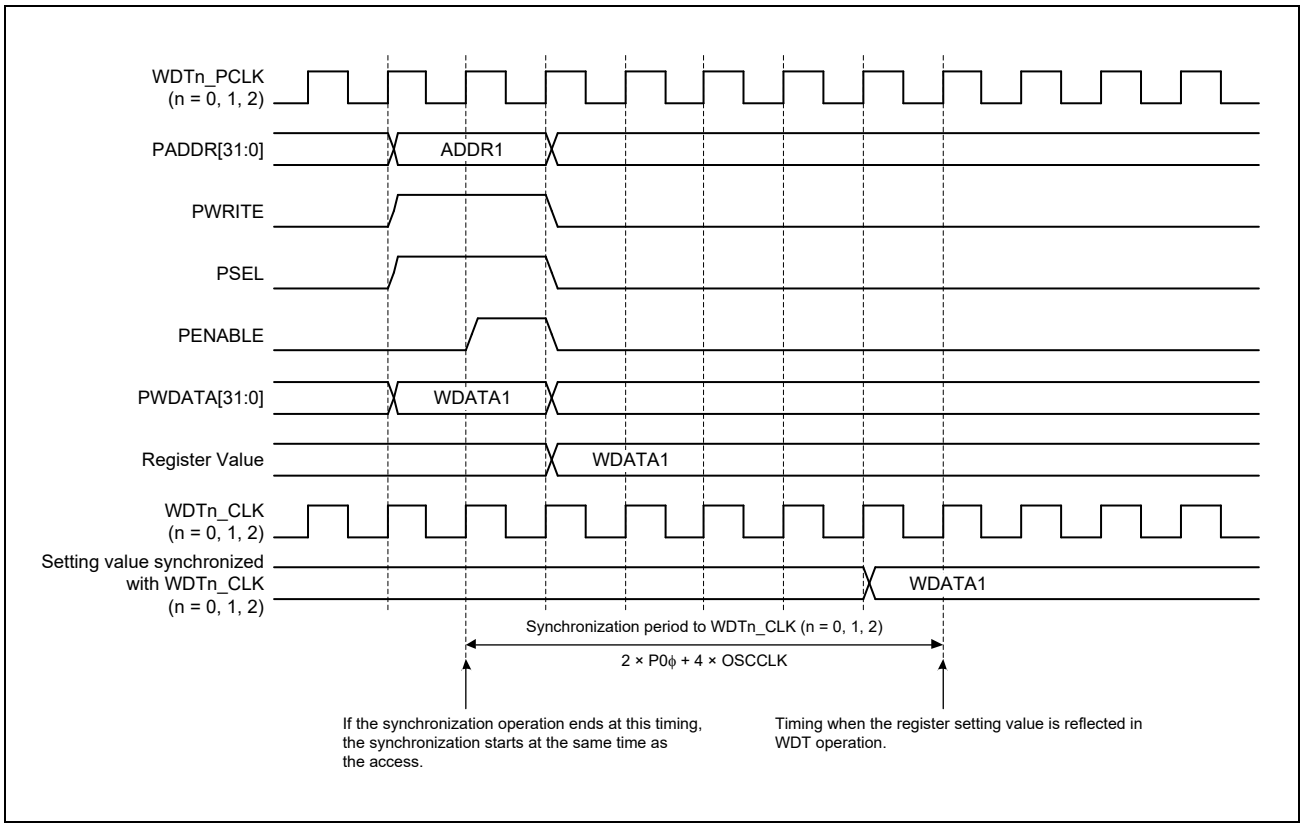


Figure 21.7 Watchdog Timer Operation Reflection Timing of Register Setting Value - 1

In addition, **Figure 21.8** (Watchdog timer operation reflection timing 2 of register setting value) shows how write data is large and synchronized.

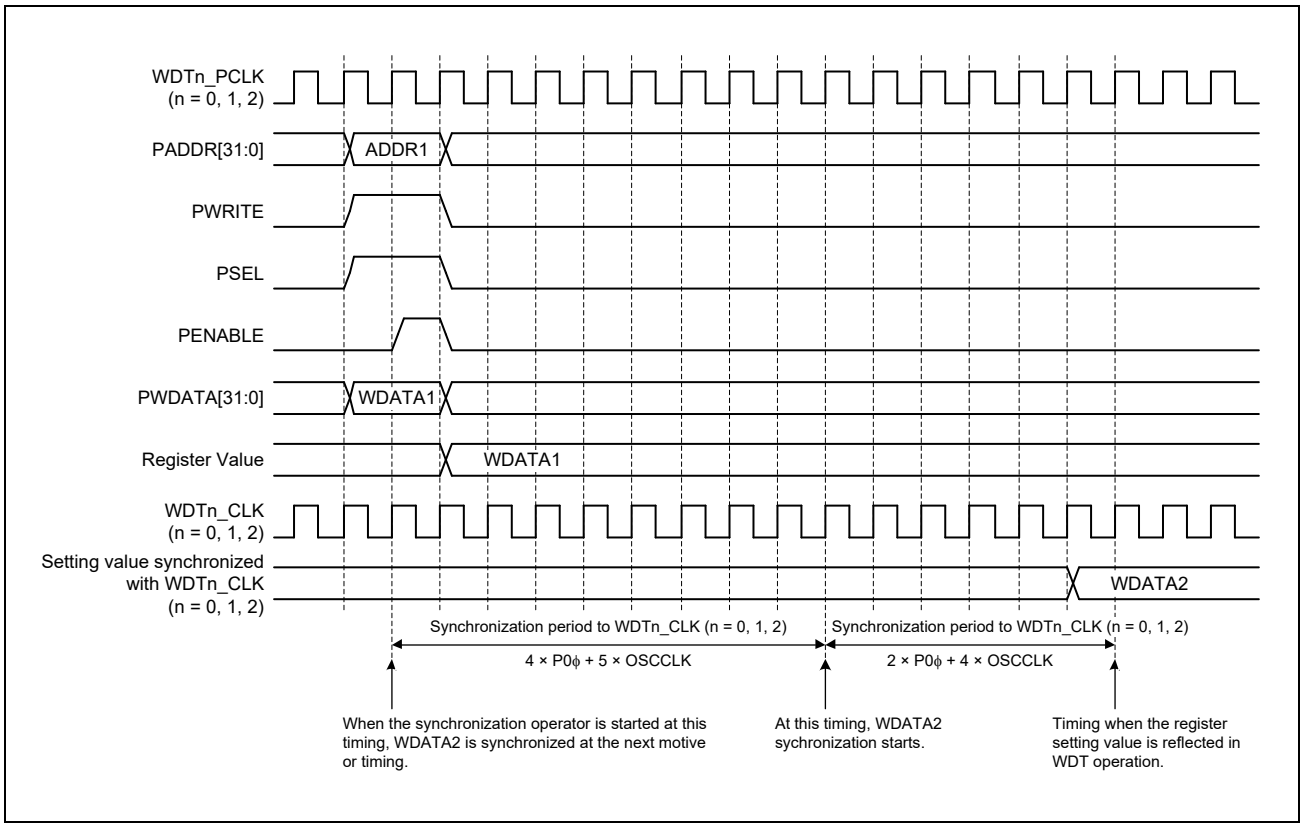


Figure 21.8 Watchdog Timer Operation Reflection Timing of Register Setting Value - 2

21.4.4 Timing of Reflecting WDT Counter Value to WDTTIM Register

The value stored in the WDTTIM register is the value obtained by synchronizing the counter value with P0φ. Therefore, the interval at which the value is updated in the WDTTIM register is 5 × OSCCLK+5 × P0ϕ at most, and the value before the actual counter value is read.

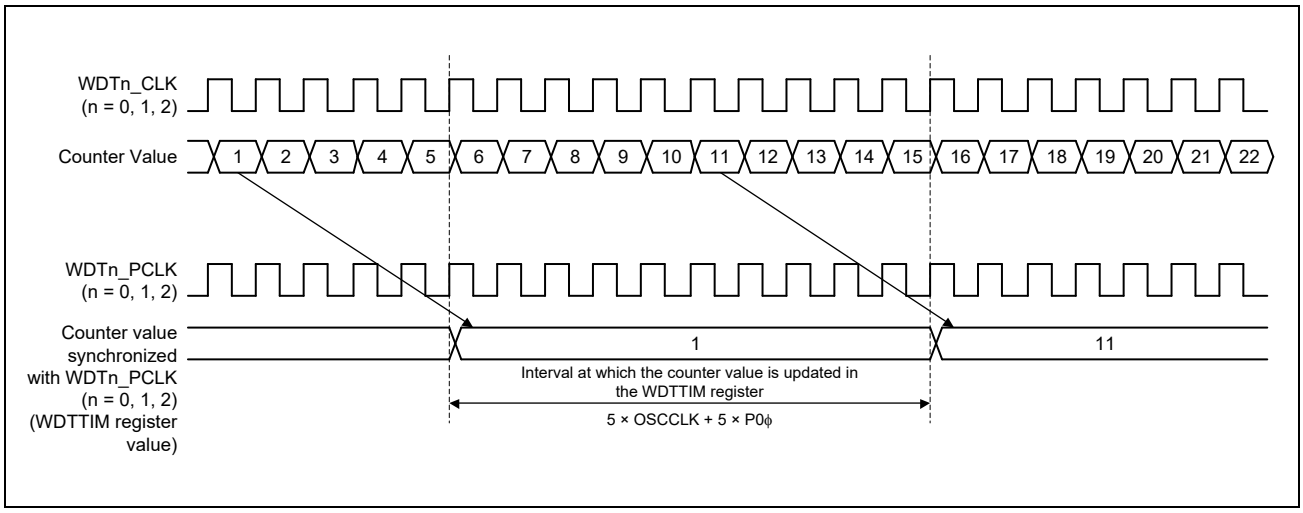


Figure 21.9 Timing of Reflecting the Counter Value to the WDTTIM Register

21.4.5 Conditions Under which WDTRSTB and PERROUT Operate

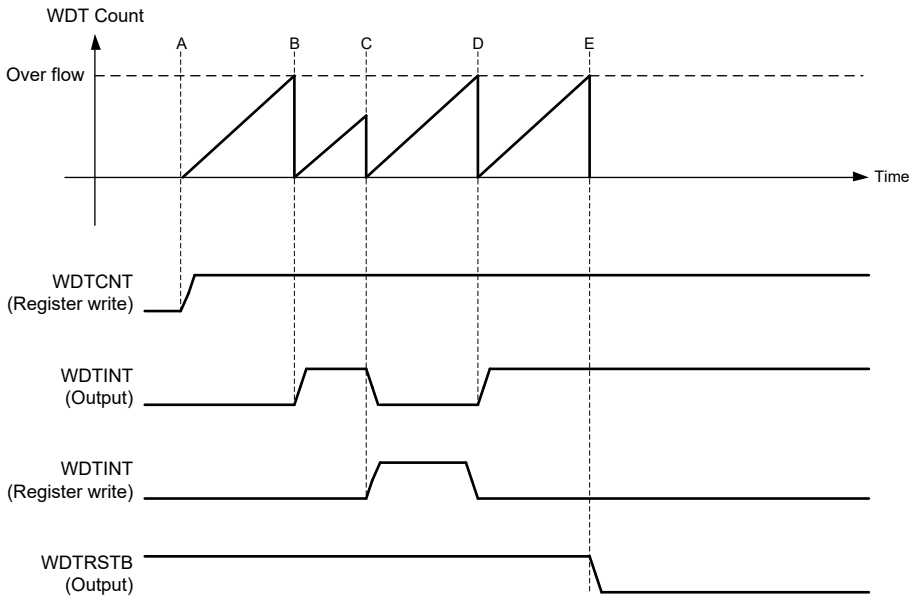
Table 21.5 shows the conditions under which WDTRSTB and PERROUT operate.

When PEER = 0, WDTRSTB follows the watchdog timer operation described in **Section 21.4.1** to **Section 21.4.5** regardless of the occurrence of parity error. PERROUT holds 0. (No. 1, No. 2) No. 3 to No. 6 show the operating conditions when the parity error input is enabled by PEER = 1. When a parity error occurs (No. 3, No. 4), PESR holds the error status, and WDTRSTB or PERROUT is asserted depending on the value of PECR. If no parity error occurs, PESR = 0, WDTRSTB propagates the WDT-Core logic, and PERROUT is in the unasserted state. (No. 5, No. 6) When PEEN = 1 forces a parity error to occur, WDTRSTB or PERROUT is asserted depending on the PECR value regardless of the parity error.

Table 21.5 WDTRSTB/PERROUT Operating Conditions

No.	PEERR	PEPO	PEER	PEEN	PECR	PESR	WDTRSTB	PERROUT
1	X	1/0	0	0	0	0 (Fixed)	Original WDT	0: de-assert
2		1/0	0	0	1	0 (Fixed)	Original WDT	0: de-assert
3	With parity	1/0	1	0	0	1	0: assert	0: de-assert
4		1/0	1	0	1	1	Original WDT	1: assert
5	No parity	1/0	1	0	0	0	Original WDT	0: de-assert
6		1/0	1	0	1	0	Original WDT	0: de-assert
7	X	X	X	1	0	X	0: assert	0: de-assert
8		X	X	1	1	X	Original WDT	1: assert

The timing waveforms for No. 1 to No. 4 in **Table 21.5** are shown below.



Note: A) Set WDTCNT = 1 to start WDT counting.

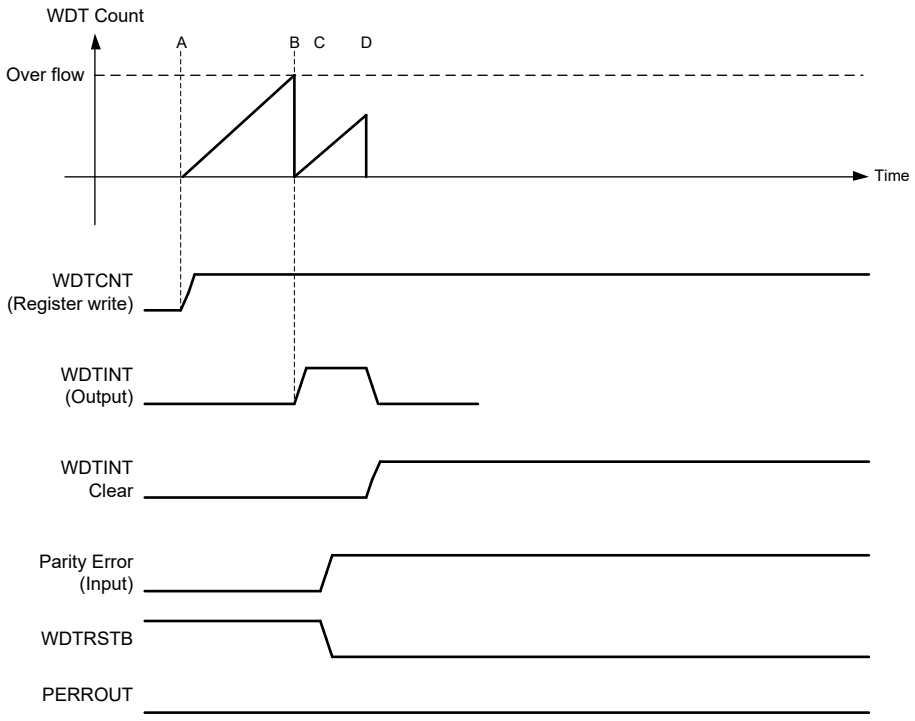
B) The interrupt request signal (WDTINT) is asserted at the first overflow of the WDT count.

C) If WDTINT (INTDISP) = 1 is written before the WDT count overflows, the interrupt request is cleared and the WDT count becomes zero.

D) Again, the interrupt request signal is asserted at the first overflow.

E) When the WDT count overflows for the second time, WDTRSTB is asserted.

Figure 21.10 Normal Watchdog Timer Operation No. 1/No. 2



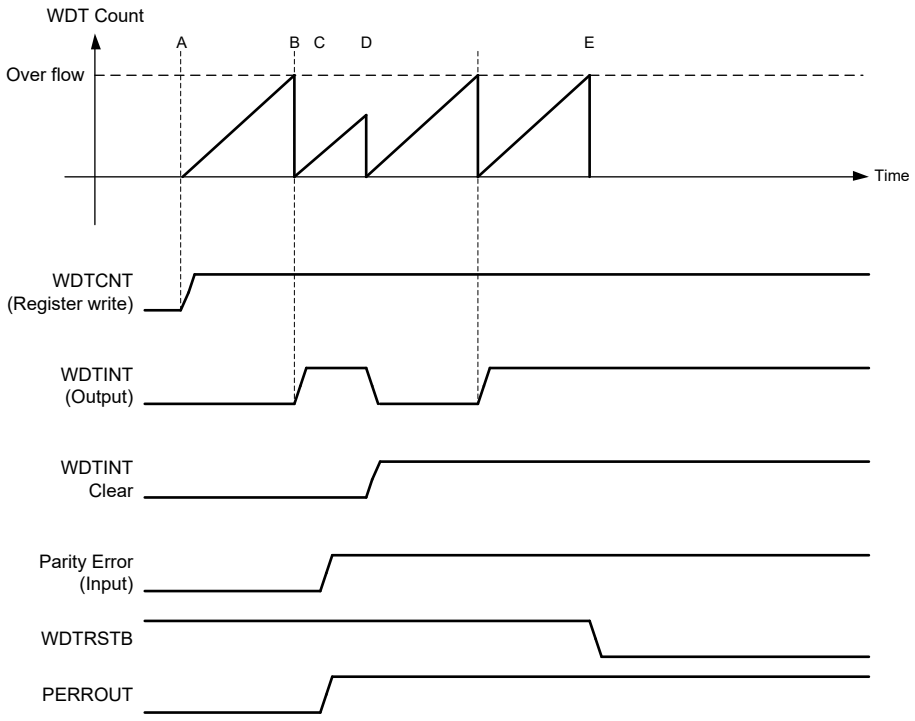
Note: A) Set WDTCNT = 1 to start WDT counting.

B) The interrupt request signal (WDTINT) is asserted at the first overflow of the WDT count.

C) When a parity error occurs, WDTRSTB = 0. At the same time, hold PERROUT = 0.

D) Even if WDT Count is reset by WDTINT = 1, WDTRSTB = 0 and PERROUT = 0 are retained.

Figure 21.11 Watchdog Timer Operation when a Parity Error Signal Occurs No. 3



Note: A) Set WDTCNT = 1 to start WDT counting.

B) The interrupt request signal (WDTINT) is asserted at the first overflow of the WDT count.

C) When a parity error occurs, WDTRSTB = 1. At the same time, hold PERROUT = 1.

D) Even if WDT Count is reset by WDTINT = 1, WDTRSTB = 1 and PERROUT = 1 are retained.

E) WDTRSTB = 0 is asserted due to WDT Count overflow.

Figure 21.12 Watchdog Timer Operation when a Parity Error Signal Occurs No. 4

21.5 Reset Control

21.5.1 WDTRST Control of Each WDT Channels

WDT CH0 generates WDTINT every cycle set in WDTSET. The CPU (Cortex-A55) receives the WDT CH0 interrupt signal WDTINT, clears the WDT CH0 counter at regular intervals, and suppresses the overflow of the WDT counter.

WDT CH1 generates WDTINT every cycle set in WDTSET. The CPU (Cortex-M33) receives the WDT CH1 interrupt signal WDTINT and clears the WDT CH1 counter at regular intervals to suppress the overflow of the WDT counter.

WDT CH2 generates WDTINT every cycle set in WDTSET. The CPU (Cortex-M33_FPU) receives the WDT CH2 interrupt signal WDTINT and clears the WDT CH2 counter at regular intervals to prevent the WDT counter from overflowing.

If the counter is not cleared before the cycle set for each WDT elapses, the uncleared WDT sends a reset request signal to the CPG. Based on the setting of the built-in register CPG_WDTRST_SEL, CPG can select whether to reset only the CPU corresponding to the WDT that caused the reset from the WDT reset request signal from each channel, or to reset the entire system. is. You can also choose whether to assert the WDTOVF_PERROUT# pin to notify the outside world that the entire system has been reset.

(For details about CPG_WDTRST_SEL register in CPG, refer to the CPG chapter.)

Table 21.6 WDT Reset Target by CPG_WDTRST_SEL Register

Bit	Initial Value	R/W	Reset Factor	Function
CPG_WDTRST_SEL[0]	0	R/W	WDT CH0 (CA55)	When the target WDT executes a reset request due to overflow etc. 0: Mask system reset assert. 1: System reset assert.
CPG_WDTRST_SEL[1]	0	R/W	WDT CH1 (CM33)	
CPG_WDTRST_SEL[2]	0	R/W	WDT CH2 (CM33_FPU)	
CPG_WDTRST_SEL[4]	0	R/W	WDT CH0 (CA55)	When the target WDT executes a reset request due to overflow etc. 0: Mask WDTOVF_PERROUT assert. 1: WDTOVF_PERROUT assert.
CPG_WDTRST_SEL[5]	0	R/W	WDT CH1 (CM33)	
CPG_WDTRST_SEL[6]	0	R/W	WDT CH2 (CM33_FPU)	
CPG_WDTRST_SEL[8]	0	R/W	WDT CH0 (CA55)	When WDT CH0 makes a reset request 0: Mask the Cold-Reset of Cortex-A55 1: Implemented Cortex-A55 Cold-Reset
CPG_WDTRST_SEL[9]	0	R/W	WDT CH1 (CM33)	When WDT CH1 makes a reset request 0: Mask the Cold-Reset of Cortex-M33 1: Implemented Cortex-M33 Cold-Reset
CPG_WDTRST_SEL[10]	0	R/W	WDT CH2 (CM33_FPU)	When WDT CH2 makes a reset request 0: Mask the Cold-Reset of Cortex-M33_FPU 1: Implemented Cortex-M33_FPU Cold-Reset

When the WDT system reset is executed, it is also possible to set the CPG_WDTRST_SEL[6:4] bit to notify the system outside the LSI from the WDTOVF_PERROUT# pin of the GPIO pin that the system reset has occurred due to the WDT overflow. When using WDTOVF_PERROUT# as a signal to notify the system reset to the outside, set the CPG_WDTRST_SEL[6:4] bit to 1 when CPG_WDTRST_SEL[2:0] is 1 as shown below.

CPG_WDTRST_SEL[4] Set 1 when CPG_WDTRST_SEL[0] = 1
GPG_WDTRST_SEL[5] Set 1 when CPG_WDTRST_SEL[1] = 1
CPG_WDTRST_SEL[6] Set 1 when CPG_WDTRST_SEL[2] = 1

For example, if CPG_WDTRST_SEL[0] = 0 and WDTRST_SEL[4] = 1 are set, the system reset will not be executed even if WDT CH0 asserts a reset request because CPG_WDTRST_SEL[0] = 0, but WDTRST_SEL[4] = 1 so

WDTOVF_PERROUT# pin will be asserted. In other words, the system reset execution status and WDTOVF_PERROUT do not match.

21.6 Usage Note

[Cautions]

- When setting a value in the WDTTIM register, do not set a value that is WDTSET register + H'F_FFFF.
- The reset request signal WDTRSTB and WDT interrupt signal WDTINT cannot be masked.

22. Realtime Clock (RTCA-3)

22.1 Overview

The RTC has two types of counting modes: calendar count mode and binary count mode. They are used by switching the register settings.

For calendar count mode, the RTC has a 100-year calendar from 2000 to 2099 and automatically adjusts dates for leap years.

For binary count mode, the RTC does not count in terms of years, months, dates, day-of-week, hours, or minutes; it counts seconds, and retains the information as a serial value. This mode can be used for calendars other than the Gregorian calendar.

22.1.1 Features

Table 22.1 lists the specifications of the RTC.

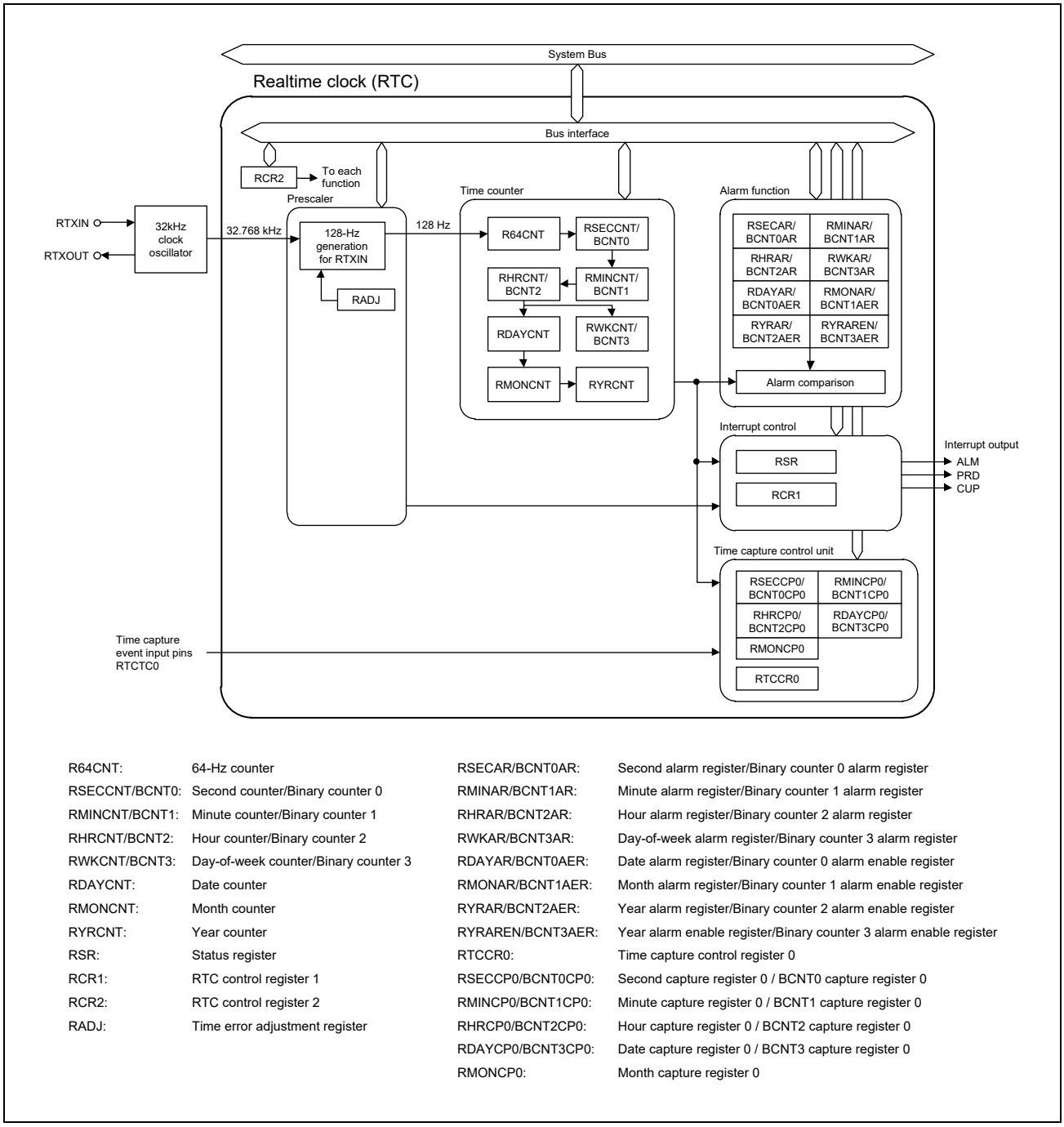
Table 22.1 RTC Specifications

Item	Description
Count mode	Calendar count mode/binary count mode
Count source*1	32 kHz clock (RTXIN)
Clock and calendar functions	• Calendar count mode – Year, month, date, day-of-week, hour, minute, second are counted, BCD display – 12 hours/24 hours mode switching function – 30 seconds adjustment function (a number less than 30 is rounded down to 00 seconds, and 30 seconds or more are rounded up to one minute) – Automatic adjustment function for leap years • Binary count mode – Count seconds in 32 bits, binary display • Common to both modes – Start/stop function – The sub-second digit is displayed in binary units (1 Hz, 2 Hz, 4 Hz, 8 Hz, 16 Hz, 32 Hz, or 64 Hz). – Clock error correction function (Minimum correction amount = 1 cycle of 32 kHz clock) – Clock (1 Hz/64 Hz) output
Interrupts	• Alarm interrupt (ALM) As an alarm interrupt condition, selectable which of the below is compared with: – Calendar count mode: Year, month, date, day-of-week, hour, minute, or second can be selected – Binary count mode: Each bit of the 32-bit binary counter • Periodic interrupt (PRD) – 2 seconds, 1 second, 1/2 second, 1/4 second, 1/8 second, 1/16 second, 1/32 second, 1/64 second, 1/128 second, or 1/256 second can be selected as an interrupt period. • Carry interrupt (CUP) An interrupt is generated at either of the following timings: – When a carry from the 64-Hz counter to the second counter/ binary counter0 is generated. – When a carry occurs from the prescaler to the 64Hz counter when reading the 64Hz counter.
Time capture function	Time can be captured by event input. For event input, month, date, hour, minute, and second are captured or 32-bit binary counter value is captured.

Note 1. Satisfy the frequency of the peripheral module clock ≥ the frequency of the count source.

22.1.2 Block diagram

Figure 22.1 shows a block diagram of the RTC.



R64CNT:	64-Hz counter
RSECCNT/BCNT0:	Second counter/Binary counter 0
RMINCNT/BCNT1:	Minute counter/Binary counter 1
RHRCNT/BCNT2:	Hour counter/Binary counter 2
RWKCNT/BCNT3:	Day-of-week counter/Binary counter 3
RDAYCNT:	Date counter
RMONCNT:	Month counter
RYRCNT:	Year counter
RSR:	Status register
RCR1:	RTC control register 1
RCR2:	RTC control register 2
RADJ:	Time error adjustment register

RSECAR/BCNT0AR:	Second alarm register/Binary counter 0 alarm register
RMINAR/BCNT1AR:	Minute alarm register/Binary counter 1 alarm register
RHRAR/BCNT2AR:	Hour alarm register/Binary counter 2 alarm register
RWKAR/BCNT3AR:	Day-of-week alarm register/Binary counter 3 alarm register
RDAYAR/BCNT0AER:	Date alarm register/Binary counter 0 alarm enable register
RMONAR/BCNT1AER:	Month alarm register/Binary counter 1 alarm enable register
RYRAR/BCNT2AER:	Year alarm register/Binary counter 2 alarm enable register
RYRAREN/BCNT3AER:	Year alarm enable register/Binary counter 3 alarm enable register
RTCCR0:	Time capture control register 0
RSECCP0/BCNT0CP0:	Second capture register 0 / BCNT0 capture register 0
RMINCP0/BCNT1CP0:	Minute capture register 0 / BCNT1 capture register 0
RHRCP0/BCNT2CP0:	Hour capture register 0 / BCNT2 capture register 0
RDAYCP0/BCNT3CP0:	Date capture register 0 / BCNT3 capture register 0
RMONCP0:	Month capture register 0

Figure 22.1 Block Diagram of RTC

22.1.3 External pins

Table 22.2 shows the pin configuration of the RTC.

Table 22.2 Pin Configuration of RTC

Pin Name	I/O	Function
RTXIN	I	Connect a 32.768-kHz crystal to these pins.
RTXIN	O	
RTCTC0	I	Time capture event input pins

22.2 Register Configuration

Base Address: H'0_1004_EC00 (Cortex-A55 Address Space)
H'4004_EC00 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
H'5004_EC00 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 22.3 shows RTC Control Register list.

Table 22.3 RTC Control Register List (1/2)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
64-Hz Counter	R64CNT	R	H'xx	H'00	8
Second Counter/Binary Counter 0	RSECCNT/ BCNT0	R/W	H'xx	H'02	8
Minute Counter/Binary Counter 1	RMINCNT/ BCNT1	R/W	H'xx	H'04	8
Hour Counter/Binary Counter 2	RHRCNT/ BCNT2	R/W	H'xx	H'06	8
Day-of-Week Counter/Binary Counter 3	RWKCNT/ BCNT3	R/W	H'xx	H'08	8
Date Counter	RDAYCNT	R/W	H'xx	H'0A	8
Month Counter	RMONCNT	R/W	H'xx	H'0C	8
Year Counter	RYRCNT	R/W	H'00xx	H'0E	16
Second Alarm Register/ Binary Counter 0 Alarm Register	RSECAR/ BCNT0AR	R/W	H'xx	H'10	8
Minute Alarm Register/ Binary Counter 1 Alarm Register	RMINAR/ BCNT1AR	R/W	H'xx	H'12	8
Hour Alarm Register/ Binary Counter 2 Alarm Register	RHRAR/ BCNT2AR	R/W	H'xx	H'14	8
Day-of-Week Alarm Register/ Binary Counter 3 Alarm Register	RWKAR/ BCNT3AR	R/W	H'xx	H'16	8
Date Alarm Register/ Binary Counter 0 Alarm Enable Register	RDAYAR/ BCNT0AER	R/W	H'xx	H'18	8
Month Alarm Register/ Binary Counter 1 Alarm Enable Register	RMONAR/ BCNT1AER	R/W	H'xx	H'1A	8
Year Alarm Register/ Binary Counter 2 Alarm Enable Register	RYRAR/ BCNT2AER	R/W	H'00xx	H'1C	16
Year Alarm Enable Register/ Binary Counter 3 Alarm Enable Register	RYRAREN/ BCNT3AER	R/W	H'xx	H'1E	8
RTC Status Register	RSR	R/W	H'00*1	H'20	8
RTC Control Register 1	RCR1	R/W	H'xx	H'22	8
RTC Control Register 2	RCR2	R/W	H'xx	H'24	8
Time Error Adjustment Register	RADJ	R/W	H'xx	H'2E	8
Time Capture Control Register 0	RTCCR0	R/W	H'xx	H'40	8
Second Capture Register 0/ BCNT0 Capture Register 0	RSECCP0/ BCNT0CP0	R	H'xx	H'52	8

Table 22.3 RTC Control Register List (2/2)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Minute Capture Register 0/ BCNT1 Capture Register 0	RMINCP0/ BCNT1CP0	R	H'xx	H'54	8
Hour Capture Register 0/ BCNT2 Capture Register 0	RHRCP0/ BCNT2CP0	R	H'xx	H'56	8
Date Capture Register 0/ BCNT3 Capture Register 0	RDAYCP0/ BCNT3CP0	R	H'xx	H'5A	8
Month Capture Register 0	RMONCP0	R	H'xx	H'5C	8

Note 1. Refer to **Section 22.3.17**.

Note: x: Undefined

22.3 Register Description

When writing to or reading from RTC registers, do so in accordance with **Section 22.6.4, Notes on Writing to and Reading from Registers**.

If the value in an RTC register after a reset is given as x (undefined bits) in the list, it is not initialized by a reset. When RTC enters the reset state or a low power consumption state during counting operations (i.e. while the RCR2.START bit is 1), the year, month, day of the week, date, hours, minutes, seconds counters in 64Hz counter and calendar count mode, and binary counter in binary count mode continue to operate. Note that a reset generated during writing to or updating of a register might destroy the register value. For details, refer to **Section 22.6.3, Transitions to Low Power Consumption Modes after Setting Registers**.

22.3.1 64-Hz Counter (R64CNT)

Bit	7	6	5	4	3	2	1	0
	—	1HZ	2HZ	4HZ	8HZ	16HZ	32HZ	64HZ
Initial Value	0	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R

x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
0	64HZ	x	R	Indicate the state between 1 Hz and 64 Hz.
1	32HZ	x	R	
2	16HZ	x	R	
3	8HZ	x	R	
4	4HZ	x	R	
5	2HZ	x	R	
6	1HZ	x	R	
7	—	0	R	Reserved This bit is read as 0. Writing to this bit has no effect.

The R64CNT counter is used in both calendar count mode and in binary count mode.

The 64-Hz counter (R64CNT) generates a period of one second by counting the 128-Hz clock.

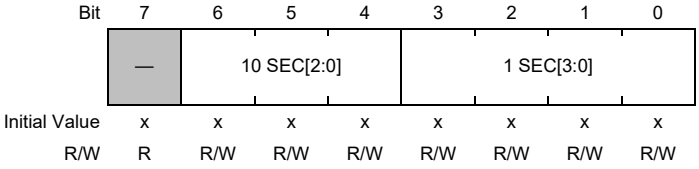
The state in the sub-second range can be confirmed by reading this counter.

This counter is set to H'00 by an RTC software reset or executing 30-second adjustment.

To read this counter, follow the procedure in **Section 22.4.5, Reading 64-Hz Counter and Time**.

22.3.2 Second Counter (RSECCNT)/Binary Counter 0 (BCNT0)

a. In calendar count mode:



x: Undefined

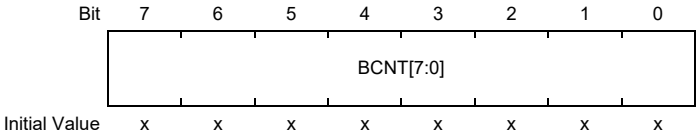
Bit	Bit Name	Initial Value	R/W	Description
3 to 0	1 SEC[3:0]	x	R/W	1-Second Count Counts from 0 to 9 every second. When a carry is generated, 1 is added to the tens place.
6 to 4	10 SEC[2:0]	x	R/W	10-Second Count Counts from 0 to 5 for 60-second counting.
7	—	x	R	Reserved Set this bit to 0. It is read as undefined value.

The RSECCNT counter is used for setting and counting the BCD-coded second value. It counts carries generated once per second in the 64-Hz counter.

The setting range is decimal 00 to 59. The RTC will not operate normally if any other value is set. Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

Refer to **Section 22.4.5, Reading 64-Hz Counter and Time** for notes on accessing registers.

b. In binary count mode:



x: Undefined

The BCNT0 counter is a readable/writable 32-bit binary counter b7 to b0.

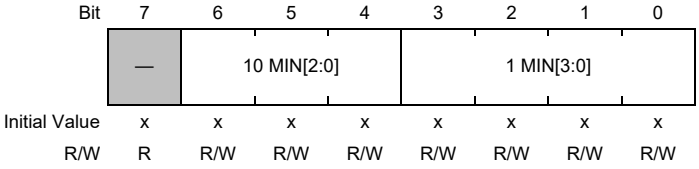
The 32-bit binary counter performs count operation by a carry generating for each second of the 64-Hz counter.

Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

To read this counter, follow the procedure in **Section 22.4.5, Reading 64-Hz Counter and Time**.

22.3.3 Minute Counter (RMINCNT)/Binary Counter 1 (BCNT1)

a. In calendar count mode:



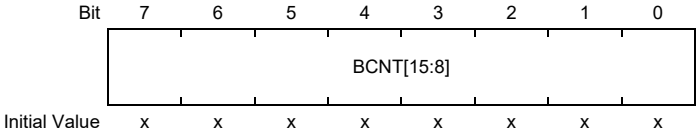
x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	MIN1[3:0]	x	R/W	1-Minute Count Counts from 0 to 9 every minute. When a carry is generated, 1 is added to the tens place.
6 to 4	MIN10[2:0]	x	R/W	10-Minute Count Counts from 0 to 5 for 60-minute counting.
7	—	x	R	Reserved Set this bit to 0. It is read as undefined value.

The RMINCNT counter is used for setting and counting the BCD-coded minute value. It counts carries generated once per minute in the second counter.

A value from 00 through 59 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2. To read this counter, follow the procedure in **Section 22.4.5, Reading 64-Hz Counter and Time**.

b. In binary count mode:



x: Undefined

The BCNT1 counter is a readable/writable 32-bit binary counter b15 to b8.

The 32-bit binary counter performs count operation by a carry generating for each second of the 64-Hz counter.

Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

To read this counter, follow the procedure in **Section 22.4.5, Reading 64-Hz Counter and Time**.

22.3.4 Hour Counter (RHRCNT)/Binary Counter 2 (BCNT2)

a. In calendar count mode:

Bit	7	6	5	4	3	2	1	0
	—	PM	10HR[1:0]		1HR[3:0]			
Initial Value	x	x	x	x	x	x	x	x
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	HR1[3:0]	x	R/W	1-Hour Count Counts from 0 to 9 once per hour. When a carry is generated, 1 is added to the tens place.
5, 4	HR10[1:0]	x	R/W	10-Hour Count Counts from 0 to 2 once per carry from the ones place.
6	PM	x	R/W	PM Time Counter Setting for a.m./p.m. 0: a.m. 1: p.m.
7	—	x	R	Reserved Set this bit to 0. It is read as undefined value.

The RHRCNT counter is used for setting and counting the BCD-coded hour value. It counts carries generated once per hour in the minute counter.

The specifiable time differs according to the setting in the hours mode bit (RCR2.HR24).

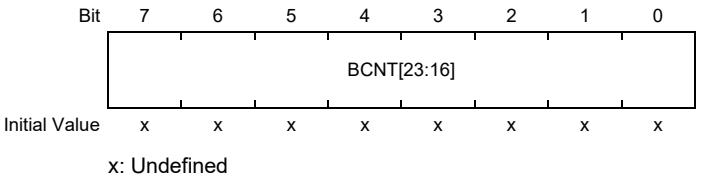
When the RCR2.HR24 bit is 0: From 00 to 11 (in BCD)
When the RCR2.HR24 bit is 1: From 00 to 23 (in BCD)

If a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

The PM bit is only enabled when the RCR2.HR24 bit is 0. Otherwise, the setting in the PM bit has no effect.

After writing to the RHRCNT counter, confirm that its value has actually changed before proceeding with further processing. To read this counter, follow the procedure in **Section 22.4.5, Reading 64-Hz Counter and Time**.

b. In binary count mode:



The BCNT2 counter is a readable/writable 32-bit binary counter b23 to b16.

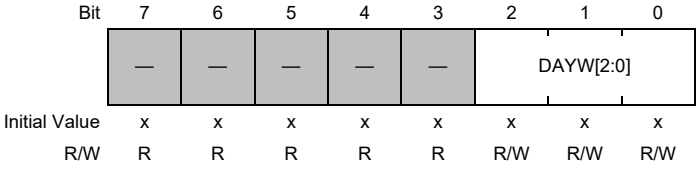
The 32-bit binary counter performs count operation by a carry generating for each second of the 64-Hz counter.

Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

To read this counter, follow the procedure in **Section 22.4.5, Reading 64-Hz Counter and Time**.

22.3.5 Day-of-Week Counter (RWKCNT)/Binary Counter 3 (BCNT3)

a. In calendar count mode:



x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	DAYW[2:0]	x	R/W	Day-of-Week Counting b2 b0 0 0 0: Sunday 0 0 1: Monday 0 1 0: Tuesday 0 1 1: Wednesday 1 0 0: Thursday 1 0 1: Friday 1 1 0: Saturday 1 1 1: Setting Prohibited
7 to 3	—	x	R	Reserved Set these bits to 0. They are read as undefined value

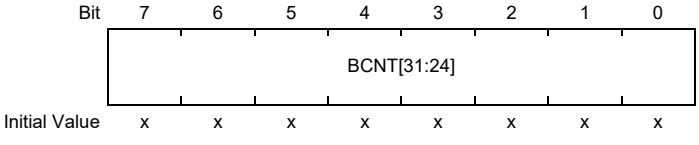
The RWKCNT counter is used for setting and counting in the coded day-of-week value. It counts carries generated once per day in the hour counter.

A value from 0 through 6 can be specified; if a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

To read this counter, follow the procedure in **Section 22.4.5, Reading 64-Hz Counter and Time**.

b. In binary count mode:

Address(es): RTC.BCNT3 H'0008_C408



x: Undefined

The BCNT3 counter is a readable/writable 32-bit binary counter b31 to b24.

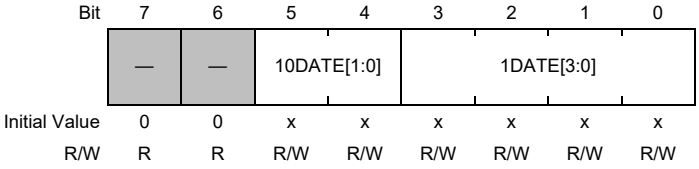
The 32-bit binary counter performs count operation by a carry generating for each second of the 64-Hz counter.

Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

To read this counter, follow the procedure in **Section 22.4.5, Reading 64-Hz Counter and Time**.

22.3.6 Data Counter (RDAYCNT)

a. In calendar count mode:



x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	DATE1[3:0]	x	R/W	1-Day Count Counts from 0 to 9 once per day. When a carry is generated, 1 is added to the tens place.
5, 4	DATE10[1:0]	x	R/W	10-Day Count Counts from 0 to 3 once per carry from the ones place.
7, 6	—	x	R	Reserved These bits are read as 0. The write value should be 0.

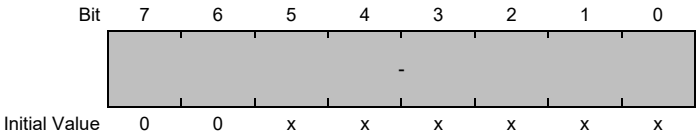
The RDAYCNT counter is used for setting and counting the BCD-coded date value. It counts carries generated once per day in the hour counter. The count operation depends on the month and whether the year is a leap year.

Leap years are determined according to whether the year counter (RYRCNT) value is divisible by 400, 100, and 4.

A value from 01 through 31 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly. (When specifying a value, note that the range of specifiable days depends on the month and whether the year is a leap year.) Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

To read this counter, follow the procedure in **Section 22.4.5, Reading 64-Hz Counter and Time**.

b. In binary count mode:



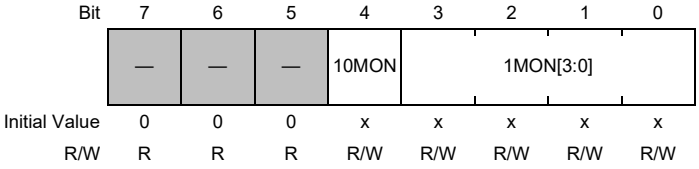
x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
5 to 0	—	x	R	Reserved Set these bits to 0. They are read as Undefined value.
7, 6	—	0	R	Reserved These bits are read as 0. The write value should be 0.

The RDAYCNT counter is not used in binary count mode.

22.3.7 Month Counter (RMONCNT)

a. In calendar count mode:



x: Undefined

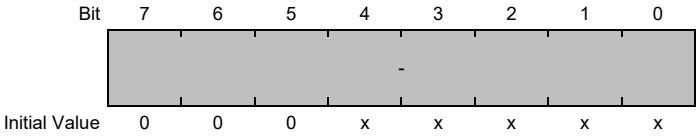
Bit	Bit Name	Initial Value	R/W	Description
3 to 0	1MON[3:0]	x	R/W	1-Month Count Counts from 0 to 9 once per month. When a carry is generated, 1 is added to the tens place.
4	10MON	x	R/W	10-Month Count Counts from 0 to 1 once per carry from the ones place.
7 to 5	—	x	R	Reserved These bits are read as 0. The write value should be 0.

The RMONCNT counter is used for setting and counting the BCD-coded month value. It counts carries generated once per month in the date counter.

A value from 01 through 12 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

To read this counter, follow the procedure in **Section 22.4.5, Reading 64-Hz Counter and Time**.

b. In binary count mode:



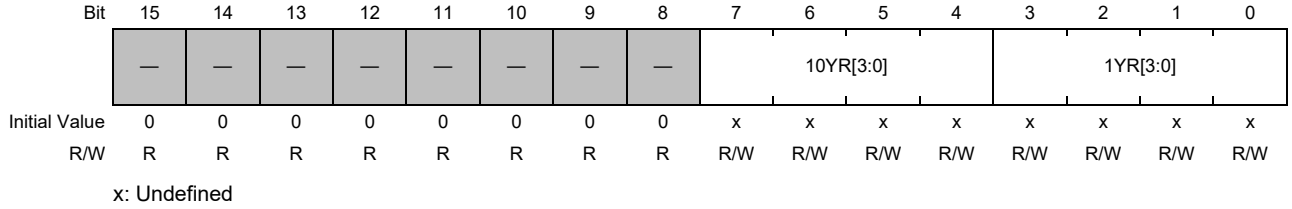
x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
4 to 0	—	x	R	Reserved Set these bits to 0. They are read as Undefined value.
7 to 5	—	0	R	Reserved These bits are read as 0. The write value should be 0.

The RMONCNT counter is not used in binary count mode.

22.3.8 Year Counter (RYRCNT)

a. In calendar count mode:



x: Undefined

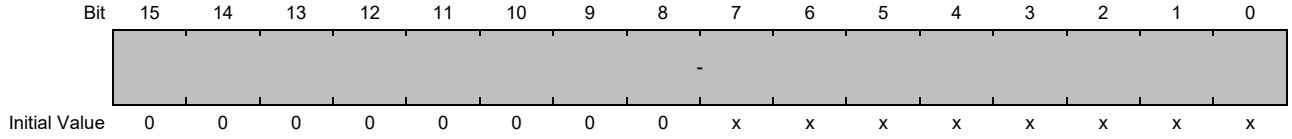
Bit	Bit Name	Initial Value	R/W	Description
3 to 0	1YR[3:0]	x	R/W	1-Year Count Counts from 0 to 9 once per year. When a carry is generated, 1 is added to the tens place.
7 to 4	10YR[3:0]	x	R/W	10-Year Count Counts from 0 to 9 once per carry from ones place. When a carry is generated in the tens place, 1 is added to the hundreds place.
15 to 8	—	0	R	Reserved These bits are read as 0. The write value should be 0.

The RYRCNT counter is used for setting and counting the BCD-coded year value. It counts carries generated once per year in the month counter.

A value from 00 through 99 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

To read this counter, follow the procedure in **Section 22.4.5, Reading 64-Hz Counter and Time**.

b. In binary count mode:



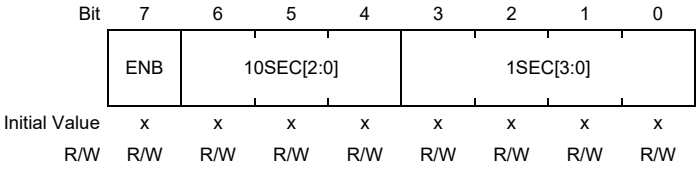
x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	—	x	R	Reserved Set these bits to 0. They are read as Undefined value.
15 to 8	—	0	R	Reserved These bits are read as 0. The write value should be 0.

The RYRCNT counter is not used in binary count mode.

22.3.9 Second Alarm Register (RSECAR)/Binary Counter 0 Alarm Register (BCNT0AR)

a. In calendar count mode:



x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	1SEC[3:0]	x	R/W	1 Second Value for the ones place of seconds
6 to 4	10SEC[2:0]	x	R/W	10 Seconds Value for the tens place of seconds
7	ENB	x	R/W	ENB 0: The register value is not compared with the RSECCNT counter value. 1: The register value is compared with the RSECCNT counter value.

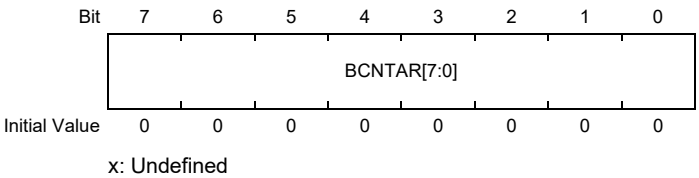
The RSECAR register is an alarm register corresponding to the BCD-coded second counter RSECCNT. When the ENB bit is set to 1, the RSECAR value is compared with the RSECCNT value. From among the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAREN), only those selected with the ENB bits set to 1 are compared with the corresponding counters. When the respective values all match, the alarm flag of RSR is set to 1.

RSECAR values from 00 through 59 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly.

This register is set to H’00 by an RTC software reset.

b. In binary count mode:

Address(es): RTC.BCNT0AR H’0008_C410



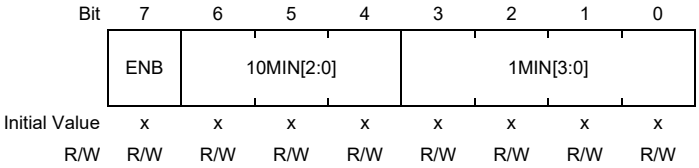
x: Undefined

The BCNT0AR counter is a readable/writable alarm register corresponding to 32-bit binary counter b7 to b0.

This register is set to H'00 by an RTC software reset.

22.3.10 Minute Alarm Register (RMINAR)/Binary Counter 1 Alarm Register (BCNT1AR)

a. In calendar count mode:



x: Undefined

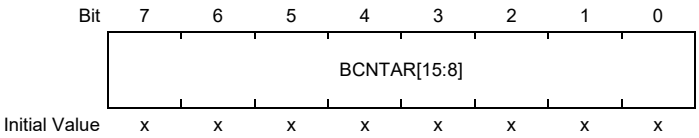
Bit	Bit Name	Initial Value	R/W	Description
3 to 0	1MIN[3:0]	x	R/W	1 Minute Value for the ones place of minutes
6 to 4	10MIN[2:0]	x	R/W	10 Minutes Value for the tens place of minutes
7	ENB	x	R/W	ENB 0: The register value is not compared with the RMINCNT counter value. 1: The register value is compared with the RMINCNT counter value.

The RMINAR register is an alarm register corresponding to the BCD-coded minute counter RMINCNT. When the ENB bit is set to 1, the RMINAR value is compared with the RMINCNT value. From among the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAREN), only those selected with the ENB bits set to 1 are compared with the corresponding counters. When the respective values all match, the alarm flag of RSR is set to 1.

RMINAR values from 00 through 59 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly.

This register is set to H’00 by an RTC software reset.

b. In binary count mode:



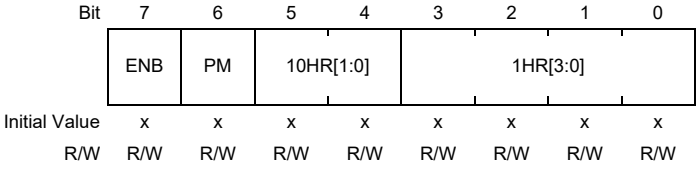
x: Undefined

The BCNT1AR counter is a readable/writable alarm register corresponding to 32-bit binary counter b15 to b8.

This register is set to H'00 by an RTC software reset.

22.3.11 Hour Alarm Register (RHRAR)/Binary Counter 2 Alarm Register (BCNT2AR)

a. In calendar count mode:



Bit	Bit Name	Initial Value	R/W	Description
3 to 0	1HR[3:0]	x	R/W	1 Hour Value for the ones place of hours
5, 4	10HR[1:0]	x	R/W	10 Hours Value for the tens place of hours
6	PM	x	R/W	PM Time Alarm Setting for a.m./p.m. 0: a.m. 1: p.m.
7	ENB	x	R/W	ENB 0: The register value is not compared with the RHRCNT counter value. 1: The register value is compared with the RHRCNT counter value.

The RHRAR register is an alarm register corresponding to the BCD-coded hour counter RHRCNT. When the ENB bit is set to 1, the RHRAR value is compared with the RHRCNT value. From among the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAREN), only those selected with the ENB bits set to 1 are compared with the corresponding counters. When the respective values all match, the alarm flag of RSR is set to 1.

The specifiable time differs according to the setting in the hours mode bit (RCR2.HR24).

When the RCR2.HR24 bit is 0: From 00 to 11 (in BCD)
When the RCR2.HR24 bit is 1: From 00 to 23 (in BCD)

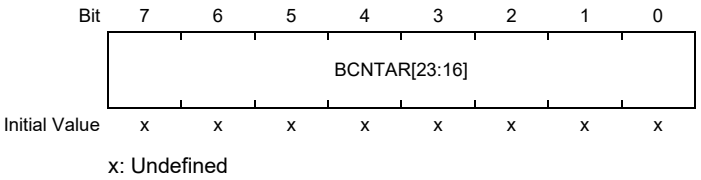
If a value outside of this range is specified, the RTC does not operate correctly.

When the RCR2.HR24 bit is 0, be sure to set the PM bit.

When the RCR2.HR24 bit is 1, the setting in the PM bit has no effect.

This register is set to H’00 by an RTC software reset.

b. In binary count mode:



The BCNT2AR counter is a readable/writable alarm register corresponding to 32-bit binary counter b23 to b16.

This register is set to H’00 by an RTC software reset.

22.3.12 Day-of-Week Alarm Register (RWKAR)/Binary Counter 3 Alarm Register (BCNT3AR)

a. In calendar count mode:

Bit	7	6	5	4	3	2	1	0
	ENB	—	—	—	—	DAYW[2:0]		
Initial Value	x	x	x	x	x	x	x	x
R/W	R/W	R	R	R	R	R/W	R/W	R/W

x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	DAYW[2:0]	x	R/W	Day-of-Week Setting b2 b0 0 0 0: Sunday 0 0 1: Monday 0 1 0: Tuesday 0 1 1: Wednesday 1 0 0: Thursday 1 0 1: Friday 1 1 0: Saturday 1 1 1: Setting Prohibited
6 to 3	—	x	R	Reserved Set these bits to 0. They are read as the Undefined value.
7	ENB	x	R/W	ENB 0: The register value is not compared with the RWKCNT counter value. 1: The register value is compared with the RWKCNT counter value.

The RWKAR register is an alarm register corresponding to the coded day-of-week counter RWKCNT. When the ENB bit is set to 1, the RWKAR value is compared with the RWKCNT value. From among the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAREN), only those selected with the ENB bits set to 1 are compared with the corresponding counters. When the respective values all match, the alarm flag of RSR is set to 1.

RWKAR values from 0 through 6 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly.

This register is set to H'00 by an RTC software reset.

b. In binary count mode:

Bit	7	6	5	4	3	2	1	0
	BCNTAR[31:24]							
Initial Value	x	x	x	x	x	x	x	x

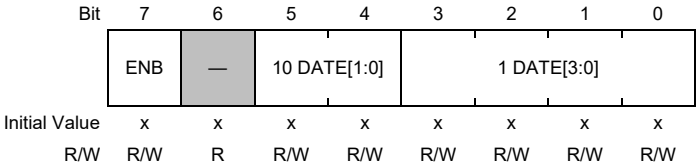
x: Undefined

The BCNT3AR counter is a readable/writable alarm register corresponding to 32-bit binary counter b31 to b24.

This register is set to H'00 by an RTC software reset.

22.3.13 Date Alarm Register (RDAYAR)/Binary Counter 0 Alarm Enable Register (BCNT0AER)

a. In calendar count mode:



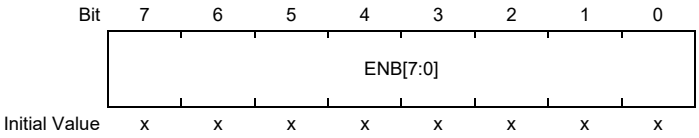
Bit	Bit Name	Initial Value	R/W	Description
3 to 0	1 DATE[3:0]	x	R/W	1 Day Value for the ones place of days
5, 4	10 DATE[1:0]	x	R/W	10 Days Value for the tens place of days
6	—	x	R	Reserved Set this bit to 0. It is read as Undefined value.
7	ENB	x	R/W	ENB 0: The register value is not compared with the RDAYCNT counter value. 1: The register value is compared with the RDAYCNT counter value.

The RDAYAR register is an alarm register corresponding to the BCD-coded date counter RDAYCNT. When the ENB bit is set to 1, the RDAYAR value is compared with the RDAYCNT value. From among the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAREN), only those selected with the ENB bits set to 1 are compared with the corresponding counters. When the respective values all match, the alarm flag of RSR is set to 1.

RDAYAR values from 01 through 31 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly.

This register is set to H'00 by an RTC software reset.

b. In binary count mode:



x: Undefined

The BCNT0AER register is a readable/writable register for setting the alarm enable corresponding to 32-bit binary counter b7 to b0. Among the ENB[31:0] bits, the binary counter (BCNT[31:0]) corresponding to the bits which are set to 1 and the binary alarm register (BCNTAR[31:0]) are compared, and when all match, the alarm flag of RSR is set to 1.

This register is set to H'00 by an RTC software reset.

22.3.14 Month Alarm Register (RMONAR)/Binary Counter 1 Alarm Enable Register (BCNT1AER)

a. In calendar count mode:

Bit	7	6	5	4	3	2	1	0
	ENB	—	—	10 MON	1 MON[3:0]			
Initial Value	x	x	x	x	x	x	x	x
R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	1 MON[3:0]	x	R/W	1 Month Value for the ones place of months
4	10 MON	x	R/W	10 Months Value for the tens place of months
6, 5	—	x	R	Reserved Set these bits to 0. They are read as the set value.
7	ENB	x	R/W	ENB 0: The register value is not compared with the RMONCNT counter value. 1: The register value is compared with the RMONCNT counter value.

The RMONAR register is an alarm register corresponding to the BCD-coded month counter RMONCNT. When the ENB bit is set to 1, the RMONAR value is compared with the RMONCNT value. From among the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAREN), only those selected with the ENB bits set to 1 are compared with the corresponding counters. When the respective values all match, the alarm flag of RSR is set to 1.

RMONAR values from 01 through 12 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly.

This register is set to H’00 by an RTC software reset.

b. In binary count mode:

Bit	7	6	5	4	3	2	1	0
	ENB[15:8]							
Initial Value	x	x	x	x	x	x	x	x

x: Undefined

The BCNT1AER register is a readable/writable register for setting the alarm enable corresponding to 32-bit binary counter b15 to b8. Among the ENB[31:0] bits, the binary counter (BCNT[31:0]) corresponding to the bits which are set to 1 and the binary alarm register (BCNTAR[31:0]) are compared, and when all match, the alarm flag of RSR is set to 1.

This register is set to H’00 by an RTC software reset.

22.3.15 Year Alarm Register (RYRAR)/Binary Counter 2 Alarm Enable Register (BCNT2AER)

a. In calendar count mode:

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	10 YR[3:0]				1 YR[3:0]			
Initial Value	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	1 YR[3:0]	x	R/W	1 Year Value for the ones place of years
7 to 4	10 YR[3:0]	x	R/W	10 Years Value for the tens place of years
15 to 8	—	0	R	Reserved These bits are read as 0. The write value should be 0.

The RYRAR register is an alarm register corresponding to the BCD-coded year counter RYRCNT.

RYRAR values from 00 through 99 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly.

b. In binary count mode:

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ENB[23:16]							
Initial Value	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x

x: Undefined

The BCNT2AER register is a readable/writable register for setting the alarm enable corresponding to 32-bit binary counter b23 to b16. Among the ENB[31:0] bits, the binary counter (BCNT[31:0]) corresponding to the bits which are set to 1 and the binary alarm register (BCNTAR[31:0]) are compared, and when all match, the alarm flag of RSR is set to 1.

This register is set to H’0000 by an RTC software reset.

22.3.16 Year Alarm Enable Register (RYRAREN)/Binary Counter 3 Alarm Enable Register (BCNT3AER)

a. In calendar count mode:

Bit	7	6	5	4	3	2	1	0
	ENB	—	—	—	—	—	—	—
Initial Value	x	x	x	x	x	x	x	x
R/W	R/W	R	R	R	R	R	R	R

x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
6 to 0	—	x	R	Reserved Set these bits to 0. They are read as Undefined value.
7	ENB	x	R/W	ENB 0: The register value is not compared with the RYRCNT counter value. 1: The register value is compared with the RYRCNT counter value.

The RYRAREN register is an enable register of Year Alarm.

When the ENB bit in the RYRAREN register is set to 1, the RYRAR value is compared with the RYRCNT value. From among the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAREN), only those selected with the ENB bits set to 1 are compared with the corresponding counters. When the respective values all match, the alarm flag of RSR is set to 1.

This register is set to H'00 by an RTC software reset.

b. In binary count mode:

Bit	7	6	5	4	3	2	1	0
	ENB[31:24]							
Initial Value	x	x	x	x	x	x	x	x

x: Undefined

The BCNT3AER register is a readable/writable register for setting the alarm enable corresponding to 32-bit binary counter b31 to b24. Among the ENB[31:0] bits, the binary counter (BCNT[31:0]) corresponding to the bits which are set to 1 and the binary alarm register (BCNTAR[31:0]) are compared, and when all match, the alarm flag of RSR is set to 1.

This register is set to H'00 by an RTC software reset.

22.3.17 RTC Status Register (RSR)

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	PF	CF	AF
Initial Value	0	0	0	0	0	0*1	0*1	0*1
R/W	R	R	R	R	R	R/W	R/W	R/W

x: Undefined

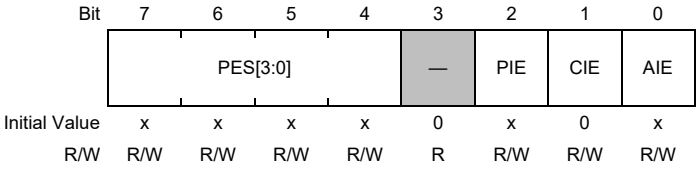
Bit	Bit Name	Initial Value	R/W	Description
0	AF	0*1	R/W	Alarm flag The alarm time set in the alarm registers (calendar count mode: RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, RYRAR, binary count mode: BCNT0AR, BCNT1AR, BCNT2AR, BCNT3AR) (only for registers with the ENB bit set to 1) and Set to 1 when the counters match. [Condition to become 0] • When 0 is written to AF. [Condition to become 1] • When the alarm register (only for registers with the ENB bit set to 1) matches the counter. 0: Alarm register and counter do not match. 1: Alarm register and counter match.*2
1	CF	0*1	R/W	Carry flag When the CF flag is set to 1, it indicates that a carry to Second Counter/Binary Counter 0 or a carry to 64-Hz counter has occurred when reading the 64-Hz counter, and the count register read at this time. The value of is not guaranteed. Need to read again. [Condition to become 0] • When 0 is written to the CF flag. [Condition to become 1] • When a carry to the second counter/binary counter 0 or a carry to the 64-Hz counter occurs when reading the 64-Hz counter. • When 1 is written to the CF flag. 0: No carry to the second counter/binary counter 0 and no carry to the 64-Hz counter when reading the 64-Hz counter. 1: With carry to the second counter/binary Counter 0 and carry to the 64-Hz counter when reading the 64-Hz counter.
2	PF	0*1	R/W	Periodic flag The PF flag is a flag that indicates interrupt generation at the cycle set by the RCR1.PES[3:0] bits. When this flag is set to 1, a periodic interrupt is generated. [Condition to become 0] • When 0 is written to the PF flag. [Condition to become 1] • When an interrupt occurs at the cycle set by the RCR1.PES[3:0] bits. 0: No interrupt occurs at the cycle set by the RCR1.PES[3:0] bits. 1: An interrupt occurs at the cycle set by the RCR1.PES[3:0] bits.*2
7 to 3	—	0	R	Reserved These bits are read as 0. The write value should be 0.

Note 1. The read value may be 1 after reset release.

Note 2. Writing 1 to this bit has no effect.

The RSR register is a periodic interrupt, carry and alarm flag register. The RCR1 register is used in both calendar count mode and in binary count mode. Each flag is set to 1 when the prescaler and watch counter match each interrupt setting condition. Since the prescaler, clock counter, and each interrupt setting register are not reset, flags may be set before each flag is read. This register is set to H'00 by an RTC software reset.

22.3.18 RTC Control Register 1 (RCR1)



x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
0	AIE	x	R/W	Alarm Interrupt Enable This bit enables or disables alarm interrupt requests. 0: An alarm interrupt request is disabled. 1: An alarm interrupt request is enabled.
1	CIE	0	R/W	Carry Interrupt Enable This bit enables and disables interrupt requests when a carry to the RSECCNT/BCNT0 register occurs, or when a carry to the 64-Hz counter (R64CNT) occurs while reading the 64-Hz counter. 0: A carry interrupt request is disabled. 1: A carry interrupt request is enabled.
2	PIE	x	R/W	Periodic Interrupt Enable This bit enables or disabled a periodic interrupt. 0: A periodic interrupt request is disabled. 1: A periodic interrupt request is enabled.
3	—	0	R	Reserved This bit is read as 0. The write value should be 0.
7 to 4	PES[3:0]	x	R/W	Periodic Interrupt Select These bits specify the period for the periodic interrupt. A periodic interrupt is generated with the period specified by these bits. b7 b4 0 1 1 0: A periodic interrupt is generated every 1/256 second. 0 1 1 1: A periodic interrupt is generated every 1/128 second. 1 0 0 0: A periodic interrupt is generated every 1/64 second. 1 0 0 1: A periodic interrupt is generated every 1/32 second. 1 0 1 0: A periodic interrupt is generated every 1/16 second. 1 0 1 1: A periodic interrupt is generated every 1/8 second. 1 1 0 0: A periodic interrupt is generated every 1/4 second. 1 1 0 1: A periodic interrupt is generated every 1/2 second. 1 1 1 0: A periodic interrupt is generated every 1 second. 1 1 1 1: A periodic interrupt is generated every 2 seconds. Other than above: No periodic interrupts are generated.

The RCR1 register is used in both calendar count mode and in binary count mode.

Bits AIE, PIE, and PES[3:0] are updated synchronously with the count source. When the RCR1 register is modified, check that all the bits have been updated before proceeding to the next processing.

22.3.19 RTC Control Register 2 (RCR2)

Bit	7	6	5	4	3	2	1	0
	CNTMD	HR24	AADJP	AADJE	—	ADJ30	RESET	START
Initial Value	x	x	x	x	0	0	0	x
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
0	START	x	R/W	Start*4 This bit stops or restarts the prescaler or counter (clock) operation. The START bit is updated in synchronization with the count source. When the START bit is modified, check that the bit is updated before proceeding to the next processing. 0: Prescaler and counter are stopped. 1: Prescaler and counter operate normally.
1	RESET	x	R/W	RTC Software Reset This bit initializes the prescaler and registers to be reset by RTC software reset. When 1 is written to the RESET bit, the initialization starts in synchronization with the count source. When the initialization is completed, the RESET bit is automatically set to 0. When 1 is written to the RESET bit, check that the bit is set to 0, and then make next settings. • In writing 0: Writing is invalid. 1: The prescaler and the target registers for RTC software reset*1 are initialized. • In reading 0: In normal time operation, or an RTC software reset has completed. 1: During an RTC software reset
2	ADJ30	x	R/W	30-Second Adjustment*2 This bit is for 30-second adjustment. When 1 is written to the ADJ30 bit, the RSECCNT value of 30 seconds or less is rounded down to 00 second and the value of 30 seconds or more is rounded up to 1 minute. The 30-second adjustment is performed in synchronization with the count source. When 1 is written to this bit, the ADJ30 bit is automatically set to 0 after the 30-second adjustment is completed. In case when 1 is written to the ADJ30 bit, check that the bit is set to 0, and then make next settings. When the 30-second adjustment is performed, the prescaler and R64CNT are also reset. The ADJ30 bit is set to 0 by an RTC software reset. This bit is reserved in binary counter mode. The write value should be 0. • In writing 0: Writing is invalid. 1: 30-second adjustment is executed. • In reading 0: In normal time operation, or 30-second adjustment has completed. 1: During 30-second adjustment
3	—	0	R	Reserved This bit is read as 0. The write value should be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	AADJE	0	R/W	Automatic Adjustment Enable*3,*4 This bit controls (enables or disables) automatic adjustment. Set the plus–minus bits (RADJ.PMADJ[1:0]) to H'00 (adjustment is not performed) before changing the value of the AADJE bit. The AADJE bit is set to 0 by an RTC software reset. 0: Automatic adjustment is disabled. 1: Automatic adjustment is enabled.
5	AADJP	0	R/W	Automatic Adjustment Period Select*3,*4 This bit selects the automatic-adjustment period. Set the plus–minus bits (RADJ.PMADJ[1:0]) to H'00 (adjustment is not performed) before changing the value of the AADJP bit. The AADJP bit is set to 0 by an RTC software reset. 0: The RADJ.ADJ[5:0] setting value is adjusted from the count value of the prescaler every minute (every 32 seconds in binary counter mode). 1: The RADJ.ADJ[5:0] setting value is adjusted from the count value of the prescaler every 10 seconds (every 8 seconds in binary counter mode).
6	HR24	x	R/W	Hours Mode*2,*4 This bit specifies whether the RTC will operate in 12- or 24-hour mode. Use the START bit to stop counting by the counters before changing the value of the HR24 bit. Do not stop counting (write 0 to the START bit) and change the value of the HR24 bit at the same time. This bit is reserved in binary counter mode. The write value should be 0. 0: The RTC operates in 12-hour mode. 1: The RTC operates in 24-hour mode.
7	CNTMD	x	R/W	Count Mode Select*4 This bit specifies whether the RTC count mode is operated in calendar count mode or in binary count mode. After setting the count mode, execute an RTC software reset and start again from the initial settings. The CNTMD bit is updated in synchronization with the count source, so when the value of the CNTMD bit has been changed, check that the value of the bit has actually been updated before applying the RTC software reset. The count mode changes to that which was specified beforehand in the CNTMD bit after the RTC software reset is applied. For details on initial settings, refer to **Section 22.4.1, Outline of Initial Settings of Registers after Power On**. 0: The calendar count mode. 1: The binary count mode.

Note 1. R64CNT, RSECAR/BCNT0AR, RMINAR/BCNT1AR, RHRAR/BCNT2AR, RWKAR/BCNT3AR, RDAYAR/BCNT0AER, RMONAR/BCNT1AER, RYRAR/BCNT2AER, RYRAREN/BCNT3AER, RADJ, RTCCR0, RSECCP0/BCNT0CP0, RMINCP0/BCNT1CP0, RHRCP0/BCNT2CP0, RDAYCP0/BCNT3CP0, RMONCP0, RCR2.ADJ30, RCR2.AADJE, RCR2.AADJP

Note 2. This bit is reserved in binary counter mode. The write value should be 0.

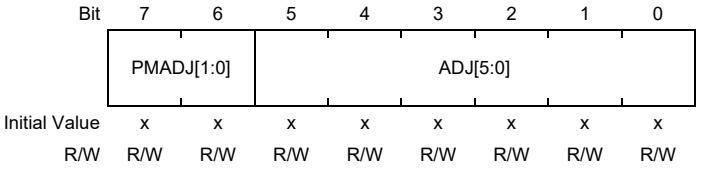
Note 3. When the main clock is selected, the setting of this bit is disabled.

Note 4. After writing to this bit, confirm that its value has actually changed before proceeding with further processing.
Refer to **Section 22.6.4, Notes on Writing to and Reading from Registers** regarding changes to the values of the AADJE, AADJP, and HR24 bits.

The RCR2 register is related to hours mode, automatic adjustment function, 30-second adjustment, RTC software reset, and controlling count operation.

Do not rewrite the RCR2 register by read-modify-write.

22.3.20 Time Error Adjustment Register (RADJ)



x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
5 to 0	ADJ[5:0]	x	R/W	Adjustment Value These bits specify the adjustment value (the number of sub-clock cycles) from the prescaler.
7, 6	PMADJ[1:0]	x	R/W	Plus–Minus These bits select whether the clock is set ahead or back depending on the error-adjustment value set in the ADJ[5:0] bits. b7 b6 0 0: Adjustment is not performed. 0 1: Adjustment is performed by the addition to the prescaler. 1 0: Adjustment is performed by the subtraction from the prescaler. 1 1: Setting prohibited

Adjustment is performed by the addition to or subtraction from the prescaler.

In case when the automatic adjustment enable (RCR2.AADJE) bit is 0, adjustment is performed when writing to the RADJ.

In case when the RCR2.AADJE bit is 1, adjustment is performed in the interval specified by the automatic adjustment period select (RCR2.AADJP) bit.

The current adjustment by software (disabling automatic adjustment) may be invalid if the following adjustment value is specified within 320 cycles of the count source after the register setting. To perform adjustment consecutively, wait for 320 cycles or more of the count source after the register setting and then specify the next adjustment value.

RADJ is updated in synchronization with the count source. When RADJ is modified, check that all the bits have been updated before continuing with further processing.

This register is set to H’00 by an RTC software reset.

ADJ[5:0] Bits (Adjustment Value)

These bits specify the adjustment value (the number of 32kHz-clock cycles) from the prescaler.

PMADJ[1:0] Bits (Plus–Minus)

These bits select whether the clock is set ahead or back depending on the error-adjustment value set in the ADJ[5:0] bits.

22.3.21 Time Capture Control Register (RTCCR0)

Bit	7	6	5	4	3	2	1	0
	TCEN	—	TCNF[1:0]		—	TCST	TCCT[1:0]	
Initial Value	x	0	x	x	0	x	x	x
R/W	R/W	R	R/W	R/W	R	R/W	R/W	R/W

x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
1, 0	TCCT[1:0]	x	R/W	Time Capture Control These bits control the edge detection of the time capture event input pins (RTCTC0). The detection edge is selectable. The TCCT[1:0] bits should be set while the TCEN bit is 1. b1 b0 0 0: No event is detected. 0 1: Rising edge is detected. 1 0: Falling edge is detected. 1 1: Both edges are detected.
2	TCST	x	R/W	Time Capture Status This bit indicates that an event of the time capture event input pins (RTCTC0) has been detected. When the TCST bit is 0, no event is detected. When the TCST bit is 1, this bit indicates that an event of the corresponding pin has been detected and the capture register is valid. When multiple events have been detected, the capture time for the first event is retained. Writing 0 sets the TCST bit to 0. In addition, writing any other value except 0 has no effect. Set the TCST bit while the TCCT[1:0] bits are b'00 (no event is detected). The TCST bit is set to 0 in synchronization with the count source. When the TCST bit is set to 0, check that the bit has been updated before continuing with further processing. 0: No event is detected. 1: An event is detected.*1
3	—	x	R	Reserved This bit is read as 0. The write value should be 0.
5, 4	TCNF[1:0]	x	R/W	Time Capture Noise Filter Control These bits control the noise filter of the time capture event input pin (RTCTC0). When the noise filter is on, the count source divided by 1 or divided by 32 is selectable. In this case, when the input level on the time capture event input pin matches three consecutive times at the set sampling period, the input level is determined. Set the TCNF[1:0] bits while the TCCT[1:0] bits are b'00 (no event is detected). When the noise filter is used, after setting the TCNF[1:0] bits to 1, wait for three cycles of the specified sampling period, and then set the TCCT[1:0] bits. Set the TCNF[1:0] bits when the TCEN bit is 1. b5 b4 0 0: The noise filter is off. 0 1: Setting prohibited 1 0: The noise filter is on (count source). 1 1: The noise filter is on (count source by divided by 32).
6	—	x	R	Reserved This bit is read as 0. The write value should be 0.

Bit	Bit Name	Initial Value	R/W	Description
7	TCEN	x	R/W	Time Capture Event Input Pin Enable This bit enables or disables the time capture event input pin (RTCTC0). To enable the time capture event input pin, set also the TAMPICR1.CH0EN bit*2 to 1. If the TCEN bit is set to 0, set also the TCCT[1:0] bits to 00b. 0: The RTCTC0 pin is disabled as the time capture event input. 1: The RTCTC0 pin is enabled as the time capture event input.

Note 1. Indicates that an event has been detected. Writing 1 to this bit has no effect. Writing 0 sets this bit to 0.

Note 2. For details, refer to **Section 42, Battery Backup Function (VBATTB)**.

The RTCCR0 register is used both in calendar count mode and in binary count mode.

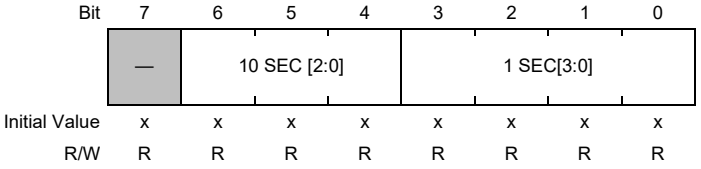
RTCCR0 control the RTCTC0 pin, respectively.

RTCCR0 is updated in synchronization with the count source. When RTCCR0 is modified, check that all the bits except for the TCST bit have been updated before continuing with further processing.

This register is set to H’00 by an RTC software reset.

22.3.22 Second Capture Register (RSECCP0)/BCNT0 Capture Register (BCNT0CP0)

a. In calendar count mode:



x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	1 SEC[3:0]	x	R	1-Second Capture Capture value for the ones place of seconds
6 to 4	10 SEC[2:0]	x	R	10-Second Capture Capture value for the tens place of seconds
7	—	x	R	Reserved This bit is read as undefined.

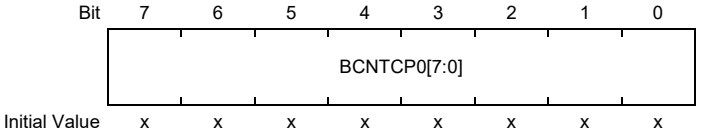
RSECCP0 is a read-only register that captures the RSECCNT value when a time capture event is detected.

The event detection times detected by the RTCTC0 pin are stored in the RSECCP0 register, respectively.

This register is set to H'00 by an RTC software reset.

Before reading from this register, be sure to stop the time capture event detection through the setting of the RTCCR0.TCCT[1:0] bits.

b. In binary count mode:



x: Undefined

BCNT0CP0 is a read-only register that captures the BCNT0 value when a time capture event is detected.

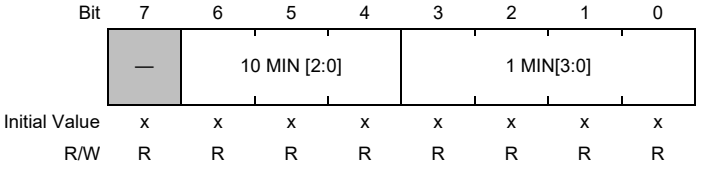
The event detection times detected by the RTCTC0 pin are stored in the BCNT0CP0 register, respectively.

This register is set to H'00 by an RTC software reset.

Before reading from this register, be sure to stop the time capture event detection through the setting of the RTCCR0.TCCT[1:0] bits.

22.3.23 Minute Capture Register (RMINCP0) /BCNT1 Capture Register (BCNT1CP0)

a. In calendar count mode:



x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	1 MIN[3:0]	x	R	1-Minute Capture Capture value for the ones place of minutes
6 to 4	10 MIN [2:0]	x	R	10-Minute Capture Capture value for the tens place of minutes
7	—	x	R	Reserved This bit is read as undefined.

RMINCP0 is a read-only register that captures the RMINCNT value when a time capture event is detected.

The event detection times detected by the RTCTC0 pin are stored in the RMINCP0 register, respectively.

This register is set to H’00 by an RTC software reset.

Before reading from this register, be sure to stop the time capture event detection through the setting of the RTCCR0.TCCT[1:0] bits.

b. In binary count mode:

x: Undefined

BCNT1CP0 is a read-only register that captures the BCNT1 value when a time capture event is detected.

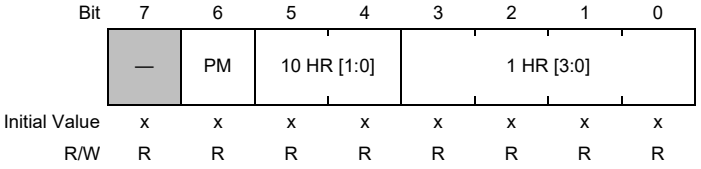
The event detection times detected by the RTCTC0 pin are stored in the BCNT1CP0 register, respectively.

This register is set to H’00 by an RTC software reset.

Before reading from this register, be sure to stop the time capture event detection through the setting of the RTCCR0.TCCT[1:0] bits.

22.3.24 Hour Capture Register (RHRCP0) /BCNT2 Capture Register (BCNT2CP0)

a. In calendar count mode:



x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	1 HR [3:0]	x	R	1-Hour Capture Capture value for the ones place of hours
5, 4	10 HR [1:0]	x	R	10-Hour Capture Capture value for the tens place of hours
6	PM	x	R	PM 0: a.m. 1: p.m.
7	—	x	R	Reserved This bit is read as undefined.

RHRCP0 is a read-only register that captures the RHRCNT value when a time capture event is detected.

The event detection times detected by the RTCTC0 pin are stored in the RHRCP0 register, respectively.

The PM bit is only enabled when the RCR2.HR24 bit is 0 (in 12-hour mode).

This register is set to H'00 by an RTC software reset.

Before reading from this register, be sure to stop the time capture event detection through the setting of the RTCCR0.TCCT[1:0] bits.

b. In binary count mode:

x: Undefined

BCNT2CP0 is a read-only register that captures the BCNT2 value when a time capture event is detected.

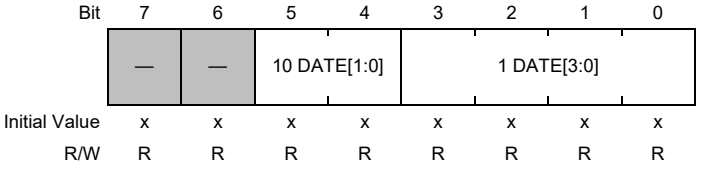
The event detection times detected by the RTCTC0 pin are stored in the BCNT2CP0 register, respectively.

This register is set to H'00 by an RTC software reset.

Before reading from this register, be sure to stop the time capture event detection through the setting of the RTCCR0.TCCT[1:0] bits.

22.3.25 Date Capture Register (RDAYCP0) /BCNT3 Capture Register (BCNT3CP0)

a. In calendar count mode:



x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	1 DATE[3:0]	x	R	1-Day Capture Capture value for the ones place of days
5, 4	10 DATE[1:0]	x	R	10-Day Capture Capture value for the tens place of days
7, 6	—	x	R	Reserved These bits are read as undefined.

RDAYCP0 is a read-only register that captures the RDAYCNT value when a time capture event is detected.

The event detection times detected by the RTCTC0 pin are stored in the RDAYCP0 register, respectively.

This register is set to H’00 by an RTC software reset.

Before reading from this register, be sure to stop the time capture event detection through the setting of the RTCCR0.TCCT[1:0] bits.

b. In binary count mode:

x: Undefined

BCNT3CP0 is a read-only register that captures the BCNT3 value when a time capture event is detected.

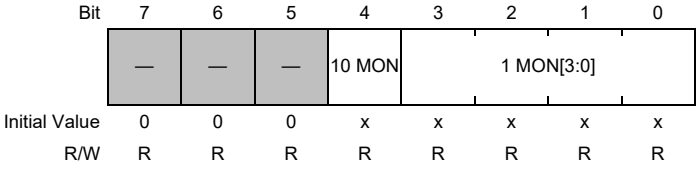
The event detection times detected by the RTCTC0 pin are stored in the BCNT3CP0 register, respectively.

This register is set to H’00 by an RTC software reset.

Before reading from this register, be sure to stop the time capture event detection through the setting of the RTCCR0.TCCT[1:0] bits.

22.3.26 Month Capture Register (RMONCP0)

a. In calendar count mode:



x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	1 MON[3:0]	x	R	1-Month Capture Capture value for the ones place of months
4	10 MON	x	R	10-Month Capture Capture value for the tens place of months
7 to 5	—	0	R	Reserved These bits are read as 0.

RMONCP0 is a read-only register that captures the RMONCNT value when a time capture event is detected.

The event detection times detected by the RTCTC0 pin are stored in the RMONCP0 register, respectively.

This register is set to H’00 by an RTC software reset.

Before reading from this register, be sure to stop the time capture event detection through the setting of the RTCCR0.TCCT[1:0] bits.

CAUTION

This register is not used in binary count mode.

22.4 Operation

22.4.1 Outline of Initial Settings of Registers after Power On

After the power is turned on, the initial settings for the clock setting, count mode setting, time error adjustment, time setting, alarm, interrupt, and time capture control register should be performed.

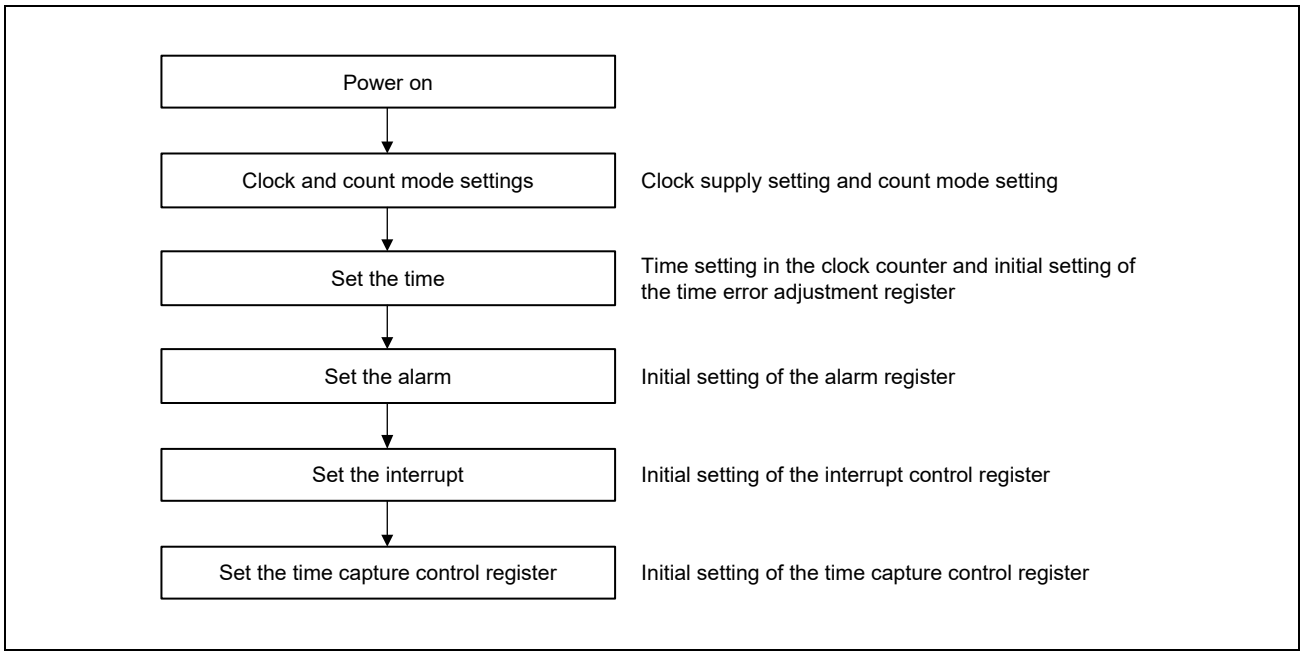
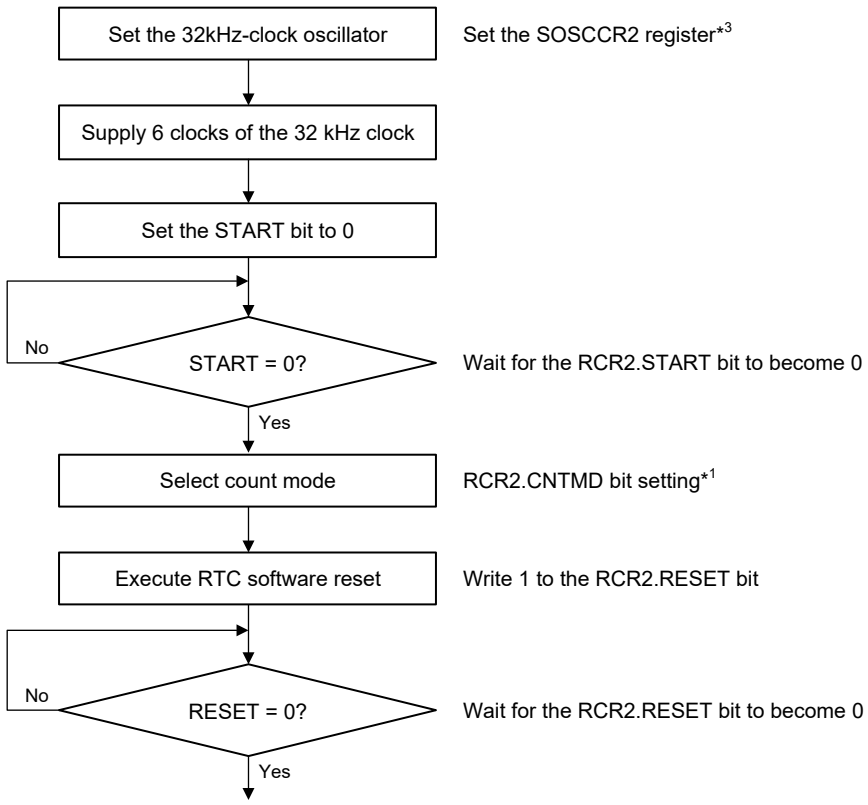


Figure 22.2 Outline of Initial Settings after Power On

22.4.2 Clock and Count Mode Setting Procedure

Figure 22.3 shows how to set the clock and the count mode.



Note 1. This step is not necessary if the count mode has been set concurrently with setting the START bit to 0. A value corresponding to the count mode setting must be written to the RCR2.CNTMD bit.

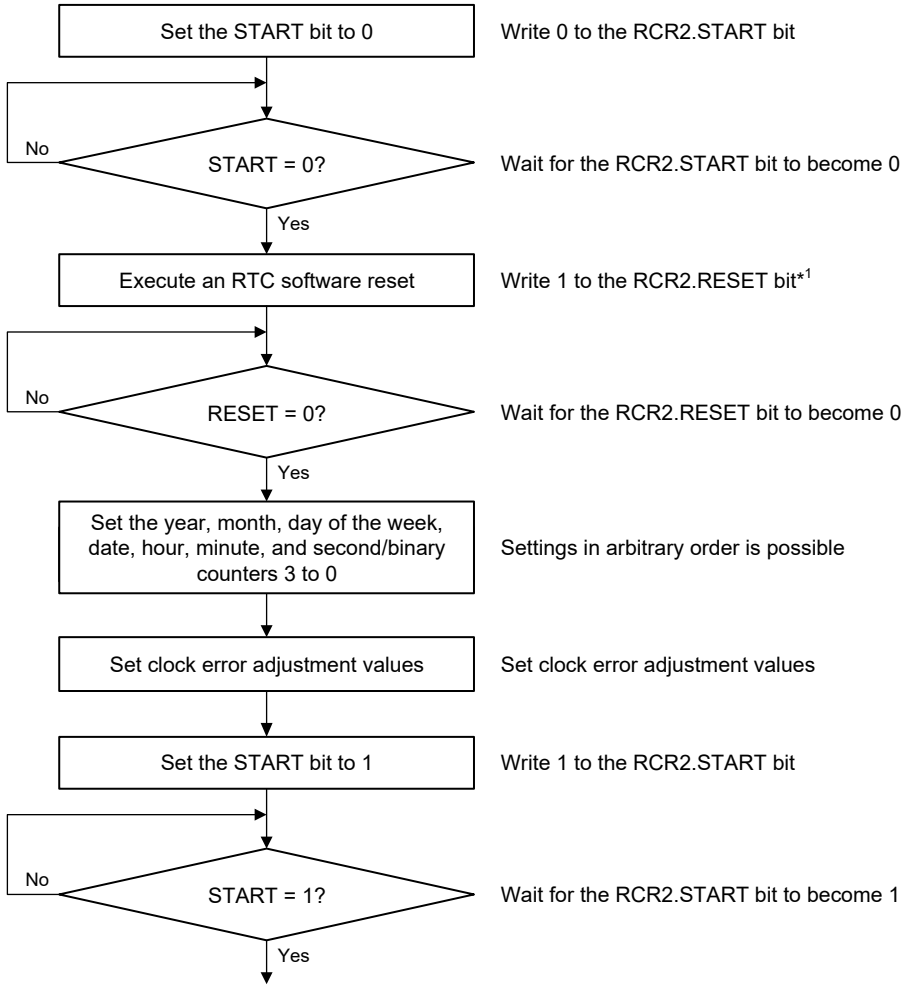
Note 2. Do not rewrite the RCR2 register by read-modify-write.

Note 3. For details, refer to **Section 42, Battery Backup Function (VBATTB)**.

Figure 22.3 Clock and Count Mode Setting Procedure

22.4.3 Setting the Time

Figure 22.4 shows how to set the time.



Note 1. This step is not necessary for the time-setting procedure because an RTC software reset is executed in the clock setting procedure of the initial settings for the power supply.

Figure 22.4 Setting the Time

22.4.4 30-Second Adjustment

Figure 22.5 shows how to execute 30-second adjustment.

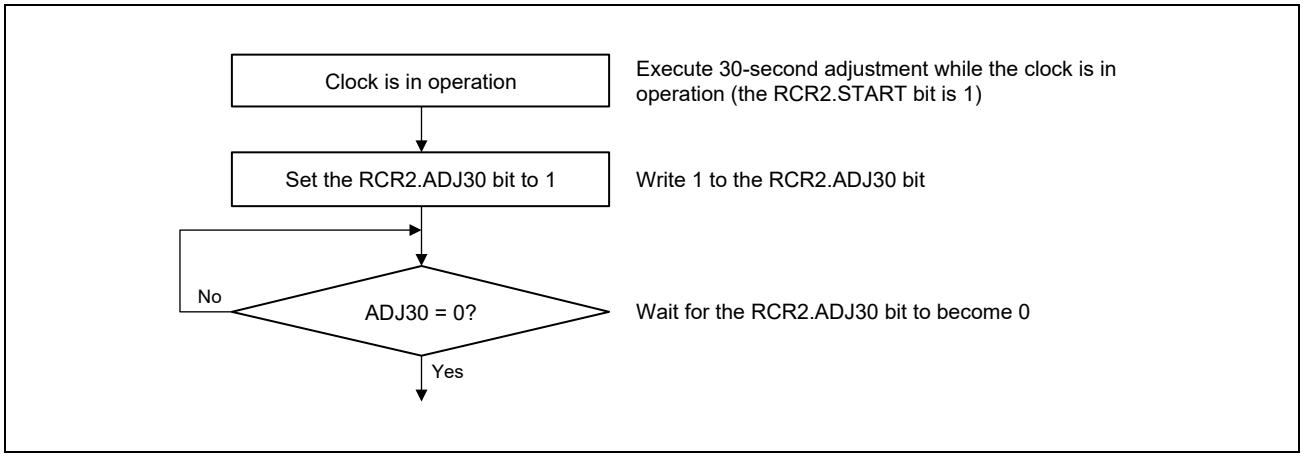


Figure 22.5 30-Second Adjustment

22.4.5 Reading 64-Hz Counter and Time

Figure 22.6 shows how to read the 64-Hz counter and time.

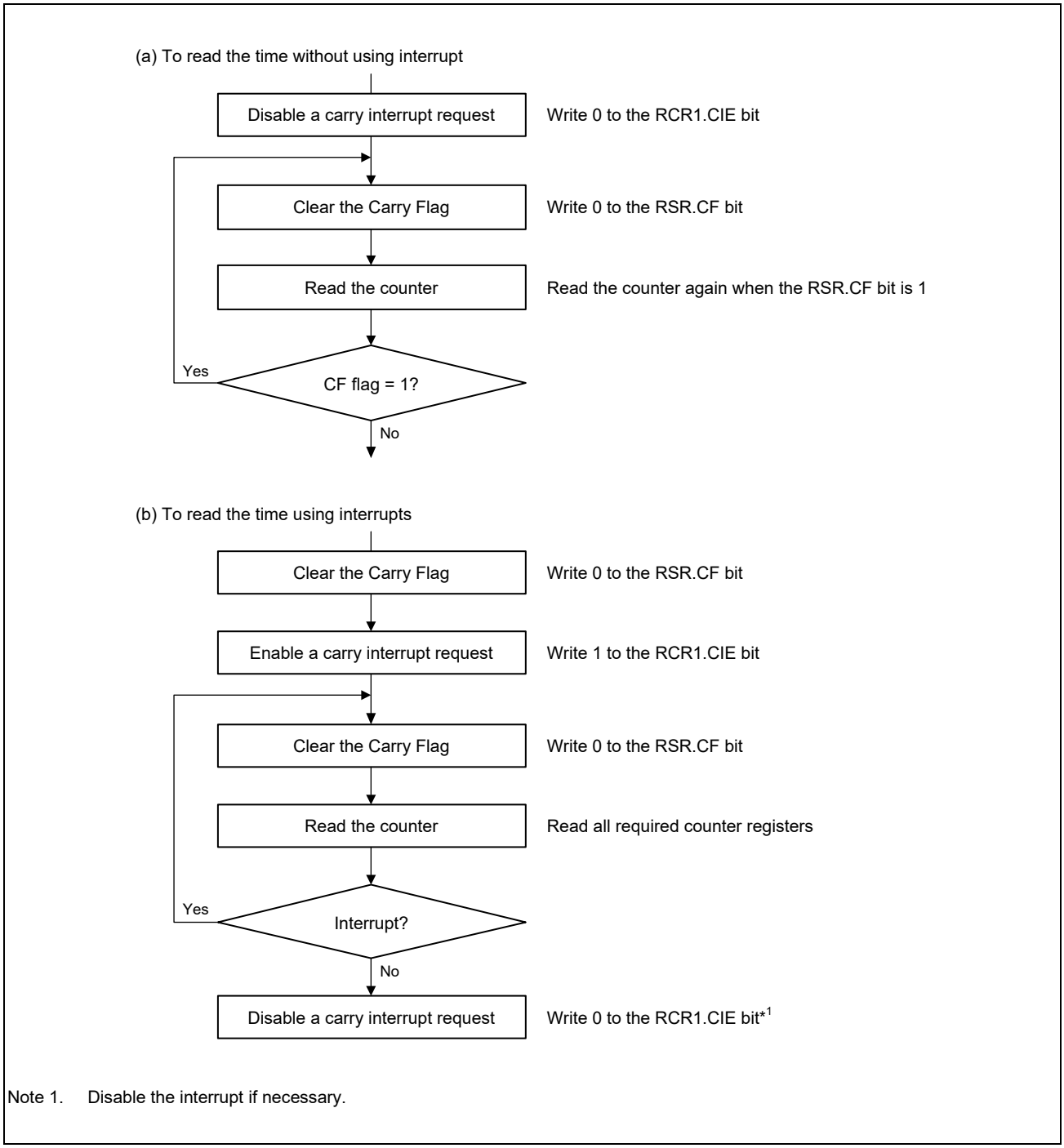


Figure 22.6 Reading Time

If a carry occurs while the 64-Hz counter and time are being read, the correct time will not be obtained, so they must be read again. The procedure for reading the time without using interrupts is shown in (a) in **Figure 22.6**, and the procedure using carry interrupts in (b). To keep the program simple, method (a) should be used in most cases.

22.4.6 Alarm Function

Figure 22.7 shows how to use the alarm function.

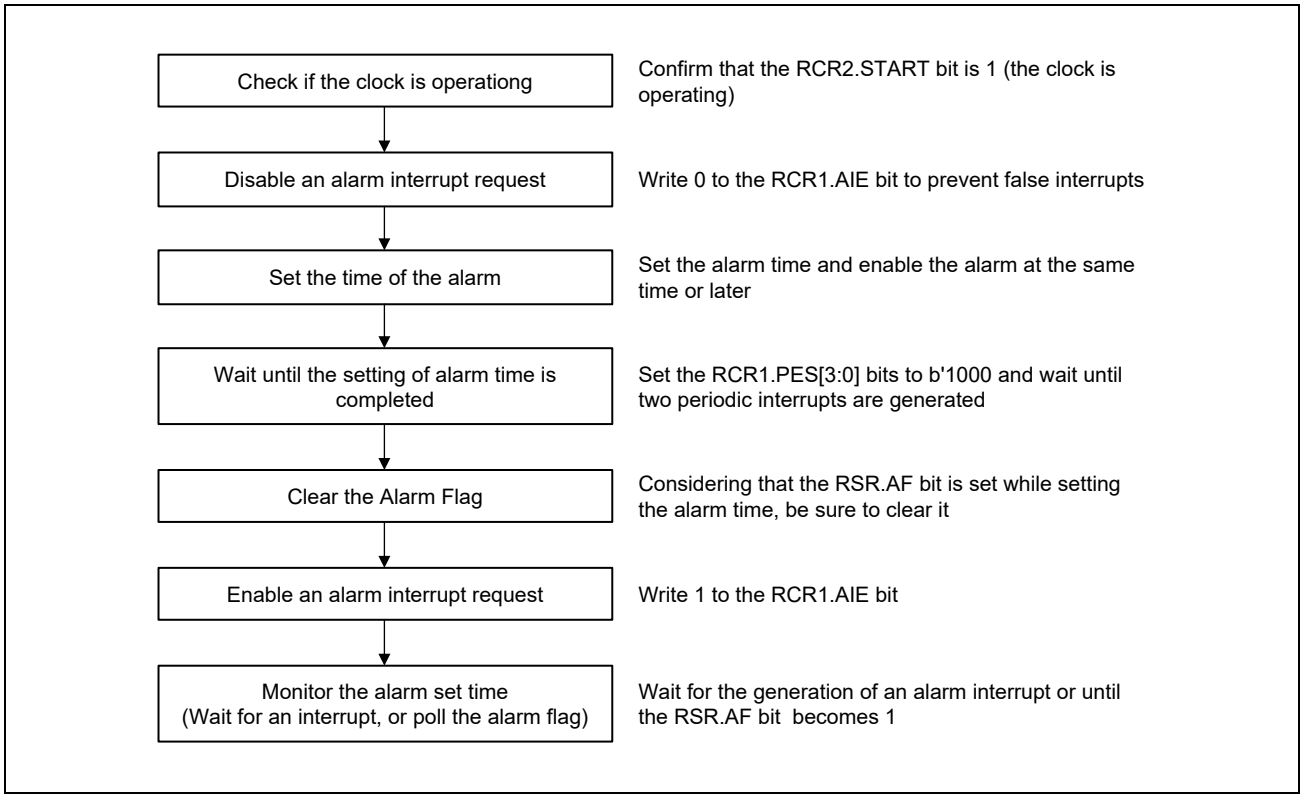


Figure 22.7 Using Alarm Function

In calendar count mode, an alarm can be generated by any one of year, month, date, day-of-week, hour, minute or second, or any combination of those. Write 1 to the ENB bit in the alarm registers involved in the alarm setting, and set the alarm time in the lower bits. Write 0 to the ENB bit in registers not involved in the alarm setting.

In binary count mode, an alarm can be generated in any bit combination of 32 bits. Write 1 to the ENB bit of the alarm enable register corresponding to the target bit of the alarm, and set the alarm time to the alarm register. For bits that are not target of the alarm, write 0 to the ENB bit of the alarm enable register.

When the counter and the alarm time match, the Alarm flag in RSR is set to 1. Alarm detection can be confirmed by reading this bit, but an interrupt should be used in most cases. If 1 has been set in the AIE bit in RCR1 register, an alarm interrupt is generated in the event of alarm, enabling the alarm to be detected.

Writing 0 sets the Alarm flag in RSR to 0.

22.4.7 Time Error Adjustment Function

The time error adjustment function is used to correct errors (running fast or slow) in the time due to the precision of oscillation by the 32kHz-clock. Since 32,768 cycles of the 32kHz-clock constitute 1 second of operation, the clock runs fast if the 32kHz-clock frequency is high and slow if the 32kHz-clock frequency is low. This function can be used to correct errors due to the clock running fast or slow.

Two types of time error adjustment functions are provided: automatic adjustment and adjustment by software.

Use the RCR2.AADJE bit to select automatic adjustment or adjustment by software.

22.4.7.1 Automatic Adjustment

Enable automatic adjustment by setting the RCR2.AADJE bit to 1.

Automatic adjustment is the addition or subtraction of the value counted by the prescaler to or from the value in the RADJ register every time the adjustment period selected by the RCR2.AADJE bit elapses.

Examples are shown below.

[Example 1] 32kHz-clock running at 32.769 kHz

- Adjustment procedure:

  When the 32kHz-clock is running at 32.769 kHz, 1 second elapses every 32,769 clock cycles. The RTC is meant to run at 32,768 clock cycles, so the clock runs fast by one clock cycle every second. The time on the clock is fast by 60 clock cycles per minute, so adjustment can take the form of setting the clock back by 60 cycles every minute.

- Register settings (when RCR2.CNTMD = 0):
  - RCR2.AADJP = 0 (adjustment every minute)
  - RADJ.PMADJ[1:0] = b’10 (adjustment is performed by the subtraction from the prescaler.)
  - RADJ.ADJ[5:0] = 60 (H’3C)

[Example 2] 32kHz-clock running at 32.766 kHz

- Adjustment procedure:

  When the 32kHz-clock is running at 32.766 kHz, 1 second elapses every 32,766 clock cycles. The RTC is meant to run at 32,768 clock cycles, so the clock runs slow by two clock cycles every second. The time on the clock is slow by 20 clock cycles every 10 seconds, so adjustment can take the form of setting the clock forward by 20 cycles every 10 seconds.

- Register settings (when RCR2.CNTMD = 0):
  - RCR2.AADJP = 1 (adjustment every 10 seconds)
  - RADJ.PMADJ[1:0] = b’01 (adjustment is performed by the addition to the prescaler.)
  - RADJ.ADJ[5:0] = 20 (H’14)

[Example 3] 32kHz-clock running at 32.764 kHz

- Adjustment procedure:

  When the 32kHz-clock is running at 32.764 kHz, 1 second elapses every 32,764 clock cycles. The RTC is meant to run at 32,768 clock cycles, so the clock runs slow by four clock cycles every second. The time on the clock is slow by 32 clock cycles every 8 seconds, so adjustment can take the form of setting the clock forward by 32 cycles every 8 seconds.

- Register settings (when RCR2.CNTMD = 1):
  - RCR2.AADJP = 1 (adjustment every 8 seconds)
  - RADJ.PMADJ[1:0] = b'01 (adjustment is performed by the addition to the prescaler.)
  - RADJ.ADJ[5:0] = 32 (H'20)

22.4.7.2 Adjustment by Software

Enable adjustment by software by setting the RCR2.AADJE bit to 0.

Adjustment by software is the addition or subtraction of the value counted by the prescaler to or from the value in the RADJ register at the time of execution of an instruction for writing to the RADJ register.

An example is shown below.

[Example 1] 32kHz-clock running at 32.769 kHz

- Adjustment procedure:

  When the 32kHz-clock is running at 32.769 kHz, 1 second elapses every 32,769 clock cycles. The RTC is meant to run at 32,768 clock cycles, so the clock runs fast by one clock cycle every second. The time on the clock is fast by one clock cycle per second, so adjustment can take the form of setting the clock back by one cycle every second.

- Register settings:
  - RADJ.PMADJ[1:0] = b'10 (adjustment is performed by the subtraction from the prescaler.)
  - RADJ.ADJ[5:0] = 1 (H'01)
    This is written to the RADJ register once per 1-second interrupt.

22.4.7.3 Procedure for Changing the Mode of Adjustment

When changing the mode of adjustment, change the value of the AADJE bit in RCR2 after setting the RADJ.PMADJ[1:0] bits to b'00 (adjustment is not performed).

Changing from adjustment by software to automatic adjustment:

(1) Set the RADJ.PMADJ[1:0] bits to b'00 (adjustment is not performed).

(2) Set the RCR2.AADJE bit to 1 (automatic adjustment is enabled).

(3) Use the RCR2.AADJP bit to select the period of adjustment.

(4) In RADJ, set the PMADJ[1:0] bits for addition or subtraction and the ADJ[5:0] bits to the value for use in time error adjustment.

Changing from adjustment by software to automatic adjustment:

(1) Set the RADJ.PMADJ[1:0] bits to b'00 (adjustment is not performed).

(2) Set the RCR2.AADJE bit to 0 (adjustment by software is enabled).

(3) Proceed with adjustment by setting the RADJ.PMADJ[1:0] bits for addition or subtraction and the RADJ.ADJ[5:0] bits to the value for use in time error adjustment at the desired time. After that, the time is adjusted every time a value is written to the RADJ register.

22.4.7.4 Procedure for Stopping Adjustment

Stop adjustment by setting the RADJ.PMADJ[1:0] bits to b'00 (adjustment is not performed).

22.4.8 Time Capture Function

The RTC is capable of storing the month, date, hour, minute and second/binary counters 3 to 0 by detecting an edge of a signal on a time capture event input pin.

The noise filter transmits the matched level to the inside of the RTC when the level of the pin sampled according to the set sampling period matches 3 times, and the level inside the RTC is maintained until the level of the sampled pin matches 3 times.

The noise filter can be switched on or off for each of the time capture event input pins. Operation when the noise filter is off is shown in **Figure 22.8** and operation when the noise filter is on is shown in **Figure 22.9**.

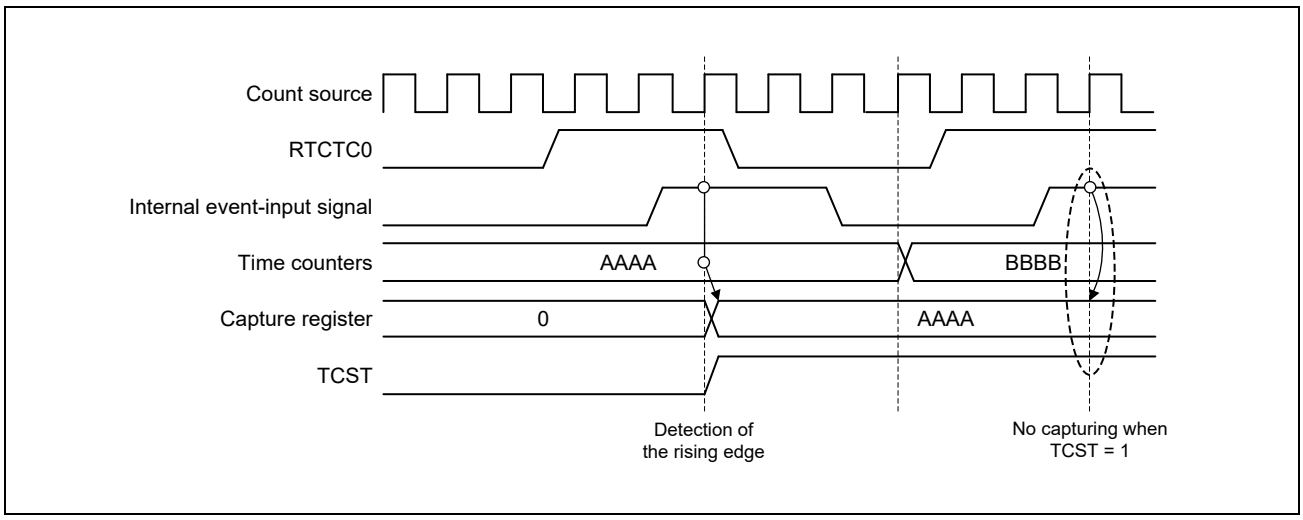


Figure 22.8 Timing of a Time Capture Operation (with the Filter Off)

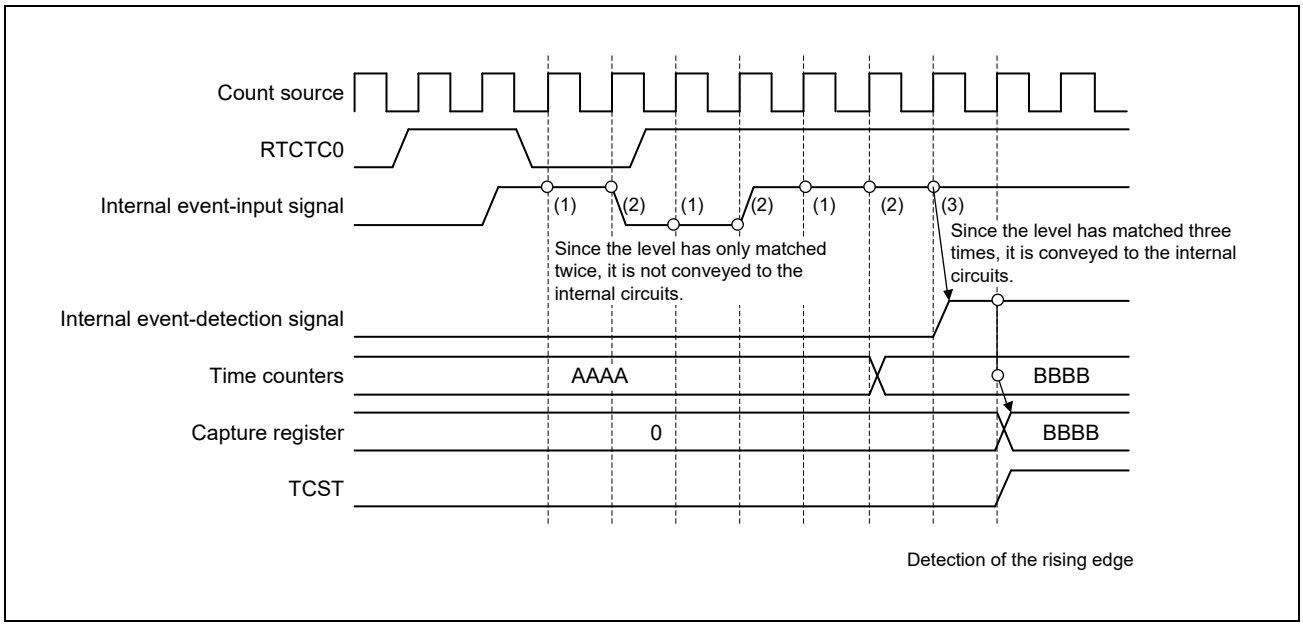


Figure 22.9 Timing of a Time Capture Operation (with the Filter On)

22.5 Interrupt Sources

There are three interrupt sources in the realtime clock. **Table 22.4** lists interrupt sources for the RTC.

Table 22.4 RTC Interrupt Sources

Name	Interrupt Sources	Interrupt Status Flag
ALM	Alarm interrupt	RSR.AF
PRD	Periodic interrupt	RSR.PF
CUP	Carry interrupt	RSR.CF

22.5.1 Alarm Interrupt (ALM)

This interrupt is generated according to the result of comparison between the alarm registers and realtime clock counters (for details, refer to **Section 22.4.6, Alarm Function**).

Since there is a possibility that the interrupt flag may be set to 1 when the settings of the alarm registers match the clock counters, wait for the alarm time settings to be confirmed and set the Alarm Flag in RSR to 0 again after modifying values of the alarm registers. Once the interrupt flag for the alarm interrupt has been set to 1 and the state has returned to non-matching of the alarm registers and clock counters, the flag will not be set again until there is a further match or the values of the alarm registers are modified again.

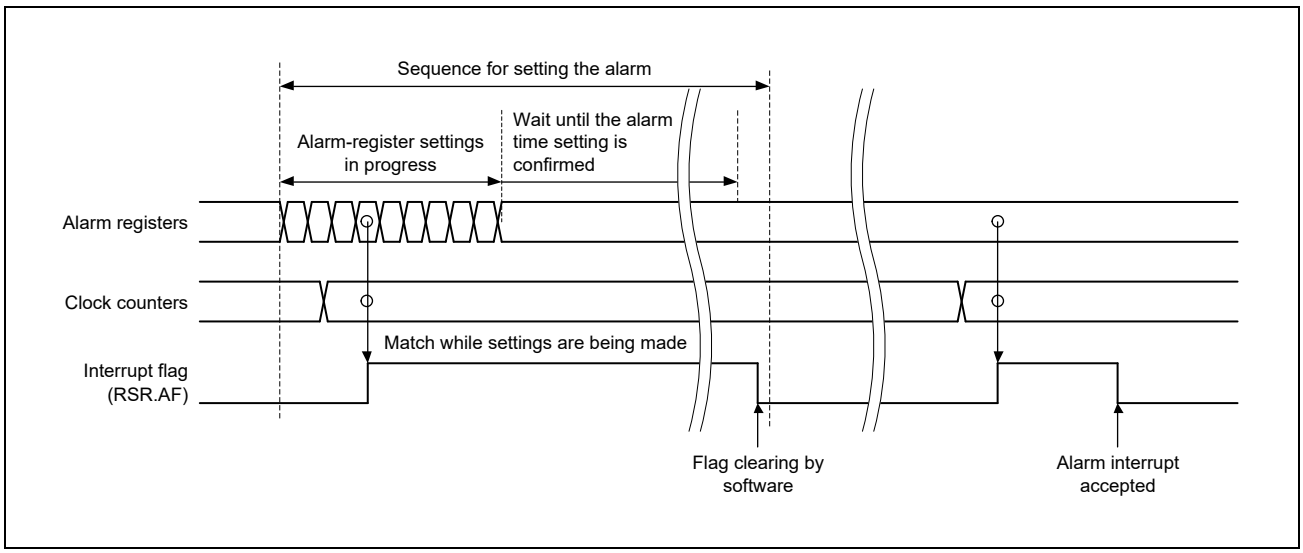


Figure 22.10 Timing Chart for the Alarm Interrupt (ALM)

22.5.2 Periodic Interrupt (PRD)

This interrupt is generated at intervals of 2 seconds, 1 second, 1/2 second, 1/4 second, 1/8 second, 1/16 second, 1/32 second, 1/64 second, 1/128 second, or 1/256 second. The interrupt interval can be selected through the RCR1.PES[3:0] bits.

22.5.3 Carry Interrupt (CUP)

This interrupt is asserted when a carry to the seconds counter/binary counter 0 occurs, or when a carry from the prescaler to R64CNT occurs on a read access to the 64-Hz counter.

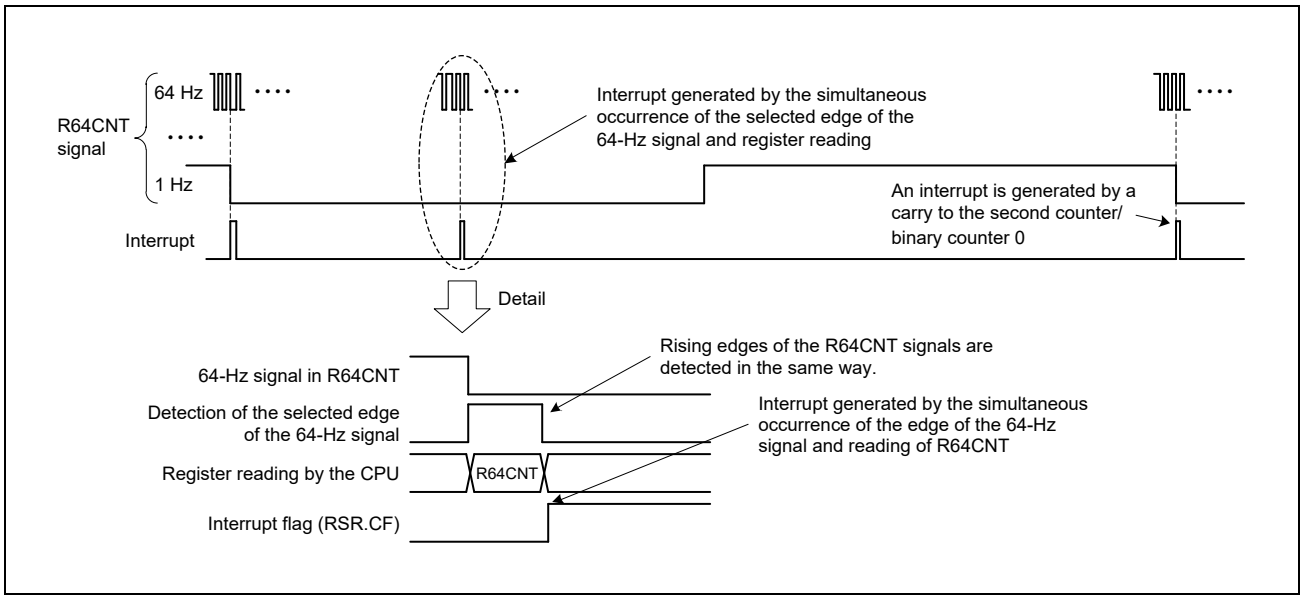


Figure 22.11 Carry Interrupt (CUP) Timing Chart

22.6 Usage Notes

22.6.1 Register Writing during Counting

The following registers should not be written to during counting (while the RCR2.START bit = 1).

RSECCNT/BCNT0, RMINCNT/BCNT1, RHRCNT/BCNT2, RDAYCNT, RWKCNT/BCNT3, RMONCNT, RYRCNT, RCR1.RTCOS, RCR2.RTCOE, RCR2.HR24

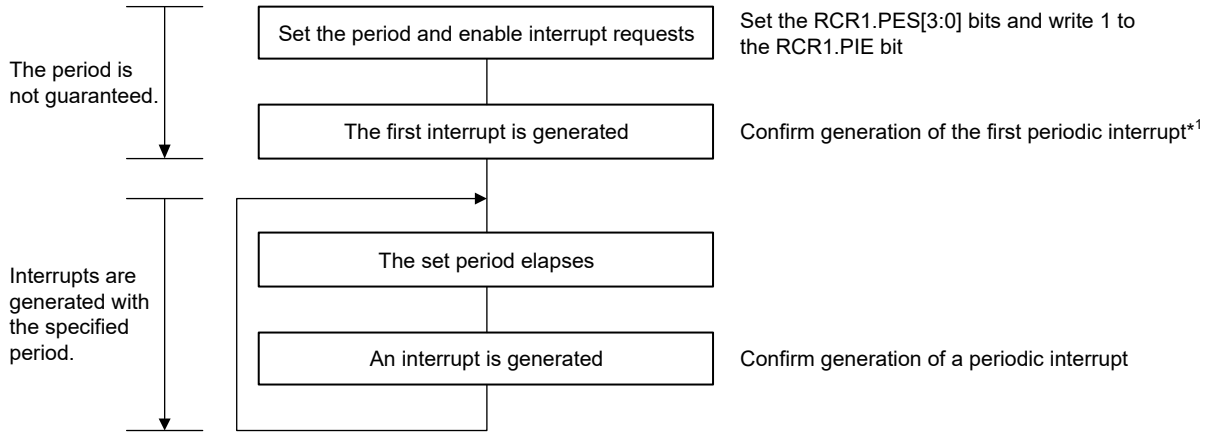
The counter must be stopped before writing to any of the above registers.

22.6.2 Use of Periodic Interrupts

The procedure for using periodic interrupts is shown in **Figure 22.12**.

The generation and period of the periodic interrupt can be changed by the setting of the RCR1.PES[3:0] bits. However, since the prescaler, R64CNT, and RSECCNT/BCNT0 are used to generate interrupts, the interrupt period is not guaranteed immediately after setting of the RCR1.PES[3:0] bits.

Furthermore, stopping/restarting or resetting counter operation, reset by RTC software, and the 30-second adjustment by changing the RCR2 value affects the interrupt period. When the time error adjustment function is used, the interrupt generation period after adjustment is added or subtracted according to the adjustment value.



Note 1. When an interrupt generation period is changed while the periodic interrupt is used, an interrupt may be generated at the completion of the setting. If the interrupt is generated immediately after the setting, the period is not guaranteed for two interrupts including the current interrupt.

Figure 22.12 Using Periodic Interrupt Function

22.6.3 Transitions to Low Power Consumption Modes after Setting Registers

A transition to a low power consumption state (battery backup) during writing to or updating of an RTC register might destroy the register's value. After setting a register, be sure to confirm that the setting is in place before initiating a transition to a low power consumption state.

22.6.4 Notes on Writing to and Reading from Registers

- When reading a counter register such as the second counter/binary counter 0 after having written to the counter register, follow the procedure in **Section 22.4.5, Reading 64-Hz Counter and Time**.
- After written to the count registers, alarm registers, year alarm enable register, bits RCR2.AADJE, AADJP, and HR24 register, the read value will be reflected from the read after 3 empty reads.
- The values written to the RSR register, RCR1.CIE, RTCOS, and RCR2.RTCOE bits can be read immediately after writing.
- To read the value from the timer counter after return from a reset, low power consumption state, wait for 1/128 second while the clock is operating (RCR2.START bit = 1).

  * Low power consumption states refer to the following modes: battery backup function
- After a reset is generated, write to the RTC register when six cycles of the count source have elapsed.

22.6.5 Changing the Count Mode

When changing the count mode (calendar/binary), set the RCR2.START bit to 0, stop counting operation, then start again from the initial setting. For details on initial setting, refer to **Section 22.4.1, Outline of Initial Settings of Registers after Power On**.

23. Serial Communications Interface with FIFO (SCIFA)

This LSI has five channels of serial communication interface (SCIFA) with FIFO that support both asynchronous and clock synchronous serial communication. The SCIFA has 16-stage FIFO buffers for transmission and reception, respectively, for each channel that enable this LSI to perform efficient high-speed continuous communication.

23.1 Overview

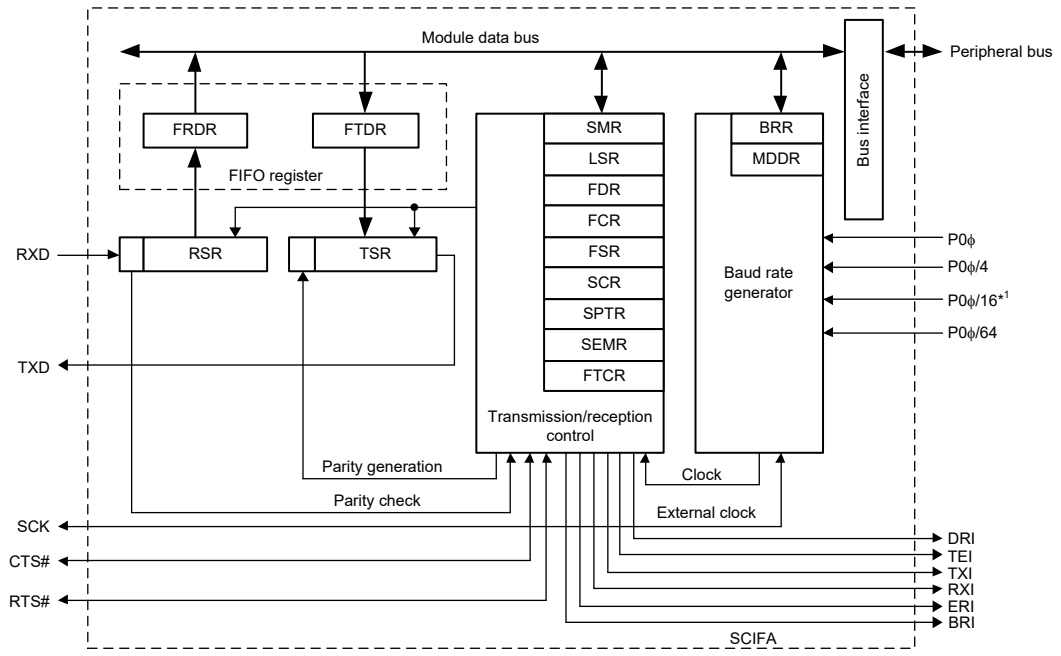
Table 23.1 lists the specifications of the SCIFA.

Table 23.1 Specifications of SCIFA

Item		Description
Channel		6 channels (ch 0, 1, 2, 3, 4, 5)
Serial communication method		Asynchronous communication mode and clock synchronous communication mode
Transfer speed		Selectable bit rate with an on-chip baud rate generator
Full duplex communication		Transmitting section: realizes continuous data transmission using 16-stage FIFO buffer Receiving section: realizes continuous data reception using 16-stage FIFO buffer
Data transmission		Selectable either LSB-first or MSB-first transfer
Interrupt source		The following six sources: • Transmit-end (TEIF) • Transmit-FIFO-data-empty (TXIF) • Receive-FIFO-data-full (RXIF) • Receive-data-ready (DRIF)*1 • Receive-error (ERIF) • Break detection or overrun (BRIF)
Asynchronous communication mode	Character length	7 or 8 bits
	Transmission stop bit length	1 or 2 bits
	Parity	Even, odd, or none
	Receive error detection	Parity, overrun, and framing errors
	Hardware flow control	Controls data transmission and reception using the CTS# and RTS# pins. Not applicable to channel 3 (SCIF3), channel 4 (SCIF4) and channel 5 (SCIF5)
	Break detection	Break signal detection function by hardware.
	Clock source	Selectable from internal or external clock
	Noise cancellation	Incorporates a digital noise filter in the RXD pin input path.
Clock synchronous communication mode	Character length	8 bits
	Receive error detection	Detects an overrun error as a receive error.
	Clock source	Selectable either internal or external clock
Bit rate modulation		Enables errors to be decreased by correcting the output of the on-chip baud rate generator.

Note 1. Effective only for asynchronous communication mode

Figure 23.1 shows a block diagram of the SCIFA.



RSR: Receive shift register
FRDR: Receive FIFO data register
TSR: Transmit shift register
FTDR: Transmit FIFO data register
SMR: Serial mode register
SCR: Serial control register
FSR: Serial status register
MDDR: Modulation duty register
BRR: Bit rate register
SPTR: Serial port register
FCR: FIFO control register
FDR: FIFO data count register
LSR: Line status register
SEMR: Serial extended mode register
FTCR: FIFO trigger control register

Note 1. The Pϕ/16 clock is not input in the case of the channel 0 interface.

Figure 23.1 Block Diagram of SCIFA

Table 23.2 lists the input/output pins of the SCIFA.

Table 23.2 Pin Configuration of the SCIFA

Channel	Item	Pin Name	I/O	Function
0 to 5 (n = 0-5)	Serial clock pin	SCIFn_SCK	I/O	Transmission/reception clock input/output, general output
	Receive data pin	SCIFn_RXD	Input	Receive data input
	Transmit data pin	SCIFn_TXD	Output	Transmit data output
0 to 2 (n = 0-2)	Transmission/reception start control pin	SCIFn_CTS#	I/O	Input for hardware flow control (transmission enable signal) / general output
		SCIFn_RTS#	Output	Output for hardware flow control (transmission request signal) / general output

Note: Channels of each pin is omitted.

23.2 Register Descriptions

Table 23.3 Base Address Table

Address Space	Channel 0	Channel 1	Channel 2	Channel 3	Channel 4	Channel 5
Cortex-A55 Address Space	H'0_1004_B800	H'0_1004_BC00	H'0_1004_C000	H'0_1004_C400	H'0_1004_C800	H'0_1004_E000
Cortex-M33/Cortex-M33_FPU Address Space Secure	H'4004_B800	H'4004_BC00	H'4004_C000	H'4004_C400	H'4004_C800	H'4004_E000
Cortex-M33/Cortex-M33_FPU Address Space Non-Secure	H'5004_B800	H'5004_BC00	H'5004_C000	H'5004_C400	H'5004_C800	H'5004_E000

Note: Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 23.4 List of Registers

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
Serial mode register	SMR	R/W	H'0000	H'00	16
Bit rate register	BRR	R/W	H'FF	H'02	8
Modulation duty register	MDDR	R/W	H'FF	H'02	8
Serial control register	SCR	R/W	H'0000	H'04	16
Transmit FIFO data register	FTDR	W	Undefined	H'06	8
Serial status register	FSR	R/W	H'0020	H'08	16
Receive FIFO data register	FRDR	R	Undefined	H'0A	8
FIFO control register	FCR	R/W	H'0000	H'0C	16
FIFO data count register	FDR	R	H'0000	H'0E	16
Serial port register	SPTR	R/W	H'00xx	H'10	16
Line status register	LSR	R/W	H'0000	H'12	16
Serial extended mode register	SEMR	R/W	H'00	H'14	8
FIFO trigger control register	FTCR	R/W	H'1F1F	H'16	16

Note: BRR and MDDR are located in the same address. Setting the MDDRS bit of the SEMR register switches these registers.

23.2.1 Receive Shift Register (RSR)

The RSR register receives serial data and temporally stores the data. The SCIFA stores the serial data input via the RXD pin into the RSR register and converts the data to the parallel form. When one byte of data has been received, it is automatically transferred to the receive FIFO data register (FRDR).

The CPU cannot read from or write to the RSR register directly.

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0

23.2.2 Receive FIFO Data Register (FRDR)

The FRDR register is a 8-bit, 16-stage FIFO register that stores the received serial data. When the SCIFA receives one byte of serial data, it transfers the received data from the receive shift register (RSR) to the FRDR register and completes the receive operation. Continuous reception is possible until the received 16 bytes of data are stored. If the FRDR register is read when there is no received data in the FRDR register, an undefined value is read.

When the FRDR register is full of received data, subsequently received serial data is lost.

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0

23.2.3 Transmit Shift Register (TSR)

The SCIFA transfers the transmit data from the transmit FIFO data register (FTDR) to the TSR register, and then transmits the data serially to the TXD pin. After transmitting one byte of data, the SCIFA automatically transfers the next transmit data from the FTDR register into the TSR register and starts transmission.

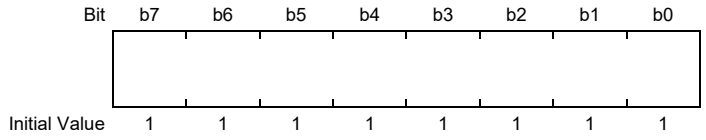
The CPU cannot read from or write to the TSR register directly.

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1

23.2.4 Transmit FIFO Data Register (FTDR)

The FTDR register is a 8-bit, 16-stage FIFO register that stores serial transmission data. When the SCIFA detects that the transmit shift register (TSR) is empty, it transmits data written in the FTDR register to the TSR register and starts serial transmission. Continuous serial transmission is executed until there is no transmit data left in the FTDR register. Writing the transmit data to the FTDR register should be done when a transmit data empty interrupt (TXI) request is generated. When the FTDR register becomes full of transmit data (16 bytes), no more data can be written. Even if new data is written, the data is ignored.

CPU can read from the FTDR register but cannot write to it.



23.2.5 Serial Mode Register (SMR)

The SMR register specifies the SCIFA serial communication format and selects the clock source for the baud rate generator.

The CPU can always read and write to the SMR register.

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CM	CHR	PE	PM	STOP	—	CKS[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	CKS[1:0]	All 0	R/W	Clock Select b1 b0 0 0: 1 × Pϕ*1 0 1: 1/4 × Pϕ*1 1 0: 1/16 × Pϕ*1,*2 1 1: 1/64 × Pϕ*1
b2	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b3	STOP	0	R/W	Stop Bit Length 0: One stop bit 1: Two stop bits
b4	PM	0	R/W	Parity Mode 0: Even parity 1: Odd parity
b5	PE	0	R/W	Parity Enable 0: Parity bit addition or check is disabled. 1: Parity bit addition or check is enabled.
b6	CHR	0	R/W	Character Length 0: 8-bit data 1: 7-bit data*3
b7	CM	0	R/W	Communication Mode 0: Asynchronous mode 1: Clock synchronous mode
b15 to b8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Pϕ: SCIFn_CLK_PCK (n = 0 to 5)

Note 2. This setting is not applicable to channel 0. It is only applicable to channels 1 to 5.

Note 3. When 7-bit data is selected, the MSB (bit 7) of the transmit FIFO data register is not transmitted.

CKS[1:0] Bits (Clock Select)

Select an internal clock source for the on-chip baud rate generator. For further information on the clock source, bit rate register settings, and baud rates, see **Section 23.2.8, Bit Rate Register (BRR)**.

STOP Bit (Stop Bit Length)

Selects one bit or two bits as the stop bit length in asynchronous mode. This setting is used only in asynchronous mode. It is ignored in clock synchronous mode because no stop bits are added. When receiving, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1, it is treated as a stop bit, but if the second stop bit is 0, it is treated as the start bit of the next incoming character.

Note: When transmitting with one stop bit, a single 1 bit (stop bit) is added at the end of each transmission character.

Note: When transmitting with two stop bits, two 1 bits (stop bits) are added at the end of each transmission character.

PM Bit (Parity Mode)

Selects either the even or odd parity check. The setting of this bit is effective only when the parity enable (PE) bit of this register is set to 1 in asynchronous mode. The setting of this bit is ignored in clock synchronous mode, or when parity addition/check is disabled in asynchronous mode.

Note: If even parity is selected, the parity bit is added to data to be transmitted to make the total number of 1s even in the transmission character and parity bit combined. When receiving, the SCIFA verifies that the total number of 1s in the received character and parity bit combined is even.

Note: If odd parity is selected, the parity bit is added to data to be transmitted to make the total number of 1s odd in the transmission character and parity bit combined. When receiving, the SCIFA verifies that the total number of 1s in the received character and parity bit combined is odd.

PE Bit (Parity Enable)

Selects whether to add a parity bit on data transmission and whether to enable/disable the parity check on data reception in asynchronous mode. In clock synchronous mode, a parity bit is neither added nor checked, regardless of the setting of this bit.

Note: When this bit is set to 1, an even or odd parity bit specified in the PM bit is added to data to be transmitted. The SCIFA verifies whether the parity bit of the received data is even or odd as specified in the PM bit when receiving.

CHR Bit (Character Length)

Selects 7- or 8-bit data length in asynchronous mode. In clock synchronous mode, the data length is always 8 bits, regardless of the CHR setting.

CM Bit (Communication Mode)

Selects whether the SCIFA operates in asynchronous or clock synchronous mode.

23.2.6 Serial Control Register (SCR)

The SCR register enables or disables the SCIFA transmission/reception and interrupt requests, and selects the transmit/receive clock source. The CPU can always read from and write to the SCR register.

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	TIE	RIE	TE	RE	REIE	TEIE	CKE[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	CKE[1:0]	All 0	R/W	Clock Enable In asynchronous mode: b1 b0 0 0: Internal clock or SCK pin is used for input pin (input signal is ignored). The SCK pin state depends on the SCKIO and SCKDT bits in SPTR. 0 1: Internal clock or SCK pin is used for clock output (The output clock frequency is 16 or 8 times of the bit rate). 1 0: External clock or SCK pin is used for clock input (The input clock frequency is 16 or 8 times of the bit rate). 1 1: Setting prohibited In clock synchronous mode: b1 b0 0 0: Internal clock or SCK pin is used for synchronous clock output. 0 1: Internal clock or SCK pin is used for synchronous clock output. 1 0: External clock or SCK pin is used for synchronous clock input. 1 1: Setting prohibited
b2	TEIE*1	0	R/W	Transmit End Interrupt Enable 0: Transmit end interrupt (TEI) request is disabled. 1: Transmit end interrupt (TEI) request is enabled.
b3	REIE	0	R/W	Receive Error Interrupt Enable 0: Receive-error interrupt (ERI) and break interrupt (BRI) requests are disabled. 1: Receive-error interrupt (ERI) and break interrupt (BRI) requests are enabled.
b4	RE	0	R/W	Receive Enable 0: Data reception is disabled. 1: Data reception is enabled.
b5	TE	0	R/W	Transmit Enable 0: Data transmission is disabled. 1: Data transmission is enabled.
b6	RIE	0	R/W	Receive Interrupt Enable 0: Receive-FIFO-data-full interrupt (RXI), receive-data ready interrupt (DRI), receive-error interrupt (ERI), and break interrupt (BRI) requests are disabled. 1: Receive-FIFO-data-full interrupt (RXI), receive-data ready interrupt (DRI), receive-error interrupt (ERI), and break interrupt (BRI) requests are enabled.
b7	TIE	0	R/W	Transmit Interrupt Enable 0: Transmit-FIFO-data-empty interrupt request (TXI) is disabled. 1: Transmit-FIFO-data-empty interrupt request (TXI) is enabled.
b15 to b8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. TEI interrupt requests can be cleared by reading 1 from TEND flag, and then clearing the setting to 0, or by setting the TEIE bit to 0.

CKE[1:0] Bits (Clock Enable)

Select the SCIFA clock source and enable or disable clock output from the SCK pin. Depending on the settings of these bits, the SCK pin can be used for serial clock output or serial clock input. If the SCK pin is set for the synchronous clock output in the clock synchronous mode, set the CM bit in the SMR register to 1, and then set the CKE[1:0] bits. The settings of the CKE[1:0] bits are listed in **Table 23.17**.

REIE Bit (Receive Error Interrupt Enable)

Specifies whether to enable or disable a receive-error interrupt (ERI) request and a break interrupt (BRI) request. The setting of this bit is only valid when the RIE bit is set to 0.

Note: ERI interrupt requests can be cleared by reading 1 from the ER bit in the FSR register, and then clearing the setting to 0, or by clearing both the RIE and REIE bits in this register to 0. BRI interrupt requests can be cleared by reading 1 from the BRK bit in the FSR register, or from the ORER flag in the LSR register, and then clearing the setting to 0, or by clearing both the RIE and REIE bits in this register to 0.

RE Bit (Receive Enable)

Specifies whether to enable or disable the serial data reception.

Note: Setting this bit to 0 does not affect the receive flags (DR, ER, BRK, RDF, FER, and PER in the FSR register, and ORER in the LSR register). These flags retain their previous values.

Note: Serial reception starts when a start bit is detected in asynchronous mode, or a synchronous clock input is detected in clock synchronous mode. Before setting this bit to 1, be sure to set the serial mode register (SMR) and the FIFO control register (FCR) to select the receive format and reset the receive FIFO.

TE Bit (Transmit Enable)

Specifies whether to enable or disable the serial data transmission.

Note: Serial transmission starts after writing of data to be transmitted into the FTDR register under this condition. Before setting this bit to 1, be sure to set the serial mode register (SMR) and the FIFO control register (FCR) to select the transmit format and reset the transmit FIFO.

RIE Bit (Receive Interrupt Enable)

Specifies whether to enable or disable a receive-FIFO-data-full (RXI) interrupt request when the RDF flag in the serial status register (FSR) is set to 1, a receive-data ready (DRI) interrupt request when the DR flag in the FSR register is set to 1, a receive-error (ERI) interrupt request when the ER flag in the FSR register is set to 1, and a break (BRI) interrupt request when the BRK flag in the FSR register or the ORER flag in the line status register (LSR) is set to 1.

Note: RXI interrupt requests can be cleared by reading 1 from the DR or RDF flag in the FSR register, then clearing the flag to 0, or by clearing the RIE bit to 0. DRI interrupt requests can be cleared by reading 1 from the DR flag in the FSR register, and then clearing the setting to 0, or by clearing the RIE bit in this register to 0. Receive error interrupt (ERI) requests and break interrupt (BRI) requests can be cleared by clearing both the RIE and REIE bits in this register to 0.

TIE Bit (Transmit Interrupt Enable)

Specifies whether to enable or disable a transmit-FIFO-data-empty interrupt (TXI) request when the serial transmit data is transferred from the transmit FIFO data register (FTDR) into the transmit shift register (TSR), the quantity of data in the transmit FIFO data register falls below the specified trigger number for transmission, and the TDFE flag in the serial status register (FSR) is set to 1.

Note: TXI interrupt requests can be cleared either by writing a greater quantity of transmit data than the specified transmission trigger number into the FTDR register, reading 1 from the TDFE flag, and then clearing the TDFE flag to 0, or by clearing this bit to 0.

23.2.7 Serial Status Register (FSR)

The FSR register is a 16-bit register. The 8 lower-order bits indicate the status flag representing the SCIFA operating state.

The CPU can always read and write to the FSR register, but cannot write 1 to the status flags (ER, TEND, TDFE, BRK, RDF, and DR bits) in this register. These flags can be only cleared to 0 when they have first been read (after being set to 1). b3 (FER) and b2 (PER) are read-only bits that cannot be written.

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ER	TEND	TDFE	BRK	FER	PER	RDF	DR
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R(/W)*1	R(/W)*1	R(/W)*1	R(/W)*1	R	R	R(/W)*1	R(/W)*1

Bit	Bit Name	Initial Value	R/W	Description
b0	DR	0	R/(W) *1	Receive Data Ready Flag 0: Reception is in progress, or no received data has remained in the FRDR register after normally completed receiving. 1: Next receive data has not been received.
b1	RDF	0	R/(W) *1	Receive FIFO Data Full Flag 0: The quantity of receive data in the FRDR register falls below the specified reception trigger number. 1: The quantity of receive data written in the FRDR register is equal to or greater than the specified reception trigger number.
b2	PER	0	R	Parity Error Flag*4 0: No receive parity error occurred in the next receive data read from the FRDR register. 1: A receive parity error occurred in the next receive data read from the FRDR register.
b3	FER	0	R	Framing Error Flag*4 0: No receive framing error occurred in the next data read from the FRDR register. 1: A receive framing error occurred in the next data read from the FRDR register.
b4	BRK	0	R/(W) *1	Break Detect Flag 0: No break signal is received. 1: A break signal is received.*2
b5	TDFE	1	R/(W) *1	Transmit FIFO Data Empty Flag 0: The quantity of transmit data written in the FTDR register exceeds the specified transmission trigger number. 1: The quantity of transmit data written in the FTDR register is equal to or less than the specified transmission trigger number.*3
b6	TEND	0	R/(W) *1	Transmit End Flag 0: Transmission is in the waiting state or in progress. 1: Transmission is completed.
b7	ER	0	R/(W) *1	Receive Error Flag 0: Reception is in progress or has normally completed. 1: A framing error or parity error has occurred during reception.
b15 to b8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. 0 can be only written to clear the flag after 1 is read.

Note 2. When a break signal is detected, transfer of the receive data (H'00) to the FRDR register stops after the detection. When the break ends and the receive signal becomes mark state (high level), the transfer of receive data resumes.

Note 3. Since the FTDR register is a 16-byte FIFO register, the maximum quantity of data that can be written when TDFE is 1 is "16 minus the number of non-transmitted data units". If additional data is written, the data is ignored. The quantity of data in the FTDR register is indicated by the 8 higher-order bits of the FDR register.

Note 4. When the DMAC is used to read data, the generation of errors cannot be checked by reading this flag.

DR Bit (Receive Data Ready Flag)

Indicates that the quantity of data stored in the receive FIFO data register (FRDR) falls below the specified reception trigger number, and that no next data has been received yet after the elapse of 15 ETUs from the last stop bit in asynchronous mode. In clock synchronous mode, this bit is not set to 1.

[Setting condition]

- DR is set to 1 when the FRDR register contains less data than the specified reception trigger number, and no next data has been received yet after the elapse of 15 ETUs*[1] from the last stop bit.

[Clearing conditions]

When either of the following is satisfied:

- DR is cleared to 0 when DR = 1 is read and then 0 is written to the DR flag.
- DR is cleared to 0 when all received data in the FRDR register are read.

Note 1. This is equivalent to one and a half (1.5) frames in the 8-bit format with one stop bit (ETU: elementary time unit).

Note: When the RE bit in SCR is cleared to 0, the DR bit is not affected and retains its previous value.

RDF Bit (Receive FIFO Data Full Flag)

Indicates that receive data has been transferred to the receive FIFO data register (FRDR), and the quantity of data in FRDR becomes equal to or greater than the specified reception trigger number.

[Setting condition]

- RDF is set to 1 when the quantity of receive data which is equal to or greater than the specified reception trigger number are stored in the FRDR register*[1].

[Clearing conditions]

- RDF is cleared to 0 when RDF = 1 is read and then 0 is written to this bit.
- RDF is cleared to 0 when the FRDR register is read.

Note 1. Since the FRDR register is a 16-byte FIFO register, the maximum quantity of data that can be read when this bit is 1 is equivalent to the specified reception trigger number. If an attempt is made to read after all the data in the FRDR register has been read, the read data is undefined. The quantity of receive data in the FRDR register is indicated by the 8 lower-order bits of the FDR register.

PER Bit (Parity Error Flag)

Indicates whether there is a parity error in the data read from the receive FIFO data register (FRDR) in asynchronous mode.

[Setting condition]

- PER is set to 1 when a parity error is present in the next data read from the FRDR register.

[Clearing condition]

- PER is cleared to 0 when no parity error is present in the next data read from the FRDR register.

FER Bit (Framing Error Flag)

Indicates whether there is a framing error in the data read from the receive FIFO data register (FRDR) in asynchronous mode.

[Setting condition]

- FER is set to 1 when a framing error is present in the next data read from the FRDR register.

[Clearing condition]

- FER is cleared to 0 when no framing error is present in the next data read from the FRDR register.

BRK Bit (Break Detect Flag)

Indicates that a break signal has been detected in receive data.

[Setting condition]

- BRK is set to 1 when data including a framing error is received, and the framing error is followed by at least one frame of data received at the space 0 level (low level).

[Clearing condition]

- BRK is cleared to 0 when software reads BRK after it has been set to 1 and then writes 0 to BRK.

TDFE Bit (Transmit FIFO Data Empty Flag)

Indicates that data has been transferred from the transmit FIFO data register (FTDR) into the transmit shift register (TSR), the quantity of data in the FRDR register becomes equal to or less than the specified transmission trigger number, and writing of transmit data to the FRDR register is enabled.

[Setting conditions]

When either of the following is satisfied:

- TDFE is set to 1 when the TE bit in SCR is 0.
- TDFE is set to 1 when the quantity of transmit data written in the FRDR register is equal to or less than the specified transmission trigger number.

[Clearing conditions]

- TDFE is cleared to 0 when 0 is written in the TDFE bit after reading TDFE = 1.
- When transmit data is written to the FTDR register

TEND Bit (Transmit End Flag)

Indicates that the FRDR register contains no more valid data and transmission is completed when transmitting the last bit of the transmit data.

[Setting condition]

When the following is satisfied:

- TEND is set to 1 when the FTDR register does not contain transmit data when the last bit of the serial transmission data is transmitted.

[Clearing conditions]

- When transmit data is written to the FTDR register
- When 0 is written to TEND after it has been read as 1

ER Bit (Receive Error Flag)

Indicates the occurrence of a framing error, or of a parity error when receiving the parity-added data*[1].

[Setting conditions]

When either of the following is satisfied:

- ER is set to 1 when the stop bit is found to be 0 after checking whether the stop bit of the received data is 1 at the end of one data receive operation*[1].
- ER is set to 1 when the total number of 1s in the received data and parity bit combined does not match the even or odd parity setting specified by the PM bit in the SMR register.

[Clearing condition]

- When 0 is written to ER after it has been read as 1

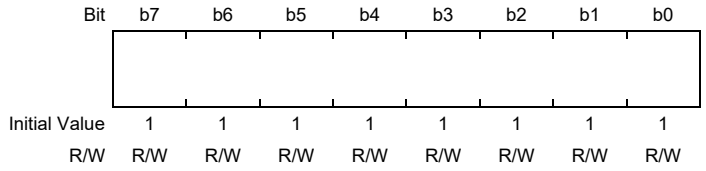
Note 1. Clearing the RE bit to 0 in the SCR register does not affect this bit, which retains its previous value. Even if a receive error occurs, the receive data is transferred to the FRDR register and the receive operation is continued. Whether the data read from the FRDR register includes a receive error can be detected by the FER and PER bits in the FSR register.

Note: In two stop bits mode, only the first stop bit is checked; the second stop bit is not checked.

23.2.8 Bit Rate Register (BRR)

The BRR register is an 8-bit register that, together with the baud rate generator clock source selected by the CKS[1:0] bits in the serial mode register (SMR), determines the serial transmit/receive bit rate.

This register is located in the same address as that of the MDDR register and selected when the MDDRS bit in SEMR is 0. The CPU can read and write to BRR. Writing to BRR should be executed when TE = RE = 0 in the SCR register.



The BRR setting is calculated using the following formulae.

[Asynchronous mode]

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$N = \frac{P0\phi}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0))

$$N = \frac{P0\phi}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$N = \frac{P0\phi}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0))

$$N = \frac{P0\phi}{16 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1))

[Clock synchronous mode]

$$N = \frac{P0\phi}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B: Bit rate (bit/s)

N: Setting of the BRR register ($0 \leq N \leq 255$) (The setting must satisfy the electrical characteristics).

P0ϕ: Operating frequency for peripheral modules (MHz)

n: Baud rate generator clock source (n = 0, 1, 2, 3) (For the clock sources and values of n, see **Table 23.5**).

Remark The MDDR register is used to adjust the bit rate. For details, see **Section 23.2.9, Modulation Duty Register (MDDR)**.

Table 23.5 SMR Register Setting

n	Clock Source	SMR Register Settings	
		CKS1	CKS0
0	P0ϕ	0	0
1	P0ϕ/4	0	1
2	P0ϕ/16	1	0
3	P0ϕ/64	1	1

The bit rate error in asynchronous mode is calculated using the following formulae.

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$\text{Error (\%)} = \left\{\frac{\text{P0}\phi \times 10^6}{(\text{N}+1) \times \text{B} \times 64 \times 2^{2\text{n}-1}} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0))

$$\text{Error (\%)} = \left\{\frac{\text{P0}\phi \times 10^6}{(\text{N}+1) \times \text{B} \times 32 \times 2^{2\text{n}-1}} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$\text{Error (\%)} = \left\{\frac{\text{P0}\phi \times 10^6}{(\text{N}+1) \times \text{B} \times 32 \times 2^{2\text{n}-1}} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0))

$$\text{Error (\%)} = \left\{\frac{\text{P0}\phi \times 10^6}{(\text{N}+1) \times \text{B} \times 16 \times 2^{2\text{n}-1}} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1))

Table 23.6 list the examples of the BRR register setting in asynchronous mode, and **Table 23.7** list the examples of the BRR register setting in clock synchronous mode.

Table 23.6 Bit Rates and BRR Register Settings in Asynchronous Mode

	P0ϕ (MHz)		
	100		
Bit Rate (bps)	n	N	Error (%)
150	—	—	—
300	3	162	−0.15
600	2	80	0.47
1200	2	162	−0.15
2400	2	80	0.47
4800	1	162	−0.15
9600	1	80	0.47
14400	0	216	0.01
19200	0	162	−0.15
28800	0	108	−0.45
31250	0	99	0.00
38400	0	80	0.47
115200	0	26	0.47
500000	0	5	*1

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 0.
When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is doubled.
When bits SEMR.ABCS0 and SEMR.BGDM are both 1, the bit rate is quadrupled.
Configure settings so the range of error is no greater than 1%.

Note 1. Values for the blank cells in the table can be set using the MDDR register.
For details, see **Section 23.2.9, Modulation Duty Register (MDDR)** and the **Table 23.12**.

Table 23.7 Bit Rates and BRR Register Settings in Clock Synchronous Mode

	Pϕ (MHz)	
	100	
Bit Rate (bps)	n	N
250	—	—
500	—	—
1000	—	—
2500	3	155
5000	3	77
10000	3	38
25000	1	249
50000	1	124
100000	0	249
250000	0	99
500000	0	49
1000000	0	24
2500000	0	9
5000000	0	4

Remarks —: Setting is prohibited.

Note: Continuous transmission or reception is not possible.
Set the BRR register so that the range of error can fall within 1% or less.

Table 23.8 lists the maximum bit rates for various frequencies in asynchronous mode when the baud rate generator is used. **Table 23.9** lists the maximum bit rates for various frequencies in clock synchronous mode when the baud rate generator is used. **Table 23.10** and **Table 23.11** list the maximum rates for external clock inputs in asynchronous mode and clock synchronous mode, respectively.

Table 23.8 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (in Asynchronous Mode)

		Settings	
Pϕ (MHz)	Maximum Bit Rate (bit/s)	n	N
100	12500000	0	0

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 1. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is 1/2. When bits SEMR.ABCS0 and SEMR.BGDM are both 1, the bit rate is 1/4.

Table 23.9 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (in Clock Synchronous Mode)

	Discontinuous Transmission/Reception			Continuous Transmission/Reception		
		Settings			Settings	
Pϕ (MHz)	Maximum Bit Rate (bit/s)	n	N	Maximum Bit Rate (bit/s)	n	N
100	25000000	0	0	12500000	0	1

Table 23.10 Maximum Bit Rates with External Clock Input (in Asynchronous Mode)

Pϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bit/s)
100	25.00	3125000

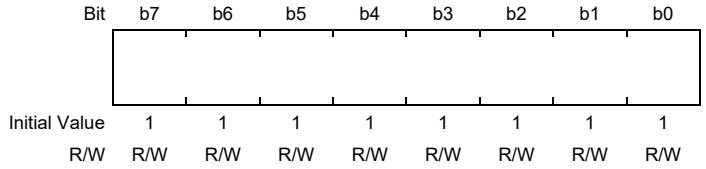
Note: This is an example when the SEMR.ABCS0 bit is 1. When the ABCS0 bit is set to 0, the bit rate is 1/2.

Table 23.11 Maximum Bit Rates with External Clock Input (in Clock Synchronous Mode)

Pϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bit/s)
100	8.33	8330000

23.2.9 Modulation Duty Register (MDDR)

The MDDR register corrects the bit rate adjusted by the BRR register. The value after reset of this register is H'FF. When the BRME bit in SEMR is set to 1, the bit rate generated by the on-chip baud rate generator is evenly corrected according to the settings of MDDR (MDDR/256). The relationship between the MDDR register setting and the bit rate (B) is given by the following formula. The MDDR register is located in the same address as that of the BRR register and is selected when the MDDRS bit in SEMR is 1. This register is only writable when TE = RE = 0 in the SCR register. b7 in this register is fixed to 1.



Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The formulae below show the relationships between the MDDR setting and the bit rate (B) when the bit rate modulation function is used.

[Asynchronous mode]

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$B = \frac{P0\phi \times 10^6}{64 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$B = \frac{P0\phi \times 10^6}{32 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$B = \frac{P0\phi \times 10^6}{32 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$B = \frac{P0\phi \times 10^6}{64 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

[Clock synchronous mode]

$$B = \frac{P0\phi \times 10^6}{8 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$$

When the bit rate modulation is used, the bit rate average error is given by the following formulae.

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$\text{Error (\%)} = \left\{\frac{\text{P0}\phi \times 10^6}{\text{B} \times 64 \times 2^{2\text{n}-1} \times (256/\text{MDDR}) \times (\text{N}+1)} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$\text{Error (\%)} = \left\{\frac{\text{P0}\phi \times 10^6}{\text{B} \times 32 \times 2^{2\text{n}-1} \times (256/\text{MDDR}) \times (\text{N}+1)} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$\text{Error (\%)} = \left\{\frac{\text{P0}\phi \times 10^6}{\text{B} \times 32 \times 2^{2\text{n}-1} \times (256/\text{MDDR}) \times (\text{N}+1)} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$\text{Error (\%)} = \left\{\frac{\text{P0}\phi \times 10^6}{\text{B} \times 16 \times 2^{2\text{n}-1} \times (256/\text{MDDR}) \times (\text{N}+1)} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

B: Bit rate (bits/s)

N: BRR register setting ($0 \leq N \leq 255$) (The setting must satisfy the electrical characteristics).

P0ϕ: Operating frequency for peripheral modules (MHz)

MDDR: MDDR setting ($128 \leq MDDR \leq 256$)

n: Baud rate generator clock source (n = 0, 1, 2, 3) (For the clock sources and values of n, see **Table 23.5**).

Table 23.12 Bit Rates and BRR and MDDR Registers Settings in Asynchronous Mode

	P0ϕ (MHz)			
	100			
Bit Rate (bps)	n	N	MDDR	Error (%)
150	3	252	199	0.02
300	3	161	255	0.08
600	3	80	255	0.08
1200	2	161	255	0.08
2400	2	80	255	0.08
4800	1	161	255	0.08
9600	1	80	255	0.08
14400	—	—	—	—
19200	0	161	255	0.08
28800	0	107	254	0.08
31250	—	—	—	—
38400	0	80	253	0.08
115200	0	26	255	0.08
500000	0	5	246	0.10

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 0. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is doubled. When bits SEMR.ABCS0 and SEMR.BGDM are both 1, the bit rate is quadrupled. Configure settings so the range of error is no greater than 1%.

23.2.10 FIFO Control Register (FCR)

The FCR register resets the quantity of data in the transmit FIFO data register (FTDR) and the receive FIFO data register (FRDR) and specifies the number of triggers. This register also specifies whether to enable the loop-back test.

The CPU can always read and write to the FCR register.

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	RSTRG[2:0]			RTRG[1:0]		TTRG[1:0]		MCE	TFRST	RFRST	LOOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	LOOP	0	R/W	Loop-Back Test 0: Loop back test is disabled. 1: Loop back test is enabled
b1	RFRST	0	R/W	Receive FIFO Data Register Reset 0: Normal operation 1: Resets the FRDR register.
b2	TFRST	0	R/W	Transmit FIFO Data Register Reset 0: Normal operation 1: Resets the FTDR register.
b3	MCE	0	R/W	Modem Control Enable 0: Model signal is disabled.*1 1: Model signal is enabled.
b5, b4	TTRG[1:0]	All 0	R/W	Transmit FIFO Data Trigger Number Select b5 b4 0 0: 8 (8)*2 0 1: 4 (12)*2 1 0: 2 (14)*2 1 1: 0 (16)*2
b7, b6	RTRG[1:0]	All 0	R/W	Receive FIFO Data Trigger Number Select In asynchronous mode: b7 b6 0 0: 1 0 1: 4 1 0: 8 1 1: 14 In clock synchronous mode: b7 b6 0 0: 1 0 1: 2 1 0: 8 1 1: 14
b10 to b8	RSTRG[2:0]	All 0	R/W	RTS# Output Active Trigger Number Select b10 b8 0 0 0: 15 0 0 1: 1 0 1 0: 4 0 1 1: 6 1 0 0: 8 1 0 1: 10 1 1 0: 12 1 1 1: 14

Bit	Bit Name	Initial Value	R/W	Description
b15 to b11	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. The CTS# input level does not affect the transmit operation. Similarly, the RTS# input level does not affect the receive operation.

Note 2. Values in parentheses mean the number of empty bytes in the FTDR register when the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXI) request is generated.

LOOP Bit (Loop-Back Test)

Internally connects between the transmit output pin (TXD) and the receive input pin (RXD) and between the RTS# pin and the CTS# pin, to perform loop-back testing.

RFRST Bit (Receive FIFO Data Register Reset)

Disables the receive data in the receive FIFO data register (FRDR) and makes the data to the empty state. If you set this bit to 1, be sure to clear it to 0 afterward.

TFRST Bit (Transmit FIFO Data Register Reset)

Disables the transmit data in the transmit FIFO data register (FTDR) and makes the data to the empty state. If you set this bit to 1, be sure to clear it to 0 afterward.

MCE Bit (Modem Control Enable)

Specifies whether to enable or disable the modem control signals, CTS# and RTS#. In clock synchronous mode, this bit should always be set to 0.

TTRG[1:0] Bits (Transmit FIFO Data Trigger Number Select)

Specify the reference quantity of data for transmission (i.e., the threshold number of entries to trigger the writing of further data for transmission) for setting of the TDFE flag in the serial status register (FSR). When the number of entries for transmission in the transmission FIFO, i.e. the number of entries written to the transmit FIFO data register (FTDR) that are yet to be transmitted, falls to or below the specified trigger number for transmission, the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXI) request is generated.

The setting in these bits is valid when the TTRGS bit in the FTCR register is 0. When the TTRGS bit in the FTCR register is 1, the setting of the TFTC[4:0] bits in the FTCR register is valid.

RTRG[1:0] Bits (Receive FIFO Data Trigger Number Select)

Specify the reference quantity of receive data (i.e., the threshold number of entries to trigger the reading of received data) for setting of the RDF flag in the serial status register (FSR). When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the specified trigger number for reception, the RDF flag is set to 1 and a receive FIFO data full interrupt (RXI) request is generated.

The setting in these bits is valid when the RTRGS bit in the FTCR register is 0. When the RTRGS bit in the FTCR register is 1, the setting of the RFTC[4:0] bits in the FTCR register is valid.

RSTRG[2:0] Bits (RTS# Output Active Trigger Number Select)

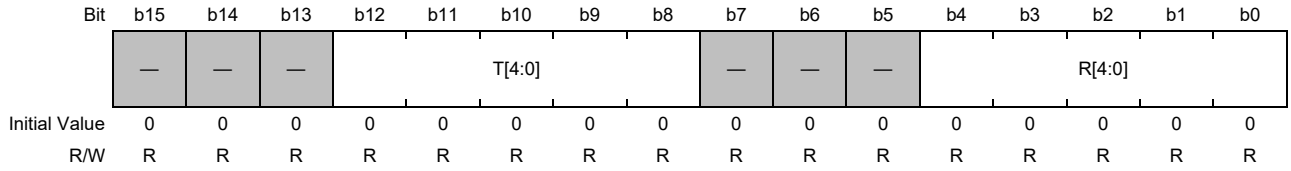
When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the specified trigger number, the RTS# signal is in the high state.

The setting in these bits is only valid when a modem signal is enabled by the MCE bit in this register in asynchronous mode.

23.2.11 FIFO Data Count Register (FDR)

The FDR register indicates the quantity of data stored in the transmit FIFO data register (FTDR) and the receive FIFO data register (FRDR).

This register indicates the quantity of transmit data in the FTDR register with the 8 higher-order bits, and the quantity of receive data in the FRDR register with the 8 lower-order bits. The CPU can always read the FDR register.



Bit	Bit Name	Initial Value	R/W	Description
b4 to b0	R[4:0]	All 0	R	Receive Data Quantity in FRDR Indicate the quantity of receive data stored in the FRDR register.
b7 to b5	—	All 0	R	Reserved These bits are read as 0.
b12 to b8	T[4:0]	All 0	R	Non-Transmitted Data Quantity in FTDR Indicate the quantity of non-transmitted data stored in the FTDR register.
b15 to b13	—	All 0	R	Reserved These bits are read as 0.

R[4:0] Bits

Indicate the quantity of receive data stored in the FRDR register.

H’00 means no received data, and H’10 means that all of the received data is stored in the FRDR register.

T[4:0] Bits

Indicate the quantity of non-transmitted data stored in the FTDR register.

H’00 means no transmit data, and H’10 means that all of the data for transmission is stored in the FTDR register.

23.2.12 Serial Port Register (SPTR)

The SPTR register controls input/output and data of the pins multiplexed to SCIFA function. The CPU can always read and write to the SPTR register.

NOTE

b6, b4, b2, and b0 of this register respectively indicate the input status of their corresponding pins. See the descriptions for each bit for details. Writings to these bits in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	RTS2IO	RTS2DT	CTS2IO	CTS2DT	SCKIO	SCKDT	SPB2IO	SPB2DT
Initial Value	0	0	0	0	0	0	0	0	0	x	0	x	0	x	0	x
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Remarks: x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
b0	SPB2DT	x	R/W	Serial Port Break Data Select Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2IO bit. See **Table 23.15**.
b1	SPB2IO	0	R/W	Serial Port Break Input/Output Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2DT bit.
b2	SCKDT	x	R/W	SCK Port Data Select Controls the SCK pin in combination with the CM bit in the SMR register, the SCKIO bit, and the CKE1 and CKE0 bits in the SCR register. See **Table 23.17**.
b3	SCKIO	0	R/W	SCK Port Input/Output Controls the SCK pin in combination with the CM bit in the SMR register, the SCKDT bit, and the CKE1 and CKE0 bits in the SCR register. See **Table 23.17**.
b4	CTS2DT	x	R/W	CTS# Port Data Select Controls the CTS# pin in combination with MCE bit in FCR and CTS2IO bit. See **Table 23.14**. The SCIF 3 and SCIF 4 channels are not supported.
b5	CTS2IO	0	R/W	CTS# Port Output Specify Controls the CTS# pin in combination with MCE bit in FCR and CTS2IO bit. See **Table 23.14**. The SCIF 3 and SCIF 4 channels are not supported.
b6	RTS2DT	x	R/W	RTS# Port Data Select Controls the RTS# pin in combination with MCE bit in FCR and RTS2IO bit. See **Table 23.13**. The SCIF 3 and SCIF 4 channels are not supported.
b7	RTS2IO	0	R/W	RTS# Port Output Specify Controls the RTS# pin in combination with MCE bit in FCR and RTS2IO bit. See **Table 23.13**. The SCIF 3 and SCIF 4 channels are not supported.
b15 to b8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Remarks: x: Undefined

SPB2DT Bit (Serial Port Break Data Select)

This bit specifies the output level of the TXD pin when the setting of the SCR.TE bit is 0. The RXD pin input status can be read from this bit regardless of the SPB2IO bit setting. However, the RXD pin function must have been selected with the general I/O port.

SPB2IO Bit (Serial Port Break Input/Output)

Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2DT bit.

SCKDT Bit (SCK Port Data Select)

The SCK pin status can be read from this bit regardless of the SCKIO bit setting. (When the SCK pin is used for input, the input signal is invalid (has no means) but the pin status can be read.) However, the SCK pin function must have been selected with the general I/O port.

SCKIO Bit (SCK Port Input/Output)

Specifies input or output status of the SCK pin. This bit controls the SCK pin in combination with the SCKDT bit, the CM bit in the SMR register, and the CKE1 and CKE0 bits in the SCR register.

CTS2DT Bit (CTS# Port Data Select)

The status of the CTS# pin can be read from this bit regardless of the CTS2IO bit setting. However, the CTS# pin function must have been selected with the general I/O port.

RTS2DT Bit (RTS# Port Data Select)

The status of the RTS# pin can be read from this bit regardless of the RTS2IO bit setting. However, the RTS# pin function must have been selected with the general I/O port

Table 23.13 RTS# Pin Status

FCR.MCE Bit Setting	RTS2IO Bit Setting	RTS2DT Bit Setting	RTS# Pin Status
0	0	×	Setting prohibited*1
0	1	0	Low output
0	1	1	High output
1	×	×	Modem control output

Remarks: ×: Don't care

Note 1. There is no problem with the initial setting if the RTS# pin is not used.

Table 23.14 CTS# Pin Status

FCR.MCE Bit Setting	CTS2IO Bit Setting	CTS2DT Bit Setting	CTS# Pin Status
0	0	×	Setting prohibited*1
0	1	0	Low output
0	1	1	High output
1	×	×	Modem control input

Remarks: ×: Don't care

Note 1. There is no problem with the initial setting if the CTS# pin is not used.

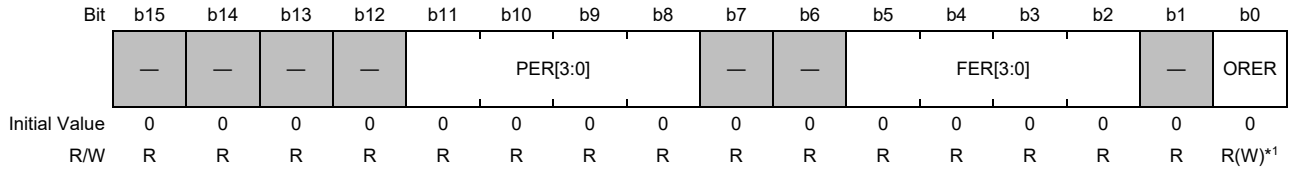
Table 23.15 TXD Pin Status

SCR.TE Bit Setting	SPB2IO Bit Setting	SPB2DT Bit Setting	TXD Pin Status
0	0	×	Setting prohibited
0	1	0	Low output
0	1	1	High output
1	×	×	Transmit data output

Remarks: ×: Don’t care

23.2.13 Line Status Register (LSR)

The LSR register is a 16-bit register. The PER and FER bits indicate the number of receive errors in the receive FIFO data register. 1 cannot be written to the OREER status flag. The flag should be read as 1 prior to clearing it to 0.



Bit	Bit Name	Initial Value	R/W	Description
b0	ORER	0	R/(W) *1	Overrun Error Flag 0: Reception is in progress or has normally completed. 1: An overrun error has occurred during reception.
b1	—	0	R	Reserved This bit is read as 0.
b5 to b2	FER[3:0]	All 0	R	Framing Error Count Indicates the quantity of data with a framing error among the receive data stored in the receive FIFO data register (FRDR).
b7, b6	—	All 0	R	Reserved These bits are read as 0.
b11 to b8	PER[3:0]	All 0	R	Parity Error Count Indicates the quantity of data with a parity error among the receive data stored in the receive FIFO data register (FRDR).
b15 to b12	—	All 0	R	Reserved These bits are read as 0.

Note 1. To clear the flag, 0 can be only written after 1 is read.

ORER Bit (Overrun Error Flag)

Indicates that receive operation abnormally stops due to occurrence of an overrun error. This flag is not affected and retains its previous state if the RE bit in the serial control register (SCR) is cleared to 0. The receive FIFO data register (FRDR) retains the data before an overrun error occurred, and newly received data is lost. When the ORER bit is set to 1, the SCIFA cannot continue subsequent serial reception.

[Setting condition]

- When the next serial reception is completed with the receive FIFO in full state (16-byte data is received)

[Clearing condition]

- When 0 is written to ORER after being read as 1.

Note: When the internal clock is selected while the SCIFA is in clock synchronous mode, the amount of receive data can be controlled, so no overrun occurs.

FER[3:0] Bits (Framing Error Count)

The values of bits 5 to 2 indicate the quantity of data with a framing error after the ER bit in the FSR register is set. Reading 0000 from the FER[3:0] bits means all 16-byte receive data in the FRDR register have a framing error.

PER[3:0] Bits (Parity Error Count)

The values of bits 11 to 8 indicate the quantity of data with a parity error after the ER bit in the FSR register is set. Reading 0000 from the PER[3:0] bits means all 16-byte receive data in the FRDR register have a parity error.

23.2.14 Serial Extended Mode Register (SEMR)

The SEMR register specifies either LSB or MSB first, enables the noise cancellation, operation in normal or double-speed mode of the baud rate generator, and bit rate modulation, and selects the modulation register and the sampling count (either 8 or 16 times).

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	BGDM	—	BRME	MDDRS	DIR	NFEN	—	ABCS0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	ABCS0	0	R/W	Asynchronous Base Clock Select 0: Operates on a frequency 16 times the transfer rate as the base clock. 1: Operates on a frequency 8 times the transfer rate as the base clock.
b1	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b2	NFEN	0	R/W	Noise Cancellation Enable 0: Noise cancellation for the RxD pin is disabled. 1: Noise cancellation for the RxD pin is enabled.
b3	DIR	0	R/W	Data Transfer Direction Select 0: Transmits the data in the FTDR register by the LSB-first method. The received data is stored in the FRDR register by the LSB- first method. 1: Transmits the data in the FTDR register by the MSB-first method. The received data is stored in the FRDR register by the MSB- first method.
b4	MDDRS	0	R/W	Modulation Duty Register Select 0: BRR register is accessible. 1: MDDR register is accessible.
b5	BRME	0	R/W	Bit Rate Modulation Enable 0: Bit rate modulation is disabled. 1: Bit rate modulation is enabled.
b6	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b7	BGDM	0	R/W	Baud Rate Generator Double- Speed Mode Select 0: Baud rate generator normal mode: Baud rate generator operates on the clock signal produced by dividing the clock source by two. 1: Baud rate generator double-speed mode: Baud rate generator operates on the clock signal produced by the clock source (no frequency division).

ABCS0 Bit (Asynchronous Base Clock Select)

Selects the base clock for 1-bit period in asynchronous mode.

This bit setting is valid only in asynchronous mode (i.e., when the CM bit in the SMR register is 0).

NFEN Bit (Noise Cancellation Enable)

Reduces noise of the input to the RxD pin. This function is only valid in asynchronous mode. For details, see **Section 23.7, Noise Cancellation**.

In clock synchronous mode, this bit should always be set to 0.

DIR Bit (Data Transfer Direction Select)

Selects the serial communication format. This bit is valid only when the transmit/receive data is in 8-bit formats.*[1]

Note 1. Asynchronous mode or clock synchronous mode with the 8-bit data length

MDDRS Bit (Modulation Duty Register Select)

Selects the register to be enabled access to it.

BRME Bit (Bit Rate Modulation Enable)

Specifies whether to enable or disable the bit rate modulation.

BGDM Bit (Baud Rate Generator Double-Speed Mode Select)

Selects operating mode of the baud rate generator. When setting 1 in this bit, the baud rate generator included in the SCIFA operates in double-speed mode. The setting of this bit is only effective in asynchronous mode (SMR.CM bit = 0) when the internal clock is selected as the clock source (SCR.CKE[1:0] = 00b). Use normal mode under any other settings.

23.2.15 FIFO Trigger Control Register (FTCR)

The FTCR register is a 16-bit register that specifies FIFO trigger conditions. The CPU can always read from and write to the FTCR register.

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RTRGS	—	—	RFTC[4:0]					TTRGS	—	—	TFTC[4:0]				
Initial Value	0	0	0	1	1	1	1	1	0	0	0	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b4 to b0	TFTC[4:0]	All 1	R/W	Transmit FIFO Data Trigger Number H'00: Transmit data trigger number is 0. H'0F: Transmit data trigger number is 15. Do not set H'10 to H'1F in these bits.
b6, b5	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b7	TTRGS	0	R/W	Transmit Trigger Select 0: TTRG[1:0] bits in FCR are valid. 1: TFTC[4:0] bits in FTCR are valid.
b12 to b8	RFTC[4:0]	All 1	R/W	Receive FIFO Data Trigger Number H'01: Receive data trigger number is 1. H'10: Receive data trigger number is 16. Do not set H'00 and H'11 to H'1F in these bits.
b14, b13	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b15	RTRGS	0	R/W	Receive Trigger Select 0: RTRG[1:0] bits in FCR are valid. 1: RFTC[4:0] bits in FTCR are valid.

TFTC[4:0] Bits (Transmit FIFO Data Trigger Number)

Specify the reference quantity of data for transmission (i.e., the threshold number of entries to trigger the writing of further data for transmission) for setting of the TDFE flag in the serial status register (FSR).

When the number of entries for transmission in the transmission FIFO, i.e. the number of entries written to the transmit FIFO data register (FTDR) that are yet to be transmitted, falls to or below the specified trigger number for transmission, the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXI) request is generated.

RFTC[4:0] Bits (Receive FIFO Data Trigger Number)

Specify the reference quantity of receive data (i.e., the threshold number of entries to trigger the reading of received data) for setting of the RDF flag in the serial status register (FSR).

When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the specified trigger number for reception, the RDF flag is set to 1 and a receive FIFO data full interrupt (RXI) request is generated.

23.3 Operation

23.3.1 Overview

For serial communication, the SCIFA can select either asynchronous mode in which characters are synchronized individually or a clock synchronous mode in which communication is synchronized with clock pulses.

The SCIFA has a 16-stage FIFO buffer for both transmission and receptions, reducing the overhead of the CPU and enabling continuous high-speed communication. The RTS# and CTS# signals are provided as modem control signals. Selection of a transmission/reception format is enabled with the serial mode register (SMR). **Table 23.16** shows the transmission format which can be selected in the serial mode register (SMR). As shown in **Table 23.17**, the SCIFA clock source can be selected by the combination of the CKE1 and CKE0 bits in the serial control register (SCR).

(1) Asynchronous Mode

- Data length is selectable either 7 or 8 bits
- Parity addition and 1- or 2-bit stop bit addition are selectable.
  (The combination of the preceding selections determines the transmission/reception format and character length).
- In reception, it is possible to detect framing errors, parity errors, receive FIFO data full, overrun errors, receive data ready, and breaks.
- The stored data quantities are indicated in the FIFO data count register (FDR), respectively for transmit and receive FIFO data.
- An internal or external clock can be selected as the SCIFA clock source.
  When an internal clock is selected, the SCIFA operates using the clock of on-chip baud rate generator and can output the clock with a frequency 16 (or 8) times the bit rate.
  When an external clock is selected, the external clock input must have a frequency 16 (or 8) times the bit rate. (The on-chip baud rate generator is not used.)

(2) Clock Synchronous Mode

- The transmission/reception format is fixed to the 8-bit data length.
- In reception, it is possible to detect overrun errors (ORER).
- An internal or external clock can be selected as the SCIFA clock source.
  When an internal clock is selected, the SCIFA operates using the clock of the on-chip baud rate generator, and outputs this clock to external devices as the synchronous clock.
  When an external clock is selected, the SCIFA operates on the input synchronous clock not using the on-chip baud rate generator.

Table 23.16 SMR Register Settings and SCIFA Communication Formats

SMR Register					SCIFA Transmission/Reception Format		
b7	b6	b5	b3				
CM	CHR	PE	STOP	Mode	Data Length	Parity Bit	Stop Bit Length
0	0	0	0	Asynchronous mode	8 bits	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
	1	0	0		7 bits	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
1	×	×	×	Clock synchronous mode	8 bits	Not set	None

Remarks ×: Don't care

Table 23.17 SMR, SCR, and SPTR Register Settings and SCIFA Clock Source Selection

SMR Register	SCR Register		SPTR Register				
b7	b1	b0	b3	b2			
CM	CKE1	CKE0	SCKIO	SCKDT	Mode	Clock Source	SCK Pin Function
0	0	0	0	×	Asynchronous mode	Internal	Input pin (input signal invalid) (Initial state)
			1	0			SCK pin state: Low
			1	1			SCK pin state: High
		1	×	×			Outputs a clock with frequency 16/8 times the bit rate*1
	1	0	×	×		External	Inputs a clock with frequency 16/8 times the bit rate*2
		1	×	×		Setting prohibited	
1	0	×	×	×	Clock synchronous mode	Internal	Outputs the synchronous clock
	1	0	×	×		External	Inputs the synchronous clock
		1	×	×		Setting prohibited	

Remarks: ×: Don't care

Note 1. SEMR.ABCS0 = 0: Output a clock that has a frequency 16 times the bit rate. SEMR.ABCS0 = 1: Output a clock that has a frequency 8 times the bit rate.

Note 2. SEMR.ABCS0 = 0: Input a clock that has a frequency 16 times the bit rate. SEMR.ABCS0 = 1: Input a clock that has a frequency 8 times the bit rate.

23.3.2 Operation in Asynchronous Mode

In asynchronous mode, each transmitted or received character begins with a start bit and ends with a stop bit. Serial communication is synchronized one character at a time.

The transmitting and receiving sections of the SCIFA are independent, so full duplex communication is possible. The transmitter and receiver are 16-stage FIFO buffered, so data can be written and read while transmitting and receiving are in progress, enabling continuous transmission and reception.

Figure 23.2 shows the general format of asynchronous serial communication.

In asynchronous serial communication, the communication line is normally held in the mark (high) state. The SCIFA monitors the line and starts serial communication when the line goes to the space (low) state, considered as a start bit. One serial character consists of a start bit (low), data (LSB first when LSB-first transfer is selected), parity bit (high or low), and stop bit (high), in that order.

When receiving in asynchronous mode, the SCIFA synchronizes at the falling edge of the start bit. The SCIFA samples each data bit on the eighth pulse of a clock with a frequency 16 or 8 times the bit rate*1. Receive data is latched at the center of each bit.

Note 1. When the SEMR.ABCS0 bit = 0, data is sampled on the eighth pulse of a clock with a frequency 16 times the bit rate.
When the SEMR.ABCS0 bit = 1, data is sampled on the fourth pulse of a clock with a frequency 8 times the bit rate.

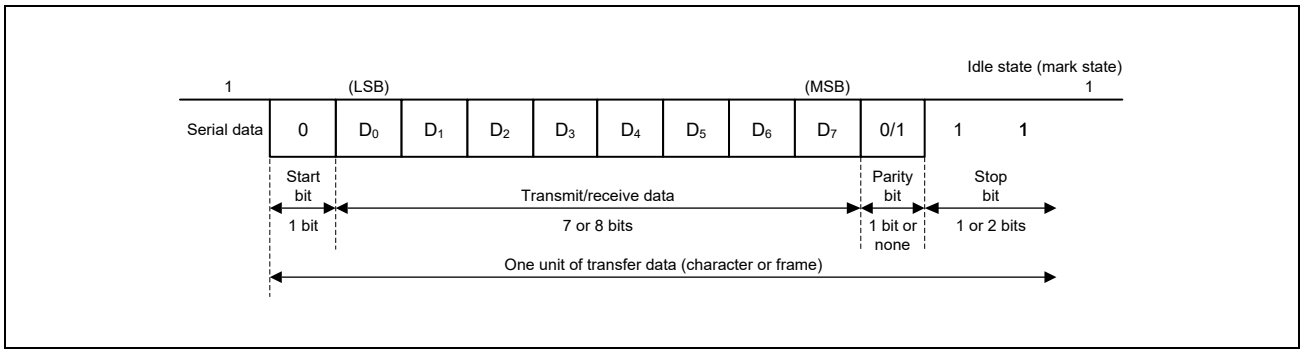


Figure 23.2 Data Format in Asynchronous Communication
(8-Bit Data with Parity and Two Stop Bits when LSB-First Transfer is Selected)

(1) Transmit/Receive Formats

Table 23.18 lists the eight communications formats that can be selected in asynchronous mode. The format is selected by setting in the serial mode register (SMR).

Table 23.18 Serial Communications Formats (in Asynchronous Mode)

SMR Setting			Serial Transmit/Receive Format and Frame Length											
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	START	8-bit data								STOP		
		1	START	8-bit data								STOP	STOP	
	1	0	START	8-bit data								P	STOP	
		1	START	8-bit data								P	STOP	STOP
1	0	0	START	7-bit data							STOP			
		1	START	7-bit data							STOP	STOP		
	1	0	START	7-bit data							P	STOP		
		1	START	7-bit data							P	STOP	STOP	

Note: START: Start bit
STOP: Stop bit
P: Parity bit

(2) Clock

An SCIFA transmit/receive clock can be selected from two types of clock sources: the internal clock generated by the on- chip baud rate generator, the external clock input from the SCK pin. The clock source is selected by the settings of the CM bit in the serial mode register (SMR), the CKE[1:0] bits in the serial control register (SCR), and the ACS0 bit in the serial extended mode register (SEMR). For clock source selection, refer to **Table 23.17**.

When an external clock is input at the SCK pin, it must have a frequency equal to 16/8 times the desired bit rate.

When the SCIFA operates on an internal clock, it can output a clock signal on the SCK pin. The frequency of this output clock is 16/8 times the desired bit rate.

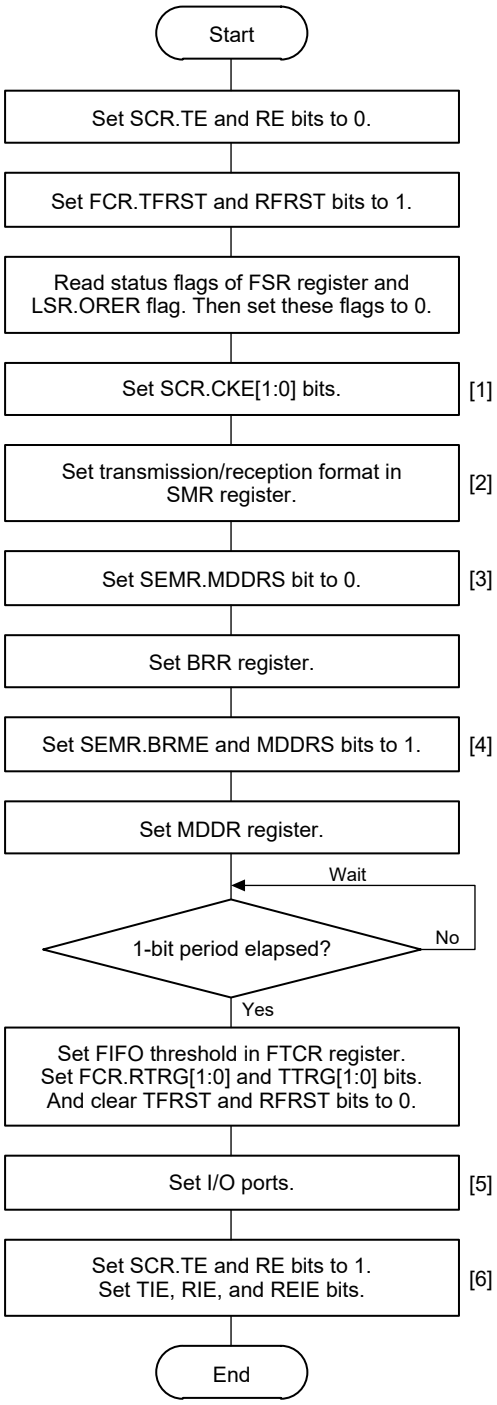
(3) Transmitting and Receiving Data

SCIFA Initialization (in Asynchronous Mode)

Before transmitting or receiving data, clear the TE and RE bits to 0 in the serial control register (SCR), and then initialize the SCIFA as follows.

When changing operating mode or communication format, always clear the TE and RE bits in the SCR register to 0 before following the procedure given below. Clearing TE to 0 initializes the transmit shift register (TSR). Clearing TE and RE to 0, however, does not initialize the serial status register (FSR), transmit FIFO data register (FTDR), or receive FIFO data register (FRDR), which retain their previous contents. Clear TE to 0 after all transmit data has been transmitted and TEND flag in the FSR register is set. The TE bit can be cleared to 0 during transmission, but the transmit data (the TXD pin output level) after the TE bit is cleared to 0 depends on the settings of the SPB2IO and SPB2DT bits in the SPTR register. Set the TFRST bit in the FCR register to 1 and reset the FTDR register before TE is set to 1 again to start transmission.

When an external clock is used, the clock should not be stopped during initialization or subsequent operation. SCIFA operation becomes unreliable if the clock is stopped. **Figure 23.3** shows a sample flowchart for initializing the SCIFA in asynchronous mode.



[1] Set the TIE, RIE, TE, and RE bits in the SCR register to 0 before setting the CKE[1:0] bits.

[2] Set the transmission/reception format in the SMR register.

[3] Clear the SEMR.MDDRS bit to 0 and write a value corresponding to the bit rate in the BRR register. Not necessary if an external clock is used.

[4] Set the SEMR.BRME and MDDRS bits to 1 and write a value corresponding to the bit rate in the MDDR register. Not necessary if an external clock is used. Wait for at least 1-bit period before proceeding to the next step.

[5] Set I/O ports and the MPC register so that they can be used as the SCK, TXD, and RXD pins.

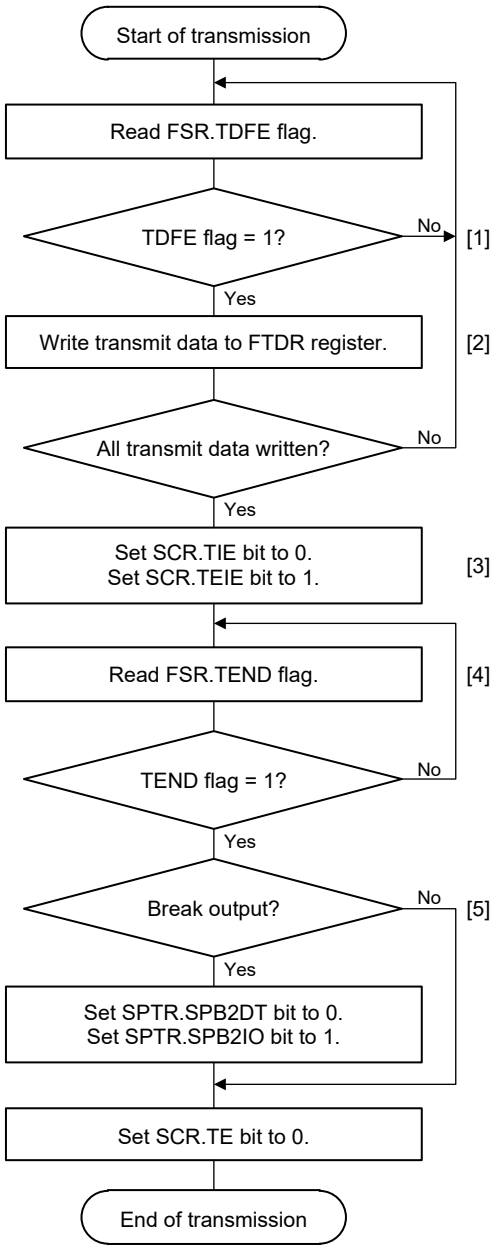
[6] After setting the MDDR register, wait for at least 1-bit period and then write 1 to the SCR.TE or RE bit. To permit interrupt requests, set TIE, RIE and REIE bits to 1. Setting the TE and RE bits to 1 makes the TXD and RXD pins available to be used. The TXD pin enters the mark state and the RXD pin enters the idle state waiting for a start bit.

Figure 23.3 Sample Flowchart for SCIFA Initialization in Asynchronous Mode

Transmitting Serial Data (in Asynchronous Mode)

Figure 23.4 shows a sample flowchart for serial transmission in asynchronous mode.

Follow the procedure given below for serial data transmission after enabling the SCIFA for transmission.



[1] Check the SCIFA status and write the transmit data:
Read the FSR register, confirm that the TDFE flag is set to 1, and then write the transmit data to the FTDR register. The transmit data quantity that can be written is “16 - FDR.T[4:0]”. A change of the TDFE flag from 0 to 1 can be known by a TXIF interrupt request.

[2] Serial transmission continuation procedure:
When transmit data is written to the FTDR register, the TDFE flag is automatically cleared to 0.

[3] Serial transmission end procedure (when TXIF or TEIF interrupt is used):
After the last transmission data is written to the FTDR register, set the TIE bit to 0 and the TEIE bit to 1 in the SCR register. If the DMAC is activated to write the transmit data to the FTDR register, enable the transmit end interrupt. This will generate a CPU interrupt request when the last transmit data is written in the FTDR register. The TIE and TEIE bit settings will be changed by the CPU interrupt process.
This procedure is unnecessary if the TXIF or TEIF interrupt is not used.

[4] Serial transmission end confirmation:
Read the FSR register and confirm that the TEND flag has changed to 1.
A change of the TEND flag from 0 to 1 can be known by a TEIF interrupt request.

[5] Break output during serial transmission:
To cause a break to be output during serial transmission, set the SPB2DT bit to 0 and the SPB2IO bit to 1 in the SPTR register. Then set the SCR.TE bit to 0.*1

In Step [1], the number of data bytes that can be written can be checked by the FDR.T[4:0] flag (non-transmitted data quantity in the FTDR register).

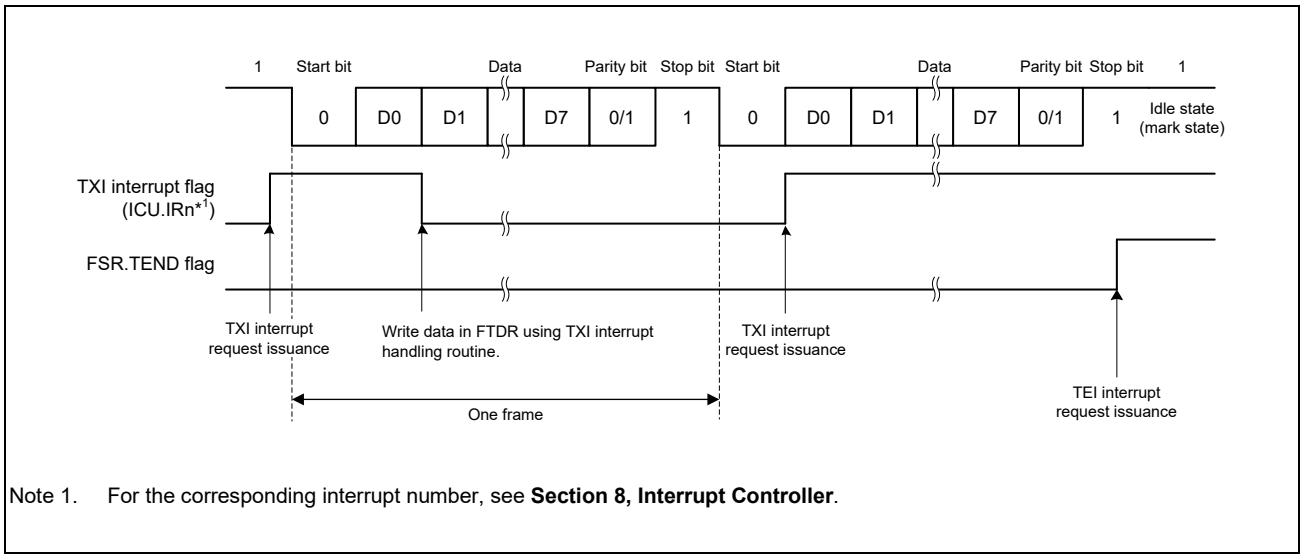
Note 1. The SPB2DT bit of the SPTR register indicates the input status of the RXD pin. Writings to the SPB2DT and SPB2IO bits of this register in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written to the SPB2DT bit. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

Figure 23.4 Sample Flowchart for Transmitting Serial Data in Asynchronous Mode

In asynchronous mode, the SCIFA performs serial transmission as described below.

1. When data is written into the transmit FIFO data register (FTDR) by the TXI interrupt processing routine, the SCIFA transfers the data from the FTDR register to the transmit shift register (TSR) and starts transmission. Confirm that the TDFE flag in the serial status register (FSR) is set to 1 before writing transmit data to the FTDR register. The number of data bytes that can be written is “16 minus the number of non-transmitted data units”.
2. When data is transferred from the FTDR register to the TSR register and transmission is started, consecutive transmit operations are performed until there is no transmit data left in the FTDR register. When the number of transmit data bytes in the FTDR register becomes equal to or less than the transmission trigger number specified in the FIFO control register (FCR) or FIFO trigger control register (FTCR), the TDFE flag is set. If the TIE bit in the serial control register (SR) is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXI) request is generated. The serial transmit data is output from the TXD pin in the following order.
   a) Start bit: One-bit 0 is output.
   b) Transmit data: 8- or 7-bit data is output in LSB-first order (when LSB-first transfer is selected).
   c) Parity bit: One parity bit (even or odd parity) is output. (A format in which a parity bit is not output can also be selected.)
   d) Stop bit(s): One or two 1 bits (stop bits) are output.
   e) Mark state: 1 is output continuously until the start bit that starts the next transmission is output.
3. The SCIFA checks the transmit data of the FTDR register at the timing for sending the stop bit. If data is present, the data is transferred from the FTDR register to the TSR register, the stop bit is output, and then serial transmission of the next frame is started. If there is no data to be transmitted, TEND flag in the FSR register is set to 1, the stop bit is output, and then the SCIFA enters the mark state (high level) in which 1 is output continuously.

Figure 23.5 shows an example of the operation for transmission in asynchronous mode.



Note 1. For the corresponding interrupt number, see **Section 8, Interrupt Controller**.

Figure 23.5 Example of Transmit Operation in Asynchronous Mode
(8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)

4. When modem control is enabled, transmission can be stopped/resumed by the input level to the CTS# pin. When a high level is input to the CTS# pin during transmission, the SCIFA enters the mark state (high level) after completion of one-frame data transmission. When a low level is input to the CTS# pin, output of the next data to be transmitted begins with a start bit. **Figure 23.6** shows an example of the operation for transmission when using the modem control function.

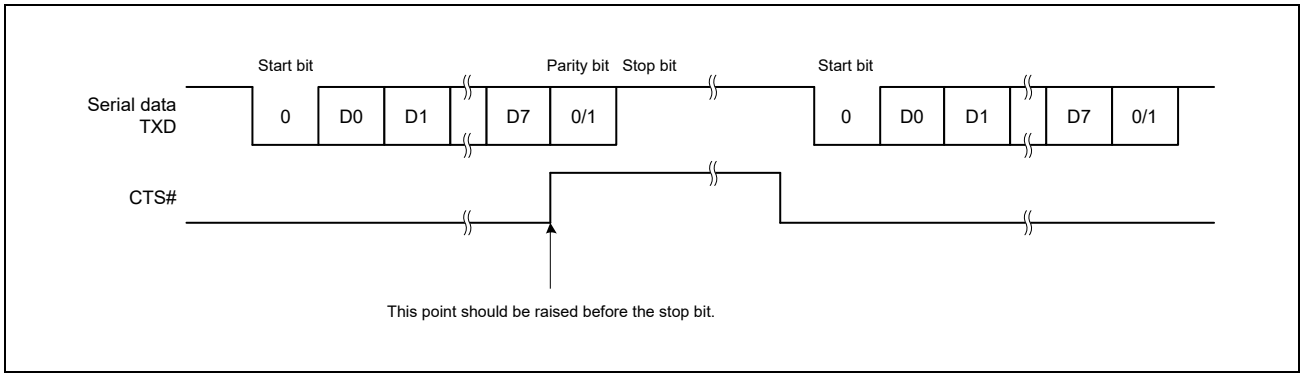
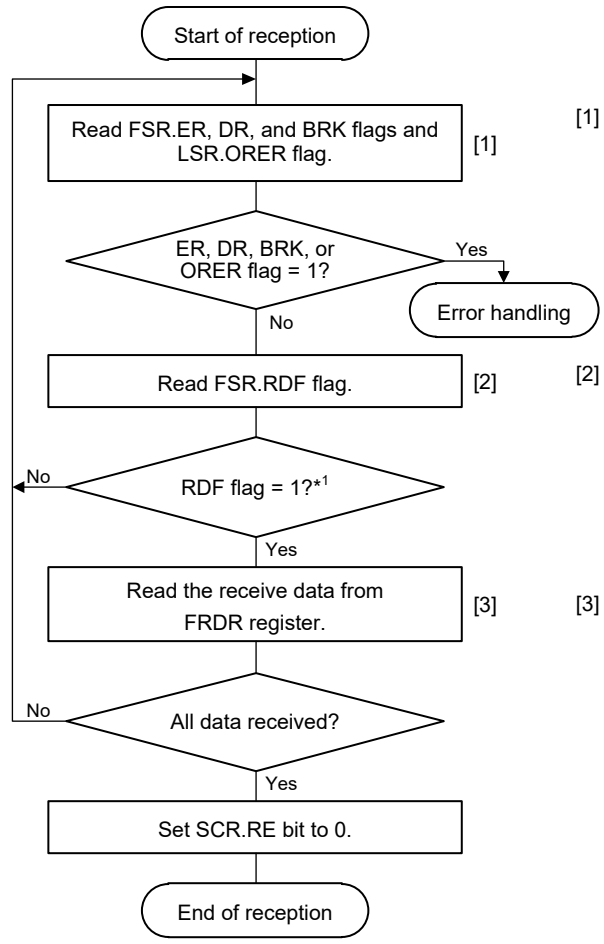


Figure 23.6 Example of Transmit Operation in Asynchronous Mode Using Modem Control Function (CTS#)

Receiving Serial Data (in Asynchronous Mode)

Figure 23.7 and **Figure 23.8** show sample flowcharts for serial reception in asynchronous mode. Follow the procedure given below for serial data reception after enabling the SCIFA for reception.



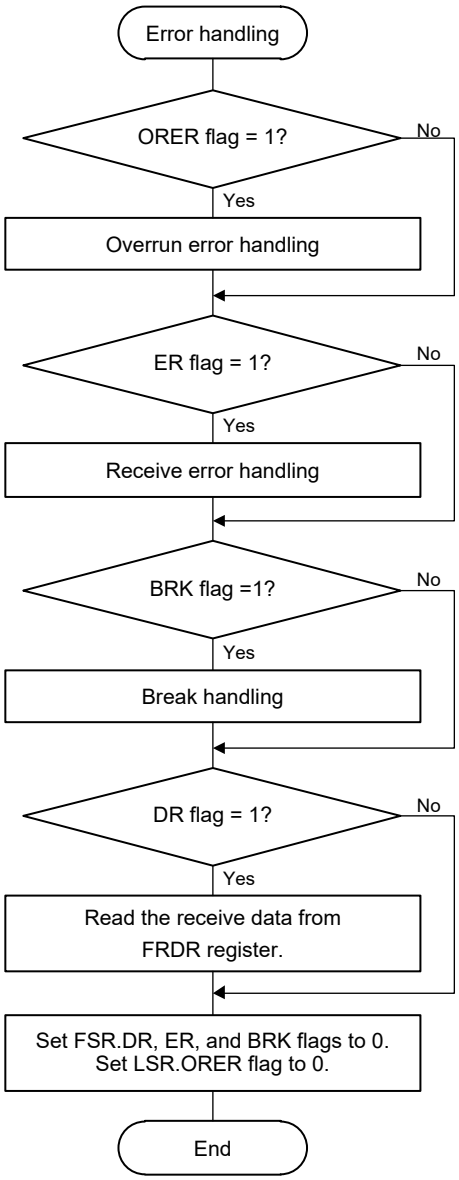
[1] Receive error handling and break detection:
When a receive error occurs, read the ER, DR, and BRK flags in the FSR register and the LSR.ORER flag to identify the error. After handling the error as specified, clear the ER, DR, BRK, and ORER flags to 0.
In the case where a framing error occurs, a break can be detected by reading the value of the RXD pin.
Error occurrence can be known by a BRIF, ERIF, or DRIF interrupt request.

[2] Check the SCIFA status and read the receive data:
Read the FSR register, confirm that the RDF flag is set to 1, and then read the receive data at least as much as the amount specified by the receive FIFO threshold from those stored in the FEDR register. The receive data quantity that can be read can be checked by the FDR.R[4:0] flag.
A change of the RDF flag from 0 to 1 can be known by an RXIF interrupt request.

[3] Serial reception continuation procedure:
When receive data is read from the FRDR register, the RDF flag is automatically cleared to 0.

Note 1. The number of all the received data units should be an integral multiple of the receive FIFO threshold. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 23.7 Sample Flowchart for Receiving Serial Data in Asynchronous Mode (1)



- Whether a framing error or parity error has occurred in the received data that is read from the FRDR register can be checked with the FER and PER flags in the FSR register.
- When a break signal is received, the received data is not transferred to the FRDR register while the BRK flag is set to 1. However, note that the last data in the FRDR register is H'00 and the break data in which a framing error occurred is stored.

Figure 23.8 Sample Flowchart for Receiving Serial Data in Asynchronous Mode (2)

In asynchronous mode, the SCIFA performs serial reception as described below.

1. The SCIFA monitors the communication line, and if a 0 start bit is detected, it performs internal synchronization to start reception.
2. The received data is stored into the RSR register in LSB-to-MSB order (when LSB-first transfer is selected).
3. The parity bit and stop bit are received.
   After receiving these bits, the SCIFA carries out the following checks.
   a) Stop bit check: The SCIFA checks whether the stop bit is 1. If there are two stop bits, only the first is checked.
   b) The SCIFA checks whether receive data can be transferred from the receive shift register (RSR) to the receive FIFO data register (FRDR).
   c) Parity bit check: The SCIFA checks whether the parity bit is an expected value.
   d) Overrun error check: The SCIFA checks whether the ORER flag is 0, indicating that the overrun error has not occurred.
   e) Break check: The SCIFA checks whether the BRK flag is 0, indicating that the break state is not set.

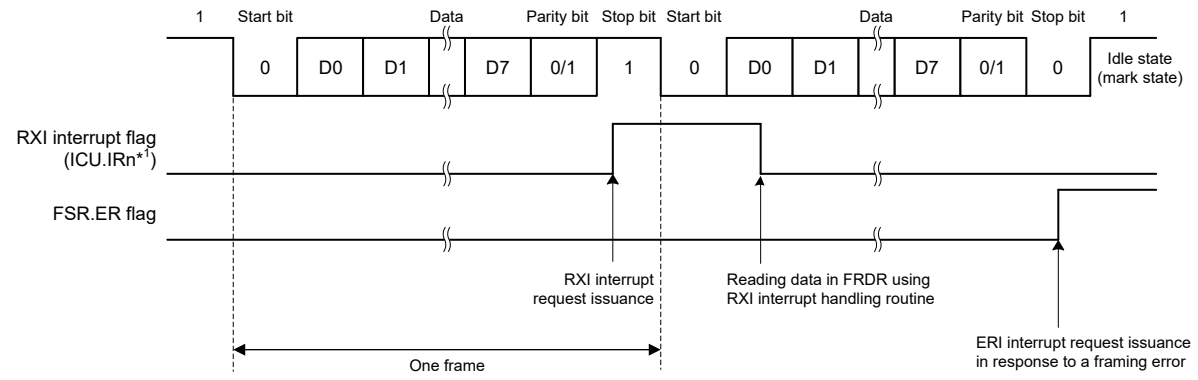
If all the above checks are passed, the receive data is stored in the FRDR register.

Remark When a parity error or a framing error occurs, reception is not suspended.

4. When receive data units equaling or exceeding the specified reception trigger number are stored in the receive FIFO data register (FRDR) and the RDF flag is changed to 1, a receive FIFO data full interrupt (RXI) request is generated while the RIE bit in the SCR register is set to 1. When the quantity of data in the FRDR register falls below the specified reception trigger number and the RIE bit in the SCR register is set to 1, a receive data ready interrupt (DRI) request is generated if no next data is received after the elapse of 15 ETUs*[1] from the last stop bit (the DR flag in the FSR register is 1). When the ER flag in the FSR register is changed to 1, a receive error interrupt (ERI) request is generated while the RIE or REIE bit in the SCR register is set to 1. When the BRK or ORER flag is changed to 1 in the FSR register, a break reception interrupt (BRI) request is generated while the RIE or REIE bit in the SCR register is set to 1.

Note 1. It is equivalent to 1 and half frames of 8-bit format with one stop bit (ETU: Element Time Unit).

Figure 23.9 shows an example of the operation for reception in asynchronous mode.



Note 1. For the corresponding interrupt number, see **Section 8, Interrupt Controller**.

Figure 23.9 Example of SCIFA Receive Operation in Asynchronous Mode
(8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)

5. When modem control is enabled, the RTS# signal that indicates the FRDR register has space is output. When the RTS# pin is at low level, reception is possible. The RTS# pin being at the high level indicates that the number of entries in the FRDR register is equal to or greater than the threshold for output of the active level of the RTS# signal and that the transmission of further data needs to be suspended until the FRDR register has enough space. **Figure 23.10** shows an example of the operation for reception in asynchronous mode when using the modem control function.

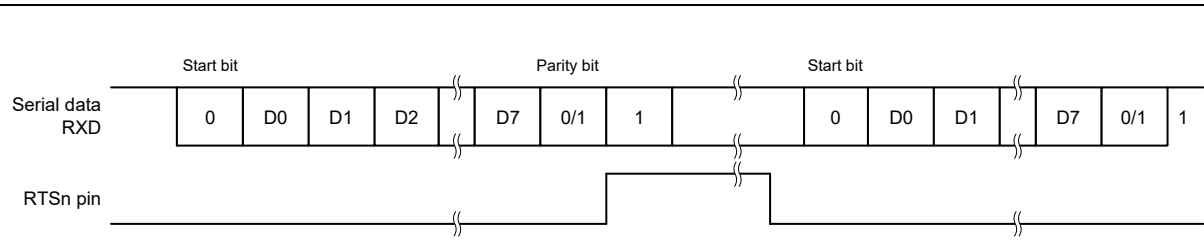


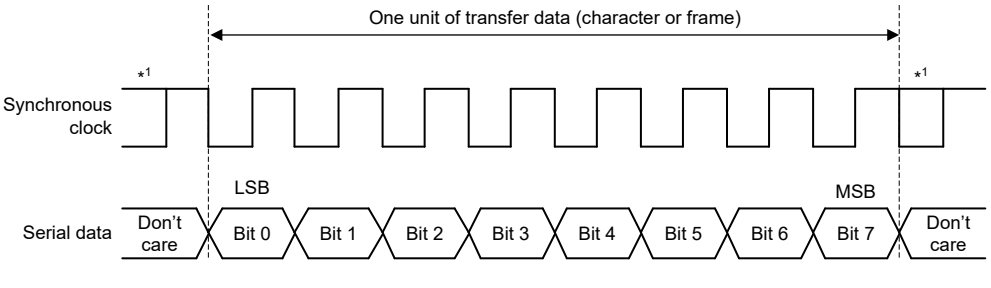
Figure 23.10 Example of SCIFA Receive Operation in Asynchronous Mode Using Model Control Function (RTS#)

23.3.3 Operation in Clock Synchronous Mode

In clock synchronous mode, the SCIFA transmits and receives data in synchronization with clock pulses. This mode is suitable for high-speed serial communication.

Full-duplex communication is possible because the SCIFA transmitter and receiver are independent and share the same clock. Since the transmitter and the receiver have 16-stage FIFO buffers, respectively, continuous transmission or reception is possible by reading or writing data while transmission or reception is in progress.

Figure 23.11 shows the general format in clock synchronous serial communication.



Note 1. High level except in continuous transmission/reception

Figure 23.11 Data Format in Clock Synchronous Communication (when LSB-First Transfer is Selected)

In clock synchronous serial communication, each data bit is output on the communication line from one falling edge of the synchronous clock to the next. Data is guaranteed valid at the rising edge of the synchronous clock.

In each character, the serial data bits are transmitted in order from the LSB (first) to the MSB (last). After output of the MSB, the communication line remains in the state of the MSB (when LSB-first transfer is selected).

In clock synchronous mode, the SCIFA receives data by synchronizing with the rising edge of the synchronous clock.

(1) Transmit/Receive Formats

The data length is fixed at eight bits.

No parity bit can be added.

(2) Clock

An internal clock generated by the on-chip baud rate generator or an external clock input from the SCK pin can be selected as the SCIFA transmit/receive clock according to the settings of the CM bit in the serial mode register (SMR) and the CKE[1:0] bits in the serial control register (SCR).

When the SCIFA operates on an internal clock, it outputs the synchronous clock signal at the SCK pin. Eight synchronous clock pulses are output per transmitted or received character. Unless the SCIFA is transmitting or receiving, the synchronous clock signal remains in the high state. When the SCIFA only receives data on an internal clock, the internal clock signal outputs while the RE bit in the SCR register is 1 until the number of data units in the receive FIFO reaches the specified reception trigger number.

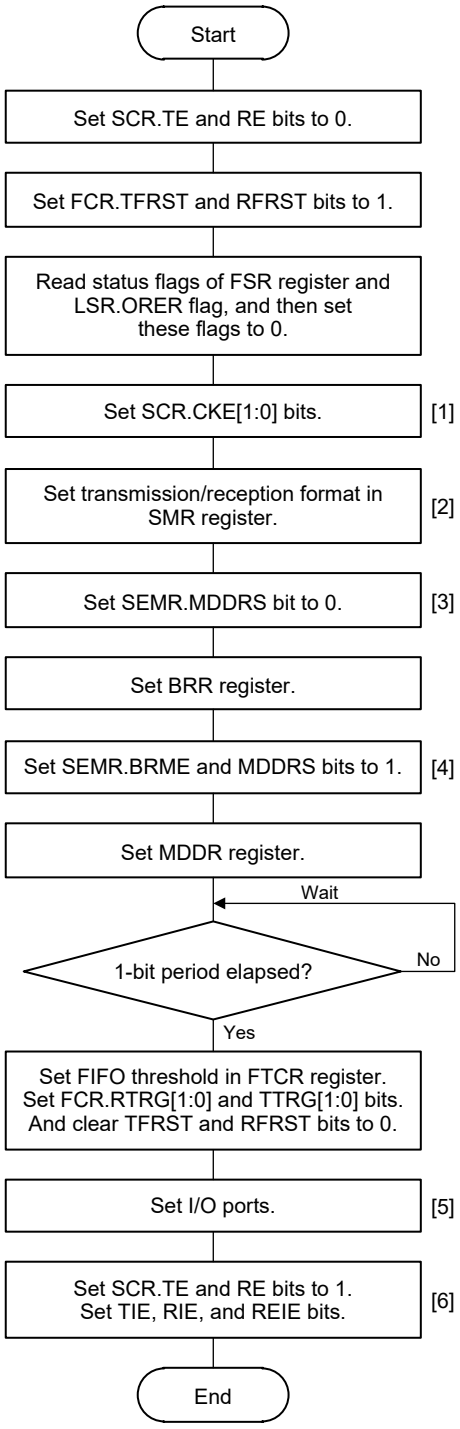
(3) Transmitting and Receiving Data

SCIFA Initialization (in Clock Synchronous Mode)

Before transmitting or receiving, clear the TE and RE bits to 0 in the serial control register (SCR), and then initialize the SCIFA by performing the following procedure.

Similarly, before changing the mode or communication format, clear the TE and RE bits to 0, and then change it by performing the following procedure. Clearing TE to 0 initializes the transmit shift register (TSR). Clearing RE to 0, however, does not initialize the RDF, PER, FER, and ORER flags and the receive FIFO data register (FRDR), which retain their previous contents.

Figure 23.12 shows a sample flowchart for initializing the SCIFA in clock synchronous mode.



[1] Set the TIE, RIE, TE, and RE bits in the SCR register to 0 before setting the CKE[1:0] bits.

[2] Set the transmission/reception format in the SMR register.

[3] Clear the SEMR.MDDRS bit to 0 and write a value corresponding to the bit rate in the BRR register. Not necessary if an external clock is used.

[4] Set the SEMR.BRME and MDDRS bits to 1 and write a value corresponding to the bit rate in the MDDR register. Not necessary if an external clock is used. Wait for at least 1-bit period before proceeding to the next step.

[5] Set I/O ports and the MPC register so that they can be used as the SCK, TXD, and RXD pin.

[6] After setting the MDDR register, wait for at least 1-bit, and then write 1 to the SCR.TE or RE bit. To permit interrupt requests, set TIE, RIE, and REIE bits to 1. Setting the TE and RE bits to 1 makes the TXD and RXD pins available to be used. In transmission, those pins enter the mark state, and in reception, they enter the idle state waiting for a start bit.

Figure 23.12 Sample Flowchart for SCIFA Initialization in Clock Synchronous Mode

Transmitting Serial Data (in Clock Synchronous Mode)

Figure 23.13 shows a sample flowchart for transmitting serial data in clock synchronous mode.

Follow the procedure given below for serial data transmission after enabling the SCIFA for transmission.

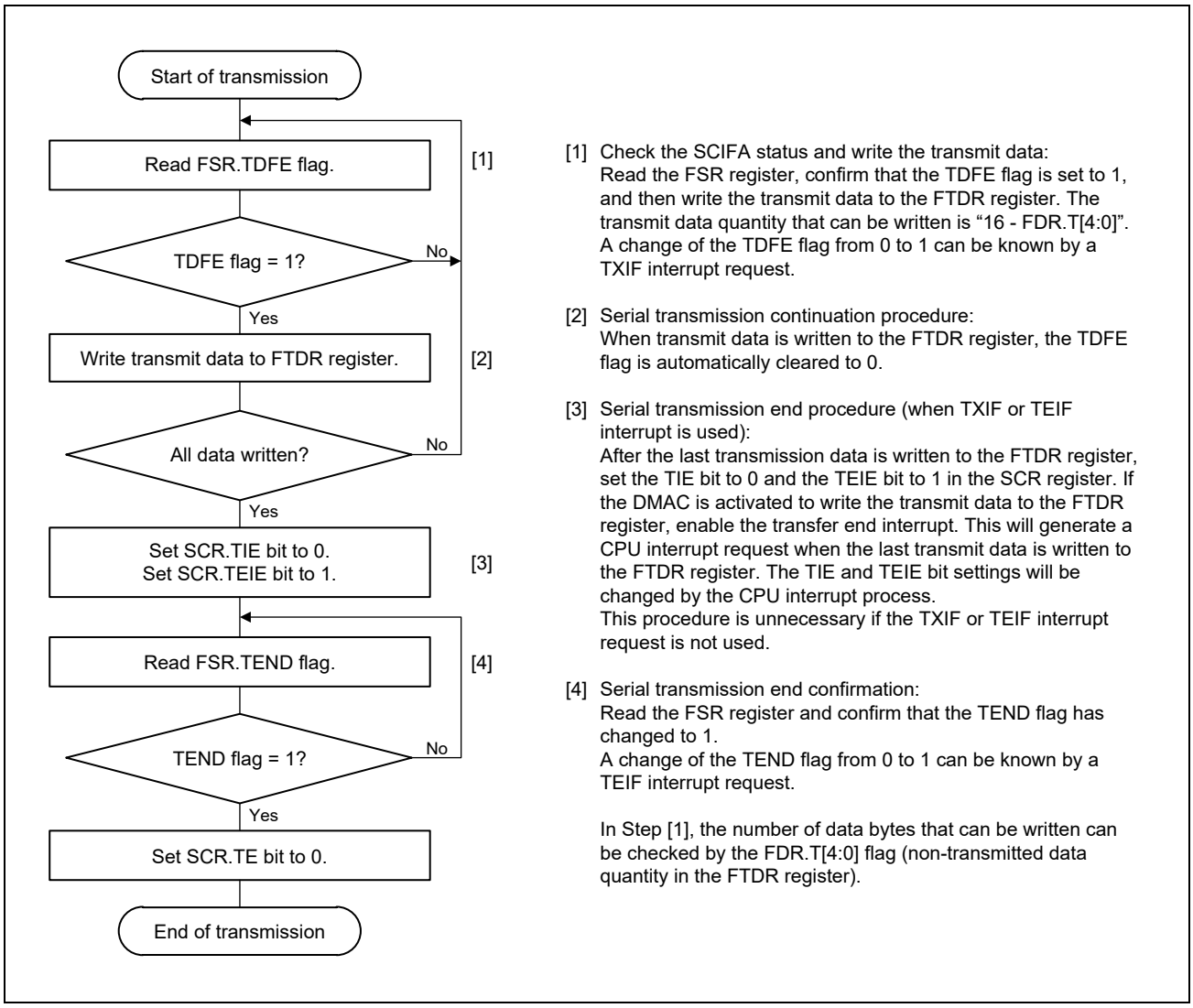
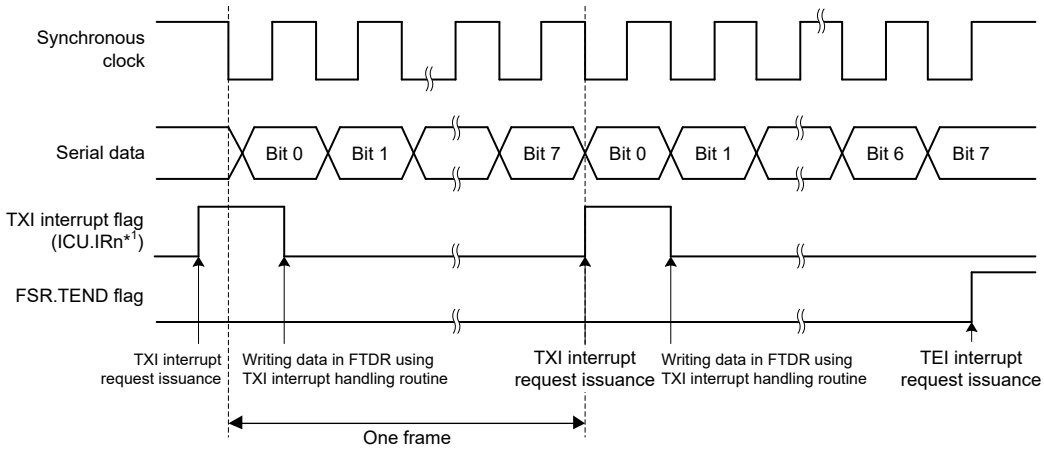


Figure 23.13 Sample Flowchart for Transmitting Serial Data in Clock Synchronous Mode

In clock synchronous mode, the SCIFA performs serial transmission as described below.

1. When data is written into the transmit FIFO data register (FTDR) by the TXI interrupt processing routine, the SCIFA transfers the data from the FTDR register to the transmit shift register (TSR) and starts transmission. Confirm that the TDFE flag in the serial status register (FSR) is set to 1 before writing transmit data to the FTDR register. The number of data bytes that can be written is “16 minus the specified number of non-transmitted data units”.
2. When data is transferred from the FTDR register to the TSR register and transmission is started, consecutive transmit operations are performed until there is no transmit data left in the FTDR register. When the number of transmit data bytes in the FTDR register becomes equal to or less than the transmission trigger number set in the FIFO control register (FCR) or FIFO trigger control register (FTCR), the TDFE flag in the FSR register is set. If the TIE bit in the serial control register (SCR) is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXI) request is generated.
   If clock output mode is selected, the SCIFA outputs eight synchronous clock pulses. If an external clock source is selected, the SCIFA outputs data in synchronization with the input clock. Data is output from the TXD pin in order from the LSB (b0) to the MSB (b7) (when LSB-first transfer is selected).
3. The SCIFA checks the transmit data of the FTDR register at the timing for sending the MSB (bit 7). If data is present, the data is transferred from the FTDR register to the TSR register, and then serial transmission of the next frame is started. If there is no data, the TXD pin holds the output level of the last data after TEND flag in the FSR register is set to 1 and the MSB (bit 7) is output.
4. After the end of serial transmission, the SCK pin is held in the high state.

Figure 23.14 shows an example of SCIFA transmit operation in clock synchronous mode.



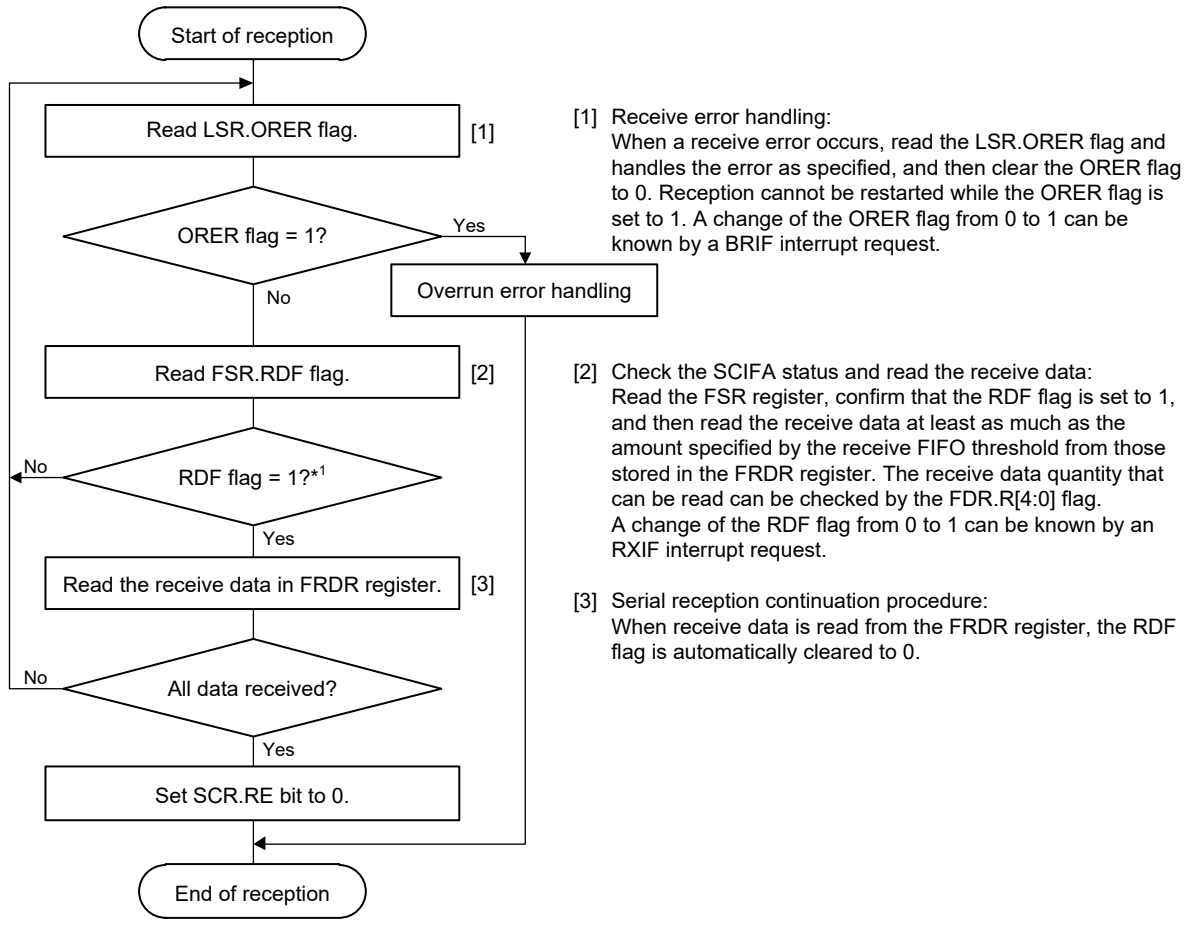
Note 1. For the corresponding interrupt number, see **Section 8, Interrupt Controller**.

Figure 23.14 Example of SCIFA Transmit Operation in Clock Synchronous Mode (when LSB-First Transfer is Selected)

Receiving Serial Data (in Clock Synchronous Mode)

Figure 23.15 and **Figure 23.16** show sample flowcharts for receiving serial data in clock synchronous mode.

Follow the procedure given below for serial data reception after enabling the SCIFA for reception. When switching from asynchronous mode to clock synchronous mode without SCIFA initialization, make sure that the ORER, PER, and FER flags in the line status register (LSR) are cleared to 0.



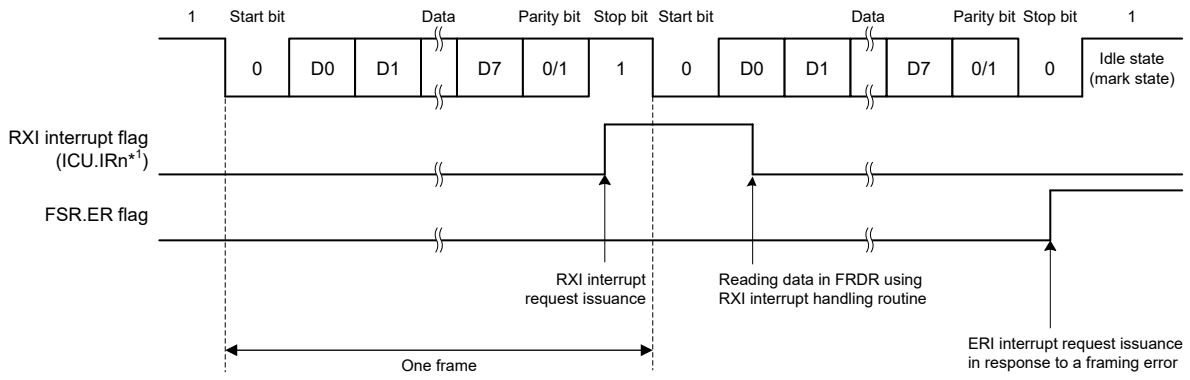
Note 1. The number of all the received data units should be an integral multiple of the receive FIFO threshold. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 23.15 Sample Flowchart for Receiving Serial Data in Clock Synchronous Mode

In clock synchronous mode, the SCIFA performs serial reception as described below.

1. The SCIFA synchronizes with the synchronous clock input or output and starts reception.
2. Receive data is stored into the receive shift register (RSR) in order from the LSB to the MSB (when LSB-first transfer is selected). After receiving the data, the SCIFA checks whether the receive data can be transferred from the RSR register to the FRDR register. If data can be transferred, the SCIFA stores the received data in the FRDR register. If an overrun error is detected during the error check, further reception is not performed.
3. After the received data units equaling or exceeding the specified reception trigger number are stored in the FRDR register and the RDF flag is set to 1, a receive-data-full interrupt (RXI) request is generated when the RIE bit in the serial control register (SCR) is set to 1. When the ORER flag in the line status register (LSR) is set to 1 and the RIE or REIE bit in the SCR register is also set to 1, a break interrupt (BRI) request is generated.

Figure 23.16 shows an example of SCIFA receive operation in clock synchronous mode.



Note 1. For the corresponding interrupt number, see **Section 8, Interrupt Controller**.

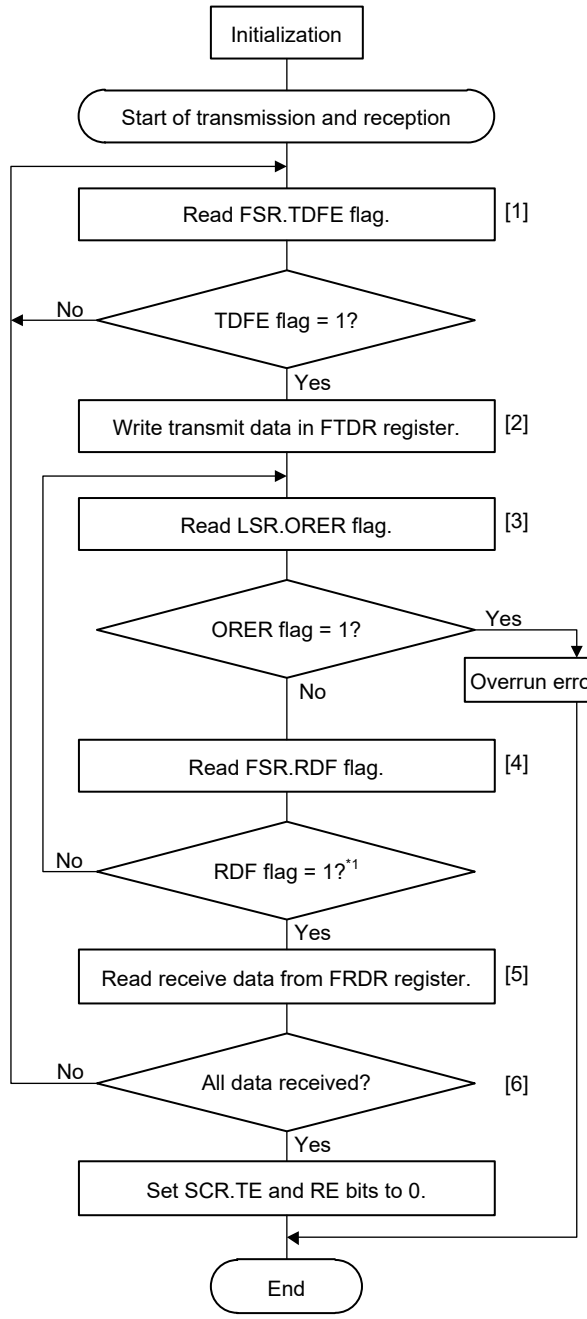
Figure 23.16 Example of SCIFA Receive Operation (when LSB-First Transfer is Selected)

Transmitting and Receiving Serial Data Simultaneously (in Clock Synchronous Mode)

Figure 23.17 shows a sample flowchart for transmitting and receiving serial data simultaneously in clock synchronous mode.

In simultaneous transmission/reception of serial data, number of receive data = number of transmit data = number of transmit data to be written to the FTDR register.

Follow the procedure given below for the simultaneous transmission/reception of serial data, after enabling the SCIFA for transmission/reception.



[1] Check the SCIFA status and write the transmit data:
Read the FSR register, confirm that the TDFE flag is set to 1, and then write the transmit data in the FTDR register.
A change of the TDFE flag from 0 to 1 can be known by a TXIF interrupt request.

[2] Serial transmission continuation procedure:
When transmit data is written to the FTDR register, the TDFE flag is automatically cleared to 0.

[3] Receive error handling:
When a receive error occurs, read the LSR.ORER flag and handles the error as specified, and then clear the ORER flag to 0. Reception cannot be restarted while the ORER flag is set to 1.
A change of the ORER flag from 0 to 1 can be known by a BRIF interrupt request.

[4] Check the SCIFA status and read the receive data:
Read the FSR register, confirm that the RDF flag is set to 1, and then read the receive data stored in the FRDR register.
A change of the RDF flag from 0 to 1 can be known by an RXIF interrupt request.

[5] Serial reception continuation procedure:
When receive data is read from the FRDR register, the RDF flag is automatically cleared to 0.

[6] Serial transmission/reception continuation procedure (supplementary):
To continue serial transmission/reception, read the FRDR register and clear the RDF flag to 0 before the FRDR register overflows and the ORER flag changes to 1. In addition, before the last bit of the last data written to the FTDR register is transmitted, write the data in the FTDR register and clear the TDFE flag to 0.
When writing data to the FTDR register or reading data from the FRDR register, the TDFE flag and RDF flag are automatically cleared to 0.

Note: When switching the operation from transmission or reception to the simultaneous transmission and reception, clear the TE and RE bits to 0 and then simultaneously set these bits to 1.

Note 1. The number of all the received data units should be an integral multiple of the receive FIFO threshold. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 23.17 Sample Flowchart for Simultaneous Transmitting/Receiving Serial Data in Clock Synchronous Mode

23.4 Bit Rate Modulation

Using the bit rate modulation, the bit rate can be corrected by skipping the specified number of clock pulses input to the baud rate generator. To correct the bit rate, only the number of clock pulses specified in the MDDR register are enabled among 256 internal clock pulses specified by the CKS1 and CKS0 bits in the SMR register in a way that forms average intervals.

Figure 23.18 shows an example where P0φ is selected by the CKS[1:0] bits in SMR and BRR and MDDR are set to 0 and 160, respectively, in asynchronous mode. In this example, the cycle of the base clock is evenly corrected (256/160) and the bit rate is also corrected (160/256). Note that skipping an internal clock causes bias and expansion or contraction is generated in the pulse width of the base clock.

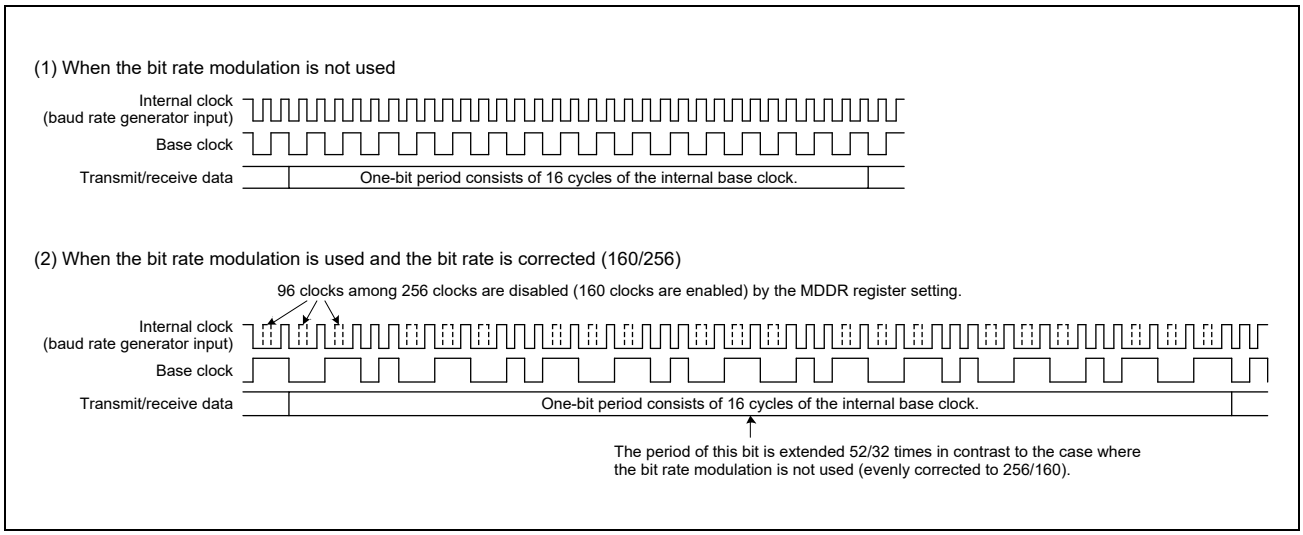


Figure 23.18 Example of Internal Base Clock when Bit Modulation is Used

23.5 Interrupt Sources

The SCIFA has six interrupt sources: transmit-FIFO-data-empty (TXI), receive-error (ERI), receive-FIFO-data-full (RXI), break (BRI), transmit-end (TEI), receive-data-ready (DRI). The TEI, DRI, ERI, and BRI interrupts share the same vector number.

Table 23.19 shows the interrupt sources and priority. The interrupt sources can be enabled or disabled using the TIE, RIE, REIE, TEIE bits in the SCR register and are separately input to the interrupt controller.

When the quantity of transmit data written in the FTDR register as a result of transmission is equal to or less than the specified transmission trigger number, the TDFE flag in the serial status register (FSR) is set to 1 and a TXI interrupt request is generated.

When the data units equaling or exceeding the specified transmission trigger number are stored in the receive FIFO register (FRDR) and the RDF flag in the FSR register is set to 1, a receive data full interrupt (RXI) request is generated. When the quantity of received data in the FRDR register is below the specified reception trigger number and no next data has been received yet even after the period of 15 ETUs elapsed*1 from the last stop bit, the DR flag in the FSR register is set to 1 and a receive data ready interrupt (DRI) request is generated. In clock synchronous mode, a DRI interrupt request is not generated.

When the BRK flag in the FSR register or the ORER flag in the LSR register is set to 1, a BRI interrupt request is issued. When the ER flag in the FSR register is set to 1, an ERI interrupt request is issued.

When TEND flag in the FSR register is set to 1, a TEI interrupt request is issued.

When the RIE bit is cleared to 0 and the REIE bit in the SCR register is set to 1, an ERI and a BRI interrupt requests are issued but an RXI interrupt request is not.

An TXI interrupt indicates that transmit data can be written and an RXI interrupt indicates that receive data is stored in the FRDR register.

Note 1. It is equivalent to 1 and half frames of 8-bit format with one stop bit (ETU: Element Time Unit).

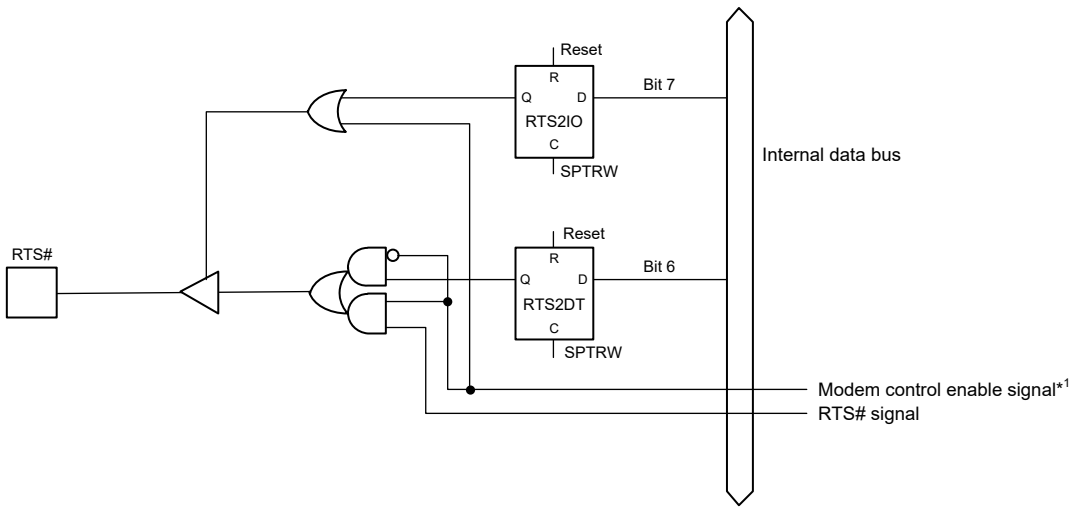
Table 23.19 SCIFA Interrupt Sources

Name	Level/Edge	Interrupt Source	Interrupt Enable Bit	DMAC Activation	Priority
BRI	Level	Interrupt caused by break (BRK) or overrun (ORER).	RIE or REIE	Impossible	High
ERI	Level	Interrupt caused by framing or parity (ER).	RIE or REIE	Impossible	↕
RXI	Level	Interrupt caused by receive FIFO data full (RDF).	RIE	Possible	
TXI	Level	Interrupt caused by transmit FIFO data empty (TDFE).	TIE	Possible	
TEI	Level	Interrupt caused by transmit end (TEND).	TEIE	Impossible	
DRI	Level	Interrupt caused by receive data ready (DR).	RIE	Impossible	Low

Note: The TEI and DRI interrupts share the same vector number.
The ERI and BRI interrupts share the same vector number.
If CPU processing is used, clear the flag after the block transfer. If the DMAC is activated, access to the flags is prohibited.

23.6 Serial Port Register (SPTR) and SCIFA-Related Pins

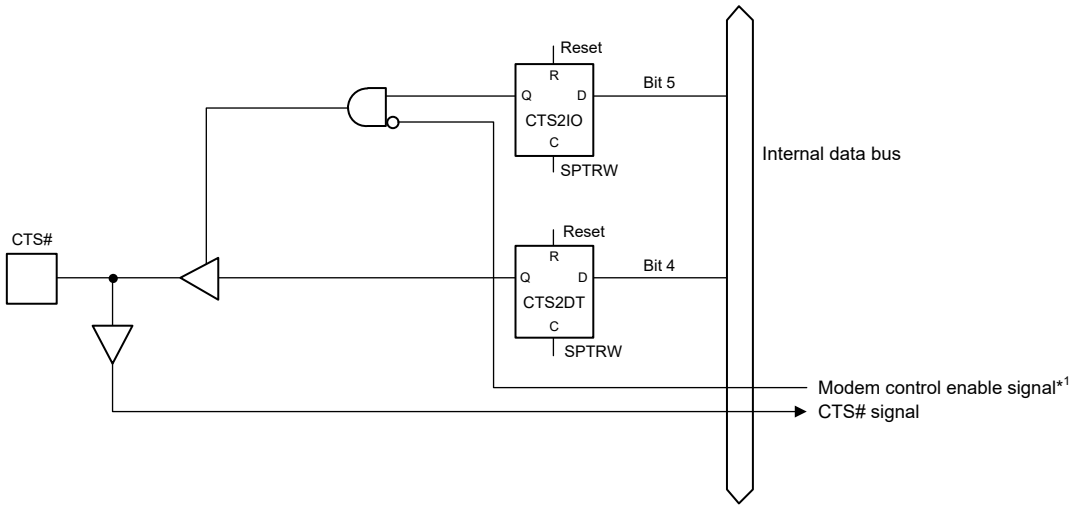
Figure 23.19 to **Figure 23.22** show the relationships between the SPTR register and the SCIFA-related pins.



Note: SPTRW: SPTR write

Note 1. These signals allow the RTS# pin to enter the modem control state with the MCE bits in FCR.

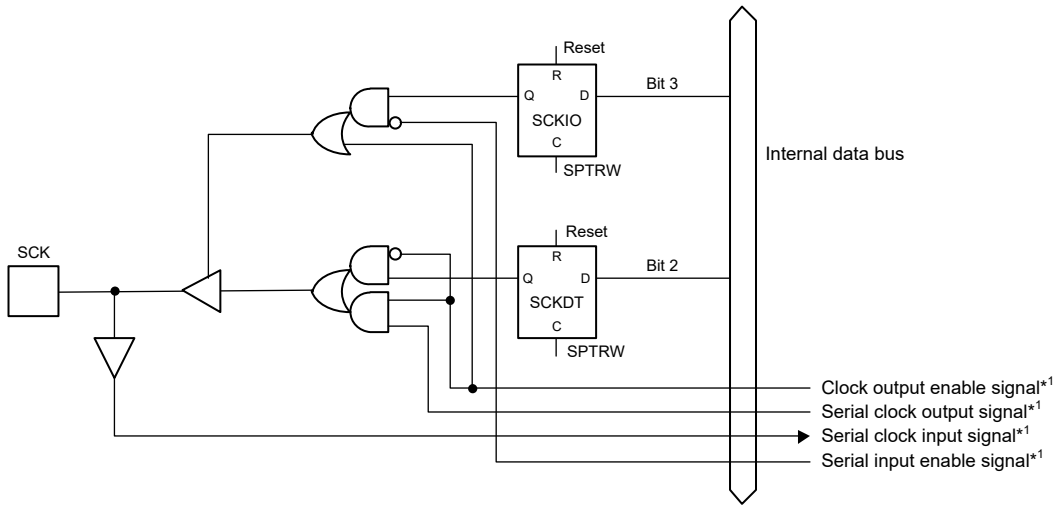
Figure 23.19 RTS2IO Bit and RTS2DT Bit in the SPTR Register, and RTS# Pin



Note: SPTRW: SPTR write

Note 1. These signals allow the CTS# pin to enter the modem control state with the MCE bits in FCR.

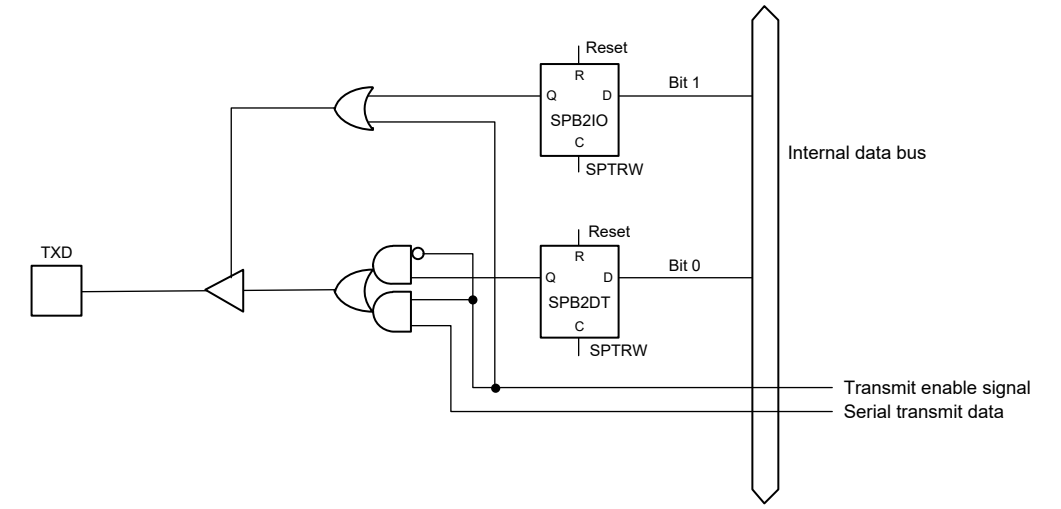
Figure 23.20 CTS2IO Bit and CTS2DT Bit in the SPTR Register, and CTS# Pin



Note: SPTRW: SPTR write

Note 1. These signals control the SCK pin according to the settings of the CM bit in SMR and CKE[1:0] bits in SCR.

Figure 23.21 SCKIO Bit and SCKDT Bit in the SPTR Register, and SCK Pin



Note: SPTRW: SPTR write

Figure 23.22 SPB2IO Bit and SPB2DT Bit in the SPTR Register, and TXD Pin

23.7 Noise Cancellation

Figure 23.23 shows the configuration of the noise filter used for noise cancellation. The noise filter consists of a two-stage flip-flop circuits and a match detection circuit. When the input signals of the noise filter and the output signals of the two-stage flip-flop circuits completely match, the matched level is conveyed as an internal signal. Unless otherwise matched, the previous value is retained. (When the levels sampled on three consecutive cycles of the sampling clock of the noise filter match, the signal is considered valid. If three consecutive sampled values do not match, the signal is considered to be noise rather than a received signal).

In asynchronous mode, the noise cancellation can be applied to the receive signal input to the RXDn pin. The receive level of the RXDn pin is taken in the flip-flop circuit of the noise filter on the base clock (the clock with a frequency 16 or 8 times the transfer rate*[1]).

If the base clock is stopped once with the noise filter enabled and then the base clock input is restarted again, the noise filter operation resumes from the state where the clock was stopped. When SCR.RE is set to 0 during input of the base clock, the noise filter outputs 0 as the internal RxDn signal. The internal match detector continues operating even while operations for reception are stopped, and the result from the last time previous consecutive samples matched is output at the same time as operations for reception are resumed.

Note 1. A frequency 16 times bit rate when the SEMR.ABCS0 bit and the SEMR.BGDM bit are both 0, and a frequency 8 times bit rate when either the SEMR.ABCS0 bit or the SEMR.BGDM bit is 1.

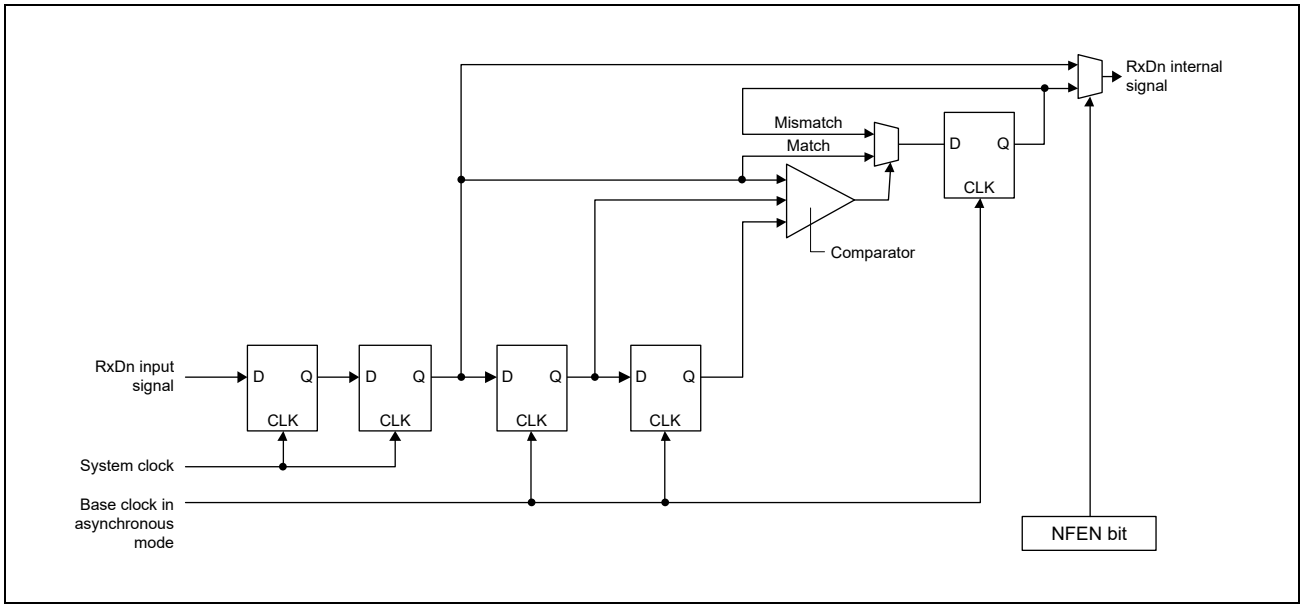


Figure 23.23 Block Diagram of Digital Noise Filter Circuit

23.8 Usage Notes

The following is the notes on using the SCIFA.

23.8.1 FTDR Register Writing and TDFE Flag

The TDFE flag in the serial status register (FSR) is set when the number of transmit data bytes written in the transmit FIFO data register (FTDR) has fallen below the transmission trigger number set by bits TTRG[1:0] in the FIFO control register (FCR) or bits TFTC[4:0] in the FIFO trigger control register (FTCR). After the TDFE flag is set, transmit data up to the number of empty bytes in the FTDR register can be written, allowing efficient continuous transmission.

However, if the number of data bytes written in the FTDR register is equal to or less than the specified transmission trigger number, the TDFE flag will be set to 1 again even after being read as 1 and cleared to 0.

The number of transmit data bytes in the FTDR register can be checked by the 8 higher-order bits of the FIFO data count register (FDR).

23.8.2 FRDR Register Reading and RDF Flag

The RDF flag in the serial status register (FSR) is set when the number of receive data bytes in the receive FIFO data register (FRDR) has become equal to or greater than the reception trigger number set by bits RTRG[1:0] in the FIFO control register (FCR) or bits RFTC[4:0] in the FIFO trigger control register (FTCR). After the RDF flag is set, receive data equivalent to the trigger number can be read from the FRDR register, allowing efficient continuous reception.

However, if the number of data bytes in the FRDR register exceeds the reception trigger number, the RDF flag will be set to 1 again even after being read as 1 and cleared to 0.

The number of receive data bytes in the FRDR register can be checked by the 8 lower-order bits of the FIFO data count register (FDR).

23.8.3 Break Detection and Processing

When a framing error (FER) is detected, a break signal can be detected by reading the RXD pin value directly. In a break, the input from the RXD pin becomes all low. Therefore, the FER flag in the serial status register (FSR) is set to 1 and the parity error flag (PER) may also be set to 1.

Upon detection of a break signal, the SCIFA stops the received data transfer to the FRDR register but continues the receive operation.

23.8.4 Writing to the SPTR Register

b6, b4, b2, and b0 of the SPTR register respectively indicate the input status of their corresponding pins. (See the description of each bit of **Section 23.2.12, Serial Port Register (SPTR)** for details.)

Writings to these bits in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

23.8.5 Break Signal Transmission

The output signal from the TXD pin is determined by the SPB2IO bit and the SPB2DT bit in the serial port register (SPTR). The break signal can be sent by using these bits.

The TXD pin does not function as a transmit data output pin during the period from when the SCIFA is initialized to when the TE bit in the SCR register is set to 1 (transmission possible). The TXD pin status during this period is replaced by the SPB2DT bit value. Therefore, the SPB2IO and SPB2DT bits in the SPTR register must have been set to 1 (high output) at first (mark (high) status).

To transmit the break signal during serial transmission, set the SPB2IO bit in the SPTR register to 1, clear the SPB2DT bit to 0 (specify a low level), and then clear the TE bit in the SCR register to 0 (transmission stop). Clearing the TE bit to 0 initializes the transmitter regardless of the current transmission status, and outputs a low level from the TXD pin.

23.8.6 Receive Data Sampling Timing and Receive Margin in Asynchronous Mode

The SCIFA operates on a base clock with a frequency 16 times the transfer rate*[1]. In reception, the SCIFA internally latches the received data at the rising edge of the eighth base clock pulse*[1]. The timing is shown in **Figure 23.24**.

Note 1. This is an example when the SEMR.ABCS0 bit is 0. When the ABCS0 bit is 1, a frequency of 8 times the bit rate is used as a base clock and receive data is sampled at the rising edge of the 4th pulse of the base clock. The receive margin in asynchronous mode can therefore be expressed as shown in equation 1.

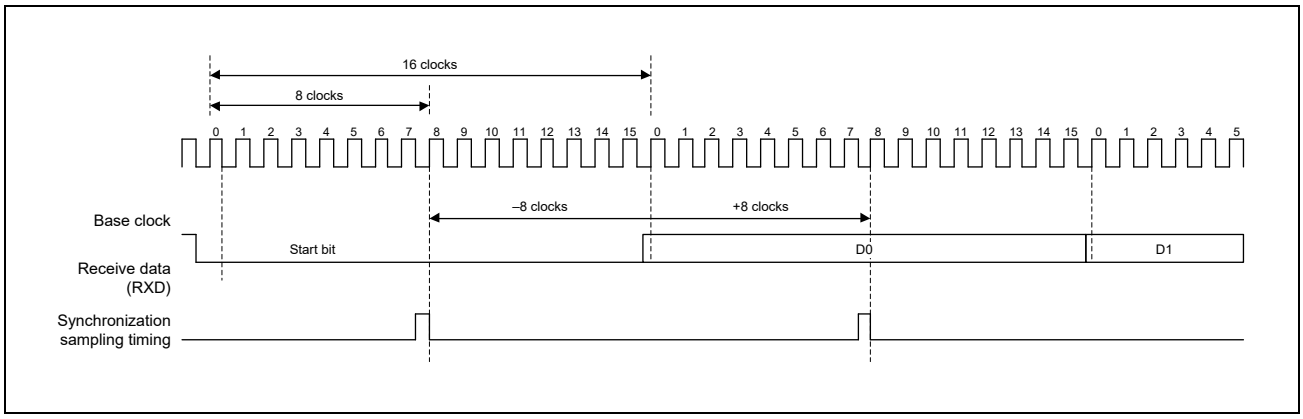


Figure 23.24 Receive Data Sampling Timing in Asynchronous Mode

The receive margin in asynchronous mode can therefore be expressed as shown in equation 1.

[Equation 1]

$$M = \left\{0.5 - \frac{1}{2N} - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F)\right\} \times 100[\%]$$

Where: M: Receive margin (%)

N: Ratio of clock frequency to bit rate (N = 16)

D: Clock duty (D = 0 to 1.0)

L: Frame length (L = 9 to 12)

F: Absolute deviation of clock frequency

From equation 1, if F = 0 and D = 0.5, the receive margin is 46.875%, as given by equation 2.

[Equation 2]

When D = 0.5 and F = 0:

$$M = \left(0.5 - \frac{1}{(2 \times 16)}\right) \times 100\% = 46.875\%$$

This is a theoretical value. A reasonable margin to allow in system designs is 20% to 30%.

23.8.7 Note on FER Flag and PER Flag in Serial Status Register (FSR)

The FER flag and PER flag in the serial status register (FSR) are status flags that apply to next entry to be read from the receive FIFO data register (FRDR). After the CPU or DMAC reads the receive FIFO data register, the flags of framing errors and parity errors in the receive data will be cleared. To check the received data for the states of framing errors and parity errors, only read the receive FIFO data register after reading the serial status register.

23.8.8 Notes on External Clock Input in Clock Synchronous Mode

Before setting the TE and RE bits in the serial control register (SCR) to 1, wait for four or more cycles of the peripheral operating clock after the external clock (SCK) is changed from 0 (low) to 1 (high). To input the external clock (SCK) (to start communication), wait for one or more cycles of the external clock after the TE and RE bits in the SCR register are set to 1.

23.8.9 Module Standby Mode Setting

SCIFA operation can be disabled or enabled using the standby control register. As the initial setting, the SCIFA operation is halted. Register access is enabled by clearing module standby mode. For details, refer to **Section 41, Low Power Mode**.

23.8.10 Notes on Operation for Reception when an Internal Clock is Selected in Clock Synchronous Mode

When an internal clock is selected as the clock for reception in clock-synchronous mode, if the number of data stored through the receive FIFO data register (FRDR) becomes equal to or greater than the specified reception trigger number, the RDF flag is set, the RXI interrupt request is generated and, at the same time, output of the synchronizing clock and reception of serial data are stopped. Once the number of data are again less than the specified reception trigger number, output of the synchronizing clock and the reception of serial data are restarted. In addition, if an internal clock is selected for reception in clock synchronous mode, the ORER flag is not set to 1 since no overrun occurs. Accordingly, overruns (indicated by the ORER flag) cannot be used as a BRI interrupt source.

24. Serial Communications Interface (SCIg)

This MCU has two independent serial communications interface (SCI) channels. The SCIs consist of the SCI0 and SCI1 modules, referred to in common as “SCIg modules”.

The SCIg modules (SCI0 and SCI1) can handle both asynchronous and clock synchronous serial communications. Asynchronous serial data communications can be carried out with standard asynchronous communications chips such as a Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communications Interface Adapter (ACIA). As an extended function in asynchronous communications mode, the SCI also supports smart card (IC card) interfaces conforming to ISO/IEC 7816-3 (standard for Identification Cards).

24.1 Overview

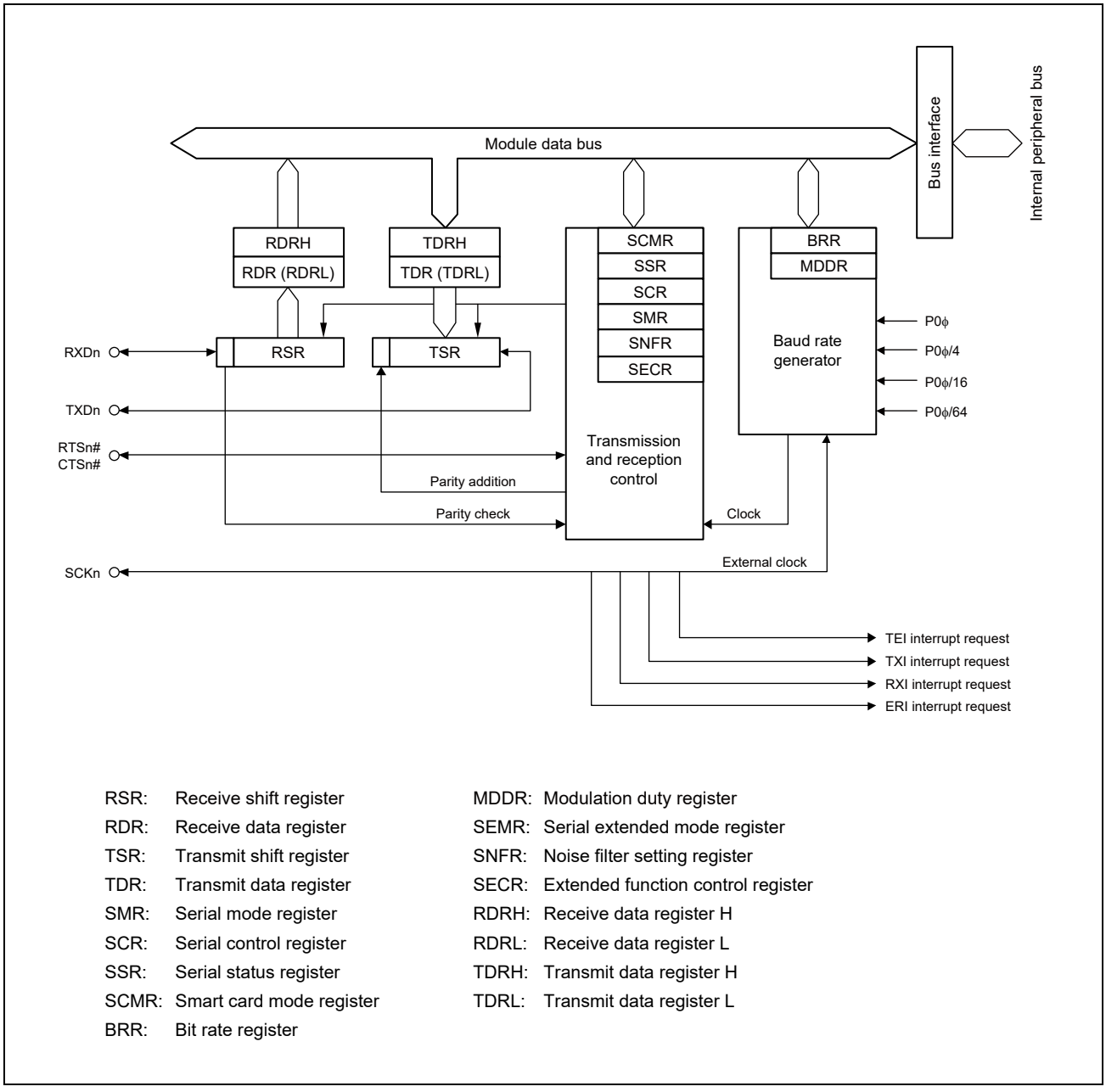
Table 24.1 lists the specifications of the SCIg modules. **Figure 24.1** shows the block diagram of the SCI0 and SCI1 modules.

Table 24.1 SCIg Specifications (1/2)

Item		Description
Serial communications modes		• Asynchronous • Clock synchronous • Smart card interface • IrDA (only for channel 0)
Transfer speed		Bit rate specifiable with the on-chip baud rate generator.
Full-duplex communications		Transmitter: Continuous transmission possible using double-buffer structure. Receiver: Continuous reception possible using double-buffer structure.
I/O pins		See **Table 24.2**.
Data transfer		Selectable as LSB first or MSB first transfer
Interrupt sources		Transmit end, transmit data empty, receive data full, and receive error
Low power consumption function		Module stop state can be set for each channel.
Asynchronous mode	Data length	7, 8, or 9 bits
	Transmission stop bit	1 or 2 bits
	Parity	Even parity, odd parity, or no parity
	Receive error detection	Parity, overrun, and framing errors
	Hardware flow control	CTSn# and RTSn# pins can be used in controlling transmission/reception.
	Start-bit detection	Low level or falling edge is selectable.
	Break detection	When a framing error occurs, a break can be detected by reading the RXDn pin level directly.
	Clock source	An internal or external clock can be selected.
	Double-speed mode	Baud rate generator double-speed mode is selectable.
	Multi-processor communications function	Serial communication among multiple processors
	Noise cancellation	The signal paths from input on the RXDn pins incorporate digital noise filters.
Clock synchronous mode	Data length	8 bits
	Receive error detection	Overrun error
	Hardware flow control	CTSn# and RTSn# pins can be used in controlling transmission/reception.

Table 24.1 SCIg Specifications (2/2)

Item		Description
Smart card interface mode	Error processing	An error signal can be automatically transmitted when detecting a parity error during reception
		Data can be automatically retransmitted when receiving an error signal during transmission
	Data type	Both direct convention and inverse convention are supported.
Bit rate modulation function		Correction of outputs from the on-chip baud rate generator can reduce errors.



RSR: Receive shift register
RDR: Receive data register
TSR: Transmit shift register
TDR: Transmit data register
SMR: Serial mode register
SCR: Serial control register
SSR: Serial status register
SCMR: Smart card mode register
BRR: Bit rate register
MDDR: Modulation duty register
SEMR: Serial extended mode register
SNFR: Noise filter setting register
SECR: Extended function control register
RDRH: Receive data register H
RDRL: Receive data register L
TDRH: Transmit data register H
TDRL: Transmit data register L

Figure 24.1 Block Diagram of SCIg (SCI0 and SCI1)

Table 24.2 lists the pin configuration of the SCIs for the individual modes.

Table 24.2 SCI Pin Configuration in Asynchronous Mode and Clock Synchronous Mode

Channel	Pin Name	I/O	Function
SCI0	SCI0_SCK	I/O	SCI0 clock input/output
	SCI0_RXD	Input	SCI0 receive data input
	SCI0_TXD	Output	SCI0 transmit data output
	SCI0_CTS#/RTS#	I/O	SCI0 transfer start control input/output
SCI1	SCI1_SCK	I/O	SCI1 clock input/output
	SCI1_RXD	Input	SCI1 receive data input
	SCI1_TXD	Output	SCI1 transmit data output
	SCI1_CTS#/RTS#	I/O	SCI1 transfer start control input/output

24.2 Register Descriptions

BASE Address (Cortex A55 Address Space): (ch0) H'0_1004_D000 (ch1) H'0_1004_D400
BASE Address (Cortex M33/Cortex-M33_FPU address Space Secure): (ch0) H'4004_D000 (ch1) H'4004_D400
BASE Address (Cortex M33/Cortex-M33_FPU Address Space Non-Secure): (ch0) H'5004_D000 (ch1) H'5004_D400

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 24.3 Register Configuration

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
0	Serial Mode Register	SMR	R/W	H'00	BASE (ch0) + H'0000	8
	Bit Rate Register	BRR	R/W	H'FF	BASE (ch0) + H'0001	8
	Serial Control Register	SCR	R/W	H'00	BASE (ch0) + H'0002	8
	Transmit Data Register	TDR	R/W	H'FF	BASE (ch0) + H'0003	8
	Serial Status Register	SSR	R/W*1	H'84	BASE (ch0) + H'0004	8
	Receive Data Register	RDR	R	H'00	BASE (ch0) + H'0005	8
	Smart Card Mode Register	SCMR	R/W*1	H'F2	BASE (ch0) + H'0006	8
	Serial Extended Mode Register	SEMR	R	H'00	BASE (ch0) + H'0007	8
	Noise Filter Setting Register	SNFR	R	H'00	BASE (ch0) + H'0008	8
	Extended Function Control Register	SECR	R/W	H'00	BASE (ch0) + H'000D	8
	Transmit Data Register H	TDRH	R/W	H'FF	BASE (ch0) + H'000E	8
	Transmit Data Register L	TDRL	R/W	H'FF	BASE (ch0) + H'000F	8
	Transmit Data Register HL	TDRHL	R/W	H'FFFF	BASE (ch0) + H'000E	16
	Receive Data Register H	RDRH	R	H'00	BASE (ch0) + H'0010	8
	Receive Data Register L	RDRL	R	H'00	BASE (ch0) + H'0011	8
	Receive Data Register HL	RDRHL	R	H'0000	BASE (ch0) + H'0010	16
	Modulation Duty Register	MDDR	R/W	H'FF	BASE (ch0) + H'0012	8
1	Serial Mode Register	SMR	R/W	H'00	BASE (ch1) + H'0000	8
	Bit Rate Register	BRR	R/W	H'FF	BASE (ch1) + H'0001	8
	Serial Control Register	SCR	R/W	H'00	BASE (ch1) + H'0002	8
	Transmit Data Register	TDR	R/W	H'FF	BASE (ch1) + H'0003	8
	Serial Status Register	SSR	R/W*1	H'84	BASE (ch1) + H'0004	8
	Receive Data Register	RDR	R	H'00	BASE (ch1) + H'0005	8
	Smart Card Mode Register	SCMR	R/W*1	H'F2	BASE (ch1) + H'0006	8
	Serial Extended Mode Register	SEMR	R	H'00	BASE (ch1) + H'0007	8
	Noise Filter Setting Register	SNFR	R	H'00	BASE (ch1) + H'0008	8
	Extended Function Control Register	SECR	R/W	H'00	BASE (ch1) + H'000D	8
	Transmit Data Register H	TDRH	R/W	H'FF	BASE (ch1) + H'000E	8
	Transmit Data Register L	TDRL	R/W	H'FF	BASE (ch1) + H'000F	8
	Transmit Data Register HL	TDRHL	R/W	H'FFFF	BASE (ch1) + H'000E	16
	Receive Data Register H	RDRH	R	H'00	BASE (ch1) + H'0010	8
	Receive Data Register L	RDRL	R	H'00	BASE (ch1) + H'0011	8
	Receive Data Register HL	RDRHL	R	H'0000	BASE (ch1) + H'0010	16
	Modulation Duty Register	MDDR	R/W	H'FF	BASE (ch1) + H'0012	8

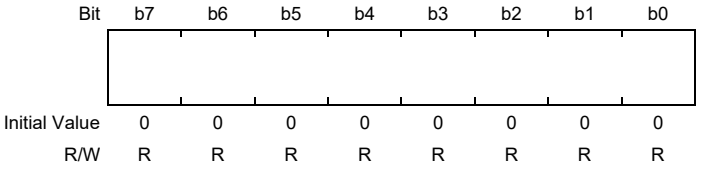
Note 1. Partly, R only

24.2.1 Receive Shift Register (RSR)

RSR is a shift register which is used to receive serial data input from the RXDn pin and converts it into parallel data. When one frame of data has been received, it is automatically transferred to the RDR register.

The RSR register cannot be directly accessed by the CPU.

24.2.2 Receive Data Register (RDR)



Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

RDR is an 8-bit register that stores receive data.

When one frame of serial data has been received, the received serial data is transferred from RSR to RDR. Then the RSR register can receive the next data.

Since RSR and RDR function as a double buffer in this way, continuous receive operations can be performed.

Read RDR only once after a receive data full interrupt (RXI) has occurred. Note that if next one frame of data is received before reading receive data from RDR, an overrun error occurs.

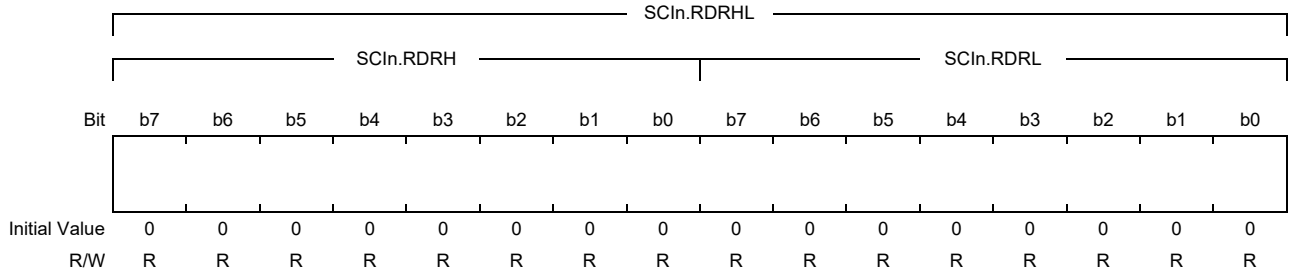
RDR cannot be written to by the CPU.

24.2.3 Receive Data Register H, L, HL (RDRH, RDRL, RDRHL)

Receive Data Register H (RDRH)

Receive Data Register L (RDRL)

Receive Data Register HL (RDRHL)



RDRH and RDRL are 8-bit registers that store receive data. Use these registers when asynchronous mode and 9-bit data length are selected.

RDRL is the shadow register of RDR; i.e. access to RDRL is equivalent to access to RDR.

After one frame of data is received, the received data is transferred from the RSR register to these registers, thus allowing the RSR register to receive the next data.

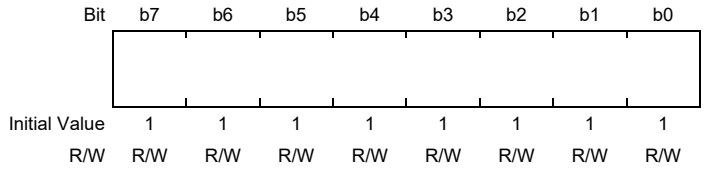
The RSR, RDRH and RDRL registers have a double-buffered construction to enable continuous reception.

Read RDRH and RDRL should be performed only once in the order from RDRH to RDRL when a receive data full interrupt (RXI) request is issued. Note that an overrun error occurs when the next frame of data is received before the received data has been read from RDRL.

The CPU cannot write to the RDRH and RDRL registers. Bits 0 to 7 in RDRH are fixed to 0. These bits are read as 0. The write value should be 0.

The RDRHL register can be accessed in 16-bit units.

24.2.4 Transmit Data Register (TDR)



TDR is an 8-bit register that stores transmit data.

When the SCI detects that TSR is empty, it transfers the transmit data written in TDR to TSR and starts transmission. The double-buffered structures of TDR and TSR enable continuous serial transmission. If the next transmit data has already been written to TDR when one frame of data is transmitted, the SCI transfers the written data to TSR to continue transmission.

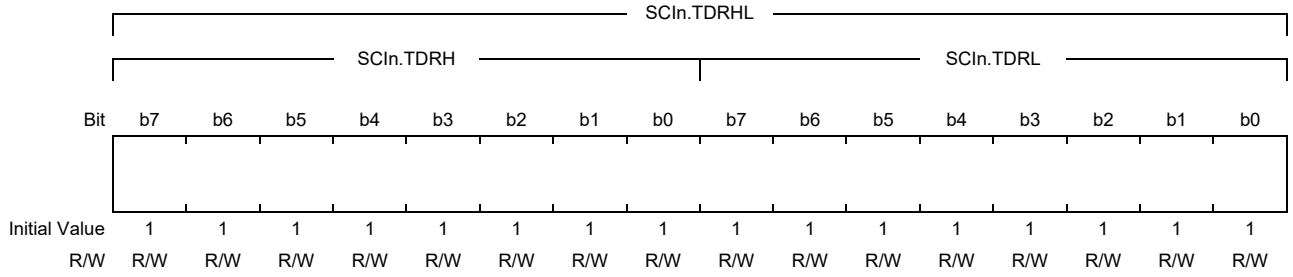
The CPU is able to read from or write to TDR at any time. Only write transmit data to TDR once after each instance of the transmit data empty interrupt (TXI).

24.2.5 Transmit Data Register H, L, HL (TDRH, TDRL, TDRHL)

Transmit Data Register H (TDRH)

Transmit Data Register L (TDRL)

Transmit Data Register HL (TDRHL)



TDRH and TDRL are 8-bit registers that store transmit data. Use these registers when asynchronous mode and 9-bit data length are selected.

TDRL is the shadow register of TDR; i.e. access to TDRL is equivalent to access to TDR.

When empty space is detected in the TSR register, the transmit data stored in the TDRH and TDRL registers is transferred to TSR; i.e., transmitting is started.

The TSR, TDRH and TDRL registers have a double-buffered construction to realize continuous reception. When the next data to be transmitted is stored in TDRL after one frame of data has been transmitted, the transmitting operation is continued by transfer to the TSR register.

The CPU can read and write to the TDRH and TDRL registers. Bits 0 to 7 in RDRH are fixed to 1. These bits are read as 1. The write value should be 1.

Writing transmit data to the TDRH and TDRL registers should be performed only once in the order from TDRH to TDRL when a transmit data empty interrupt (TXI) request is issued.

The TDRHL register can be accessed in 16-bit units.

24.2.6 Transmit Shift Register (TSR)

TSR is a shift register that transmits serial data.

To perform serial data transmission, the SCI first automatically transfers transmit data from TDR to TSR, and then sends the data to the TXDn pin.

TSR cannot be directly accessed by the CPU.

24.2.7 Serial Mode Register (SMR)

Remark Some bits in SMR have different functions in smart card interface mode and non-smart card interface mode.

(1) Non-Smart Card Interface Mode (SCMR.SMIF = 0)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	CM	CHR	PE	PM	STOP	MP	CKS[1:0]	
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W*4	R/W*4	R/W*4	R/W*4	R/W*4	R/W*4	R/W*4	R/W*4

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	CKS[1:0]	00b	R/W*4	Clock Select b1 b0 0 0: P0ϕ clock (n = 0)*1 0 1: P0ϕ/4 clock (n = 1)*1 1 0: P0ϕ/16 clock (n = 2)*1 1 1: P0ϕ/64 clock (n = 3)*1
b2	MP	0b	R/W*4	Multi-Processor Mode (Valid only in asynchronous mode) 0: Multi-processor communications function is disabled 1: Multi-processor communications function is enabled
b3	STOP	0b	R/W*4	Stop Bit Length (Valid only in asynchronous mode) 0: 1 stop bit 1: 2 stop bits
b4	PM	0b	R/W*4	Parity Mode (Valid only when the PE bit is 1) 0: Selects even parity 1: Selects odd parity
b5	PE	0b	R/W*4	Parity Enable (Valid only in asynchronous mode) • When transmitting 0: Parity bit addition is not performed 1: The parity bit is added • When receiving 0: Parity bit checking is not performed 1: The parity bit is checked
b6	CHR	0b	R/W*4	Character Length (Valid only in asynchronous mode*2) Selects in combination with the SCMR.CHR1 bit. CHR1 CHR 0 0: Transmit/receive in 9-bit data length 0 1: Transmit/receive in 9-bit data length 1 0: Transmit/receive in 8-bit data length (initial value) 1 1: Transmit/receive in 7-bit data length*3
b7	CM	0b	R/W*4	Communications Mode 0: Asynchronous mode 1: Clock synchronous mode

Note 1. n is the decimal notation of the value of n in BRR (refer to **Section 24.2.11, Bit Rate Register (BRR)**).

Note 2. In other than asynchronous mode, this bit setting is invalid and a fixed data length of 8 bits is used.

Note 3. LSB first is fixed and the MSB (bit 7) in TDR is not transmitted in transmission.

Note 4. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

CKS[1:0] Bits (Clock Select)

These bits select the clock source for the on-chip baud rate generator.

For the relation between the settings of these bits and the baud rate, refer to **Section 24.2.11, Bit Rate Register (BRR)**.

MP Bit (Multi-Processor Mode)

Disables/enables the multi-processor communications function. The settings of the PE bit and PM bit are invalid in multi-processor mode.

STOP Bit (Stop Bit Length)

Selects the stop bit length in transmission.

In reception, only the first stop bit is checked regardless of this bit setting. If the second stop bit is 0, it is treated as the start bit of the next transmit frame.

PM Bit (Parity Mode)

Selects the parity mode (even or odd) for transmission and reception. The setting of the PM bit is invalid in multi-processor mode.

PE Bit (Parity Enable)

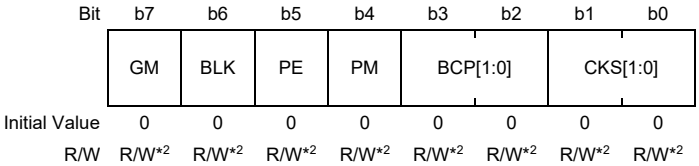
When this bit is set to 1, the parity bit is added to transmit data, and the parity bit is checked in reception. Irrespective of the setting of the PE bit, the parity bit is not added or checked in multi-processor format.

CHR Bit (Character Length)

Selects the data length for transmission and reception. Selects in combination with the CHR1 bit in SCMR.

In other than asynchronous mode, a fixed data length of 8 bits is used.

(2) Smart Card Interface Mode (SCMR.SMIF = 1)



Bit	Bit Name	Initial Value	R/W	Description
b1, b0	CKS[1:0]	00b	R/W*2	Clock Select b1 b0 0 0: P0ϕ clock (n = 0)*1 0 1: P0ϕ/4 clock (n = 1)*1 1 0: P0ϕ/16 clock (n = 2)*1 1 1: P0ϕ/64 clock (n = 3)*1
b3, b2	BCP[1:0]	00b	R/W*2	Base Clock Pulse Selects the number of base clock cycles in combination with the SCMR.BCP2 bit. **Table 24.4** lists the combinations of the SCMR.BCP2 bit and SMR.BCP[1:0] bits.
b4	PM	0b	R/W*2	Parity Mode (Valid only when the PE bit is 1) 0: Selects even parity 1: Selects odd parity
b5	PE	0b	R/W*2	Parity Enable When this bit is set to 1, a parity bit is added to transmit data, and the parity of received data is checked. Set this bit to 1 in smart card interface mode.
b6	BLK	0b	R/W*2	Block Transfer Mode 0: Normal mode operation 1: Block transfer mode operation
b7	GM	0b	R/W*2	GSM Mode 0: Normal mode operation 1: GSM mode operation

Note 1. n is the decimal notation of the value of n in BRR (refer to **Section 24.2.11, Bit Rate Register (BRR)**).

Note 2. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

CKS[1:0] Bits (Clock Select)

These bits select the clock source for the on-chip baud rate generator.

For the relationship between the settings of these bits and the baud rate, refer to **Section 24.2.11, Bit Rate Register (BRR)**.

BCP[1:0] Bits (Base Clock Pulse)

These bits select the number of base clock cycles in a 1-bit data transfer time in smart card interface mode. Set these bits in combination with the SCMR.BCP2 bit.

For details, refer to **Section 24.6.4, Receive Data Sampling Timing and Reception Margin**.

Table 24.4 Combinations of the SCMR.BCP2 Bit and SMR.BCP[1:0] Bits

SCMR.BCP2 Bit	SMR.BCP[1:0] Bits		Number of Base Clock Cycles for 1-Bit Transfer Period
0	0	0	93 clock cycles (S = 93)*1
0	0	1	128 clock cycles (S = 128)*1
0	1	0	186 clock cycles (S = 186)*1
0	1	1	512 clock cycles (S = 512)*1
1	0	0	32 clock cycles (S = 32)*1 (Initial Value)
1	0	1	64 clock cycles (S = 64)*1
1	1	0	372 clock cycles (S = 372)*1
1	1	1	256 clock cycles (S = 256)*1

Note 1. S is the value of Base Clock Cycles for 1-bit Period in BRR (refer to **Section 24.2.11, Bit Rate Register (BRR), Table 24.8, Base Clock Settings in Smart Card Interface Mode**).

PM Bit (Parity Mode)

Selects the parity mode for transmission and reception (even or odd).

For details on the usage of this bit in smart card interface mode, refer to **Section 24.6.2, Data Format (Except in Block Transfer Mode)**.

PE Bit (Parity Enable)

Set the PE bit to 1.

The parity bit is added to transmit data before transmission, and the parity bit is checked in reception.

BLK Bit (Block Transfer Mode)

Setting this bit to 1 allows block transfer mode operation.

For details, refer to **Section 24.6.3, Block Transfer Mode**.

GM Bit (GSM Mode)

Setting this bit to 1 allows GSM mode operation.

In GSM mode, the SSR.TEND flag set timing is put forward to 11.0 etu (elementary time unit = 1-bit transfer time) from the start and the clock output control function is appended. For details, refer to **Section 24.6.6, Serial Data Transmission (Except in Block Transfer Mode)** and **Section 24.6.8, Clock Output Control**.

24.2.8 Serial Control Register (SCR)

Remark Some bits in SCR have different functions in smart card interface mode and non-smart card interface mode.

(1) Non-Smart Card Interface Mode (SCMR.SMIF = 0)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]	
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W*2	R/W*2	R/W	R/W	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	CKE[1:0]	00b	R/W*1	Clock Enable • For SCI0 and SCI1 (Asynchronous mode) b1 b0 0 0: On-chip baud rate generator The SCKn pin functions as I/O port. 0 1: On-chip baud rate generator The clock with the same frequency as the bit rate is output from the SCKn pin. 1 x: External clock The clock with a frequency 16 times the bit rate should be input from the SCKn pin. Input a clock signal with a frequency eight times the bit rate when the SEMR.ABCS bit is 1. (Clock synchronous mode) b1 b0 0 x: Internal clock The SCKn pin functions as the clock output pin. 1 x: External clock The SCKn pin functions as the clock input pin.
b2	TEIE	0b	R/W	Transmit End Interrupt Enable 0: A TEI interrupt request is disabled 1: A TEI interrupt request is enabled
b3	MPIE	0b	R/W	Multi-Processor Interrupt Enable (Valid in asynchronous mode when SMR.MP = 1) 0: Normal reception 1: When the data with the multi-processor bit set to 0 is received, the data is not read, and setting the status flags ORER and FER in SSR to 1 is disabled. When the data with the multi-processor bit set to 1 is received, the MPIE bit is automatically cleared to 0, and normal reception is resumed.
b4	RE	0b	R/W*2	Receive Enable 0: Serial reception is disabled 1: Serial reception is enabled
b5	TE	0b	R/W*2	Transmit Enable 0: Serial transmission is disabled 1: Serial transmission is enabled
b6	RIE	0b	R/W	Receive Interrupt Enable 0: RXI and ERI interrupt requests are disabled 1: RXI and ERI interrupt requests are enabled
b7	TIE	0b	R/W	Transmit Interrupt Enable 0: A TXI interrupt request is disabled 1: A TXI interrupt request is enabled

Remarks: x: Don't care

Note 1. Writable only when TE = 0 and RE = 0.

Note 2. 1 can be written only when TE = 0 and RE = 0, while the SMR.CM bit is 1. After setting TE or RE to 1, only 0 can be written to TE and RE. While the SMR.CM bit is 0 and the SIMR1.IICM bit is 0, writing is enabled under any condition.

CKE[1:0] Bits (Clock Enable)

These bits select the clock source and SCKn pin function.

The combination of the settings of these bits and of the SEMR.ACS0 bit sets the internal TMR clock.

TEIE Bit (Transmit End Interrupt Enable)

Enables or disables a TEI interrupt request.

A TEI interrupt request is disabled by setting the TEIE bit to 0.

MPIE Bit (Multi-Processor Interrupt Enable)

When this bit is set to 1 and the data with the multi-processor bit set to 0 is received, the data is not read and setting the status flags ORER and FER in SSR to 1 is disabled. When the data with the multi-processor bit set to 1 is received, the MPIE is automatically cleared to 0, and normal reception is resumed. For details, refer to **Section 24.4, Multi-Processor Communications Function**.

When the receive data includes the MPB bit is SSR set to 0, the receive data is not transferred from the RSR to the RDR, a receive error is not detected, and setting the flags ORER and FER to 1 is disabled.

When the receive data includes the MPB bit set to 1, the MPB bit is set to 1, the MPIE bit is automatically cleared to 0, the RXI and ERI interrupt requests are enabled (if the RIE bit in SCR is set to 1), and setting the flags ORER and FER to 1 is enabled.

MPIE should be set to 0 if multi-processor communications function is not to be used.

RE Bit (Receive Enable)

Enables or disables serial reception.

When this bit is set to 1, serial reception is started by detecting the start bit in asynchronous mode or the synchronous clock input in clock synchronous mode. Note that SMR should be set prior to setting the RE bit to 1 in order to designate the reception format.

Even if reception is halted by setting the RE bit to 0, the ORER, FER, and PER flags in SSR are not affected and the previous value is retained.

TE Bit (Transmit Enable)

Enables or disables serial transmission.

When this bit is set to 1, serial transmission is started by writing transmit data to TDR. Note that SMR should be set prior to setting the TE bit to 1 in order to designate the transmission format.

RIE Bit (Receive Interrupt Enable)

Enables or disables RXI and ERI interrupt requests.

An RXI interrupt request is disabled by setting the RIE bit to 0.

An ERI interrupt request can be canceled by reading 1 from the ORER, FER, or PER flag in SSR and then setting the flag to 0, or setting the RIE bit to 0.

(2) Smart Card Interface Mode (SCMR.SMIF = 1)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]	
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W*2	R/W*2	R/W	R/W	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	CKE[1:0]	00b	R/W*1	Clock Enable • When SMR.GM = 0 b1 b0 0 0: Output disabled (The SCKn pin is available for use as an I/O port according to the I/O port settings.) 0 1: Clock output 1 x: (Setting prohibited) • When SMR.GM = 1 b1 b0 0 0: Output fixed low x 1: Clock output 1 0: Output fixed high
b2	TEIE	0b	R/W	Transmit End Interrupt Enable This bit should be 0 in smart card interface mode.
b3	MPIE	0b	R/W	Multi-Processor Interrupt Enable This bit should be 0 in smart card interface mode.
b4	RE	0b	R/W*2	Receive Enable 0: Serial reception is disabled 1: Serial reception is enabled
b5	TE	0b	R/W*2	Transmit Enable 0: Serial transmission is disabled 1: Serial transmission is enabled
b6	RIE	0b	R/W	Receive Interrupt Enable 0: RXI and ERI interrupt requests are disabled 1: RXI and ERI interrupt requests are enabled
b7	TIE	0b	R/W	Transmit Interrupt Enable 0: A TXI interrupt request is disabled 1: A TXI interrupt request is enabled

Remarks: x: Don't care

Note 1. Writable only when TE = 0 and RE = 0.

Note 2. 1 can be written only when TE = 0 and RE = 0. After setting TE or RE to 1, only 0 can be written in TE and RE.

For details on interrupt requests, refer to **Section 24.8, Interrupt Sources**.

CKE[1:0] Bits (Clock Enable)

These bits control the clock output from the SCKn pin.

In GSM mode, clock output can be dynamically switched. For details, refer to **Section 24.6.8, Clock Output Control**.

TEIE Bit (Transmit End Interrupt Enable)

This bit should be 0 in smart card interface mode.

MPIE Bit (Multi-Processor Interrupt Enable)

This bit should be 0 in smart card interface mode.

RE Bit (Receive Enable)

Enables or disables serial reception.

When this bit is set to 1, serial reception is started by detecting the start bit. Note that SMR should be set prior to setting the RE bit to 1 in order to designate the reception format.

Even if reception is halted by setting the RE bit to 0, the ORER, FER, and PER flags in SSR are not affected and the previous value is retained.

TE Bit (Transmit Enable)

Enables or disables serial transmission.

When this bit is set to 1, serial transmission is started by writing transmit data to TDR. Note that SMR should be set prior to setting the TE bit to 1 in order to designate the transmission format.

RIE Bit (Receive Interrupt Enable)

Enables or disables RXI and ERI interrupt requests.

An RXI interrupt request is disabled by setting the RIE bit to 0.

An ERI interrupt request can be canceled by reading 1 from the ORER, FER, or PER flag in SSR and then setting the flag to 0, or setting the RIE bit to 0.

24.2.9 Serial Status Register (SSR)

Remark Some bits in SSR have different functions in smart card interface mode and non-smart card interface mode.

(1) Non-Smart Card Interface Mode (SCMR.SMIF = 0)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT
Initial Value	1	0	0	0	0	1	0	0
R/W	R/(W)*2	R/(W)*2	R/(W)*1	R/(W)*1	R/(W)*1	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	MPBT	0b	R/W	Multi-Processor Bit Transfer Sets the multi-processor bit for adding to the transmission frame 0: Data transmission cycles 1: ID transmission cycles
b1	MPB	0b	R	Multi-Processor Value of the multi-processor bit in the reception frame 0: Data transmission cycles 1: ID transmission cycles
b2	TEND	1b	R	Transmit End Flag 0: A character is being transmitted. 1: Character transfer has been completed.
b3	PER	0b	R/(W) *1	Parity Error Flag 0: No parity error occurred 1: A parity error has occurred
b4	FER	0b	R/(W) *1	Framing Error Flag 0: No framing error occurred 1: A framing error has occurred
b5	ORER	0b	R/(W) *1	Overrun Error Flag 0: No overrun error occurred 1: An overrun error has occurred
b6	RDRF	0b	R/(W) *2	Receive Data Full Flag 0: When data is transferred from RDR 1: When data has been received normally, and transferred from RSR to RDR
b7	TDRE	1b	R/(W) *2	Transmit Data Empty Flag 0: When data is transferred to TDR 1: When data is transferred from TDR to TSR

Note 1. Only 0 can be written to this bit, to clear the flag.

Note 2. Write 1 when writing is necessary.

MPB Bit (Multi-Processor)

Holds the value of the multi-processor bit in the reception frame. This bit does not change when the SCR.RE bit is 0.

TEND Flag (Transmit End Flag)

Indicates completion of transmission.

[Setting conditions]

- When the SCR.TE bit is set to 0 (serial transmission is disabled)
  When the SCR.TE bit is changed from 0 to 1, the TEND flag is not affected and retains the value 1.
- When the TDR register is not updated at the time of transmission of the tail-end bit of a character being transmitted

[Clearing condition]

- When transmit data are written to the TDR register while the SCR.TE bit is 1

PER Flag (Parity Error Flag)

Indicates that a parity error has occurred during reception in asynchronous mode and the reception ends abnormally.

[Setting condition]

- When a parity error is detected during reception
  Although receive data when the parity error occurs is transferred to RDR, no RXI interrupt request occurs. Note that when the PER flag is being set to 1, the subsequent receive data is not transferred to RDR.

[Clearing condition]

- When 0 is written to PER after reading PER = 1
  Even when the SCR.RE bit is set to 0 (serial reception is disabled), the PER flag is not affected and retains its previous value.

FER Flag (Framing Error Flag)

Indicates that a framing error has occurred during reception in asynchronous mode and the reception ends abnormally.

[Setting condition]

- When the stop bit is 0
  In 2-stop-bit mode, only the first stop bit is checked whether it is 1 but the second stop bit is not checked. Note that although receive data when the framing error occurs is transferred to RDR, no RXI interrupt request occurs. In addition, when the FER flag is being set to 1, the subsequent receive data is not transferred to RDR.

[Clearing condition]

- When 0 is written to FER after reading FER = 1
  Even when the SCR.RE bit is set to 0, the FER flag is not affected and retains its previous value.

ORER Flag (Overrun Error Flag)

Indicates that an overrun error has occurred during reception and the reception ends abnormally.

[Setting condition]

- When the next data is received before receive data is read from RDR
  In RDR, receive data prior to an overrun error occurrence is retained, but data received after the overrun error occurrence is lost. When the ORER flag is set to 1, subsequent serial reception cannot be performed. Note that, in clock synchronous mode, serial transmission also cannot continue.

[Clearing condition]

- When 0 is written to ORER after reading ORER = 1
  Even when the SCR.RE bit is set to 0, the ORER flag is not affected and retains its previous value.

RDRF Flag (Receive Data Full Flag)

Indicates whether RDR has received data.

[Setting condition]

- When data has been received normally, and transferred from RSR to RDR

[Clearing condition]

- When data is transferred from RDR

TDRE Flag (Transmit Data Empty Flag)

Indicates whether TDR has data to be transmitted.

[Setting condition]

- When data is transferred from TDR to TSR

[Clearing condition]

- When data is transferred to TDR

(2) Smart Card Interface Mode (SCMR.SMIF = 1)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TDRE	RDRF	ORER	ERS	PER	TEND	MPB	MPBT
Initial Value	1	0	0	0	0	1	0	0
R/W	R/(W)*2	R/(W)*2	R/(W)*1	R/(W)*1	R/(W)*1	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	MPBT	0b	R/W	Multi-Processor Bit Transfer This bit should be set to 0 in smart card interface mode.
b1	MPB	0b	R	Multi-Processor This bit is not used in smart card interface mode. It should be set to 0.
b2	TEND	1b	R	Transmit End Flag 0: A character is being transmitted. 1: Character transfer has been completed.
b3	PER	0b	R/(W) *1	Parity Error Flag 0: No parity error occurred 1: A parity error has occurred
b4	ERS	0b	R/(W) *1	Error Signal Status Flag 0: Low error signal not responded 1: Low error signal responded
b5	ORER	0b	R/(W) *1	Overrun Error Flag 0: No overrun error occurred 1: An overrun error has occurred
b6	RDRF	0b	R/(W) *2	Receive Data Full Flag 0: When data is transferred from RDR 1: When data has been received normally, and transferred from RSR to RDR
b7	TDRE	1b	R/(W) *2	Transmit Data Empty Flag 0: When data is transferred to TDR 1: When data is transferred from TDR to TSR

Note 1. Only 0 can be written to this bit, to clear the flag.

Note 2. Write 1 when writing is necessary.

TEND Flag (Transmit End Flag)

With no error signal from the receiving side, this bit is set to 1 when further data for transfer is ready to be transferred to the TDR register.

[Setting conditions]

- When the SCR.TE bit = 0 (serial transmission is disabled)
  When the SCR.TE bit is changed from 0 to 1, TEND flag is not affected and retains the value 1.
- When a specified period has elapsed after the latest transmission of 1 byte, the ERS flag is 0, and the TDR register is not updated
  The set timing is determined by register settings as listed below.
  When SMR.GM = 0 and SMR.BLK = 0, 12.5 etu after the start of transmission
  When SMR.GM = 0 and SMR.BLK = 1, 11.5 etu after the start of transmission
  When SMR.GM = 1 and SMR.BLK = 0, 11.0 etu after the start of transmission
  When SMR.GM = 1 and SMR.BLK = 1, 11.0 etu after the start of transmission

[Clearing condition]

- When transmit data are written to the TDR register while the SCR.TE bit is 1

PER Flag (Parity Error Flag)

Indicates that a parity error has occurred during reception in asynchronous mode and the reception ends abnormally.

[Setting condition]

- When a parity error is detected during reception
  Although receive data when the parity error occurs is transferred to RDR, no RXI interrupt request occurs. Note that when the PER flag is being set to 1, the subsequent receive data is not transferred to RDR.

[Clearing condition]

- When 0 is written to PER after reading PER = 1
  Even when the SCR.RE bit is set to 0 (serial reception is disabled), the PER flag is not affected and retains its previous value.

ERS Flag (Error Signal Status Flag)

[Setting condition]

- When a low error signal is sampled

[Clearing condition]

- When 0 is written to ERS after reading ERS = 1
  Even when the SCR.RE bit is set to 0, the ERS flag is not affected and retains its previous value.

ORER Flag (Overrun Error Flag)

Indicates that an overrun error has occurred during reception and the reception ends abnormally.

[Setting condition]

- When the next data is received before receive data is read from RDR
  In RDR, the receive data prior to an overrun error occurrence is retained, but data received following the overrun error occurrence is lost. When the ORER flag is set to 1, subsequent serial reception cannot be performed.

[Clearing condition]

- When 0 is written to ORER after reading ORER = 1
  Even when the SCR.RE bit is set to 0, the ORER flag is not affected and retains its previous value.

RDRF Flag (Receive Data Full Flag)

Indicates whether RDR has received data.

[Setting condition]

- When data has been received normally, and transferred from RSR to RDR

[Clearing condition]

- When data is transferred from RDR

TDRE Flag (Transmit Data Empty Flag)

Indicates whether TDR has data to be transmitted.

[Setting condition]

- When data is transferred from TDR to TSR

[Clearing condition]

- When data is transferred to TDR

24.2.10 Smart Card Mode Register (SCMR)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	BCP2	—	—	CHR1	SDIR	SINV	—	SMIF
Initial Value	1	1	1	1	0	0	1	0
R/W	R/W*1	R/W	R/W	R/W*1	R/W*1	R/W*1	R/W	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b0	SMIF	0b	R/W*1	Smart Card Interface Mode Select 0: Non-smart card interface mode (Asynchronous mode, clock synchronous mode,) 1: Smart card interface mode
b1	—	1b	R/W	Reserved This bit is read as 1. The write value should be 1.
b2	SINV	0b	R/W*1	Transmitted/Received Data Invert 0: TDR contents are transmitted as they are. Receive data is stored as it is in RDR. 1: TDR contents are inverted before being transmitted. Receive data is stored in inverted form in RDR.
b3	SDIR	0b	R/W*1	Transmitted/Received Data Transfer Direction This bit can be used in the following modes. • Smart card interface mode • Asynchronous mode (multi-processor mode) • Clock synchronous mode 0: Transfer with LSB first 1: Transfer with MSB first
b4	CHR1	1b	R/W*1	Character Length 1 (Only valid in asynchronous mode)*2 Selects in combination with the SMR.CHR bit. CHR1 CHR 0 0: Transmit/receive in 9-bit data length 0 1: Transmit/receive in 9-bit data length 1 0: Transmit/receive in 8-bit data length (initial value) 1 1: Transmit/receive in 7-bit data length*3
b6, b5	—	11b	R/W	Reserved These bits are read as 1. The write value should be 1.
b7	BCP2	1b	R/W*1	Base Clock Pulse 2 Selects the number of base clock cycles in combination with the SMR.BCP[1:0] bits. **Table 24.5** lists the combinations of the SCMR.BCP2 bit and SMR.BCP[1:0] bits.

Note 1. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

Note 2. The setting is invalid and a fixed data length of 8 bits is used in modes other than asynchronous mode.

Note 3. LSB first should be selected and the value of MSB (b7) in TDR cannot be transmitted.

SMIF Bit (Smart Card Interface Mode Select)

When this bit is set to 1, smart card interface mode is selected.

When this bit is set to 0, non-smart card interface mode, i.e., asynchronous mode (including multi-processor mode), clock synchronous mode is selected.

SINV Bit (Transmitted/Received Data Invert)

Inverts the transmit/receive data logic level. This bit does not affect the logic level of the parity bit. To invert the parity bit, invert the PM bit in SMR.

CHR1 bit (Character Length 1)

Selects the data length of transmit/receive data. Selects in combination with the CHR bit in SMR.

A fixed data length of 8 bits is used in modes other than asynchronous mode.

BCP2 Bit (Base Clock Pulse 2)

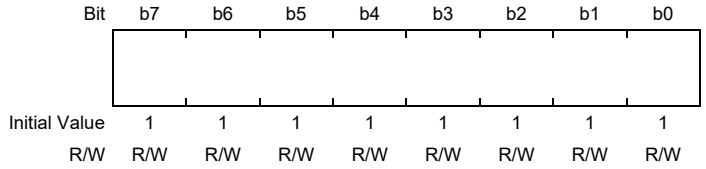
Selects the number of base clock cycles in a 1-bit data transfer time in smart card interface mode. Set this bit in combination with the SMR.BCP[1:0] bits.

Table 24.5 Combinations of the SCMR.BCP2 Bit and SMR.BCP[1:0] Bits

SCMR.BCP2 Bit	SMR.BCP[1:0] Bits		Number of Base Clock Cycles for 1-Bit Transfer Period
0	0	0	93 clock cycles (S = 93)*1
0	0	1	128 clock cycles (S = 128)*1
0	1	0	186 clock cycles (S = 186)*1
0	1	1	512 clock cycles (S = 512)*1
1	0	0	32 clock cycles (S = 32)*1 (Initial Value)
1	0	1	64 clock cycles (S = 64)*1
1	1	0	372 clock cycles (S = 372)*1
1	1	1	256 clock cycles (S = 256)*1

Note 1. S is the value of Base Clock Cycles for 1-bit Period BRR (refer to **Section 24.2.11, Bit Rate Register (BRR), Table 24.8 Base Clock Settings in Smart Card Interface Mode**).

24.2.11 Bit Rate Register (BRR)



BRR is an 8-bit register that adjusts the bit rate.

As each SCI channel has independent baud rate generator control, different bit rates can be set for each. **Table 24.6** shows the relationship between the setting (N) in the BRR and the bit rate (B) for normal asynchronous mode, multi-processor communication, clock synchronous mode, and smart card interface mode.

The initial value of BRR is H'FF.

BRR can be read from by the CPU, but it can be written to only when the TE and RE bits in SCR are 0.

Table 24.6 Relationship between N Setting in BRR and Bit Rate B

Mode	SEMR BGDM Bit	Setting ABCS Bit	BRR Setting	Error
Asynchro-nous or multi-processor communi-cation	0	0	$N = \frac{P0\phi \times 10^6}{64 \times 2^{2n-1} \times B} - 1$	$Error\ (\%) = \left\{\frac{P0\phi \times 10^6}{B \times 64 \times 2^{2n-1} \times (N+1)} - 1\right\} \times 100$
	0	1	$N = \frac{P0\phi \times 10^6}{32 \times 2^{2n-1} \times B} - 1$	$Error\ (\%) = \left\{\frac{P0\phi \times 10^6}{B \times 32 \times 2^{2n-1} \times (N+1)} - 1\right\} \times 100$
	1	0		
	1	1	$N = \frac{P0\phi \times 10^6}{16 \times 2^{2n-1} \times B} - 1$	$Error\ (\%) = \left\{\frac{P0\phi \times 10^6}{B \times 16 \times 2^{2n-1} \times (N+1)} - 1\right\} \times 100$
Clock synchronous			$N = \frac{P0\phi \times 10^6}{8 \times 2^{2n-1} \times B} - 1$	
Smart card interface			$N = \frac{P0\phi \times 10^6}{S \times 2^{2n-1} \times B} - 1$	$Error\ (\%) = \left\{\frac{P0\phi \times 10^6}{B \times S \times 2^{2n-1} \times (N+1)} - 1\right\} \times 100$

Remarks: B: Bit rate (bps)
N: BRR setting for on-chip baud rate generator (0 ≤ N ≤ 255)
P0ϕ: Operating frequency (MHz)
n and S: Determined by the settings of the SMR and SCMR registers as listed in the table below.

Table 24.7 Clock Source Settings

SMR.CKS[1:0] Bit Setting	Clock Source	n
0 0	P0ϕ clock	0
0 1	P0ϕ/4 clock	1
1 0	P0ϕ/16 clock	2
1 1	P0ϕ/64 clock	3

Table 24.8 Base Clock Settings in Smart Card Interface Mode

SCMR.BCP2 Bit Setting	SMR.BCP[1:0] Bit Setting	Base Clock Cycles for 1-bit Period	S
0	00	93 clock cycles	93
0	01	128 clock cycles	128
0	10	186 clock cycles	186
0	11	512 clock cycles	512
1	00	32 clock cycles	32
1	01	64 clock cycles	64
1	10	372 clock cycles	372
1	11	256 clock cycles	256

Table 24.9 lists examples of N settings in BRR in normal asynchronous mode. **Table 24.10** lists the maximum bit rate settable for each operating frequency. Examples of BRR (N) settings in clock synchronous mode are listed in **Table 24.12**. Examples of BRR (N) settings in smart card interface mode are listed in **Table 24.14**. In smart card interface mode, the number of base clock cycles S in a 1-bit data transfer time can be selected. For details, refer to **Section 24.6.4, Receive Data Sampling Timing and Reception Margin**. **Table 24.11** and **Table 24.13** list the maximum bit rates with external clock input.

When either the asynchronous mode base clock select bit (ABCS) or the baud rate generator double-speed mode select bit (BGDM) in the serial extended mode register (SEMR) is set to 1 in asynchronous mode, the bit rate becomes twice that listed in **Table 24.9**. When both of those registers are set to 1, the bit rate becomes four times the listed value.

Table 24.9 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode)

	Operating Frequency P0ϕ (MHz)		
	100		
Bit Rate (bps)	n	N	Error (%)
110	—	—	—
150	—	—	—
300	3	162	−0.15
600	3	80	0.47
1200	2	162	−0.15
2400	2	80	0.47
4800	1	162	−0.15
9600	1	80	0.47
19200	0	162	−0.15
31250	0	99	0.00
38400	0	80	0.47

Note: This is an example when the ABCS and BGDM bits in SEMR are 0.
When either the ABCS bit or BGDM bit is set to 1, the bit rate doubles.
When both ABCS and BGDM bits in SEMR are set to 1, the bit rate increases four times.

Table 24.10 Maximum Bit Rate for Each Operating Frequency (Asynchronous Mode)

	SEMR Settings				
P0ϕ (MHz)	BGDM Bit	ABCS Bit	n	N	Maximum Bit Rate (bps)
100	0	0	0	0	3125000
		1	0	0	6250000
	1	0	0	0	6250000
		1	0	0	12500000

Table 24.11 Maximum Bit Rate with External Clock Input (Asynchronous Mode)

	Maximum Bit Rate (bps)		
P0ϕ (MHz)	External Input Clock (MHz)	SEMR.ABCS Bit = 0	SEMR.ABCS Bit = 1
100	25.000	1562500	3125000

Table 24.12 BRR Settings for Various Bit Rates (Clock Synchronous Mode)

	Operating Frequency P0ϕ (MHz)	
	100	
Bit Rate (bps)	n	N
110	—	—
250	—	—
500	—	—
1 k	—	—
2.5 k	3	155
5 k	3	77
10 k	2	155
25 k	1	249
50 k	1	124
100 k	0	249
250 k	0	99
500 k	0	49
1 M	0	24
2.5 M	0	9
5 M	0	4
7.5 M	0	2

Note: Continuous transmission or reception is impossible. After transmitting or receiving one frame of data, there is an interval of a 1-bit period before starting transmitting/receiving the next frame of data. The output of the synchronization clock is stopped for a 1-bit period. For this reason, it takes 9 bits worth of time to transfer one frame (8 bits) of data, and the average transfer rate is 8/9 times the bit rate.

Remarks: Space: Setting prohibited.
—: Can be set, but an error will occur.

Table 24.13 Maximum Bit Rate with External Clock Input (Clock Synchronous Mode)

P0φ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (Mbps)
100	16.6667	16.6667

Table 24.14 BRR Settings for Various Bit Rates (Smart Card Interface Mode, n = 0, S = 372)

Bit Rate (bps)	P0φ (MHz)	n	N	Error (%)
9600	100	0	13	0.01

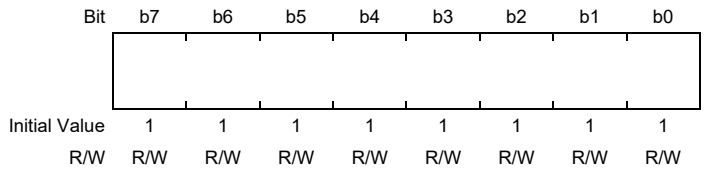
Table 24.15 Maximum Bit Rate for Each Operating Frequency (Smart Card Interface Mode, S = 32)

P0φ (MHz)	Maximum Bit Rate (bps)	n	N
100.00	1562500	0	0

Table 24.16 Maximum Bit Rate for Each Operating Frequency (Smart Card Interface Mode, S = 372)

P0φ (MHz)	Maximum Bit Rate (bps)	n	N
100.00	134409	0	0

24.2.12 Modulation Duty Register (MDDR)



MDDR corrects the bit rate adjusted by the BRR register.

When the BRME bit in SEMR is set to 1, the bit rate generated by the on-chip baud rate generator is evenly corrected according to the settings of MDDR (M/256). The relationship between the MDDR setting (M) and the bit rate (B) is given in **Table 24.17**.

The initial value of MDDR is H'FF. Bit 7 in this register is fixed to 1.

The CPU can read the MDDR register, but this register is only writable when the TE and RE bits in SCR are 0.

Table 24.17 Relationship between MDDR Setting (M) and Bit Rate (B)
When Bit Rate Modulation Function is Used

Mode	SEMR BGDM Bit	Setting ABCS Bit	BRR Setting	Error
Asynchro-nous or multi-processor communi-cation	0	0	$N = \frac{P0\phi \times 10^6}{64 \times 2^{2n-1} \times (256/M) \times B} - 1$	$Error\ (\%) = \left\{\frac{P0\phi \times 10^6}{B \times 64 \times 2^{2n-1} \times (256M) \times (N+1)} - 1\right\} \times 100$
	0	1	$N = \frac{P0\phi \times 10^6}{32 \times 2^{2n-1} \times (256/M) \times} - 1$	$Error\ (\%) = \left\{\frac{P0\phi \times 10^6}{B \times 32 \times 2^{2n-1} \times (256M) \times (N+1)} - 1\right\} \times 100$
	1	0		
	1	1	$N = \frac{P0\phi \times 10^6}{16 \times 2^{2n-1} \times (256/M) \times B} - 1$	$Error\ (\%) = \left\{\frac{P0\phi \times 10^6}{B \times 16 \times 2^{2n-1} \times (256M) \times (N+1)} - 1\right\} \times 100$
Clock synchronous			$N = \frac{P0\phi \times 10^6}{8 \times 2^{2n-1} \times (256/M) \times B} - 1$	
Smart card interface			$N = \frac{P0\phi \times 10^6}{S \times 2^{2n-1} \times (256/M) \times B} - 1$	$Error\ (\%) = \left\{\frac{P0\phi \times 10^6}{B \times S \times 2^{2n-1} \times (256M) \times (N+1)} - 1\right\} \times 100$

Note: Do not use this function in clock synchronous mode with the highest speed setting (SMR.CKS[1:0] = 00b, SCR.CKE[1] = 0, and BRR = 0).

Remarks: B: Bit rate (bps)
M: MDDR setting (128 ≤ MDDR ≤ 256)

N: BRR setting for baud rate generator (0 ≤ N ≤ 255)
P0ϕ: Operating frequency (MHz)
n and S: Determined by the settings of the SMR and SCMR registers as listed in **Table 24.7** and **Table 24.8**, **Section 24.2.11, Bit Rate Register (BRR)**.

24.2.13 Serial Extended Mode Register (SEMR)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	RXDESEL	BGDM	NFEN	ABCS	—	BRME	—	ACS0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W*1	R/W	R/W*1	R/W*1	R/W	R/W	R/W	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b0	ACS0	0b	R/W*1	Asynchronous Mode Clock Source Select (Valid only in asynchronous mode) 0: External clock input 1: Reserved bit
b1	—	0b	R/W	Reserved This bit is read as 0. The write value should be 0.
b2	BRME	0b	R/W	Bit Rate Modulation Enable 0: Bit rate modulation function is disabled. 1: Bit rate modulation function is enabled.
b3	—	0b	R/W	Reserved This bit is read as 0. The write value should be 0.
b4	ABCS	0b	R/W*1	Asynchronous Mode Base Clock Select (Valid only in asynchronous mode) 0: Selects 16 base clock cycles for 1-bit period. 1: Selects 8 base clock cycles for 1-bit period.
b5	NFEN	0b	R/W*1	Digital Noise Filter Function Enable (In asynchronous mode) 0: Noise cancellation function for the RXDn input signal is disabled. 1: Noise cancellation function for the RXDn input signal is enabled.
b6	BGDM	0b	R/W	Baud Rate Generator Double-Speed Mode Select (Only valid the CKE[1] bit in SCR is 0 in asynchronous mode). 0: Baud rate generator outputs the clock with normal frequency. 1: Baud rate generator outputs the clock with doubled frequency.
b7	RXDESEL	0b	R/W*1	Asynchronous Start Bit Edge Detection Select (Valid only in asynchronous mode) 0: The low level on the RXDn pin is detected as the start bit. 1: A falling edge on the RXDn pin is detected as the start bit.

Note 1. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

SEMR selects the clock source for 1-bit period in asynchronous mode.

ACS0 Bit (Asynchronous Mode Clock Source Select)

Selects the clock source in the asynchronous mode.

The ACS0 bit is valid in asynchronous mode (CM bit in SMR = 0) and when an external clock input is selected (CKE[1:0] bits in SCR = 10b or 11b).

Set the ACS0 bit to 0 in other than asynchronous mode.

BRME bit (Bit Rate Modulation Enable)

Enables and disables the bit rate modulation function. The bit rate generated by on-chip baud rate generator is evenly corrected when this function is enabled.

NFEN Bit (Digital Noise Filter Function Enable)

This bit enables or disables the digital noise filter function.

When the function is enabled, noise cancellation is applied to the RXDn input signal in asynchronous mode. In any mode other than above, set the NFEN bit to 0 to disable the digital noise filter function.

When the function is disabled, input signals are transferred as is, as internal signals.

BGDM bit (Baud Rate Generator Double-Speed Mode Select)

Selects the cycle of output clock for the baud rate generator.

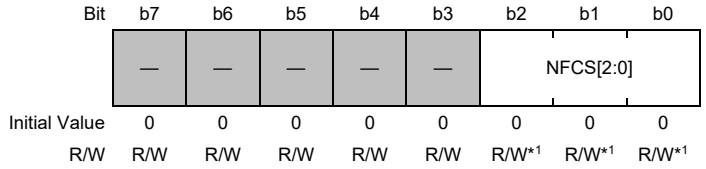
This bit is valid when the on-chip baud rate generator is selected as the clock source (SCR.CKE[1] = 0) in asynchronous mode (SMR.CM = 0). For the clock output from the baud rate generator, either normal or doubled frequency can be selected. The base clock is generated by the clock output from the baud rate generator. When the BGDM bit is set to 1, the base clock cycle is halved and the bit rate is doubled. Set this bit to 0 in modes other than asynchronous mode.

RXDESEL Bit (Asynchronous Start Bit Edge Detection Select)

Selects the detection method of the start bit for reception in asynchronous mode. When a break occurs, data receiving operation depends on the settings of this bit. Set this bit to 1 when reception should be stopped while a break occurs or when reception should be started without retaining the RXDn pin input at high level for the period of one data frame or longer after completion of the break.

Set this bit to 0 in modes other than asynchronous mode.

24.2.14 Noise Filter Setting Register (SNFR)



Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	NFCS[2:0]	All 0	R/W*1	Noise Filter Clock Select In asynchronous mode, the standard setting for the base clock is as follows. b2 b0 0 0 0: The clock signal divided by 1 is used with the noise filter. Settings other than above are prohibited.
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Writing to these bits is only possible when the RE and TE bits in the SCR are 0 (serial reception and transmission disabled).

NFCS[2:0] Bits (Noise Filter Clock Select)

These bits select the sampling clock for the digital noise filter. To use the noise filter in asynchronous mode, set these bits to 000b.

24.2.15 Extended function control register (SECR)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	CTSE	—
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	—	0b	R/W	Reserved This bit is read as 0. The write value should be 0.
b1	CTSE	0b	R/W*1	CTS Enable 0: CTS function is disabled (RTS output function is enabled). 1: CTS function is enabled.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Writing to the bit is only possible when the SCR.RE and SCR.TE bits are 0.

SECR is used to select the extension settings in asynchronous and clock synchronous modes.

CTSE Bit (CTS Enable)

Set this bit to 1 if the SSn# pin is to be used for inputting of the CTS control signal to control of transmission and reception. The RTS signal is output when this bit is set to 0. Set this bit to 0 in smart card interface mode. Do not set both the CTSE and SSE bits to enabled (even if this setting is made, operation is the same as that when these bits are set to 0).

24.3 Operation in Asynchronous Mode

Figure 24.2 shows the general format for asynchronous serial communications.

One frame consists of a start bit (low level), transmit/receive data, a parity bit, and stop bits (high level).

In asynchronous serial communications, the communications line is usually held in the mark state (high level).

The SCI monitors the communications line. When the SCI detects a low, it regards that as a start bit and starts serial communication.

Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communications. Both the transmitter and the receiver also have a double-buffered structure, so that data can be read or written during transmission or reception, enabling continuous data transmission and reception.

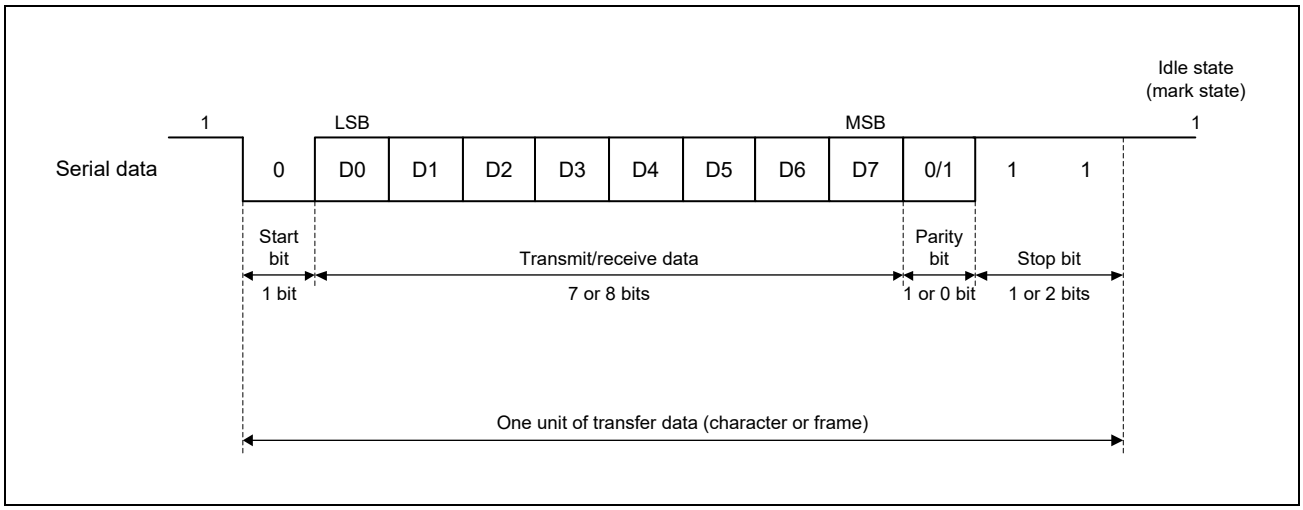


Figure 24.2 Data Format in Asynchronous Serial Communications (Example with 8-Bit Data, Parity, 2 Stop Bits)

24.3.1 Serial Data Transfer Format

Table 24.18 lists the serial data transfer formats that can be used in asynchronous mode.

Any of 18 transfer formats can be selected according to the SMR and SCMR setting. For details of multi-processor function, refer to **Section 24.4, Multi-Processor Communications Function**.

Table 24.18 Serial Transfer Formats (Asynchronous Mode)

SCMR Setting	SMR Setting				Serial Transfer Format and Frame Length												
CHR1	CHR	PE	MP	STOP	1	2	3	4	5	6	7	8	9	10	11	12	13
0	0	0	0	0	S	9-bit data									STOP		
0	0	0	0	1	S	9-bit data									STOP	STOP	
0	0	1	0	0	S	9-bit data									P	STOP	
0	0	1	0	1	S	9-bit data									P	STOP	STOP
1	0	0	0	0	S	8-bit data								STOP			
1	0	0	0	1	S	8-bit data								STOP	STOP		
1	0	1	0	0	S	8-bit data								P	STOP		
1	0	1	0	1	S	8-bit data								P	STOP	STOP	
1	1	0	0	0	S	7-bit data							STOP				
1	1	0	0	1	S	7-bit data							STOP	STOP			
1	1	1	0	0	S	7-bit data							P	STOP			
1	1	1	0	1	S	7-bit data							P	STOP	STOP		
0	0	—	1	0	S	9-bit data									MPB	STOP	
0	0	—	1	1	S	9-bit data									MPB	STOP	STOP
1	0	—	1	0	S	8-bit data								MPB	STOP		
1	0	—	1	1	S	8-bit data								MPB	STOP	STOP	
1	1	—	1	0	S	7-bit data							MPB	STOP			
1	1	—	1	1	S	7-bit data							MPB	STOP	STOP		

Note: S: Start bit
STOP: Stop bit
P: Parity bit
MPB: Multi-processor bit

24.3.2 Receive Data Sampling Timing and Reception Margin in Asynchronous Mode

In asynchronous mode, the SCI operates on a base clock with a frequency of 16 times*1 the bit rate.

In reception, the SCI samples the falling edge of the start bit using the base clock, and performs internal synchronization. Since receive data is sampled at the rising edge of the 8th pulse*1 of the base clock, data is latched at the middle of each bit, as shown in **Figure 24.3**. Thus the reception margin in asynchronous mode is determined by formula (1) below.

$$M = \left|(0.5 - \frac{1}{2N}) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F)\right| \times 100\ [\%] \cdots \text{Formula (1)}$$

M: Reception margin
N: Ratio of bit rate to clock (N = 16 when ABCS in SEMR = 0, N = 8 when ABCS in SEMR = 1)
D: Duty cycle of clock (D = 0.5 to 1.0)
L: Frame length (L = 9 to 13)
F: Absolute value of clock frequency deviation

Assuming values of F = 0 and D = 0.5 in formula (1), the reception margin is determined by the formula below.

$$M = \{0.5 - \frac{1}{(2 \times 16)}\} \times 100\ (\%) = 46.875\%$$

However, this is only the computed value, and a margin of 20% to 30% should be allowed in system design.

Note 1. This is an example when the ABCS bit in SEMR is 0. When the ABCS bit is 1, a frequency of 8 times the bit rate is used as a base clock and receive data is sampled at the rising edge of the 4th pulse of the base clock.

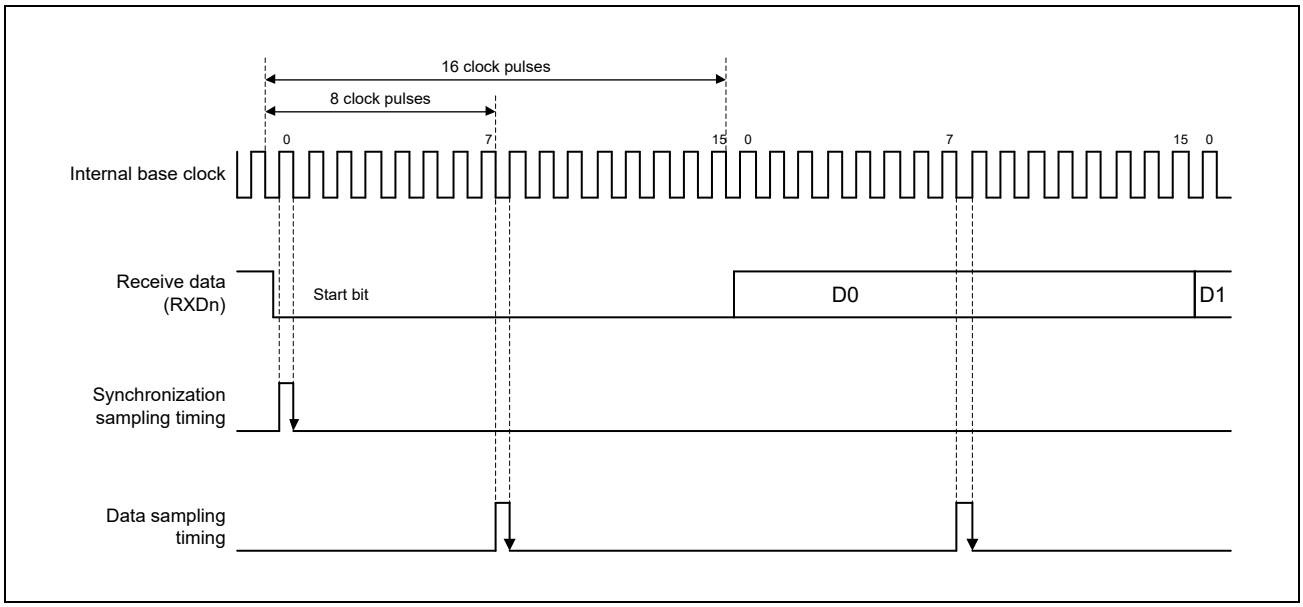


Figure 24.3 Receive Data Sampling Timing in Asynchronous Mode

24.3.3 Clock

Either an internal clock generated by the on-chip baud rate generator or an external clock input to the SCKn pin can be selected as the SCI's transfer clock, according to the setting of the CM bit in SMR and the CKE[1:0] bits in SCR. When an external clock is input to the SCKn pin, the clock frequency should be 16 times the bit rate (when ABCS in SEMR = 0) and 8 times the bit rate (when ABCS in SEMR = 1).

When the SCI is operated on an internal clock, the clock can be output from the SCKn pin. The frequency of the clock output in this case is equal to the bit rate, and the phase is such that the rising edge of the clock is in the middle of the transmit data, as shown in **Figure 24.4**.

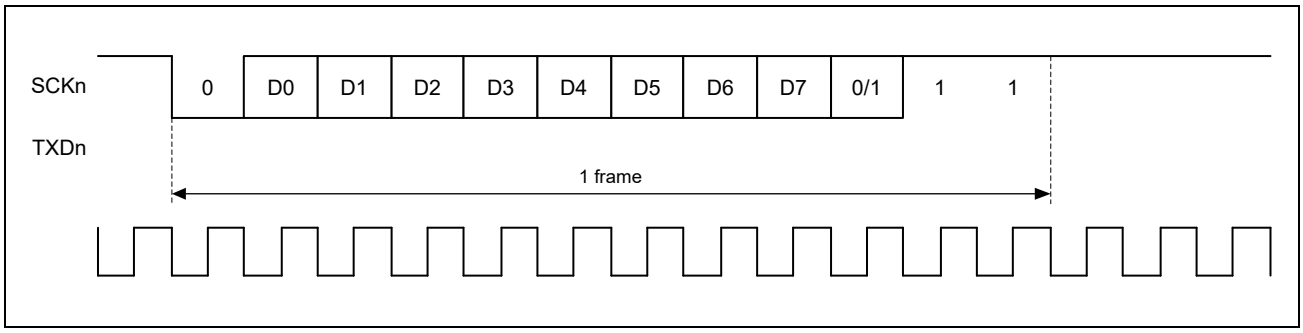


Figure 24.4 Phase Relationship between Output Clock and Transmit Data
(Asynchronous Mode: SMR.CHR = 0, PE = 1, MP = 0, STOP = 1)

24.3.4 Double-Speed Mode

The output clock frequency of the on-chip baud rate generator is doubled by setting the SEMR.BGDM bit to 1, enabling high-speed communication at a doubled bit rate. If the SEMR.ABCS bit is set to 1 under the above condition, the number of base clock cycles changes from 16 to 8, so the bit rate becomes four times faster than the initial state.

As shown by Formula (1) in **Section 24.3.2, Receive Data Sampling Timing and Reception Margin in Asynchronous Mode**, setting the SEMR.ABCS bit to 1 changes the number of cycles to 8, and the sampling interval becomes longer. This causes the reception margin to decrease. Therefore, setting the SEMR.BGDM bit to 1 and the SEMR.ABCS bit to 0 is recommended instead of setting the SEMR.BGDM bit to 0 and the SEMR.ABCS bit to 1 for high-speed operation at a doubled bit rate.

24.3.5 CTS and RTS Functions

The CTS function is the use of input on the CTSn# pin in transmission control. Setting the SECR.CTSE bit to 1 enables the CTS function. When the CTS function is enabled, placing the low level on the CTSn# pin causes transmission to start.

Applying the high level to the CTS# pin while transmission is in progress does not affect transmission of the current frame, which continues.

In the RTS function, by using the function of output on the RTSn# pin, a low level is output when reception becomes possible. Conditions for output of the low and high level are shown below.

[Conditions for low-level output]

Satisfaction of all conditions listed below

- The value of the RE bit in the SCR is 1
- Reception is not in progress
- There are no received data yet to be read
- The ORER, FER, and PER flags in the SSR are all 0

[Condition for high-level output]

- The conditions for low-level output have not been satisfied.

24.3.6 SCI Initialization (Asynchronous Mode)

Before transmitting and receiving data, start by writing the initial value H'00 to SCR and then continue through the procedure for SCI given in **Figure 24.5**. Whenever the operating mode or transfer format is changed, SCR must be initialized before the change is made.

When the external clock is used in asynchronous mode, ensure that the clock signal is supplied even during initialization. Note that setting the SCR.RE bit to 0 initializes neither the ORER, FER, and PER flags in SSR nor RDR, RDRH, and RDRL.

Moreover, note that switching the value of the SCR.TE bit from 1 to 0 or 0 to 1 while the SCR.TIE bit is 1 leads to the generation of a TXI interrupt request.

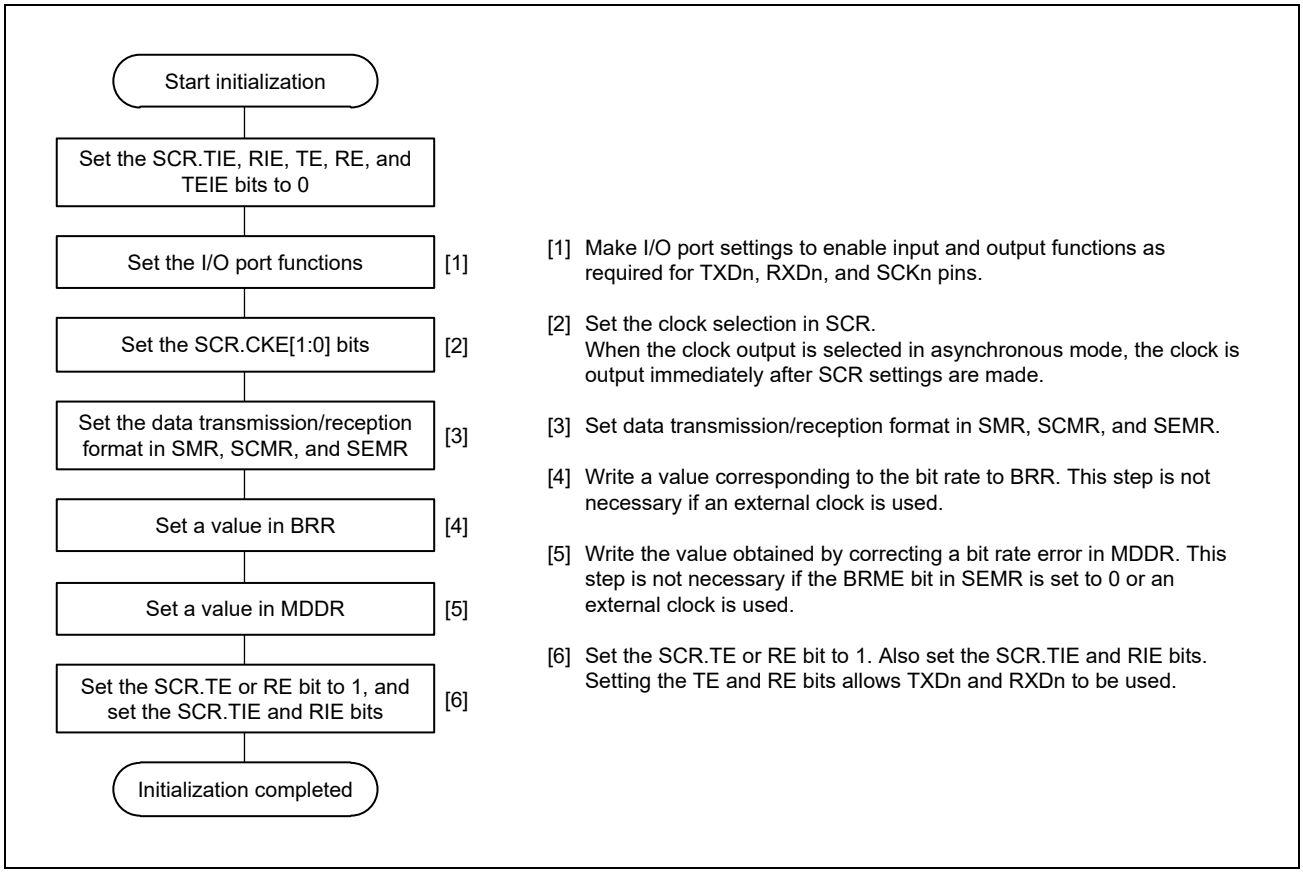


Figure 24.5 Sample SCI Initialization Flowchart (Asynchronous Mode)

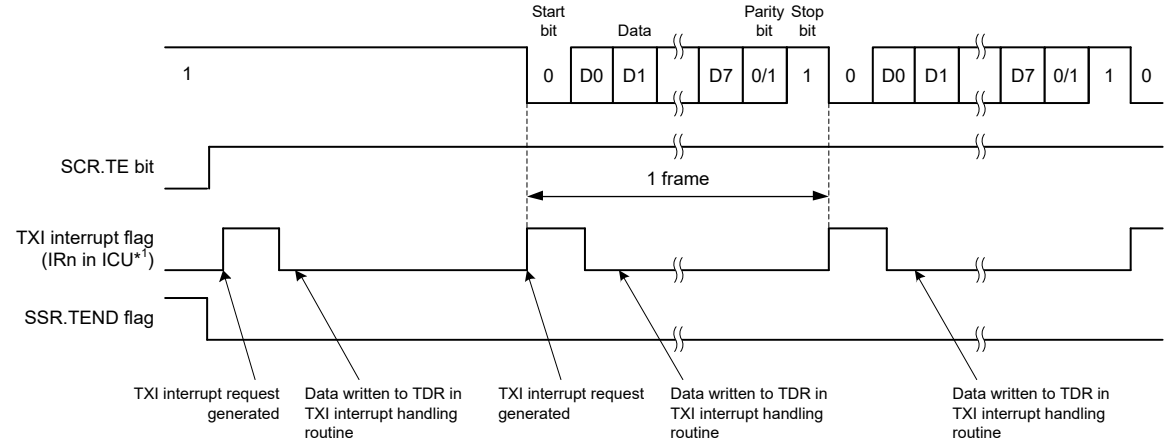
24.3.7 Serial Data Transmission (Asynchronous Mode)

Figure 24.6 to **Figure 24.8** show an example of the operation for serial transmission in asynchronous mode. In serial transmission, the SCI operates as described below.

1. The SCI transfers data from TDR*1 to TSR when data is written to TDR*1 in the TXI interrupt handling routine. The TXI interrupt request at the beginning of transmission is generated when the TE bit in SCR is set to 1 after the TIE bit in SCR is set to 1 or when these 2 bits are set to 1 simultaneously by a single instruction.
2. Transmission starts after the CTSE bit in SECR is set to 0 (CTS function is disabled) and a low level on the CTSn# pin causes data transfer from TDR*1 to TSR. If the TIE bit in SCR is 1 at this time, a TXI interrupt request is generated. Continuous transmission is obtainable by writing the next transmit data to TDR*1 in the TXI interrupt handling routine before transmission of the current transmit data is completed. When TEI interrupt requests are in use, set the SCR.TIE bit to 0 (a TXI interrupt request is disabled) and the SCR.TEIE bit to 1 (a TEI interrupt request is enabled) after the last of the data to be transmitted are written to the TDR*1*2 from the handling routine for TXI requests.
3. Data is sent from the TXDn pin in the following order: start bit, transmit data, parity bit or multi-processor bit (may be omitted depending on the format), and stop bit.
4. The SCI checks for updating of (writing to) TDR*3 at the time of stop bit output.
5. When TDR*3 is updated, setting of the CTSE bit in SECR to 0 (CTS function is disabled) or a low level input on the CTSn# pin cause the next transfer of the next transmit data from TDR*1 to TSR and sending of the stop bit, after which serial transmission of the next frame starts.
6. If TDR*3 is not updated, the TEND flag in SSR is set to 1, the stop bit is sent, and then the mark state is entered in which 1 is output. If the TEIE bit in SCR is 1 at this time, the TEND flag in SSR is set to 1 and a TEI interrupt request is generated.

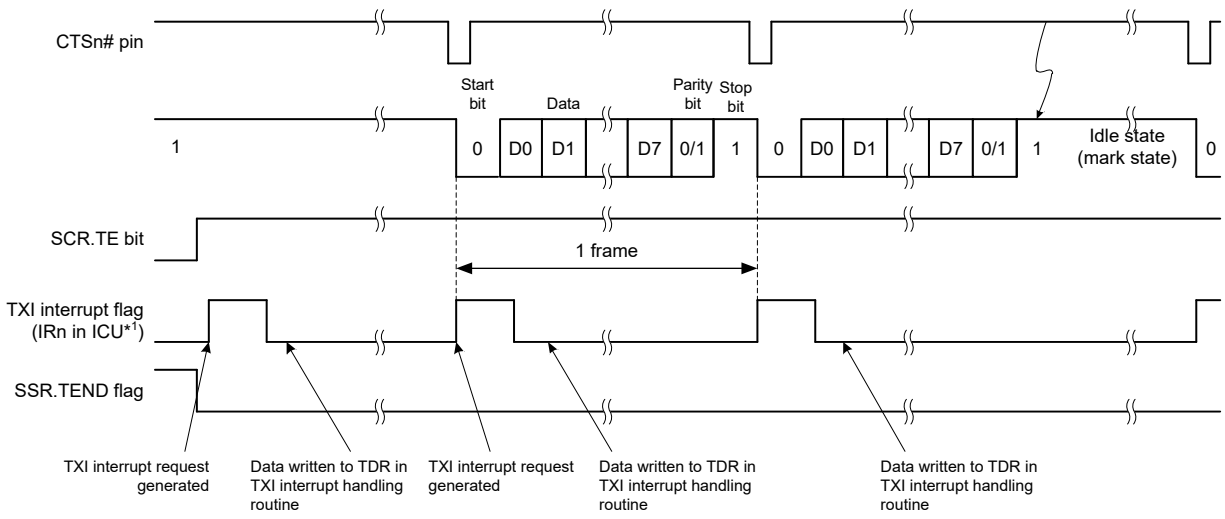
Note 1. Write data not to TDR but to the TDRH and TDRL registers when 9-bit data length is selected.
Note 2. Write data in the order from TDRH to TDRL when 9-bit data length is selected.
Note 3. The SCI checks for updating of the TDRL register only and does not check for updating of the TDRH register when 9-bit data length is selected.

Figure 24.9 shows a sample flowchart for serial transmission in asynchronous mode.



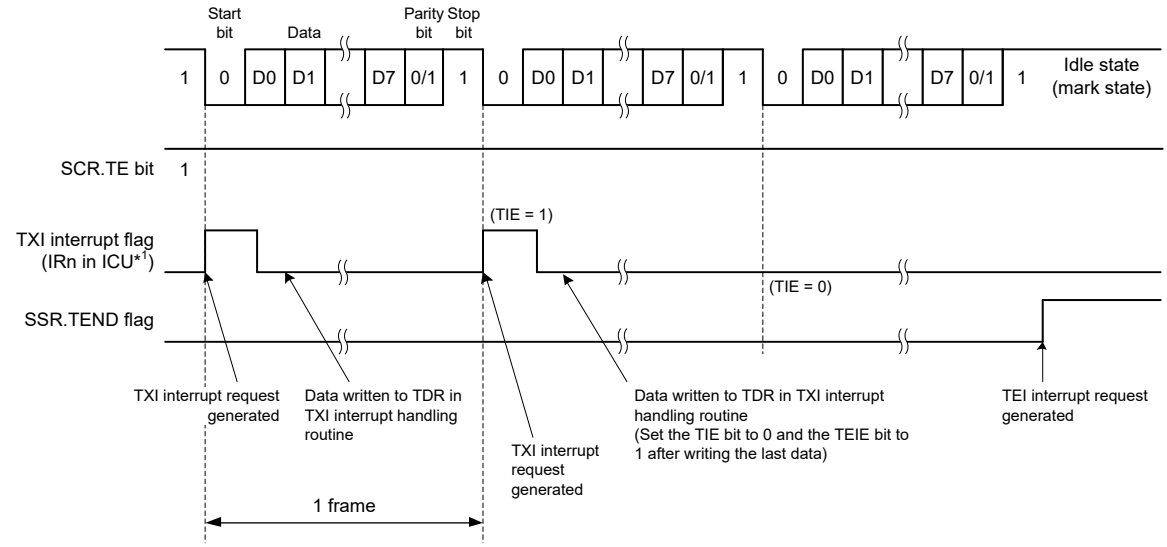
Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 24.6 Example of Operation for Serial Transmission in Asynchronous Mode (1)
(with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Not Used, at the Beginning of Transmission)



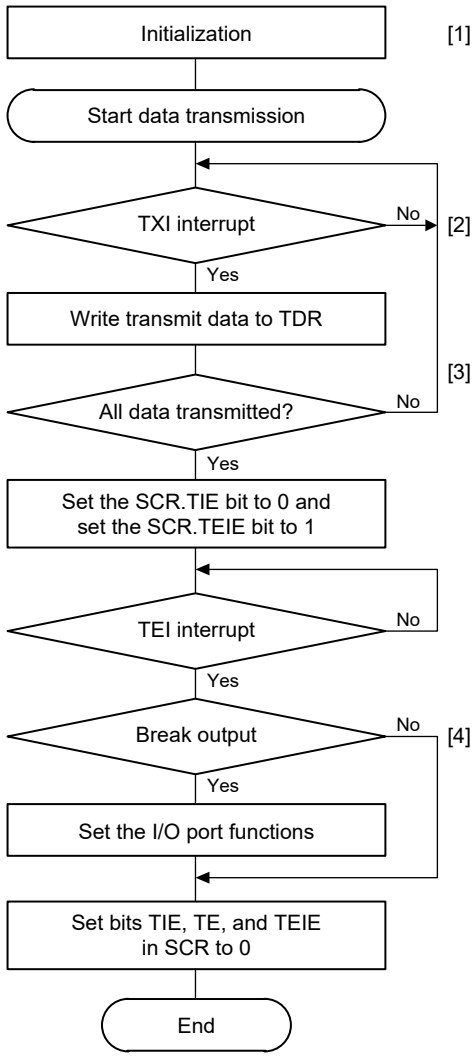
Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 24.7 Example of Operation for Serial Transmission in Asynchronous Mode (2)
(with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Used, at the Beginning of Transmission)



Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 24.8 Example of Operation for Serial Transmission in Asynchronous Mode (3)
(with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Not Used, from the Middle of Transmission until Transmission Completion)



[1] SCI initialization:
Set data transmission.
After the TE bit in SCR is set to 1, 1 is output for a frame, and transmission is enabled.

[2] Transmit data write to TDR by a TXI interrupt request:
When transmit data is transferred from TDR to TSR, a transmit data empty interrupt (TXI) request is generated.
Write transmit data to TDR once in the TXI interrupt handling routine.

[3] Serial transmission continuation procedure:
To continue serial transmission, write transmit data to TDR once using a TXI interrupt request.
Transmit data can also be written to TDR by activating the DMAC.
When TEI interrupt requests are in use, set the SCR.TIE bit to 0 and the SCR.TEIE bit to 1 after the last of the data to be transmitted are written to the TDR.

[4] Break output at the end of serial transmission:
To output a break in serial transmission, set the I/O port function corresponding to the TXDn pin (corresponding to output of the low level), and after switching the TXDn pin to the general I/O port function, set the TE bit in the SCR to 0.

Note: The TDR register becomes the TDRH and TDRL registers when 9-bit data length is selected. Write data in the order from TDRH to TDRL.

Figure 24.9 Example of Serial Transmission Flowchart in Asynchronous Mode

24.3.8 Serial Data Reception (Asynchronous Mode)

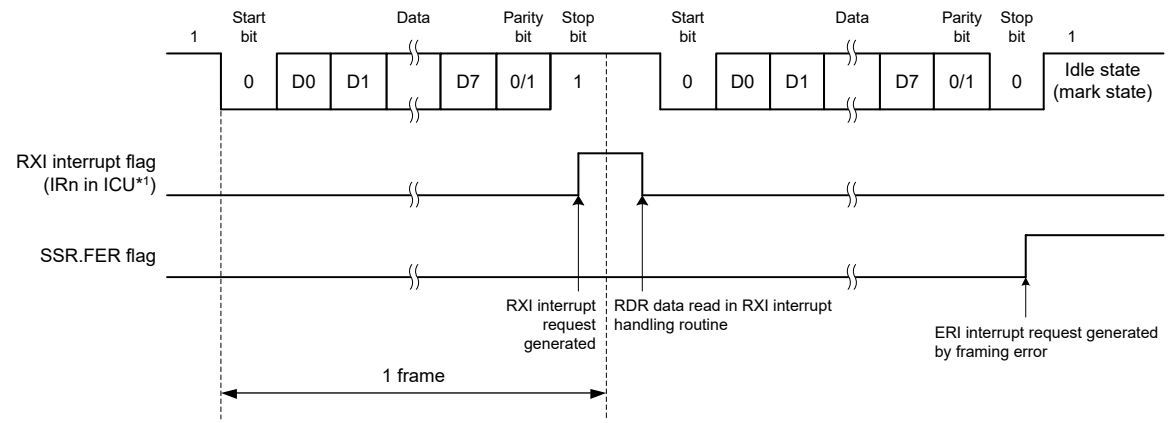
Figure 24.10 and **Figure 24.11** show an example of the operation for serial data reception in asynchronous mode. In serial data reception, the SCI operates as described below.

1. When the value of the RE bit in SCR becomes 1, the output signal on the RTSn# pin goes to the low level.
2. When the SCI monitors the communications line and detects a start bit, it performs internal synchronization, stores receive data in RSR, and checks the parity bit and stop bit.
3. If an overrun error occurs, the ORER flag in SSR is set to 1. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR*[1].
4. If a parity error is detected, the PER bit in SSR is set to 1 and receive data is transferred to RDR*[1]. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated.
5. If a framing error (when the stop bit is 0) is detected, the FER bit in SSR is set to 1 and receive data is transferred to RDR*[1]. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated.
6. When reception finishes successfully, receive data is transferred to RDR*[1]. If the RIE bit in SCR is 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to RDR*[1] in this RXI interrupt handling routine before reception of the next receive data is completed. Reading the received data that have been transferred to RDR*[1] causes the RTSn# pin to output the low level.

Note 1. Read data not in RDR but in the RDRH and RDRL registers when 9-bit data length is selected.

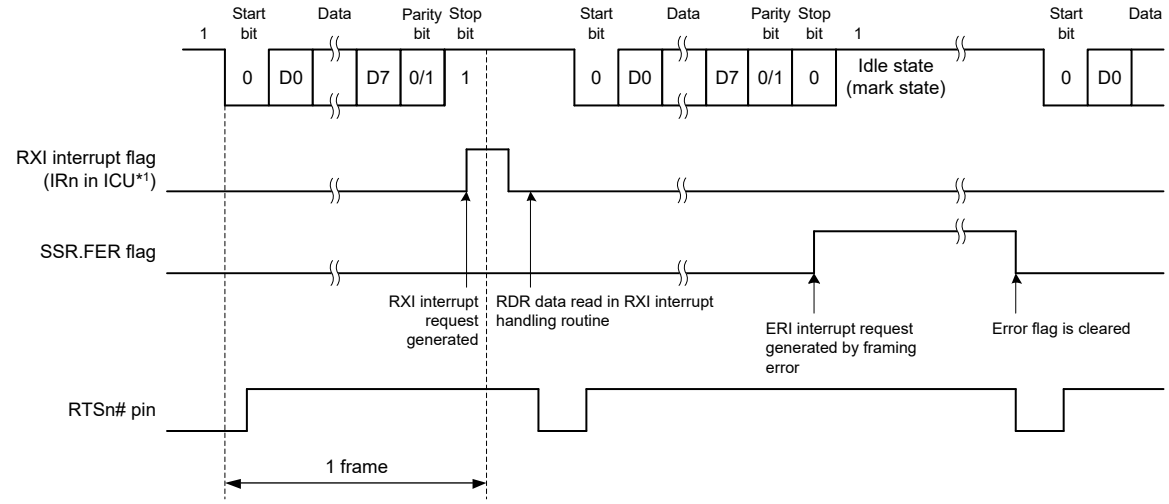
NOTE

The SCI checks for reading of the RDRL register only and does not check for reading of the RDRH register when 9-bit data length is selected.



Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 24.10 Example of SCI Operation for Serial Reception in Asynchronous Mode (1)
(When RTS Function is Not Used) (Example with 8-Bit Data, Parity, 1 Stop Bit)



Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 24.11 Example of SCI Operation for Serial Reception in Asynchronous Mode (2)
(When RTS Function is Used) (Example with 8-Bit Data, Parity, 1 Stop Bit)

Table 24.19 lists the states of the flags in the SSR status register and receive data handling when a receive error is detected.

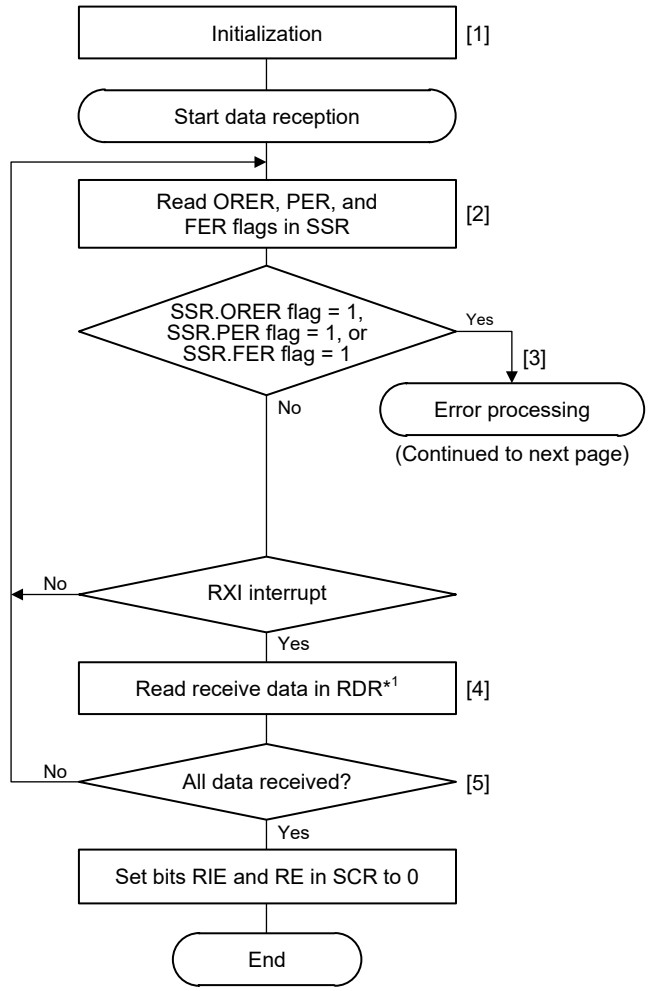
If a receive error is detected, an ERI interrupt request is generated but an RXI interrupt request is not generated. Data reception cannot be resumed while the receive error flag is 1. Accordingly, set the ORER, FER, and PER bits to 0 before resuming reception. Moreover, be sure to read the RDR (or the RDRL) during overrun error processing. When a reception is forcibly terminated by setting the SCR.RE bit to 0 during operation, read the RDR (or the RDRL) register because received data which has not yet been read may be left in RDR (or the RDRL).

Figure 24.12 and **Figure 24.13** show samples of flowcharts for serial data reception.

Table 24.19 Flags in the SSR Status Register and Receive Data Handling

Flags in the SSR Status Register				
ORER	FER	PER	Receive Data	Receive Error Type
1	0	0	Lost	Overrun error
0	1	0	Transferred to RDR*1	Framing error
0	0	1	Transferred to RDR*1	Parity error
1	1	0	Lost	Overrun error + framing error
1	0	1	Lost	Overrun error + parity error
0	1	1	Transferred to RDR*1	Framing error + parity error
1	1	1	Lost	Overrun error + framing error + parity error

Note 1. Read data not in RDR but in the RDRH and RDRL registers when 9-bit data length is selected.



[1] SCI initialization:
Set data reception.

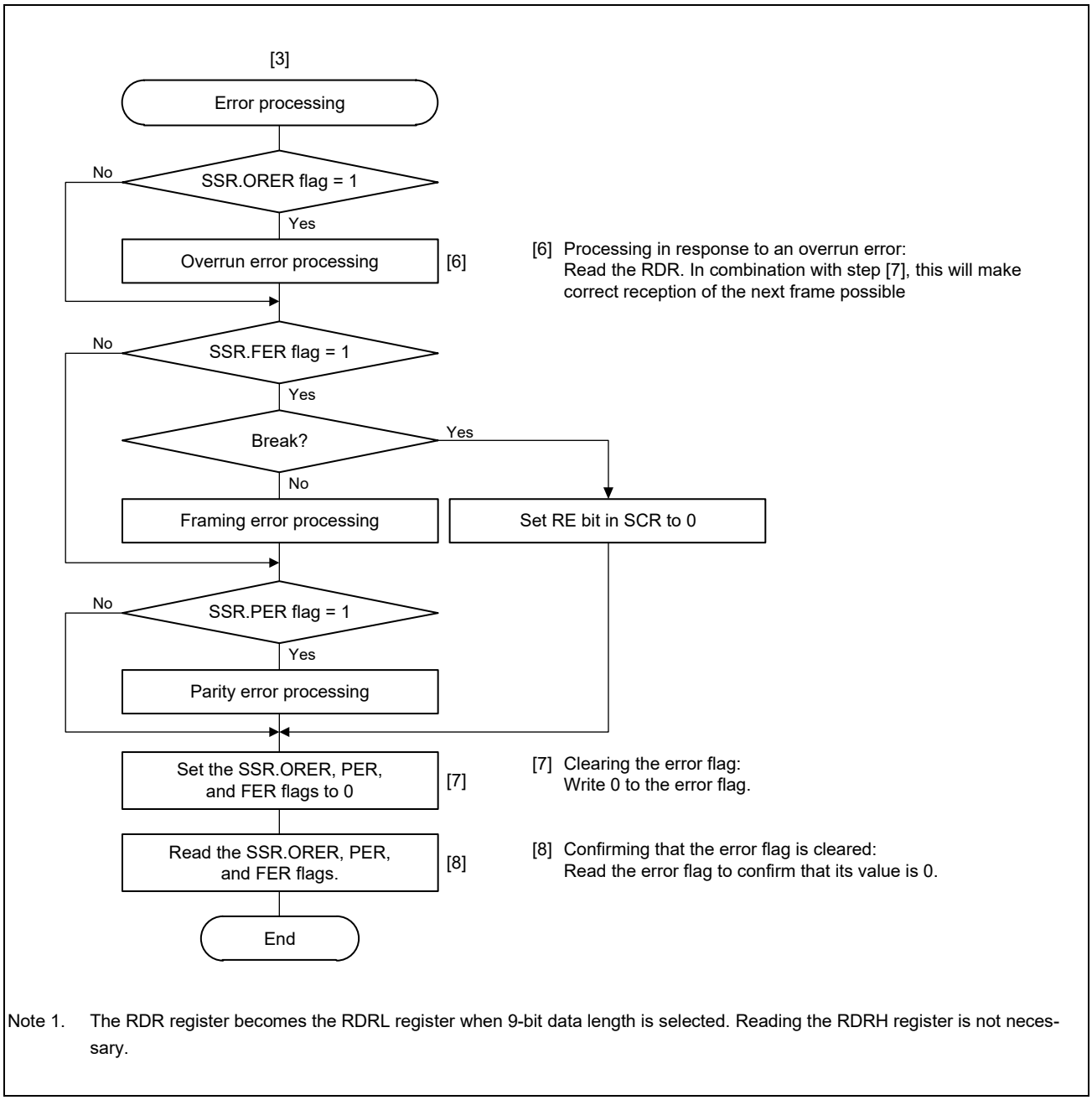
[2][3] Receive error processing and break detection:
If a receive error occurs, an ERI interrupt is generated. An error is identified by reading the ORER, PER, and FER flags in SSR. After performing the appropriate error processing, be sure to set the ORER, PER, and FER flags to 0. Reception cannot be resumed if any of these flags is set to 1. In the case of a framing error, a break can be detected by reading the value of the input port corresponding to the RXDn pin.

[4] Read the receive data in RDR once in the RXI interrupt handling routine.

[5] Serial reception continuation procedure:
To continue serial reception, before the stop bit of the current frame is received, read data from RDR in the RXI interrupt processing routine.
The RDR data can also be read by activating the DMAC.

Note 1. The RDR register becomes the RDRH and RDRL registers when 9-bit data length is selected. Read data in the order from RDRH to RDRL.

Figure 24.12 Example Flowchart of Serial Reception in Asynchronous Mode (1)



Note 1. The RDR register becomes the RDRL register when 9-bit data length is selected. Reading the RDRH register is not necessary.

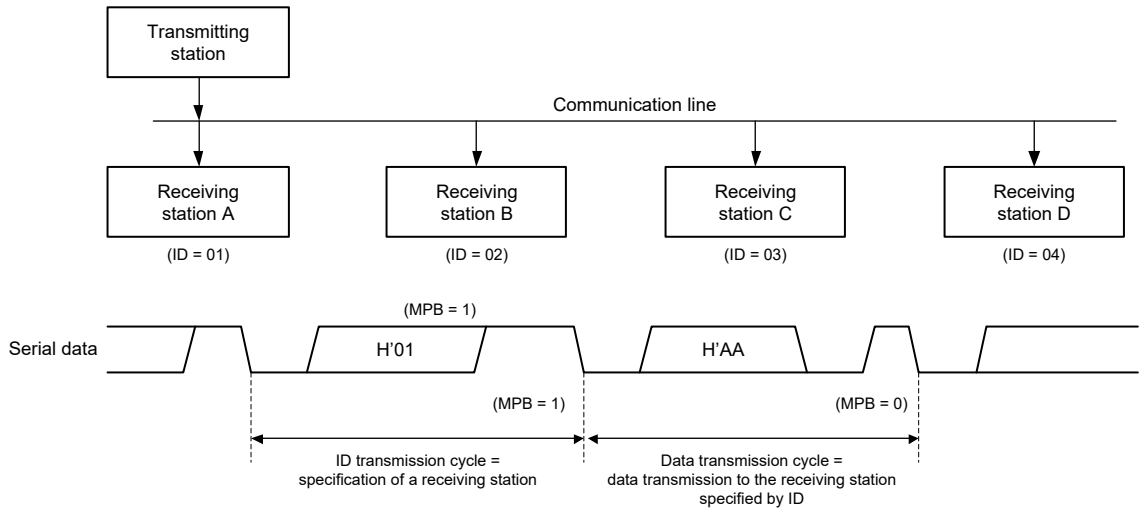
Figure 24.13 Example Flowchart of Serial Reception in Asynchronous Mode (2)

24.4 Multi-Processor Communications Function

Using the multi-processor communication functions enables to transmit and receive data by sharing a communication line between multiple processors by using asynchronous serial communication in which the multi-processor bit is added. In multi-processor communication, a unique ID code is allocated to each receiving station. Serial communication cycles consist of an ID transmission cycle to specify the receiving station and a data transmission cycle to transmit data to the specified receiving station. The multi-processor bit is used to distinguish between the ID transmission cycle and the data transmission cycle. When the multi-processor bit is set to 1, it indicates the ID transmission cycle and when the multi- processor bit is set to 0, it indicates the data transmission cycle. **Figure 24.14** shows an example of communication between processors by using a multi-processor format. First, a transmitting station transmits communication data in which the multi-processor bit set to 1 is added to the ID code of the receiving station. Next, the transmitting station transmits the communication data in which the multi-processor bit set to 0 is added to the transmit data. Upon receiving the communication data in which the multi-processor bit is set to 1, the receiving station compares the received ID with the ID of the receiving station itself and if the two match, receives the communication data that is subsequently transmitted. If the received ID does not match with the ID of the receiving station, the receiving station skips the communication data until again receiving the communication data in which the multi-processor bit is set to 1.

For supporting this function, the SCI provides the MPIE bit in SCR. When the MPIE bit is set to 1, transfer of receive data from the RSR to the RDR (the RDRH and RDRL registers when 9-bit data length is selected), detection of a receive error, and setting the respective status flags ORER and FER in SSR are disabled until reception of data in which the multi-processor bit is set to 1. Upon receiving a reception character in which the multi-processor bit is set to 1, the MPBT bit in SSR is set to 1 and the MPIE bit in SCR is automatically cleared, thus returning to a normal reception operation. During this time, an RXI interrupt is generated if the RIE bit in SCR is set.

When the multi-processor format is specified, specification of the parity bit is disabled. Apart from this, there is no difference from the operation in the normal asynchronous mode. A clock which is used for the multi-processor communication is also the same as the clock used in the normal asynchronous mode.

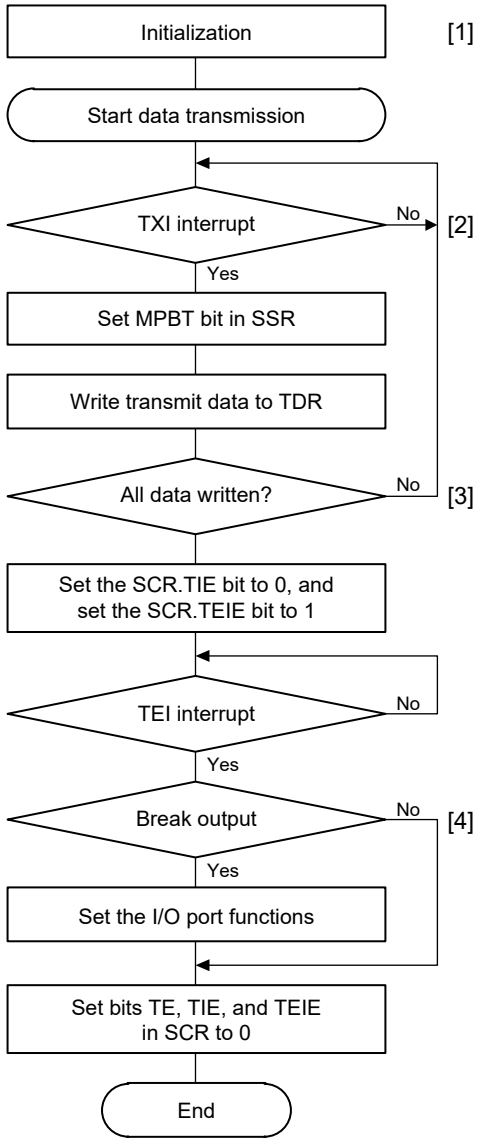


Note: MPB: Multi-processor bit

Figure 24.14 An Example of Communication using the Multi-Processor Format (Example of Transmission of Data H'AA to Receiving Station A)

24.4.1 Multi-Processor Serial Data Transmission

Figure 24.15 is a sample flowchart of multi-processor data transmission. In the ID transmission cycle, the ID should be transmitted with the MPBT bit in SSR set to 1. In the data transmission cycle, the data should be transmitted with the MPBT bit set to 0. The other operations are the same as the operations in asynchronous mode.



[1] SCI initialization:
Set data transmission.
After the TE bit in SCR is set to 1, 1 is output for a frame, and transmission is enabled.

[2] TXI interrupt request:
When transmit data is transferred from TDR to TSR, a transmit data empty interrupt (TXI) request is generated.
Set the MPBT bit in SSR to 0 or 1, and write transmit data to TDR once in the TXI interrupt handling routine.

[3] Serial transmission continuation procedure:
To continue serial transmission, write transmit data to TDR once using a TXI interrupt request.
Transmit data can also be written to TDR by activating the DMAC.
When TEI interrupt requests are in use, set the SCR.TIE bit to 0 and the SCR.TEIE bit to 1 after the last of the data to be transmitted are written to the TDR.

[4] Break output at the end of serial transmission:
To output a break in serial transmission, set the I/O port function corresponding to the TXDn pin (corresponding to output of the low level), and after switching the TXDn pin to the general I/O port function, set the TE bit in the SCR to 0.

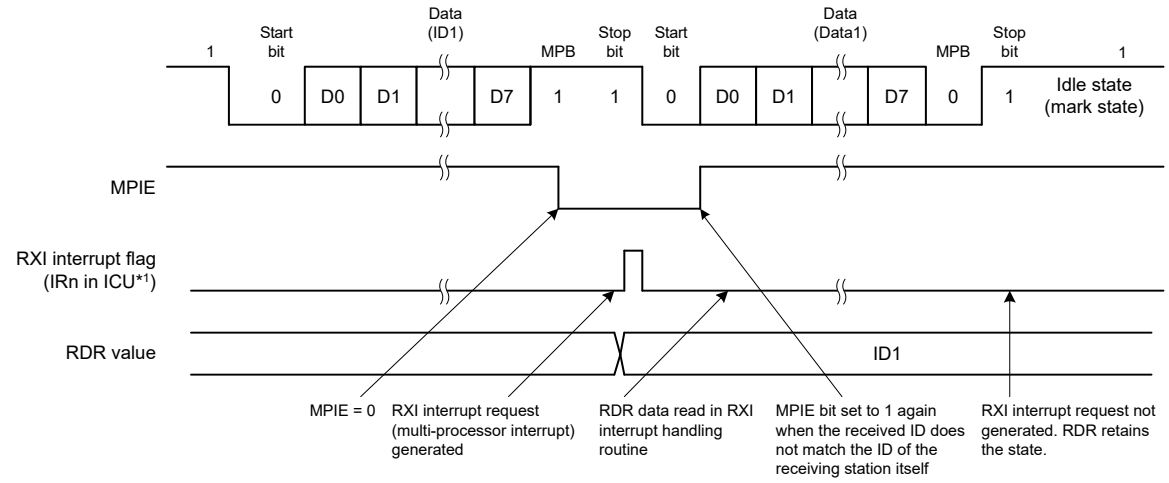
Note: The TDR register becomes the TDRH and TDRL registers when 9-bit data length is selected. Write data in the order from TDRH to TDRL.

Figure 24.15 Example of Multi-Processor Serial Transmission Flowchart

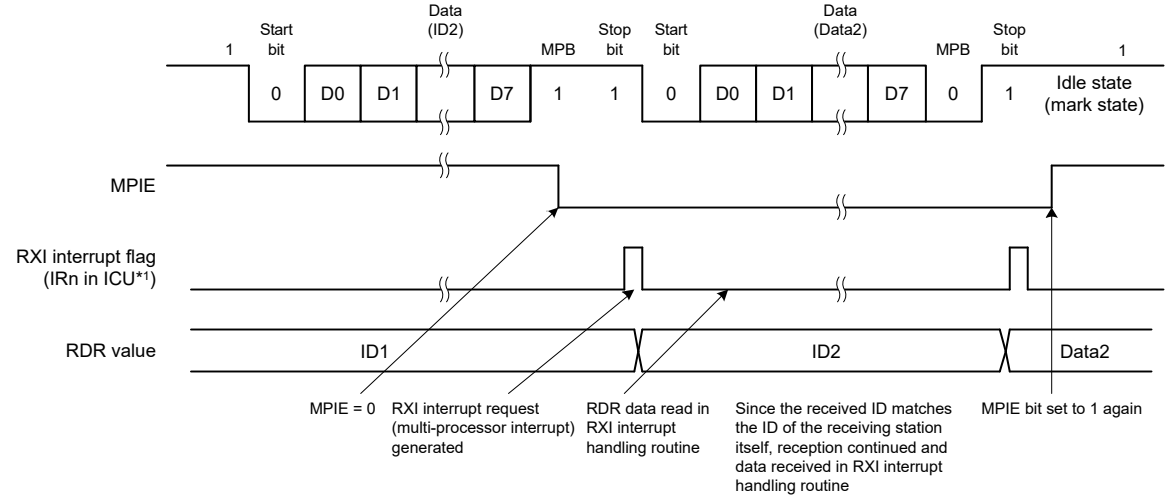
24.4.2 Multi-Processor Serial Data Reception

Figure 24.17 and **Figure 24.18** are sample flowcharts of multi-processor data reception. When the MPIE bit in SCR is set to 1, reading the communication data is skipped until reception of the communication data in which the multi-processor bit is set to 1. When the communication data in which the multi-processor bit is set to 1 is received, the received data is transferred to RDR (the RDRH and RDRL registers when 9-bit data length is selected). During this time, the RXI interrupt request is generated. The other operations are the same as the operations in asynchronous mode.

Figure 24.16 is the example of operation for reception.



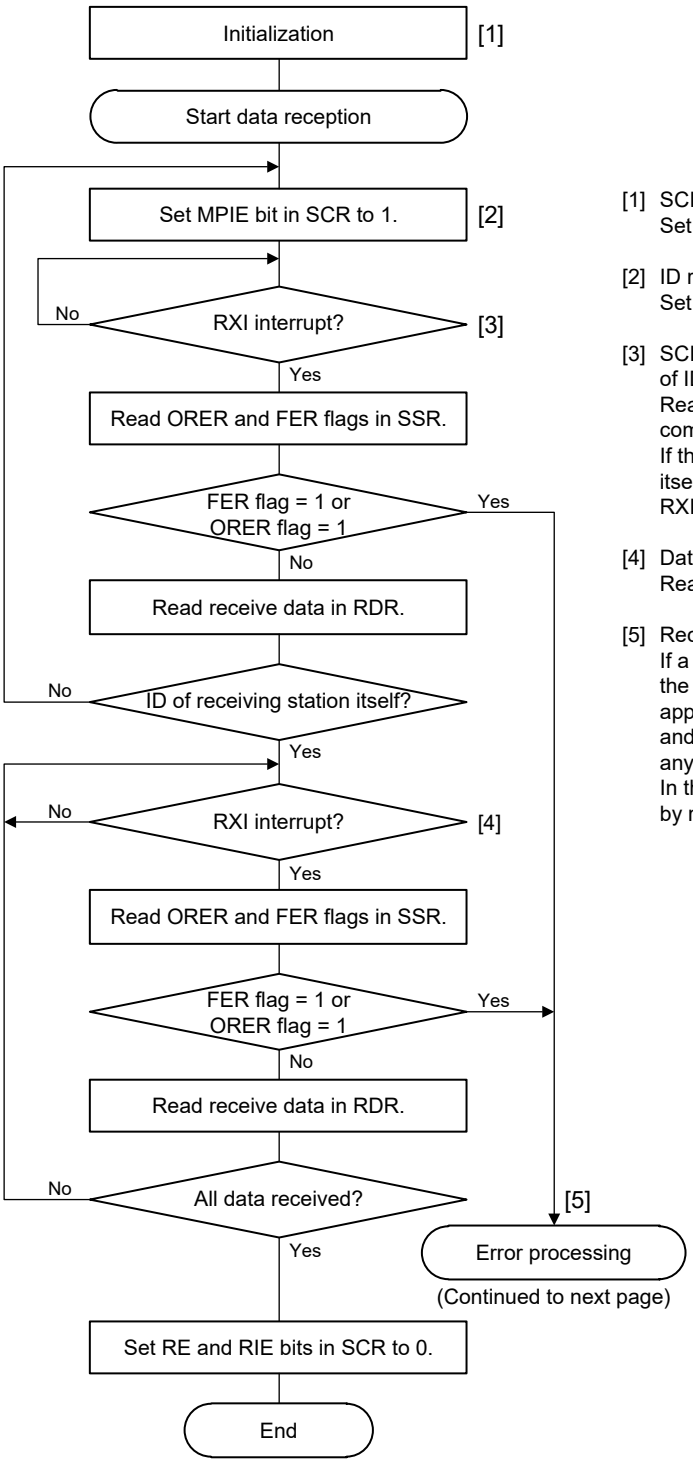
(a) When the received ID does not match the ID of the receiving station itself



(b) When the received ID matches the ID of the receiving station itself

Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 24.16 Example of SCI Reception (8-Bit Data/Multi-Processor Bit/1 Stop Bit)



[1] SCI initialization:
Set data reception.

[2] ID reception cycle:
Set the MPIE bit in SCR to 1 and wait for ID reception.

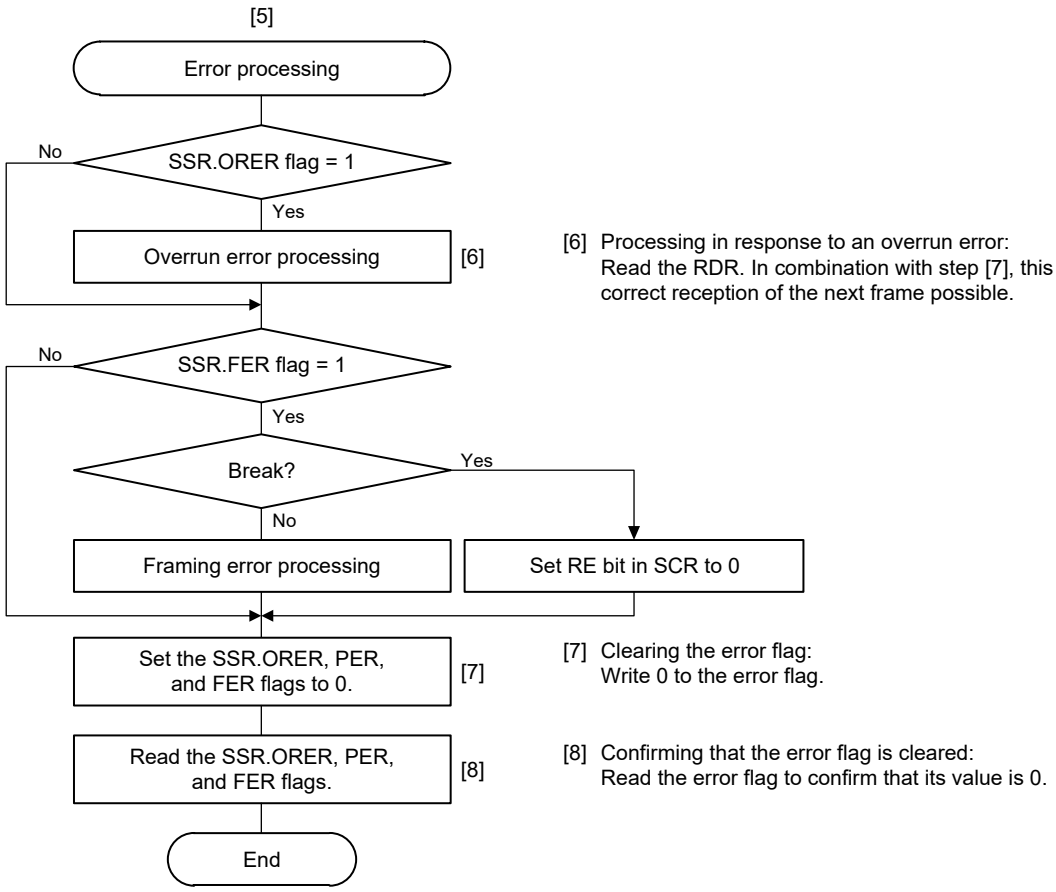
[3] SCI status confirmation and reception and comparison of ID:
Read data in RDR at the first RXI interrupt, and compare it with the ID of the receiving station itself.
If the ID does not match the ID of the receiving station itself, set the MPIE bit to 1 again, and wait for another RXI interrupt RXI interrupt request.

[4] Data reception at an RXI interrupt:
Read data in RDR once in the RXI interrupt routine.

[5] Receive error processing and break detection:
If a receive error occurs, an error is identified by reading the ORER and FER flags in SSR. After performing the appropriate error processing, be sure to set the ORER and FER flags to 0. Reception cannot be resumed if any of these flags is set to 1.
In the case of a framing error, a break can be detected by reading the value of the RXDn pin.

Note: The RDR register becomes the RDRH and RDRL registers when 9-bit data length is selected. Read data in the order from RDRH to RDRL.

Figure 24.17 Example of Multi-Processor Serial Reception Flowchart (1)



Note: The RDR register becomes the RDRH and RDRL registers when 9-bit data length is selected. Read data in the order from RDRH to RDRL.

Figure 24.18 Example of Multi-Processor Serial Reception Flowchart (2)

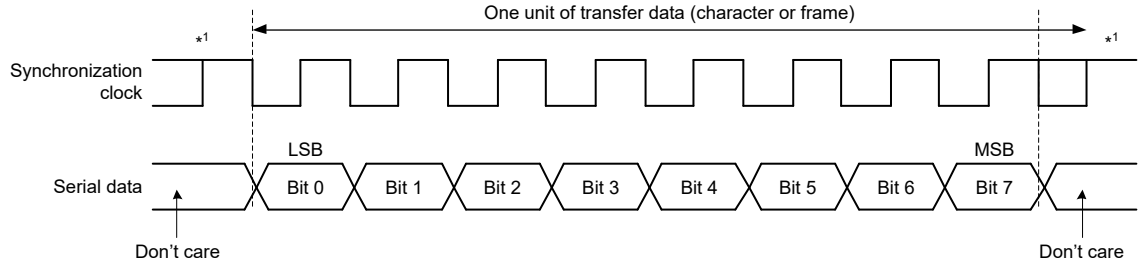
24.5 Operation in Clock Synchronous Mode

Figure 24.19 shows the data format for clock synchronous serial data communications.

In clock synchronous mode, data is transmitted or received in synchronization with clock pulses. One character in transfer data consists of 8-bit data. In clock synchronous mode, no parity bit can be added.

In data transmission, the SCI outputs data from one falling edge of the synchronization clock to the next. In data reception, the SCI receives data in synchronization with the rising edge of the synchronization clock. After 8-bit data is output, the communication line holds the last bit output state.

Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communications by use of a common clock. Both the transmitter and the receiver also have a double-buffered structure, so that the next transmit data can be written during transmission or the previous receive data can be read during reception, enabling continuous data transfer.



Note 1. Holds a high level except during continuous transfer.

Figure 24.19 Data Format in Clock Synchronous Serial Communications (LSB First)

24.5.1 Clock

Either an internal clock generated by the on-chip baud rate generator or an external synchronization clock input at the SCKn pin can be selected, according to the setting of the SCR.CKE[1:0] bits.

When the SCI is operated on an internal clock, the synchronization clock is output from the SCKn pin. Eight synchronization clock pulses are output in the transfer of one character, and when no transfer is performed the clock is held high. However, when only data reception is performed while the CTS function is disabled, the synchronization clock output is started at the same time when the SCR.RE bit set to 1. The synchronization clock is stopped at the high level when an overrun error occurs or the SCR.RE bit is set to 0.

When only data reception is performed and the CTS function is enabled, the clock output is not started even when the SCR.RE bit set to 1 if the CTSn# pin input is high when the SCR.RE bit is 0. The synchronization clock output is started when the SCR.RE bit is set to 1 and the CTSn# pin input is low. After that, if the CTSn# pin input is high on completion of the frame reception, the synchronization clock output is stopped at the high level. If the CTSn# pin input continues to be low, the synchronization clock is stopped at the high level when an overrun error occurs or the SCR.RE bit is set to 0.

24.5.2 CTS and RTS Functions

In the CTS function, CTSn# pin input is used to control reception/transmission start when the clock source is the internal clock. Setting the SECR.CTSE bit to 1 enables the CTS function.

When the CTS function is enabled, placing the low level on the CTSn# pin causes reception/transmission to start.

In the RTS function, RTSn# pin output is used to request reception/transmission start when the clock source is an external synchronizing clock. A low level is output when serial communications become possible. Conditions for output of the low and high level are shown below.

[Conditions for low-level output] Satisfaction of all conditions listed below

- The value of the RE or TE bit in the SCR is 1
- Neither transmission nor reception is in progress
- There are no received data yet to be read (when the SCR.RE bit is 1)
- Transmit data has been written (when the SCR.TE bit is 1)
- ORER flag in SSR is 0

[Condition for high-level output]

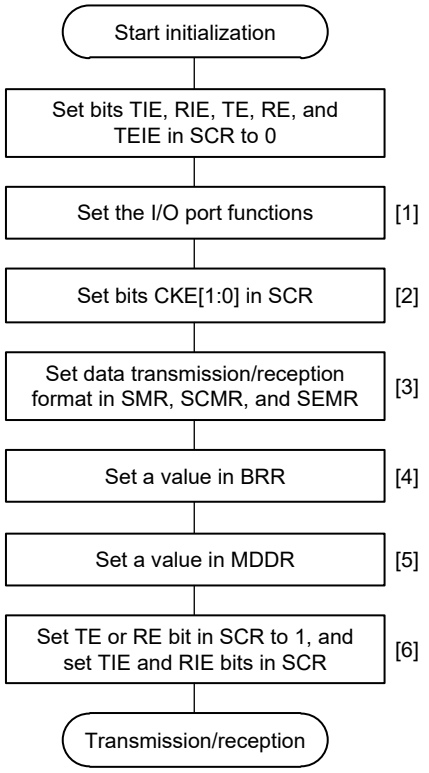
- The conditions for low-level output have not been satisfied.

24.5.3 SCI Initialization (Clock Synchronous Mode)

Before transmitting and receiving data, start by writing the initial value H'00 to the SCR and then continue through the procedure for SCI given in **Figure 24.20**. Whenever the operating mode or transfer format is changed, the SCR must be initialized before the change is made.

Note that setting the SCR.RE bit to 0 initializes neither the ORER, FER, and PER flags in SSR nor RDR.

Moreover, note that switching the value of the SCR.TE bit from 1 to 0 or 0 to 1 while the SCR.TIE bit is 1 leads to the generation of a TXI interrupt request.



[1] Make I/O port settings to enable input and output functions as required for TXDn, RXDn, and SCKn pins.

[2] Set the clock selection in SCR.
When an internal clock is selected, the SCK pin functions as the clock output pin.

[3] Set the data transmission/reception format in SMR, SCMR, and SEMR.

[4] Write a value corresponding to the bit rate to BRR. This step is not necessary if an external clock is used.

[5] Write the value obtained by correcting a bit rate error in MDDR. This step is not necessary if the BRME bit in SEMR is cleared to 0 or an external clock is used.

[6] Set the TE bit or RE bit in SCR to 1. Also set the TIE and RIE bits in SCR.
Setting the TE and RE bits allows TXDn and RXDn to be used.

Note: In simultaneous transmit and receive operations, the TE and RE bits in SCR should both be set to 0 or 1 simultaneously.

Figure 24.20 Example of SCI Initialization Flowchart (Clock Synchronous Mode)

24.5.4 Serial Data Transmission (Clock Synchronous Mode)

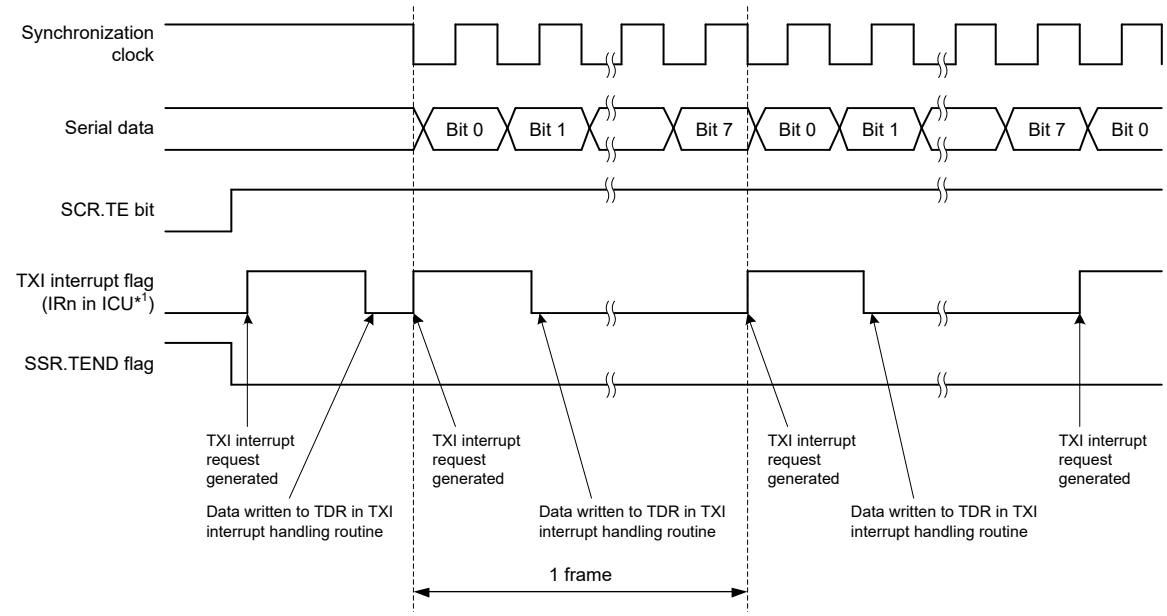
Figure 24.21, **Figure 24.22**, and **Figure 24.23** show an example of the operation for serial transmission in clock synchronous mode.

In serial data transmission, the SCI operates as described below.

1. The SCI transfers data from TDR to TSR when data is written to TDR in the TXI interrupt handling routine. The TXI interrupt request at the beginning of transmission is generated when the TE bit in SCR is set to 1 after the TIE bit in SCR is set to 1 or when these 2 bits are set to 1 simultaneously by a single instruction.
2. After transferring data from TDR to TSR, the SCI starts transmission. When the SCR.TIE bit is set to 1 at this time, a TXI interrupt request is generated. Continuous transmission is enabled by writing the next transmit data to TDR in this TXI interrupt handling routine before transmission of the current transmit data has finished. When TEI interrupt requests are in use, set the SCR.TIE bit to 0 (a TXI interrupt request is disabled) and the SCR.TEIE bit to 1 (a TEI interrupt request is enabled) after the last of the data to be transmitted are written to the TDR from the handling routine for TXI requests.
3. 8-bit data is sent from the TXDn pin in synchronization with the output clock when clock output mode has been specified and in synchronization with the input clock when use of an external clock has been specified. Output of the clock signal is suspended until the input CTS signal is at the low level while the CTSE bit in SECR is 1 (CTS function is enabled).
4. The SCI checks for updating of (writing to) the TDR at the time of the last bit output.
5. When TDR is updated, the next transmit data is transferred from TDR to TSR, and serial transmission of the next frame is started.
6. If TDR is not updated, set the SSR.TEND flag to 1 and the TXDn pin retains the output state of the last bit. If the TEIE bit in SCR is 1 at this time, a TEI interrupt request is generated. The SCKn pin is held high.

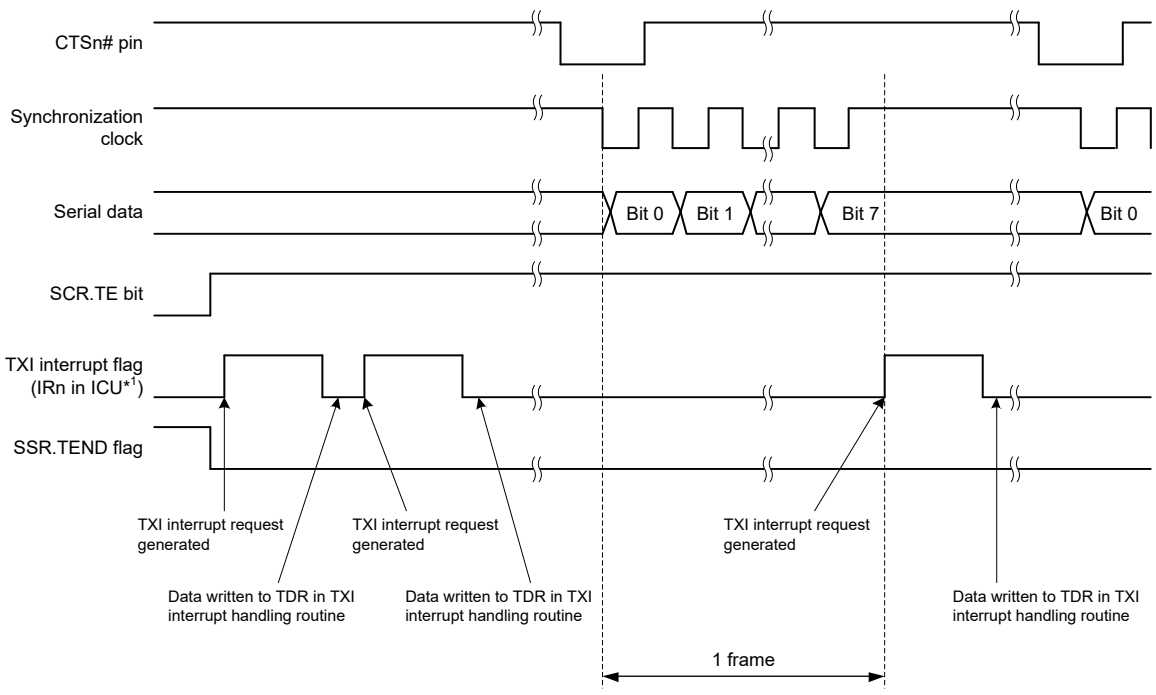
Figure 24.24 shows a sample flowchart of serial data transmission.

Transmission will not start while a receive error flag (ORER, FER, or PER in SSR) is set to 1. Be sure to set the receive error flags to 0 before starting transmission. Note that setting the RE bit in SCR to 0 does not clear the receive error flags.



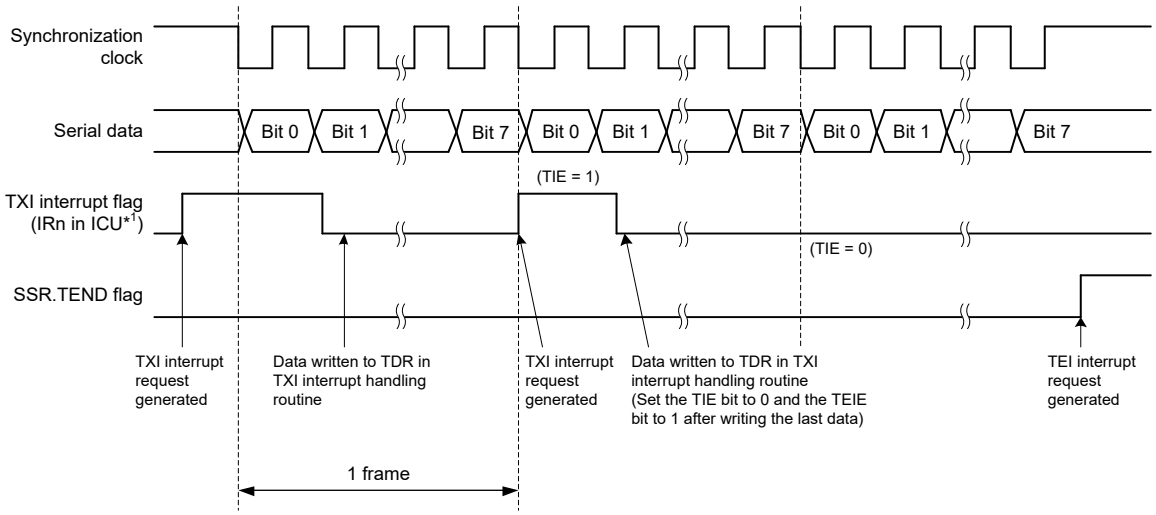
Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 24.21 Example of Serial Data Transmission in Clock Synchronous Mode (1)
(When the CTS Function is Not Used at the Beginning of Transmission)



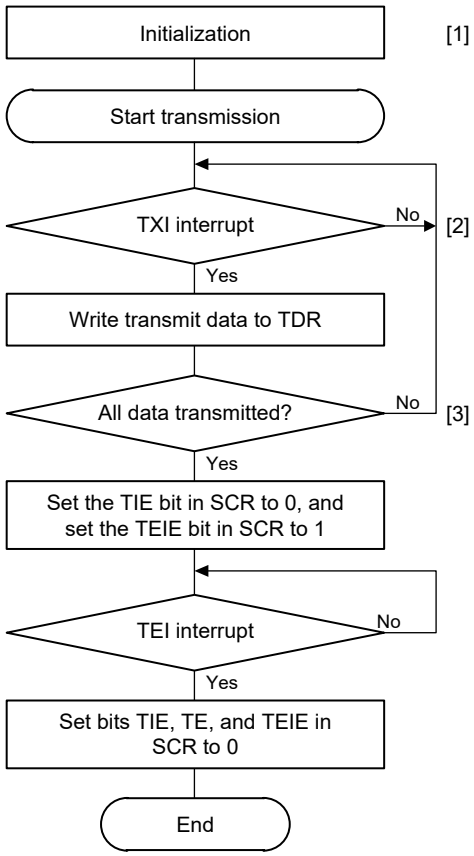
Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 24.22 Example of Serial Data Transmission in Clock Synchronous Mode (2)
(When the CTS Function is Used at the Beginning of Transmission)



Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 24.23 Example of Serial Data Transmission in Clock Synchronous Mode (3)
(From the Middle of Transmission until Transmission Completion)



[1] SCI initialization:
Set data transmission.

[2] Writing transmit data write to TDR by a TXI interrupt request:
When transmit data is transferred from TDR to TSR, a transmit data empty interrupt (TXI) request is generated.
Transmit data is written to TDR once from the handling routine for TXI requests.

[3] Serial transmission continuation procedure:
To continue serial transmission, write transmit data to TDR upon accepting a transmit data empty interrupt (TXI) request. Transmit data can also be written to TDR by activating the DMAC by the TXI interrupt request. When TEI interrupt requests are in use, set the SCR.TIE bit to 0 and the SCR.TEIE bit to 1 after the last of the data to be transmitted are written to the TDR.

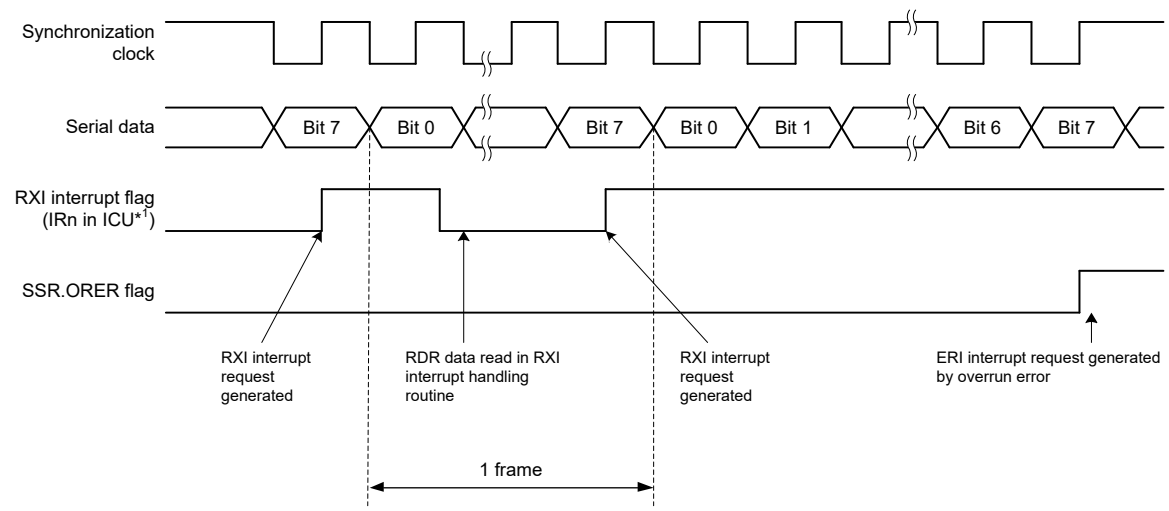
Note: When the external clock is in use (the value of the SCR.CKE[1:0] bits is 10b or 11b), the rising edge on the SCK pin for the last bit sets the SSR.TEND flag to 1. Setting the SCR.TE bit to 0 immediately after this may lead to insufficient received-data hold time on the receiver side.

Figure 24.24 Example Flowchart of Serial Transmission in Clock Synchronous Mode

24.5.5 Serial Data Reception (Clock Synchronous Mode)

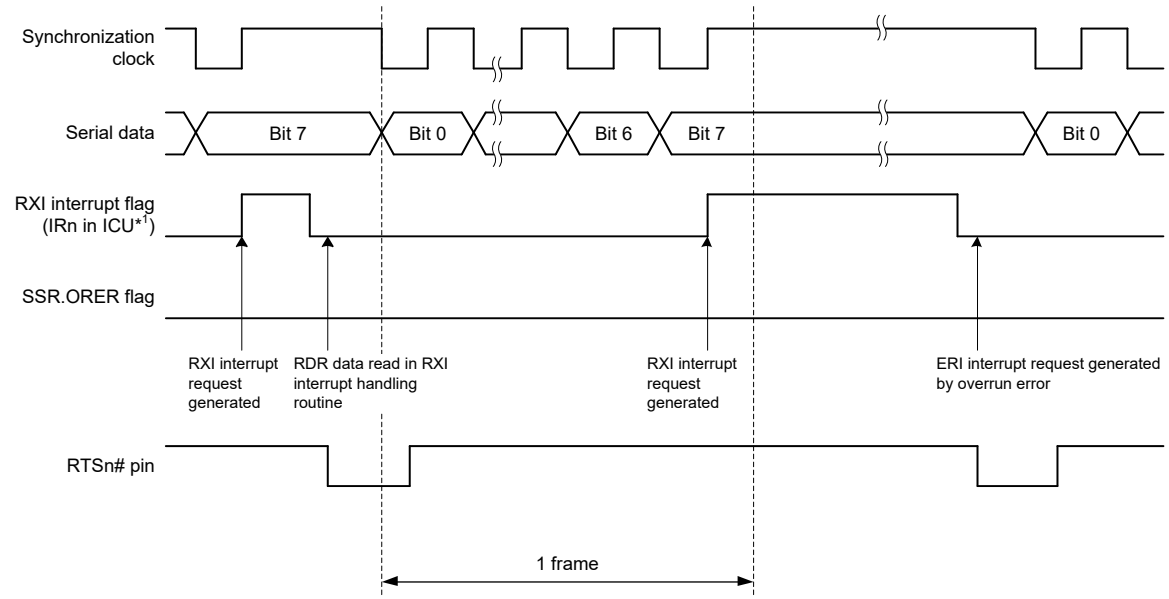
Figure 24.25 and **Figure 24.26** show an example of SCI operation for serial reception in clock synchronous mode. In serial data reception, the SCI operates as described below.

1. The value of the RE bit in SCR becoming 1 places the signal output on the RTSn# pin at the low level (when the RTS function is in use).
2. The SCI performs internal initialization and starts receiving data in synchronization with a synchronization clock input or output, and stores the receive data in RSR.
3. If an overrun error occurs, the ORER bit in SSR is set to 1. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR.
4. When reception finishes successfully, receive data is transferred to RDR. If the RIE bit in SCR is 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to RDR in this RXI interrupt handling routine before reception of the next receive data is completed. Reading out the received data that have been transferred to RDR causes the RTSn# pin to output the low level (when the RTS function is in use).



Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 24.25 Example of Operation for Serial Reception in Clock Synchronous Mode (1)
(When RTS Function is Not Used)



Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 24.26 Example of Operation for Serial Reception in Clock Synchronous Mode (2) (When RTS Function is Used)

Data transfer cannot be resumed while a receive error flag is 1. Accordingly, clear the ORER, FER, and PER bits in SSR to 0 before resuming reception. Moreover, be sure to read the RDR during overrun error processing. When a reception is forcibly terminated by setting the SCR.RE bit to 0 during operation, read RDR because received data which has not yet been read may be left in RDR.

Figure 24.27 shows a sample flowchart for serial data reception.

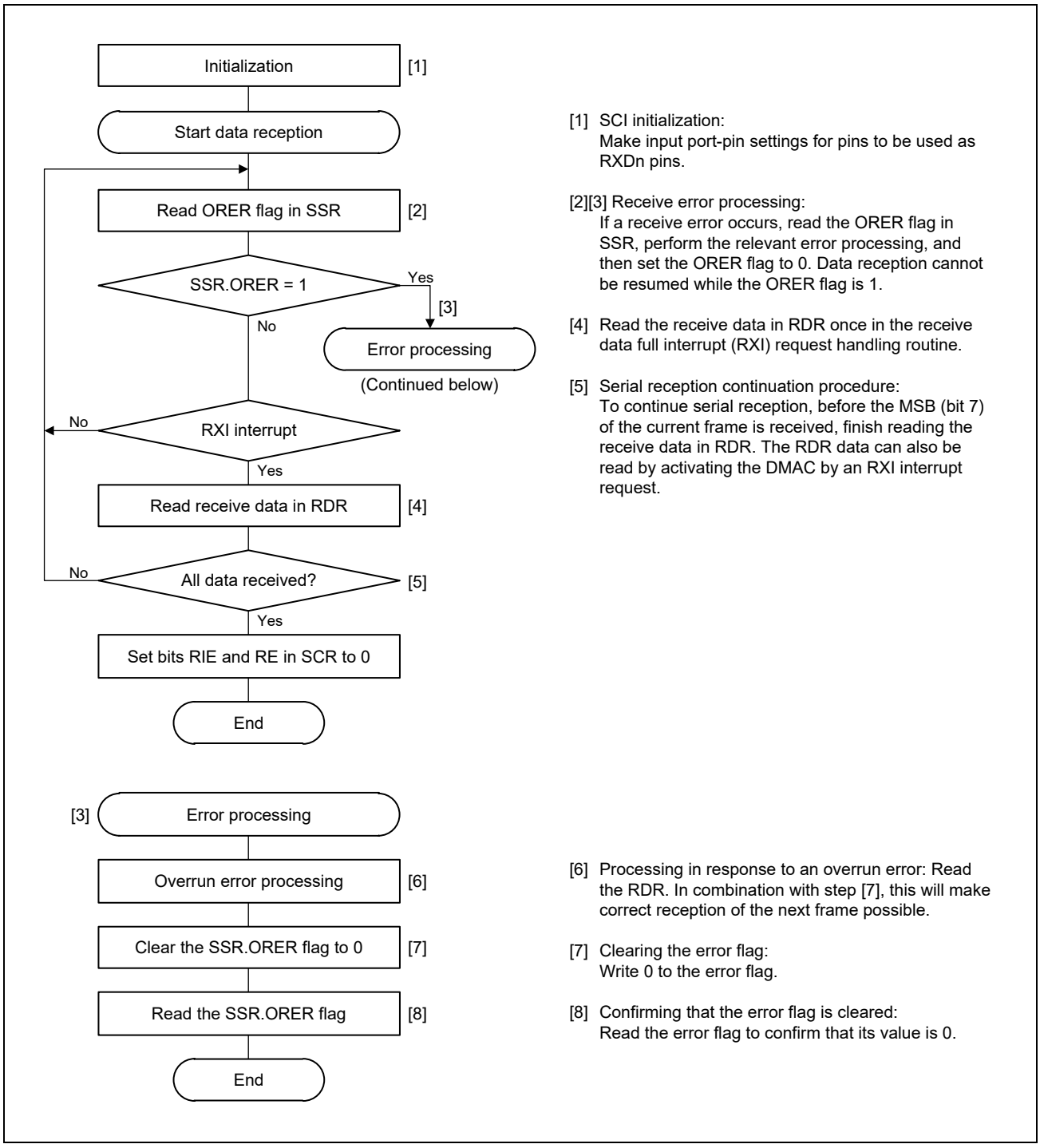


Figure 24.27 Example Flowchart of Serial Reception in Clock Synchronous Mode

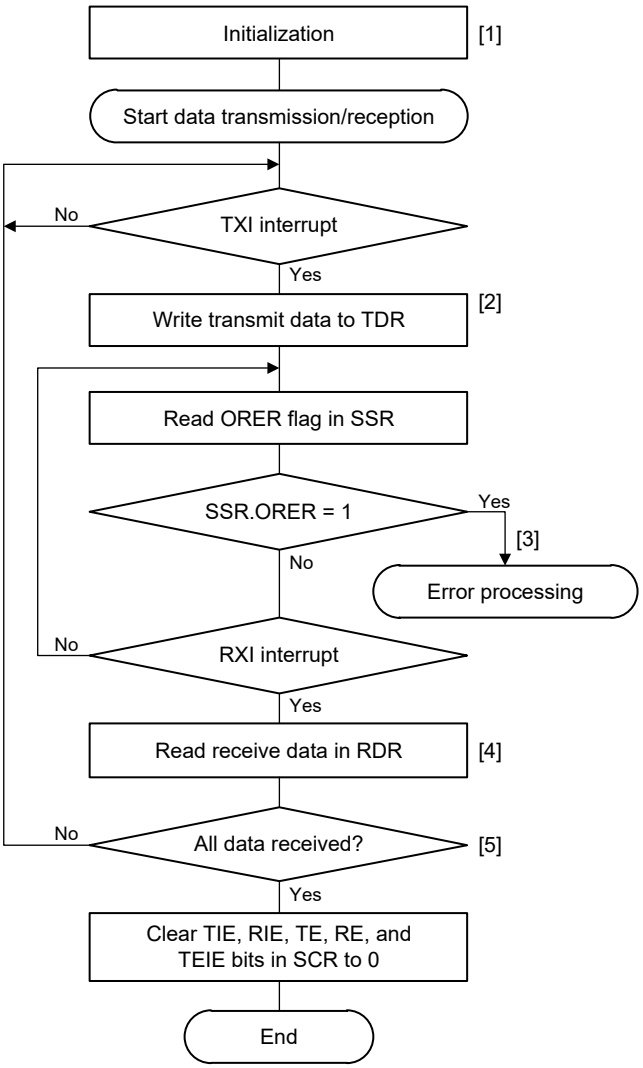
24.5.6 Simultaneous Serial Data Transmission and Reception (Clock Synchronous Mode)

Figure 24.28 shows a sample flowchart for simultaneous serial transmit and receive operations in clock synchronous mode.

After initializing the SCI, the following procedure should be used for simultaneous serial data transmit and receive operations.

To switch from transmit mode to simultaneous transmit and receive mode, check that the SCI has finished transmission by reading that the TEND flag in SSR is 1, and then initialize the SCR register. Then set the TIE, RIE, TE, and RE bits in SCR to 1 simultaneously by a single instruction.

To switch from receive mode to simultaneous transmit and receive mode, check that the SCI has finished reception, and then set the RIE and RE bits to 0. Then check that the receive error flags (ORER, FER, and PER in SSR) are 0, and then set the TIE, RIE, TE, and RE bits in SCR to 1 simultaneously by a single instruction.



[1] SCI initialization:
The TXDn pin can act as the output pin for transmitted data and the RXDn pin can act as the input pin for received data at the same time.

[2] Transmit data write:
Write transmit data to TDR once in the TXI interrupt handling routine.

[3] Receive error processing:
If a receive error occurs, read the ORER flag in SSR, perform the relevant error processing, and then set the ORER flag to 0. Data reception cannot be resumed while the ORER flag is 1.

[4] Reading receive data:
Read the receive data in RDR once in the RXI interrupt request handling routine.

[5] Serial transmission/reception continuation procedure:
To continue serial transmission and reception, before the MSB (bit 7) of the current frame is received, finish reading the receive data in RDR by the RXI interrupt. Also, before the MSB (bit 7) of the current frame is transmitted, write data to TDR by the TXI interrupt.
Transmit data can also be written to TDR by activating the DMAC by a transmit data empty interrupt (TXI) request. Similarly, the RDR data can also be read by activating the DMAC by a receive data full interrupt (RXI) request.

Note: When switching from transmit or receive operation to simultaneous transmit and receive operations first set the TIE, RIE, TE, RE, and TEIE bits in SCR to 0, and then set TIE, RIE, TE, and RE bits to 1 simultaneously.

Figure 24.28 Example Flowchart of Simultaneous Serial Transmission and Reception in Clock Synchronous Mode

24.6 Operation in Smart Card Interface Mode

The SCI supports smart card (IC card) interfaces conforming to ISO/IEC 7816-3 (standard for Identification Cards), as an extended function of the SCI.

Smart card interface mode can be selected using the appropriate register.

24.6.1 Sample Connection

Figure 24.29 shows a sample connection between a smart card (IC card) and this MCU.

As in the figure, since this MCU communicates with an IC card using a single transmission line, interconnect the TXDn and RXDn pins and pull up the data transmission line to VCC using a resistor.

Setting the TE and RE bits in SCR to 1 with an IC card disconnected enables closed transmission/reception allowing self-diagnosis.

To supply an IC card with the clock pulses generated by the SCI, input the SCKn pin output to the CLK pin of an IC card. The output port of this MCU can be used to output a reset signal.

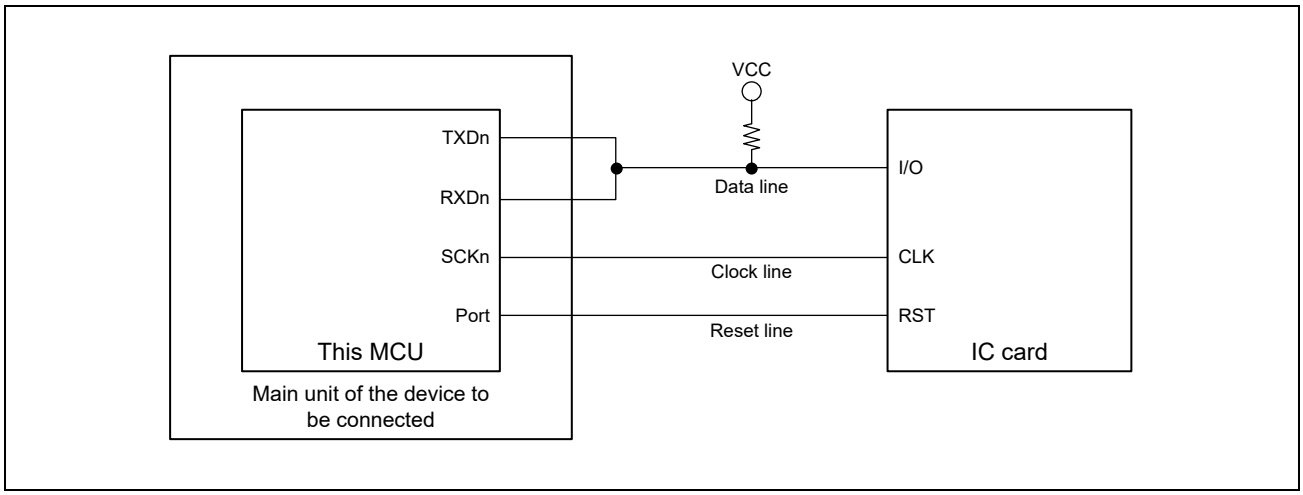
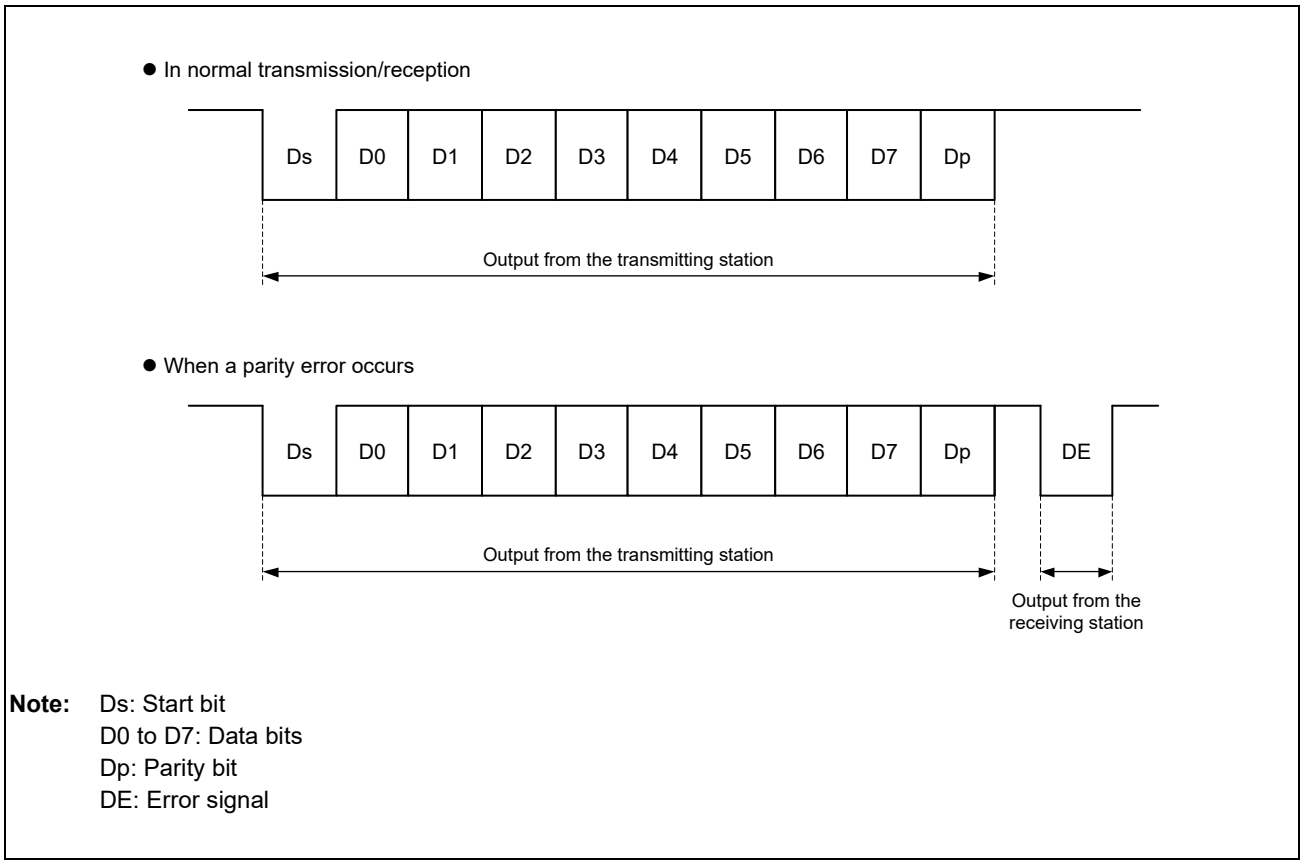


Figure 24.29 Sample Connection with a Smart Card (IC Card)

24.6.2 Data Format (Except in Block Transfer Mode)

Figure 24.30 shows the data transfer formats in smart card interface mode.

- One frame consists of 8-bit data and a parity bit in asynchronous mode.
- During transmission, at least 2 etu (elementary time unit: time required for transferring 1 bit) is secured as a guard time from the end of the parity bit until the start of the next frame.
- If a parity error is detected during reception, a low-level error signal is output for 1 etu after 10.5 etu has passed from the start bit.
- If an error signal is sampled during transmission, the same data is automatically retransmitted after at least 2 etu.



Note: Ds: Start bit
D0 to D7: Data bits
Dp: Parity bit
DE: Error signal

Figure 24.30 Data Formats in Smart Card Interface Mode

For communications with IC cards of the direct convention type and inverse convention type, follow the procedure below.

(1) Direct Convention Type

For the direct convention type, logic levels 1 and 0 correspond to states Z and A, respectively, and data is transferred with LSB first as the start character, as shown in **Figure 24.31**. Therefore, data in the start character in the figure is H'3B.

When using the direct convention type, write 0 to both the SDIR and SINV bits in SCMR. Write 0 to the PM bit in SMR in order to use even parity, which is prescribed by the smart card standard.

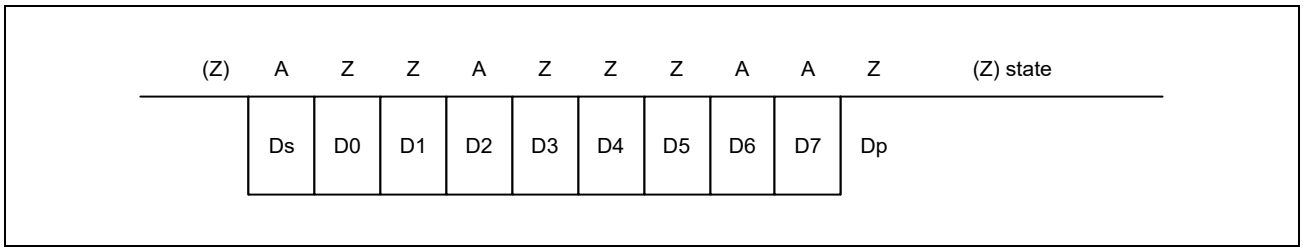


Figure 24.31 Direct Convention (SDIR in SCMR = 0, SINV in SCMR = 0, PM in SMR = 0)

(2) Inverse Convention Type

For the inverse convention type, logic levels 1 and 0 correspond to states A and Z, respectively and data is transferred with MSB first as the start character, as shown in **Figure 24.32**. Therefore, data in the start character in the figure is H'3F. When using the inverse convention type, write 1 to both the SDIR and SINV bits in SCMR. The parity bit is logic level 0 to produce even parity, which is prescribed by the smart card standard, and corresponds to state Z. Since the SINV bit of this MCU only inverts data bits D7 to D0, write 1 to the PM bit in SMR to invert the parity bit for both transmission and reception.

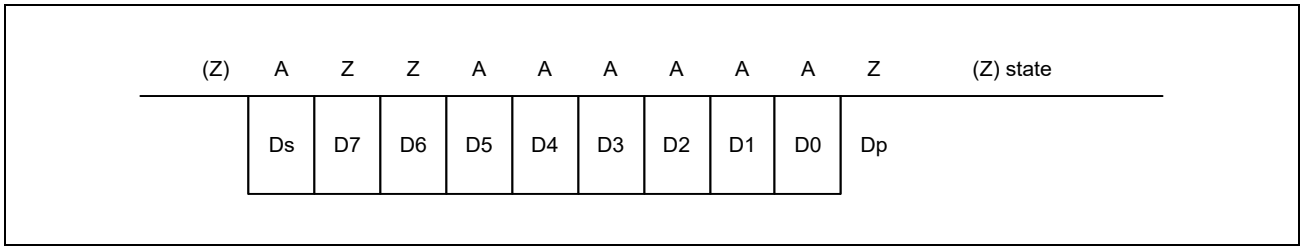


Figure 24.32 Inverse Convention (SDIR in SCMR = 1, SINV in SCMR = 1, PM in SMR = 1)

24.6.3 Block Transfer Mode

Block transfer mode is different from normal smart card interface mode in the following respects.

- Even if a parity error is detected during reception, no error signal is output. Since the PER bit in SSR is set by error detection, clear the PER bit before receiving the parity bit of the next frame.
- During transmission, at least 1 etu is secured as a guard time from the end of the parity bit until the start of the next frame.
- Since the same data is not retransmitted during transmission, the TEND flag in SSR is set 11.5 etu after transmission start.
- In block transfer mode, the ERS flag in SSR indicates the error signal status as in normal smart card interface mode, but the flag is read as 0 because no error signal is transferred.

24.6.4 Receive Data Sampling Timing and Reception Margin

Only the internal clock generated by the on-chip baud rate generator can be used as a transfer clock in smart card interface mode.

In this mode, the SCI can operate on a base clock with a frequency of 32, 64, 372, 256, 93, 128, 186, or 512 times the bit rate according to the settings of the BCP2 bit in SCMR and the BCP[1:0] bits in SMR (the frequency is always 16 times the bit rate in normal asynchronous mode).

For data reception, the falling edge of the start bit is sampled with the base clock to perform internal synchronization. Receive data is sampled on the 16th, 32nd, 186th, 128th, 46th, 64th, 93rd, and 256th rising edges of the base clock so that it can be latched at the middle of each bit as shown in **Figure 24.33**. The reception margin here is determined by the following formula.

$$M = \left| (0.5 - \frac{1}{2N}) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100\ [\%]$$

M: Reception margin (%)
N: Ratio of bit rate to clock (N = 32, 64, 372, 256) D: Duty cycle of clock (D = 0 to 1.0)
L: Frame length (L = 10)
F: Absolute value of clock frequency deviation

Assuming values of F = 0, D = 0.5, and N = 372 in the above formula, the reception margin is determined by the formula below.

$$M = \left\{ 0.5 - \frac{1}{(2 \times 372)} \right\} \times 100\ [\%] = 49.866\%$$

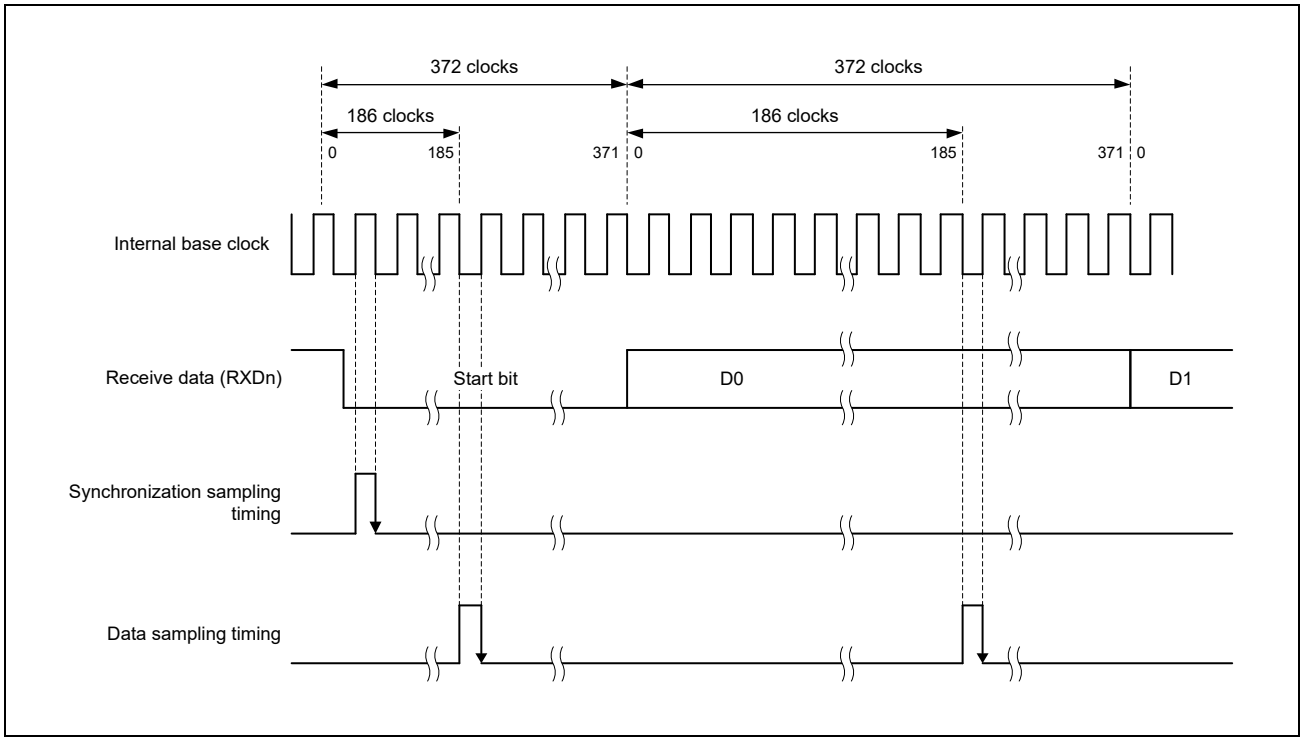


Figure 24.33 Receive Data Sampling Timing in Smart Card Interface Mode (When Clock Frequency is 372 Times the Bit Rate)

24.6.5 SCI Initialization (Smart Card Interface Mode)

Initialize the SCI following the example of flowchart shown in **Figure 24.34**.

Be sure to initialize the SCI before switching from transmission mode to reception mode and vice versa. Even if the RE bit is set to 0, the RDR register is not initialized.

To change reception mode to transmission mode, first check that reception has completed, and then initialize the SCI. At the end of initialization, set TE = 1 and RE = 0. Reception completion can be verified by reading the RXI request, ORER, or PER flag in SSR.

To change transmission mode to reception mode, first check that transmission has completed, and then initialize the SCI. At the end of initialization, set TE = 0 and RE = 1. Transmission completion can be verified by reading the TEND flag in SSR.

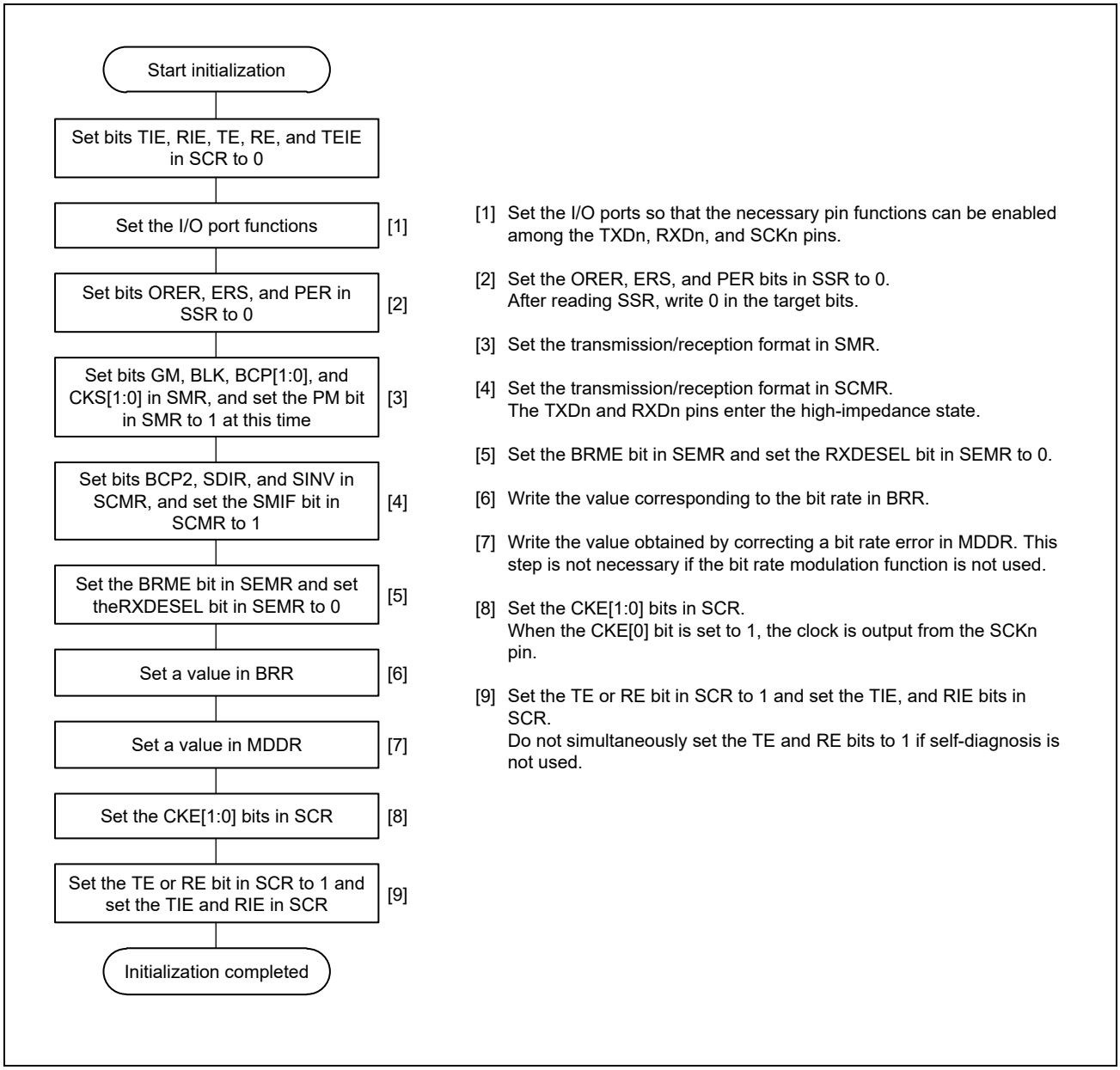


Figure 24.34 Example of SCI Initialization Flowchart (Smart Card Interface Mode)

24.6.6 Serial Data Transmission (Except in Block Transfer Mode)

Serial data transmission in smart card interface mode (except in block transfer mode), in that an error signal is sampled and data can be retransmitted, is different from that in non-smart card interface mode. **Figure 24.35** shows the data retransfer operation during transmission.

1. When an error signal from the receiver end is sampled after one-frame data has been transmitted, the ERS flag in SSR is set to 1. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Clear the ERS flag to 0 before the next parity bit is sampled.
2. For a frame in which an error signal is received, the TEND flag in SSR is not set. Data is retransferred from TDR to TSR allowing automatic data retransmission.
3. If no error signal is returned from the receiver, the ERS flag is not set to 1.
4. In this case, the SCI judges that transmission of one-frame data (including retransfer) has been completed, and the TEND flag is set. If the TIE bit in SCR is 1 at this time, a TXI interrupt request is generated. Writing transmit data to TDR starts transmission of the next data.

Figure 24.37 shows a sample flowchart of serial transmission. All the processing steps are automatically performed using a TXI interrupt request to activate the DMAC.

When the TEND flag in SSR is set to 1 in transmission, if the TIE bit in SCR is 1, a TXI interrupt request is generated. The DMAC is activated by a TXI interrupt request if the TXI interrupt request is specified as a source of DMAC activation beforehand, allowing transfer of transmit data. The TEND flag is automatically set to 0 when the DMAC transfers the data.

If an error occurs, the SCI automatically retransmits the same data. During this retransmission, the TEND flag is kept to 0 and the DMAC is not activated. Therefore, the SCI and DMAC automatically transmit the specified number of bytes, including retransmission in the case of error occurrence. However, since the ERS flag is not automatically cleared, set the RIE bit to 1 beforehand to enable an ERI interrupt request to be generated at error occurrence, and clear the ERS flag to 0. When transmitting/receiving data using the DMAC, be sure to make settings to enable the DMAC before making SCI settings.

For DMAC settings, refer to **Section 14, Direct Memory Access Controller**.

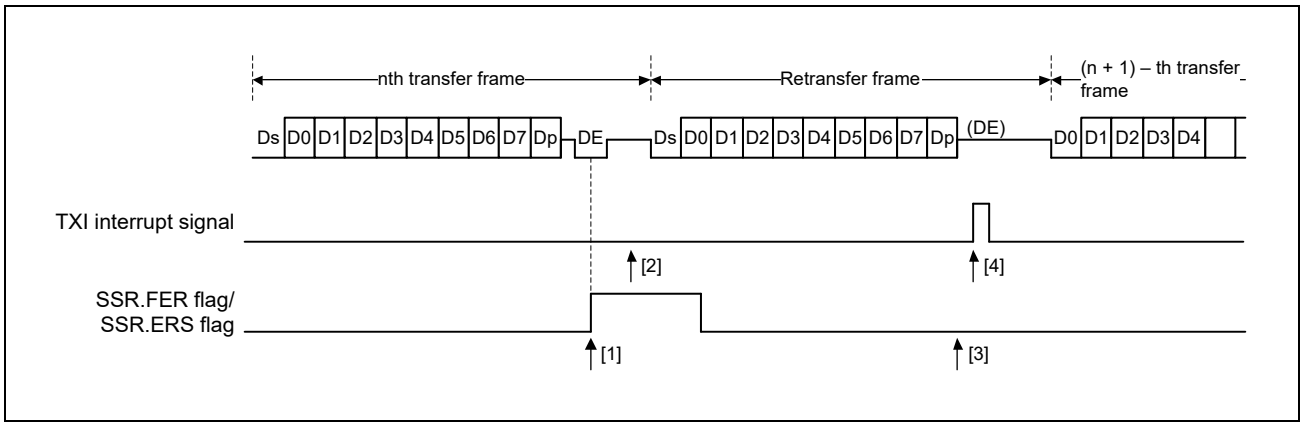


Figure 24.35 Data Retransfer Operation in SCI Transmission Mode

Note that the SSR.TEND flag is set in different timings depending on the GM bit setting in SMR. **Figure 24.36** shows the TEND flag generation timing.

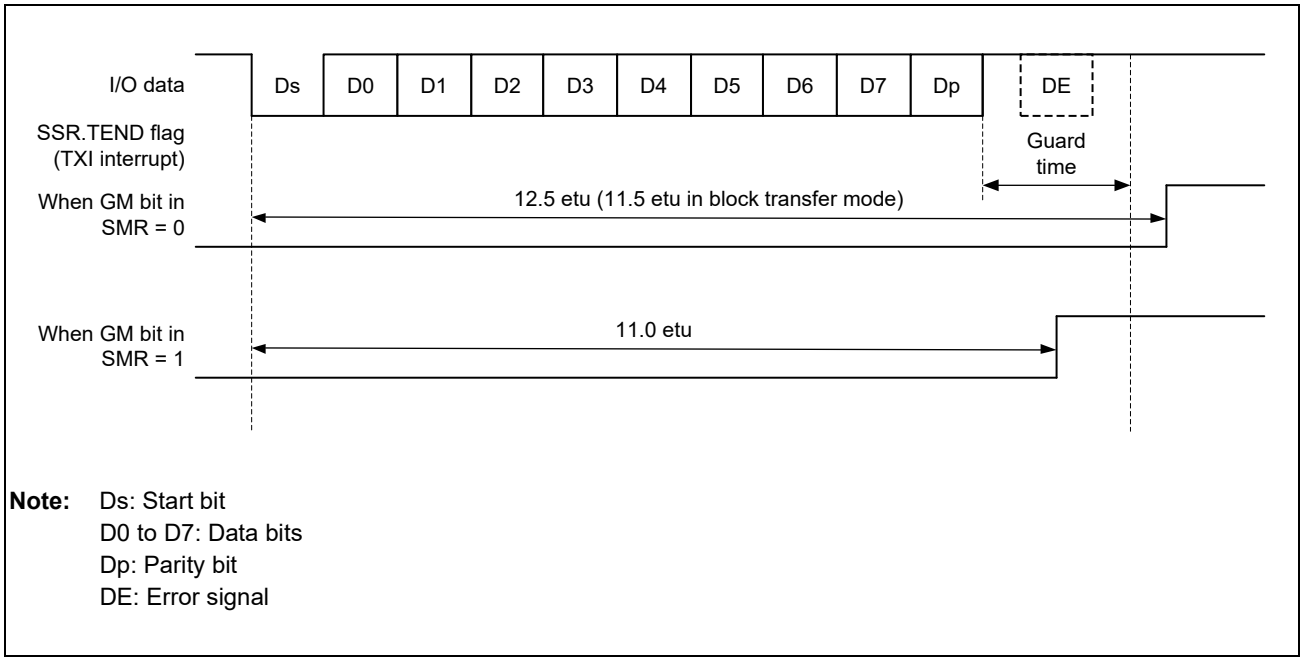


Figure 24.36 SSR.TEND Flag Generation Timing during Transmission

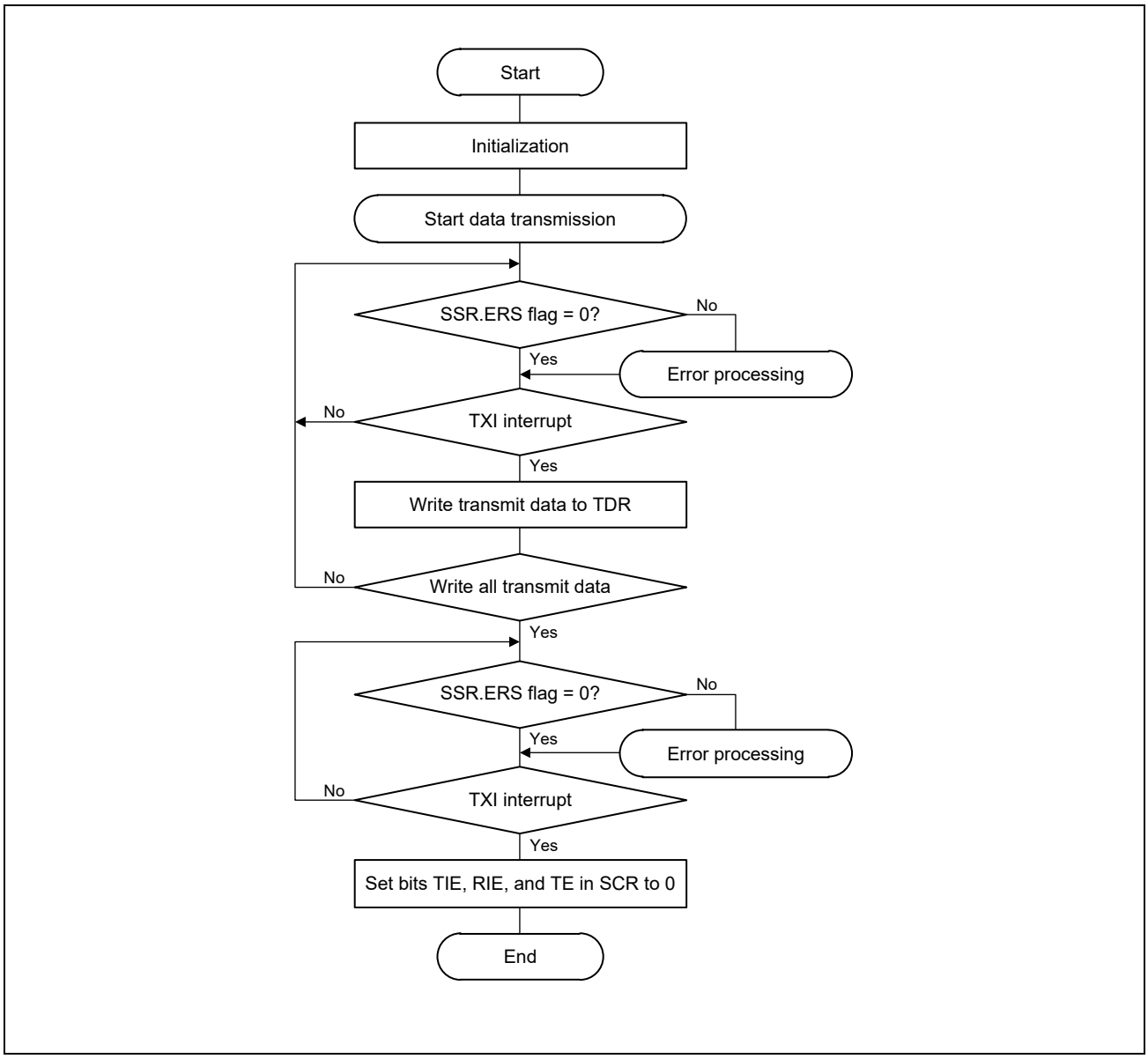


Figure 24.37 Sample Smart Card Interface Transmission Flowchart

24.6.7 Serial Data Reception (Except in Block Transfer Mode)

Serial data reception in smart card interface mode is similar to that in non-smart card interface mode. **Figure 24.38** shows the data retransfer operation in reception mode.

1. If a parity error is detected in receive data, the PER flag in SSR is set to 1. When the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Clear the PER flag to 0 before the next parity bit is sampled.
2. For a frame in which a parity error is detected, no RXI interrupt is generated.
3. When no parity error is detected, the PER flag in SSR is not set to 1.
4. In this case, data is determined to have been received successfully. When the RIE bit in SCR is 1, an RXI interrupt request is generated.

Figure 24.39 shows a sample flowchart for serial data reception. All the processing steps are automatically performed using an RXI interrupt request to activate the DMAC.

In reception, setting the RIE bit to 1 allows an RXI interrupt request to be generated. The DMAC is activated by an RXI interrupt request if the RXI interrupt request is specified as a source of DMAC activation beforehand, allowing transfer of receive data.

If an error occurs during reception and either the ORER or PER flag in SSR is set to 1, a receive error interrupt (ERI) request is generated. Clear the error flag after the error occurrence. If an error occurs, the DMAC is not activated and receive data is skipped. Therefore, the number of bytes of receive data specified in the DMAC is transferred.

Even if a parity error occurs and the PER flag is set to 1 during reception, receive data is transferred to RDR, thus allowing the data to be read.

When a reception is forcibly terminated by setting the RE bit in SCR to 0 during operation, read the RDR register because the received data which has not yet been read may be left in RDR.

NOTE

For operations in block transfer mode, refer to **Section 24.3, Operation in Asynchronous Mode**.

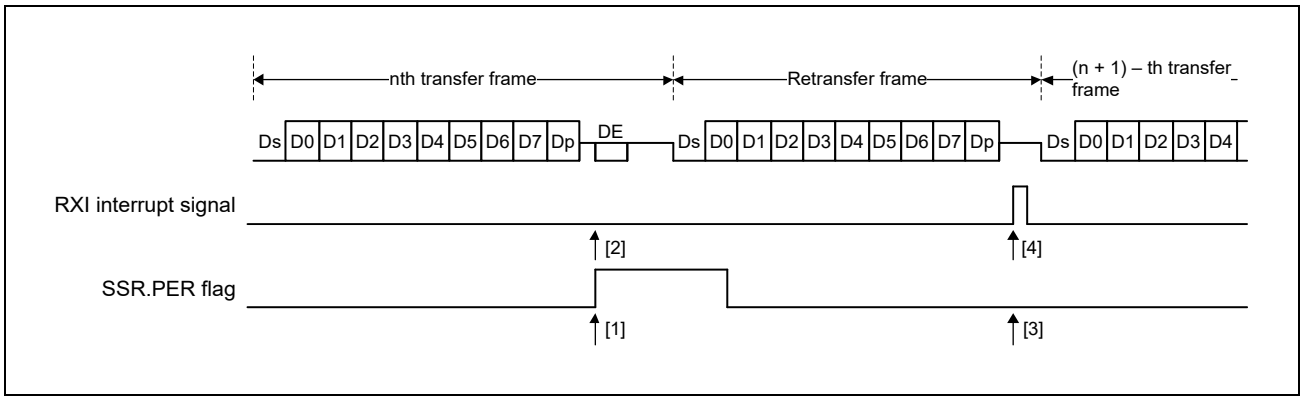


Figure 24.38 Data Retransfer Operation in SCI Reception Mode (Data Retransfer Operation during Reception)

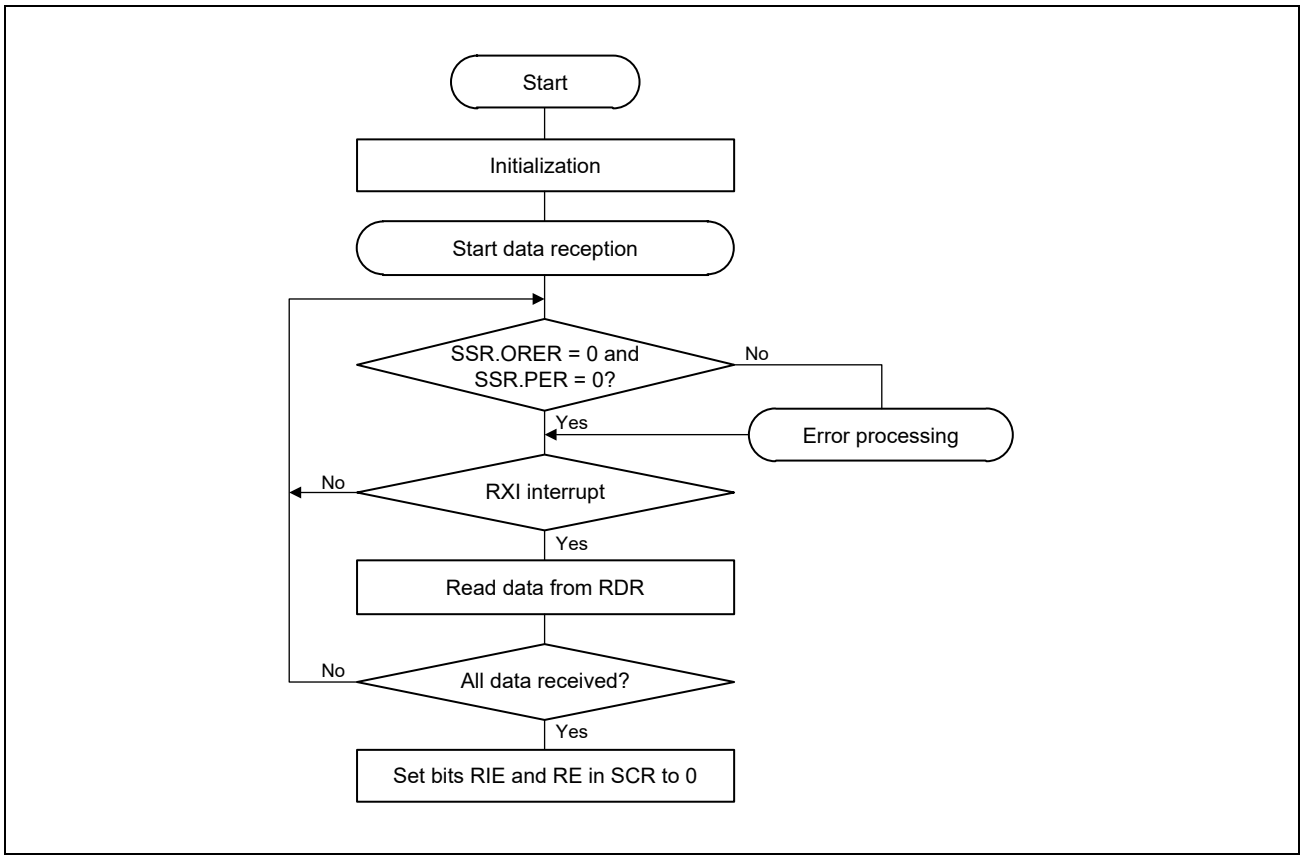


Figure 24.39 Sample Smart Card Interface Reception Flowchart

24.6.8 Clock Output Control

Clock output can be fixed using the CKE[1:0] bits in SCR when the GM bit in SMR is 1. Specifically, the minimum width of a clock pulse can be specified.

Figure 24.40 shows an example of clock output fixing timing when the CKE[0] bit is controlled with GM = 1 and CKE[1] = 0.

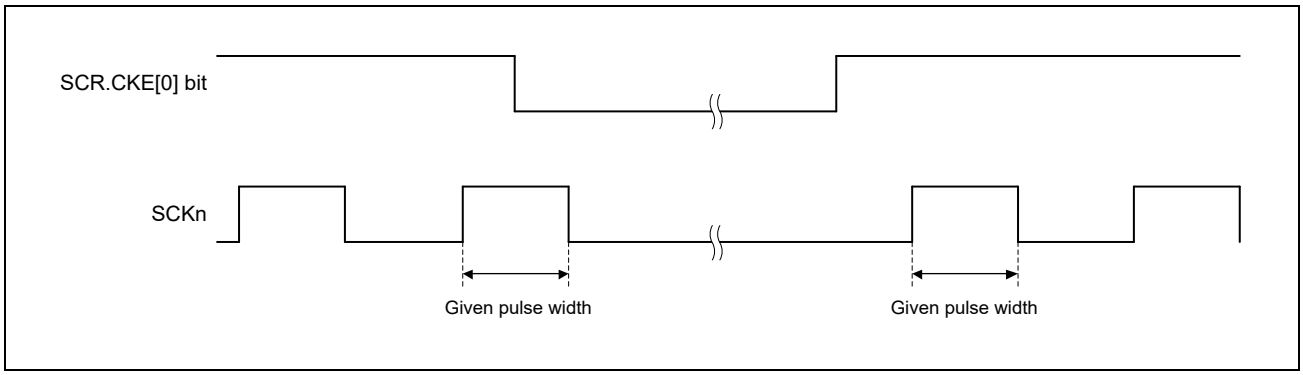


Figure 24.40 Clock Output Fixing Timing

At power-on, use the following procedure to secure the appropriate clock duty cycle.

(1) At Power-On

To secure the appropriate clock duty cycle simultaneously with power-on, use the following procedure.

1. Initially, port input is enabled in the high-impedance state. To fix the potential level, use a pull-up or pull-down resistor.
2. Fix the SCKn pin to the specified output by setting the SCR.CKE[1] bit and I/O port functions.
3. Set SMR and SCMR to enable smart card interface mode.
4. Set the SCR.CKE[0] bit to 1 to start clock output.

24.7 Noise Cancellation Function

Figure 24.41 shows the configuration of the noise filter used for noise cancellation. The noise filter consists of two stages of flip-flop circuits and a match-detection circuit. When the level on the pin matches in three consecutive samples taken at the set sampling interval, the matching level continues to be conveyed internally until the level on the pin again matches in three consecutive samples.

In asynchronous mode, the noise cancellation function can be applied on the RXDn input signal. The period of the base clock (1/16th of a bit-period when SEMR.ABCS = 0 and 1/8th of a bit-period when SEMR.ABCS = 1) is the sampling interval.

If the base clock is stopped with the noise filter enabled and then the clock input is started again, the noise filter operation resumes from where the clock was stopped. If SCR.TE and SCR.RE are set to 0 during base clock input, all of the noise filter flip-flop values are initialized to 1. Accordingly, if the input data is 1 when reception operation resumes, it is determined that a level match is detected and is conveyed to the internal signal. When the level being input corresponds to 0, the initial output of the noise filter is retained until the level matches in three consecutive samples.

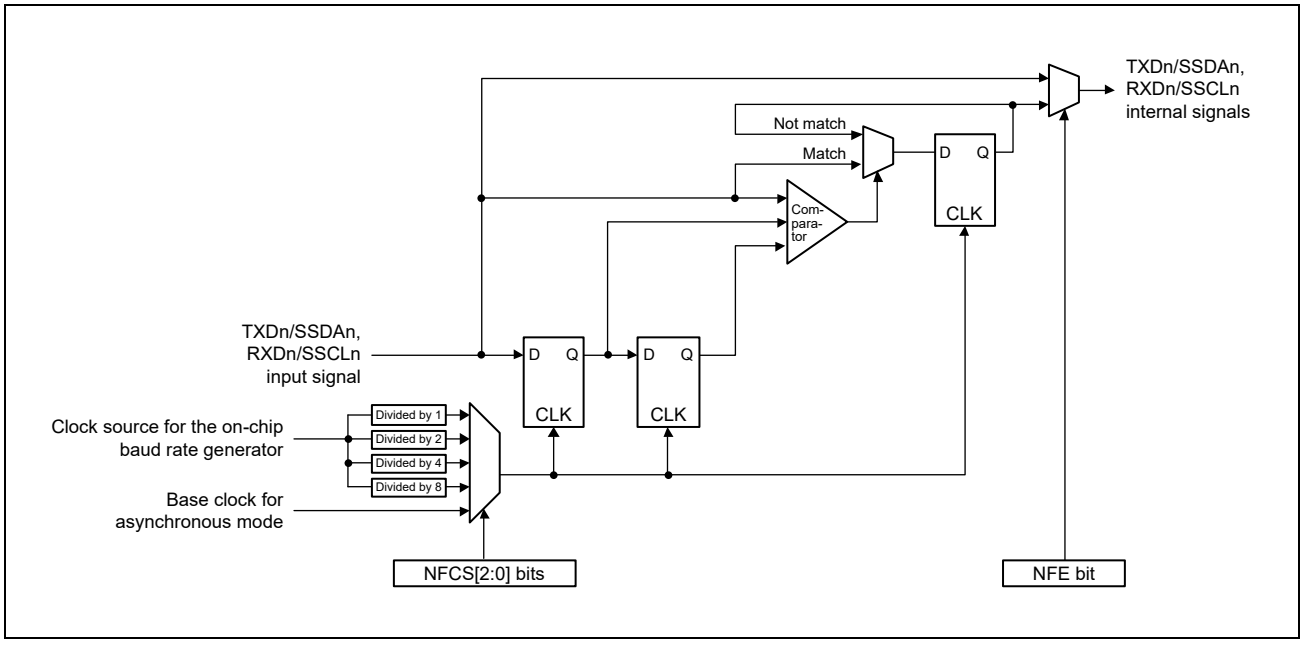


Figure 24.41 Block Diagram of Digital Noise Filter Circuit

24.8 Interrupt Sources

24.8.1 Buffer Operations for TXI and RXI Interrupts

If the conditions for a TXI and RXI interrupt are satisfied while the interrupt status flag in the interrupt controller is 1, the interrupt controller does not output the interrupt request but retains it internally (with a capacity for retention of one request per source).

When the value of the interrupt status flag in the interrupt controller becomes 0, the interrupt request retained within the interrupt controller is output. The internally retained interrupt request is automatically discarded once the actual interrupt is output. Clearing of the corresponding interrupt enable bit (the TIE or RIE bit in the SCR) can also be used to discard an internally retained interrupt request.

24.8.2 Interrupts in Asynchronous Mode and Clock Synchronous Mode

Table 24.20 lists interrupt sources in asynchronous mode and clock synchronous mode. A different interrupt vector is assigned to each interrupt source, and individual interrupt sources can be enabled or disabled with the enable bits in SCR. If the SCR.TIE bit is 1, a TXI interrupt request is generated when transmit data is transferred from the TDR or TDRL register*[1] to the TSR. A TXI interrupt request can also be generated by setting the SCR.TE bit to 1 after setting the SCR.TIE bit to 1 or by using a single instruction to set the SCR.TE and SCR.TIE bit to 1 at the same time. A TXI interrupt request can activate the DMAC to handle data transfer.

A TXI interrupt request is not generated by setting the SCR.TE bit to 1 while the setting of the SCR.TIE bit is 0 or by setting the SCR.TIE bit to 1 while the setting of the SCR.TE bit is 1.*[1]

When new data is not written by the time of transmission of the last bit of the current transmit data and the setting of the SCR.TEIE bit is 1, the SSR.TEND flag becomes 1 and a TEI interrupt request is generated. Furthermore, when the setting of the SCR.TE bit is 1, the SSR.TEND flag retains the value 1 until further transmit data are written to the TDR or TDRL register*[1], and setting the SCR.TEIE bit to 1 leads to the generation of a TEI interrupt request.

Writing data to the TDR or TDRL register*[1] leads to clearing of the SSR.TEND flag and, after a certain time, discarding of the TEI interrupt request.

If the SCR.RIE bit is 1, an RXI interrupt request is generated when received data is stored in the RDR. An RXI interrupt request can activate the DMAC to handle data transfer.

Setting of any from among the ORER, FER, and PER flags in the SSR to 1 while the SCR.RIE bit is 1 leads to the generation of an ERI interrupt request. An RXI interrupt request is not generated at this time. Clearing all three flags (ORER, FER, and PER) leads to discarding of the ERI interrupt request.

Note 1. In the case where asynchronous mode and 9-bit data length are selected

NOTE

To temporarily prohibit TXI interrupts at the time of transmission of the last of the data and so on when you wish a new round of transmission to start after handling of the transmission-completed interrupt, control prohibiting and permitting of the interrupt by using the interrupt request enable bit in the interrupt controller rather than using the SCR.TIE bit. This can prevent the suppression of TXI interrupt requests in the transfer of new data.

Table 24.20 Interrupt Sources

Name	Interrupt Source	Interrupt Flag	DMAC Activation	Priority
ERI	Receive error	ORER, FER, or PER	Not possible	High
RXI	Receive data full	—	Possible	↑
TXI	Transmit data empty	—	Possible	
TEI	Transmit end	TEND	Not possible	Low

24.8.3 Interrupts in Smart Card Interface Mode

Table 24.21 lists interrupt sources in smart card interface mode. A transmit end interrupt (TEI) request cannot be used in this mode.

Table 24.21 SCI Interrupt Sources

Name	Interrupt Source	Interrupt Flag	DMAC Activation	Priority
ERI	Receive error or error signal detection	ORER, PER, or ERS	Not possible	High
RXI	Receive data full	—	Possible	↑
TXI	Transmit data empty	TEND	Possible	Low

Data transmission/reception using the DMAC is also possible in smart card interface mode, similar to in the normal SCI mode. In transmission, when the TEND flag in SSR is set to 1, a TXI interrupt request is generated. This TXI interrupt request activates the DMAC allowing transfer of transmit data if the TXI request is specified beforehand as a source of DMAC activation. The TEND flag is automatically set to 0 when the DMAC transfers the data.

If an error occurs, the SCI automatically retransmits the same data. During the retransmission, the TEND flag is kept to 0 and the DMAC is not activated. Therefore, the SCI and DMAC automatically transmit the specified number of bytes, including retransmission in the case of error occurrence. However, the ERS flag in SSR is not automatically cleared to 0 at error occurrence. Therefore, the ERS flag must be cleared by previously setting the RIE bit in SCR to 1 to enable an ERI interrupt request to be generated at error occurrence.

When transmitting/receiving data using the DMAC, be sure to make settings to enable the DMAC before making SCI settings. For DMAC settings, refer to **Section 14, Direct Memory Access Controller**.

In reception, an RXI interrupt request is generated when receive data is set to RDR. This RXI interrupt request activates the DMAC allowing transfer of receive data if the RXI request is specified beforehand as a source of DMAC activation. If an error occurs, the error flag is set. Therefore, the DMAC is not activated and an ERI interrupt request is issued to the CPU instead; the error flag must be cleared.

24.9 Usage Notes

24.9.1 Break Detection and Processing

When a framing error is detected, a break can be detected by reading the RXDn pin value directly. In a break, the input from the RXDn pin becomes all 0s, and so the FER flag in SSR is set to 1 (framing error has occurred), and the PER flag in SSR may also be set to 1 (parity error has occurred). The SCI continues the receive operation even after a break is received. Therefore, note that even if the FER flag is set to 0 (no framing error occurred), it will be set to 1 again. When the SEMR.RXDESEL bit is 1, the SCI sets the SSR.FER flag to 1 and stops receiving operation until a start bit of the next data frame is detected. If the SSR.FER flag is set to 0 at this time, the SSR.FER flag retains 0 during the break.

When the RXDn pin is set to 1 and the break ends, detecting the beginning of the start bit at the first falling edge of the RXDn pin allows the SCI to start the receiving operation.

24.9.2 Mark State and Production of Breaks

When the SCR.TE bit is 0 (serial transmission is disabled), setting the I/O port function makes selection of the level and direction (input or output) of the TXDn pin possible. If this is done, the TXDn pin can be placed in the mark state to send a break at the time of data transmission. Until the SCR.TE bit is set to 1 (serial transmission is enabled), the I/O port function is used to set the TXDn pin to output 1 and set the pin mode to a general I/O port pin, and thus place the communication line in the mark state (state of having the value 1). On the other hand, to output a break at the time of data transmission, set the TXDn pin to output 0 and make the pin mode settings for a general I/O port pin. When the SCR.TE bit is set to 0, the transmitter is initialized regardless of the current state of transmission.

24.9.3 Receive Error Flags and Transmit Operations (Clock Synchronous Mode)

Transmission cannot be started when a receive error flag (ORER) in SSR is set to 1, even if data is written to TDR. Be sure to set the receive error flags to 0 before starting transmission. Note also that the receive error flags cannot be set to 0 even if the RE bit in SCR is set to 0 (serial reception is disabled).

24.9.4 Writing Data to TDR

Data can be written to TDR, TDRH, and TDRL. However, if new data is written to TDR, TDRH, and TDRL when transmit data is remaining in TDR, TDRH, and TDRL, the previous data in TDR, TDRH, and TDRL is lost because it has not been transferred to TSR yet. Be sure to write transmit data to TDR, TDRH, and TDRL in the TXI interrupt request handling routine.

24.9.5 Restrictions on Clock Synchronous Transmission (Clock Synchronous Mode)

When the external clock source is used as a synchronization clock, the following restrictions apply.

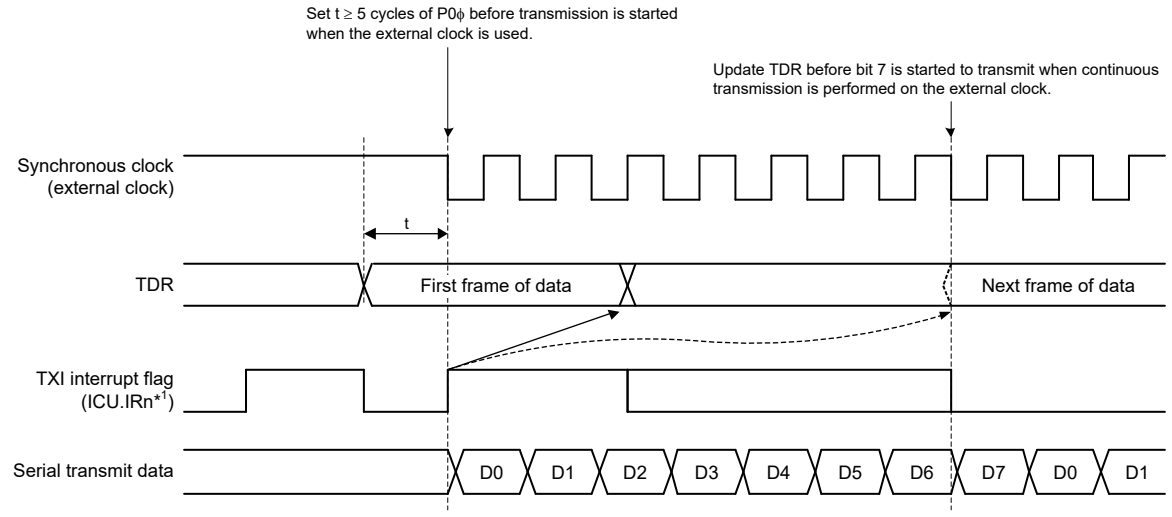
(1) Start of transmission

Update TDR by the CPU or DMAC and wait for at least five P0ϕ cycles before allowing the transmit clock to be input (see **Figure 24.42**).

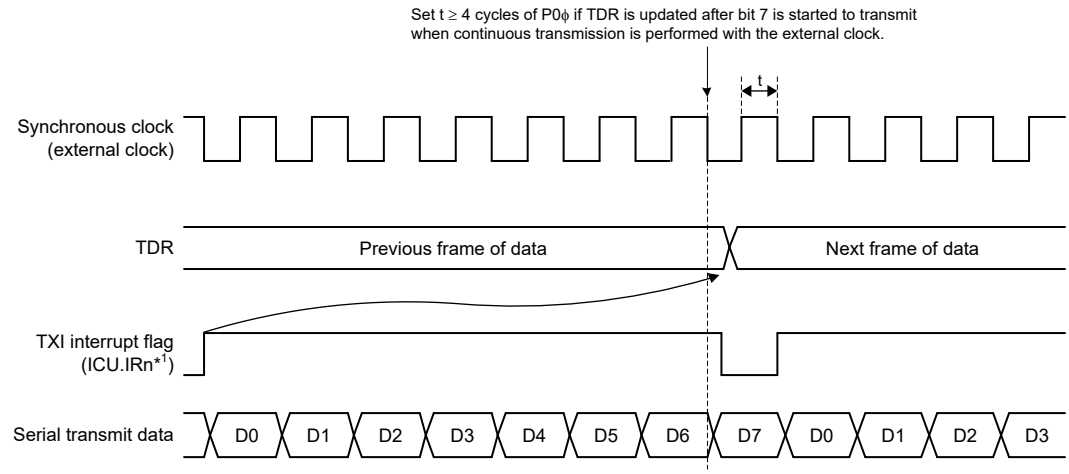
(2) Continuous transmission

(a) Write the next transmit data to TDR or TDRL before the falling edge of the transmit clock (bit 7) (see **Figure 24.42**).

(b) When updating TDR after bit 7 has started to transmit, update TDR while the synchronization clock is in the low-level period, and set the high-level width of the transmit clock (bit 7) to four P0ϕ cycles or longer (see **Figure 24.42**).



(a) Start of transmission and Continuous transmission (1)



(b) Continuous transmission (2)

Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 24.42 Restrictions on Use of External Clock in Clock Synchronous Transmission

24.9.6 Restrictions on Using DMAC

When using the DMAC or DTC to read RDR, RDRH, and RDRL, be sure to set the receive data full interrupt (RXI) as the activation source of the relevant SCI.

24.9.7 Notes on Starting Transfer

At the point where transfer starts when the interrupt status flag (IRn.IR bit) in the interrupt controller is 1, follow the procedure below to clear interrupt requests before permitting operations (by setting the SCR.TE or SCR.RE bit to 1). For details on the interrupt status flag, refer to **Section 8, Interrupt Controller**.

- Confirm that transfer has stopped (the setting of the SCR.TE or SCR.RE bits is 0).
- Set the corresponding interrupt enable bit (SCR.TIE or SCR.RIE) to 0.
- Read the corresponding interrupt enable bit (SCR.TIE or SCR.RIE bit) to check that it has become 0.
- Set the interrupt status flag (IRn.IR bit) in the interrupt controller to 0.

24.9.8 External Clock Input in Clock Synchronous Mode

In clock synchronous mode, the external clock SCKn must be input as follows:

High-pulse period, low-pulse period = 2 P0ϕ cycles or more, period = 6 P0ϕ cycles or more

24.9.9 Note on Transmit Enable Bit (TE bit)

When setting the SCR.TE bit to 0 (serial transmission is disabled) while the pin function is “TXDn”, output of the pin becomes high impedance.

Prevent the TXDn line from becoming high impedance by any of the following ways:

(1) Connect a pull-up resistor to the TXDn line.

(2) Change the pin function to “general-purpose I/O port, output” before setting the SCR.TE bit to 0. Set the SCR.TE bit to 1 before changing the pin function to “TXDn”.

24.10 IrDA Communications

The channel 0 serial communications interface (SCI) is capable of working with the Infrared Data Association (IrDA) module to handle transfer through IrDA communications in conformance with version 1.0 of the IrDA protocol.

Using the IRE bit in IRCR to enable the IrDA function leads to encoding and decoding of the SCI0_TXD and SCI0_RXD signals of the channel 0 serial communications interface, respectively, to convert them to and from waveforms in conformance with version 1.0 of the IrDA protocol. Infrared transfer in conformance with version 1.0 of the IrDA protocol can be handled by connecting a transceiver for use in infrared communications.

In version 1.0 of the IrDA protocol, IrDA transfer starts at a transfer rate of 9600 bps. The transfer rate is changeable during transfer as required. The IrDA module itself does not have a function for automatically changing the transfer rate. Accordingly, change the transfer rate by changing that for the serial communications interface.

A block diagram and pin configuration for IrDA transfer are shown in **Figure 24.43** and **Table 24.22**, respectively.

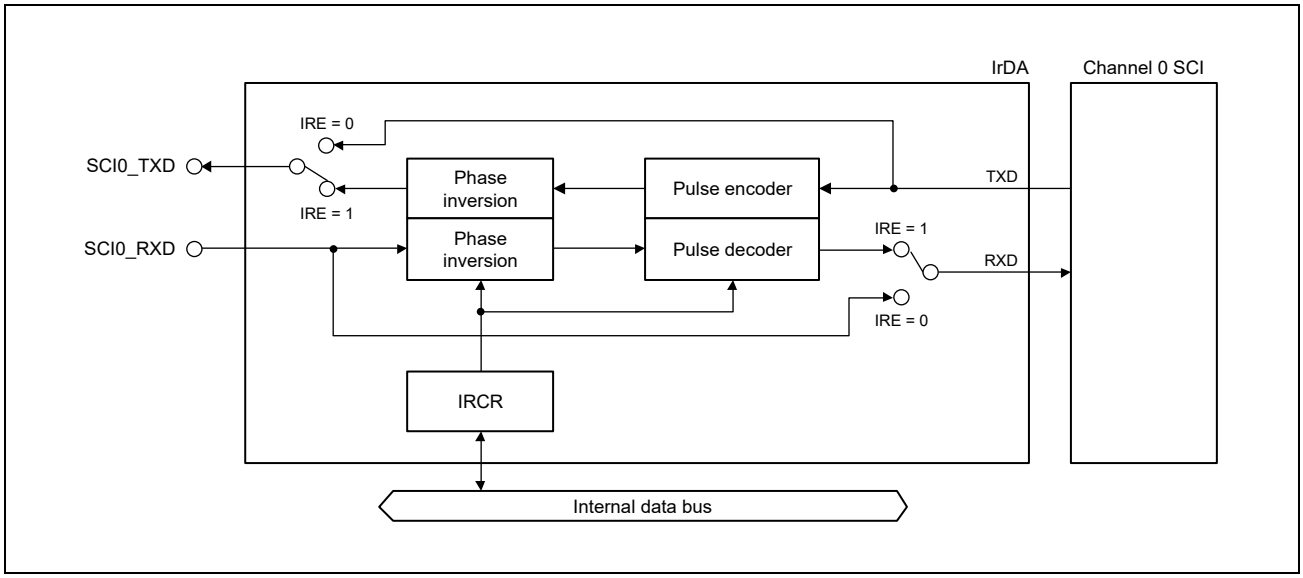


Figure 24.43 Block Diagram

Table 24.22 Pin Configuration

Pin Name	Symbol	I/O	Function
IrDA data transmission pin	SCI0_TXD	Output	IrDA data transmission output
IrDA data reception pin	SCI0_RXD	Input	IrDA data reception input

24.11 Description of the IrDA Register

Table 24.23 shows the register configuration.

BASE Address (Cortex-A55 Address Space): H'0_1004_CC00
BASE Address (Cortex-M33 Address Space Non-Secure): H'4004_CC00
BASE Address (Cortex-M33 Address Space Secure): H'5004_CC00

Table 24.23 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
IrDA Control Register	IRCR	R/W	H'00	H'0_1004_CC00	8

24.11.1 IrDA Control Register (IRCR)

IRCR sets the operation of the IrDA module.

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	IRE	IRCKS[2:0]			IRTXINV	IRRXINV	—	—
Initial Value	0	0	0	0	0	0	0	0
R/W								

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	—		R	Reserved These bits are read as 0. The write value should be 0.
b2	IRRXINV		R/W	SCI_RXD0 Data Level Switching Sets inversion of the logic level of inputs on SCI0_RXD. 0: The input on SCI0_RXD is used without change as the received data. 1: The input on SCI0_RXD is inverted before it is treated as the received data.
b3	IRTXINV		R/W	SCI_TXD0 Data Level Switching Sets inversion of the logic level of outputs on SCI0_TXD. 0: Data for transmission is output without change to SCI0_TXD. The pulse-width setting in the IRCKS bits determines the period over which the pulses are at the high level. 1: Data for transmission is inverted before being output to SCI0_TXD. The pulse-width setting in the IRCKS bits determines the period over which the pulses are at the low level.
b6 to b4	IRCKS[2:0]		R/W	IrDA Clock Select Selects the pulse width for use in encoding output pulses from SCI0_TXD when the setting of the IRE bit is 1. 000: Bit rate × 3/16 Settings other than above are prohibited.
b7	IRE		R/W	IrDA Enable Sets the function for use with the SCI0_TXD and SCI0_RXD pins as normal serial or IrDA. 0: Disables the IrDA function. Data from the TXD pin of the channel 0 serial communications interface is output without change to SCI0_TXD. Data on SCI0_RXD is output without change to the RXD pin of the channel 0 serial communications interface. 1: Enables the IrDA function. Data from the TXD pin of the channel 0 serial communications interface is encoded before being output through SCI0_TXD. Data on SCI0_RXD is decoded before being output to the RXD pin of the channel 0 serial communications interface.

24.12 IrDA Operation

24.12.1 Flow of Settings for IrDA Operation

Set up IrDA operation by following the procedure listed below.

1. Make the general-purpose I/O port-pin settings.
2. Set the IRCR register.
3. Set the registers related to the serial communications interface.

24.12.2 Transmission

In transmission with the IrDA function enabled, serial data (UART frames of data) from the TXD pin of the serial communications interface are converted to IR frames (see **Figure 24.44**). If the IRTXINV bit is set to 0, when the value of the serial data is 0, a high-level pulse with the width equivalent to three sixteenths of the bit period is output to the SCI0_TXD pin by default. In the IrDA protocol, the specifiable high-level pulse widths are from a minimum of 1.41 µs to a maximum of (3/16 + 2.5%) × bit period or (3/16 × bit period) + 1.08 µs. On the other hand, if the value of the serial data is 1, no high-level pulse is output.

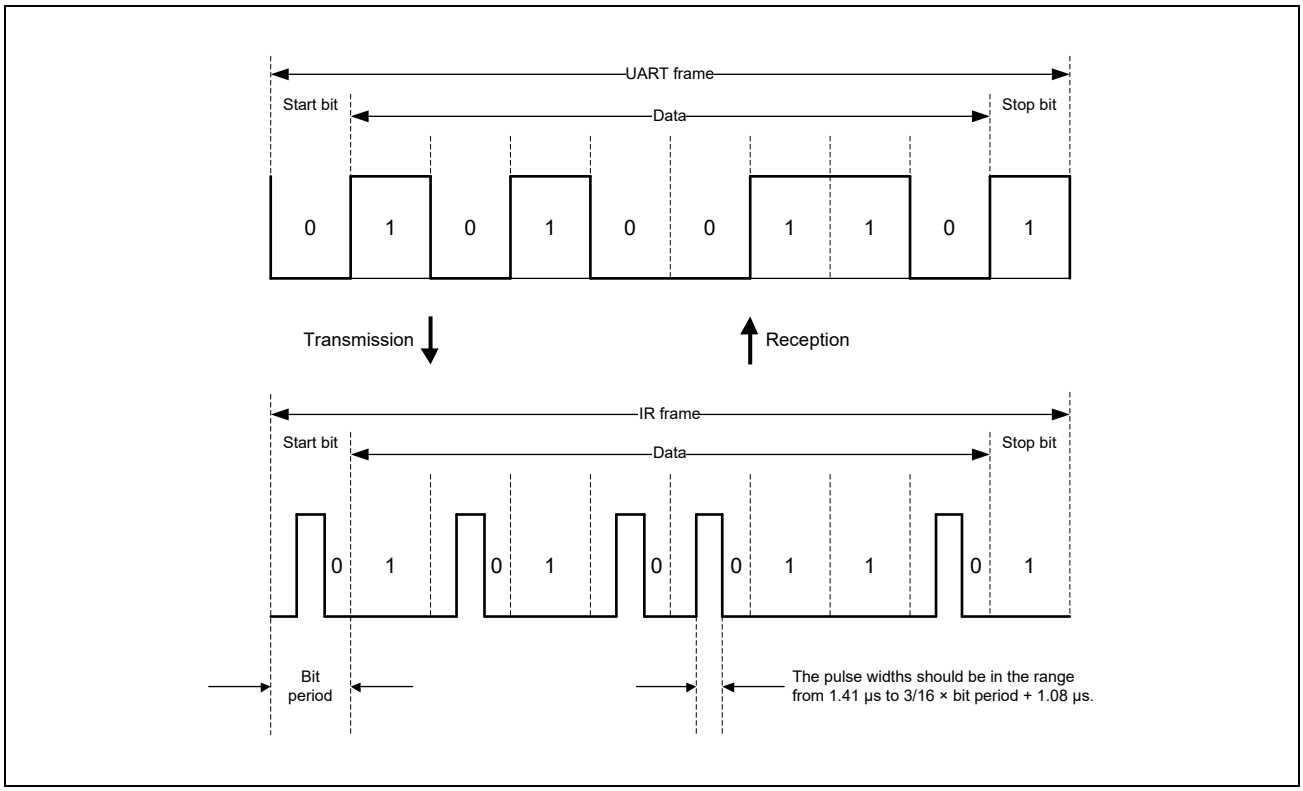


Figure 24.44 Example of the Operation in IrDA Transfer

24.12.3 Reception

In reception with the IrDA function enabled, IR frames of data from the SCI0_RXD pin are converted to serial data and output to the RXD pin of the serial communications interface. If the setting of the IRRXINV bit is 0, the value 0 is output when a high-level pulse is detected and the value 1 is output when no high-level pulse is detected during each bit period. Note that pulses with a width shorter than the minimum pulse width of 1.41 µs cannot be recognized.

24.12.4 Relationship between Bit Rate and High-Level Pulse Width

The relationship between the Bit Rate during transmission and the High-Level Pulse is shown in **Table 24.24**.

Table 24.24 Upper Row: Bit Rate (bps); Lower Row: Bit Period × 3/16 (µs)

2400	9600	19200	38400	57600	115200
78.13	19.53	9.77	4.88	3.26	1.63

24.13 Note on IrDA Usage

24.13.1 Minimum Pulse Width in Reception

Pulses with a width shorter than the minimum pulse width of 1.41 μs cannot be recognized

24.13.2 Base Clock in Asynchronous Mode for the Serial Communications Interface

The IrDA module works with the serial communications interface by receiving the base clock with a frequency 16 times the bit rate for communications from the interface. For the serial communications interface, either 16 or 8 cycles of the base clock are selectable as the bit period. On the other hand, the IrDA module is only capable of operation when the setting for the bit period of the serial communications interface is 16 cycles of the base clock.

25. I²C Bus Interface

This section gives an overall description of the I²C bus interface (RIIC).

The first section describes the features specific to this LSI, including the number of units and the register base addresses. The subsequent sections describe the RIIC's functions and registers.

25.1 Features

25.1.1 Channels

This LSI has the following number of channels of the I²C bus interface (RIIC).

Table 25.1 Channels of RIIC

Item	Description
Number of channels	4
Name	RIICn (n = 0 to 3)

Table 25.2 Index

Index	Description
n	Throughout this section, the individual channels of the I²C bus interface are identified by the index "n" (n = 0 to 3); for example, RIICnCR1 for the I²C bus control register 1.

25.1.2 Register Base Addresses

The base address <RIICn_base> of each RIICn is listed in the following table.

All RIICn register addresses are given as values obtained by adding offsets to the register base address <RIICn_base> for each channel.

Table 25.3 Register Base Address

Channel	Base Address Name	Base Address
RIIC0	<RIIC0_base>	H'0_1009_0000 (Cortex-A55 Address Space)
RIIC1	<RIIC1_base>	H'0_1009_0400 (Cortex-A55 Address Space)
RIIC2	<RIIC2_base>	H'0_1009_0800 (Cortex-A55 Address Space)
RIIC3	<RIIC3_base>	H'0_1009_0C00 (Cortex-A55 Address Space)
RIIC0	<RIIC0_base>	H'4009_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
RIIC1	<RIIC1_base>	H'4009_0400 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
RIIC2	<RIIC2_base>	H'4009_0800 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
RIIC3	<RIIC3_base>	H'4009_0C00 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
RIIC0	<RIIC0_base>	H'5009_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)
RIIC1	<RIIC1_base>	H'5009_0400 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)
RIIC2	<RIIC2_base>	H'5009_0800 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)
RIIC3	<RIIC3_base>	H'5009_0C00 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Note: Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

25.1.3 External I/O Signals

The following table shows the external I/O signals of the RIIC.

Table 25.4 RIICn Pin Configuration

Channel	Alternative Port Pin Name	Function
RIIC0	RIIC0_SCL	RIIC0 serial clock I/O pin
	RIIC0_SDA	RIIC0 serial data I/O pin
RIIC1	RIIC1_SCL	RIIC1 serial clock I/O pin
	RIIC1_SDA	RIIC1 serial data I/O pin
RIIC2	RIIC2_SCL	RIIC2 serial clock I/O pin
	RIIC2_SDA	RIIC2 serial data I/O pin
RIIC3	RIIC3_SCL	RIIC3 serial clock I/O pin
	RIIC3_SDA	RIIC3 serial data I/O pin

Table 25.5 I^2C Bus Interface

Channel Number*3	Pin Name*1	I/O	Function	This LSI Output Buffer Type*2
0	SCL	I/O	SCL: I^2C serial clock input/output pin SDA: I^2C serial data input/output pin	OD
	SDA			
1	SCL			OD
	SDA			
2	SCL			LVTTL
	SDA			
3	SCL			LVTTL
	SDA			

Note 1. The actual pin names are RIICnSCL and RIICnSDA.

Note 2. Output buffer type: “OD” is open drain buffer and “LVTTL” is low level drive only LVTTL buffer.

Note 3. AC specification is different between channel 0, 1 and channel 2, 3.

25.2 Overview

25.2.1 Functional Overview

Communications format

- I^2C bus format or SMBus format
- Master mode or slave mode selectable
- Automatic securing of the various set-up times, hold times, and bus-free times for the transfer rate

Transfer rate

Up to 1 Mbps

SCL clock

For master operation, the duty cycle of the SCL clock is selectable in the following range. 4% < Duty cycle < 96%.

Issuing and detecting conditions

- Start, restart, and stop conditions are automatically generated.
- Start conditions (including restart conditions) and stop conditions are detected.

Slave address

- Up to three slave-address settings can be made.
- Seven- and ten-bit address formats are supported (along with the use of both at once).
- General call addresses, device ID addresses, and SMBus host addresses are detected.

Acknowledgement

- For transmission, the acknowledge bit is automatically loaded
  - Transfer of the next data for transmission can be automatically suspended on detection of a not-acknowledge bit.
- For reception, the acknowledge bit is automatically transmitted
  - If a wait between the eighth and ninth clock cycles has been selected, software control of the value in the acknowledge field in response to the received value is possible.

Wait function

- In reception, the following periods of waiting can be obtained by holding the clock signal (SCL) at the low level:
  - Waiting between the eighth and ninth clock cycles
  - Waiting between the ninth clock cycle and the first clock cycle of the next transfer (WAIT function)

SDA output delay function

Timing of the output of transmitted data, including the acknowledge bit, can be delayed.

Arbitration

- For multi-master operation
  - Operation to synchronize the SCL (clock) signal in cases of conflict with the SCL signal from another master is possible.
  - When issuing the start condition would create conflict on the bus, loss of arbitration is detected by testing for non-matching between the internal signal for the SDA line and the level on the SDA line.
  - In master operation, loss of arbitration is detected by testing for non-matching between the signal on the SDA line and the internal signal for the SDA line.
- Loss of arbitration due to detection of the start condition while the bus is busy is detectable (to prevent the issuing of double start conditions).
- Loss of arbitration in transfer of a not-acknowledge bit due to the internal signal for the SDA line and the level on the SDA line not matching is detectable.
- Loss of arbitration due to non-matching of internal and line levels for data is detectable in slave transmission.

Timeout function

The internal time-out function is capable of detecting long-interval stop of the SCL (clock signal).

Noise removal

The interface incorporates analog noise filters and digital noise filters for input on the RIICnSCL and RIICnSDA pins, and the width for noise cancellation by the digital noise filters is adjustable by software.

Interrupt sources

- Eight sources:
  - Transmission complete
  - Receive-data-full
  - Transmit-data-empty
  - Detection of a stop condition
  - Detection of a start condition
  - Reception of a NACK
  - Arbitration lost
  - Timeout

Low power consumption function

Module-stop state can be set.

25.2.2 Block Diagram

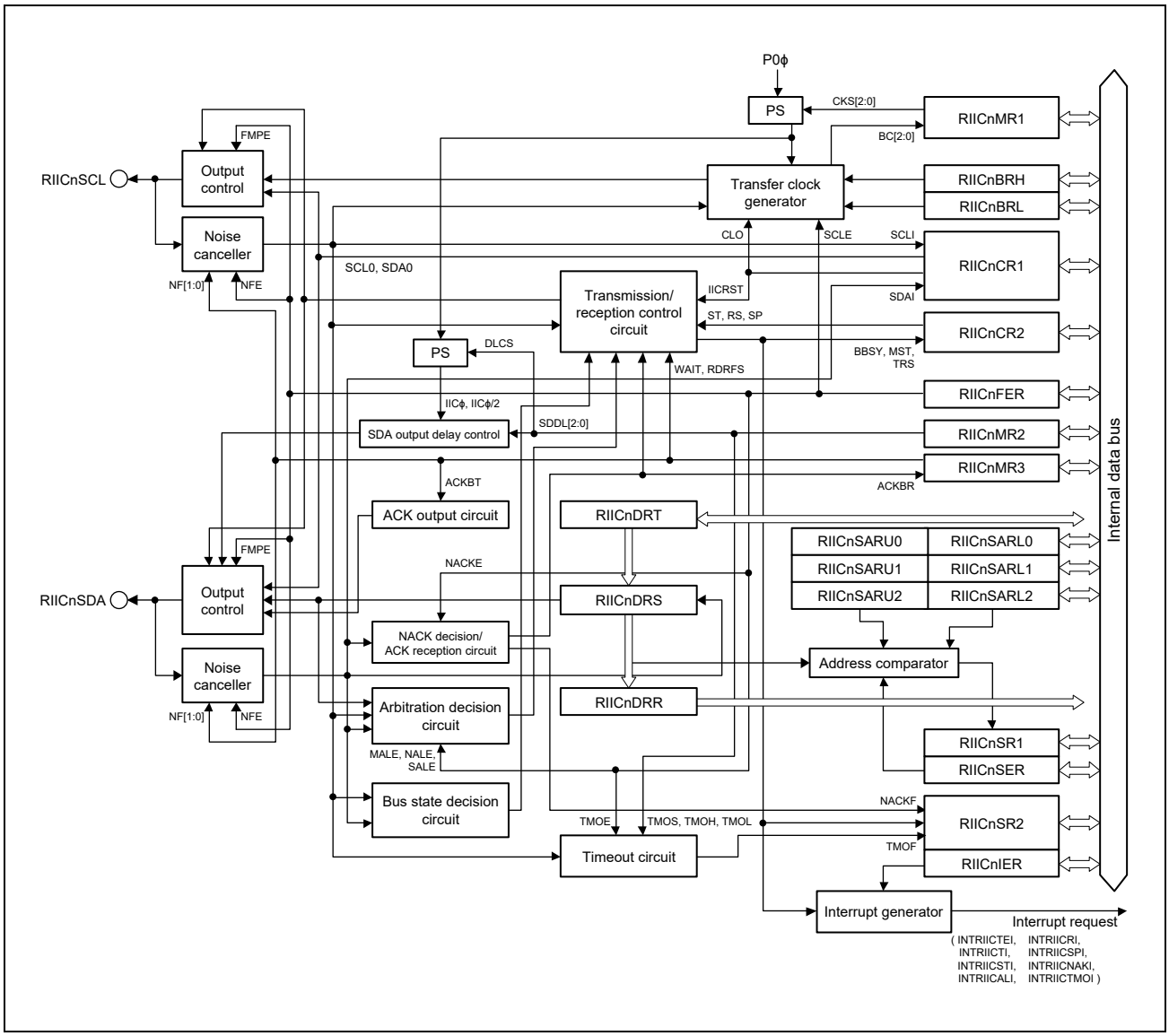


Figure 25.1 Block Diagram of RIIC

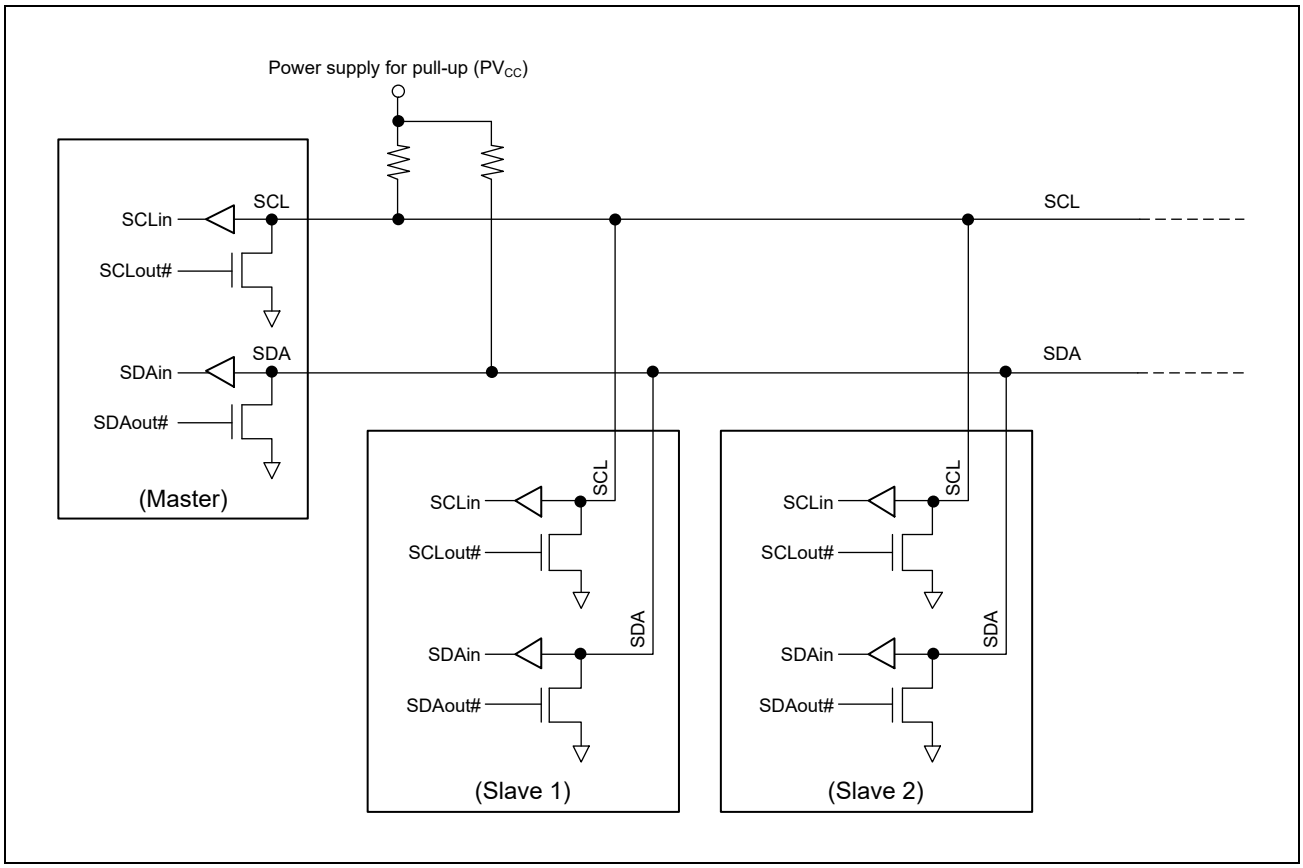


Figure 25.2 Connections to the External Circuit by the I/O Pins (I^2C Bus Configuration Example)

RIICnSCL and RIICnSDA are Schmitt input/open-drain output pins for both master and slave operations. Because the output is open drain, an external pull-up resistor is required.

25.3 Registers

Table 25.6 RIICn (n = 0 to 3) Register Configuration

Channel	Register Name	Abbreviation	Offset Address	Access Size
RIICn	I²C bus control register 1	ICCR1	H'00	8
(n = 0 to 3)	I²C bus control register 2	ICCR2	H'01	8
	I²C bus mode register 1	ICMR1	H'02	8
	I²C bus mode register 2	ICMR2	H'03	8
	I²C bus mode register 3	ICMR3	H'04	8
	I²C bus function enable register	ICFER	H'05	8
	I²C bus status enable register	ICSER	H'06	8
	I²C bus interrupt enable register	ICIER	H'07	8
	I²C bus status register 1	ICSR1	H'08	8
	I²C bus status register 2	ICSR2	H'09	8
	I²C slave address register Lower 0	ICSARL0	H'0A	8
	I²C slave address register Lower 1	ICSARL1	H'0C	8
	I²C slave address register Lower 2	ICSARL2	H'0E	8
	I²C slave address register Upper 0	ICSARU0	H'0B	8
	I²C slave address register Upper 1	ICSARU1	H'0D	8
	I²C slave address register Upper 2	ICSARU2	H'0F	8
	I²C bus bit rate low-level register	ICBRL	H'10	8
	I²C bus bit rate high-level register	ICBRH	H'11	8
	I²C bus transmit data register	ICDRT	H'12	8
	I²C bus receive data register	ICDRR	H'13	8
	I²C bus shift register	ICDRS	—	—

25.3.1 I²C Bus Control Register 1 (RIICnCR1)

Access Size: RIICnCR1 is a 8-bit readable/writable register.

Address(es): RIICnCR1: <RIICn_base> + H'0000

Initial Value: H'1F. This register is initialized by any reset.

Bit	7	6	5	4	3	2	1	0
	ICE	IICRST	CLO	SOWP	SCLO	SDAO	SCLI	SDAI
Initial Value	0	0	0	1	1	1	1	1
R/W	R/W	R/W	R/W	W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	ICE	0	R/W	I²C Bus Interface Enable 0: Output to the RIICnSCL and RIICnSDA pins is disabled. (Input to the RIICnSCL and RIICnSDA pins is enabled.) 1: Enabled (RIICnSCL and RIICnSDA pins are in the driving state) (An RIIC reset or an internal reset is selected according to the combination of this bit and IICRST bit settings.)
6	IICRST	0	R/W	I²C Bus Interface Internal Reset 0: Clears the RIIC reset or internal reset. 1: Initiates the RIIC reset or internal reset. (Clears the bit counter and the SCL/SDA output latch)
5	CLO	0	R/W	Extra SCL Clock Cycle Output 0: Does not output an extra SCL clock cycle (default). 1: Outputs an extra SCL clock cycle. (The CLO bit is cleared automatically after one clock cycle is output.)
4	SOWP*2	1	W	SCLO/SDAO Write Protect 0: Allows the SCLO and SDAO bits to be rewritten. 1: Bits SCLO and SDAO are protected. (This bit is read as 1.)
3	SCLO*1, *2	1	R/W	SCL Output Control • Read: 0: RIICnSCL pin output is at a low level. 1: RIICnSCL pin is in a high-impedance state. • Write: 0: Changes the RIICnSCL pin output to a low level. 1: Changes the RIICnSCL pin in a high-impedance state. (High level output is achieved through an external pull-up resistor.)
2	SDAO*1,*2	1	R/W	SDA Output Control • Read: 0: RIICnSDA pin output is at a low level. 1: RIICnSDA pin is in a high-impedance state. • Write: 0: Changes the RIICnSDA pin output to a low level. 1: Changes the RIICnSDA pin in a high-impedance state. (High level output is achieved through an external pull-up resistor.)
1	SCLI	1	R	SCL Bus Input Monitor 0: RIICnSCL pin input is at a low level. 1: RIICnSCL pin input is at a high level.
0	SDAI	1	R	SDA Bus Input Monitor 0: RIICnSDA pin input is at a low level. 1: RIICnSDA pin input is at a high level.

Note 1. Do not write to these bits during communication. Changing a value during communication may cause a transmission or reception failure or an AL error.

Note 2. To change the SDAO and SCLO bits, set the SOWP bit to 0 at the same timing to set the SDAO and SCLO bits to 0.

SDAO Bit (SDA Output Control/Monitor) and SCLO Bit (SCL Output Control/Monitor)

These bits are used to directly control the SDAn and SCLn signals output from the RIIC.

When writing to these bits, also write 0 to the SOWP bit. The result of setting these bits is input to the RIIC via the input buffer. When slave mode is selected, a START condition may be detected and the bus may be released depending on the bit settings.

Do not rewrite these bits during a START condition, STOP condition, repeated START condition, or during transmission or reception. Operation after rewriting under the above conditions is not guaranteed.

When reading these bits, the state of signals output from the RIIC can be read.

CLO Bit (Extra SCL Clock Cycle Output)

This bit is used to output an extra SCL clock cycle for debugging or error processing.

Normally, set the bit to 0. Setting the bit to 1 in a normal communication state causes a communication error. For details on this function, see **Section 25.13.2, Extra SCL Clock Cycle Output Function**.

IICRST Bit (I^2C Bus Interface Internal Reset)

This bit is used to reset the internal states of the RIIC. Setting this bit to 1 initiates an RIIC reset or internal reset.

Whether an RIIC reset or internal reset is initiated is determined according to the combination with the ICE bit. **Table 25.7** lists the resets of the RIIC.

The RIIC reset resets all registers including the RIICnCR2.BBSY flag (except ICE and IICRST) and internal states of the RIIC, and the internal reset resets the bit counter (RIICnMR1.BC[2:0] bits), the I^2C bus shift register (RIICnDRS), and the I^2C bus status registers (RIICnSR1 and RIICnSR2) as well as the internal states of the RIIC. For the reset conditions for each register, see **Section 25.15, Reset Function of RIIC**.

An internal reset initiated with the IICRST bit set to 1 during operation (with the ICE bit set to 1) resets the internal states of the RIIC without initializing the port settings and the control and setting registers of the RIIC when the bus or RIIC hangs up due to a communication error.

If the RIIC hangs up in a low level output state, resetting the internal states cancels the low level output state and releases the bus with the RIICnSCL pin and RIICnSDA pin at a high impedance.

Remark If an internal reset is initiated using the IICRST bit for a bus hang-up occurred during communication with the master device in slave mode, the states may become different between the slave device and the master device (due to the difference in the bit counter information). For this reason, do not initiate an internal reset in slave mode, but initiate restoration processing from the master device. If an internal reset is necessary because the RIIC hangs up with the SCL line in a low level output state in slave mode, initiate an internal reset and then issue a restart condition from the master device or resume communication from the start condition issuance after issuing a stop condition. If communication is restarted by initiating a reset solely in the slave device without issuing a start condition or restart condition from the master device, synchronization will be lost because the master and slave devices operate asynchronously.

Table 25.7 RIIC Resets

IICRST	ICE	State	Specifications
1	0	RIIC reset	Resets all registers (except ICE and IICRST) and internal states of the RIIC.
	1	Internal reset	Reset the RIICnMR1.BC[2:0] bits, and the RIICnSR1, RIICnSR2, RIICnDRS registers and the internal states of the RIIC.

ICE Bit (I^2C Bus Interface Enable)

The ICE bit selects driving or non-driving of the RIICnSCL and RIICnSDA pins. Moreover, this bit can perform two types of reset in combination with the IICRST bit. For the types of reset, see **Table 25.7**.

Set the ICE bit to 1 when using RIIC. Setting the ICE bit to 1 selects driving of the RIICnSCL and RIICnSDA pins. Set the ICE bit to 0 when RIIC is not to be used. Clearing the ICE bit to 0 disables output from the RIICnSCL and RIICnSDA pins.

25.3.2 I²C Bus Control Register 2 (RIICnCR2)

Access Size: RIICnCR2 is a 8-bit readable/writable register.

Address(es): RIICnCR2: <RIICn_base> + H'0001

Initial Value: H'00 This register is initialized by any reset.

Bit	7	6	5	4	3	2	1	0
	BBSY	MST	TRS	—	SP	RS	ST	—
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R/W*1	R/W*1	R	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7	BBSY	0	R	Bus Busy Detection Flag 0: The I²C bus is released (bus free state). 1: The I²C bus is occupied (bus busy state or in the bus free state).
6	MST	0	R/W*1	Master/Slave Mode 0: Slave mode 1: Master mode
5	TRS	0	R/W*1	Transmit/Receive Mode 0: Receive mode 1: Transmit mode
4	—	0	R	Reserved This bit is read as 0. The write value should be 0.
3	SP	0	R/W	Stop Condition Issuance Request 0: Does not request to issue a stop condition. 1: Requests to issue a stop condition.
2	RS	0	R/W	Restart Condition Issuance Request 0: Does not request to issue a restart condition. 1: Requests to issue a restart condition.
1	ST	0	R/W	Start Condition Issuance Request 0: Does not request to issue a start condition. 1: Requests to issue a start condition.
0	—	0	R	Reserved This bit is read as 0. The write value should be 0.

Note 1. Set RIICnMR1.MTWP bit before writing MST and TRS bits.

ST Bit (Start Condition Issuance Request)

This bit is used to request transition to master mode and issuance of a start condition.

When this bit is set to 1 to request to issue a start condition, a start condition is issued when the BBSY flag is set to 0 (bus free).

For details on the start condition issuance, see **Section 25.12, Start Condition/Restart Condition/Stop Condition Issuing Function**.

[Setting condition]

When 1 is written to the ST bit

[Clearing conditions]

- When 0 is written to the ST bit
- When a start condition has been issued

- When the RIICnSR2.AL (arbitration-lost) flag is set to 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

Remark Set the ST bit to 1 (start condition issuance request) when the BBSY flag is set to 0 (bus free). Note that arbitration may be lost if the ST bit is set to 1 (start condition issuance request) when the BBSY flag is set to 1 (bus busy).

RS Bit (Restart Condition Issuance Request)

This bit is used to request that a restart condition be issued in master mode.

When this bit is set to 1 to request to issue a restart condition, a restart condition is issued when the BBSY flag is set to 1 (bus busy) and the MST bit is set to 1 (master mode).

For details on the restart condition issuance, see **Section 25.12, Start Condition/Restart Condition/Stop Condition Issuing Function**.

[Setting condition]

When 1 is written to the RS bit with the RIICnCR2.BBSY flag set to 1

[Clearing conditions]

- When 0 is written to the RS bit
- When a restart condition has been issued or a start condition is detected
- When a stop condition is detected
- When the RIICnCR2.AL (arbitration-lost) flag is set to 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

Remark 1. Do not set the RS bit to 1 while issuing a stop condition.

Remark 2. It is commended to issue a restart condition in master transmit mode. If the RS bit is set to 1 (restart condition issuance request) in mode other than master mode, the restart condition is not issued in this mode but the restart condition issuance request bit remains set. If the operating mode changes to master mode with the bit not being cleared, the restart condition may be issued.

SP Bit (Stop Condition Issuance Request)

This bit is used to request that a stop condition be issued in master mode.

When this bit is set to 1 to request to issue a stop condition, a stop condition is issued when the BBSY flag is set to 1 (bus busy) and the MST bit is set to 1 (master mode).

For details on the stop condition issuance, see **Section 25.12, Start Condition/Restart Condition/Stop Condition Issuing Function**.

[Setting condition]

When 1 is written to the SP bit with both the RIICnCR2.BBSY flag and the RIICnCR2.MST bit set to 1

[Clearing conditions]

- When 0 is written to the SP bit
- When a stop condition has been issued or a stop condition is detected
- When the RIICnSR2.AL (arbitration-lost) flag is set to 1
- When a start condition and a restart condition are detected

- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

Remark 1. Wring to the SP bit is not possible while the setting of the BBSY flag is 0 (bus free).

Remark 2. Do not set the SP bit to 1 while a restart condition is being issued.

TRS Bit (Transmit/Receive Mode)

This bit indicates transmit or receive mode.

The RIIC is in receive mode when the TRS bit is set to 0 and is in transmit mode when the bit is set to 1. Combination of this bit and the MST bit indicates the operating mode of the RIIC.

The value of the TRS bit is automatically changed to 1 for transmit mode or 0 for receive mode by issuing or detection of a start condition and setting or clearing of the R/W# bit. Although writing to the TRS bit is possible when the MTWP bit in RIICnMR1 is set to 1, writing to this bit is not necessary during normal usage.

[Setting conditions]

- When a start condition is issued normally according to the start condition issuance request (when a start condition is detected with the ST bit set to 1)
- When the R/W# bit added to the slave address is set to 0 in master mode
- When the address received in slave mode matches the address enabled in RIICnSER, with the R/W# bit set to 1
- When 1 is written to the TRS bit with the MTWP bit in RIICnMR1 set to 1

[Clearing conditions]

- When a stop condition is detected
- The RIICnSR2.AL (arbitration-lost) flag being set to 1
- In master mode, reception of a slave address to which an R/W# bit with the value 1 is appended
- In slave mode, a match between the received address and the address enabled in RIICnSER when the value of the received R/W# bit is 0 (including cases where the received address is the general call address)
- In slave transmit mode, a restart condition is detected (a restart condition is detected with RIICnCR2.BBSY = 1 and RIICnCR2.MST = 0)
- When 0 is written to the TRS bit with the MTWP bit in RIICnMR1 set to 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

MST Bit (Master/Slave Mode)

This bit indicates master or slave mode.

The RIIC is in slave mode when the MST bit is set to 0 and is in master mode when the bit is set to 1. Combination of this bit and the TRS bit indicates the operating mode of the RIIC.

The value of the MST bit is automatically changed to 1 for master mode or 0 for slave mode by issuing of a start condition and issuing of detection, or of a stop condition, etc. Although writing to the MST bit is possible when the RIICnMR1.MTWP bit is set to 1, writing to this bit is not necessary during normal usage.

[Setting conditions]

- When a start condition is issued normally according to the start condition issuance request (when a start condition is detected with the ST bit set to 1)
- When 1 is written to the MST bit with the RIICnMR1.MTWP bit set to 1

[Clearing conditions]

- When a stop condition is detected
- When the RIICnSR2.AL (arbitration-lost) flag is set to 1
- When 0 is written to the MST bit with the RIICnMR1.MTWP bit to 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

BBSY Flag (Bus Busy Detection)

The BBSY flag indicates whether the I^2C bus is occupied (bus busy) or released (bus free).

This bit is set to 1 when the SDA line changes from high to low under the condition of SCL = high, assuming that a start condition has been issued.

When the SDA line changes from low to high under the condition of SCL = high, this bit is cleared to 0 after the bus free time (specified in RIICnBRL) start condition is not detected, assuming that a stop condition has been issued.

[Setting condition]

- When a start condition is detected

[Clearing conditions]

- When the bus free time (specified in RIICnBRL) start condition is not detected after detecting a stop condition
- When 1 is written to the RIICnCR1.IICRST bit with the RIICnCR1.ICE bit set to 0 (RIIC reset)

25.3.3 I²C Bus Mode Register 1 (RIICnMR1)

Access Size: RIICnMR1 is a 8-bit readable/writable register.

Address(es): RIICnMR1: <RIICn_base> + H'0002

Initial Value: H'08 This register is initialized by any reset.

Bit	7	6	5	4	3	2	1	0
	MTWP	CKS[2:0]			BCWP	BC[2:0]		
Initial Value	0	0	0	0	1	0	0	0
R/W	R/W	R/W	R/W	R/W	W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MTWP	0	R/W	MST/TRS Write Protect 0: Disables writing to the MST and TRS bits in ICCR2. 1: Enables writing to the MST and TRS bits in ICCR2.
6 to 4	CKS[2:0]	000	R/W	Internal Reference Clock (IICϕ) Selection b6 b4 0 0 0: IICϕ = P0ϕ/1 0 0 1: IICϕ = P0ϕ/2 0 1 0: IICϕ = P0ϕ/4 0 1 1: IICϕ = P0ϕ/8 1 0 0: IICϕ = P0ϕ/16 1 0 1: IICϕ = P0ϕ/32 1 1 0: IICϕ = P0ϕ/64 1 1 1: IICϕ = P0ϕ/128
3	BCWP*1	1	W	BC Write Protect 0: Enables a value to be written in the BC[2:0] bits. (This bit is read as 1.)
2 to 0	BC[2:0]	000	R/W	Bit Counter b2 b0 0 0 0: 9 bits 0 0 1: 2 bits 0 1 0: 3 bits 0 1 1: 4 bits 1 0 0: 5 bits 1 0 1: 6 bits 1 1 0: 7 bits 1 1 1: 8 bits

Note 1. When rewriting the BC[2:0] bits, write 0 to the BCWP bit simultaneously.

BC[2:0] Bits (Bit Counter)

These bits function as a counter that indicates the number of bits remaining to be transferred at the detection of a rising edge on the SCL line. Although these bits are writable and readable, it is not necessary to access these bits under normal conditions.

To write to these bits, specify the number of bits to be transferred plus one (data is transferred with an additional acknowledge bit) between transferred frames when the SCL line is at a low level.

The values of the BC[2:0] bits return to 000b at the end of a data transfer including the acknowledge bit or when a start condition including a restart condition is detected.

25.3.4 I²C Bus Mode Register 2 (RIICnMR2)

Access Size: RIICnMR2 is a 8-bit readable/writable register.

Address(es): RIICnMR2: <RIICn_base> + H'0003

Initial Value: H'06 This register is initialized by any reset.

Bit	7	6	5	4	3	2	1	0
	DLCS	SDDL[2:0]			—	TMOH	TMOL	TMOS
Initial Value	0	0	0	0	0	1	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	DLCS	0	R/W	SDA Output Delay Clock Source Selection 0: The internal reference clock (IICϕ) is selected as the clock source of the SDA output delay counter. 1: The internal reference clock divided by 2 (IICϕ/2) is selected as the clock source of the SDA output delay counter.*1
6 to 4	SDDL[2:0]	000	R/W	SDA Output Delay Counter • When RIICnMR2.DLCS = 0 (IICϕ) b6 b4 0 0: No output delay 0 1: 1 IICϕ cycle 1 0: 2 IICϕ cycles 1 1: 3 IICϕ cycles 0 0: 4 IICϕ cycles 0 1: 5 IICϕ cycles 1 0: 6 IICϕ cycles 1 1: 7 IICϕ cycles • When RIICnMR2.DLCS = 1 (IICϕ/2) b6 b4 0 0: No output delay 0 1: 1 or 2 IICϕ cycles 1 0: 3 or 4 IICϕ cycles 1 1: 5 or 6 IICϕ cycles 0 0: 7 or 8 IICϕ cycles 0 1: 9 or 10 IICϕ cycles 1 0: 11 or 12 IICϕ cycles 1 1: 13 or 14 IICϕ cycles
3	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
2	TMOH	1	R/W	Timeout H Count Control 0: Count is disabled while the SCL line is at a high level. 1: Count is enabled while the SCL line is at a high level.
1	TMOL	1	R/W	Timeout L Count Control 0: Count is disabled while the SCL line is at a low level. 1: Count is enabled while the SCL line is at a low level.
0	TMOS	0	R/W	Timeout Detection Time Selection 0: Long mode is selected. 1: Short mode is selected.

Note 1. The setting DLCS = 1 (IICϕ/2) only becomes valid when SCL is at the low level. When SCL is at the high level, the setting DLCS = 1 becomes invalid and the clock source becomes the internal reference clock (IICϕ).

TMOS Bit (Timeout Detection Time Selection)

This bit is used to select long mode or short mode for the timeout detection time when the timeout function is enabled (RIICnFER.TMOE bit = 1). When this bit is set to 0, long mode is selected. When this bit is set to 1, short mode is selected. In long mode, the timeout detection internal counter functions as a 16 bit-counter. In short mode, the counter functions as a 14 bit-counter. While the SCL line is in the state that enables this counter as specified by bits TMOH and TMOL, the counter counts up in synchronization with the internal reference clock (IICϕ) as a count source.

For details on the timeout function, see **Section 25.13.1, Timeout Function**.

TMOL Bit (Timeout L Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCL line is held low when the timeout function is enabled (RIICnFER.TMOE bit = 1).

TMOH Bit (Timeout H Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCL line is held high when the timeout function is enabled (RIICnFER.TMOE bit = 1).

SDDL[2:0] Bits (SDA Output Delay Setup Counter)

The SDA output can be delayed by the SDDL[2:0] setting. This counter works with the clock source selected by the DLCS bit. The setting of this function can be used for all types of SDA output, including the transmission of the acknowledge bit.

For details on this function, see **Section 25.7, Facility for Delaying SDA Output**.

Remark Set the SDA output delay time to meet the I^2C bus standard (within the data enable time/acknowledge enable time*1) or the SMBus standard (within the data hold time: 300 [ns] or more, and SCL-clock low-level period - the data setup time: 250 [ns]). Note that, if a value outside the standard is set, communication with communication devices may malfunction or it may seemingly become a start condition or stop condition depending on the bus state.

Note 1. Data enable time/acknowledge enable time
- 3,450 [ns] (0 to 100 [kbps]: standard mode (Sm))
- 900 [ns] (0 to 400 [kbps]: fast mode (Fm))
- 450 [ns] (0 to 1 [Mbps]: fast mode plus (Fm+))

25.3.5 I²C Bus Mode Register 3 (RIICnMR3)

Access Size: RIICnMR3 is a 8-bit readable/writable register.

Address(es): RIICnMR3: <RIICn_base> + H'0004

Initial Value: H'00 This register is initialized by any reset.

Bit	7	6	5	4	3	2	1	0
	SMBS	WAIT	RDRFS	ACKWP	ACKBT	ACKBR	NF[1:0]	
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	SMBS	0	R/W	SMBus/I²C Bus Selection 0: I²C bus is selected. 1: SMBus is selected.
6	WAIT*2	0	R/W	WAIT 0: No WAIT (The period between ninth clock cycle and first clock cycle is not held low.) 1: WAIT (The period between ninth clock cycle and first clock cycle is held low.) Low-hold is released by reading RIICnDRR.
5	RDRFS*2	0	R/W	RDRF Flag Set Timing Selection 0: The RDRF flag is set at the rising edge of the ninth SCL clock cycle. (The SCL line is not held low at the falling edge of the eighth clock cycle.) 1: The RDRF flag is set at the rising edge of the eighth SCL clock cycle. (The SCL line is held low at the falling edge of the eighth clock cycle.) Low-hold is released by writing a value to the ACKBT bit.
4	ACKWP*1	0	R/W	ACKBT Write Protect 0: Modification of the ACKBT bit is disabled. 1: Modification of the ACKBT bit is enabled.
3	ACKBT*1	0	R/W	Transmit Acknowledge 0: A 0 is sent as the acknowledge bit (ACK transmission). 1: A 1 is sent as the acknowledge bit (NACK transmission).
2	ACKBR	0	R	Receive Acknowledge 0: A 0 is received as the acknowledge bit (ACK reception). 1: A 1 is received as the acknowledge bit (NACK reception).
1, 0	NF[1:0]	00	R/W	Noise Filter Stage Selection b1 b0 0 0: Noise of up to one IICϕ cycle is filtered out (single-stage filter). 0 1: Noise of up to two IICϕ cycles is filtered out (2-stage filter). 1 0: Noise of up to three IICϕ cycles is filtered out (3-stage filter). 1 1: Noise of up to four IICϕ cycles is filtered out (4-stage filter).

Note 1. If it is attempted to write 1 to both ACKWP and ACKBT bits, the ACKBT bit cannot be set to 1.

Note 2. The WAIT and RDRFS bits are valid only in receive mode (invalid in transmit mode).

NF[1:0] Bits (Noise Filter Stage Selection)

These bits are used to select the width of noise that can be removed from the signals input to RIICnSCL or RIICnSDA pin.

For details on the digital noise filter function, refer to **Section 25.8, Digital Noise-Filter Circuits**.

Remark Set the noise range to be filtered out by the noise filter within a range less than the SCLn line high-level period or low-level period. If the noise range is set to a value of (SCL clock width: high-level period or low-level period, whichever is shorter) − [1.5 internal reference clock (IICϕ) cycles + analog noise filter: 120 ns (reference values)] or more, the SCL clock is regarded as noise by the noise filter function of the RIIC, which may prevent the RIIC from operating normally.

ACKBR Bit (Receive knowledge)

This bit is used to store the acknowledge bit information received from the receive device in transmit mode.

[Setting condition]

- When 1 is received as the acknowledge bit with the RIICnCR2.TRS bit set to 1

[Clearing conditions]

- When 0 is received as the acknowledge bit with the RIICnCR2.TRS bit set to 1
- When 1 is written to the RIICnCR1.IICRST bit while the RIICnCR1.ICE bit is 0 (RIIC reset)

ACKBT Bit (Transmit Acknowledge)

This bit is used to set the bit to be sent at the acknowledge timing in receive mode.

[Setting condition]

- When 1 is written to this bit with the ACKWP bit set to 1

[Clearing conditions]

- When 0 is written to this bit with the ACKWP bit set to 1
- When stop condition issuance is detected
- When 1 is written to the RIICnCR1.IICRST bit while the RIICnCR1.ICE bit is 0 (RIIC reset)

Remark The ACKBT bit must be written to while the ACKWP bit is 1. If the ACKBT bit is written to with the ACKWP bit cleared to 0, writing to the ACKBT bit is disabled.

ACKWP Bit (ACKBT Write Protect)

This bit is used to control the modification of the ACKBT bit.

RDRFS Bit (RDRF Flag Set Timing Selection)

This bit is used to select the RDRF flag set timing in receive mode and also to select whether to hold the SCL line low at the falling edge of the eighth SCL clock cycle.

When the RDRFS bit is 0, the SCL line is not held low at the falling edge of the eighth SCL clock cycle, and the RDRF flag is set to 1 at the rising edge of the ninth SCL clock cycle.

When the RDRFS bit is 1, the RDRF flag is set to 1 at the rising edge of the eighth SCL clock cycle and the SCL line is held low at the falling edge of the eighth SCL clock cycle. The low-hold of the SCL line is released by writing a value to the ACKBT bit.

After data is received with this setting, the SCL line is automatically held low before the acknowledge bit is sent. This enables processing to send ACK (ACKBT = 0) or NACK (ACKBT = 1) according to receive data.

WAIT Bit (WAIT)

This bit is used to control whether to hold the period between the ninth SCL clock cycle and the first SCL clock cycle low until the receive data buffer (RIICnDRR) is completely read each time single-byte data is received in receive mode. When the WAIT bit is 0, the receive operation is continued without holding the period between the ninth and the first SCL clock cycle low. When both the RDRFS and WAIT bits are 0, continuous receive operation is enabled with the double buffer.

When the WAIT bit is 1, the SCL line is held low from the falling edge of the ninth clock cycle until the RIICnDRR value is read each time single-byte data is received. This enables receive operation in byte units.

Remark When the value of the WAIT bit is to be read, be sure to read the RIICnDRR beforehand.

SMBS Bit (SMBus/I^2C Selection)

Setting this bit to 1 selects the SMBus and enables the RIICnSER.HOAE bit.

25.3.6 I²C Bus Function Enable Register (RIICnFER)

Access Size: RIICnFER is a 8-bit readable/writable register.

Address(es): RIICnFER: <RIICn_base> + H'0005

Initial Value: H'72 This register is initialized by any reset.

Bit	7	6	5	4	3	2	1	0
	FMPE	SCLE	NFE	NACKE	SALE	NALE	MALE	TMOE
Initial Value	0	1	1	1	0	0	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	FMPE*1	0	R/W	Fast-mode Plus Enable 0: No Fm+ slope control circuit is used for the SCLn pin and SDAn pin 1: An Fm+ slope control circuit is used for the SCLn pin and SDAn pin
6	SCLE	1	R/W	SCL Synchronous Circuit Enable 0: No SCL synchronous circuit is used. 1: An SCL synchronous circuit is used.
5	NFE	1	R/W	Digital Noise Filter Circuit Enable 0: No digital noise filter circuit is used. 1: A digital noise filter circuit is used.
4	NACKE	1	R/W	NACK Reception Transfer Suspension Enable 0: Transfer operation is not suspended during NACK reception (transfer suspension disabled). 1: Transfer operation is suspended during NACK reception (transfer suspension enabled).
3	SALE	0	R/W	Slave Arbitration-Lost Detection Enable 0: Slave arbitration-lost detection is disabled. 1: Slave arbitration-lost detection is enabled.
2	NALE	0	R/W	NACK Transmission Arbitration-Lost Detection Enable 0: NACK transmission arbitration-lost detection is disabled. 1: NACK transmission arbitration-lost detection is enabled.
1	MALE	1	R/W	Master Arbitration-Lost Detection Enable 0: Master arbitration-lost detection is disabled. (Disables the arbitration-lost detection function and does not clear the RIICnCR2.MST and RIICnCR2.TRS bits automatically when arbitration is lost.) 1: Master arbitration-lost detection is enabled. (Enables the arbitration-lost detection function and clears the RIICnCR2.MST and RIICnCR2.TRS bits automatically when arbitration is lost.)
0	TMOE	0	R/W	Timeout Function Enable 0: The timeout function is disabled. 1: The timeout function is enabled.

Note 1. The fast-mode plus enable bit (FMPE) is only supported by RIIC0. In RIIC2, bit 7 is reserved.

TMOE Bit (Timeout Function Enable)

This bit is used to enable or disable the timeout function.

For details on the timeout function, see **Section 25.13.1, Timeout Function**.

MALE Bit (Master Arbitration-Lost Detection Enable)

This bit is used to specify whether to use the arbitration-lost detection function in master mode. Normally, set this bit to 1.

NALE Bit (NACK Transmission Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when ACK is detected during transmission of NACK in receive mode (such as when slaves with the same address exist on the bus or when two or more masters select the same slave device simultaneously with different number of receive bytes).

SALE Bit (Slave Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when a value different from the value being transmitted is detected on the bus in slave transmit mode (such as when slaves with the same address exist on the bus or when a mismatch with the transmit data occurs due to noise).

NACKE Bit (NACK Reception Transfer Suspension Enable)

This bit is used to specify whether to continue or discontinue the transfer operation when NACK is received in transmit mode. Normally, set this bit to 1.

When NACK is received with the NACKE bit set to 1, the next transfer operation is suspended.

When the NACKE bit is 0, the next transfer operation is continued regardless of the received acknowledge content. For details on the NACK reception transfer suspension function, refer to **Section 25.10.2, NACK Reception Transfer Suspension Function**.

SCLE Bit (SCL Synchronous Circuit Enable)

This bit is used to specify whether to synchronize the SCL clock with the SCL input clock. Normally, set this bit to 1.

When the SCLE bit is cleared to 0 (SCL synchronous circuit not used), the RIIC does not synchronize the SCL clock with the SCL input clock. In this setting, the RIIC outputs the SCL clock with the transfer rate set in RIICnBRH and RIICnBRL regardless of the SCL line state.

For this reason, if the bus load of the I^2C bus line is much larger than the specification value or if the SCL clock output overlaps in multiple masters, the short-cycle SCL clock that does not meet the specification may be output.

When no SCL synchronous circuit is used, it also affects the issuance of a start condition, restart condition, and stop condition, and the continuous output of extra SCL clock cycles.

This bit must not be cleared to 0 except for checking the output of the transfer rate.

FMPE Bit (Fast-mode Plus Enable)

This bit is used to specify whether to use a slope control circuit for Fast-mode Plus [Fm+].

When this bit is set to 1, a slope control circuit conforming to the Fast-mode Plus [Fm+] slope control standard (tof) of the I^2C bus is selected. When this bit is cleared to 0, a slope control circuit conforming to the Standard-mode [Sm] and Fast- mode [Fm] slope control standard (tof) of the I^2C bus is selected.

Set this bit to 1 when using the transmission rate within a range up to 1 Mbps (Fast-mode Plus[Fm+]) of the I^2C bus standard. Clear this bit to 0 when using the transmission rate at other rates (up to 100 kbps[Sm], up to 400 kbps[Fm]) or for SMBus (10 to 100 kbps).

25.3.7 I²C Bus Status Enable Register (RIICnSER)

Access Size: RIICnSER is a 8-bit readable/writable register.

Address(es): RIICnSER: <RIICn_base> + H'0006

Initial Value: H'09 This register is initialized by any reset.

Bit	7	6	5	4	3	2	1	0
	HOAE	—	DIDE	—	GCE	SAR2E	SAR1E	SAR0E
Initial Value	0	0	0	0	1	0	0	1
R/W	R/W	R	R/W	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	HOAE	0	R/W	Host Address Enable 0: Host address detection is disabled. 1: Host address detection is enabled.
6	—	0	R	Reserved This bit is read as 0. The write value should be 0.
5	DIDE	0	R/W	Device-ID Address Detection Enable 0: Device-ID address detection is disabled. 1: Device-ID address detection is enabled.
4	—	0	R	Reserved This bit is read as 0. The write value should be 0.
3	GCE	1	R/W	General Call Address Enable 0: General call address detection is disabled. 1: General call address detection is enabled.
2	SAR2E	0	R/W	Slave Address Register 2 Enable 0: Slave address in RIICnSARL2 and RIICnSARU2 are disabled. 1: Slave address in RIICnSARL2 and RIICnSARU2 are enabled.
1	SAR1E	0	R/W	Slave Address Register 1 Enable 0: Slave address in RIICnSARL1 and RIICnSARU1 are disabled. 1: Slave address in RIICnSARL1 and RIICnSARU1 are enabled.
0	SAR0E	1	R/W	Slave Address Register 0 Enable 0: Slave address in RIICnSARL0 and RIICnSARU0 are disabled. 1: Slave address in RIICnSARL0 and RIICnSARU0 are enabled.

SARyE Bit (Slave Address Register y Enable) (y = 0 to 2)

This bits are used to enable or disable the received slave address and the slave address set in RIICnSAR(L,U)y.

When these bits are set to 1, the slave address set in RIICnSAR(L,U)y are enabled and are compared with the received slave address. When these bits are cleared to 0, the slave address set in RIICnSAR(L,U)y are disabled and are ignored even if it match the received slave address.

GCE Bit (General Call Address Enable)

This bit is used to specify whether to ignore the general call address (0000_000b + 0 [W]: All 0) when it is received.

When this bit is set to 1, if the received slave address matches the general call address, the RIIC recognizes the received slave address as the general call address independently of the slave addresses set in RIICnSAR(L,U)y (y = 0 to 2) and performs data receive operation.

When this bit is cleared to 0, the received slave address is ignored even if it matches the general call address.

DIDE Bit (Device-ID Address Detection Enable)

This bit is used to specify whether to recognize and execute the Device-ID address when a device ID (1111_100b) is received in the first frame after a start condition or restart condition is detected.

When this bit is set to 1, if the received first frame matches the device ID, the RIIC recognizes that the Device-ID address has been received. When the following R/W# bit is 0 [W], the RIIC recognizes the second and the following frames as slave addresses and continues the receive operation.

When this bit is cleared to 0, the RIIC ignores the received first frame even if it matches the device ID address and recognizes the first frame as a normal slave address.

For details on the device-ID address detection, see **Section 25.9.3, Device-ID Address Detection**.

HOAE Bit (Host Address Enable)

This bit is used to specify whether to ignore received host address (0001_000b) when the RIICnMR3.SMBS bit is 1.

When this bit is set to 1 while the RIICnMR3.SMBS bit is 1, if the received slave address matches the host address, the RIIC recognizes the received slave address as the host address independently of the slave addresses set in RIICnSAR(L,U)y (y = 0 to 2) and performs the receive operation.

When the RIICnMR3.SMBS bit or the HOAE bit is cleared to 0, the received slave address is ignored even if it matches the host address.

25.3.8 I²C Bus Interrupt Enable Register (RIICnIER)

Access Size: RIICnIER is a 8-bit readable/writable register.

Address(es): RIICnIER: <RIICn_base> + H'0007

Initial Value: H'00 This register is initialized by any reset.

Bit	7	6	5	4	3	2	1	0
	TIE	TEIE	RIE	NAKIE	SPIE	STIE	ALIE	TMOIE
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TIE	0	R/W	Transmit Data Empty Interrupt Enable 0: Transmit data empty interrupt request (INTRIICTI) is disabled. 1: Transmit data empty interrupt request (INTRIICTI) is enabled.
6	TEIE	0	R/W	Transmit End Interrupt Enable 0: Transmit end interrupt request (INTRIICTEI) is disabled. 1: Transmit end interrupt request (INTRIICTEI) is enabled.
5	RIE	0	R/W	Receive Data Full Interrupt Enable 0: Receive data full interrupt request (INTRIICRI) is disabled. 1: Receive data full interrupt request (INTRIICRI) is enabled.
4	NAKIE	0	R/W	NACK Reception Interrupt Enable 0: NACK reception interrupt request (INTRIICNAKI) is disabled. 1: NACK reception interrupt request (INTRIICNAKI) is enabled.
3	SPIE	0	R/W	Stop Condition Detection Interrupt Enable 0: Stop condition detection interrupt request (INTRIICSPI) is disabled. 1: Stop condition detection interrupt request (INTRIICSPI) is enabled.
2	STIE	0	R/W	Start Condition Detection Interrupt Enable 0: Start condition detection interrupt request (IINTRIICSTI) is disabled. 1: Start condition detection interrupt request (INTRIICSTI) is enabled.
1	ALIE	0	R/W	Arbitration-Lost Interrupt Enable 0: Arbitration-lost interrupt request (INTRIICALI) is disabled. 1: Arbitration-lost interrupt request (INTRIICALI) is enabled.
0	TMOIE	0	R/W	Timeout Interrupt Enable 0: Timeout interrupt request (INTRIICTMOI) is disabled. 1: Timeout interrupt request (INTRIICTMOI) is enabled.

TMOIE Bit (Timeout Interrupt Enable)

This bit is used to enable or disable timeout interrupt requests (INTRIICTMOI) when the RIICnSR2.TMOF flag is set to 1. An INTRIICTMOI interrupt request is canceled by clearing the TMOF flag or the TMOIE bit to 0.

ALIE Bit (Arbitration-Lost Interrupt Enable)

This bit is used to enable or disable arbitration-lost interrupt requests (INTRIICALI) when the RIICnSR2.AL flag is set to 1. An INTRIICALI interrupt request is canceled by clearing the AL flag or the ALIE bit to 0.

STIE Bit (Start Condition Detection Interrupt Enable)

This bit is used to enable or disable start condition detection interrupt requests (INTRIICSTI) when the RIICnSR2.START flag is set to 1. An INTRIICSTI interrupt request is canceled by clearing the START flag or the STIE bit to 0.

SPIE Bit (Stop Condition Detection Interrupt Enable)

This bit is used to enable or disable stop condition detection interrupt requests (INTRIICSPI) when the RIICnSR2.STOP flag is set to 1. An INTRIICSPI interrupt request is canceled by clearing the STOP flag or the SPIE bit to 0.

NAKIE Bit (NACK Reception Interrupt Enable)

This bit is used to enable or disable NACK reception interrupt requests (INTRIICNAKI) when the RIICnSR2.NACKF flag is set to 1. An INTRIICNAKI interrupt request is canceled by clearing the NACKF flag or the NAKIE bit to 0.

RIE Bit (Receive Data Full Interrupt Enable)

This bit is used to enable or disable receive data full interrupt requests (INTRIICRI) when the RIICnSR2.RDRF flag in ICSR2 is set to 1. An INTRIICRI interrupt request is canceled by clearing the RDRF flag or the RIE bit to 0.

TEIE Bit (Transmit End Interrupt Enable)

This bit is used to enable or disable transmit end interrupts (INTRIICTEI) when the RIICnSR2.TEND flag is set to 1. An INTRIICTEI interrupt request is canceled by clearing the TEND flag or the TEIE bit to 0.

TIE Bit (Transmit Data Empty Interrupt Enable)

This bit is used to enable or disable transmit data empty interrupts (INTRIICTI) when the RIICnSR2.TDRE flag is set to 1.

25.3.9 I²C Bus Status Register 1 (RIICnSR1)

Access Size: RIICnSR1 is a 8-bit readable/writable register

Address(es): RIICnSR1: <RIICn_base> + H'0008

Initial Value: H'00 This register is initialized by any reset.

Bit	7	6	5	4	3	2	1	0
	HOA	—	DID	—	GCA	AAS2	AAS1	AAS0
Initial Value	0	0	0	0	0	0	0	0
R/W	R(/W)*1	R	R(/W)*1	R	R(/W)*1	R(/W)*1	R(/W)*1	R(/W)*1

Bit	Bit Name	Initial Value	R/W	Description
7	HOA	0	R(/W)*1	Host Address Detection Flag 0: Host address is not detected. 1: Host address is detected.
6	—	0	R	Reserved This bit is read as 0. The write value should be 0.
5	DID	0	R(/W)*1	Device-ID Address Detection Flag 0: Device-ID command is not detected. 1: Device-ID command is detected.
4	—	0	R	Reserved This bit is read as 0. The write value should be 0.
3	GCA	0	R(/W)*1	General Call Address Detection Flag 0: General call address is not detected. 1: General call address is detected.
2	AAS2	0	R(/W)*1	Slave Address 2 Detection Flag 0: Slave address 2 is not detected. 1: Slave address 2 is detected.
1	AAS1	0	R(/W)*1	Slave Address 1 Detection Flag 0: Slave address 1 is not detected. 1: Slave address 1 is detected.
0	AAS0	0	R(/W)*1	Slave Address 0 Detection Flag 0: Slave address 0 is not detected. 1: Slave address 0 is detected.

Note 1. Only 0 can be written to this bit.

AASy Flag (Slave Address y Detection) (y = 0 to 2)

[Setting conditions]

<For 7-bit address format: RIICnSARUy.FS = 0>

- When the received slave address matches the RIICnSARLy.SVA[7:1] value with the RIICnSER.SARyE bit set to 1 (slave address y detection enabled)

  This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

<For 10-bit address format: RIICnSARUy.FS = 1>

- When the received slave address matches a value of (1111_0b + RIICnSARUy.SVA[9:8]) and the following address matches the RIICnSARLy.SVA[7:0] value with the RIICnSER.SARyE bit set to 1 (slave address y detection enabled) This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the AASy bit after reading AASy = 1
- When a stop condition is detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

<For 7-bit address format: RIICnSARUy.FS = 0>

- When the received slave address does not match the RIICnSARLy.SVA[7:1] value with the RIICnSER.SARyE bit set to 1 (slave address y detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.

<For 10-bit address format: RIICnSARUy.FS = 1>

- When the received slave address does not match a value of (1111_0b + RIICnSARUy.SVA[9:8]) with the RIICnSER.SARyE bit set to 1 (slave address y detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When the received slave address matches a value of (1111_0b + RIICnSARUy.SVA[9:8]) and the following address does not match the RIICnSARLy.SVA[7:0] value with the RIICnSER.SARyE bit set to 1 (slave address y detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.

GCA Flag (General Call Address Detection)

[Setting condition]

- When the received slave address matches the general call address (0000_000b + 0 [W]) with the RIICnSER.GCE bit set to 1 (general call address detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the GCA bit after reading GCA = 1
- When a stop condition is detected
- When the received slave address does not match the general call address (0000_000b + 0 [W]) with the RIICnSER.GCE bit set to 1 (general call address detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

DID Flag (Device-ID Address Detection)

[Setting condition]

- When the first frame received immediately after a start condition or restart condition is detected matches a value of (device ID (1111_100b) + 0 [W]) with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled) This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.
- When a restart condition is detected after a match with the device ID address and the device ID address (1111_100b) + 1 [R] has matched with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled)
  This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the DID bit after reading DID = 1
- When a stop condition is detected

- When the first frame received immediately after a start condition or restart condition is detected does not match a value of (device ID (1111_100b)) with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled) This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When the first frame received immediately after a start condition or restart condition is detected matches a value of (device ID (1111_100b) + 0 [W]) and the second frame does not match any of slave addresses 0 to 2 with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

HOA Flag (Host Address Detection)

[Setting condition]

- When the received slave address matches the host address (0001_000b) while the RIICnMR3.SMBS bit and RIICnSER.HOAE bit are set to 1 (host address detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the HOA bit after reading HOA = 1
- When a stop condition is detected
- When 0 is written to the RIICnMR3.SMBS bit in ICMR3 or the RIICnSER.HOAE bit
- When the received slave address does not match the host address (0001_000b) with the RIICnSER.HOAE bit set to 1 (host address detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the RIICnCR1.IICRST bit 1 to apply an RIIC reset or an internal reset

25.3.10 I²C Bus Status Register 2 (RIICnSR2)

Access Size: RIICnSR2 is a 8-bit readable/writable register

Address(es): RIICnSR2: <RIICn_base> + H'0009

Initial Value: H'00 This register is initialized by any reset.

Bit	7	6	5	4	3	2	1	0
	TDRE	TEND	RDRF	NACKF	STOP	START	AL	TMOF
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R(/W)*1	R(/W)*1	R(/W)*1	R(/W)*1	R(/W)*1	R(/W)*1	R(/W)*1

Bit	Bit Name	Initial Value	R/W	Description
7	TDRE	0	R	Transmit Data Empty Flag 0: RIICnDRT contains transmit data. 1: RIICnDRT contains no transmit data.
6	TEND	0	R(/W)*1	Transmit End Flag 0: Data is being transmitted. 1: Data has been transmitted.
5	RDRF	0	R(/W)*1	Receive Data Full Flag 0: RIICnDRR contains no receive data. 1: RIICnDRR contains receive data.
4	NACKF	0	R(/W)*1	NACK Reception Flag 0: NACK is not received. 1: NACK is received.
3	STOP	0	R(/W)*1	Stop Condition Detection Flag 0: Stop condition is not detected. 1: Stop condition is detected.
2	START	0	R(/W)*1	Start Condition Detection Flag 0: Start condition is not detected. 1: Start condition is detected.
1	AL	0	R(/W)*1	Arbitration-Lost Flag 0: Arbitration is not lost. 1: Arbitration is lost.
0	TMOF	0	R(/W)*1	Timeout Flag 0: No timeout has occurred. 1: Timeout has occurred.

Note 1. Only 0 can be written to this bit.

TMOF Flag (Timeout Flag)

This flag is set to 1 when the RIIC recognizes timeout after the SCL line state remains unchanged for a certain period.

[Setting condition]

The timeout function is enabled when the RIICnFER.TMOE bit is 1. It detects an abnormal bus state that the SCL line is held low or high during the following conditions:

- The bus is busy (RIICnCR2.BBSY = 1) in master mode (RIICnCR2.MST = 1).
- The slave address matches that of this module (RIICnSR1 register is not H'00) and the bus is busy (RIICnCR2.BBSY = 1) in slave mode (RIICnCR2.MST = 0).
- Issuing of a start condition is being requested (RIICnCR2.ST = 1) and the bus is free (RIICnCR2.BBSY = 0).

[Clearing conditions]

- When 0 is written to the TMOF bit after reading TMOF = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

AL Flag (Arbitration-Lost Flag)

This flag shows that bus mastership has been lost (loss in arbitration) due to a bus conflict or some other reason when a start condition is issued or an address and data are transmitted. The RIIC monitors the level on the SDA line during transmission and, if the level on the line does not match the value of the bit being output, sets the value of the AL bit to 1 to indicate that the bus is occupied by another device.

The RIIC can also set the flag to indicate the detection of loss of arbitration during NACK transmission in receive mode or during data transmission in slave mode.

[Setting conditions]

<When master arbitration-lost detection is enabled: RIICnFER.MALE = 1>

- When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock except for the ACK period during data (including slave address) transmission in master transmit mode (when the SDA line is driven low while the internal SDA output is at a high level (the SDA pin is in the high-impedance state))
- When a start condition is detected while the RIICnCR2.ST bit is 1 (start condition issuance request) or the internal SDA output state does not match the SDA line level
- When the RIICnCR2.ST bit is set to 1 (start condition issuance request) with the RIICnCR2.BBSY flag set to 1.

<When NACK arbitration-lost detection is enabled: RIICnFER.NALE = 1>

- When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock in the ACK period during NACK transmission in receive mode

<When slave arbitration-lost detection is enabled: RIICnFER.SALE = 1>

- When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock except for the ACK period during data transmission in slave transmit mode

[Clearing conditions]

- When 0 is written to the AL bit after reading AL = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

Table 25.8 Relationship between Arbitration-Lost Generation Sources and Arbitration-Lost Enable Functions

RIICnFER			RIICnSR2		
MALE	NALE	SALE	AL	Error	Arbitration-Lost Generation Source
1	x	x	1	Start condition issuance error	When internal SDA output state does not match SDA line level when a start condition is detected while the RIICnCR2.ST bit is 1
					When RIICnCR2.ST is set to 1 with RIICnCR2.BBSY set to 1
			1	Transmit data mismatch	When transmit data (including slave address) does not match the bus state in master transmit mode
x	1	x	1	NACK transmission mismatch	When ACK is detected during transmission of NACK in master receive mode or slave receive mode
x	x	1	1	Transmit data mismatch	When transmit data does not match the bus state in slave transmit mode

Note: x: Don't care

START Flag (Start Condition Detection Flag)

[Setting condition]

- When a start condition (or a restart condition) is detected

[Clearing conditions]

- When 0 is written to the START bit after reading START = 1
- When a stop condition is detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

STOP Flag (Stop Condition Detection Flag)

[Setting condition]

- When a stop condition is detected

[Clearing conditions]

- When 0 is written to the STOP bit after reading STOP = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

NACKF Flag (NACK Reception Flag)

[Setting condition]

- When acknowledge is not received (NACK is received) from the receive device in transmit mode with the RIICnFER.NACKE bit set to 1 (transfer suspension enabled)

[Clearing conditions]

- When 0 is written to the NACKF bit after reading NACKF = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

Remark When the NACKF flag is set to 1, the RIIC suspends data transmission/reception. Writing to RIICnDRT in transmit mode or reading from RIICnDRR in receive mode with the NACKF flag set to 1 does not enable data transmit/ receive operation. To restart data transmission/reception, clear the NACKF flag to 0.

RDRF Flag (Receive Data Full Flag)

[Setting conditions]

- Slave receive mode
  - When the received slave address matches and the RIICnCR2.TRS bit is cleared to 0 after a start condition (or a restart condition) is detected
  - At the rising edge of the eighth or ninth SCL clock cycle (selected by the RIICnMR3.RDRFS bit) after receive data is transferred from RIICnDRS to RIICnDRR
- Master receive mode
  - When the slave address and the data direction are transmitted and the receive mode is entered (the RIICnCR2.TRS bit is set to 1) after a start condition (or a restart condition) is issued
  - At the rising edge of the eighth or ninth SCL clock cycle (selected by the RIICnMR3.RDRFS bit) after receive data is transferred from RIICnDRS to RIICnDRR

[Clearing conditions]

- When 0 is written to the RDRF bit after reading RDRF = 1
- When data is read from RIICnDRR
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

TEND Flag (Transmit End Flag)

[Setting condition]

At the rising edge of the ninth SCL clock cycle while the TDRE flag is 1

[Clearing conditions]

- When 0 is written to the TEND bit after reading TEND = 1
- When data is written to RIICnDRT
- When a stop condition is detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

TDRE Flag (Transmit Data Empty Flag)

[Setting conditions]

- When data has been transferred from RIICnDRT to RIICnDRS and RIICnDRT becomes empty
- When the RIICnCR2.TRS bit is set to 1
  - When the RIICnCR2.MST bit is set to 1 after a start condition (or a restart condition) is detected
  - When the RIIC enters transmit mode from receive mode
- When the received slave address matches while the TRS bit is 1

[Clearing conditions]

- When data is written to RIICnDRT
- When the RIICnCR2.TRS bit is cleared to 0
  - When a stop condition is detected
  - When the RIIC enters receive mode from transmit mode
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

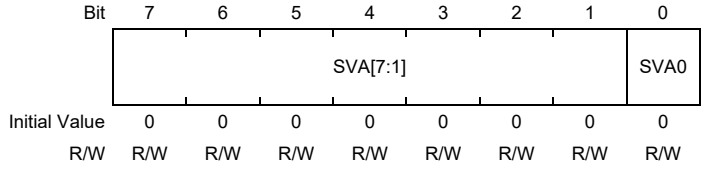
Remark When the NACKF flag is set to 1 while the RIICnFER.NACKE bit is 1, the RIIC suspends data transmission/reception. Here, if the TDRE flag is 0 (next transmit data has been written), data is transferred to the RIICnDRS register and the RIICnDRT register becomes empty at the rising edge of the ninth clock cycle, but the TDRE flag is not set to 1.

25.3.11 I²C Slave Address Register Lower y (y = 0 to 2) (RIICnSARLy)

Access Size: RIICnSARLy is a 8-bit readable/writable register.

Address(es): RIICnSARL0: <RIICn_base> + H'000A
RIICnSARL1: <RIICn_base> + H'000C
RIICnSARL2: <RIICn_base> + H'000E

Initial Value: H'00 This register is initialized by any reset.



Bit	Bit Name	Initial Value	R/W	Description
7 to 1	SVA[7:1]	All 0	R/W	7-Bit Address Bits [6:0]/10-Bit Address Lower Bits [7:1] A slave address is set.
0	SVA0	0	R/W	10-Bit Address LSB A slave address is set.

SVA0 Bit (10-Bit Address LSB [0])

When the 10-bit address format is selected (RIICnSARUy.FS = 1), this bit functions as the LSB of a 10-bit address and forms the lower 8 bits of a 10-bit address in combination with the SVA[7:1] bits.

When the RIICnSER.SARyE bit is set to 1 (RIICnSARLy and RIICnSARUy enabled) and the RIICnSARUy.FS bit is 1, this bit is valid.

While the RIICnSARUy.FS bit or the RIICnSER.SARyE bit is 0, the setting of this bit is ignored.

SVA[7:1] Bits (7-Bit Address Bits[6:0]/10-Bit Address Lower Bits [7:1])

When the 7-bit address format is selected (RIICnSARUy.FS = 0), these bits function as a 7-bit address. When the 10-bit address format is selected (RIICnSARUy.FS = 1), these bits function as the lower 8 bits of a 10-bit address in combination with the SVA0 bit.

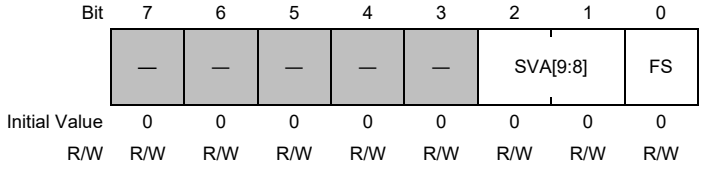
While the RIICnSER.SARyE bit is 0, the setting of these bits is ignored

25.3.12 I²C Slave Address Register Upper y (y = 0 to 2) (RIICnSARUy)

Access Size: RIICnSARy is a 8 bit readable/writable register.

Address(es): RIICnSARU0: <RIICn_base> + H'000B
RIICnSARU1: <RIICn_base> + H'000D
RIICnSARU2: <RIICn_base> + H'000F

Initial Value: H'00 This register is initialized by any reset.



Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
2, 1	SVA[9:8]	00	R/W	10-Bit Address Upper Bits [9:8] A slave address is set.
0	FS	0	R/W	7-Bit/10-Bit Address Format Selection 0: The 7-bit address format is selected. 1: The 10-bit address format is selected.

FS Bit (7-Bit/10-Bit Address Format Selection)

This bit is used to select 7-bit address or 10-bit address for slave address y (in SARLy and SARUy).

When the RIICnSER.SARyE bit is set to 1 (RIICnSARLy and RIICnSARUy enabled) and the RIICnSARUy.FS bit is 0, the 7-bit address format is selected for slave address y, the SVA[7:1] setting in RIICnSARLy is valid, and the settings of the SVA[9:8] bits and the SVA0 bit in RIICnSARLy are ignored.

When the RIICnSER.SARyE bit is set to 1 (RIICnSARLy and RIICnSARUy enabled) and the RIICnSARUy.FS bit is 1, the 10-bit address format is selected for slave address y and the settings of the SVA[9:8] bits and RIICnSARLy are valid.

While the RIICnSER.SARyE bit is 0 (RIICnSARLy and RIICnSARUy disabled), the setting of the RIICnSARUy.FS bit is invalid.

SVA[9:8] Bits (10-Bit Address Upper Bits [9:8])

When the 10-bit address format is selected (FS = 1), these bits function as the upper 2 bits of a 10-bit address.

When the RIICnSER. SARyE bit is set to 1 (RIICnSARLy and RIICnSARUy enabled) and the RIICnSARUy.FS bit is 1, these bits are valid.

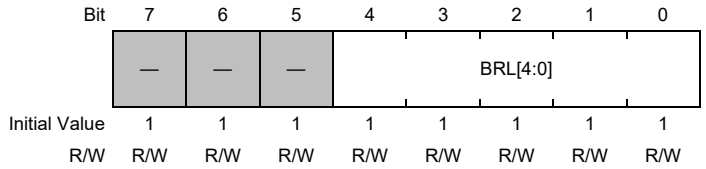
While the RIICnSARUy.FS bit or RIICnSER.SARyE bit is 0, the setting of these bits is ignored.

25.3.13 I²C Bus Bit Rate Low-Level Register (RIICnBRL)

Access Size: RIICnBRL is a 8 bit readable/writable register.

Address(es): RIICnBRL: <RIICn_base> + H'0010

Initial Value: H' FF This register is initialized by any reset.



Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	111	R/W	Reserved These bits are read as 1. The write value should be 1.
4 to 0	BRL[4:0]	H'1F	R/W	Bit Rate Low-Level Period Low-level period of SCL clock

The RIICnBRL register is a 5-bit register that is used to set the width at low level for the SCL clock.

It also works to generate the data setup time for automatic SCL low-hold operation (see **Section 25.10, Automatic Low-Hold Function for SCL**); when the RIIC is used only in slave mode, this register needs to be set to a value longer than the data setup time*[1].

RIICnBRL counts the low-level period with the internal reference clock source (IICϕ) specified by the RIICnMR1.CKS[2:0] bits.

If the digital noise filter is enabled (the RIICnFER.NFE bit is 1), set the RIICnBRL register to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the RIICnMR3.NF[1:0] bits.

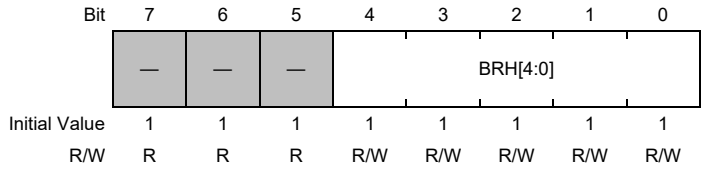
Note 1. Data setup time (t_{SU}: DAT)
- 250 [ns] (0 to 100 [kbps]: standard mode (Sm))
- 100 [ns] (0 to 400 [kbps]: fast mode (Fm))
- 50 [ns] (0 to 1 [Mbps]: fast mode plus (Fm+))

25.3.14 I²C Bus Bit Rate High-Level Register (RIICnBRH)

Access Size: RIICnBRH is a 8 bit readable/writable register..

Address(es): RIICnBRH: <RIICn_base> + H'0011

Initial Value: H' FF This register is initialized by any reset.



Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	111	R	Reserved These bits are read as 1. The write value should be 1.
4 to 0	BRH[4:0]	H'1F	R/W	Bit Rate High-Level Period High-level period of SCL clock

RIICnBRH is a 5-bit register to set the high-level period of SCL clock. RIICnBRH is valid in master mode. If the RIIC is used only in slave mode, this register need not to set the high-level period.

RIICnBRH counts the high-level period with the internal reference clock source (IICϕ) specified by the RIICnMR1.CKS[2:0] bits in ICMR1.

If the digital noise filter is enabled (the RIICnFER.NFE bit is 1), set the RIICnBRH register to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the RIICnMR3.NF[1:0] bits.

The I²C transfer rate and the SCL clock duty cycles are calculated using one of the following expressions (1) to (5) according to the register settings.

Remark The minimum value that can be specified in RIICnBRL and RIICnBRH is determined according to the values of the SCLE and NFE bits in RIICnFER and the NF bit in RIICnMR3. For details of the minimum specifiable value, see **Table 25.9**.

(1) When SCLE = 0

Transfer rate = 1 / {[(BRH + 1) + (BRL + 1)] / IICϕ + tr + tf}
Duty cycle = {tr + (BRH + 1) / IICϕ} / {tr + tf + [(BRH + 1) + (BRL + 1)] / IICϕ}

(2) When SCLE = 1, NFE = 0, CKS = 000 (IICϕ = P0ϕ)

Transfer rate = 1 / {[(BRH + 3) + (BRL + 3)] / IICϕ + tr + tf}
Duty cycle = {tr + (BRH + 3) / IICϕ} / {tr + tf + [(BRH + 3) + (BRL + 3)] / IICϕ}

(3) When SCLE = 1, NFE = 1, CKS = 000 (IICϕ = P0ϕ)

Transfer rate = 1 / {[(BRH + 3 + nf) + (BRL + 3 + nf)] / IICϕ + tr + tf}
Duty cycle = {tr + (BRH + 3 + nf) / IICϕ} / {tr + tf + [(BRH + 3 + nf) + (BRL + 3 + nf)] / IICϕ}

(4) When SCLE = 1, NFE = 0, CKS ≠ 000 (IICϕ < P0ϕ)

Transfer rate = 1 / {[(BRH + 2) + (BRL + 2)] / IICϕ + tr + tf}
Duty cycle = {tr + (BRH + 2) / IICϕ} / {tr + tf + [(BRH + 2) + (BRL + 2)] / IICϕ}

(5) When SCLE = 1, NFE = 1, CKS ≠ 000 (IICϕ < P0ϕ)

Transfer rate = 1 / {[(BRH + 2 + nf) + (BRL + 2 + nf)] / IICϕ + tr + tf}
Duty cycle = {tr + (BRH + 2 + nf) / IICϕ} / {tr + tf + [(BRH + 2 + nf) + (BRL + 2 + nf)] / IICϕ}

[Symbols in the expressions]

SCLE: RIICnFER.SCLE bit

BRH: RIICnBRH.BRH[4:0] bits

BRL: RIICnBRL.BRL[4:0] bits

CKS: RIICnMR1.CKS bits

NFE: RIICnFER.NFE bit

IICϕ: Internal reference clock selected by the CKS bits*1

tf: SCL signal falling time [s]*2

tr: SCL signal rising time [s]*2

nf: Number of digital noise filter stages specified in the RIICnMR3.NF[1:0] bits*3

Note 1. IICϕ = PCLK × Division ratio

Note 2. The SCLn line rising time [tr] and SCLn line falling time [tf] depend on the total bus line capacitance [Cb] and the pull-up resistor [Rp]. For details, see the I2C bus standard from NXP Semiconductors.

Note 3. nf = Number of digital noise filter selected by ICMR3.NF bit.

Table 25.9 Minimum Specifiable Value for RIICnBRL and RIICnBRH

SCLE	NFE	nf	Minimum Pulse Width that Passes through Digital Filter	Minimum Specifiable Value for BRH and BRL	Pulse Width when Minimum Value is Specified
0	0	—	1 × IICϕ	1	2 × IICϕ
0	1	1	2 × IICϕ	2	3 × IICϕ
0	1	2	3 × IICϕ	3	4 × IICϕ
0	1	3	4 × IICϕ	4	5 × IICϕ
0	1	4	5 × IICϕ	5	6 × IICϕ
IICϕ cycle > P0ϕ cycle (CKS ≠ 000)					
1	0	—	1 × IICϕ	0	2 × IICϕ
1	1	1	2 × IICϕ	1	4 × IICϕ
1	1	2	3 × IICϕ	2	6 × IICϕ
1	1	3	4 × IICϕ	3	8 × IICϕ
1	1	4	5 × IICϕ	4	10 × IICϕ
IICϕ cycle = P0ϕ cycle (CKS = 000)					
1	0	—	2 × P0ϕ	0	3 × IICϕ
1	1	1	3 × P0ϕ	1	5 × IICϕ
1	1	2	4 × P0ϕ	2	7 × IICϕ
1	1	3	5 × P0ϕ	3	9 × IICϕ
1	1	4	6 × P0ϕ	4	11 × IICϕ

Table 25.10 and **Table 25.11** list examples of RIICnBRH/RIICnBRL settings.

Table 25.10 Examples of RIICnBRH/RIICnBRL Settings for Transfer Rate (when RIICnFER.SCLE = 1 and RIICnFER.NFE= 0)

Transfer Rate [kbps]	Peripheral Clock Operating Frequency P0ϕ [MHz]			
	100			
	RIICnMR1.CKS[2:0]	RIICnBRH.BRH	RIICnBRL.BRL	Actual transfer rate [kbps]
12	111b	30 (H'FE)	31 (H'FF)	12
50	101b	28 (H'FC)	30 (H'FE)	50.4
100	100b	28 (H'FC)	31 (H'FF)	99.2
400	010b	29 (H'FD)	30 (H'FE)	396.8
1000	001b	20 (H'F4)	25 (H'F9)	1020.4

Table 25.11 Examples of RIICnBRH/RIICnBRL Settings for Transfer Rate (when RIICnFER.SCLE = 1, RIICnFER.NFE= 1, and Number of NF Stages = 4)

Transfer Rate [kbps]	Peripheral Clock Operating Frequency P0ϕ [MHz]			
	100			
	RIICnMR1.CKS[2:0]	RIICnBRH.BRH	RIICnBRL.BRL	Actual transfer rate [kbps]
12	111b	25 (H'F9)	28 (H'FC)	12
50	101b	23 (H'F7)	28 (H'FC)	49.6
100	100b	24 (H'F8)	27 (H'FB)	99.2
400	010b	25 (H'F9)	26 (H'FA)	396.8
1000	001b	14 (H'EE)	24 (H'F8)	1000

25.3.15 I²C Bus Transmit Data Register (RIICnDRT)

Access Size: RIICnDRT is a 8 bit readable/writable register.

Address(es): RIICnDRT: <RIICn_base> + H'0012

Initial Value: H'FF This register is initialized by any reset.

Bit	7	6	5	4	3	2	1	0
	DRT[7:0]							
Initial Value	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When RIICnDRT detects a space in the I²C bus shift register (RIICnDRS), it transfers the transmit data that has been written to RIICnDRT to RIICnDRS and starts transmitting data in transmit mode.

The double-buffer structure of RIICnDRT and RIICnDRS allows continuous transmit operation if the next transmit data has been written to RIICnDRT while the RIICnDRS data is being transmitted.

RIICnDRT can always be read and written. Write transmit data to RIICnDRT once when a transmit data empty interrupt (INTRIICTI) request is generated.

25.3.16 I²C Bus Receive Data Register (RIICnDRR)

Access Size: RIICnDRR is a 8 bit readable/writable register.

Address(es): RIICnDRR: <RIICn_base> + H'0013

Initial Value: H'00 This register is initialized by any reset.

Bit	7	6	5	4	3	2	1	0
	DRR[7:0]							
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

When 1 byte of data has been received, the received data is transferred from the I²C bus shift register (RIICnDRS) to RIICnDRR to enable the next data to be received.

The double-buffer structure of RIICnDRS and RIICnDRR allows continuous receive operation if the received data has been read from RIICnDRR while RIICnDRS is receiving data.

RIICnDRR cannot be written. Read data from RIICnDRR once when a receive data full interrupt (INTRIICRI) request is generated.

If RIICnDRR.DRR[7:0] receives the next receive data before the current data is read from RIICnDRR (while the RIICnSR2.RDRF flag is 1), the RIIC automatically holds the SCL clock low one cycle before the RIICnSR2.RDRF flag is set to 1 next.

25.3.17 I²C Bus Shift Register (RIICnDRS)

Access Size: This register is not accessible.

Address(es): —

Initial Value: —

Bit	7	6	5	4	3	2	1	0
	DRS[7:0]							
Initial Value	—	—	—	—	—	—	—	—
R/W	—	—	—	—	—	—	—	—

RIICnDRS is an 8-bit shift register to transmit and receive data.

During transmission, transmit data is transferred from RIICnDRT to RIICnDRS and is sent from the SDAn pin. During reception, data is transferred from RIICnDRS to RIICnDRR after 1 byte of data has been received.

RIICnDRS cannot be accessed directly.

25.4 Interrupt Sources

The RIIC issues eight types of interrupt request: transmit end, receive data full, transmit data empty, stop condition detection, start condition detection, NACK reception, arbitration-lost, and timeout.

Table 25.12 lists details of the several interrupt requests. The receive data full and transmit data empty sources are both capable of launching data transfer by the DMAC.

Table 25.12 Interrupt Sources

Symbol	Interrupt Source	Interrupt Flag	DMAC Launching	Priority*1	Interrupt Condition
INTRIICRI	Receive-data-full	RDRF	Possible	High	RDRF = 1 • RIE = 1
INTRIICTI	Transmit-data-empty	TDRE	Possible		TDRE = 1 • TIE = 1
INTRIICTEI	Transmission complete	TEND	Not possible		TEND = 1 • TEIE = 1
INTRIICNAKI	Reception of a NACK	NACKF	Not possible		NACKF = 1 • NAKIE = 1
INTRIICSPI	Detection of a stop condition	STOP	Not possible		STOP = 1 • SPIE = 1
INTRIICSTI	Detection of a start condition	START	Not possible		START = 1 • STIE = 1
INTRIICALI	Arbitration lost	AL	Not possible		AL = 1 • ALIE = 1
INTRIICTMOI	Timeout	TMOF	Not possible	Low	TMOF = 1 • TMOIE = 1

Note 1. When the interrupt priority register (ICDIPRn) setting is the same

Clear or mask each flag during interrupt handling.

CAUTIONS

1. There is a latency (delay) between the execution of a write instruction for a peripheral module by the CPU and actual writing to the module. Thus, when an interrupt flag has been cleared or masked, read the relevant flag again to check whether clearing or masking has been completed, and then return from interrupt processing. Returning from interrupt processing without checking that writing to the module has been completed creates a possibility of repeated processing of the same interrupt.
2. Since INTRIICRI and INTRIICTI are edge-detected interrupts, they do not require clearing.
3. When using the INTRIICTEI interrupt, clear the RIICnSR2.TEND flag in the INTRIICTEI interrupt processing.
4. When using the INTRIICSPI interrupt, clear the RIICnSR2.STOP flag in the INTRIICSPI interrupt processing.
5. When using the INTRIICSTI interrupt, clear the RIICnSR2.START flag in the INTRIICSTI interrupt processing.
6. When using the INTRIICNAKI interrupt, clear the RIICnSR2.NACKF flag in the INTRIICNAKI interrupt processing.
7. When using the INTRIICALI interrupt, clear the RIICnSR2.AL flag in the INTRIICALI interrupt processing.
8. When using the INTRIICTMOI interrupt, clear the RIICnSR2.TMOF flag in the INTRIICTMOI interrupt processing.

25.5 Operation

25.5.1 Communication Data Format

The I²C bus format consists of 8-bit data and 1-bit acknowledge (one frame). After a start condition or restart condition is issued, the master device sends the slave address and data direction in the first frame. The specified slave is valid until a stop condition is issued or a new slave is specified by a restart condition.

Figure 25.3 shows the I²C bus format, and **Figure 25.4** shows the I²C bus timing.

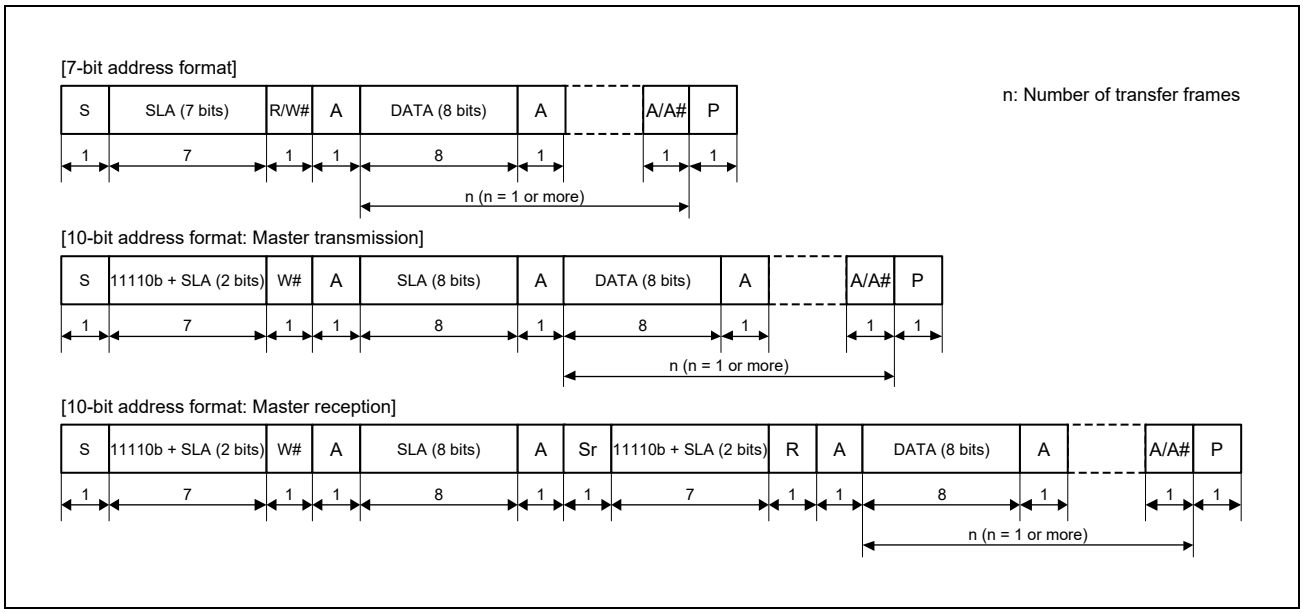
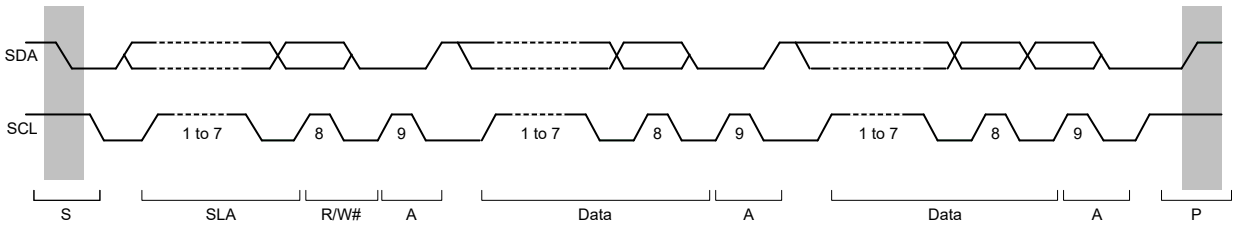


Figure 25.3 I²C Bus Format



S: Start condition. The master device drives the SDA line low from high level while the SCL line is at a high level.

SLA: Slave address, by which the master device selects a slave device.

R/W#: Indicates the direction of data transfer: from the slave device to the master device when R/W is 1, or from the master device to the slave device when R/W is 0.

A: Acknowledge. The receive device drives the SDA line low. (In master transmit mode, the slave device returns acknowledge. In master receive mode, the master device returns acknowledge.)

A#: Not-acknowledge. The receiving device has not returned a response or is not present so the SDA line has remained at the high level.

Sr: Restart condition. The master device drives the SDA line low from the high level after the setup time has elapsed with the SCL line at the high level.

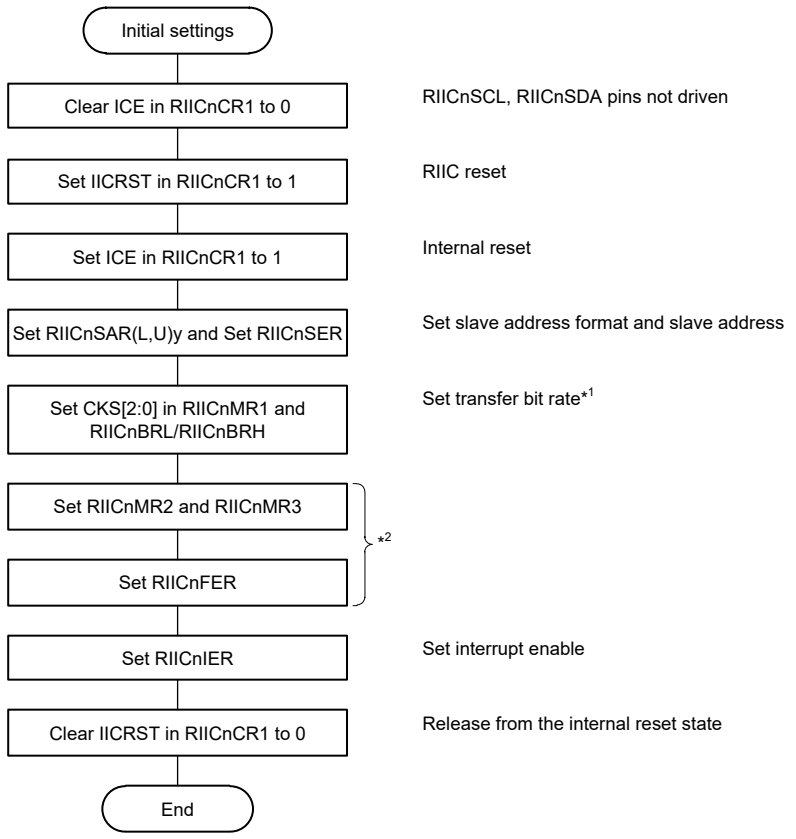
DATA: Transmitted or received data

P: Stop condition. The master device drives the SDA line high from low level while the SCL line is at a high level.

Figure 25.4 I^2C Bus Timing (SLA = 7 Bits)

25.5.2 Initial Settings

Before starting data transmission and reception, initialize the RIIC according to the procedure in **Figure 25.5**.



Note: y = 0 to 2

Note 1. When the RIIC is used only in slave mode, set the RIICnBRL register to a value at least the same as the data setup time.

Note 2. Set these registers, as necessary.

Figure 25.5 Example of RIIC Initialization Flowchart

25.5.3 Master Transmit Operation

In master transmit operation, the RIIC outputs the SCL (clock) and transmitted data signals as the master device, and the slave device returns acknowledgements. **Figure 25.6** shows an example of usage of master transmission and **Figure 25.7** to **Figure 25.9** show the timing of operations in master transmission.

The following describes the procedure and operations for master transmission.

(1) Set the RIICnCR1.IICRST bit 1 to 1 (RIIC reset) and then set the RIICnCR1.ICE bit to 1 (internal reset) with the RIICnCR1.ICE bit cleared to 0 (RIICnSCL and RIICnSDA pins not driven). This initializes the internal state and the various flags of RIICnSR1. After that, set registers RIICnSAR(L,U)y, RIICnSER, RIICnMR1, RIICnBRH, and RIICnBRL (y = 0 to 2), and set the other registers as necessary (for initial settings of the RIIC, see **Figure 25.5**). When the necessary register settings have been completed, set the RIICnCR1.IICRST bit to 0 (for release from the reset state). This step is not necessary if initialization of the RIIC has already been completed.

(2) Read the RIICnCR2.BBSY flag to check that the bus is open, and then set the RIICnCR2.ST bit to 1 (start condition issuance request). Upon receiving the request, the RIIC issues a start condition. At the same time, the BBSY flag and the RIICnSR2.START flag are automatically set to 1 and the ST bit is automatically cleared to 0. At this time, if the start condition is detected and the internal levels for the SDA output state and the levels on the SDA line have matched while the ST bit is 1, the RIIC recognizes that issuing of the start condition as requested by the ST bit has been successfully completed, and the RIICnCR2.MST and TRS bits are automatically set to 1, placing the RIIC in master transmit mode. The RIICnSR2.TDRE flag is also automatically set to 1 in response to setting of the TRS bit to 1.

(3) Check that the RIICnSR2.TDRE flag is 1, and then write the value for transmission (the slave address and the R/W# bit) to RIICnDRT. Once the data for transmission are written to RIICnDRT, the TDRE flag is automatically cleared to 0, the data are transferred from RIICnDRT to RIICnDRS, and the TDRE flag is again set to 1. After the byte containing the slave address and R/W# bit has been transmitted, the value of the TRS bit is automatically updated to select master transmit or master receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 0, the RIIC continues in master transmit mode.

Since the RIICnSR2.NACKF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the RIICnSR2.STOP bit and then write 1 to the RIICnCR2.SP bit to issue a stop condition.

For data transmission with an address in the 10-bit format, start by writing 1111_0b, the two higher-order bits of the slave address, and W# to RIICnDRT as the first address transmission. Then, as the second address transmission, write the eight lower-order bits of the slave address to RIICnDRT.

(4) After confirming that the RIICnSR2.TDRE flag is 1, write the data for transmission to the RIICnDRT register. The RIIC automatically holds the SCL line low until the data for transmission are ready or a stop condition is issued.

(5) After the last byte of the data to be transmitted is written to the RIICnDRT register, wait until the value of the RIICnSR2.TEND flag returns to 1, after that write 1 to the RIICnSR2.STOP bit and then set the RIICnCR2.SP bit to 1 (stop condition issuance request). Upon receiving a stop condition issuance request, the RIIC issues the stop condition.

(6) Upon detecting the stop condition, the RIIC automatically clears the RIICnCR2.MST and TRS bits to 00b and enters slave receive mode. Furthermore, it automatically clears the RIICnSR2.TDRE and TEND flags to 0, and sets the RIICnSR2.STOP flag in to 1.

(7) After checking that the RIICnSR2.STOP flag is 1, set the RIICnSR2.NACKF and STOP flags to 0 for the next transfer operation.

CAUTION

Operations for transfer start if the RIICnSR2.NACKF flag is cleared to 0 before RIICnSR2.STOP is set to 1. Be sure to confirm that RIICnSR2.STOP is set to 1 before clearing RIICnSR2.NACKF to 0. In particular, when the NACK receive interrupt (INTRIICNAKI) is in use, take care not to clear the NACKF flag to 0 before the STOP flag is set to 1 during interrupt processing.

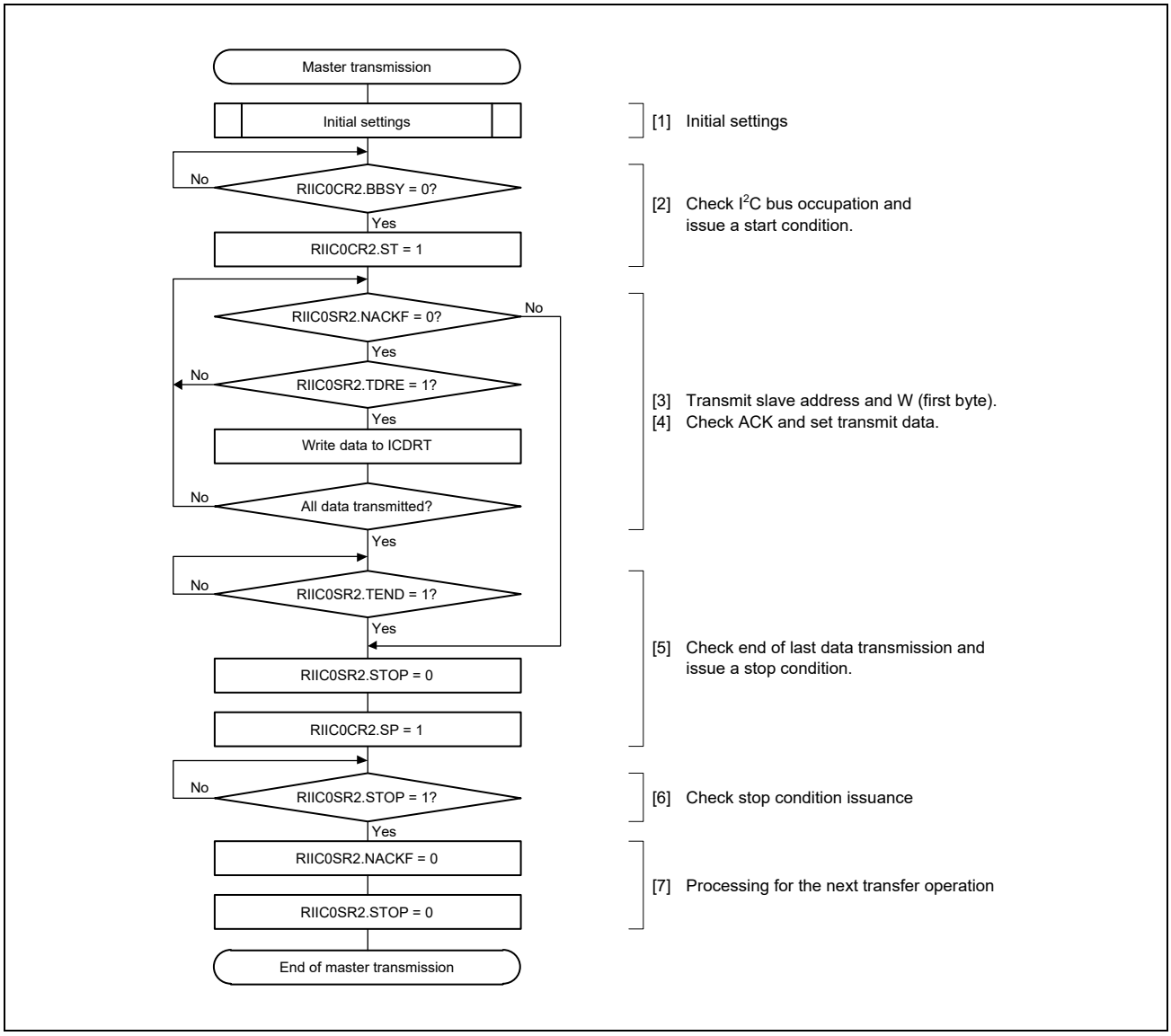


Figure 25.6 Example of Master Transmission Flowchart

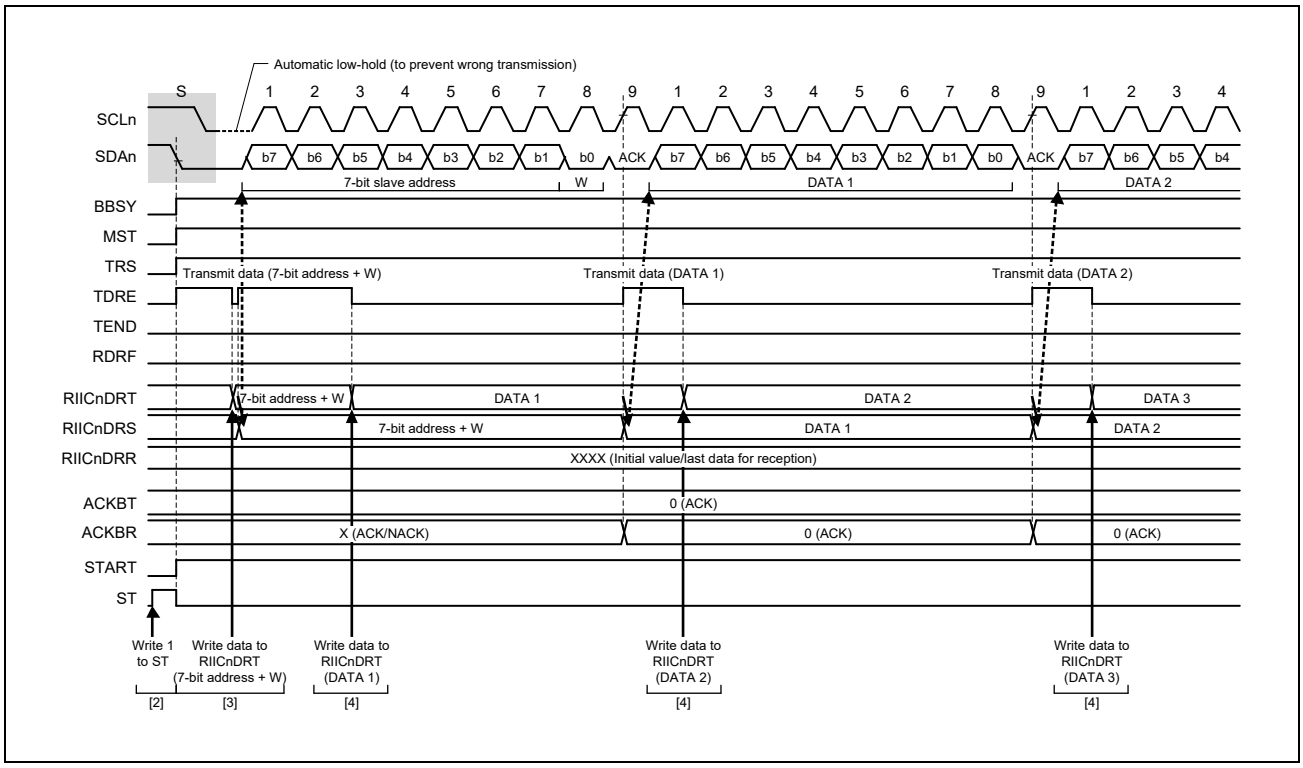


Figure 25.7 Master Transmit Operation Timing (1) (7-Bit Address Format)

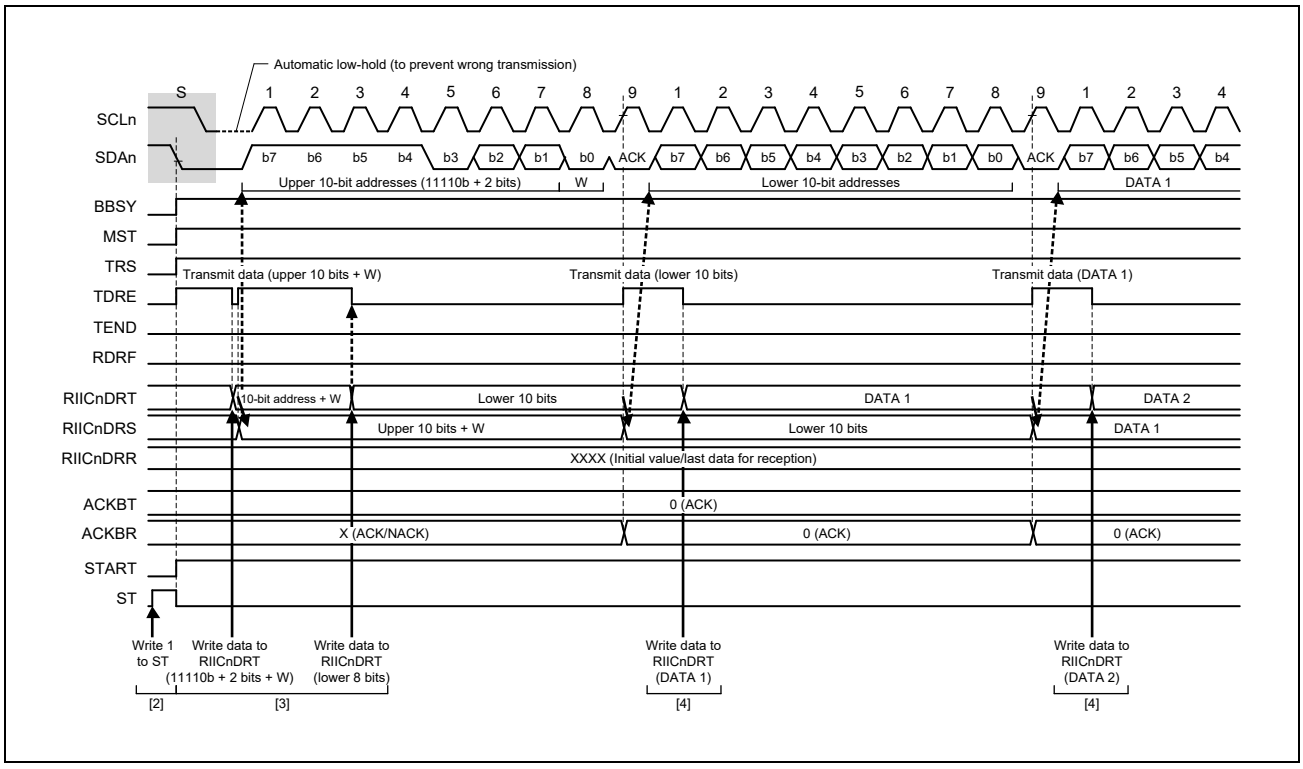


Figure 25.8 Master Transmit Operation Timing (2) (10-Bit Address Format)

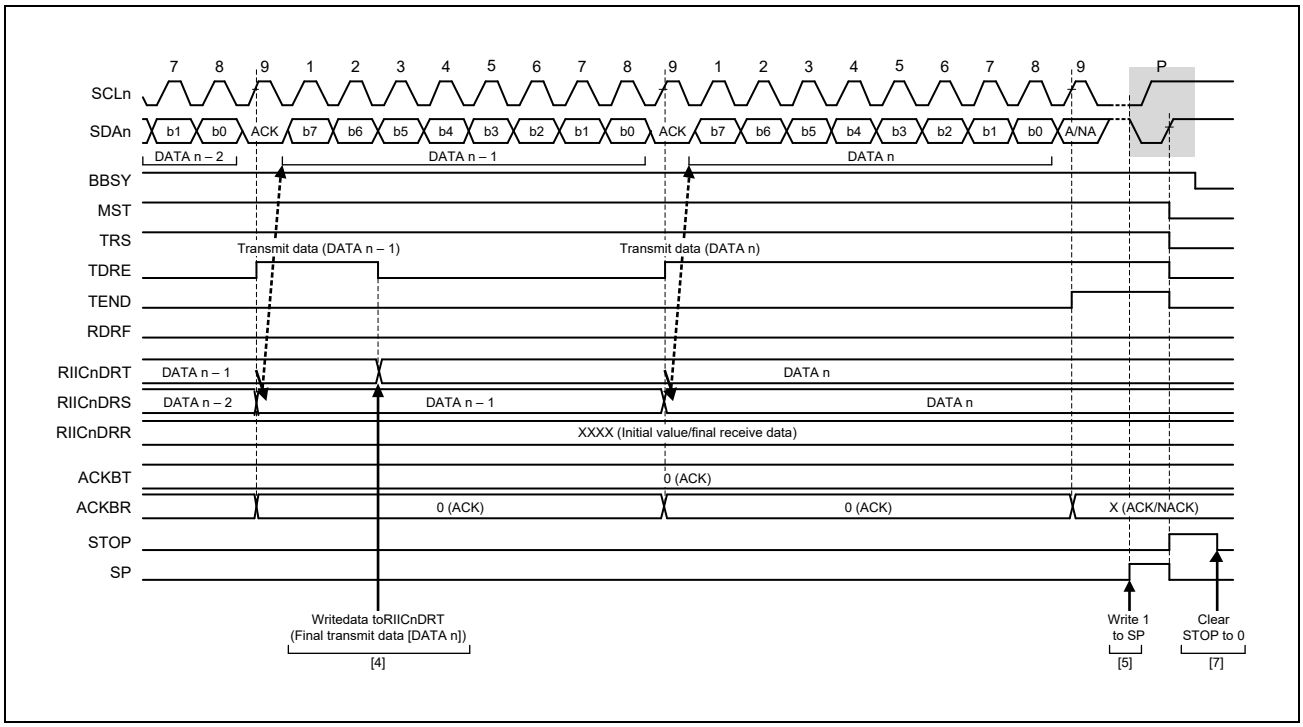


Figure 25.9 Master Transmit Operation Timing (3)

25.5.4 Master Receive Operation

In master receive operation, the RIIC as a master device outputs the SCL (clock) signal, receives data from the slave device, and returns acknowledgements. Since the RIIC must start by sending a slave address to the corresponding slave device, this part of the procedure is performed in master transmit mode, but the subsequent steps are in master receive mode.

Figure 25.10 shows an example of usage for the master reception of 3 or more bytes (7-bit address format), **Figure 25.14** shows an example of usage for the master reception of 1 or 2 bytes (7-bit address format), and **Figure 25.11** to **Figure 25.13** show the timing of operations in master reception.

The following describes the procedure and operations for master reception.

(1) Set the RIICnCR1.IICRST bit to 1 (RIIC reset) and then set the RIICnCR1.ICE bit to 1 (internal reset) with the RIICnCR1.ICE bit cleared to 0 (RIICnSCL and RIICnSDA pins not driven). This initializes the internal state and the various flags of RIICnSR1. After that, set registers RIICnSAR(L,U)y, RIICnSER, RIICnMR1, RIICnBRH, and RIICnBRL (y = 0 to 2), and set the other registers as necessary (for initial settings of the RIIC, see **Figure 25.5**). When the necessary register settings have been completed, set the RIICnCR1.IICRST bit to 0 (for release from the reset state). This step is not necessary if initialization of the RIIC has already been completed.

(2) Read the RIICnCR2.BBSY flag to check that the bus is open, and then set the RIICnCR2.ST bit to 1 (start condition issuance request). Upon receiving the request, the RIIC issues a start condition. When the RIIC detects the start condition, the BBSY flag and the RIICnSR2.START flag are automatically set to 1 and the ST bit is automatically cleared to 0. At this time, if the start condition is detected and the levels for the SDA output and the levels on the SDA line have matched while the ST bit is 1, the RIIC recognizes that issuing of the start condition as requested by the ST bit has been successfully completed, and the RIICnCR2.MST and TRS bits are automatically set to 1, placing the RIIC in master transmit mode. The RIICnSR2.TDRE flag is also automatically set to 1 in response to setting of the TRS bit to 1.

(3) Check that the RIICnSR2.TDRE flag is 1, and then write the value for transmission (the first byte indicates the slave address and value of the R/W# bit) to RIICnDRT. Once the data for transmission are written to RIICnDRT, the TDRE flag is automatically cleared to 0, the data are transferred from RIICnDRT to RIICnDRS, and the TDRE flag is again set to 1. Once the byte containing the slave address and R/W# bit has been transmitted, the value of the RIICnCR2.TRS bit is automatically updated to select transmit or receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 1, the RIICnCR2.TRS bit is cleared to 0 on the rising edge of the ninth cycle of SCL (the clock signal), placing the RIIC in master receive mode. At this time, the TDRE flag is automatically cleared to 0 and the RIICnSR2.RDRF flag is automatically set to 1.

Since the RIICnSR2.NACKF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the RIICnCR2.SP bit to issue a stop condition.

For master reception from a device with a 10-bit address, start by using master transmission to issue the 10-bit address, and then issue a restart condition. After that, transmitting 1111_0b, the two higher-order bits of the slave address, and the R bit places the RIIC in master receive mode.

(4) Dummy read RIICnDRR after confirming that the RIICnSR2.RDRF flag is 1; this makes the RIIC start output of the SCL (clock) signal and start data reception.

(5) After 1 byte of data has been received, the RIICnSR2.RDRF flag is set to 1 on the rising edge of the eighth or ninth cycle of SCL clock (the clock signal) as selected by the RIICnMR3.RDRFS bit. Reading out RIICnDRR at this time will produce the received data, and the RDRF flag is automatically cleared to 0 at the same time. Furthermore, the value of the acknowledgement field received during the ninth cycle of SCL clock is returned as the value set in the RIICnMR3.ACKBT bit. Furthermore, if the next byte to be received is the next to last byte, set the RIICnMR3.WAIT bit to 1 (for wait insertion) before reading the RIICnDRR (containing the second byte from last). As well as enabling NACK output even in the case of delays in processing to set the RIICnMR3.ACKBT bit

to 1 (NACK) in step (6), due to other interrupts, etc., this fixes the SCL line to the low level on the rising edge of the ninth clock cycle in reception of the last byte, so the state is such that issuing a stop condition is possible.

(6) When the RIICnMR3.RDRFS bit is 0 and the slave device must be notified that it is to end transfer for data reception after transfer of the next (final) byte, set the RIICnMR3.ACKBT bit to 1 (NACK).

(7) After reading out the byte before last from the RIICnDRR register, if the value of the RIICnSR2.RDRF flag is confirmed to be 1, write 1 to the RIICnSR2.STOP bit and then write 1 to the RIICnCR2.SP bit (stop condition issuance request) and then read the last byte from RIICnDRR. When RIICnDRR is read, the RIIC is released from the wait state and issues the stop condition after low-level output in the ninth clock cycle is completed or the SCL line is released from the low-hold state.

(8) Upon detecting the stop condition, the RIIC automatically clears the RIICnCR2.MST and TRS bits to 00b and enters slave receive mode. Furthermore, detection of the stop condition leads to setting of the RIICnSR2.STOP flag to 1.

(9) After checking that the RIICnSR2.STOP flag is 1, clear the RIICnSR2.NACKF and STOP flags to 0 for the next transfer operation.

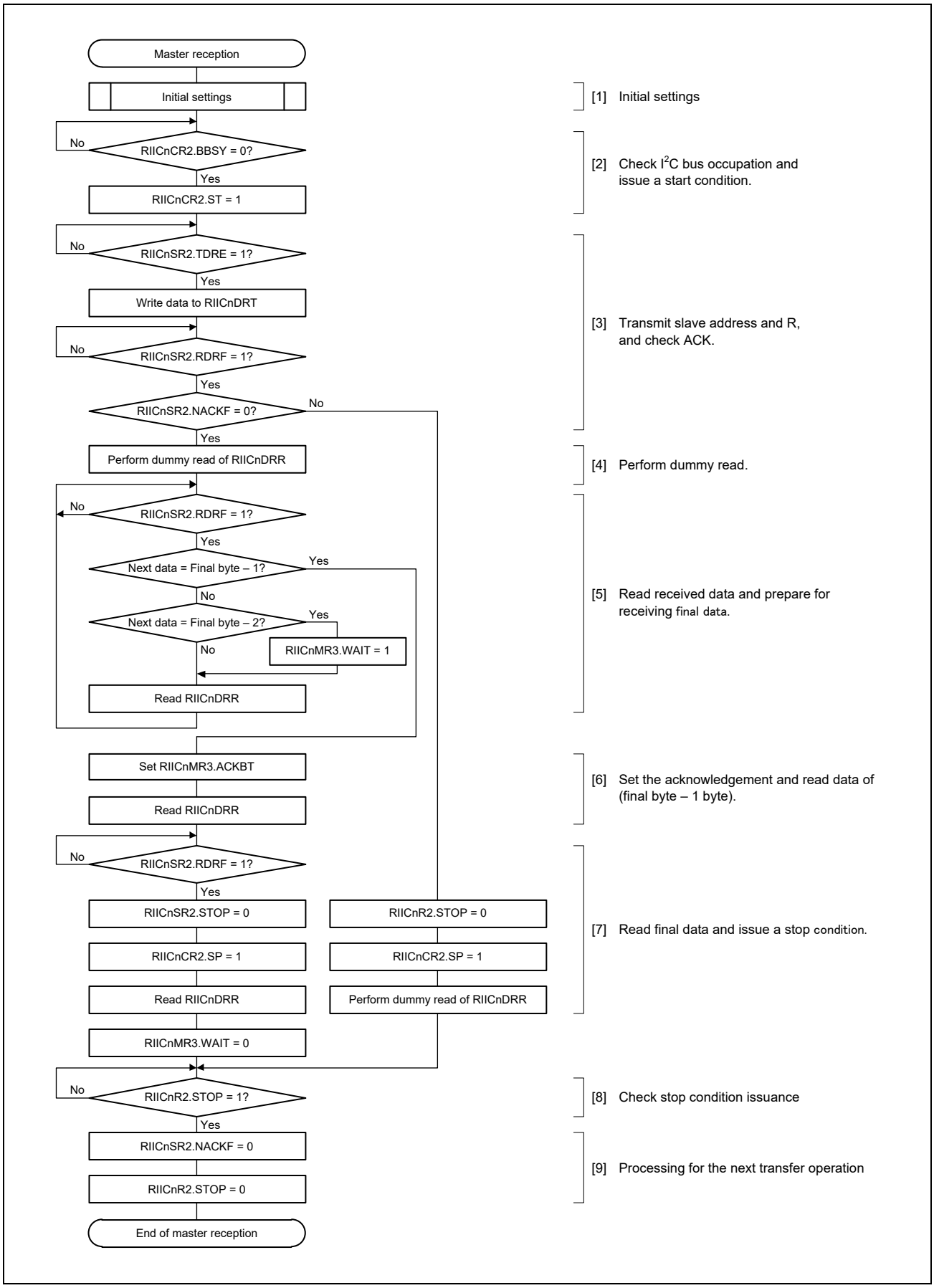


Figure 25.10 Example Flowchart for the Master Reception of 3 or More Bytes (7-Bit Address Format)

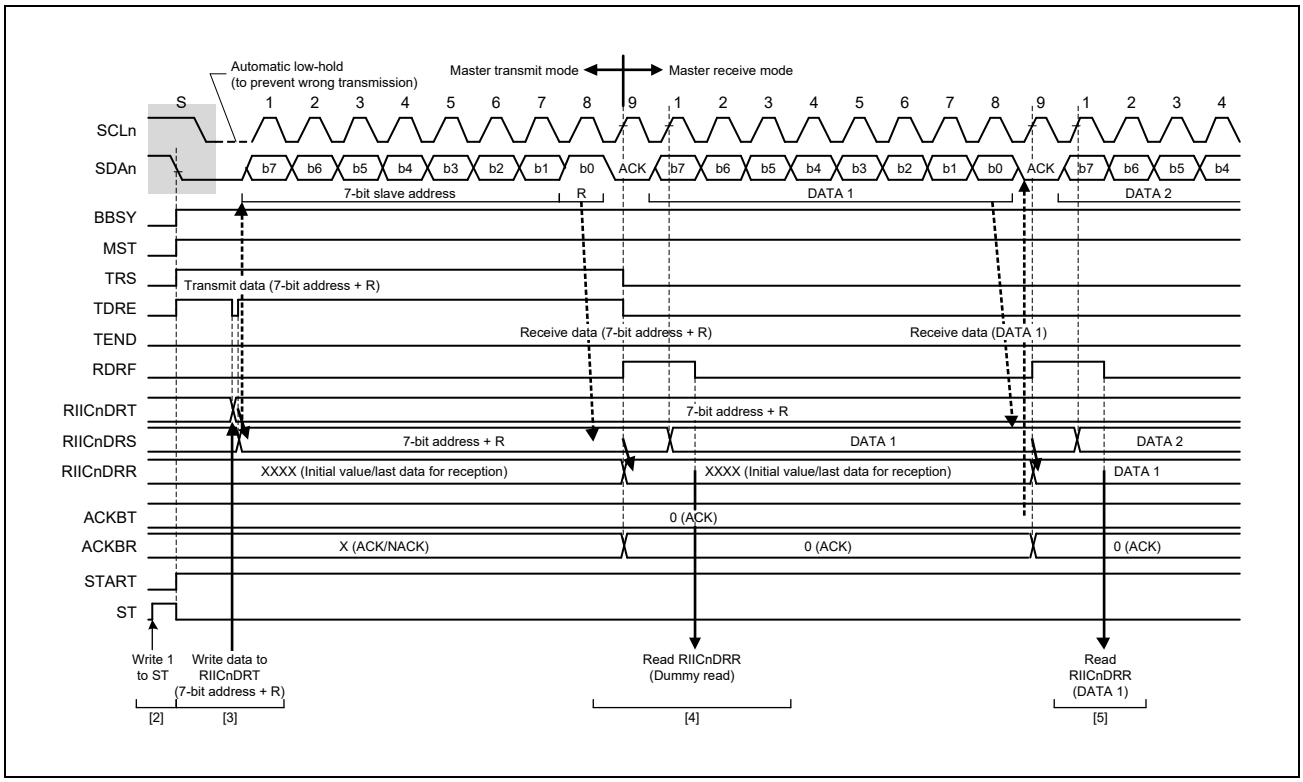


Figure 25.11 Master Receive Operation Timing (1) (7-Bit Address Format, when RDRFS = 0)

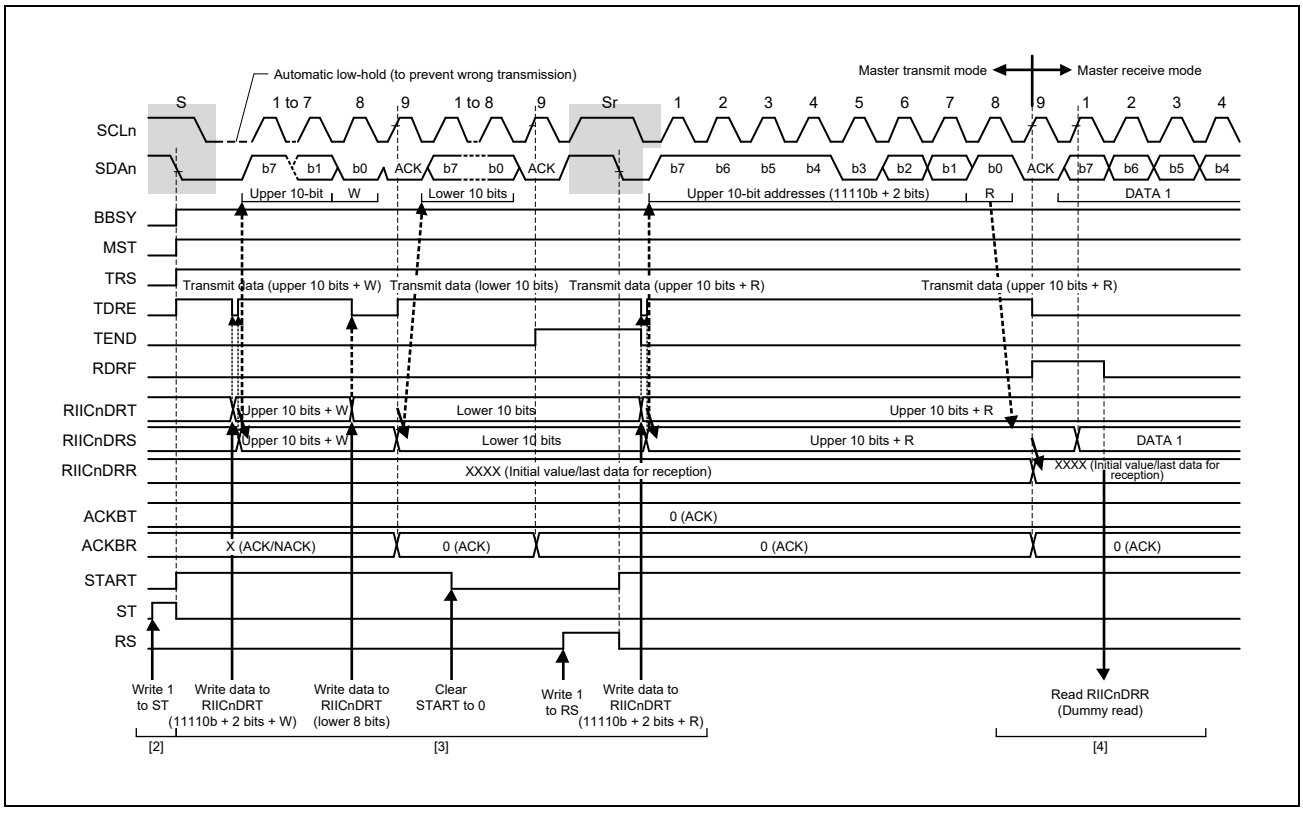


Figure 25.12 Master Receive Operation Timing (2) (10-Bit Address Format, when RDRFS = 0)

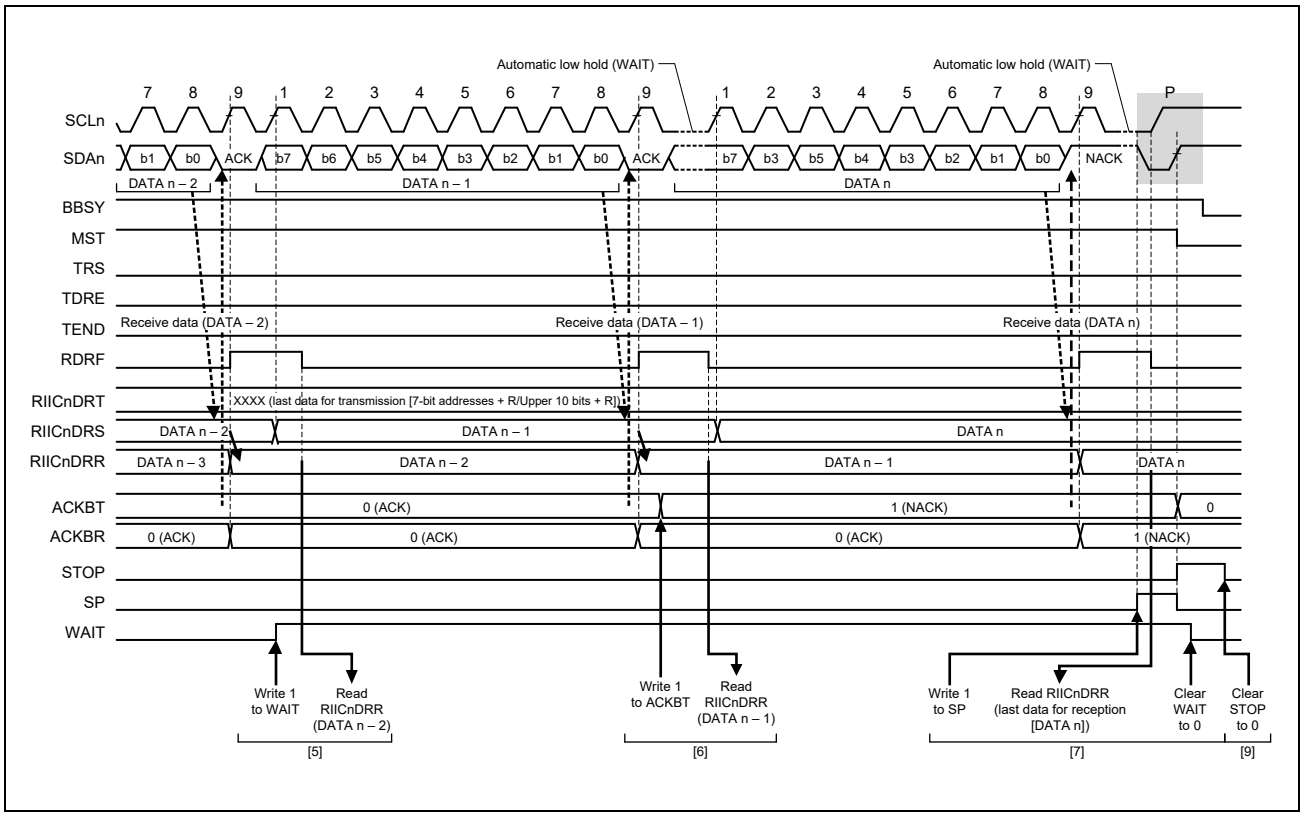


Figure 25.13 Master Receive Operation Timing (3) (when RDRFS = 0)

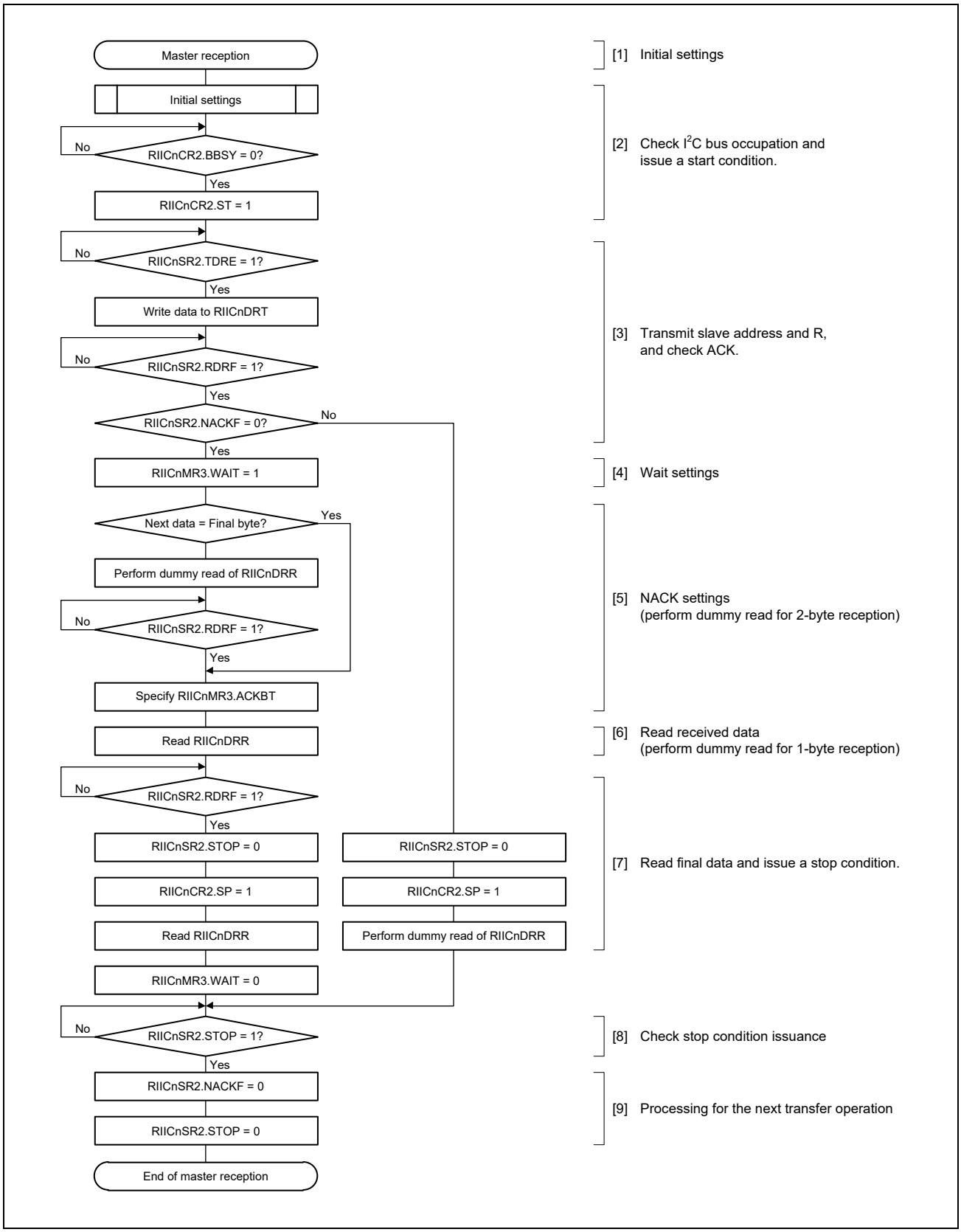


Figure 25.14 Example Flowchart for the Master Reception of 1 or 2 Bytes (7-Bit Address Format)

25.5.5 Slave Transmit Operation

In slave transmit operation, the master device outputs the SCL (clock) signal, the RIIC transmits data as a slave device, and the master device returns acknowledgements.

Figure 25.15 shows an example of usage of slave transmission and **Figure 25.16** and **Figure 25.17** show the timing of operations in slave transmission.

The following describes the procedure and operations for slave transmission.

(1) Follow the procedure in **Figure 25.5** to make initial settings for the RIIC. This step is not necessary if initialization of the RIIC has already been completed. After initial settings, the RIIC will stay in the standby state until it receives a slave address that it matches.

(2) After receiving a matching slave address, the RIIC sets one of the corresponding bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2) to 1 on the rising edge of the ninth cycle of SCL clock (the clock signal) and outputs the value set in the RIICnMR3.ACKBT bit to the acknowledge bit on the ninth cycle of SCL clock. If the value of the R/W# bit that was also received at this time is 1, the RIIC automatically places itself in slave transmit mode by setting both the RIICnCR2.TRS bit and the RIICnSR2.TDRE flag to 1.

(3) After the RIICnSR2.TEND flag is confirmed to be 1, write the data for transmission to the RIICnDRT register. At this time, if the RIIC receives no acknowledge from the master device (receives an NACK signal) while the RIICnFER.NACKE bit is 1, the RIIC suspends transfer of the next data.

(4) Wait unit the RIICnSR2.TEND flag is set to 1 while the RIICnSR2.TDRE flag is 1, after the RIICnSR2.NACKF flag is set to 1 or the last byte for transmission is written to the RIICnDRT register. When the RIICnSR2.NACKF flag or the TEND flag is 1, the RIIC drives the SCL line low on the ninth falling edge of SCL clock.

(5) When the RIICnSR2.NACKF flag or the RIICnSR2.TEND flag is 1, dummy read RIICnDRR to complete the processing. This releases the SCL line.

(6) Upon detecting the stop condition, the RIIC automatically clears bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2), flags RIICnSR2.TDRE and TEND, and the RIICnCR2.TRS bit to 0, and enters slave receive mode.

(7) After checking that the RIICnSR2.STOP flag is 1, set the RIICnSR2.NACKF and STOP flags to 0 for the next transfer operation.

CAUTION

Operations for transfer start if the RIICnSR2.NACKF flag is cleared to 0 before RIICnSR2.STOP is set to 1. Be sure to confirm that RIICnSR2.STOP is set to 1 before clearing RIICnSR2.NACKF to 0. In particular, when the NACK receive interrupt (INTRIICNAKI) is in use, take care not to clear the NACKF flag to 0 before the STOP flag is set to 1 during interrupt processing.

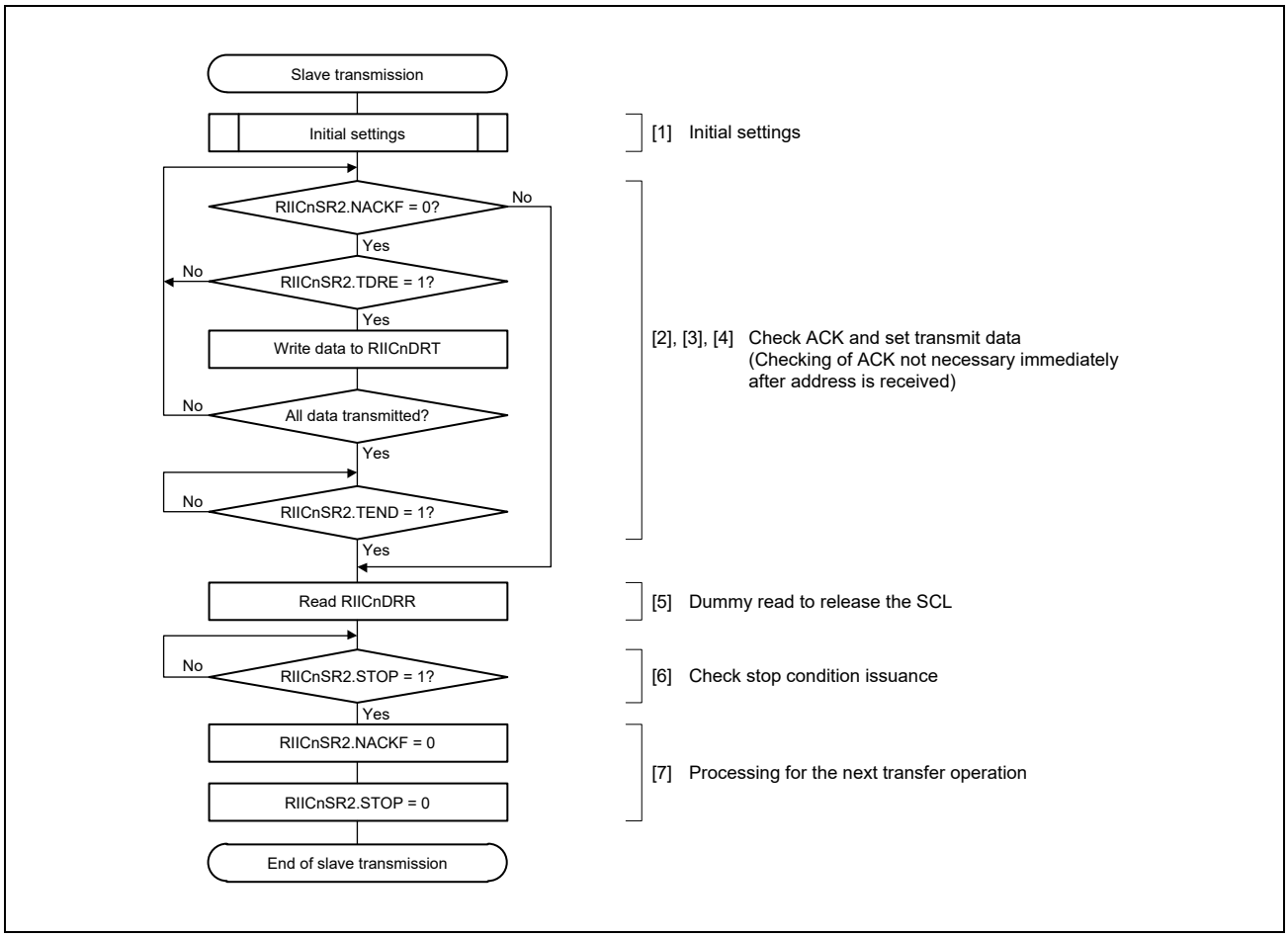


Figure 25.15 Example of Slave Transmission Flowchart

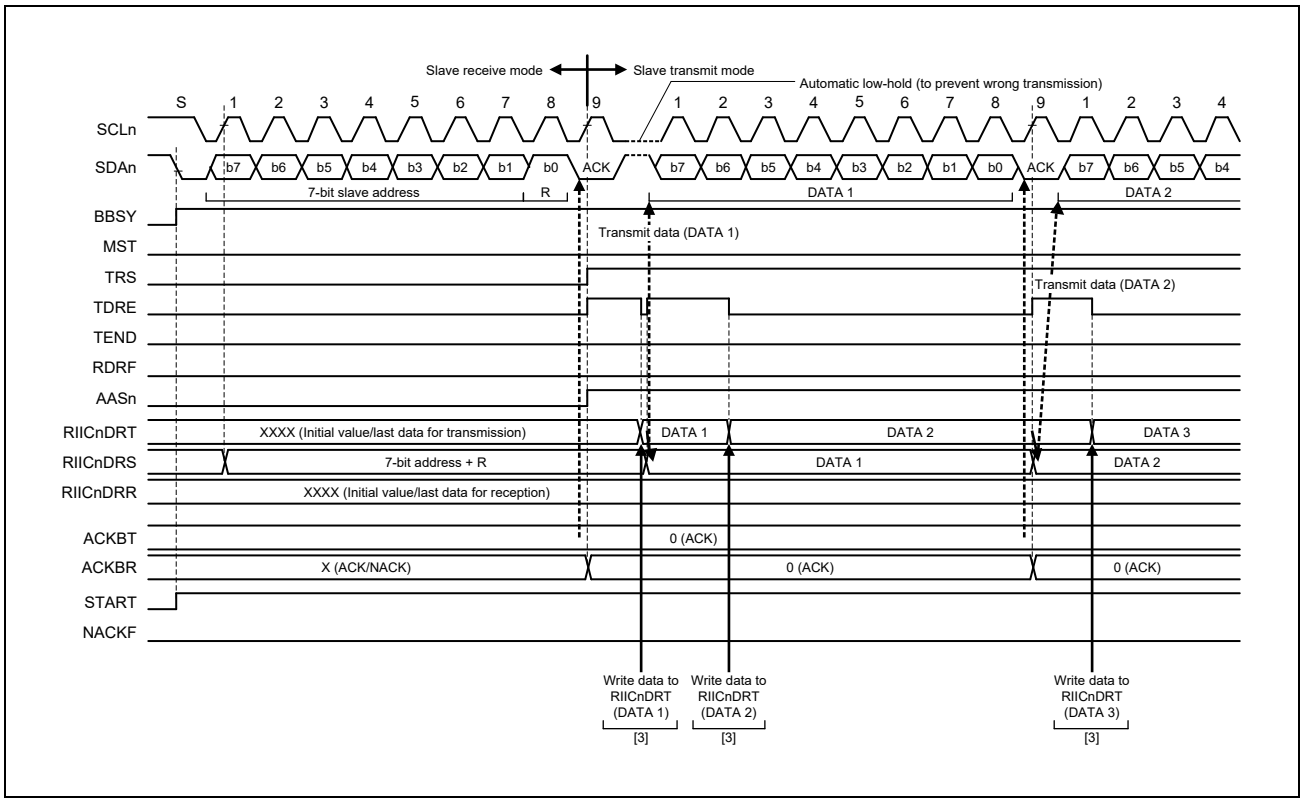


Figure 25.16 Slave Transmit Operation Timing (1) (7-Bit Address Format)

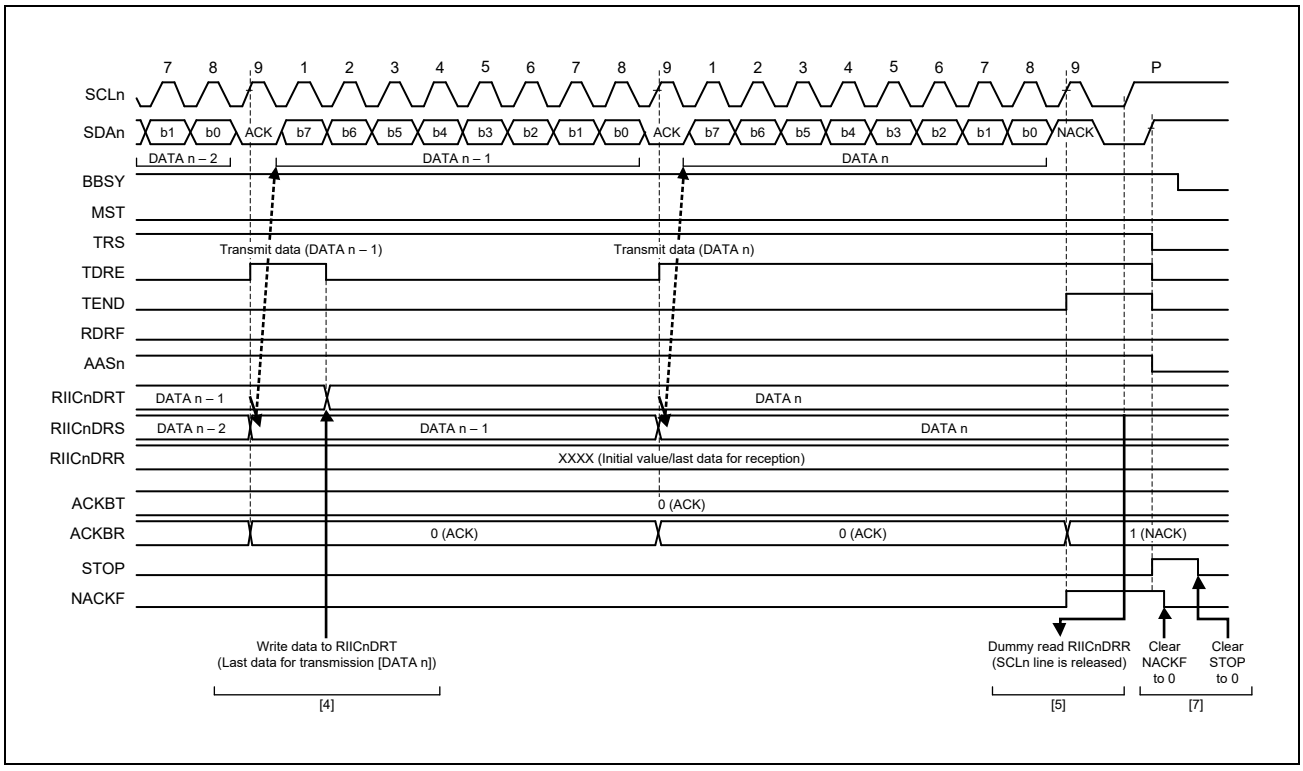


Figure 25.17 Slave Transmit Operation Timing (2)

25.5.6 Slave Receive Operation

In slave receive operation, the master device outputs the SCL clock and transmit data, and the RIIC returns acknowledgements as a slave device.

Figure 25.18 shows an example of usage of slave reception and **Figure 25.19** and **Figure 25.20** show the timing of operations in slave reception.

The following describes the procedure and operations for slave reception.

(1) Follow the procedure in **Figure 25.5** to make initial settings for the RIIC. This step is not necessary if initialization of the RIIC has already been completed. After initial settings, the RIIC will stay in the standby state until it receives a slave address that it matches.

(2) After receiving a matching slave address, the RIIC sets one of the corresponding bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2) to 1 on the rising edge of the ninth cycle of SCL clock (the clock signal) and outputs the value set in the RIICnMR3.ACKBT bit to the acknowledge bit on the ninth cycle of SCL clock. If the value of the R/W# bit that was also received at this time is 0, the RIIC continues to place itself in slave receive mode and sets the RIICnSR2.RDRF flag to 1.

(3) After the RIICnSR2.STOP flag is confirmed to be 0 and the RIICnSR2.RDRF flag to be 1, dummy read RIICnDRR (the dummy value consists of the slave address and R/W# bit when the 7-bit address format is selected, or the lower eight bits when the 10-bit address format is selected).

(4) When RIICnDRR is read, the RIIC automatically clears the RIICnSR2.RDRF flag to 0. If reading of RIICnDRR is delayed and a next byte is received while the RDRF flag is still set to 1, the RIIC holds the SCL line low from one SCL cycle before the timing with which RDRF should be set. In this case, reading RIICnDRR releases the SCL line from being held at the low level.

When the RIICnSR2.STOP flag is 1 and the RIICnSR2.RDRF flag is also 1, read RIICnDRR until all the data is completely received.

(5) Upon detecting the stop condition, the RIIC automatically clears bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2) to 0.

(6) After checking that the RIICnSR2.STOP flag is 1, clear the RIICnSR2.STOP flag to 0 for the next transfer operation.

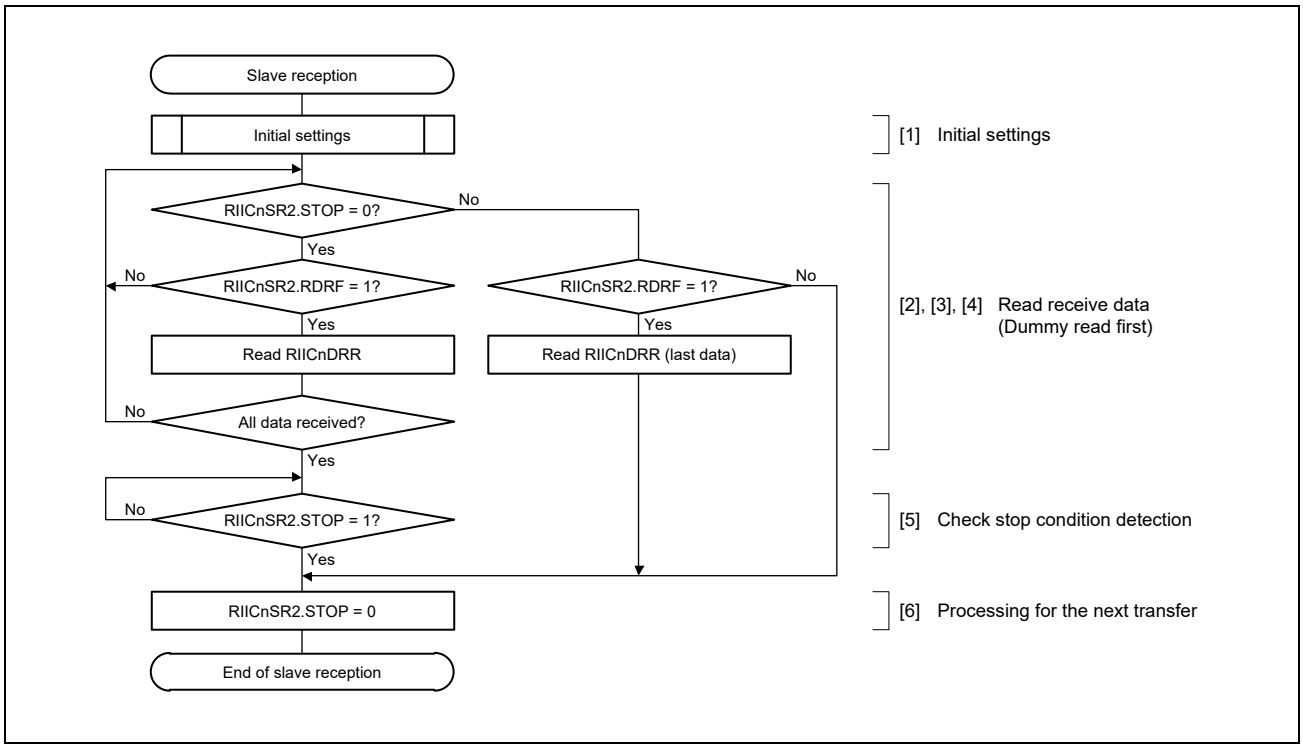


Figure 25.18 Example of Slave Reception Flowchart

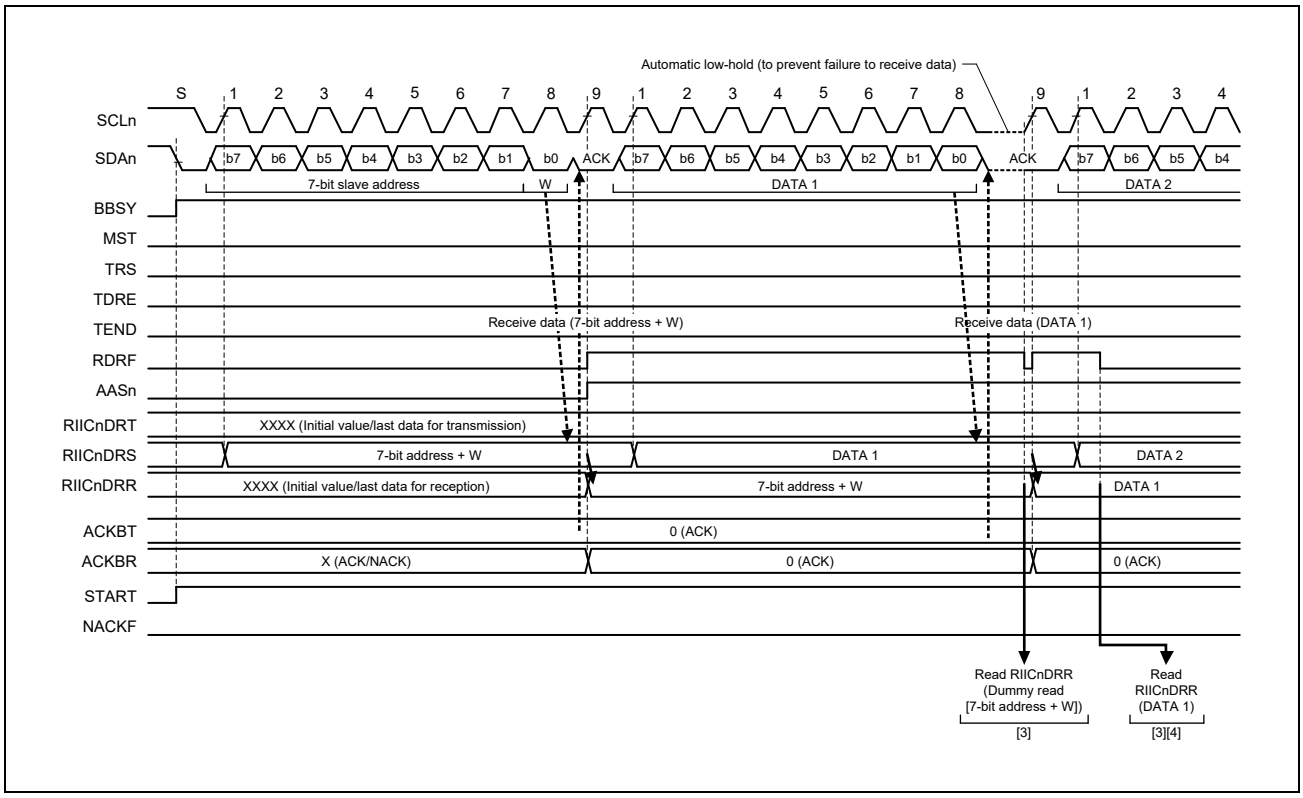


Figure 25.19 Slave Receive Operation Timing (1) (7-Bit Address Format, when RDRFS = 0)

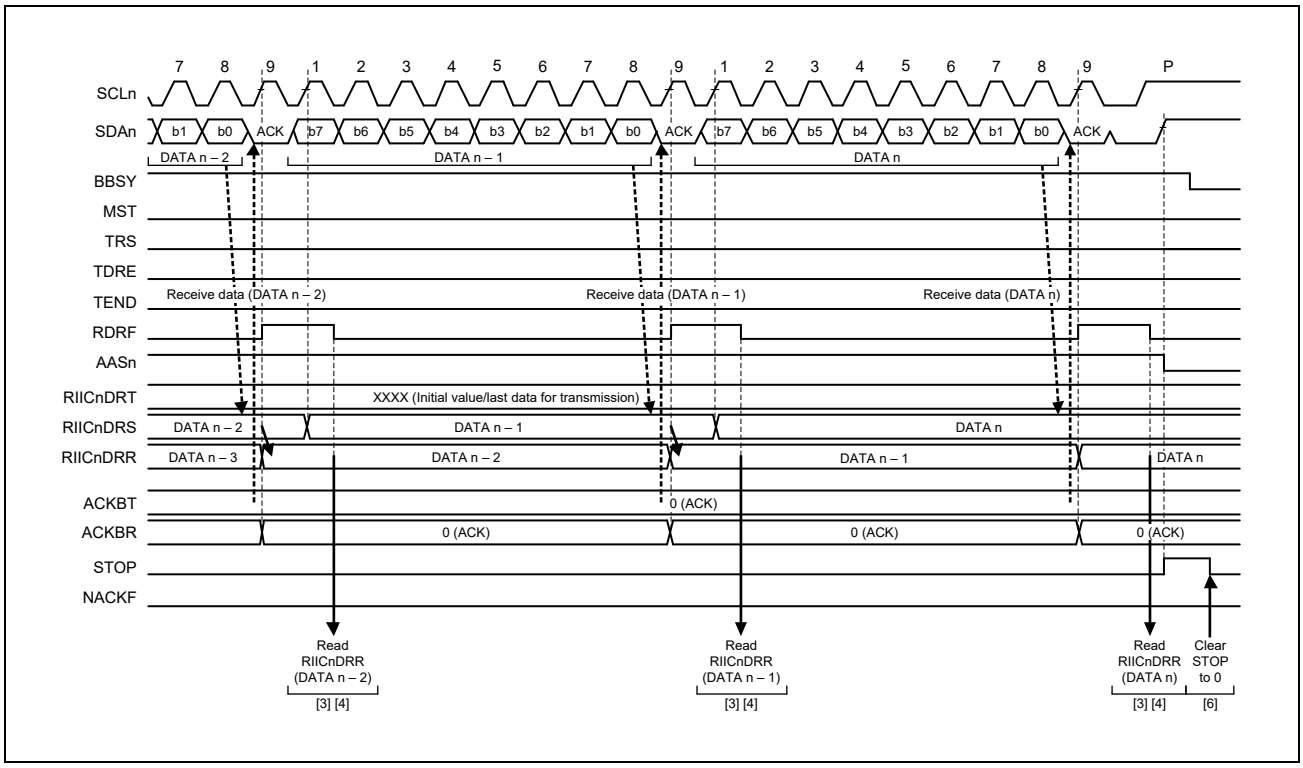


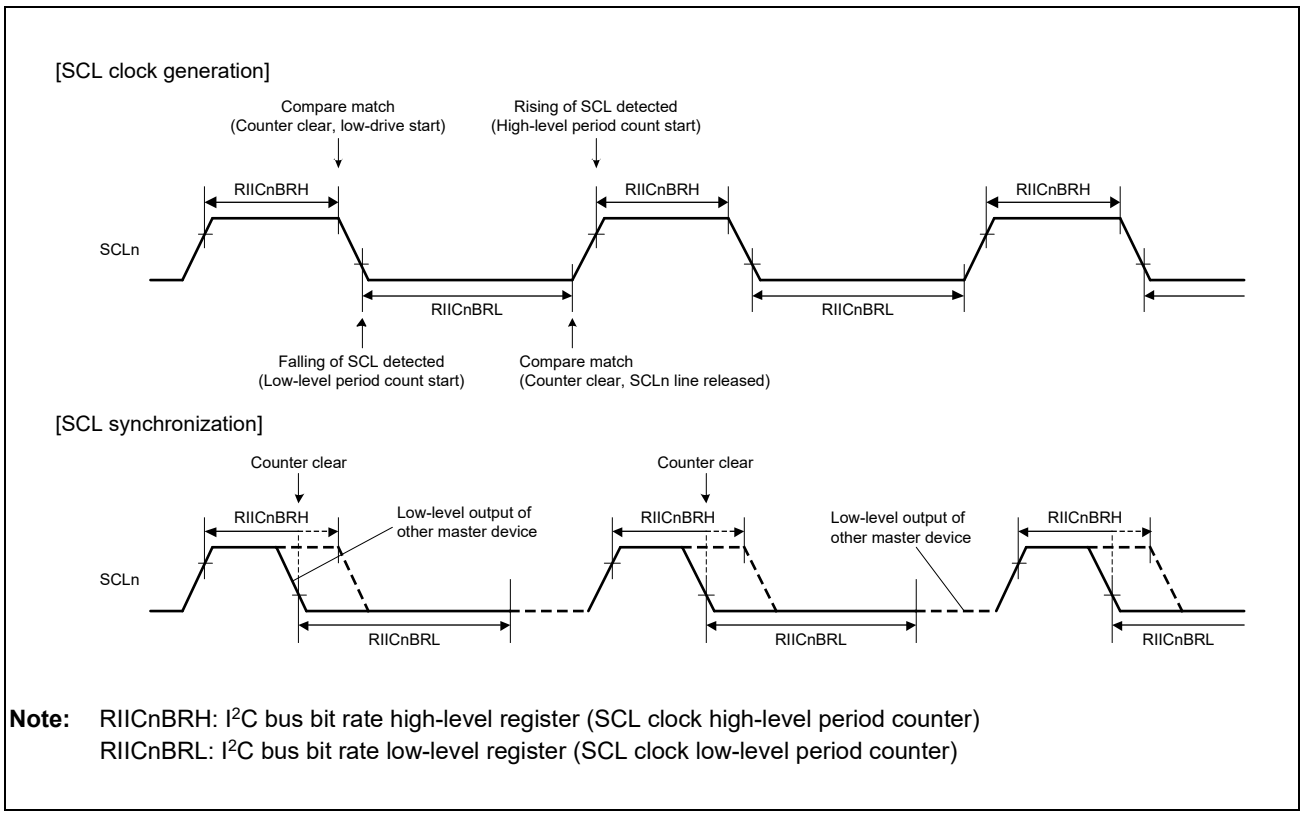
Figure 25.20 Slave Receive Operation Timing (2) (when RDRFS = 0)

25.6 SCL Synchronization Circuit

In generation of the SCL (clock) signal, the RIIC starts counting out the value for width at high level specified in RIICnBRH when it detects a rising edge on the SCL line and drives the SCL line low once counting of the width at high level is complete. When the RIIC detects the falling edge of the SCL line, it starts counting out the width at low level period specified in RIICnBRL, and then stops driving the SCL line (releases the line) once counting of the width at low level is complete. The SCL (clock) signal is thus generated.

If multiple master devices are connected to the I^2C bus, a collision of SCL signals may arise due to contention with another master device. In such cases, the master devices have to synchronize their SCL signals. Since this synchronization of SCL signals must be bit by bit, the RIIC is equipped with a facility (the SCL synchronization circuit) to obtain bit-by-bit synchronization of the SCL clock signals by monitoring the SCL line while in master mode.

When the RIIC has detected a rising edge on the SCL line and thus started counting out the width at high level specified in RIICnBRH, and the level on the SCL line falls because an SCL signal is being generated by another master device, the RIIC stops counting when it detects the falling edge, drives the level on the SCL line low, and starts counting out the width at low level specified in RIICnBRL. When the RIIC finishes counting out the width at low level, it stops driving the SCL line to the low level (i.e. releases the line). At this time, if the width at low level of the SCL clock signal from the other master device is longer than the width at low level set in the RIIC, the width at low level of the SCL signal will be extended. Once the width at low level for the other master device has ended, the SCL signal rises because the SCL line has been released. When the RIIC finishes outputting the low-level period of the SCL clock, the SCL line is released and the SCL clock rises. That is, in cases of contention of SCL signals from more than one master, the width at high level of the SCL signal is synchronized with that of the clock having the narrower width, and the width at low level of the SCL signal is synchronized with that of the clock having the broader width. However, such synchronization of the SCL signal is only enabled when the RIICnFER.SCLE bit is set to 1.



Note: RIICnBRH: I^2C bus bit rate high-level register (SCL clock high-level period counter)
RIICnBRL: I^2C bus bit rate low-level register (SCL clock low-level period counter)

Figure 25.21 Generation and Synchronization of the SCL Signal from the RIIC

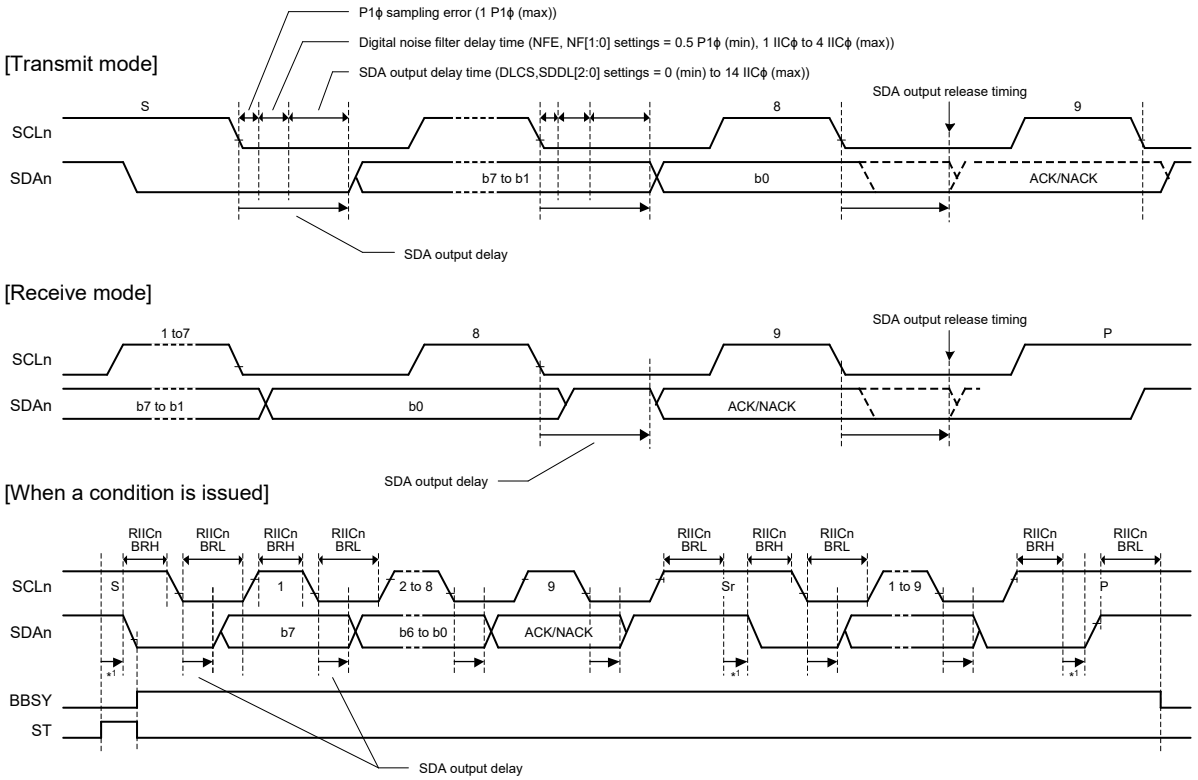
25.7 Facility for Delaying SDA Output

The RIIC module incorporates a facility for delaying output on the SDA line. The delay can be applied to all output (issuing of the start, restart, and stop conditions, data, and the ACK and NACK signals) on the SDA line.

With the SDA output delay facility, SDA output is delayed from detection of a falling edge of the SCL signal to ensure that the SDA signal is output within the interval over which the SCL (clock) signal is at the low level. Doing this leads to usage with the aim of preventing erroneous operation of communications devices, with the aim of satisfying the 300-ns (min.) data-hold time requirement of the SMBus specification.

The output delay facility is enabled by setting the RIICnMR2.SDDL[2:0] bits to any value other than 000b, and disabled by setting the same bits to 000b.

While the SDA output delay facility is enabled (i.e. while the SDDL[2:0] bits in RIICnMR2 are set to any value other than 000b), the RIICnMR2.DLCS bit selects the clock source for counting by the SDA output delay counter as the internal base clock (IICϕ) for the RIIC module or as a clock signal derived by dividing the frequency of the internal base clock by two (IICϕ/2). The counter counts the number of cycles set in the SDDL[2:0] bits in RIICnMR2. After counting of the set number of cycles of delay is completed, the RIIC module places the required output (start, restart, or stop condition, data, or an ACK or NACK signal) on the SDA line.



Note 1. The output is delayed by the number of cycles set by the SDDL [2:0] bits when a start (S), restart (Sr), or stop (P) condition is issued.

Figure 25.22 SDA Output Delay Facility

25.8 Digital Noise-Filter Circuits

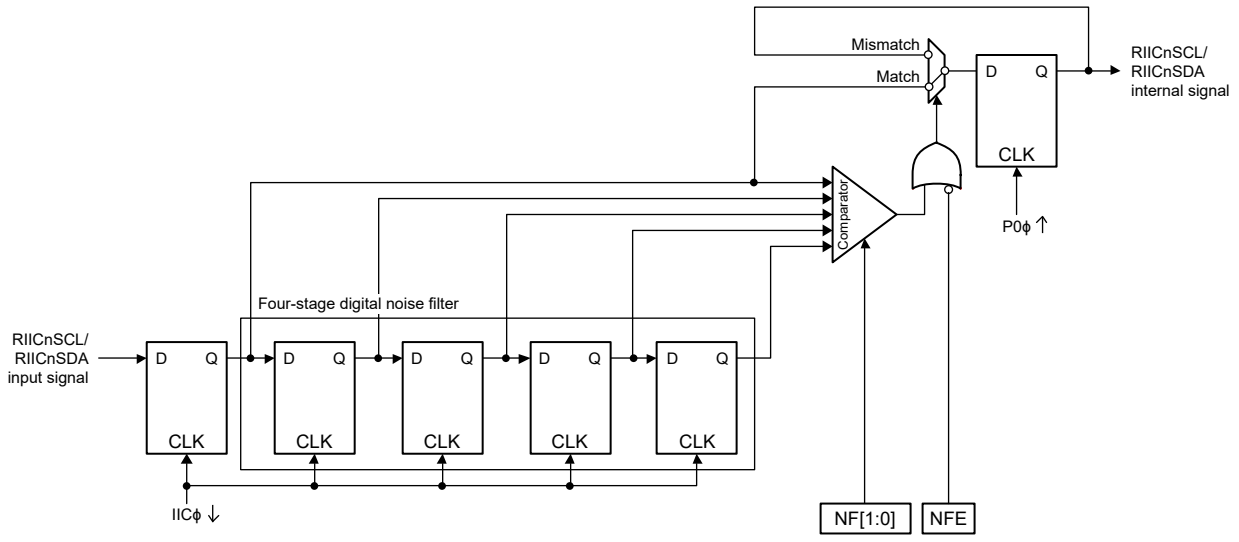
Figure 25.23 is a block diagram of the digital noise-filter circuit. When the NFE bit in the RIICnFER register is set to 1, input to the RIICnSCL and RIICnSDA pins are conveyed to the internal circuitry through digital noise-filter circuits.

The on-chip digital noise-filter circuit of the RIIC consists of four flip-flop circuit stages connected in series and a match- detection circuit.

The number of effective stages in the digital noise filter is selected by the RIICnMR3.NF[1:0] bits. The selected number of effective stages determines the noise-filtering capability as a period from one to four IICϕ cycles.

The input signal to the RIICnSCL pin (or RIICnSDA pin) is sampled on falling edges of the IICϕ signal. When the input signal level matches the output level of the number of effective flip-flop circuit stages as selected by the RIICnMR3.NF[1:0] bits, the signal level is conveyed to the subsequent stage. If the signal levels do not match, the previous value is retained.

If the ratio between P0ϕ and IICϕ is small when the RIICnMR1.CKS[2:0] bits are set to 000b, note that the characteristics of the digital noise filter may lead to the elimination of needed signals as noise.



Note: NFE: Digital noise filter circuit enable bit
NF[1:0]: Digital noise filter stage selection bits

Figure 25.23 Block Diagram of Digital Noise Filter Circuit

25.9 Address Match Detection

The RIIC can set three unique slave addresses in addition to the general call address and host address, and also can set 7- bit or 10-bit slave addresses.

25.9.1 Slave-Address Match Detection

The RIIC can set three unique slave addresses, and has a slave address detection function for each unique slave address. When the RIICnSER.SARyE bit (y = 0 to 2) is set to 1, the slave addresses set in RIICnSAR(L,U)y (y = 0 to 2) can be detected. When the RIIC detects a match of the set slave address, the corresponding RIICnSR1.AASy flag (y = 0 to 2) is set to 1 at the rising edge of the ninth SCL clock cycle, and the RIICnSR2.RDRF flag or the RIICnSR2.TDRE flag is set to 1 by the following R/W# bit. This causes a receive data full interrupt (INTRIICRI) or transmit data empty interrupt (INTRIICTI) to be generated. The AASy flag is used to identify which slave address has been specified.

Figure 25.24 to **Figure 25.26** show the AASy flag set timing in three cases.

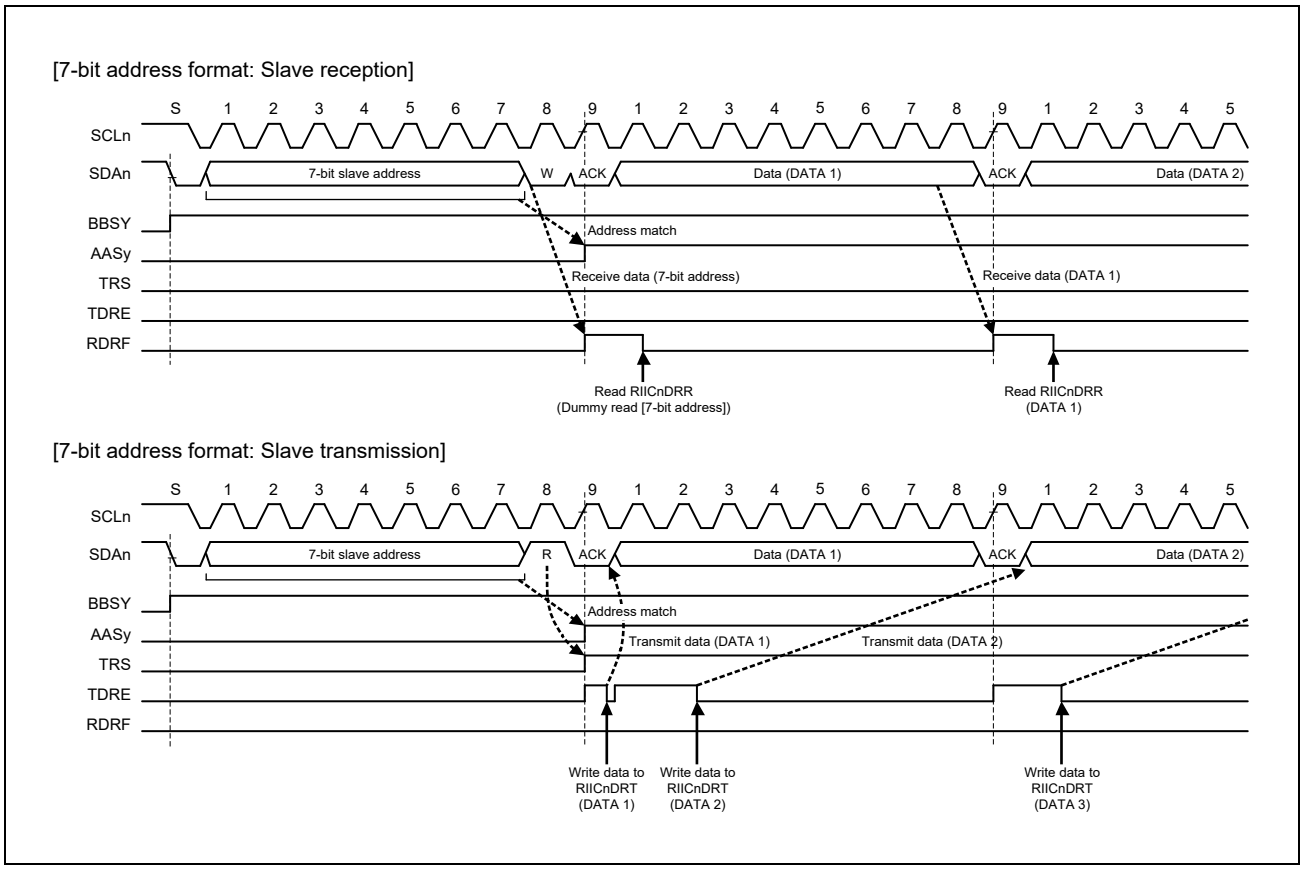


Figure 25.24 AASy Flag Set Timing with 7-Bit Address Format Selected

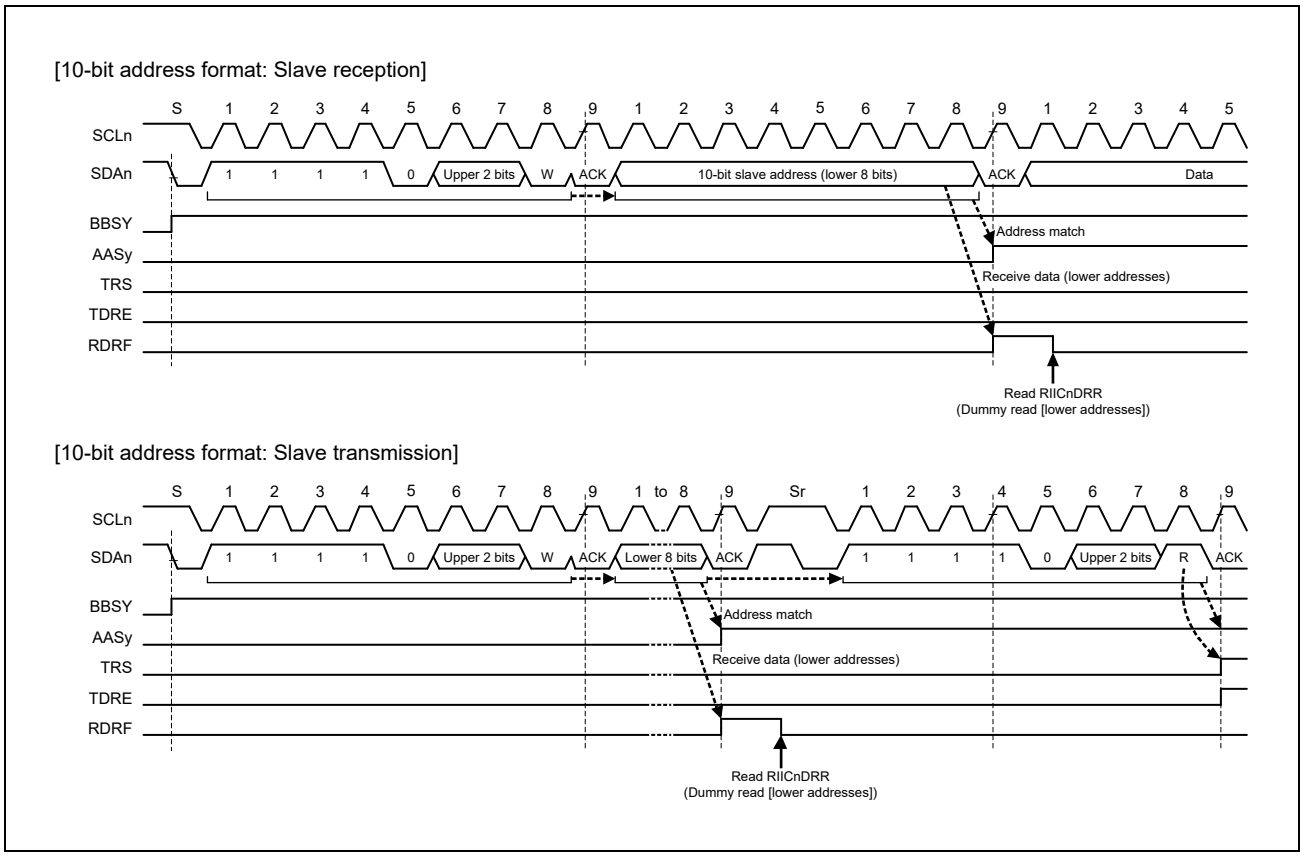
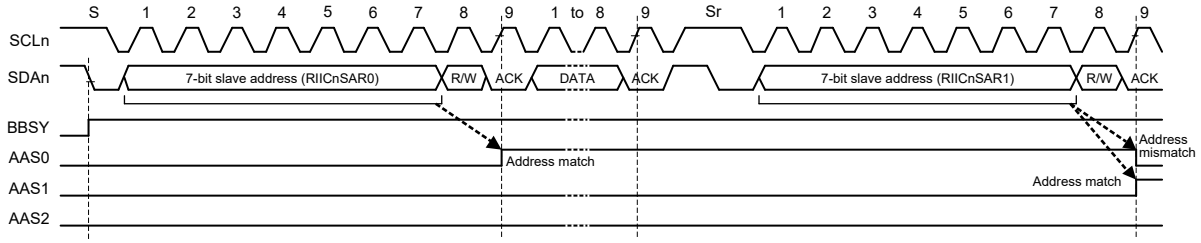
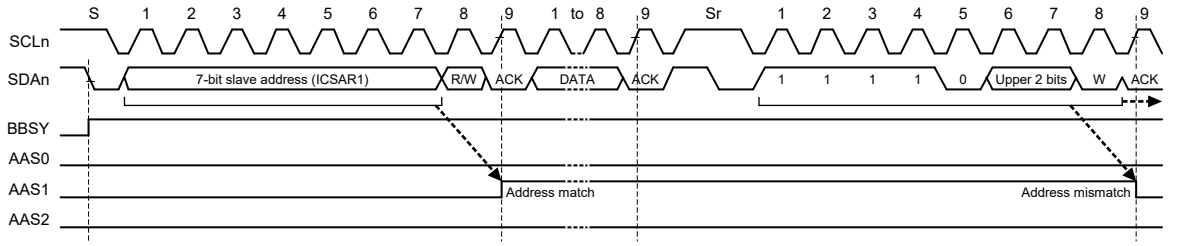


Figure 25.25 AASy Flag Set Timing with 10-Bit Address Format Selected

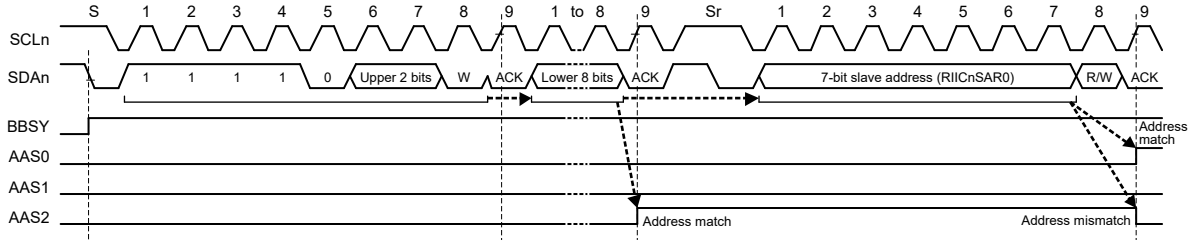
[In the case of RIICnSAR0: 7-bit address, RIICnSAR1: 7-bit address, RIICnSAR2: 10-bit address (1)]



[In the case of RIICnSAR0: 7-bit address, RIICnSAR1: 7-bit address, RIICnSAR2: 10-bit address (2)]



[In the case of RIICnSAR0: 7-bit address, RIICnSAR1: 7-bit address, RIICnSAR2: 10-bit address (3)]



Note: RIICnSARm (m = 0 to 2) indicates RIICnSARLm and RIICnSARUm (m = 0 to 2).

Figure 25.26 AASy Flag Set/Clear Timing with 7-Bit/10-Bit Address Formats Mixed

25.9.2 Detection of the General Call Address

The RIIC has a facility for detecting the general call address (0000_000b + 0 [W]). This is enabled by setting the RIICnSER.GCE bit to 1.

If the address received after a start or restart condition is issued is 0000_000b + 1[R] (start byte), the RIIC recognizes this as the address of a slave device with an “all-zero” address but not as the general call address.

When the RIIC detects the general call address, both the RIICnSR1.GCA flag and the RIICnSR2.RDRF flag are set to 1 on the rising edge of the ninth cycle of SCL clock. This leads to the generation of a receive data full interrupt (INTRIICRI). The value of the GCA flag can be confirmed to recognize that the general call address has been transmitted.

Operation after detection of the general call address is the same as normal slave receive operation.

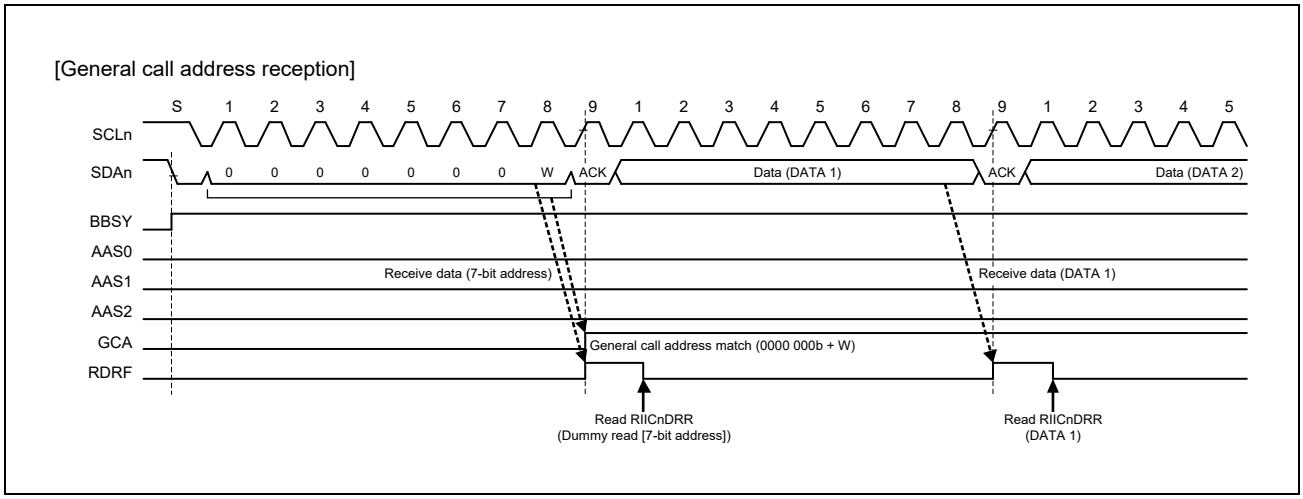


Figure 25.27 Timing of GCA Flag Setting during Reception of General Call Address

25.9.3 Device-ID Address Detection

The RIIC module has a facility for detecting device-ID addresses conformant with the I^2C bus specification (Rev. 03). When the RIIC receives 1111_100b as the first byte after a start condition or restart condition was issued with the RIICnSER.DIDE bit set to 1, the RIIC recognizes the address as a device ID, sets the RIICnSR1.DID flag to 1 on the rising edge of the ninth SCL clock cycle when the following R/W# bit is 0, and then compares the second and subsequent bytes with its own slave address. If the address matches the value in the slave address register, the RIIC sets the corresponding RIICnSR1.AASy flag (y = 0 to 2) to 1.

After that, when the first byte received after a start or restart condition is issued matches the device ID address (1111_100b) again and the following R/W# bit is 1, the RIIC does not compare the second and subsequent bytes and sets the RIICnSR2.TDRE flag to 1.

In the device-ID address detection function, the RIIC clears the DID flag to 0 if a match with the RIIC's own slave address is not obtained or a match with the device ID address is not obtained after a match with the RIIC's own slave address and the detection of a restart condition. If the first byte after detection of a start or restart condition matches the device ID address (1111_100b) and the R/W# bit is 0, the RIIC sets the DID flag to 1 and compares the second and subsequent bytes with the RIIC's slave address. If the R/W# bit is 1, the DID flag holds the previous value and the RIIC does not compare the second and subsequent bytes. Therefore, the reception of a device-ID address can be checked by reading the DID flag after confirming that TDRE = 1.

Furthermore, prepare the device-ID fields (three bytes: 12 bits indicating the manufacturer + 9 bits identifying the part + 3 bits indicating the revision) that must be sent to the host after reception of a continuous device-ID field as normal data for transmission. For details, see I^2C Bus Standard from NXP Semiconductors.

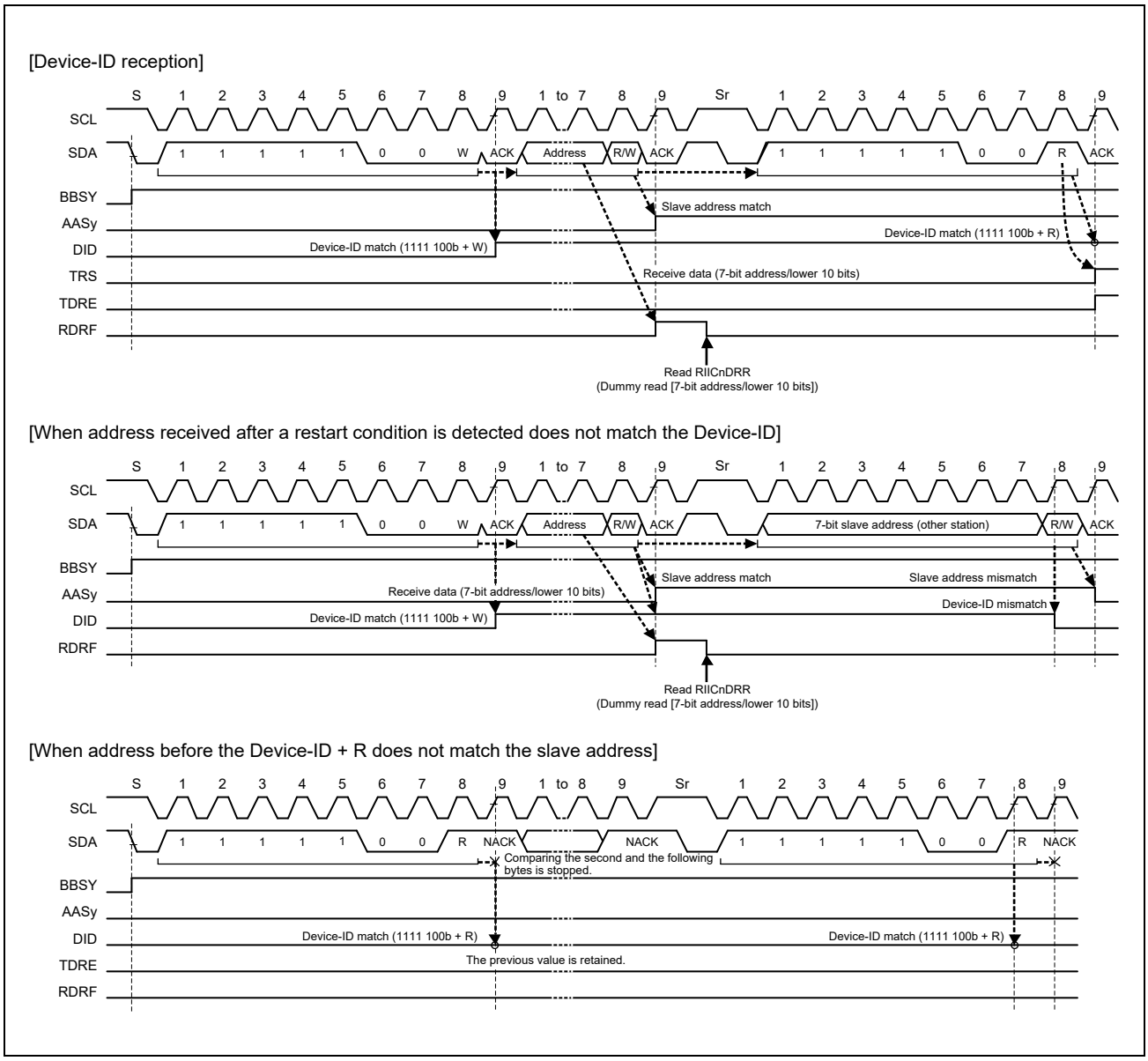


Figure 25.28 AASy/DID Flag Set/Clear Timing during Reception of Device-ID

25.9.4 Host Address Detection

The RIIC has a function to detect the host address while the SMBus is operating. When the RIICnSER.HOAE bit is set to 1 while the RIICnMR3.SMBS bit is 1, the RIIC can detect the host address (0001_000b) in slave receive mode (RIICnCR2.MST and TRS bits = 00b).

When the RIIC detects the host address, the RIICnSR1.HOA flag is set to 1 at the rising edge of the ninth SCL clock cycle, and at the same time, the RIICnSR2.RDRF flag is set to 1 when the R/W# bit is 0 (Wr bit). This causes a receive data full interrupt (INTRIICRI) to be generated. The HOA flag is used to recognize that the host address was sent from the smart battery or other devices.

If the bit following the host address (0001_000b) is an Rd bit (R/W# bit = 1), the RIIC can also detect the host address. After the host address is detected, the RIIC operates in the same manner as normal slave operation.

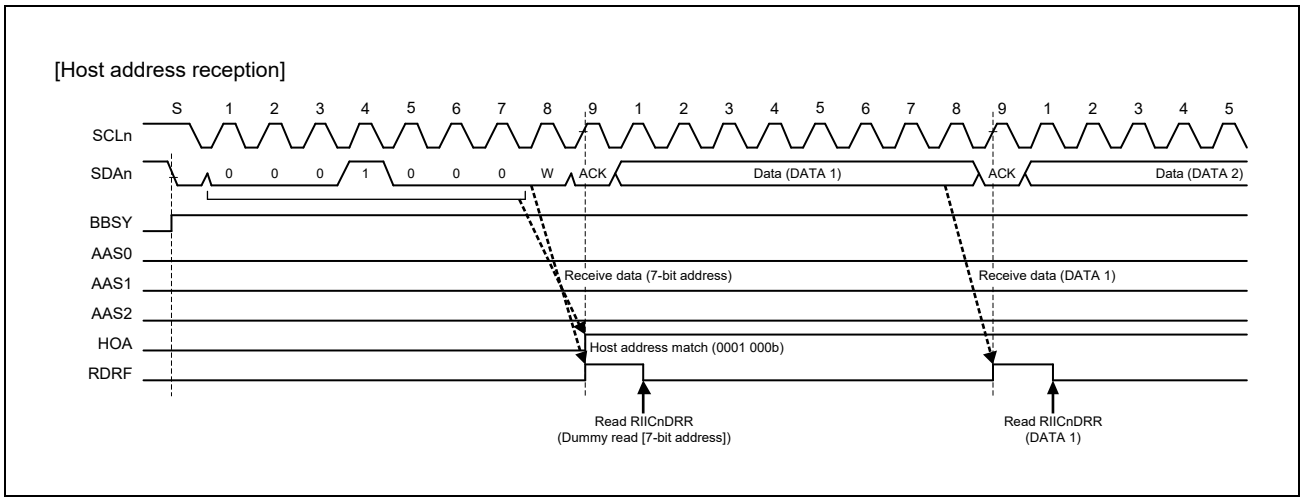


Figure 25.29 HOA Flag Set Timing during Reception of Host Address

25.10 Automatic Low-Hold Function for SCL

25.10.1 Function to Prevent Wrong Transmission of Transmit Data

If the shift register (RIICnDRS) is empty when data have not been written to the transmit data register (RIICnDRT) with the RIIC in transmission mode (RIICnCR2.TRS bit = 1), the SCL signal is automatically held at the low level over the intervals shown below. This low-hold period is extended until data for transmission have been written, which prevents the unintended transmission of erroneous data.

<Master transmit mode>

- Low-level interval after a start condition or restart condition is issued
- Low-level interval between the ninth clock cycle of one transfer and the first clock cycle of the next

<Slave transmit mode>

- Low-level interval between the ninth clock cycle of one transfer and the first clock cycle of the next

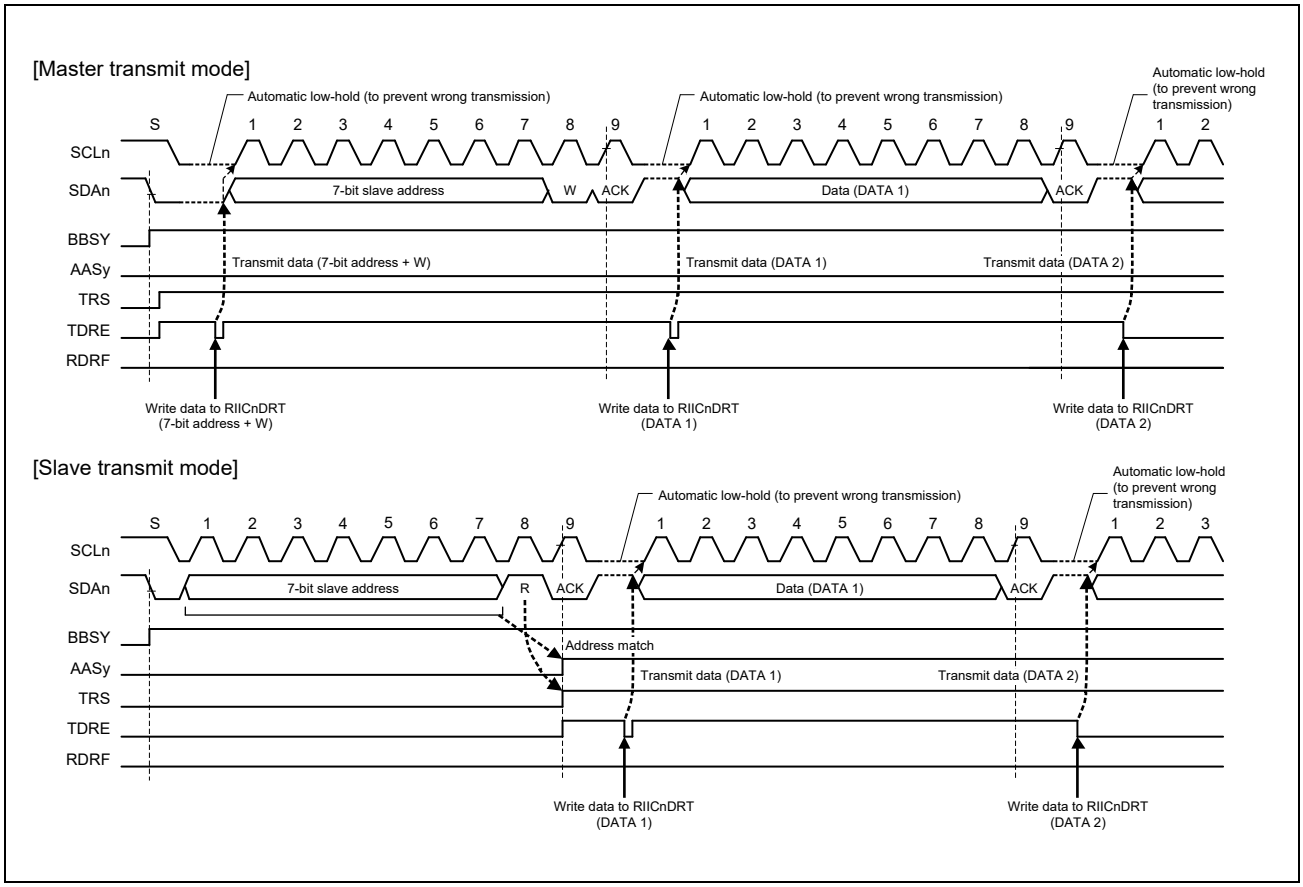


Figure 25.30 Automatic Low-Hold Operation in Transmit Mode

25.10.2 NACK Reception Transfer Suspension Function

The RIIC has a function to suspend transfer operation when NACK is received in transmit mode (RIICnCR2.TRS bit = 1). This function is enabled when the RIICnFER.NACKE bit is set to 1 (transfer suspension enabled). If the next transmit data has already been written (RIICnSR2.TDRE flag = 0) when NACK is received, next data transmission at the falling edge of the ninth SCL clock cycle is automatically suspended. This prevents the SDA line output level from being held low when the MSB of the next transmit data is 0.

If the transfer operation is suspended by this function (RIICnSR2.NACKF flag = 1), transmit operation and receive operation are discontinued. To restore transmit/receive operation, be sure to clear the NACKF flag to 0. In master transmit mode, clear the NACKF flag to 0 after issuing a restart condition or clear the NACKF and STOP flags to 0 after confirming that a stop condition has been issued, and then issue a start condition.

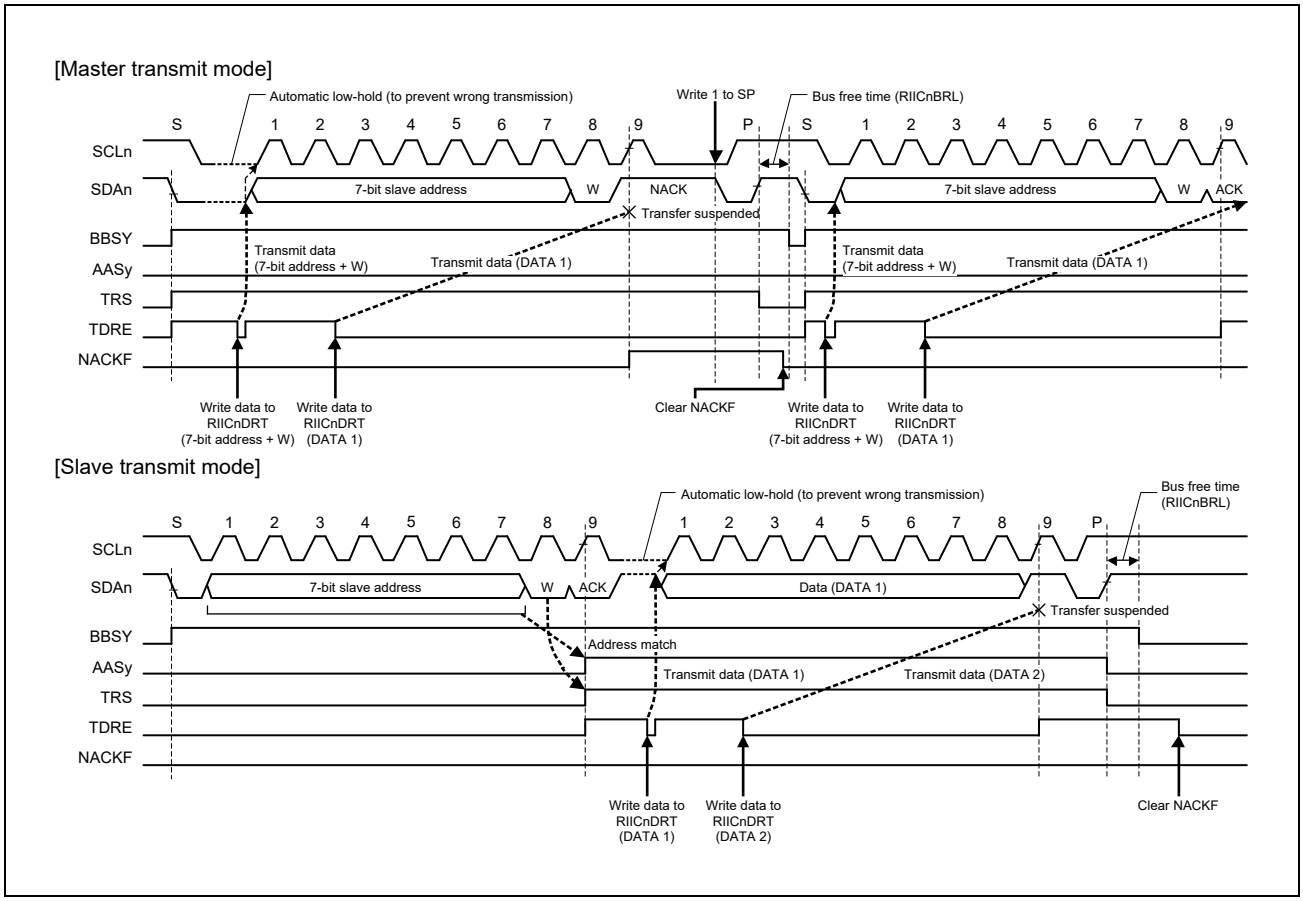


Figure 25.31 Suspension of Data Transfer when NACK is Received (NACKE = 1)

25.10.3 Function to Prevent Failure to Receive Data

If response processing is delayed when receive data (RIICnDRR) read is delayed for a period of one transfer frame or more with receive data full (RIICnSR2.RDRF flag = 1) in receive mode (RIICnCR2.TRS = 0), the RIIC holds the SCL line low automatically immediately before the next data is received to prevent failure to receive data.

This function to prevent failure to receive data using the automatic low-hold function is also enabled even if the read processing of the final receive data is delayed and, in the meantime, the RIIC's own slave address is designated after a stop condition is issued. This function does not disturb other communication because the RIIC does not hold the SCL line low when a mismatch with its own slave address occurs after a stop condition is issued.

Sections in which the SCL line is held low can be selected with a combination of the RIICnMR3.WAIT and RDRFS bits.

(1) One-Byte Receive Operation and Automatic Low-Hold Function Using the WAIT Bit

When the RIICnMR3.WAIT bit is set to 1, the RIIC performs one-byte receive operation using the WAIT bit function. Furthermore, when the RIICnMR3.RDRFS bit is 0, the RIIC automatically sends the RIICnMR3.ACKBT bit value for the acknowledge bit in the period from the falling edge of the eighth SCL clock cycle to the falling edge of the ninth SCL clock cycle, and automatically holds the SCL line low at the falling edge of the ninth SCL clock cycle using the WAIT bit function. This low-hold is released by reading data from RIICnDRR, which enables bytewise receive operation.

The WAIT bit function is enabled for receive frames after a match with the RIIC's own slave address (including the general call address and host address) is obtained in master receive mode or slave receive mode.

(2) One-Byte Receive Operation (ACK/NACK Transmission Control) and Automatic Low-Hold Function Using the RDRFS Bit

When the RIICnMR3.RDRFS bit is set to 1, the RIIC performs one-byte receive operation using the RDRFS bit function. When the RIICnSR2.RDRFS bit is set to 1, the RDRF flag (receive data full) in RIICnSR2 is set to 1 at the rising edge of the eighth SCL clock cycle, and the SCL line is automatically held low at the falling edge of the eighth SCL clock cycle. This low hold is released by writing a value to the RIICnMR3.ACKBT bit, but cannot be released by reading data from RIICnDRR, which enables receive operation by the ACK/NACK transmission control according to the data received in byte units.

The RDRFS bit function is enabled for receive frames after a match with the RIIC's own slave address (including the general call address and host address) is obtained in master receive mode or slave receive mode.

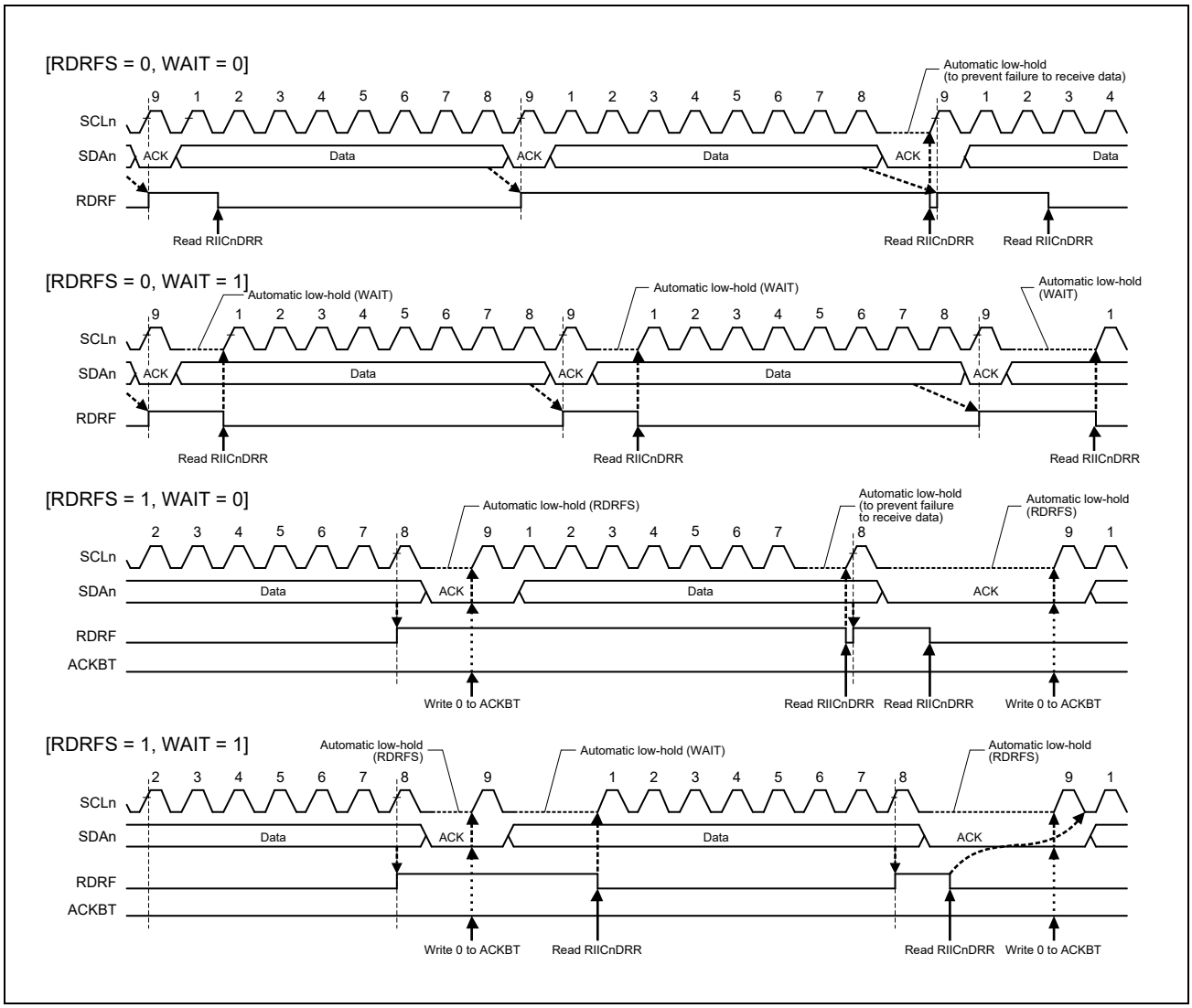


Figure 25.32 Automatic Low-Hold Operation in Receive Mode (Using RDRFS and WAIT Bits)

25.11 Arbitration-Lost Detection Functions

In addition to the normal arbitration-lost detection function defined by the I^2C bus standard, the RIIC has functions to prevent double-issue of a start condition, to detect arbitration-lost during transmission of NACK, and to detect arbitration-lost in slave transmit mode.

25.11.1 Master Arbitration-Lost Detection (MALE Bit)

The RIIC drives the SDA line low to issue a start condition. However, if the SDA line has already been driven low by another master device issuing a start condition, the RIIC regards its own issuing of a start condition as an error and considers this a loss in arbitration, so priority is given to transfer by the other master device. Similarly, if a request to issue a start condition is made by setting the RIICnCR2.ST bit to 1 while the bus is busy (RIICnCR2.BBSY flag = 1), the RIIC regards this as a double-issuing-of-start-condition error and considers itself to have lost in arbitration, thus preventing a failure of transfer due to issuing of a start condition while transfer is in progress.

When a start condition is issued successfully, if the data for transmission including the address bits (i.e. the internal SDA output level) and the level on the SDA line do not match (the high output as the internal SDA output; i.e. the SDA pin is in the high-impedance state) and the low level is detected on the SDA line, the RIIC loses in arbitration.

After a loss in arbitration of mastership, the RIIC immediately enters slave receive mode. If a slave address (including the general call address) matches its own address at this time, the RIIC continues in slave operation.

A loss in arbitration of mastership is detected when the following conditions are met while the RIICnFER.MALE bit is 1 (master arbitration-lost detection enabled).

[Master arbitration-lost conditions]

- Non-matching of the internal level for output on SDA and the level on the SDA line after a start condition was issued by setting the RIICnCR2.ST bit to 1 while the RIICnCR2.BBSY flag was cleared to 0 (erroneous issuing of a start condition)
- Setting of the RIICnCR2.ST bit to 1 (start condition double-issue error) while the RIICnCR2.BBSY flag is set to 1
- When the transmit data excluding acknowledge (internal SDA output level) does not match the level on the SDA line in master transmit mode (RIICnCR2.MST and TRS bits = 11b)

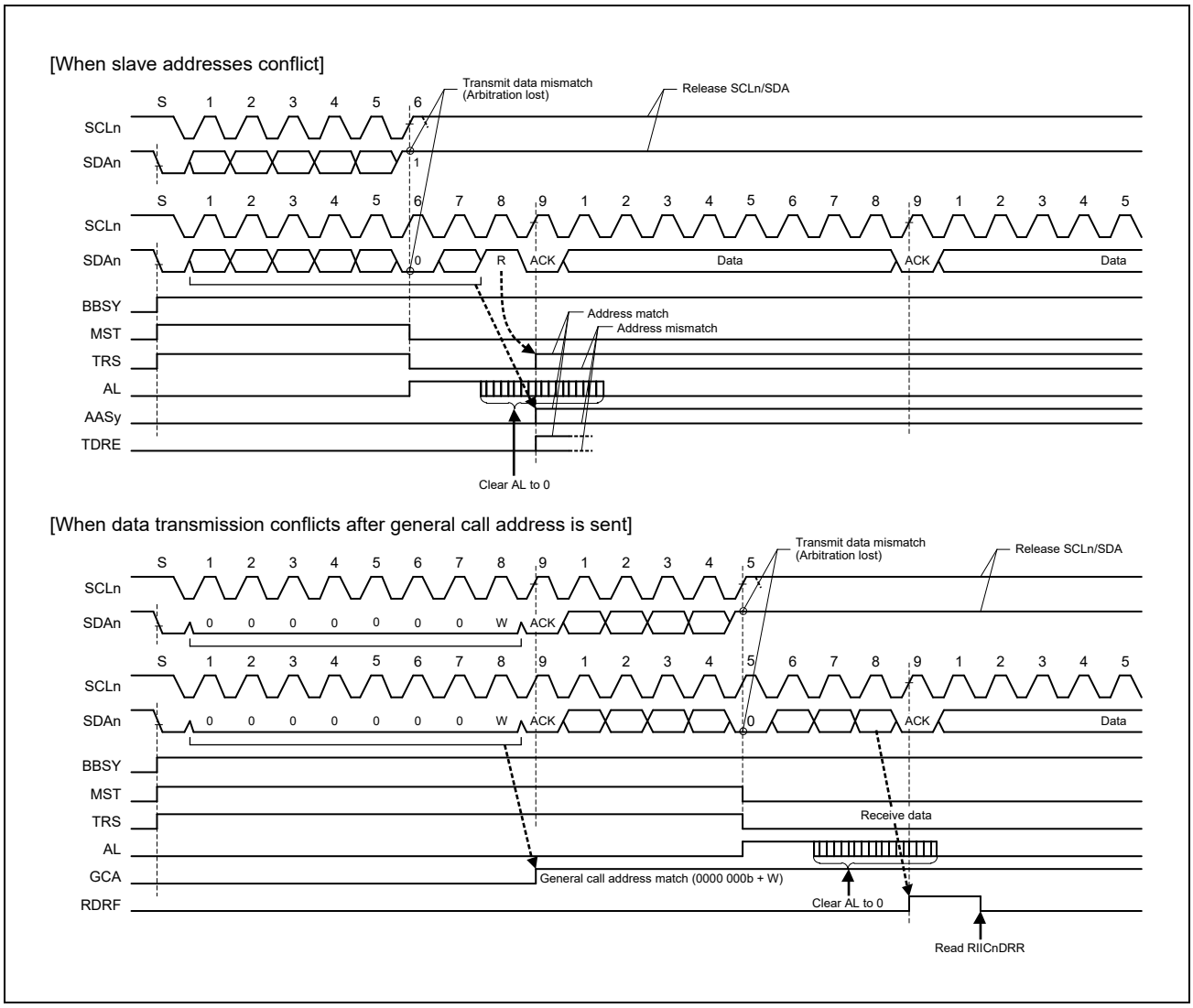


Figure 25.33 Examples of Master Arbitration-Lost Detection (MALE = 1)

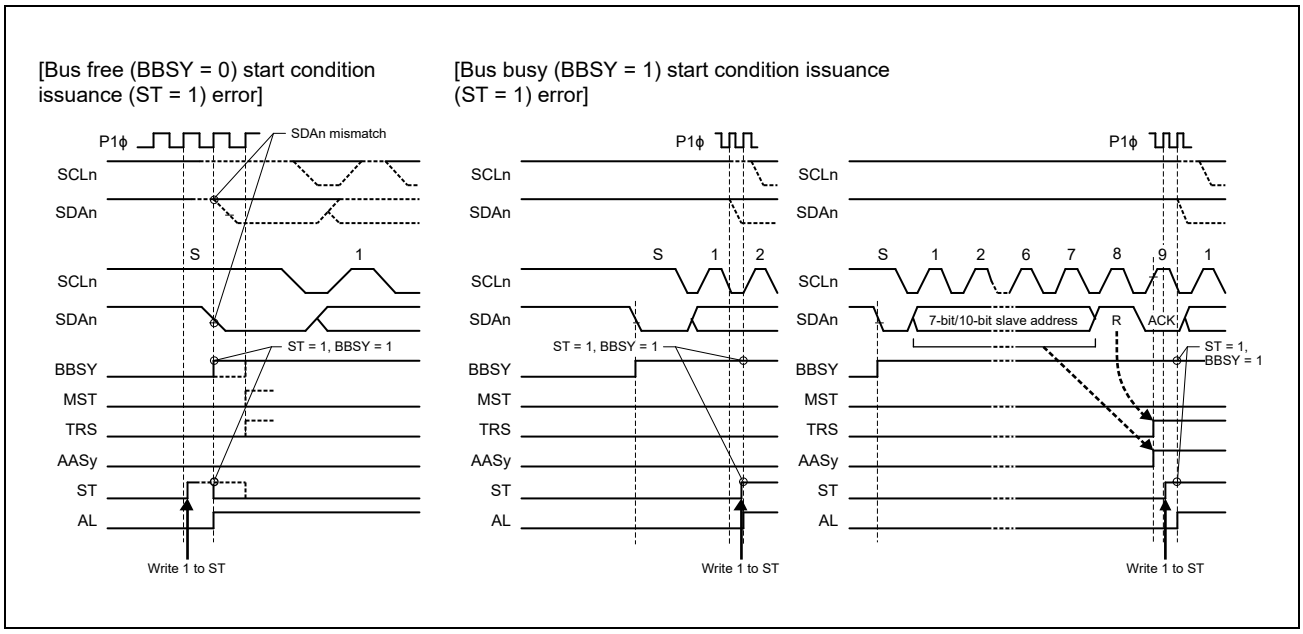


Figure 25.34 Arbitration-Lost when a Start Condition is Issued (MALE = 1)

25.11.2 Function to Detect Loss of Arbitration during NACK Transmission (NALE Bit)

The RIIC has a function to cause arbitration to be lost if the internal SDA output level does not match the level on the SDA line (the high output as the internal SDA output; i.e. the SDA pin is in the high-impedance state) and the low level is detected on the SDA line during transmission of NACK in receive mode. Arbitration is lost due to a conflict of NACK transmission and ACK transmission when two or more master devices receive data from the same slave device simultaneously in a multi-master system. Such conflict occurs when multiple master devices send/receive the same information through a single slave device. **Figure 25.35** shows an example of arbitration-lost detection during transmission of NACK.

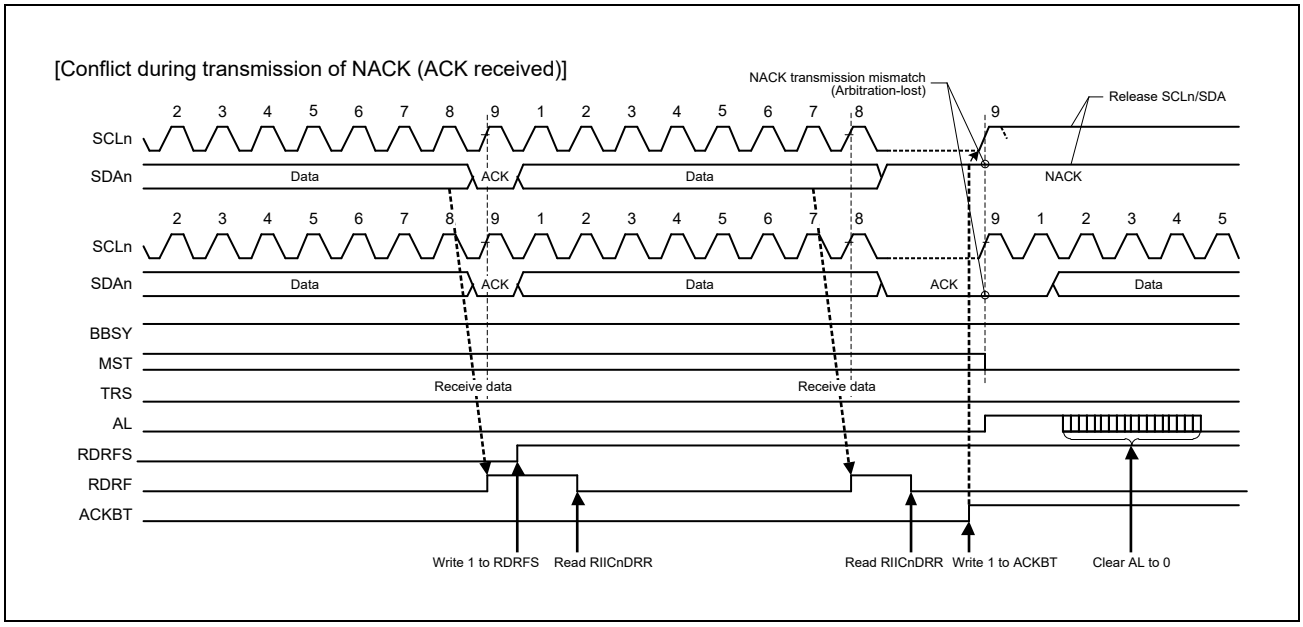


Figure 25.35 Example of Arbitration-Lost Detection during Transmission of NACK (NALE = 1)

The following explains arbitration-lost detection using an example where two master devices (master A and master B) and a single slave device are connected through the bus. In this example, master A receives two bytes of data from the slave device, and master B receives four bytes of data from the slave device.

If master A and master B access the slave device simultaneously, because the slave address is identical, arbitration is not lost in both master A and master B during access to the slave device. Therefore, both master A and master B recognize that they have obtained the bus mastership and operate as such. Here, master A sends NACK when it has received two final bytes of data from the slave device. Meanwhile, master B sends ACK because it has not received necessary four bytes of data. At this time, the NACK transmission from master A and the ACK transmission from master B conflict. In general, if a conflict like this occurs, master A cannot detect ACK transmitted by master B and issues a stop condition. Therefore, the issuance of the stop condition conflicts with the SCL clock output of master B, which disturbs communication.

When the RIIC receives ACK during transmission of NACK, it detects a defeat in conflict with other master devices and causes arbitration to be lost.

If arbitration is lost during transmission of NACK, the RIIC enters slave receive mode. This prevents a stop condition from being issued, preventing a communication failure on the bus.

Similarly, in the ARP command processing of SMBus, the function to detect loss of arbitration during transmission of NACK is also available for eliminating the extra clock cycle processing (such as H'FF transmission processing)

necessary if the UDID (Unique Device Identifier) of assign address does not match in the Get UDID (general) processing after the Assign address command.

The RIIC detects arbitration-lost during transmission of NACK when the following condition is met with the RIICnFER.NALE bit set to 1 (arbitration-lost detection during NACK transmission enabled).

[Condition for arbitration-lost during NACK transmission]

When the internal SDA output level does not match the SDA line (ACK is received) during transmission of NACK (RIICnMR3.ACKBT bit = 1).

25.11.3 Slave Arbitration-Lost Detection (SALE Bit)

The RIIC has a function to cause arbitration to be lost if the data for transmission (i.e. the internal SDA output level) and the level on the SDA line do not match (the high output as the internal SDA output; i.e. the SDA pin is in the high-impedance state) and the low level is detected on the SDA line in slave transmit mode. This arbitration-lost detection function is mainly used when transmitting a UDID (Unique Device Identifier) over an SMBus.

When it loses slave arbitration, the RIIC enters slave receive mode. This function can detect conflicts of data during transmission of UDIDs over an SMBus and eliminates subsequent redundant processing (processing for the transmission of H'FF).

The RIIC detects slave arbitration-lost when the following condition is met with the RIICnFER.SALE bit set to 1 (slave arbitration-lost detection enabled). [Condition for slave arbitration-lost]

When transmit data excluding acknowledge (internal SDA output level) does not match the SDA line in slave transmit mode (RIICnCR2.MST and TRS bits = 01b)

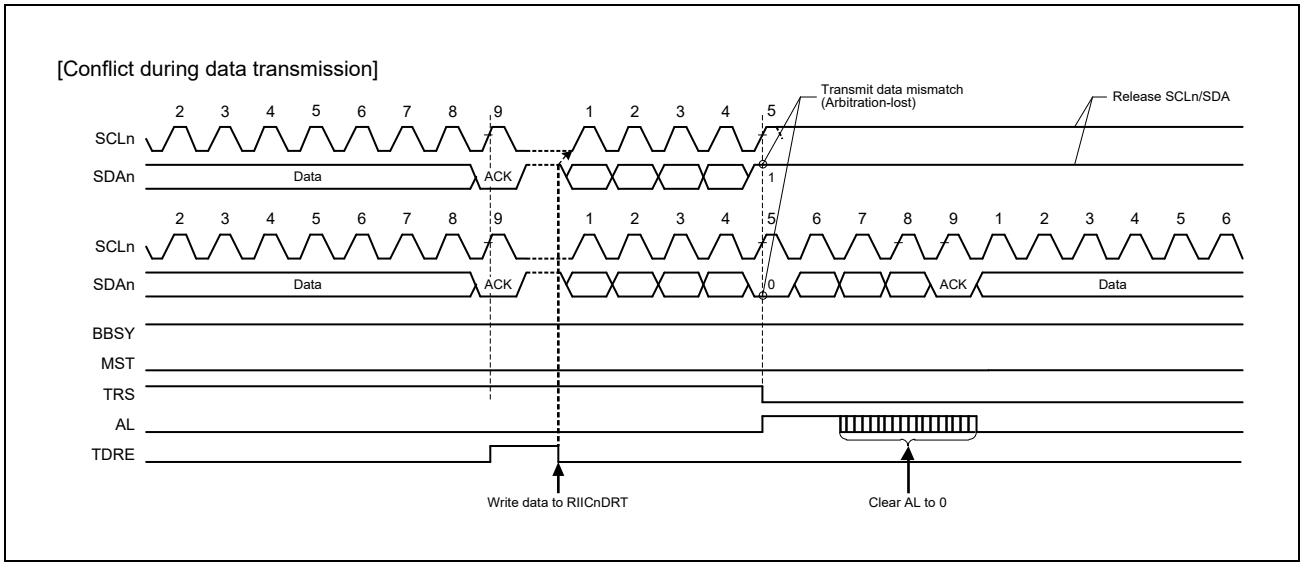


Figure 25.36 Example of Slave Arbitration-Lost Detection (SALE = 1)

25.12 Start Condition/Restart Condition/Stop Condition Issuing Function

25.12.1 Issuing a Start Condition

The RIIC issues a start condition when the RIICnCR2.ST bit is set to 1.

When the ST bit is set to 1, a start condition issuance request is made and the RIIC issues a start condition when the RIICnCR2.BBSY flag is 0 (bus free). When a start condition is issued normally, the RIIC automatically shifts to the master transmit mode.

A start condition is issued in the following sequence.

[Start condition issuance]

- Drive the SDA line low (high level to low level).
- Ensure the time set in RIICnBRH and the start condition hold time.
- Drive the SCL line low (high level to low level).
- Detect low level of the SCL line and ensure the low-level period of SCL line set in RIICnBRL.

25.12.2 Issuing a Restart Condition

The RIIC issues a restart condition when the RIICnCR2.RS bit is set to 1.

When the RS bit is set to 1, a restart condition issuance request is made and the RIIC issues a restart condition when the RIICnCR2.BBSY flag is 1 (bus busy) and the RIICnCR2.MST bit is 1 (master mode).

A restart condition is issued in the following sequence.

[Restart condition issuance]

- Release the SDA line.
- Ensure the low-level period of SCL line set in RIICnBRL.
- Release the SCL line (low level to high level).
- Detect a high level of the SCL line and ensure the time set in RIICnBRL and the restart condition setup time.
- Drive the SDA line low (high level to low level).
- Ensure the time set in RIICnBRH and the restart condition hold time.
- Drive the SCL line low (high level to low level).
- Detect a low level of the SCL line and ensure the low-level period of SCL line set in RIICnBRL.

CAUTION

To issue a restart condition request, set RIICnSR2.RS to 1 after setting RIICnSR2.START to 0. After checking that RIICnSR2.START is 1, set RIICnSR2.START to 0, and then write the slave address to RIICnDRT.

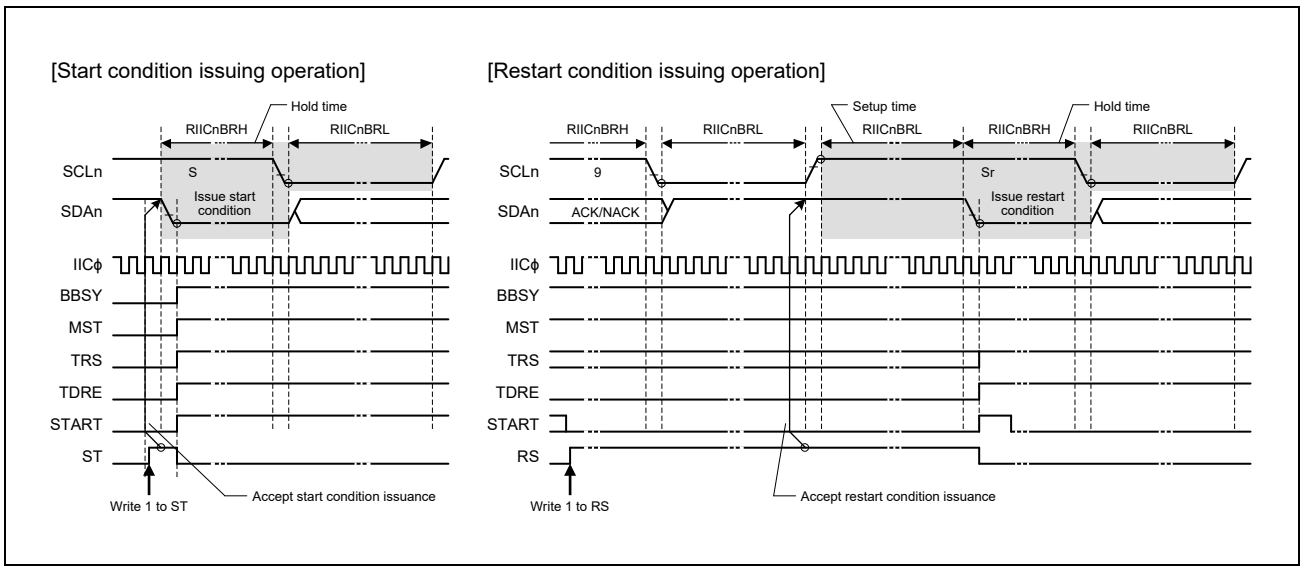


Figure 25.37 Start Condition/Restart Condition Issue Timing (ST and RS Bits)

25.12.3 Issuing a Stop Condition

The RIIC issues a stop condition when the RIICnCR2.SP bit is set to 1.

When the SP bit is set to 1, a stop condition issuance request is made and the RIIC issues a stop condition when the RIICnCR2.BBSY flag is 1 (bus busy) and the RIICnCR2.MST bit is 1 (master mode).

A stop condition is issued in the following sequence.

[Stop condition issuance]

- Drive the SDA line low (high level to low level).
- Ensure the low-level period of SCL line set in RIICnBRL.
- Release the SCL line (low level to high level).
- Detect a high level of the SCL line and ensure the time set in RIICnBRH and the stop condition setup time.
- Release the SDA line (low level to high level).
- Ensure the time set in RIICnBRL and the bus free time.
- Clear the BBSY flag to 0 (to release the bus mastership).

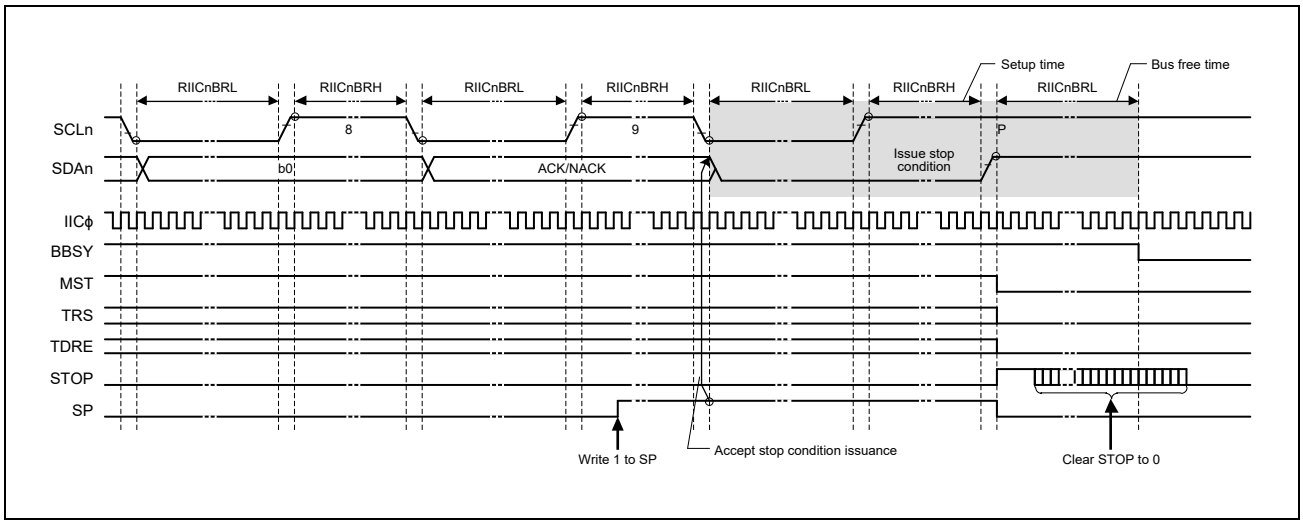


Figure 25.38 Stop Condition Issue Timing (SP Bit)

25.13 Bus Hanging

If the clock signals from the master and slave devices go out of synchronization due to noise or other factors, the I^2C bus might hang with a fixed level on the SCL line and/or SDA line.

As measures against the bus hanging, the RIIC has a timeout function to detect hanging by monitoring the SCL line, a function for the output of an extra SCL clock cycle to release the bus from a hung state due to clock signals being out of synchronization, and the RIIC/internal reset function.

By checking the RIICnCR1.SCLO, SDAO, SCLI, and SDAI bits, it is possible to see whether the RIIC or its partner in communications is placing the low level on the SCL or SDA lines.

25.13.1 Timeout Function

The RIIC has the timeout function to detect an abnormality that the SCL line is held for a certain period of time. The RIIC can detect an abnormal bus state by monitoring that the SCL line is held low or high for a predetermined time. The timeout function monitors the SCL line state and counts the low-level period or high-level period using the internal counter. The timeout function resets the internal counter each time the SCL line changes (rising or falling), but continues to count unless the SCL line changes. If the internal counter overflows due to no SCL line change, the RIIC can detect the timeout and report the bus abnormality.

This timeout function is enabled when the RIICnFER.TMOE bit is 1. It detects an abnormal bus state in which the SCL line is held low or high in the following cases.

1. When the bus is busy (RIICnCR2.BBSY = 1) in master mode (RIICnCR2.MST = 1)
2. When the bus is busy (RIICnCR2.BBSY = 1) and the RIIC's own slave address matches (RIICnSR1 is not H'00) in slave mode (RIICnCR2.MST = 0)
3. While the bus is free (RIICnCR2.BBSY = 0) and issuing of a start condition is being requested (RIICnCR2.ST = 1).

The internal counter of the timeout function works using the internal reference clock (IICϕ) set by the RIICnMR1.CKS[2:0] bits as a count source. It functions as a 16-bit counter when long mode is selected (RIICnMR2.TMOS bit = 0) or a 14-bit counter when short mode is selected (TMOS bit = 1).

The SCL line level (low/high or both levels) during which this counter is activated can be selected by the setting of the RIICnMR2.TMOH and TMOL bits. If both TMOL and TMOH bits are cleared to 0, the internal counter does not work.

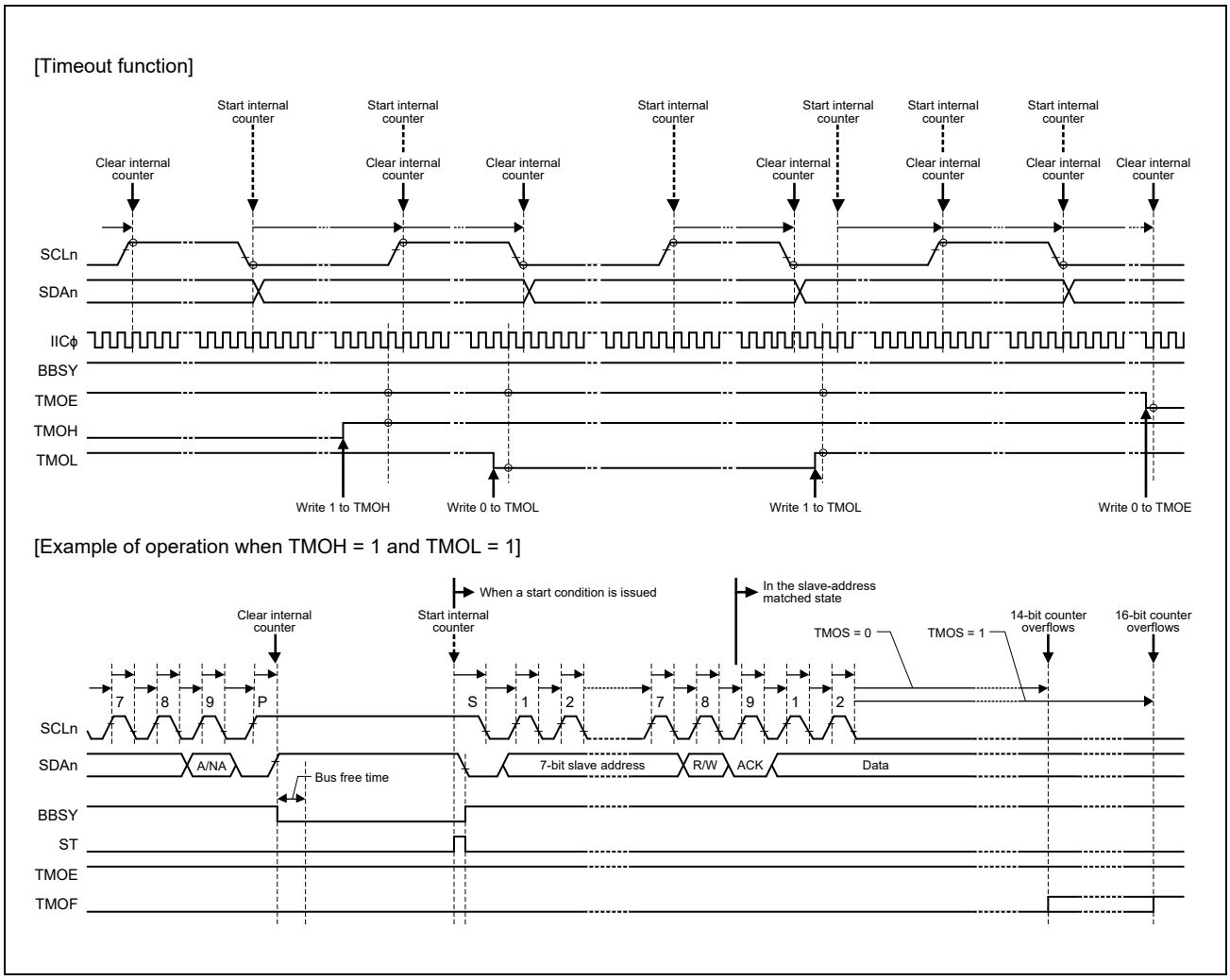


Figure 25.39 Timeout Function (TMOE, TMOS, TMOH, and TMOL Bits)

25.13.2 Extra SCL Clock Cycle Output Function

In master mode, the RIIC module has a facility for the output of extra SCL (clock) cycles to release the SDA line of the slave device from being held at the low level due to the master being out of synchronization with the slave device.

This function is mainly used in master mode to release the SDA line of the slave device from the state of being fixed to the low level by including extra cycles of SCL output from the RIIC with single cycles of the SCL (clock) signal as the unit in the case of a bus error where the RIIC cannot issue a stop condition because the slave device is holding the SDA line at the low level. Do not use this facility in normal situations. Using it when communications are proceeding correctly will lead to malfunctions.

When the RIICnCR1.CLO bit is set to 1 in master mode, a single cycle of the SCL clock at the frequency corresponding to the transfer rate settings (settings of the RIICnMR1.CKS[2:0] bits, and of the RIICnBRH and RIICnBRL registers) is output as an extra clock cycle. After output of this single cycle of the SCL clock, the CLO bit is automatically cleared to 0. At this point, the SCL pin output is held at a low level if the BBSY flag is 1, or the SCL pin outputs at high level if the BBSY flag is 0. Further clock cycles can consecutively be output by writing 1 to the CLO bit after having read CLO is 0 by software.

When the RIIC module is in master mode and the slave device is holding the SDA line at the low level because synchronization with the slave device has been lost due to the effects of noise, etc., the output of a stop condition is not possible. The facility for output of an extra cycle of the SCL (clock) signal can be used to output extra cycles of SCL one by one to make the slave device release the SDA line from being held at the low level, thus recovering the bus from an unusable state. Release of the SDA line by the slave device can be monitored by reading the RIICnCR1.SDAI bit. After confirming release of the SDA line by the slave device, complete communications by reissuing the stop condition.

[Output conditions for using the RIICnCR1.CLO bit]

- When the bus is free (RIICnCR2.BBSY flag = 0) or in master mode (RIICnCR2.MST bit = 1 and BBSY flag = 1)
- When the communication device does not hold the SCL line low

Figure 25.40 shows the operation timing of the extra SCL clock cycle output function (CLO bit).

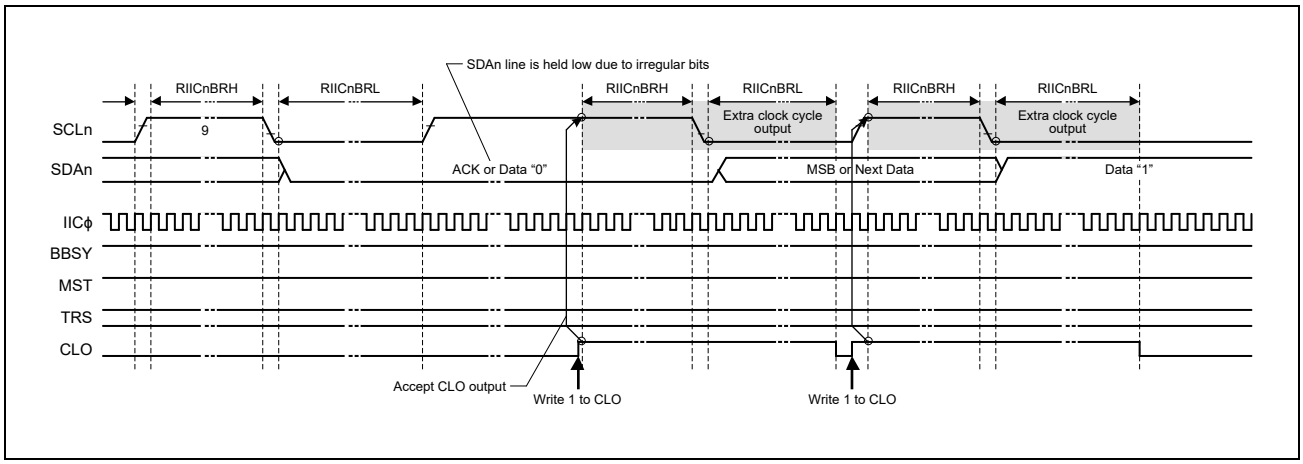


Figure 25.40 Extra SCL Clock Cycle Output Function (CLO Bit)

25.13.3 RIIC Reset and Internal Reset

The RIIC module incorporates a function for resetting itself. There are two types of reset. One is referred to as an RIIC reset; this initializes all registers including the RIICnCR2.BBSY flag. The other is referred to as an internal reset; this releases the RIIC from the slave-address matched state and initializes the internal counter while retaining other settings. After issuing a reset, be sure to clear the RIICnCR1.IICRST bit to 0.

Both types of reset are effective for release from bus-hung states since both restore the output state of the SCL and SDA pins to the high impedance state.

Issuing a reset during slave operation may lead to a loss of synchronization between the master device clock and the slave device clock, so avoided this where possible. Note that monitoring of the bus state, such as for the presence of a start condition, is not possible during an RIIC reset (RIICnCR1.ICE and IICRST bits = 01b).

For a detailed description of the RIIC and internal resets, see **Section 25.15, Reset Function of RIIC**.

25.14 SMBus Operation

The RIIC is available for data communication conforming to the SMBus (Version 2.0). To perform SMBus communication, set the RIICnMR3.SMBS bit to 1. To use the transfer rate within a range of 10 kbps to 100 kbps of the SMBus standard, set the RIICnMR1.CKS[2:0] bits, RIICnCBRH, and RIICnBRL. In addition, determine the values of the RIICnMR2.DLCS bit and the RIICnMR2.SDDL[2:0] bits to meet the data hold time specification of 300 ns or more. If the RIIC is used only as a slave device, the transfer rate setting is not necessary. When the RIIC is used only as a slave device, the transfer rate setting is not necessary, whereas the RIICnBRL needs to be set to a value at least the same as the data setup time (250 ns).

For the SMBus device default address (1100_001b), use one of the slave address registers L0 to L2 (RIICnSARL0, RIICnSARL1, and RIICnSARL2), and set the corresponding RIICnSARUy.FS bit (7-bit/10-bit address format select) (y = 0 to 2) to 0 (7-bit address format).

When transmitting the UDID (Unique Device Identifier), set the RIICnFER.SALE bit to 1 to enable the slave arbitration lost detection function.

25.14.1 SMBus Timeout Measurement

(1) Measuring timeout of slave device

The following period (timeout interval: $T_{LOW:SEXT}$) must be measured for slave devices in SMBus communication.

- From start condition to stop condition

  To measure timeout for slave devices, measure the period from start condition detection to stop condition detection with the internal timer using a start condition detection interrupt (INTRIICSTI) and stop condition detection interrupt (INTRIICSPI) of the RIIC. The measured timeout period must be within the total clock low-level period [slave device] $T_{LOW:SEXT}$: 25 ms (max.) of the SMBus standard.

  If the time measured with the internal timer exceeds the clock low-level detection timeout TTIMEOUT: 25 ms (min.) of the SMBus standard, the slave device must release the bus by writing 1 to the RIICnCR1.IICRST bit to issue an internal reset of the RIIC. When an internal reset is issued, the RIIC stops driving the bus for the SCL pin and SDA pin and make the SCL/SDA pin outputs high impedance, which releases the bus.

(2) Measuring timeout of master device

The following periods (timeout interval: $T_{LOW:MEXT}$) must be measured for master devices in SMBus communication.

- From start condition to acknowledge bit
- Between acknowledge bits
- From acknowledge bit to stop condition

To measure timeout for master devices, measure these periods with the internal timer using a start condition detection interrupt (INTRIICSTI), stop condition detection interrupt (INTRIICSPI), and transmit end interrupt (INTRIICTEI) or receive data full interrupt (INTRIICRI) of the RIIC. The measured timeout period must be within the total clock low-level extended period [master device] $T_{LOW:MEXT}$: 10 ms (max.) of the SMBus standard, and the total of all $T_{LOW:MEXT}$ from start condition to stop condition must be within $T_{LOW:SEXT}$: 25 ms (max.).

For the ACK receive timing (rising edge of the ninth SMBCLK clock cycle), monitor the RIICnSR2.TEND flag in master transmit mode (master transmitter) and the RIICnSR2.RDRF flag in master receive mode (master receiver). For this reason, perform bytewise transmit operation in master transmit mode, and hold the RIICnMR3.RDRFS bit 0 until the byte just before reception of the final byte in master receive mode. While the RDRFS bit is 0, the RDRF flag is set to 1 at the rising edge of the ninth SMBCLK clock cycle.

If the period measured with the internal timer exceeds the total clock low-level extended period [master device] $T_{LOW:MEXT}$: 10 ms (max.) of the SMBus standard or the total of measured periods exceeds the clock low-level detection timeout TTIMEOUT: 25 ms (min.) of the SMBus standard, the master device must stop the transaction by issuing a stop condition. In master transmit mode, immediately stop the transmit operation (writing data to RIICnDRT).

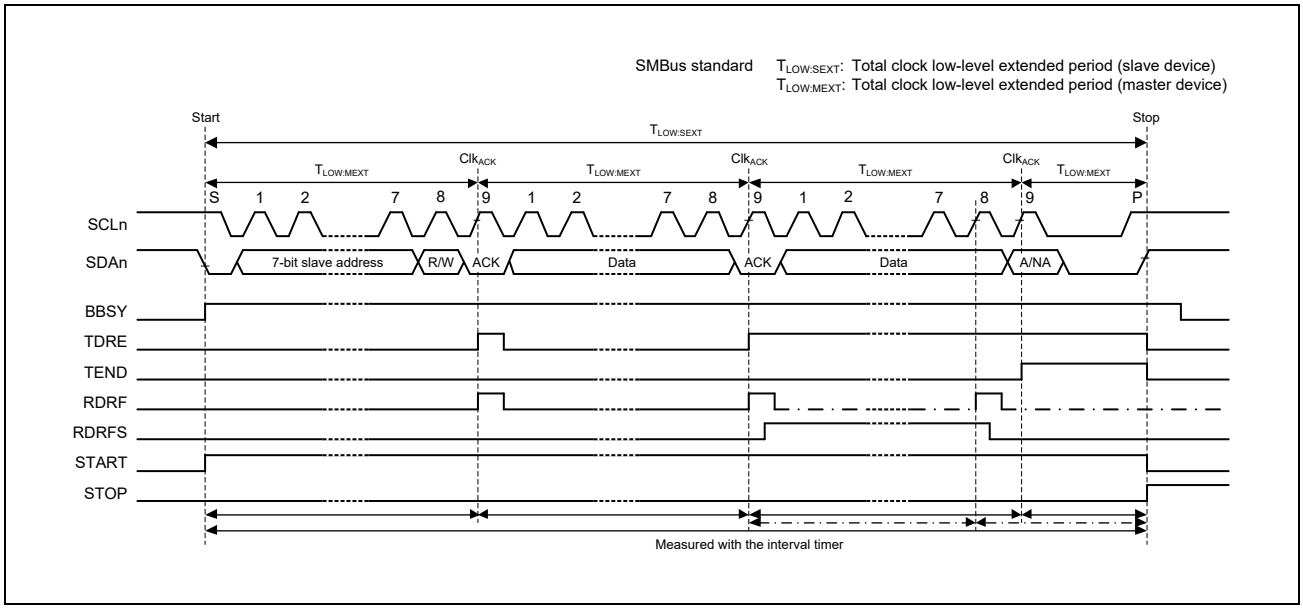


Figure 25.41 SMBus Timeout Measurement

25.14.2 SMBus Host Notification Protocol/Notify ARP Master

In communications over an SMBus, a slave device can temporarily act as a master device to notify the SMBus host (or ARP master) of (or request the SMBus host for) its own slave address or to request its own slave address from the SMBus host.

For this LSI to operate as an SMBus host (or ARP master), the host address (0001_000b) sent from the slave device must be detected as a slave address, so the RIIC has a function for detecting the host address. To detect the host address as a slave address, set the RIICnMR3.SMBS bit and the RIICnSER.HOAE bit to 1. Operation after the host address has been detected is the same as normal slave operation.

25.15 Reset Function of RIIC

The RIIC has chip reset, RIIC reset, and internal reset functions. **Table 25.13** lists the scope of each reset and reset conditions.

Table 25.13 RIIC Reset Functions (1/2)

Register		RIIC Reset (ICE = 0, IICRST = 1)	Internal Reset (ICE = 1, IICRST = 1)	Start Condition/Restart Condition Detection	Stop Condition Detection
RIICnCR1	ICE	0	1	Retained	Retained
	IICRST	1	1	Retained	Retained
	CLO	Initialized	Retained	Retained	Retained
	SOWP	Initialized	Retained	Retained	Retained
	SCLO	Initialized	Initialized	Retained	Retained
	SDAO	Initialized	Initialized	Retained	Retained
	SCLI	Initialized	Retained	Retained	Retained
	SDAI	Initialized	Retained	Retained	Retained
RIICnCR2	BBSY	Initialized	Initialized*1	Operation	Retained
	MST	Initialized	Initialized	Operation (retained)	Initialized
	TRS	Initialized	Initialized	Operation (retained)	Initialized
	SP	Initialized	Initialized	Initialized	Initialized
	RS	Initialized	Initialized	Initialized	Initialized
	ST	Initialized	Initialized	Initialized	Retained
RIICnMR1	MTWP	Initialized	Retained	Retained	Retained
	CKS[2:0]	Initialized	Retained	Retained	Retained
	BCWP	Initialized	Retained	Retained	Retained
	BC[2:0]	Initialized	Initialized	Initialized	Retained
RIICnMR2		Initialized	Retained	Retained	Retained
RIICnMR3	SMBS	Initialized	Retained	Retained	Retained
	WAIT	Initialized	Retained	Retained	Retained
	RDRFS	Initialized	Retained	Retained	Retained
	ACKWP	Initialized	Retained	Retained	Retained
	ACKBT	Initialized	Retained	Retained	Initialized
	ACKBR	Initialized	Retained	Retained	Retained
	NF[1:0]	Initialized	Retained	Retained	Retained
RIICnFER		Initialized	Retained	Retained	Retained
RIICnSER		Initialized	Retained	Retained	Retained
RIICnIER		Initialized	Retained	Retained	Retained
RIICnSR1	DID	Initialized	Initialized	Retained	Initialized
	GCA	Initialized	Initialized	Retained	Initialized
	AAS2	Initialized	Initialized	Retained	Initialized
	AAS1	Initialized	Initialized	Retained	Initialized
	AAS0	Initialized	Initialized	Retained	Initialized
RIICnSR2	TDRE	Initialized	Initialized	Retained	Initialized
	TEND	Initialized	Initialized	Retained	Initialized
	RDRF	Initialized	Initialized	Retained	Retained
	NACKF	Initialized	Initialized	Retained	Retained
	STOP	Initialized	Initialized	Retained	Operation

Table 25.13 RIIC Reset Functions (2/2)

Register		RIIC Reset (ICE = 0, IICRST = 1)	Internal Reset (ICE = 1, IICRST = 1)	Start Condition/Restart Condition Detection	Stop Condition Detection
RIICnSR2	START	Initialized	Initialized	Operation	Initialized
	AL	Initialized	Initialized	Retained	Retained
	TMOF	Initialized	Initialized	Retained	Retained
RIICnSAR(L,U)y (y = 0 to 2)		Initialized	Retained	Retained	Retained
RIICnBRH, RIICnBRL		Initialized	Retained	Retained	Retained
RIICnDRT		Initialized	Retained	Retained	Retained
RIICnDRR		Initialized	Retained	Retained	Retained
RIICnDRS		Initialized	Initialized	Retained	Retained

Note 1. When an internal reset is applied while the bus is free after detection of a stop condition, the setting of the BBSY flag is 0 while the bus is free following de-assertion of the internal reset signal.
When an internal reset is applied while the bus is not free, the BBSY flag is not cleared.

26. I3C Bus Interface (I3C)

NOTE

This LSI does not support High-speed mode (Hs-mode) for I2C mode.
Related descriptions of the Hs-mode in this section should be ignored.

26.1 Overview

26.1.1 Functional Overview

The I3C bus interface (I3C) has 1 channel. The I3C module conform with and provide a subset of the NXP I^2C (Inter-Integrated Circuit) bus interface functions and a subset of the MIPI I3C.

Table 26.1 lists the I^2C specifications, and **Table 26.2** lists the I3C specifications.

Table 26.1 I^2C Specifications

Item	Description
Operation mode	Master mode and slave mode selectable
Data handler	Single buffer transfer
Communication protocol	• I^2C bus format – Standard-mode (Sm): Up to 100 kbps – Fast-mode (Fm): Up to 400 kbps – Fast-mode Plus (Fm+): Up to 1 Mbps
Address format	• 7-bit address • 10-bit address
Address detection	• Slave address (static address) (max. 3 address) • General call address • Device ID • Host address • 10-bit slave addressing
Clock stretching	Clock stretching capability
Noise-filter	• Analog noise-filter • Digital noise-filter
Interrupt source	• Rx data buffer full • Tx data buffer empty • START condition detection • STOP condition detection • Transmit end • NACK detection • Arbitration lost • Timeout detection • Wake-up condition detection
Error detection	• NACK received • Arbitration lost error • Timeout error
Wake-Up Source	• Address detection of Slave Address
Operation mode	Master (main master/secondary master) mode and slave mode selectable
Data handler	[Master] Normal FIFO buffer transfer [Slave] Normal FIFO buffer transfer

Table 26.2 I3C Specifications

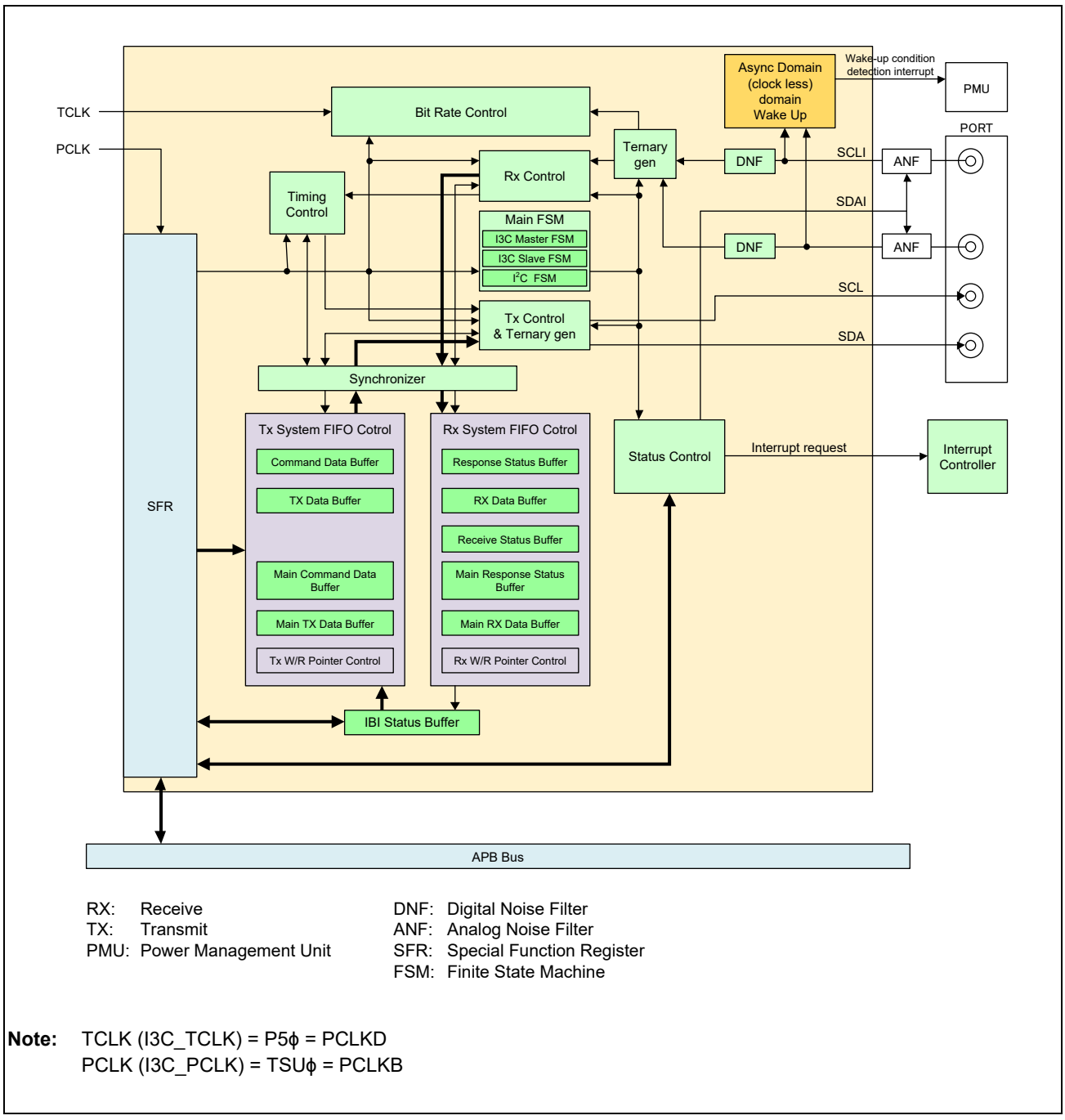
Item	Description
Communication protocol	• SDR (I3C single data rate) mode – Private message – Broadcast message (common command code) – Direct message (common command code) • Legacy I^2C message – Fast-mode (Fm): up to 400 kbps – Fast-mode Plus (Fm+): up to 1 Mbps
In-band interrupt	• Slave interrupt request • Master ship request (secondary master only) • Hot-join event
Address format	7-bit address
Address detection	• Slave address (static address or dynamic address) • Broadcast address (H'7E)
Clock stalling	Clock stalling capability
Timing Control	• Synchronous Timing Control – Sync Mode: Synchronous Basic Mode • Asynchronous Timing Control – Async Mode 0: Asynchronous Basic Mode – Async Mode 1: Asynchronous Advanced Mode
Interrupt source	• Non-recoverable internal error • Transfer error • Transfer abort • Response queue full • Command queue empty • IBI status queue full • Receive data buffer full • Transmit data buffer empty • Receive status queue full • START condition detection • STOP condition detection • HDR exit pattern detection • Timeout detection • Wake-up condition detection
Error detection	• Non-recoverable internal error • CRC error • Parity error • Frame error • Address header error • Address NACKed or dynamic address assignment NACKed • Receive overflow or transfer underflow error. • Aborted • NACK received for the I^2C write data transfer. • Timeout error
Wake-Up Source	[Master] SDA assert of IBI (START condition detection) [Slave] Address detection of Broadcast Address (H'7E) and Slave Address

Table 26.3 I3C I/O Pins

Function	Pin name	I/O	Description
I3C	I3C_SCL	I/O	Input/output pins for clock
	I3C_SDA	I/O	Input/output pins for data

26.1.2 Block Diagram [I²C/I3C common]

Figure 26.1 shows the main components of this I3C.



RX: Receive
TX: Transmit
PMU: Power Management Unit
DNF: Digital Noise Filter
ANF: Analog Noise Filter
SFR: Special Function Register
FSM: Finite State Machine

Note: TCLK (I3C_TCLK) = P5ϕ = PCLKD
PCLK (I3C_PCLK) = TSUϕ = PCLKB

Figure 26.1 I3C Block Diagram

26.2 Registers

26.2.1 List of Registers

I3C registers are listed in the following table.

Base Address: H'0_1005_B000 (Cortex-A55 Address Space)
Base Address: H'4005_B000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
Base Address: H'5005_B000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 26.4 List of I3C Registers (1/3)

Register	Symbol	Offset Address
Protocol Selection Register	PRTS	H'000
Bus Control Register	BCTL	H'014
Master Device Address Register	MSDVAD	H'018
Reset Control Register	RSTCTL	H'020
Present State Register	PRSST	H'024
Internal Status Register	INST	H'030
Internal Status Enable Register	INSTE	H'034
Internal Interrupt Enable Register	INIE	H'038
Internal Status Force Register	INSTFC	H'03C
Device Characteristic Table Register	DVCT	H'044
IBI Notify Control Register	IBINCTL	H'058
Bus Function Control Register	BFCTL	H'060
Slave Control Register	SVCTL	H'064
Reference Clock Control Register	REFCKCTL	H'070
Standard Bit Rate Register	STDBR	H'074
Extended Bit Rate Register	EXTBR	H'078
Bus Free Condition Detection Time Register	BFRECDT	H'07C
Bus Available Condition Detection Time Register	BAVLCDT	H'080
Bus Idle Condition Detection Time Register	BIDLCDT	H'084
Output Control Register	OUTCTL	H'088
Input Control Register	INCTL	H'08C
Timeout Control Register	TMOCTL	H'090
Wake Up Unit Control Register	WUCTL	H'098
Acknowledge Control Register	ACKCTL	H'0A0
SCL Stretch Control Register	SCSTRCTL	H'0A4
SCL Stalling Control Register	SCSTLCTL	H'0B0
Slave Transfer Data Length Register 0	SVTDLG0	H'0C0
Synchronous Timing Control Register	STCTL	H'120
Asynchronous Timing Control Register	ATCTL	H'124
Asynchronous Timing Trigger Register	ATTRG	H'128
Asynchronous Timing Control Counter enable Register	ATCCNTE	H'12C

Table 26.4 List of I3C Registers (2/3)

Register	Symbol	Offset Address
Condition Control Register	CNDCTL	H'140
Normal Command Queue Port Register	NCMDQP	H'150
Normal Response Queue Port Register	NRSPQP	H'154
Normal Transfer Data Buffer Port Register 0	NTDTBP0/NTDTBP0_BY	H'158
Normal IBI Queue Port Register	NIBIQP	H'17C
Normal Receive Status Queue Port Register	NRSQP	H'180
Normal Queue Threshold Control Register	NQTHCTL	H'190
Normal Transfer Data Buffer Threshold Control Register 0	NTBTHCTL0	H'194
Normal Receive Status Queue Threshold Control Register	NRQTHCTL	H'1C0
Bus Status Register	BST	H'1D0
Bus Status Enable Register	BSTE	H'1D4
Bus Interrupt Enable Register	BIE	H'1D8
Bus Status Force Register	BSTFC	H'1DC
Normal Transfer Status Register	NTST	H'1E0
Normal Transfer Status Enable Register	NTSTE	H'1E4
Normal Transfer Interrupt Enable Register	NTIE	H'1E8
Normal Transfer Status Force Register	NTSTFC	H'1EC
Bus Condition Status Register	BCST	H'210
Slave Status Register	SVST	H'214
Wake Up Unit Operating Status Register	WUST	H'218
MsyncCNT Counter Capture Register	MRCCPT	H'21C
Device Address Table Basic Register 0	DATBAS0	H'224
Device Address Table Basic Register 1	DATBAS1	H'22C
Device Address Table Basic Register 2	DATBAS2	H'234
Device Address Table Basic Register 3	DATBAS3	H'23C
Device Address Table Basic Register 4	DATBAS4	H'244
Device Address Table Basic Register 5	DATBAS5	H'24C
Device Address Table Basic Register 6	DATBAS6	H'254
Device Address Table Basic Register 7	DATBAS7	H'25C
Slave Device Address Table Basic Register 0	SDATBAS0	H'2B0
Slave Device Address Table Basic Register 1	SDATBAS1	H'2B4
Slave Device Address Table Basic Register 2	SDATBAS2	H'2B8
Master Device Characteristic Table Register 0	MSDCT0	H'2D0
Master Device Characteristic Table Register 1	MSDCT1	H'2D4
Master Device Characteristic Table Register 2	MSDCT2	H'2D8
Master Device Characteristic Table Register 3	MSDCT3	H'2DC
Master Device Characteristic Table Register 4	MSDCT4	H'2E0
Master Device Characteristic Table Register 5	MSDCT5	H'2E4
Master Device Characteristic Table Register 6	MSDCT6	H'2E8
Master Device Characteristic Table Register 7	MSDCT7	H'2EC
Extended Device Address Table Basic Register	EXDATBAS	H'310
Slave Device Characteristic Table Register	SVDCT	H'320
Slave Device Characteristic Table Provisional ID Low Register	SDCTPIDL	H'324
Slave Device Characteristic Table Provisional ID High Register	SDCTPIDH	H'328
Slave Device Address Register 0	SVDVAD0	H'330

Table 26.4 List of I3C Registers (3/3)

Register	Symbol	Offset Address
Slave Device Address Register 1	SVDVAD1	H'334
Slave Device Address Register 2	SVDVAD2	H'338
CCC Slave Events Command Register	CSECMD	H'350
CCC Enter Activity State Register	CEACTST	H'354
CCC Max Write Length Register	CMWLG	H'358
CCC Max Read Length Register	CMRLG	H'35C
CCC Enter Test Mode Register	CETSTMD	H'360
CCC Get Device Status Register	CGDVST	H'364
CCC Max Data Speed W (Write) Register	CMDSPW	H'368
CCC Max Data Speed R (Read) Register	CMDSPR	H'36C
CCC Max Data Speed T (Turnaround) Register	CMDSPT	H'370
CCC Exchange Timing Support Information M (Mode) Register	CETSM	H'374
CCC Exchange Timing Support Information S (State) Register	CETSS	H'378
Bit Count Register	BITCNT	H'380
Normal Queue Status Level Register	NQSTLV	H'394
Normal Data Buffer Status Level Register 0	NDBSTLV0	H'398
Normal Receive Status Queue Status Level Register	NRSQSTLV	H'3C0
Present State Debug Register	PRSTDBG	H'3CC
Master Error Counters Register	MSERRCNT	H'3D0
SC1 Capture monitor Register	SC1CPT	H'3E0
SC2 Capture monitor Register	SC2CPT	H'3E4

Note: 8-bit access is valid only in I^2C mode.

26.2.2 Protocol Selection Register (PRTS)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PRTMD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
0	PRTMD	1b	R/W	Protocol Mode 0: I3C protocol mode I3C FIFO buffer transfer (Equivalent to HCI) 1: I^2C protocol mode I^2C single buffer transfer

26.2.3 Bus Control Register (BCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BUSE	RSM	ABT	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	RW1C	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	HJACK CTL	—	—	—	—	—	—	—	INCBA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	BUSE	0b	R/W	Bus Enable*2 0: I3C bus operation is disabled. 1: I3C bus operation is enabled.
30	RSM	0b	RW1C	Resume*2 Values when read: 0: I3C is running. 1: I3C is suspended (RW1C).
29	ABT	0b	R/W	Abort*1 0: I3C is running. 1: I3C has aborted a transfer.
28 to 9	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
8	HJACKCTL	0b	R/W	Hot-Join Acknowledge Control*1 0: ACK the Hot-Join request 1: NACK and send broadcast CCC to disable Hot-Join
7	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
6 to 1	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
0	INCBA	0b	RW	Include I3C Broadcast Address*1 0: Do not include I3C broadcast address for private transfers 1: Include I3C broadcast address for private transfers

Note 1. This bit supports for I3C master mode and I3C secondary master mode.

Note 2. This bit supports for all I3C mode.

INCBA bit (Include I3C Broadcast Address)

This bit controls whether the I3C broadcast address (H'7E) is included for private transfers.

If the I3C broadcast address is not included for private transfers, then IBIs driven from Slaves might not win the arbitration, potentially delaying acceptance of the IBIs.

HJACKCTL bit (Hot-Join Acknowledge Control)

This bit acts as global control to either ACK (0) or NACK (1) all Hot-Join Requests arriving from the Devices on the I3C Bus. If set to NACK (1), then the NACK will be followed by the broadcast CCC to disable Hot-Join.

ABT bit (Abort)

When set to 1, this bit allows I3C to relinquish control of the I3C Bus before completing the currently issued transfer.

In response to an ABORT request, I3C issues the STOP condition on the I3C Bus after the complete data byte is transferred or received.

The Driver shall clear the ABT bit to allow operation on the Bus.

If BCTL.ABT is set and ABORT processing is performed, please ignore ERR_STATUS of Response Descriptor.

RSM bit (Resume)

This bit is used to resume I3C operation following the Halt state.

I3C enters the Halt state (as indicated in register PRSTDBG) as a result of any type of error occurring in a transfer.

The error type is indicated by the field ERR_STATUS in register NRSPQP, NRSQP and NIBIQP).

After I3C has entered the Halt state, the application must write the value 1 to the RSM bit to resume I3C operation. I3C shall auto clear the RSM bit once it has resumed making transfers (it has initiated the next Command).

BUSE bit (Bus Enable)

Enables or disables the operation on the I3C Bus by I3C.

Set the BUSE bit to 1 when using I3C. The SCL and SDA pins are placed in the active state when the BUSE bit is set to 1. Set the BUSE bit to 0 when I3C is not to be used. The SCL and SDA pins are placed in the inactive state when the BUSE bit is set to 0.

If the software sets this bit, then it also confirms that initialization is done, and that I3C can use the programmed register values (For example, generation of SCL on IBI detection, etc.). If this bit is not set, then I3C shall not generate SCL for incoming IBI.

Software may disable I3C bus operation while it is active, however:

- If a disabled request occurs while receiving IBI, the actual disabling will not occur until reception of the IBI is complete.
- When the software reads the value 0 from this field, this indicates that I3C bus operation disable operation has completed.
  If commands remain in the command queue, do not set BUSE = 0.

26.2.4 Master Device Address Register (MSDVAD)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MDYADV	—	—	—	—	—	—	—	—	MDYAD[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	MDYADV	0b	R/W	Master Dynamic Address Valid 0: The master dynamic address field is not valid. 1: The master dynamic address field is valid.
30 to 23	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
22 to 16	MDYAD[6:0]	All 0	R/W	Master Dynamic Address
15 to 0	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.

MDYAD[6:0] bits (Master Dynamic Address)

This field is used to program I3C master dynamic address. I3C uses this address to respond to master transactions in I3C interface mode (slave or secondary master role).

In I3C main master mode, the software shall program the dynamic address as it self-assigns its dynamic address.

MDYADV bit (Master Dynamic Address Valid)

This bit indicates whether or not the value in the MDYAD field is valid.

In I3C main master mode, the user sets this bit to 1 as it self-assigns its dynamic address.

NOTE

After setting MSDVAD and setting BCTL. BUSE = 1, the device will act as main master.
Without setting MSDVAD, setting SVDCT.TBCR[7:6] = 00b (Device Role Slave), and setting BCTL.BUSE = 1, the device will act as slave.
Without setting MSDVAD, setting MSDCTm.RBCR[7:6] = 01b (Device Role Master), and setting BCTL.BUSE = 1, the device will act as slave.

26.2.5 Reset Control Register (RSTCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INTLRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RSQRST	IBIQRST	RDBRST	TDBRST	RSPQRST	CMDQRST	RI3CRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
16	INTLRST	0b	R/W	Internal Software Reset 0: Releases of some registers and internal state. 1: Resets of some registers and internal state.
15 to 13	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
12 to 9	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
8, 7	—	00b	R	Reserved These bits are read as 0. The write value should be 0.
6	RSQRST	0b	R/W	Receive Status Queue Software Reset*2 0: The Receive Status Queue in I3C is not flushed. 1: The Receive Status Queue in I3C is flushed.
5	IBIQRST	0b	R/W	IBI Queue Software Reset*1 0: The IBI Queues in I3C is not flushed. 1: The IBI Queues in I3C is flushed.
4	RDBRST	0b	R/W	Receive Data Buffer Software Reset*1 0: The Receive Queues in I3C is not flushed. 1: The Receive Queues in I3C is flushed.
3	TDBRST	0b	R/W	Transmit Data Buffer Software Reset*1 0: The Transmit Queues in I3C is not flushed. 1: The Transmit Queues in I3C is flushed.
2	RSPQRST	0b	R/W	Response Queue Software Reset*1 0: The Response Queues in I3C is not flushed. 1: The Response Queues in I3C is flushed.
1	CMDQRST	0b	R/W	Command Queue Software Reset*1 0: The Command Queues in I3C is not flushed. 1: The Command Queues in I3C is flushed.
0	RI3CRST	0b	R/W	I3C Software Reset 0: Release I3C reset. 1: Initiate I3C reset.

Note 1. This bit supports for all I3C mode.

Note 2. This bit supports for I3C secondary master mode and I3C slave mode.

For details on reset for each register, refer to **Section 26.5, Reset Descriptions**.

RI3CRST bit (I3C Software Reset)

On Driver setting this bit to 1, I3C shall be reset and disabled.

All registers shall return to their reset values, and the software shall re-initialize I3C.

This field is cleared automatically upon I3C reset completion. This field also resets all Queues in I3C.

CMDQRST bit (Command Queue Software Reset)

On software setting this bit to 1, the Command Queues in I3C shall be flushed.

This field shall be cleared automatically upon Command Queue reset completion.

RSPQRST bit (Response Queue Software Reset)

On software setting this bit to 1, the Response Queues in I3C shall be flushed.

This field shall be cleared automatically upon Response Queue reset completion.

TDBRST bit (Transmit Data Buffer Software Reset)

On software setting this bit to 1, the Transmit Data Buffers in I3C shall be flushed.

This field shall be cleared automatically upon Transmit Data Buffer reset completion.

RDBRST bit (Receive Data Buffer Software Reset)

On software setting this bit to 1, the Receive Data Buffers in I3C shall be flushed.

This field shall be cleared automatically upon completion of Receive Data Buffer reset.

IBIQRST bit (IBI Queue Software Reset)

On software setting this bit to 1, the IBI Queues in I3C shall be flushed.

This field shall be cleared automatically upon completion of IBI Queue reset.

RSQRST bit (Receive Status Queue Software Reset)

On software setting this bit to 1, the Receive Status Queues in I3C shall be flushed. This field shall be cleared automatically upon Receive Status Queue reset completion.

INTLRST bit (Internal Software Reset)

When set to 1, some of registers is reset. For details on the registers to be reset, refer to **Section 26.5, Reset Descriptions**.

NOTE

Programming this field while it contains a value of 1 may result in undefined behavior.

26.2.6 Present State Register (PRSST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PRSSTWP	—	—	TRMD	—	CRMS	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	R	R	R	R	R/W*1	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7	PRSSTWP	0b	W	Present State Write Protect*2 0: CRMS bit is protected. 1: CRMS bit can be written when writing simultaneously with the value of the target bit.
6, 5	—	00b	R	Reserved These bits are read as 0. The write value should be 0.
4	TRMD	0b	R	Transmit/Receive Mode*3 0: Receive mode 1: Transmit mode
3	—	0b	R	Reserved These bits are read as 0. The write value should be 0.
2	CRMS	0b	R/W*1	Current Master*2 0: The Master is not the Current Master and must request and acquire bus ownership before initiating any transfer. 1: The Master is the Current Master, and as a result can initiate transfers.
1, 0	—	00b	R	Reserved These bits are read as 0. The write value should be 0.

Note 1. When the PRSSTWP bit is set to 1, the CRMS bit can be written to.

Note 2. This bit supports for I^2C, I3C master, and I3C secondary master mode.

Note 3. This bit supports for I^2C mode.

CRMS bit (Current Master)

Indicates the set condition and reset condition of each operation mode.

Operation Mode [I^2C/I3C common]

[Clearing conditions]

- When 1 written to the RSTCTL.RI3CRST by the software.
- When 1 written to the RSTCTL.INTLRST by the software.
- When 0 written to the PRSST.CRMS by the software.
- When STOP is issued.
- When Master Arbitration-Lost.

[Setting conditions]

- When 1 written to the PRSST.CRMS by the software.
- When START is issued.

Operation Mode [I3C Main Master]

[Clearing conditions]

- When 0 written to the MSDVAD.MDYADV by the software.
- When GETACCMST transmission is successfully completed by issuing STOP, after responding ACK to the Mastership-Request received from the Secondary Master.

[Setting conditions]

- When 1 written to the MSDVAD.MDYADV by the software.
- When GETACCMST reception is successfully completed by issuing STOP, after the ACK is responded to the Mastership-Request transmitted to the Secondary Master.

Operation Mode [I3C Secondary Master]

[Clearing condition]

- When GETACCMST transmission is successfully completed by issuing STOP, after responding ACK to the Mastership-Request received from the Non-Current Master.

[Setting condition]

- When GETACCMST reception is successfully completed by issuing STOP, after the ACK is responded to the Mastership-Request transmitted to the Current Master.

The PRSST register returns I3C current state.

State has two parts: this register, which is mandatory, and an additional optional PRSST_DEBUG register intended for debug purposes (see the Debug Capability Registers in the Extended Capabilities list).

TRMD bit (Transmit/Receive Mode*3)

This bit indicates transmit or receive mode.

I3C is in receive mode when the TRMD bit is set to 0 and is in transmit mode when the bit is set to 1. Combination of this bit and the CRMS bit indicates the operating mode of I3C.

The value of TRMD bit is automatically changed to 1 for transmit mode or 0 for receive mode by issuing or detection of a START condition and setting of the R/W# bit.

[Setting conditions]

- When a START condition is issued normally according to the START condition issuance request (when a START condition is detected with the CNDCTL.STCND bit set to 1).
- When a Repeated START condition is issued normally according to the Repeated START condition issuance request (when a Repeated START condition is detected with the CNDCTL.SRCND bit set to 1).
- When the R/W# bit added to the slave address is set to 0 in master mode.
- When the address received in slave mode matches the address enabled in SVCTL, with the R/W# bit set to 1.

[Clearing conditions]

- When a STOP condition is detected.
- The ALF (arbitration-lost) flag in BST being set to 1.

- In master mode, reception of a slave address to which an R/W# bit with the value 1 is appended.
- In slave mode, a match between the received address and the address enabled in SVCTL when the value of the received R/W# bit is 0 (including cases where the received address is the general call address).
- In slave mode, a Repeated START condition is detected (a Repeated START condition is detected with BCST.BFREF = 0 and CRMS = 0).

PRSSTWP bit (Present State Write Protect)

PRSSTWP is always 0 when reading.

When writing to PRSST, writing 1 to this bit at the same time enables writing to CRMS bit.

26.2.7 Internal Status Register (INST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	INEF	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W*1	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
10	INEF	0b	R/W*1	Internal Error Flag 0: I3C Internal Error has not detected. 1: I3C Internal Error has detected.
9 to 0	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.

Note: This register supports for all I3C mode.

Note 1. Clearing (to 0) condition: Writing 0 after the 1 state is read.

The Interrupt Status register reflects the status of outstanding interrupt (s).

The status fields are either RW0C (write 0 to clear), or else are cleared based on queue operations.

INEF bit (Internal Error Flag)

When this bit is 1, it indicates that I3C Internal Error has detected.

When this bit is 0, it indicates that I3C Internal Error has not detected.

[Setting conditions]

- The following 1 is satisfied and any of the following 2 to 9 are satisfied.

1. The INSTE.INEE bit = 1
2. When transmit data is written to the Transmit Data Buffer that is completely full.
3. When received data is read from the Receive Data Buffer that is completely empty.
4. When Command Descriptor is written to the Command Queue that is completely full.
5. When Response Descriptor is read from the Response Status Queue that is completely empty.
6. When Receive Status Descriptor is read from the Receive Status Queue that is completely empty.
7. When IBI Status Descriptor is read from the IBI Queue under the condition that the IBI Queue is completely empty and PRSST.CRMS = 1.
8. When IBI Data is written to the IBI Queue under the condition that the IBI Queue is completely full and PRSST.CRMS = 0.
9. When the Response Status Queue, IBI Status Queue or Receive Status Queue overflows.

[Clearing condition]

- When 0 is written to the INEF bit after reading INEF bit = 1.

26.2.8 Internal Status Enable Register (INSTE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	INEE	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
10	INEE	0b	R/W	Internal Error Enable 0: Disable INST.INEF 1: Enable INST.INEF
9 to 0	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.

Note: This register supports for all I3C mode.

INEE bit (Internal Error Enable)

- When this bit set to 1, it enables detection of I3C Internal Error.
- When this bit set to 0, it disables detection of I3C Internal Error.

26.2.9 Internal Interrupt Enable Register (INIE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	INEIE	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
10	INEIE	0b	R/W	Internal Error Interrupt Enable 0: Disables Non-recoverable Internal Error Interrupt Signal. 1: Enables Non-recoverable Internal Error Interrupt Signal.
9 to 0	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.

Note: This register supports for all I3C mode.

INEIE bit (Internal Error Interrupt Enable)

When set to 1 and register INEF is set, the hardware Controller asserts an interrupt to the Host.

26.2.10 Internal Status Force Register (INSTFC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	INEFC	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	W	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
10	INEFC	0b	W	Internal Error Force 0: Not force a specific interrupt 1: Force a specific interrupt
9 to 0	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.

Note: This register supports for all I3C mode.

INEFC bit (Internal Error Force)

For debug, helps to force this interrupt.

26.2.11 Device Characteristic Table Register (DVCT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	IDX[4:0]					—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are read as 0.
23 to 19	IDX[4:0]	All 0	R	DCT Table Index Current index of the DCT, which is used as the starting index for the I3C ENTDAA CCC.
18 to 0	—	All 0	R	Reserved These bits are read as 0.

Note: This register supports for I3C master mode and I3C secondary master mode.

IDX[4:0] bits (DCT Table Index)

Once the complete characteristics of device that won the arbitration are written to the DCT (during ENTDAA using Address Assignment Command) this index is incremented by 1.

Notes: How to check the progress of ENTDAA using this bit:

1. Read the value of this bit before setting the Command Descriptor for issuing the ENTDAA command.
2. After starting the ENTDAA command, until the value of this bit is updated (that is, it changes from the value read in advance), it indicates that the Dynamic Address is being assigned to the device specified by the first index value (value set in DEV_INDEX [4:0] of Command Descriptor).
3. After the value of this bit is updated, it indicates that Dynamic Address is being assigned according to the value set in DEV_INDEX [4:0] and DEV_COUNT [3:0] of Command Descriptor to the device of the first index value or later.

26.2.12 IBI Notify Control Register (IBINCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	NRSIRCTL	—	NRMRCTL	NRHJCTL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
3	NRSIRCTL	0b	R/W	Notify Rejected Slave Interrupt Request Control 0: Do not pass rejected IBI Status to the IBI Queue/Rings, if the incoming SIR is NACKed and is auto disabled based on DVSIRRJ field in relevant DAT entry. 1: Pass rejected IBI Status to the IBI Queue/Rings, if the incoming SIR is NACKed and is auto disabled based on DVSIRRJ field in relevant DAT entry.
2	—	0b	R	Reserved This bit is read as 0. The write value should be 0.
1	NRMRCTL	0b	R/W	Notify Rejected Master Request Control 0: Do not pass rejected IBI Status to IBI Queue/Ring, if the incoming Master Request is NACKed and is auto disabled based on DVMRRJ field in relevant DAT entry. 1: Pass rejected IBI Status to the IBI Queue, if the incoming Master Request is NACKed and is autodisabled based on DVMRRJ field in relevant DAT entry.
0	NRHJCTL	0b	R/W	Notify Rejected Hot-Join Control 0: Do not pass rejected IBI Status to IBI Queue, if the incoming HotJoin request is NACKed and is autodisabled based on field HJACKCTL of BCTL. 1: Pass rejected IBI Status to the IBI Queue, if the incoming Hot Join request is NACKed and is autodisabled based on field HJACKCTL of BCTL.

Note: This register supports for I3C master mode and I3C secondary master mode.

NRHJCTL bit (Notify Rejected Hot-Join Control)

Enables or disables reporting rejection of individual Hot Join requests.

NRMRCTL bit (Notify Rejected Master Request Control)

Enables or disables reporting rejection of individual Master Requests.

NRSIRCTL bit (Notify Rejected Slave Interrupt Request Control)

Enables or disables reporting rejection of individual Slave Interrupt Requests (SIR).

26.2.13 Bus Function Control Register (BFCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HSME	FMPE	—	—	—	—	—	SCSYNE	—	—	—	—	—	SALE	NALE	MALE
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R	R	R	R	R	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
15	HSME	0b	R/W	High Speed Mode Enable*1 0: Disable High Speed Mode. 1: Enable High Speed Mode.
14	FMPE	0b	R/W	Fast-mode Plus Enable*1 0: No Fm+ slope control circuit uses for the SCLn pin and SDAn pin. (n = 0) 1: An Fm+ slope control circuit uses for the SCLn pin and SDAn pin. (n = 0)
13	—	0b	R	Reserved This bit is read as 0. The write value should be 0.
12	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
11 to 9	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
8	SCSYNE	1b	R/W	SCL Synchronous Circuit Enable*1 0: No SCL synchronous circuit uses. 1: An SCL synchronous circuit uses.
7 to 3	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
2	SALE	0b	R/W	Slave Arbitration-Lost Detection Enable*1 0: Slave arbitration-lost detection disables. 1: Slave arbitration-lost detection enables.
1	NALE	0b	R/W	NACK Transmission Arbitration-Lost Detection Enable*1 0: NACK transmission arbitration-lost detection disables. 1: NACK transmission arbitration-lost detection enables.
0	MALE	1b	R/W	Master Arbitration-Lost Detection Enable*1 0: Master arbitration-lost detection disables. Disables the arbitration-lost detection function and does not clear the CRMS and TRMD bits in PRSST automatically when arbitration is lost. 1: Master arbitration-lost detection enables. Enables the arbitration-lost detection function and clears the CRMS and TRMD bits in PRSST automatically when arbitration is lost.

Note 1. This bit supports for I^2C mode.

MALE bit (Master Arbitration-Lost Detection Enable)

This bit is used to specify whether to use the arbitration-lost detection function in master mode. Normally, set this bit to 1.

NALE bit (NACK Transmission Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when ACK is detected during transmission of NACK in receive mode (such as when slaves with the same address exist on the bus or when two or more masters select the same slave device simultaneously with different number of receive bytes).

SALE bit (Slave Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when a value different from the value being transmitted is detected on the bus in slave transmit mode (such as when slaves with the same address exist on the bus or when a mismatch with the transmit data occurs due to noise).

SCSYNE bit (SCL Synchronous Circuit Enable)

This bit is used to specify whether to synchronize the SCL clock with the SCL input clock. Normally, set this bit to 1.

When the SCSYNE bit set to 0 (no SCL synchronous circuit used), I3C does not synchronize the SCL clock with the SCL input clock. In this setting, I3C outputs the SCL clock with the transfer rate set in STDBR and EXTBR regardless of the SCLn line state. For this reason, if the bus load of the I^2C bus line is much larger than the specification value or if the SCL clock output overlaps in multiple masters, the short-cycle SCL clock that does not meet the specification may be output. When no SCL synchronous circuit uses, it also affects the issuance of a START condition, Repeated START condition, and STOP condition, and the continuous output of extra SCL clock cycles.

This bit must not be set to 0 except for checking the output of the set transfer rate.

FMPE bit (Fast-mode Plus Enable*1)

This bit is used to specify whether to use a slope control circuit for Fast mode Plus [Fm+].

When this bit is set to 1, a slope control circuit conforming to the Fast-mode Plus [Fm+] slope control specification (tof) of the I3C-bus is selected. When this bit is set to 0, a slope control circuit conforming to the Standard-mode [Sm] and Fast- mode [fm] slope control specification (tof) of the I3C-bus is selected.

Set this bit to 1 when using the transmission rate within a range up to 1 Mbps (Fast-mode Plus [Fm+]) of the I3C-bus specification. Set this bit to 0 when using the transmission rate at other rates (up to 100 kbps [Sm], up to 400 kbps [Fm]).

Note: When communicating in Hs-mode, set as follows.
- Set FMPE to 0 when sending Hs-mode master code (0000_1XXXb) with Fast-mode.
- Set FMPE to 1 when sending Hs-mode master code (0000_1XXXb) with Fast-mode Plus.

HSME bit (High Speed Mode Enable)

This bit is used for communicating in Hs-mode.

When this bit is set to 1, the Hs-mode master code is recognized, and Hs-mode communication is possible.

After the START condition is detected, if Hs-mode master code (0000 1XXXb) transmission is recognized, Hs-mode communication starts from Repeated START after receiving the NACK response.

It communicates at the bit rate set in STDBR until the NACK response, and automatically switches from Repeated START condition issuance after receiving the NACK response to the bit rate set in EXTBR.

Hs-mode continues until a STOP condition is detected.

When the STOP condition is detected, the bit rate is automatically switched to the bit rate set in STDBR.

Note: When this bit is set to 1, the BST.NACKDF bit will not be set even if a NACK response is received after sending the Hs-mode master code.

26.2.14 Slave Control Register (SVCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	SVAE[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HOAE	—	—	—	—	—	—	—	—	DVIDE	HSMCE	—	—	—	—	GCAE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
18 to 16	SVAE[2:0]	000b	R/W	Slave Address Enable [2:0]*2 0: Slave [2:0] disables 1: Slave [2:0] enables
15	HOAE	0b	R/W	Host Address Enable*1 0: Host address detection disables. 1: Host address detection enables.
14 to 7	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
6	DVIDE	0b	R/W	Device-ID Address Enable*1 0: Device-ID address detection disables. 1: Device-ID address detection enables.
5	HSMCE	0b	R/W	Hs-mode Master Code Enable*1 0: Hs-mode Master Code Detection disables. 1: Hs-mode Master Code Detection enables.
4 to 1	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
0	GCAE	0b	R/W	General Call Address Enable*1 0: General call address detection disables. 1: General call address detection enables.

Note 1. This bit supports for I^2C mode.

Note 2. These bits support for I^2C, I3C secondary master, and I3C slave mode.

GCAE bit (General Call Address Enable) (y = 0 to 2)

This bit is used to specify whether to ignore the general call address (000_0000 + 0 (write): All 0) when it is received.

- When this bit is set to 1, if the received slave address matches the general call address, I3C recognizes the received slave address as the general call address independently of the slave addresses set in the SVDVADy.SVAD[9:0] bits (n = 0) and performs data receive operation.
- When this bit is set to 0, the received slave address is ignored even if it matches the general call address.

HSMCE bit (Hs-mode Master Code Enable)

This bit is used to specify whether to recognize and execute the Hs-mode master code (00001xxxb) is received in the first byte after a START condition is detected.

- When this bit is set to 1, if the received first byte matches the Hs-mode master code, IIC recognizes that the Hs-mode master code has been received.
  The first byte after Repeated START after NACK response to Hs-mode master code is recognized as a slave address and compared with the slave address set by SVDVADy.SVAD[9:0].
  If the addresses match, the transmission/reception operation continues according to the R/W# bit value.
  Hs-mode continues until a STOP condition is detected.
- When this bit is set to 0, I3C will ignore the pattern until a STOP condition is detected, even if it matches the Hs-mode master code.

Note: When this bit is set to 1, SCSTRCTL.ACKTWE bit must be set to 0 and SCSTRCTL.RWE bit must be set to 1.

DVIDE bit (Device-ID Address Enable)

This bit is used to specify whether to recognize and execute the Device-ID address when a device ID (1111 100) is received in the first byte after a START condition or Repeated START condition is detected.

- When this bit is set to 1, if the received first byte matches the Device-ID, I3C recognizes that the Device-ID address has been received. When the following R/W# bit is 0 (write), I3C recognizes the second and the following bytes as slave addresses and continues the receive operation.
- When this bit is set to 0, I3C ignores the received first byte even if it matches the Device ID address and recognizes the first byte as a normal slave address.

For details on the Device-ID address detection, refer to **Section 26.3.2.3(4)(c), Device-ID Address Detection [I2C mode]**.

SVAE[2:0] bits (Slave Address Enable [2:0]) (y = 0 to 2)

This bit is used to enable or disable the slave address set in the SVDVADy.SVAD[9:0] bits.

- When this bit is set to 1, the slave address set in the SVAD[9:0] bits is enabled and is compared with the received slave address.
- When this bit is set to 0, the slave address set in the SVAD[9:0] bits is disabled and is ignored even if it matches the received slave address.

26.2.15 Reference Clock Control Register (REFCKCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	IREFCKS[2:0]*1		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
2 to 0	IREFCKS[2:0] *1	000b	R/W	Internal Reference Clock Selection Selects the internal reference clock source (I3Cϕ) for I3C. 0 0 0: PCLKD/1 clock 0 0 1: PCLKD/2 clock 0 1 0: PCLKD/4 clock 0 1 1: PCLKD/8 clock 1 0 0: PCLKD/16 clock 1 0 1: PCLKD/32 clock 1 1 0: PCLKD/64 clock 1 1 1: PCLKD/128 clock

Note 1. Set the IREFCKS[2:0] bit to 000 in I3C mode.

26.2.16 Standard Bit Rate Register (STDBR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSBRPO	—	SBRHP[5:0]						—	—	SBRLP[5:0]					
Initial Value	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SBRHO[7:0]								SBRLO[7:0]							
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	DSBRPO	0b	R/W	Double the Standard Bit Rate Period for Open-Drain*4 0: The time period set for SBRHO[7:0] and SBRLO[7:0] is not doubled. 1: The time period set for SBRHO[7:0] and SBRLO[7:0] is doubled.
30	—	0b	R	Reserved This bit is read as 0. The write value should be 0.
29 to 24	SBRHP[5:0]	All 1	R/W	Standard Bit Rate High-Level Period Push-Pull*3 Count value of the high-level period of SCL clock
23, 22	—	00b	R	Reserved These bits are read as 0. The write value should be 0.
21 to 16	SBRLP[5:0]	All 1	R/W	Standard Bit Rate Low-level Period Push-Pull*2 Count value of the low-level period of SCL clock
15 to 8	SBRHO[7:0]	All 1	R/W	Count value of the High-level period of SCL clock*1
7 to 0	SBRLO[7:0]	All 1	R/W	Count value of the Low-level period of SCL clock*1

Note 1. These bits support for I2C, I3C master, and I3C secondary master mode.

Note 2. These bits support for I3C master mode and I3C secondary master mode.

Note 3. These bits support for all I3C mode.

Note 4. This bit supports for I2C, I3C master, and I3C secondary master mode.

The STDBR register sets the bit rate according to the operating speed.

- I2C mode: Bit rate setting when communicating with Standard-mode / Fast-mode / Fast-mode plus
- I3C master mode: Bit rate setting selected by mode bit of command descriptor
- I3C slave mode: I3C bit rate setting

The I2C transfer rate and the SCL clock duty are calculated using the following expression.

Transfer rate = 1 / {[(High-Level Period + α*1) + (Low-Level Period + α)] / I3Cϕ*2 + SCLn line rising time [tr]*3 + SCLn line falling time [tf]*3}

Duty cycle = {SCLn line rising time [tr] + (High-Level Period + α) / I3Cϕ} / {SCLn line falling time [tf] + (Low-Level Period + α) / I3Cϕ}

Note 1. α depend on the number of stages in the noise filter.

Note 2. I3Cϕ = PCLKD × Division ratio

Note 3. The SCLn line rising time [tr] and SCLn line falling time [tf] depend on the total bus line capacitance [Cb] and the pull-up resistor [Rp]. For details, see the I2C-bus specification from NXP Semiconductors.

The I3C transfer rate and the SCL clock duty are calculated using the following expression.

Transfer rate = 1 / [(High-Level Period + Low-Level Period) / I3Cϕ + SCLn line rising time [tr] + SCLn line falling time [tf]]

Duty cycle = [SCLn line rising time [tr] + High-Level Period / I3Cϕ] / [SCLn line falling time [tf] + Low-Level Period / I3Cϕ]

SBRLO[7:0] bits (Count value of the Low-level period of SCL clock)

The SBRLO[7:0] bits are used to set the low-level period of SCL clock in Open-Drain mode.

I3C counts the low-level period with the internal reference clock source (I3Cϕ) specified by the REFCKCTL.IREFCKS[2:0] bits. It also works to generate the data setup time for automatic SCL low-hold operation (refer to **Section 26.3.2.3(6), Clock Stretching [I2C mode]**); when I3C is used in I^2C slave mode, these bits need to be set to a value longer than the data setup time*1.

If the digital noise filter is enabled (INCTL.DNFE = 1), set the SBRLO[7:0] bits to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the INCTL.DNFS[3:0] bits.

Note 1. Data setup time (tSU: DAT)
250 ns (up to 100 kbps: Standard-mode [Sm])
100 ns (up to 400 kbps: Fast-mode [Fm])
50 ns (up to 1 Mbps: Fast-mode plus [Fm+])
10 ns (up to 3.4 Mbps: Hs-mode [HS])

SBRHO[7:0] bits (Count value of the High-level period of SCL clock)

The SBRHO[7:0] bits use to set the high-level period of SCL clock in Open-Drain mode. SBRHO[7:0] bits are valid in master mode. If I3C is used only in I^2C slave mode, these bits need not to set the high-level period.

I3C counts the high-level period with the internal reference clock source (I3Cϕ) specified by the REFCKCTL.IREFCKS[2:0] bits.

If the digital noise filter is enabled (the INCTL.DNFE bit = 1), set the SBRHO[7:0] bits to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the INCTL.DNFS[3:0] bits.

SBRLP[5:0] bits (Standard Bit Rate Low-level Period Push-Pull)

SBRLP[5:0] bits are used to set the low-level period of SCL clock in Push-Pull.

I3C counts the low-level period with the internal reference clock source (I3Cϕ) specified by the REFCKCTL.IREFCKS[2:0] bits.

If the digital noise filter is enabled (the INCTL.DNFE bit = 1), set the SBRLP[5:0] bits to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the INCTL.DNFS[3:0] bits.

SBRHP[5:0] bits (Standard Bit Rate High-Level Period Push-Pull)

SBRHP[5:0] bits is used to set the high-level period of SCL clock in Push-Pull mode.

SBRHP[5:0] bits are valid in master mode. If I3C is used only in I^2C slave mode, these bits need not to set the high-level period.

I3C counts the high-level period with the internal reference clock source (I3Cϕ) specified by the REFCKCTL.IREFCKS[2:0] bits.

If the digital noise filter is enabled (the INCTL.DNFE bit = 1), set the SBRHP[5:0] bits to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the INCTL.DNFS[3:0] bits.

DSBRPO bit (Double the Standard Bit Rate Period for Open-Drain)

When DSBRPO = 1, double the high-level period that is set in SBRHO[7:0] and double the low-level period that is set in SBRLO[7:0].

Table 26.5 Requirement and Usage of Setting in Each Mode

	Device Mode				
Bit Name	I^2C Master	I^2C Slave	I3C Master	I3C Secondary Master	I3C Slave
SBRHP[5:0]	do not use	do not use	Setting required*3	Setting required*4	do not use
SBRLP[5:0]	do not use	do not use	Setting required*3	Setting required*5	do not use
SBRHO[7:0]	Setting required*1	do not use	Setting required*3	Setting required*5	do not use
SBRLO[7:0]	Setting required*1	Setting required*2	Setting required*3	Setting required*5	do not use

Note 1. The setting value is used for the data rate of ST, FM, and FM+ mode.

Note 2. The setting value is used for the data setup time of automatic SCL low-hold operation.

Note 3. The setting value is used for the data rate of each communication.

Note 4. When operating with I3C Master, the setting value is used for the data rate of each communication.

Note 5. When operating with I3C Master, the setting value is used for the data rate of each communication.
When operating with I3C Slave, do not use.

26.2.17 Extended Bit Rate Register (EXTBR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	EBRHP[5:0]						—	—	EBRLP[5:0]					
Initial Value	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EBRHO[7:0]								EBRLO[7:0]							
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00b	R	Reserved These bits are read as 0. The write value should be 0.
29 to 24	EBRHP[5:0]	All 1	R/W	Extended Bit Rate Low-Level Period Push-Pull*2 Count value of the high-level period of SCL clock
23, 22	—	00b	R	Reserved These bits are read as 0. The write value should be 0.
21 to 16	EBRLP[5:0]	All 1	R/W	Extended Bit Rate Low-Level Period Push-Pull*2 Count value of the low-level period of SCL clock
15 to 8	EBRHO[7:0]	All 1	R/W	Extended Bit Rate High-Level Period Open-Drain*1 Count value of the high-level period of SCL clock
7 to 0	EBRLO[7:0]	All 1	R/W	Extended Bit Rate Low-Level Period Open-Drain*1 Count value of the low-level period of SCL clock

Note 1. These bits support for I^2C, I3C master, and I3C secondary master mode.

Note 2. These bits support for I3C master mode and I3C secondary master mode.

The EXTBR register sets the bit rate according to the operating speed.

- I^2C mode: Bit rate setting for communicating in high-speed mode
- I3C master mode: Bit rate setting selected by mode bit of command descriptor
- I3C slave mode: unused

EBRLO[7:0] bits (Extended Bit Rate Low-Level Period Open-Drain)

Refer to SBRLO[7:0] bits of **Section 26.2.16, Standard Bit Rate Register (STDBR)** for details. Watch SBRHO, SBRLO as EBRHO[7:0], EBRLO[7:0].

EBRHO[7:0] bits (Extended Bit Rate High-Level Period Open-Drain)

Refer to SBRHO[7:0] bits of **Section 26.2.16, Standard Bit Rate Register (STDBR)** for details. Watch SBRHO, SBRLO as EBRHO[7:0], EBRLO[7:0].

EBRLP[5:0] bits (Extended Bit Rate Low-Level Period Push-Pull)

Refer to SBRLP[5:0] bits of **Section 26.2.16, Standard Bit Rate Register (STDBR)** for details. Watch SBRHP, SBRLP as EBRHP[5:0], EBRLP[5:0].

EBRHP[5:0] bits (Extended Bit Rate Low-Level Period Push-Pull)

Refer to SBRHP[5:0] bits of **Section 26.2.16, Standard Bit Rate Register (STDBR)** for details. Watch SBRHP, SBRLP as EBRHP[5:0], EBRLP[5:0].

Table 26.6 Requirement and Usage of Setting in Each Mode

Bit Name	Device Mode				
	I^2C Master	I^2C Slave	I3C Master	I3C Secondary Master	I3C Slave
EBRHP[5:0]	do not use	do not use	Setting required*2	Setting required*3	do not use
EBRLP[5:0]	do not use	do not use	Setting required*2	Setting required*3	do not use
EBRHO[7:0]	Setting required*1	do not use	Setting required*2	Setting required*3	do not use
EBRLO[7:0]	Setting required*1	do not use	Setting required*2	Setting required*3	do not use

Note 1. The setting value is used for the data rate of High-Speed mode.

Note 2. The setting value is used for the data rate of each communication.

Note 3. When operating with I3C Master, the setting value is used for the data rate of each communication.
When operating with I3C Slave, do not use.

26.2.18 Bus Free Condition Detection Time Register (BFRECDT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	FRECYC[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
8 to 0	FRECYC[8:0]	All 0	R/W	Bus Free Condition Detection Cycle The count value is a period for detecting the Bus free condition.

FRECYC[8:0] bits (Bus Free Condition Detection Cycle)

I3C counts the period for detecting the Bus free condition with the I3Cϕ.

These bits set the Bus Free period. This Bus Free period is counted by the internal reference clock (I3Cϕ) selected by the REFCKCTL.IREFCKS[2:0] bits. See the BCST.BFREF flag for Bus Free detection behavior.

26.2.19 Bus Available Condition Detection Time Register (BAVLCDT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AVLCYC[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
8 to 0	AVLCYC[8:0]	All 0	R/W	Bus Available Condition Detection Cycle The count value is a period for detecting the Bus available condition.

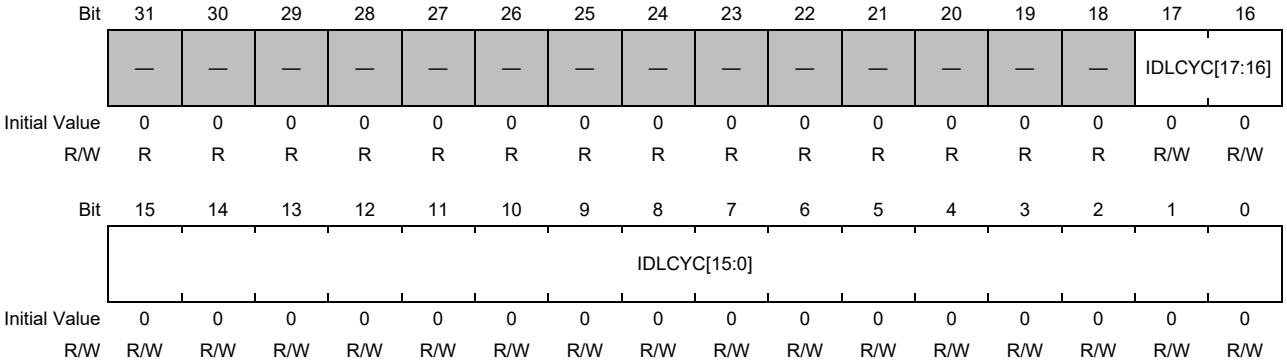
Note: This register supports for all I3C mode.

AVLCYC[8:0] bits (Bus Available Condition Detection Cycle)

I3C counts the period for detecting the Bus available condition with the I3Cϕ.

These bits set the Bus Available period. This Bus Available period is counted by the internal reference clock (I3Cϕ) selected by the REFCKCTL.IREFCKS[2:0] bits. See the BCST.BAVLF flag for Bus Available detection behavior.

26.2.20 Bus Idle Condition Detection Time Register (BIDLCDT)



Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
17 to 0	IDLCYC[17:0]	All 0	R/W	Bus Idle Condition Detection Cycle The count value is a period for detecting the Bus idle condition.

Note: This register supports for all I3C mode.

IDLCYC[17:0] bits (Bus Idle Condition Detection Cycle)

I3C counts the period for detecting the Bus idle condition with the I3Cϕ.

These bits set the Bus Idle period. This Bus Idle period is counted by the internal reference clock (I3Cϕ) selected by the REFCKCTL.IREFCKS[2:0] bits. See the BCST.BIDLF flag for Bus Available detection behavior.

26.2.21 Output Control Register (OUTCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDODCS	—	—	—	—	SDOD[2:0]			—	—	—	EXCYC	—	SOCWP	SCOC	SDOC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R/W	R	R	R	R	R/W	R/W	R/W	R	R	R	R/W	R	W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
15	SDODCS	0b	R/W	SDA Output Delay Clock Source Selection*3 0: The internal reference clock (I3Cϕ) is selected as the clock source of the SDA output delay counter. 1: The internal reference clock divided by 2 (I3Cϕ/2) is selected as the clock source of the SDA output delay counter.*4
14 to 11	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
10 to 8	SDOD[2:0]	0b	R/W	SDA Output Delay*2 000: No output delay 001: 1 I3Cϕ cycle (When OUTCTL.SDODCS = 0 (I3Cϕ)) 1 or 2 I3Cϕ cycles (When OUTCTL.SDODCS = 1 (I3Cϕ/2)) 010: 2 I3Cϕ cycles (When OUTCTL.SDODCS = 0 (I3Cϕ)) 3 or 4 I3Cϕ cycles (When OUTCTL.SDODCS = 1 (I3Cϕ/2)) 011: 3 I3Cϕ cycles (When OUTCTL.SDODCS = 0 (I3Cϕ)) 5 or 6 I3Cϕ cycles (When OUTCTL.SDODCS = 1 (I3Cϕ/2)) 100: 4 I3Cϕ cycles (When OUTCTL.SDODCS = 0 (I3Cϕ)) 7 or 8 I3Cϕ cycles (When OUTCTL.SDODCS = 1 (I3Cϕ/2)) 101: 5 I3Cϕ cycles (When OUTCTL.SDODCS = 0 (I3Cϕ)) 9 or 10 I3Cϕ cycles (When OUTCTL.SDODCS = 1 (I3Cϕ/2)) 110: 6 I3Cϕ cycles (When OUTCTL.SDODCS = 0 (I3Cϕ)) 11 or 12 I3Cϕ cycles (When OUTCTL.SDODCS = 1 (I3Cϕ/2)) 111: 7 I3Cϕ cycles (When OUTCTL.SDODCS = 0 (I3Cϕ)) 13 or 14 I3Cϕ cycles (When OUTCTL.SDODCS = 1 (I3Cϕ/2))
7 to 5	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
4	EXCYC	0b	R/W	Extra SCL Clock Cycle Output*3 The EXCYC bit is cleared automatically after one clock cycle is output. 0: Does not output an extra SCL clock cycle (default). 1: Outputs an extra SCL clock cycle.
3	—	0b	R	Reserved This bit is read as 0. The write value should be 0.
2	SOCWP	0b	W	SCL/SDA Output Control Write Protect*1 0: Bits SCOC and SDOC are protected. 1: Bits SCOC and SDOC can be written (When writing simultaneously with the value of the target bit). This bit is read as 0.

Bit	Bit Name	Initial Value	R/W	Description
1	SCOC	1b	R/W	SCL Output Control*1 High level output is achieved through an external pull-up resistor. 0: I3C drives the SCLn pin low. 1: I3C releases the SCLn pin.
0	SDOC	1b	R/W	SDA Output Control*1 0: I3C drives the SDAn pin low. 1: I3C releases the SDAn pin.

Note 1. This bit supports for I^2C, I3C master, and I3C secondary master mode.

Note 2. These bits support for I^2C mode.

Note 3. This bit supports for I^2C mode.

Note 4. The setting SDODCS = 1 (I3Cϕ/2) only becomes valid when SCL is at the low level. When SCL is at the high level, the setting SDODCS = 1 becomes invalid and the clock source becomes the internal reference clock (I3Cϕ).

SDOC bit (SDA Output Control) and SCOC bit (SCL Output Control)

These bits are used to directly control the SDAn and SCLn signals output from this module.

When writing to these bits, also write 1 to the SOCWP bit at the same time.

The result of setting these bits is input to I3C via the input buffer. When slave mode is selected, a START condition may be detected, and the bus may be released depending on the bit settings.

Do not rewrite these bits during a START condition, STOP condition, Repeated START condition, or during transmission or reception. Operation after rewriting under the above conditions is not guaranteed.

EXCYC bit (Extra SCL Clock Cycle Output)

This bit is used to output an extra SCL clock cycle for debugging or error processing.

Normally, set the bit to 0. Setting the bit to 1 in a normal communication state causes a communication error.

For details on this function, refer to **Section 26.3.2.3(11)(a), Extra SCL Clock Cycle Output Function**.

26.2.22 Input Control Register (INCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	DNFE	DNFS[3:0]			
Initial Value	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7, 6	—	11b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
5	—	0b	R	Reserved This bit is read as 0. The write value should be 0.
4	DNFE	1b	R/W	Digital Noise Filter Circuit Enable 0: No digital noise filter circuit is used. 1: A digital noise filter circuit is used.
3 to 0	DNFS[3:0]	All 0	R/W	Digital Noise Filter Stage Selection H'0: Noise of up to one I3Cϕ cycle is filtered out (singlestage filter). H'1: Noise of up to two I3Cϕ cycles is filtered out (2-stage filter). H'2: Noise of up to three I3Cϕ cycles is filtered out (3-stage filter). H'3: Noise of up to four I3Cϕ cycles is filtered out (4-stage filter). H'4: Noise of up to five I3Cϕ cycles is filtered out (5-stage filter). ⋮ H'F: Noise of up to sixteen I3Cϕ cycles is filtered out (16-stage filter).

Note: This register supports for I²C mode.

DNFS[3:0] bits (Digital Noise Filter Stage Selection)

These bits are used to select the number of stages in the digital noise filter.

For details on the digital noise filter function, refer to **Section 26.3.2.6(3), Digital Noise-Filter Circuits [I2C mode]**.

In I²C High Speed mode, the module changes the number of noise filter stage to a quarter of the number of noise filter stage automatically.

Note: Set the noise range to be filtered out by the noise filter within a range less than the SCLn line high-level period or low-level period. If the noise range is set to a value of (SCL clock width: high-level period or low-level period, whichever is shorter) - [1.5 internal reference clock (I3Cϕ) cycles] or more, the SCL clock is regarded as noise by the noise filter function of I3C, which may prevent I3C from operating normally.

Note: In I²C High Speed mode, the lower 2 bits of the DNFS[3:0] bits are ignored, and the number of filter stages for 1 to 4 stages is selected by the upper 2 bits.

26.2.23 Timeout Control Register (TMOCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TOMDS[1:0]		TOHCTL	TOLCTL	—	—	TODTS[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7, 6	TOMDS[1:0]	00b	R/W	Timeout Operation Mode Selection 00: Timeout is detected during the following conditions: • The bus is busy (BCST.BFREF = 0) in master mode. • I3C's own slave address is detected, and the bus is busy in slave mode. • The bus is free (BCST.BFREF = 1) while generation of a START condition is requested (CNDCTL.STCND = 1). 01: Timeout is detected while the bus is busy. 10: Timeout is detected while the bus is free. 11: Setting prohibited
5	TOHCTL	1b	R/W	Timeout H Count Control 0: Count is disabled while the SCLn line is at a high level. 1: Count is enabled while the SCLn line is at a high level.
4	TOLCTL	1b	R/W	Timeout L Count Control 0: Count is disabled while the SCLn line is at a low level. 1: Count is enabled while the SCLn line is at a low level.
3, 2	—	00b	R	Reserved These bits are read as 0. The write value should be 0.
1, 0	TODTS[1:0]	00b	R/W	Timeout Detection Time Selection 0 0: 16bit-timeout 0 1: 14bit-timeout 1 0: 8bit-timeout 1 1: 6bit-timeout

TODTS[1:0] bits (Timeout Detection Time Selection)

These bits are used to select for the timeout detection time when the timeout function is enabled (BSTE.TODE bit = 1).

- When these bits are set to 00b, the timeout detection internal counter functions as a 16-bit counter.
- When these bits are set to 01b, the counter functions as a 14-bit counter.
- When these bits are set to 10b, the counter functions as a 8-bit counter.
- When these bits are set to 11b, the counter functions as a 6-bit counter.

While the SCLn line is in the state that enables this counter as specified by bits TOHCTL and TOLCTL, the counter counts up in synchronization with the internal reference clock (I3Cϕ) as a count source.

For details on the timeout function, refer to **Section 26.3.2.4(3), Timeout Error Detection**.

TOLCTL bit (Timeout L Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCLn line is held low when the timeout function is enabled (BSTE.TODE = 1).

TOHCTL bit (Timeout H Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCLn line is held high when the timeout function is enabled (BSTE.TODE = 1).

TOMDS[1:0] bits (Timeout Operation Mode Selection)

These bits are used to select the detection condition for timeout when the timeout function is enabled.

NOTE

When working with I^2C Slave, during 10-bit address communication, the timeout count starts when the upper address match is detected.

26.2.24 Wake Up Unit Control Register (WUCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	WUFE	WUFSYNE	—	WUANFS	—	—	—	WUACKS
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7	WUFE	0b	R/W	Wake Up function enable 0: Wake-up function disables 1: Wake-up function enables. Do not set WUFE = 0 during WakeUp operation.
6	WUFSYNE	1b	R/W	Wake-Up function Synchronous Enable 0: This IP asynchronous circuit enable 1: This IP synchronous circuit enable
5	—	0b	R	Reserved These bits are read as 0. The write value should be 0.
4	WUANFS	0b	R/W	Wake-Up Analog Noise Filter Selection 0: Do n1ot add the Wake-Up analog filter. 1: Add the Wake-Up analog filter.
3 to 1	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
0	WUACKS	1b	R/W	Wake-Up Acknowledge Selection Choice of four response mode with a combination of RSTCTL.INTLRST bit and WUACKS bit. Shown in the **Table 26.7**.

Table 26.7 Wake-Up Mode

INTLRST	WUACKS	Operation Mode	Description
0	0	Normal Wake-Up mode 1	ACK response at 9th SCL and SCL low hold after at 9th SCL.
0	1	Normal Wake-Up mode 2	No ACK response immediately and SCL low hold between 8th and 9th SCL. Release SCL low hold and ACK response at 9th SCL.
1	0	Command recovery mode	ACK response at 9th SCL and not SCL low hold.
1	1	EEP response mode	NACK response at 9th SCL and not SCL low hold.

WUFSYNE bit (Wake-Up function PCLK synchronous enable bit)

This bit is used to switch between the PCLK synchronous operation and the PCLK asynchronous operation.

The bit is used in combination with the WUASYNF flag at Wake-Up effective function (WUCTL.WUFE bit = 1).

[When switching from the PCLK synchronous operation to the PCLK asynchronous operation]

This IP operation changes into the PCLK asynchronous operation during BCST.BFREF flag = “1”, when the WUASYNF flag set to “1” during WUFSYNE bit = “0”.

The reception can operate without depending on the state of operation of PCLK (With PCLK stopped) after it switches to the PCLK asynchronous operation (Wake-Up event detection operation).

[When switching from the PCLK asynchronous operation to the PCLK synchronous operation]

- This IP operation changes into the PCLK synchronous operation at the following conditions.
  (At the same timing when WUASYNF flag becomes “0”)
  - In the case Wake-Up event detects: right after WUFSYNE bit is set to “1”.
  - In the case Wake-Up event does not detect: when STOP condition is detected after WUFSYNE bit is set to “1”.

[Setting condition]

- When 1 is written to the WUFSYNE bit.
- WUCTL.WUFE = 0

[Clearing conditions]

- When 0 is written to the WUFSYNE bit.

26.2.25 Acknowledge Control Register (ACKCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	ACKTWP	ACKT	ACKR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
2	ACKTWP	0b	W	ACKT Write Protect 0: The ACKT bit are protected. 1: The ACKT bit can be written (when writing simultaneously with the value of the target bit). This bit is read as 0.
1	ACKT	0b	R/W	Acknowledge Transmission 0: A 0 is sent as the acknowledge bit (ACK transmission). 1: A 1 is sent as the acknowledge bit (NACK transmission).
0	ACKR	0b	R	Acknowledge Reception 0: A 0 is received as the acknowledge bit (ACK reception). 1: A 1 is received as the acknowledge bit (NACK reception).

Note: This register supports for I^2C mode.

ACKR bit (Acknowledge Reception)

This bit is used to store the acknowledge bit information received from the receive device in transmit mode.

[Setting condition]

- When 1 is received as the acknowledge bit with the PRSST.TRMD bit set to 1.

[Clearing condition]

- When 0 is received as the acknowledge bit with the PRSST.TRMD bit set to 1.

ACKT bit (Acknowledge Transmission)

[Setting condition]

- When 1 is written to the ACKT bit and 1 is written to the ACKTWP bit at the same time.

[Clearing conditions]

- When 0 is written to the ACKT bit and 1 is written to the ACKTWP bit at the same time.
- When a STOP condition is detected. (When a STOP condition is detected with the CNDCTL.SPCND bit set to 1.)

Note: Set the ACKT bit to 0 in I^2C Slave mode.

ACKTWP bit (ACKT Write Protect)

This bit is used to control the modification of the ACKT bit.

When changing the ACKT bit, setting this bit to 1 at the same time can change the ACKT bit.

When this bit is read, 0 is always read.

26.2.26 SCL Stretch Control Register (SCSTRCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RWE	ACKTWE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
1	RWE	0b	R/W	Receive Wait Enable 0: No WAIT (The period between ninth clock cycle and first clock cycle is not held low.) 1: WAIT (The period between ninth clock cycle and first clock cycle is held low.) Low hold is released by reading NTDTBP0NTDTBP.
0	ACKTWE	0b	R/W	Acknowledge Transmission Wait Enable 0: NTST.RDBFF0 is set at the rising edge of the ninth SCL clock cycle. (The SCLn line is not held low at the falling edge of the eighth clock cycle.) 1: NTST.RDBFF0 is set at the rising edge of the eighth SCL clock cycle. (The SCLn line is held low at the falling edge of the eighth clock cycle.) Low hold is released by writing a value to the ACKCTL.ACKT bit.

Note: This register supports for I^2C mode.

ACKTWE bit (Acknowledge Transmission Wait Enable)

This bit is used to select the NTST.RDBFF0 flag set timing in receive mode and also to select whether to hold the SCLn line low at the falling edge of the eighth SCL clock cycle.

- When ACKTWE = 0, the SCLn line is not held low at the falling edge of the eighth SCL clock cycle, and the NTST.RDBFF0 flag is set to 1 at the rising edge of the ninth SCL clock cycle.
- When ACKTWE = 1, the NTST.RDBFF0 flag is set to 1 at the rising edge of the eighth SCL clock cycle and the SCLn line is held low at the falling edge of the eighth SCL clock cycle. The low hold of the SCLn line is released by writing a value to the ACKCTL.ACKT bit.

After data is received with this setting, the SCLn line is automatically held low before the acknowledge bit is sent. This enables processing to send ACK (ACKCTL.ACKT = 0) or NACK (ACKCTL.ACKT = 1) according to receive data.

RWE bit (Receive Wait Enable)

This bit is used to control whether to hold the period between the ninth SCL clock cycle and the first SCL clock cycle low until the receive data buffer (NTDTBP0) is completely read each time single-byte data is received in receive mode.

- When RWE = 0, the receive operation is continued without holding the period between the ninth and the first SCL clock cycle low. When both the ACKTWE and RWE bits = 0, continuous receive operation is enabled with the double buffer.
- When RWE = 1, the SCLn line is held low from the falling edge of the ninth clock cycle until the NTDTBP0 value is read each time single-byte data is received.

This enables receive operation in byte units.

Note: When the value of the RWE bit is to be read, be sure to read the NTDTBP0 beforehand.

26.2.27 SCL Stalling Control Register (SCSTLCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ACKPE	PARPE	—	AAPE	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STLCYC[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	ACKPE	0b	R/W	ACK phase Enable Stall enable bit during ACK/NACK phase 0: Does not stall the SCL clock during the ACK/NACK phase. 1: Stall the SCL clock during the ACK/NACK phase.
30	PARPE	0b	R/W	Parity Phase Enable Stall enable bit in parity bit period 0: Does not stall the SCL clock during the parity bit period. 1: Stall the SCL clock during the parity bit period.
29	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
28	AAPE	0b	R/W	Assigned Address Phase Enable Enable bit that allows stall by the first bit at address assignment 0: Does not stall the SCL clock during the address assignment phase. 1: Stall the SCL clock during address assignment phase.
27 to 16	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
15 to 0	STLCYC [15:0]	All 0	R/W	Stalling Cycle Counter setting of stall period (I3Cϕ cycle). Common use for each phase.

Note: This register supports for I3C master mode and I3C secondary master mode.

When setting this register, follow *Chapter 5.1.2.5 Master Clock Stalling of MIPI I3C Spec V1.0*, and use it only when necessary, because of its negative impacts on bus performance.

STLCYC[15:0] bits (Stalling Cycle)

These bits set the SCL stall period. The SCL stall period is counted by the internal reference clock (I3Cϕ). This is a counter common to the enable bits of each phase.

AAPE bit (Assigned Address Phase Enable)

The master can stall SCL during the low period of the first bit of the assigned address phase of the Enter Dynamic Address Assignment CCC command. It can gain time in assigning dynamic address to the device based on the BCR and DCR of the slave. However, because the Dynamic Address Assignment procedure sends the dynamic address set in the DATBASm register in sequence, it is not necessary to set this bit and it is prohibited.

PARPE bit (Parity Phase Enable)

The parity bit of the transmission data of I3C write transfer can be used for SCL stalling to avoid underrun of the transmission data FIFO. However, when the transmission data FIFO of the I3C master becomes empty, SCL stalling is performed regardless of the setting of this bit, it is not necessary to set this bit and it is prohibited. It is necessary to set this bit when the I3C slave requires preparation time to receive data.

ACKPE bit (ACK phase Enable)

Determine the need to perform SCL stalling in the ACK/NACK phase based on the following criteria:

- It is necessary to set this bit when the I3C and I^2C slaves connected to the bus require preparation time to receive or transmit data.
- In legacy I^2C communication, if there is a possibility that the data FIFO of the I3C master might underrun or overflow, it is not necessary to set this bit because SCL Stalling is performed by FIFO Empty or Full regardless of the setting of this bit.
- Other than legacy I^2C communication, the data FIFO of I3C master might underrun or overflow, and if SCL stalling is required in ACK phase, this bit can be set. However, it is necessary to build the software so that the FIFO does not underrun or overflow due to the interrupt generated according to the FIFO threshold setting (NQTHCTL, NTBTHCTL, NRQTHCTL).
- When I3C master responds ACK/NACK to IBI, it is not necessary to set this bit because ACK/NACK response can be set in advance by BCTL.HJACK, DATBASm.DVMRRJ and DATBASm.DVS IRRJ.
- It is necessary to set this bit when the I3C slave connected to the bus requires preparation time to transmit data for Direct GET CCC.

26.2.28 Slave Transfer Data Length Register 0 (SVTDLG0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	STDLG[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	STDLG[15:0]	All 0	R/W	Slave Transfer Data Length Indicates the number of bytes to be transferred.
15 to 0	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.

Note: This register supports for I3C secondary master mode and I3C slave mode.

26.2.29 Synchronous Timing Control Register (STCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STOE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
0	STOE	0b	R/W	Synchronous Timing output enable. 0: Disable 1: Enable

26.2.30 Asynchronous Timing Control Register (ATCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CDIV[7:0]								—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
15 to 8	CDIV[7:0]	All 0	R/W	TCLK Counter Divide Setting
7 to 3	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
2 to 0	—	000b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

26.2.31 Asynchronous Timing Trigger Register (ATTRG)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ATSTRG
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
0	ATSTRG	0b	W	Asynchronous Timing Software Trigger 0 Write: do nothing. 1 Write: Software trigger (one-shot pulse) output. This bit is always read as 0

26.2.32 Asynchronous Timing Control Counter enable Register (ATCCNTE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ATCE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
0	ATCE	0b	R/W	Asynchronous Timing Counter Enable for MREF, MC2, SC1, SC2. 0: Disable 1: Enable

26.2.33 Condition Control Register (CNDCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	SPCND	SRCND	STCND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
2	SPCND	0b	R/W	STOP (P) Condition Issuance 0: Does not request to issue a STOP condition. 1: Requests to issue a STOP condition.
1	SRCND	0b	R/W	Repeated START (Sr) Condition Issuance 0: Does not request to issue a Repeated START condition. 1: Requests to issue a Repeated START condition.
0	STCND	0b	R/W	START (S) Condition Issuance 0: Does not request to issue a START condition. 1: Requests to issue a START condition.

Note: This register supports for I^2C mode.

STCND bit (START (S) Condition Issuance)

This bit is used to request transition to master mode and issuance of a START condition.

For details on the START condition issuance, refer to **Section 26.3.2.3(3), START Condition / Repeated START Condition / STOP Condition Issuing Function**.

[Setting condition]

- When 1 is written to the STCND bit

[Clearing conditions]

- When 0 is written to the STCND bit
- When a START condition has been issued (A START condition is detected)
- When the BST.ALF (arbitration-lost) flag is set to 1

Note: Set the STCND bit to 1 (START condition issuance request) when the BCST.BFREF flag is set to 1 (bus free state). Note that arbitration may be lost due to a START condition issuance error if the STCND bit is set to 1 (START condition issuance request) when the BFREF flag is set to 0 (bus busy state).

SRCND bit (Repeated START (Sr) Condition Issuance)

This bit is used to request that a Repeated START condition be issued in master mode.

When this bit is set to 1 to request to issue a Repeated START condition, a Repeated START condition is issued when the BFREF flag is set to 0 (bus busy state) and the PRSST.CRMS bit is set to 1 (master mode).

For details on the Repeated START condition issuance, refer to **Section 26.3.2.3(3), START Condition / Repeated START Condition / STOP Condition Issuing Function**.

[Setting condition]

- When 1 is written to the SRCND bit with the BCST.BFREF flag set to 0

[Clearing conditions]

- When 0 is written to the SRCND bit
- When a Repeated START condition has been issued (A Repeated START condition is detected)
- When the BST.ALF (arbitration-lost) flag is set to 1

Note: Do not set the SRCND bit to 1 while issuing a STOP condition.
If 1 (requests to issue a Repeated START condition) is written to the SRCND bit in slave mode, the Repeated START condition is not issued but the SRCND bit remains set to 1.
If the operating mode changes to master mode with the bit not being cleared, note that the Repeated START condition may be issued.

SPCND bit (STOP (P) Condition Issuance)

This bit is used to request that a STOP condition be issued in master mode.

When this bit is set to 1 to request to issue a STOP condition, a STOP condition is issued when the BCST.BFREF flag is set to 0 (bus busy state) and the PRSST.CRMS bit is set to 1 (master mode).

For details on the STOP condition issuance, refer to **Section 26.3.2.3(3), START Condition / Repeated START Condition / STOP Condition Issuing Function**.

[Setting condition]

- When 1 is written to the SPCND bit with the BCST.BFREF flag set to 0 and the PRSST.CRMS bit set to 1

[Clearing conditions]

- When 0 is written to the SPCND bit
- When a STOP condition has been issued (A STOP condition is detected)
- When the BST.ALF (arbitration-lost) flag is set to 1
- When a START condition and a Repeated START condition are detected

Note: Writing to the SPCND bit is not possible while the setting of the BCST.BFREF flag = 1 (bus free state).
Do not set the SPCND bit to 1 while a Repeated START condition is being issued.

26.2.34 Normal Command Queue Port Register (NCMDQP)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	n/a	All 0	W	Normal Command Queue Port

Note: This register supports for all I3C mode.

32-bit mailbox register NCMDQP contains a command descriptor structure that depends on the requested transfer type:

1. Address Assignment Command (refer to **Section 26.3.1.1(1), Address Assign Command**)
2. Immediate Data Transfer (refer to **Section 26.3.1.1(2), Immediate Transfer Command**)
3. Regular Data Transfer (refer to **Section 26.3.1.1(3), Regular Transfer Command**)
4. Write + Write/Read Combo Transfer (refer to **Section 26.3.1.1(4), Combo Transfer Command**)
5. Internal Control Command (refer to **Section 26.3.1.1(5), Internal Control Command**)

Within the command descriptor, DWORDs appear starting with the Least Significant DWORD, in order until the Most Significant DWORD.

26.2.35 Normal Response Queue Port Register (NRSPQP)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	n/a	All 0	R	Normal Response Queue Port

Note: This register supports for all I3C mode.

32-bit mailbox register NRSPQP contains a response structure (refer to **Section 26.3.1.4, Receive Status Descriptor**).

26.2.36 Normal Transfer Data Buffer Port Register 0 (NTDTBP0/NTDTBP0_BY)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	n/a	All 0	R/W	Normal Transfer Data Buffer Port NTDTBP0 is a 32-bit read/write register. NTDTBP0_BY (NTDTBP0[7:0]) is a 8-bit read/write register.

Note: NTDTBP0 is 32-bit access in I3C mode.
NTDTBP0_BY is 8-bit access in I^2C mode.

32-bit mailbox register NTDTBP0 is a 32-bit bi-directional data transfer register which is used both to read from the Normal Receive Data Buffer, and to write to the Normal Transmit Data Buffer.

In other words, the Normal Receive Data Buffer and the Normal Transmit Data Buffer have the same offset, forming a single bidirectional port for transmitting or receiving I3C data.

Read Operations:

[I3C protocol mode]

Data Read from the Normal Receive Data Buffer. Its should be read based on Normal Queue Status Level indications. The Receive data is always aligned to a 4-byte boundary and stored in the Normal Receive Data Buffer. If the length of the data transfer is not aligned to a 4-byte boundary, then there will be extra (unused) bytes at the end of the transferred data. The valid data must be identified using the DATA_LENGTH field in the Response Descriptor.

[I^2C protocol mode]

When 1 byte of data has been received, the received data is transferred from the internal shift register to NTDTBP0 to enable the next data to be received. The double-buffer structure of the internal shift register and NTDTBP0 allows continuous receive operation if the received data has been read from NTDTBP0 while the internal shift register is receiving data. Read data from NTDTBP0 once when a receive data full interrupt (I3Cn_RX) request is generated. If NTDTBP0 receives the next receive data before the current data is read from NTDTBP0 (while the RDBFF0 flag in NTST is 1), this module automatically holds the SCL clock low one cycle before the RDBFF0 flag is set to 1 next. The lower 8 bits of the read 32-bit data are valid as received data.

Write Operations:

[I3C protocol mode]

Data Written to the Normal Tx Data Buffer. Data DWORDs written to the Normal Transmit Data Buffer are placed onto the I3C bus one byte at a time, with the DWORD LSB first. Within each byte, bits are placed onto the I3C bus in big-endian order, with bit 7 going out first on the bus. The transmit data should always start aligned to a 4-byte boundary and written to the NTDTBP0 register. If the length of the transfer is not aligned to a 4-byte boundary, then

there will be extra (unused) bytes at the end of the transferred data. I3C shall only send the valid number of bytes indicated in the DATA_LENGTH field of the Command Descriptor.

[I^2C protocol mode]

When NTDTBP0 detects a space in the internal shift register, it transfers the transmit data that has been written to NTDTBP0 to the internal shift register and starts transmitting data in transmit mode. The double-buffer structure of NTDTBP0 and the internal shift register allows continuous transmit operation if the next transmit data has been written to NTDTBP0 while the the internal shift register data is being transmitted. Write transmit data to NTDTBP0 once when a transmit data empty interrupt (I3Cn_TX) request is generated. The lower 8 bits of the written 32-bit data are valid as transmission data.

26.2.37 Normal IBI Queue Port Register (NIBIQP)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	n/a	All 0	R/W	Normal IBI Queue Port

Note: This register supports for all I3C mode.

When receiving an IBI, 32-bit mailbox register NIBIQP is used for both:

- Read the IBI status descriptor (refer to **Section 26.3.1.3, IBI Status Descriptor**)
- Read the IBI data (which is raw/opaque data).

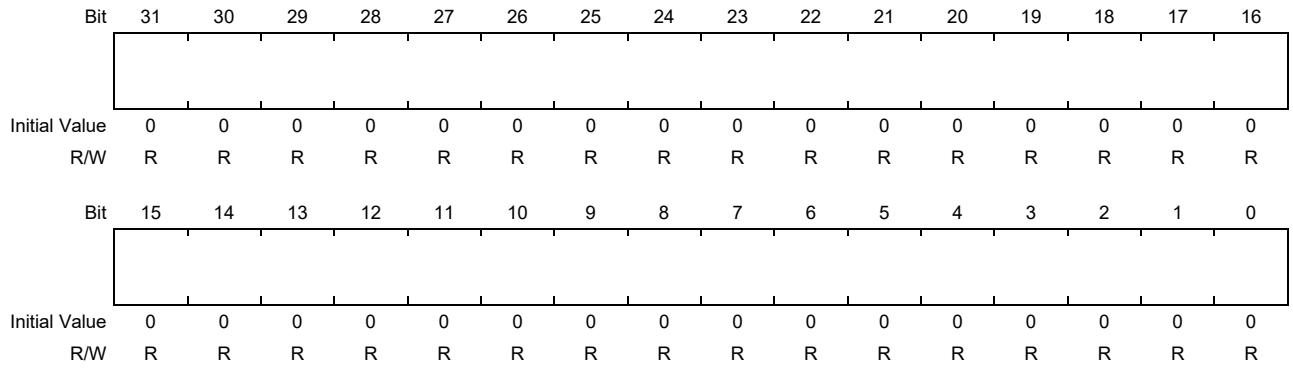
The IBI status descriptor is a read-only structure describing an IBI event received from a Slave device on the I3C bus.

NOTE

If the I3C HCI auto-read feature is used, then the IBI data includes the data received from the auto-generated private read operation.

Even if LAST_STATUS is set to 0, the driver software still evaluates the data payload length by examining the CHUNKS field.

26.2.38 Normal Receive Status Queue Port Register (NRSQP)



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	n/a	All 0	R	Normal Receive Status Queue Port

Note: This register supports for I3C secondary master mode and I3C slave mode.

32-bit mailbox register NRSQP contains a receive status structure (refer to **Section 26.3.1.4, Receive Status Descriptor**).

26.2.39 Normal Queue Threshold Control Register (NQTHCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IBIQTH[7:0]								IBIDSSZ[7:0]							
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSPQTH[7:0]								CMDQTH[7:0]							
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	IBIQTH[7:0]	H'01	R/W	Normal IBI Queue Threshold*1 H'00: I3C Protocol mode (Master): Interrupt is generated when the Outstanding IBI Status count is 1 or more. I3C Protocol mode (Slave): Interrupt is issued when IBI Data Buffer is completely empty. Others: I3C Protocol mode (Master): Interrupt is generated when the Outstanding IBI Status count is N + 1 or more. (N = CMDQTH[7:0]) I3C Protocol mode (Slave): Interrupt is issued when IBI Data Buffer contains N empties.
23 to 16	IBIDSSZ[7:0]	H'01	R/W	Normal IBI Data Segment Size*2 Supported Values: Minimum: 1 (4 bytes) Maximum: 63 (252 bytes), provided that the configured IBI Queue depth is 64 or more. When ATCCNTE.ATCE = 1, restrict to the number of slices ≥ 2.
15 to 8	RSPQTH[7:0]	H'01	R/W	Normal Response Queue Threshold*1 H'00: Interrupt is issued when Response Queue contains 1 entry (DWORD). Others: Interrupt is triggered when Response Queue contains N + 1 entries (DWORD). (N = CMDQTH[7:0])
7 to 0	CMDQTH[7:0]	H'01	R/W	Normal Command Ready Queue Threshold*1 H'00: Interrupt is issued when Command Queue is completely empty. Others: Interrupt is issued when Command Queue contains N empties. (N = CMDQTH[7:0])

Note: This register supports for I3C secondary master mode and I3C slave mode.

Note 1. These bits support for all I3C mode.

Note 2. These bits support for I3C master mode and I3C secondary master mode.

The Queue Threshold Control register controls the interrupt trigger thresholds for the Command Queue, the Response Queue, and the IBI Queue.

The specific reset values are indicative and could be hardware implementation specific.

CMDQTH[7:0] bits (Normal Command Ready Queue Threshold)

Controls the minimum number of Command Queue empties needed to trigger the INTCMD interrupt.

If this field is greater than (Command Queue size – 1), then only the number of bits required to address the full buffer depth will be considered.

Note: It is assumed that I3C has exactly one Command Queue, exactly one Response Queue, and exactly one IBI Queue.

RSPQTH[7:0] bits (Normal Response Queue Threshold)

Controls the minimum number of Response Queue entries needed to trigger the INTRESP interrupt.

If this field is greater than (Response Status Queue size – 1), then only the number of bits required to address the full buffer depth will be considered.

IBIDSSZ[7:0] bits (Normal IBI Data Segment Size)

This is the IBI data segment size, in DWORDs (4 bytes).

In PIO mode, this field allows the incoming IBI data to be sliced into multiple segments generating status individually, to support cutthrough readout of a long IBI payload data.

IBIQTH[7:0] bits (Normal IBI Queue Threshold)

For I3C protocol mode (Master): PRTS.PRTMD = 0 and PRSST.CRMS = 1.

Controls generation of the INTIBI interrupt, based on the value of the IBI Queue's Outstanding IBI status count.

Each IBI status entry can represent either the complete IBI payload (if the IBI payload byte size is 4×IBIDSSZ or less), or a segment of the IBI payload (if the IBI payload byte size is more than 4×IBIDSSZ).

For I3C protocol mode (Slave): PRTS.PRTMD bit = 0, PRSST.CRMS bit = 0.

Controls the minimum number of IBI Data Buffer empties needed to trigger the INTIBI interrupt.

If this field is greater than (IBI Data Buffer size? 1), then only the number of bits required to address the full buffer depth will be considered.

26.2.40 Normal Transfer Data Buffer Threshold Control Register 0 (NTBTHCTL0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RXSTTH[2:0]			—	—	—	—	—	TXSTTH[2:0]		
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RXDBTH[2:0]			—	—	—	—	—	TXDBTH[2:0]		
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
26 to 24	RXSTTH [2:0]*3	001b	R/W	Normal Rx Start Threshold*2 000: Wait for 2 empty DWORDs 001: Reserved Others: Setting prohibited
23 to 19	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
18 to 16	TXSTTH [2:0]*3	001b	R/W	Normal Tx Start Threshold*2 000: Wait for 2 DWORDs 001: Reserved Others: Setting prohibited
15 to 11	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
10 to 8	RXDBTH[2:0]	001b	R/W	Normal Receive Data Buffer Threshold*1 000: Interrupt triggers at 2 Rx Buffer entries, DWORDs 001: Reserved Others: Setting prohibited
7 to 3	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
2 to 0	TXDBTH[2:0]	001b	R/W	Normal Transmit Data Buffer Threshold*1 000: Interrupt triggers at 2 Tx Buffer empties, DWORDs 001: Reserved Others: Setting prohibited

Note 1. These bits support for all I3C mode.

Note 2. These bits support for I3C master mode and I3C secondary master mode.

Note 3. The RXSTTH[2:0] and TXSTTH[2:0] bits are only present in NTBTHCTL0.

The Data Buffer Control register controls the interrupt trigger thresholds for the Receive Data Buffer Queue and the Transmit Data Buffer Queue.

TXDBTH[2:0] bits (Normal Transmit Data Buffer Threshold)

Minimum number of Transmit FIFO empties, in DWORDs, that will trigger the INTTXn interrupt.

The software must program a value less than Transmit Data Buffer size in this register.

RXDBTH[2:0] bits (Normal Receive Data Buffer Threshold)

Minimum number of Receive FIFO entries in DWORDs that will trigger the INTRXn interrupt.

The software must program a value less than Receive Data Buffer size in this register.

TXSTTH[2:0] bits (Normal Tx Start Threshold)

When preparing to initiate a Write Transfer on the I3C Bus, I3C shall wait until the Transmit Buffer has at least the indicated number of locations available.

Two optional configurable Modes are available:

1. Store and Forward Mode
   If the TXSTTH[2:0] field is set to the Transmit Buffer size, then I3C shall delay initiation of the Write Command as follows:
   - If the data length to be transferred is more than the Transmit Buffer size, then this module shall wait until the Transmit FIFO is completely full.
   - If the data length to be transferred is less than the Transmit Buffer size, then I3C shall wait until enough Transmit FIFO locations are available to store the data to be transferred.
2. Threshold Mode
   If the TXSTTH[2:0] field value is less than the Transmit Buffer size, then I3C shall initiate the Write Command as soon as the indicated number of Transmit FIFO locations are entries.

RXSTTH[2:0] bits (Normal Rx Start Threshold)

When preparing to initiate a Read Transfer on the I3C bus, I3C shall wait until the Receive Buffer has at least the indicated number of empty locations in DWORDs.

Two optional configurable Modes are available:

1. Store and Forward Mode If the RXSTTH[2:0] field is set to the Receive Buffer size, then I3C shall delay initiation of the Read Command as follows:
   - If the data length to be transferred is more than the Receive Buffer size, then this module shall wait until the Receive FIFO is completely empty.
   - If the data length to be transferred is less than the Receive Buffer size, then I3C shall wait until enough Receive FIFO locations are available to store the data to be transferred.
2. Threshold Mode
   If the RXSTTH[2:0] field value is less than the Receive Buffer size, then I3C shall initiate the Read Command as soon as the indicated number of Receive FIFO locations are empty.

26.2.41 Normal Receive Status Queue Threshold Control Register (NRQTHCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RSQTH[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7 to 0	RSQTH[7:0]	H'01	R/W	Normal Receive Status Queue Threshold H'00: Interrupt is issued when Receive Status Queue contains 1 entry (DWORD). Others: Interrupt is triggered when Receive Status Queue contains N+1 entries (DWORD). (N = RSQTH[7:0])

RSQTH[7:0] bits (Normal Receive Status Queue Threshold)

Controls the minimum number of receive status queue entries needed to trigger the INTRCV interrupt.

If this field is greater than (Receive Status Queue size – 1), then only the number of bits required to address the full buffer depth will be considered.

26.2.42 Bus Status Register (BST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	WUCNDDF	—	—	—	TODF	—	—	—	ALF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TENDF	—	—	—	NACKDF	—	HDREXDF	SPCNDDF	STCNDDF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
24	WUCNDDF	0b	R/W*3	Wake-Up Condition Detection Flag 0: Wake-Up Condition is not detected. 1: Wake-Up Condition is detected.
23 to 21	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
20	TODF	0b	R/W*3	Timeout Detection Flag 0: Timeout is not detected. 1: Timeout is detected.
19 to 17	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
16	ALF	0b	R/W*3	Arbitration Lost Flag*2 0: Arbitration is not lost 1: Arbitration is lost.
15 to 9	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
8	TENDF	0b	R/W*3	Transmit End Flag*2 0: Data is being transmitted. 1: Data has been transmitted.
7 to 5	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
4	NACKDF	0b	R/W*3	NACK Detection Flag*2 0: NACK is not detected. 1: NACK is detected.
3	—	0b	R	Reserved This bit is read as 0. The write value should be 0.
2	HDREXDF	0b	R/W*3	HDR Exit Pattern Detection Flag*1 0: HDR Exit Pattern Detection Interrupt does not occur. 1: HDR Exit Pattern Detection Interrupt occurs.
1	SPCNDDF	0b	R/W*3	STOP Condition Detection Flag 0: STOP condition is not detected. 1: STOP condition is detected.
0	STCNDDF	0b	R/W*3	START Condition Detection Flag 0: START condition is not detected. 1: START condition is detected.

Note 1. This bit supports for all I3C mode.

Note 2. This bit supports for I^2C mode.

Note 3. Clearing (to 0) condition: Writing 0 after 1 is read.

STCNDDF bit (START Condition Detection Flag)

[Setting conditions]

- All of the followings are satisfied:
  1. The BSTE.STCNDDE bit = 1.
  2. When a START condition (or a Repeated START condition) is detected.

[Clearing conditions]

- When 0 is written to the STCNDDF flag after reading STCNDDF flag = 1.
- When a STOP condition is detected.

SPCNDDF bit (STOP Condition Detection Flag)

[Setting conditions]

- All of the followings are satisfied:
  1. The BSTE.SPCNDDE bit = 1.
  2. When a STOP condition is detected.

[Clearing condition]

- When 0 is written to the SPCNDDF flag after reading SPCNDDF flag = 1.

HDREXDF bit (HDR Exit Pattern Detection Flag)

[Setting conditions]

- All of the followings are satisfied:
  1. The BSTE.HDREXDE bit = 1.
  2. When a HDR EXIT pattern is detected.

[Clearing condition]

- When 0 is written to the HDREXDF flag after reading HDREXDF flag = 1.

NACKDF bit (NACK Detection Flag)

[Setting conditions]

- All of the followings are satisfied:
  1. The PRTS.PRTMD bit = 1 (I^2C protocol mode).
  2. The BSTE.NACKDE bit = 1 (Enables NACK detection interrupt status logging).
  3. When acknowledge is not received (NACK is received) from the receive device in transmit mode.

[Clearing condition]

- When 0 is written to the NACKDF flag after reading NACKDF flag = 1.

TENDF bit (Transmit End Flag)

[Setting conditions]

- All of the followings are satisfied:

1. The PRTS.PRTMD bit = 1 (I^2C protocol mode).
2. The BSTE.TENDE bit = 1 (Enables Transmit End Interrupt Status logging).
3. At the rising edge of the ninth SCL clock cycle while the NTST.TDBEF0 flag = 1. Excluding when sending an address.

[Clearing conditions]

- When 0 is written to the TENDF flag after reading TENDF flag = 1.
- When data is written to the NTDTBP0 register.
- When a STOP condition is detected.

ALF bit (Arbitration Lost Flag)

[Setting conditions]

When master arbitration-lost detection is enabled: BSTE.ALE bit = 1, BFCTL.MALE = 1.

- When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock except for the ACK period during data (including slave address) transmission in master transmit mode (when the SDA line is driven low while the internal SDA output is at a high level (the SDA pin is in the high impedance state)).
- All of the followings are satisfied.
  1. When the START condition is detected while the CNDCTL.STCND bit = 1.
  2. When the internal SDA output state does not match the SDA line level.
- When the CNDCTL.STCND bit is set to 1 (START condition issuance request) while the BCST.BFREF flag = 0.

When NACK arbitration-lost detection is enabled: BSTE.ALE bit = 1, BFCTL.NALE = 1.

- When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock in the ACK period during NACK transmission in receive mode.

When slave arbitration-lost detection is enabled: BSTE.ALE bit = 1, BFCTL.SALE = 1.

- When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock except for the ACK period during data transmission in slave transmit mode.

[Clearing condition]

- When 0 is written to the ALF flag after reading ALF flag = 1.

TODF bit (Timeout Detection Flag)

[Setting conditions]

- All of the followings are satisfied.
  1. The BSTE.TODE bit = 1 (Enables Timeout Detection Interrupt Status logging).
  2. When the master mode or the received slave address matches the slave address n in Slave mode.
  3. When the SCL line state remains unchanged for the period specified by TMOCTL register.

[Clearing condition]

- When 0 is written to the TODF flag after reading TODF flag = 1.

26.2.43 Bus Status Enable Register (BSTE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	WUCNDDE	—	—	—	TODE	—	—	—	ALE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TENDE	—	—	—	NACKDE	—	HDREXDE	SPCNDDE	STCNDDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
24	WUCNDDE	0b	R/W	Wake-up Condition Detection Enable 0: Disables Wake-up Condition Detection Status logging 1: Enables Wake-up Condition Detection Status logging
23 to 21	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
20	TODE	0b	R/W	Timeout Detection Enable 0: Disables Timeout Detection Interrupt Status logging. 1: Enables Timeout Detection Interrupt Status logging.
19 to 17	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
16	ALE	0b	R/W	Arbitration Lost Enable*2 0: Disables Arbitration Lost Interrupt Status logging. 1: Enables Arbitration Lost Interrupt Status logging.
15 to 9	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
8	TENDE	0b	R/W	Transmit End Enable*2 0: Disables Transmit End Interrupt Status logging. 1: Enables Transmit End Interrupt Status logging.
7 to 5	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
4	NACKDE	0b	R/W	NACK Detection Enable*2 0: Disables NACK Detection Interrupt Status logging. 1: Enables NACK Detection Interrupt Status logging.
3	—	0b	R	Reserved This bit is read as 0. The write value should be 0.
2	HDREXDE	0b	R/W	HDR Exit Pattern Detection Enable*1 0: Disables HDR Exit Pattern Detection Interrupt Status logging. 1: Enables HDR Exit Pattern Detection Interrupt Status logging.
1	SPCNDDE	0b	R/W	STOP Condition Detection Enable 0: Disables STOP condition Detection Interrupt Status logging. 1: Enables STOP condition Detection Interrupt Status logging.
0	STCNDDE	0b	R/W	START Condition Detection Enable 0: Disables START condition Detection Interrupt Status logging. 1: Enables START condition Detection Interrupt Status logging.

Note 1. This bit supports for all I3C mode.

Note 2. This bit supports for I^2C mode.

STCNDDE bit (START Condition Detection Enable)

When this bit is 1, operation of BST.STCNDDF is enabled. For the setting conditions and clearing conditions of the BST.STCNDDF flag, see the details of BST.STCNDDF.

SPCNDDE bit (STOP Condition Detection Enable)

When this bit is 1, operation of BST.SPCNDDF is enabled. For the setting conditions and clearing conditions of the BST.SPCNDDF flag, see the details of BST.SPCNDDF.

HDREXDE bit (HDR Exit Pattern Detection Enable)

When this bit is 1, the operation of BST.HDREXDF is enabled. For the setting conditions and clearing conditions of the BST.HDREXDF flag, see the details of BST.HDREXDF.

NACKDE bit (NACK Detection Enable)

When this bit is 1, the operation of BST.NACKDF is enabled. This bit is used to specify whether to continue or discontinue the transfer operation when NACK is received from the slave device in transmit mode. Normally, set this bit to 1. For the setting conditions and clearing conditions of the BST.NACKDF flag, see the details of BST.NACKDF.

TENDE bit (Transmit End Enable)

When this bit is 1, the operation of BST.TENDF is enabled. For the setting conditions and clearing conditions of the BST.TENDF flag, see the details of BST.TENDF.

ALE bit (Arbitration Lost Enable)

When this bit is 1, the operation of BST.ALF is enabled. For the setting conditions and clearing conditions of the BST.ALF flag, see the details of BST.ALF.

TODE bit (Timeout Detection Enable)

When this bit is 1, the operation of BST.TODF is enabled. For the setting conditions and clearing conditions of the BST.TODF flag, see the details of BST.TODF.

26.2.44 Bus Interrupt Enable Register (BIE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	WUCNDDIE	—	—	—	TODIE	—	—	—	ALIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TENDIE	—	—	—	NACKDIE	—	HDREXDIE	SPCNDDIE	STCNDDIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
24	WUCNDDIE	0b	R/W	Wake-Up Condition Detection Interrupt Enable 0: Disables Wake-Up Condition Detection Interrupt Signal. 1: Enables Wake-Up Condition Detection Interrupt Signal.
23 to 21	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
20	TODIE	0b	R/W	Timeout Detection Interrupt Enable*2 0: Disables Timeout Detection Interrupt Signal. 1: Enables Timeout Detection Interrupt Signal.
19 to 17	—	All 0	R	Reserved This bit is read as 0. The write value should be 0.
16	ALIE	0b	R/W	Arbitration Lost Interrupt Enable*2 0: Disables Arbitration Lost Interrupt Signal. 1: Enables Arbitration Lost Interrupt Signal.
15 to 9	—	All 0	R	Reserved This bit is read as 0. The write value should be 0.
8	TENDIE	0b	R/W	Transmit End Interrupt Enable*2 0: Disables Transmit End Interrupt Signal. 1: Enables Transmit End Interrupt Signal.
7 to 5	—	000b	R	Reserved This bit is read as 0. The write value should be 0.
4	NACKDIE	0b	R/W	NACK Detection Interrupt Enable*2 0: Disables NACK Detection Interrupt Signal. 1: Enables NACK Detection Interrupt Signal.
3	—	0b	R	Reserved This bit is read as 0. The write value should be 0.
2	HDREXDIE	0b	R/W	HDR Exit Pattern Detection Interrupt Enable*1 0: Disables HDR Exit Pattern Detection Interrupt Signal. 1: Enables HDR Exit Pattern Detection Interrupt Signal.
1	SPCNDDIE	0b	R/W	STOP Condition Detection Interrupt Enable 0: Disables STOP condition Detection Interrupt Signal. 1: Enables STOP condition Detection Interrupt Signal.
0	STCNDDIE	0b	R/W	START Condition Detection Interrupt Enable 0: Disables START condition Detection Interrupt Signal. 1: Enables START condition Detection Interrupt Signal.

Note 1. This bit supports for all I3C mode.

Note 2. This bit supports for I^2C mode.

The BIE register enables signaling of outstanding bus interrupts received by I3C.

STCNDDIE bit (START Condition Detection Interrupt Enable)

This bit enables or disables the START Condition Detection interrupt requests when the BST.STCNDDF flag is set to 1.

SPCNDDIE bit (STOP Condition Detection Interrupt Enable)

This bit enables or disables the STOP Condition Detection interrupt requests when the BST.SPCNDDF flag is set to 1.

HDREXDIE bit (HDR Exit Pattern Detection Interrupt Enable)

This bit enables or disables the HDR Exit Pattern Detection interrupt requests when the BST.HDREXDF flag is set to 1.

NACKDIE bit (NACK Detection Interrupt Enable)

This bit enables or disables the NACK Detection interrupt requests when the BST.NACKDF flag is set to 1.

TENDIE bit (Transmit End Interrupt Enable)

This bit enables or disables the Transmit End interrupt (I3C_TEND) requests when the BST.TENDF flag is set to 1.

ALIE bit (Arbitration Lost Interrupt Enable)

This bit enables or disables the Arbitration Llost interrupt requests when the BST.ALF flag is set to 1.

TODIE bit (Timeout Detection Interrupt Enable)

This bit enables or disables the Timeout Detection interrupt requests when the BST.TODF flag is set to 1.

WUCNDDIE Bit

This bit is used to enable or disable the Wake-up condition detection interrupt (INTWU) requests when the BST.WUCNDDF flag is set to 1.

26.2.45 Bus Status Force Register (BSTFC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	WUCNDDFC	—	—	—	TODFC	—	—	—	ALFC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W	R	R	R	W	R	R	R	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TENDFC	—	—	—	NACKDFC	—	HDREXDFC	SPCNDDFC	STCNDDFC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W	R	R	R	W	R	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
24	WUCNDDFC	0b	W	Wake-Up Condition Detection Force 0: not Force Wake-Up Condition Detection Interrupt for software testing. 1: Force Wake-Up Condition Detection Interrupt for software testing.
23 to 21	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
20	TODFC	0b	W	Timeout Detection Force*2 0: Not Force Timeout Detection Interrupt for software testing. 1: Force Timeout Detection Interrupt for software testing.
19 to 17	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
16	ALFC	0b	W	Arbitration Lost Force*2 0: Not Force Arbitration Lost Interrupt for software testing. 1: Force Arbitration Lost Interrupt for software testing.
15 to 9	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
8	TENDFC*3	0b	W	Transmit End Force*2 0: Not Force Transmit End Interrupt for software testing. 1: Force Transmit End Interrupt for software testing.
7 to 5	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
4	NACKDFC	0b	W	NACK Detection Force*2 0: Not Force NACK Detection Interrupt for software testing. 1: Force NACK Detection Interrupt for software testing.
3	—	0b	R	Reserved These bits are read as 0. The write value should be 0.
2	HDREXDFC	0b	W	HDR Exit Pattern Detection Force*1 0: Not Force HDR Exit Pattern Detection Interrupt for software testing. 1: Force HDR Exit Pattern Detection Interrupt for software testing.
1	SPCNDDFC	0b	W	STOP condition Detection Force 0: Not Force STOP condition Detection Interrupt for software testing. 1: Force STOP condition Detection Interrupt for software testing.
0	STCNDDFC	0b	W	START condition Detection Force 0: Not Force START condition Detection Interrupt for software testing. 1: Force START condition Detection Interrupt for software testing.

Note 1. This bit supports for all I3C mode.

Note 2. This bit supports for I^2C mode.

Note 3. TENDFC does not work unless TDBEF0 = 1.

26.2.46 Normal Transfer Status Register (NTST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	RSQFF	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TEF	—	—	—	TABTF	RSPQFF	CMDQEF	IBIQEFF	RDBFF0	TDBEF0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
20	RSQFF	0b	R/W*3	Normal Receive Status Queue Full Flag*2 0: The number of Receive Status Queue entries is the NRQTHCTL.RSQTH threshold or less. 1: The number of Receive Status Queue entries is more than the NRQTHCTL.RSQTH threshold.
19 to 10	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
9	TEF	0b	R/W*3	Normal Transfer Error Flag*1 0: Transfer Error does not occur. 1: Transfer Error occurs. To clear, write 0 to this bit after 1 state is read.
8 to 6	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
5	TABTF	0b	R/W*3	Normal Transfer Abort Flag*1 0: Transfer Abort does not occur. 1: Transfer Abort occur. To clear, write 0 to this bit after 1 state is read.
4	RSPQFF	0b	R/W*3	Normal Response Queue Full Flag*1 0: The number of Response Queue entries is the NQTHCTL.RSPQTH threshold or less. 1: The number of Response Queue entries is more than the NQTHCTL.RSPQTH threshold.
3	CMDQEF	0b	R/W*3	Normal Command Queue Empty Flag*1 0: If the NQTHCTL.CMDQTH = 0: The number of Command Queue empties is less than the Command Queue size. If the NQTHCTL.CMDQTH is other than 0: The number of Command Queue empties is less than the NQTHCTL.CMDQTH threshold. 1: If the NQTHCTL.CMDQTH = 0: The number of Command Queue empties is the Command Queue size. If the NQTHCTL.CMDQTH is other than 0: 1: The number of Command Queue empties is the NQTHCTL.CMDQTH threshold or more.

Bit	Bit Name	Initial Value	R/W	Description
2	IBIQEFF	0b	R/W*3	Normal IBI Queue Empty/Full Flag*1 0: For I3C protocol mode (Master): PRTS.PRTMD bit = 0, PRSST.CRMS bit = 1. The number of IBI Status Queue entries is the NQTHCTL.IBIQTH threshold or less. For I3C protocol mode (Slave) : PRTS.PRTMD bit = 0, PRSST.CRMS bit = 0. If the NQTHCTL.IBIQTH = 0: The number of IBI Data Buffer empties is less than the IBI Data Buffer size. If the NQTHCTL.IBIQTH is other than 0: The number of IBI Data Buffer empties is less than the NQTHCTL.IBIQTH threshold. 1: For I3C protocol mode (Master): PRTS.PRTMD bit = 0, PRSST.CRMS bit = 1. The number of IBI Status Queue entries is more than the NQTHCTL.IBIQTH threshold. For I3C protocol mode (Slave) : PRTS.PRTMD bit = 0, PRSST.CRMS bit = 0. If the NQTHCTL.IBIQTH = 0: The number of IBI Data Buffer empties is the IBI Data Buffer size. If the NQTHCTL.IBIQTH is other than 0: The number of IBI Data Buffer empties is the NQTHCTL.IBIQTH threshold or more.
1	RDBFF0	0b	R/W*3	Normal Receive Data Buffer Full Flag 0*1 0: For I2C protocol mode: PRTS.PRTMD bit = 1. Normal Receive Data Buffer0 contains no receive data. For I3C Protocol mode: PRTS.PRTMD bit = 0. The number of entries in the Normal Receive Data Buffer 0 is less than the NTBTHCTL0.RXDBTH[2:0] threshold. 1: For I2C protocol mode: PRTS.PRTMD bit = 1. Normal Receive Data Buffer0 contains receive data. For I3C Protocol mode: PRTS.PRTMD bit = 0. The number of entries in the Normal Receive Data Buffer 0 is the NTBTHCTL0.RXDBTH[2:0] threshold or more.
0	TDBEF0	0b	R/W*3	Normal Transmit Data Buffer Empty Flag 0*1 0: For I2C protocol mode: PRTS.PRTMD bit = 1. Normal Transmit Data Buffer 0 contains transmit data. For I3C protocol mode: PRTS.PRTMD bit = 0. The number of empties in the Normal Transmit Data Buffer 0 is less than the NTBTHCTL0.TXDBTH[2:0] threshold. 1: For I2C protocol mode: PRTS.PRTMD bit = 1. Normal Transmit Data Buffer 0 contains no transmit data. For I3C protocol mode: PRTS.PRTMD bit = 0. The number of empties in the Normal Transmit Data Buffer 0 is the NTBTHCTL0.TXDBTH[2:0] threshold or more.

Note 1. This bit supports for all I3C mode.

Note 2. This bit supports for I3C secondary master mode and I3C slave mode.

Note 3. Clearing (to 0) condition: Writing 0 after the 1 state is read.

TDBEF0 bit (Normal Transmit Data Buffer Empty Flag 0)

[Setting conditions]

For I2C Protocol mode: PRTS.PRTMD bit = 1.

The following condition 1 is satisfied and any of the following conditions 2 to 4 are satisfied:

1. The NTSTE.TDBEE0 bit = 1 (enables Tx0 Data Buffer Empty Interrupt Status logging).
2. When data has been transferred from the Normal Transmit Data Buffer 0 to the Shift Register and the Normal Transmit Data Buffer 0 becomes empty*1.
3. When the PRSST.TRMD bit is set to 1.
4. When the received slave address matches while the TRMD bit = 1.

For I3C Protocol mode: PRTS.PRTMD bit = 0.

The following conditions 1 and 2 are satisfied:

1. The NTSTE.TDBEE0 bit = 1 (enables Tx0 Data Buffer Empty Interrupt Status logging).
2. When the number of empties in the Normal Transmit Data Buffer 0 is the NTBTHCTL0.TXDBTH[2:0] threshold or more (refer to NTBTHCTL0 register).

[Clearing conditions]

For I^2C Protocol mode: PRTS.PRTMD bit = 1.

- When data is written to NTDTBP0.
- When the TRMD bit in PRSST is set to 0.

For I3C protocol mode: PRTS.PRTMD bit = 0.

- Write 0 to this bit after 1 is read.
- On completion of the last write access to Normal Transmit Data by DTC.
- When the number of empties in the Normal Transmit Data Buffer 0 is less than the NTBTHCTL0.TXDBTH[2:0] threshold (refer to NTBTHCTL0 register).

Note 1. When the BST.NACKDF flag is set to 1 while the BSTE.NACKDE bit = 1, I3C aborts data transmission/reception. If the TDBEF0 flag = 0 (next transmit data has been written), data is transferred to the Shift Register and the Normal Transmit Data Buffer 0 register becomes empty at the rising edge of the 9th clock cycle, but the TDBEF0 flag is not set to 1.

RDBFF0 bit (Normal Receive Data Buffer Full Flag 0)

[Setting conditions]

For I^2C Protocol mode: PRTS.PRTMD bit = 1.

The following condition 1 is satisfied and any of the following condition 2 or 3 is satisfied:

1. The NTSTE.RDBFE0 bit = 1 (enables Rx0 Data Buffer Full Interrupt Status logging).
2. When receive data is transferred from Shift Register to Normal Receive Data Buffer 0.
   The RDBFF0 flag is set to 1 on the rising edge of the 8th or 9th SCL clock cycle (selected in the ACKTWE bit in SCSTRCTL).
3. When the received slave address matches after a START (or Repeated START) condition is detected with the TRMD bit in PRSST set to 0.

For I3C Protocol mode: PRTS.PRTMD bit = 0. The following conditions 1 and 2 are satisfied:

1. The NTSTE.RDBFE0 bit = 1 (enables Rx0 Data Buffer Full Interrupt Status logging).
2. When the number of Normal Receive Data Buffer 0 entries is the NTBTHCTL0.RXDBTH[2:0] threshold or more (refer to NTBTHCTL0 register).

[Clearing conditions]

For I^2C Protocol mode: PRTS.PRTMD bit = 1.

- When data is read from NTDTBP0.

For I3C Protocol mode: PRTS.PRTMD bit = 0.

- Write 0 to this bit after 1 is read.
- On completion of the last read access to Normal Receive Data by DTC.

- When the number of Normal Receive Data Buffer 0 entries is less than the NTBTHCTL0.RXDBTH[2:0] threshold (refer to NTBTHCTL0 register).

IBIQEFF bit (Normal IBI Queue Empty/Full Flag)

[Setting conditions]

The following 2 conditions are satisfied:

1. The NTSTE.IBIQEFE bit = 1 (enables IBI Status Buffer Empty/Full Interrupt Status logging)
2. For I3C protocol mode (master): PRTS.PRTMD bit = 0, PRSST.CRMS bit = 1.
   - When the number of IBI Status Queue entries is more than the NQTHCTL.IBIQTH threshold (refer to NQTHCTL register).

   For I3C protocol mode (slave): PRTS.PRTMD bit = 0, PRSST.CRMS bit = 0.
   If the NQTHCTL.IBIQTH = 0:
   - When IBI Data Buffer is completely empty.

   If the NQTHCTL.IBIQTH is other than 0:
   - When the number of IBI Data Buffer empties is the NQTHCTL.IBIQTH threshold or more (refer to NQTHCTL register).

[Clearing conditions]

For I3C protocol mode (master): PRTS.PRTMD bit = 0, PRSST.CRMS bit = 1.

- Write 0 to this bit after 1 is read.
- On completion of the last read access to IBI Status by DTC.
- When the number of IBI Status Queue entries is the NQTHCTL.IBIQTH threshold or less (refer to NQTHCTL register).

For I3C protocol mode (slave): PRTS.PRTMD bit = 0, PRSST.CRMS bit = 0.

- Write 0 to this bit after 1 is read.
- On completion of the last write access to IBI Status by DTC.

If the NQTHCTL.IBIQTH = 0:

- When IBI Data Buffer is not completely empty.

If the NQTHCTL.IBIQTH is other than 0:

- When the number of IBI Data Buffer empties is less than the NQTHCTL.IBIQTH threshold (refer to NQTHCTL register).

CMDQEF bit (Normal Command Queue Empty Flag)

[Setting conditions]

The following 2 conditions are satisfied:

1. The NTSTE.CMDQEE bit = 1 (enables Command Buffer Empty Interrupt Status logging).
2. If the NQTHCTL.CMDQTH = 0:
   - When Command Queue is completely empty.

   If the NQTHCTL.CMDQTH is other than 0:

- When the number of Command Queue empties is the NQTHCTL.CMDQTH threshold or more (refer to NQTHCTL register).

[Clearing conditions]

- Write 0 to this bit after 1 is read.
- On completion of the last write access to Normal Command by DTC.

If the NQTHCTL.CMDQTH = 0:

- When Command Queue is not completely empty.

If the NQTHCTL.CMDQTH is other than 0:

- When the number of Command Queue empties is less than the NQTHCTL.CMDQTH threshold (refer to NQTHCTL register).

RSPQFF bit (Normal Response Queue Full Flag)

[Setting conditions]

The following 2 conditions are satisfied:

1. The NTSTE.RSPQFE bit = 1 (enables Response Buffer Full Interrupt Status logging).
2. When the number of Response Queue entries is more than the NQTHCTL.RSPQTH threshold (refer to NQTHCTL register).

[Clearing conditions]

- Write 0 to this bit after 1 is read.
- On completion of the last read access to Normal Receive Status by DTC.
- When the number of Response Queue entries is the NQTHCTL.RSPQTH threshold or less (refer to NQTHCTL register).

TABTF bit (Normal Transfer Abort Flag)

[Setting conditions]

The following 2 conditions are satisfied:

1. The NTSTE.TABTE bit = 1 (enables Ttransfer Abort Interrupt Status logging).
2. When any transfer is aborted.

[Clearing condition]

- Write 0 to this bit after 1 is read.

TEF bit (Normal Transfer Error Flag)

[Setting conditions]

The following 2 conditions are satisfied:

1. The NTSTE.TEE bit = 1 (enables Ttransfer Error Interrupt Status logging).
2. When any transfer error occurs on the I3C bus. The Error type for this error is available in the Response or Receive Status structure corresponding to the Transfer command.

[Clearing condition]

- Write 0 to this bit after 1 is read.

RSQFF bit (Normal Receive Status Queue Full Flag)

[Setting conditions]

The following 2 conditions are satisfied:

1. The NTSTE.RSQFE bit = 1 (Normal Receive Status Queue Full Enable).
2. When the number of Receive Status Queue entries is more than the NRQTHCTL.RSQTH threshold (refer to NRQTHCTL register).

[Clearing conditions]

- Write 0 to this bit after 1 is read.
- On completion of the last read accsess to Normal Receive Status by DTC.
- When the number of Receive Status Queue entries is the NRQTHCTL.RSQTH threshold or less (refer to NRQTHCTL register).

26.2.47 Normal Transfer Status Enable Register (NTSTE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	RSQFE	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TEE	—	—	—	TABTE	RSPQFE	CMDQEE	IBIQEFE	RDBFE0	TDBEE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
20	RSQFE	0b	R/W	Normal Receive Status Queue Full Enable*2 0: Disables Receive Status Buffer Full Interrupt Status logging. 1: Enables Receive Status Buffer Full Interrupt Status logging.
19 to 10	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
9	TEE	0b	R/W	Normal Transfer Error Enable*1 0: Disables Transfer Error Interrupt Status logging. 1: Enables Transfer Error Interrupt Status logging.
8 to 6	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
5	TABTE	0b	R/W	Normal Transfer Abort Enable*1 0: Disables Transfer Abort Interrupt Status logging. 1: Enables Transfer Abort Interrupt Status logging.
4	RSPQFE	0b	R/W	Normal Response Queue Full Enable*1 0: Disables Response Buffer Full Interrupt Status logging. 1: Enables Response Buffer Full Interrupt Status logging.
3	CMDQEE	0b	R/W	Normal Command Queue Empty Enable*1 0: Disables Command Buffer Empty Interrupt Status logging. 1: Enables Command Buffer Empty Interrupt Status logging.
2	IBIQEFE	0b	R/W	Normal IBI Queue Empty/Full Enable*1 0: Disables IBI Status Buffer Empty/Full Interrupt Status logging. 1: Enables IBI Status Buffer Empty/Full Interrupt Status logging.
1	RDBFE0	0b	R/W	Normal Receive Data Buffer Full Enable 0 0: Disables Rx0 Data Buffer Full Interrupt Status logging. 1: Enables Rx0 Data Buffer Full Interrupt Status logging.
0	TDBEE0	0b	R/W	Normal Transmit Data Buffer Empty Enable 0 0: Disables Tx0 Data Buffer Empty Interrupt Status logging. 1: Enables Tx0 Data Buffer Empty Interrupt Status logging.

Note 1. This bit supports for all I3C mode.

Note 2. This bit supports for I3C secondary master mode and I3C slave mode.

TDBEE0 bit (Normal Transmit Data Buffer Empty Enable 0)

When this bit is 1, the operation of NTST.TDBEF0 is enabled.

For the setting conditions and clearing conditions of the NTST.TDBEF0 flag, see the details of NTST.TDBEF0.

RDBFE0 bit (Normal Receive Data Buffer Full Enable 0)

When this bit is 1, the operation of NTST.RDBFF0 is enabled.

For the setting conditions and clearing conditions of the NTST.RDBFF0 flag, see the details of NTST.RDBFF0.

IBIQEFE bit (Normal IBI Queue Empty/Full Enable)

When this bit is 1, the operation of NTST.IBIQEFF is enabled.

For the setting conditions and clearing conditions of the NTST.IBIQEFF flag, see the details of NTST.IBIQEFF.

CMDQEE bit (Normal Command Queue Empty Enable)

When this bit is 1, the operation of NTST.CMDQEF is enabled.

For the setting conditions and clearing conditions of the NTST.CMDQEF flag, see the details of NTST.CMDQEF.

RSPQFE bit (Normal Response Queue Full Enable)

When this bit is 1, the operation of NTST.RSPQFF is enabled.

For the setting conditions and clearing conditions of the NTST.RSPQFF flag, see the details of NTST.RSPQFF.

TABTE bit (Normal Transfer Abort Enable)

When this bit is 1, the operation of NTST.TABTF is enabled.

For the setting conditions and clearing conditions of the NTST.TABTF flag, see the details of NTST.TABTF.

TEE bit (Normal Transfer Error Enable)

When this bit is 1, the operation of NTST.TEF is enabled.

For the setting conditions and clearing conditions of the NTST.TEF flag, see the details of NTST.TEF.

RSQFE bit (Normal Receive Status Queue Full Enable)

When this bit is 1, the operation of NTST.RSQFF is enabled.

For the setting conditions and clearing conditions of the NTST.RSQFF flag, see the details of NTST.RSQFF.

26.2.48 Normal Transfer Interrupt Enable Register (NTIE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	RSQFIE	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TEIE	—	—	—	TABTIE	RSPQFIE	CMDQEIE	IBIQEFIE	RDBFIE0	TDBEIE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
20	RSQFIE	0b	R/W	Normal Receive Status Queue Full Interrupt Enable*2 0: Disables Receive Status Buffer Full Interrupt Signal. 1: Enables Receive Status Buffer Full Interrupt Signal.
19 to 10	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
9	TEIE	0b	R/W	Normal Transfer Error Interrupt Enable*1 0: Disables Transfer Error Interrupt Signal. 1: Enables Transfer Error Interrupt Signal.
8 to 6	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
5	TABTIE	0b	R/W	Normal Transfer Abort Interrupt Enable*1 0: Disables Transfer Abort Interrupt Signal. 1: Enables Transfer Abort Interrupt Signal.
4	RSPQFIE	0b	R/W	Normal Response Queue Full Interrupt Enable*1 0: Disables Response Buffer Full Interrupt Signal. 1: Enables Response Buffer Full Interrupt Signal.
3	CMDQEIE	0b	R/W	Normal Command Queue Empty Interrupt Enable*1 0: Disables Command Buffer Empty Interrupt Signal. 1: Enables Command Buffer Empty Interrupt Signal.
2	IBIQEFIE	0b	R/W	Normal IBI Queue Empty/Full Interrupt Enable*1 0: Disables IBI Status Buffer Empty/Full Interrupt Signal. 1: Enables IBI Status Buffer Empty/Full Interrupt Signal.
1	RDBFIE0	0b	R/W	Normal Receive Data Buffer Full Interrupt Enable 0 0: Disables Rx0 Data Buffer Full Interrupt Signal. 1: Enables Rx0 Data Buffer Full Interrupt Signal.
0	TDBEIE0	0b	R/W	Normal Transmit Data Buffer Empty Interrupt Enable 0 0: Disables Tx0 Data Buffer Empty Interrupt Signal. 1: Enables Tx0 Data Buffer Empty Interrupt Signal.

Note 1. This bit supports for all I3C mode.

Note 2. This bit supports for I3C secondary master mode and I3C slave mode.

The PIO Interrupt Signal Enable register enables signaling of outstanding interrupts received by I3C.

TDBEIE0 bit (Normal Transmit Data Buffer Empty Interrupt Enable 0)

This bit is used to enable or disable the Normal Tx Data buffer 0 empty interrupt (I3Cn_TX) requests when the NTST.TDBEF0 flag is set to 1.

RDBFIE0 bit (Normal Receive Data Buffer Full Interrupt Enable 0)

This bit is used to enable or disable the Normal Rx Data buffer 0 full interrupt (I3Cn_RX) requests when the NTST.RDBFF0 flag is set to 1.

IBIQEFIE bit (Normal IBI Queue Empty/Full Interrupt Enable)

This bit is used to enable or disable the Normal IBI Status buffer full interrupt (INTIBI) requests when the NTST.IBIQEFF flag is set to 1.

CMDQEIE bit (Normal Command Queue Empty Interrupt Enable)

This bit is used to enable or disable the Normal Command buffer empty interrupt (INTCMD) requests when the NTST.CMDQEF flag is set to 1.

RSPQFIE bit (Normal Response Queue Full Interrupt Enable)

This bit is used to enable or disable the Normal Response Status buffer full interrupt (INTRESP) requests when the NTST.RSPQFF flag is set to 1.

TABTIE bit (Normal Transfer Abort Interrupt Enable)

This bit is used to enable or disable the Normal Transfer Abort interrupt (INTABORT) requests when the NTST.TABTF flag is set to 1.

TEIE bit (Normal Transfer Error Interrupt Enable)

This bit is used to enable or disable the Normal Transfer Error interrupt (INTTERR) requests when the NTST.TEF flag is set to 1.

RSQFIE bit (Normal Receive Status Queue Full Interrupt Enable)

This bit is used to enable or disable the Normal Receive Status buffer full interrupt (INTRCV) requests when the NTST.RSQFF flag is set to 1.

26.2.49 Normal Transfer Status Force Register (NTSTFC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	RSQFFC	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TEFC	—	—	—	TABTFC	RSPQFFC	CMDQEFC	IBIQEFFC	RDBFFC0	TDBEFC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	W	R	R	R	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved The write value should be 0.
20	RSQFFC	0b	W	Normal Receive Status Queue Full Force*2 0: Not Force Receive Status Buffer Full Interrupt for software testing. 1: Force Receive Status Buffer Full Interrupt for software testing.
19 to 10	—	All 0	R	Reserved The write value should be 0.
9	TEFC	0b	W	Normal Transfer Error Force*1 0: Not Force Transfer Error Interrupt for software testing. 1: Force Transfer Error Interrupt for software testing.
8 to 6	—	000b	R	Reserved The write value should be 0.
5	TABTFC	0b	W	Normal Transfer Abort Force*1 0: Not Force Transfer Abort Interrupt for software testing. 1: Force Transfer Abort Interrupt for software testing.
4	RSPQFFC	0b	W	Normal Response Queue Full Force*1 0: Not Force Response Buffer Full Interrupt for software testing. 1: Force Response Buffer Full Interrupt for software testing.
3	CMDQEFC	0b	W	Normal Command Queue Empty Force*1 0: Not Force Command Buffer Empty Interrupt for software testing. 1: Force Command Buffer Empty Interrupt for software testing.
2	IBIQEFFC	0b	W	Normal IBI Queue Empty/Full Force*1 0: Not Force IBI Status Buffer Full Interrupt for software testing. 1: Force IBI Status Buffer Full Interrupt for software testing.
1	RDBFFC0	0b	W	Normal Receive Data Buffer Full Force 0 0: Not Force Rx0 Data Buffer Full Interrupt for software testing. 1: Force Rx0 Data Buffer Full Interrupt for software testing.
0	TDBEFC0	0b	W	Normal Transmit Data Buffer Empty Force 0 0: Not Force Tx0 Data Buffer Empty Interrupt for software testing. 1: Force Tx0 Data Buffer Empty Interrupt for software testing.

Note 1. This bit supports for all I3C mode.

Note 2. This bit supports for I3C secondary master mode and I3C slave mode.

The PIO Interrupt Force register is used to force specific interrupt. It can be used for debug purposes.

TDBEFC0 bit (Normal Transmit Data Buffer Empty Force 0)

For software testing, when set to 1, forces the corresponding interrupt, subject to TDBEE0 and TDBEIE0 configuration.

RDBFFC0 bit (Normal Receive Data Buffer Full Force 0)

For software testing, when set to 1, forces the corresponding interrupt, subject to RDBFE0 and RDBFIE0 configuration.

IBIQEFFC bit (Normal IBI Queue Empty/Full Force)

For software testing, when set to 1, forces the corresponding interrupt, subject to IBIQEFE and IBIQEFIE configuration.

CMDQEFC bit (Normal Command Queue Empty Force)

For software testing, when set to 1, forces the corresponding interrupt, subject to CMDQEE and CMDQEIE configuration.

RSPQFFC bit (Normal Response Queue Full Force)

For software testing, when set to 1, forces the corresponding interrupt, subject to RSPQFEand RSPQFIE configuration.

TABTFC bit (Normal Transfer Abort Force)

For software testing, forces the corresponding interrupt, subject to TABTE and TABTIE configuration.

TEFC bit (Normal Transfer Error Force)

For software testing, when set to 1, forces the corresponding interrupt, subject to TEE and TEIE configuration.

RSQFFC bit (Normal Receive Status Queue Full Force)

For software testing, when set to 1, forces the corresponding interrupt, subject to RSQFE and RSQFIE configuration.

26.2.50 Bus Condition Status Register (BCST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	BIDLF	BAVLF	BFREF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are read as 0.
2	BIDLF	0b	R	Bus Idle Detection Flag*1 0: Have not Detected Bus Idle 1: Have Detected Bus Idle
1	BAVLF	0b	R	Bus Available Detection Flag*1 0: Have not Detected Bus Available 1: Have Detected Bus Available
0	BFREF	0b	R	Bus Free Detection Flag 0: Have not Detected Bus Free 1: Have Detected Bus Free

Note 1. This bit supports for all I3C mode.

BFREF bit (Bus Free Detection Flag)

The Bus Free Condition is a period occurring after a STOP and before a START, and with the following duration:

- For Pure Bus: A duration of at least tCAS
- For Mixed Bus (at least one Legacy I^2C is present on the I3C Bus): A duration of at least tBUF

[Setting conditions]

- After a STOP condition is detected, when the number of cycles (I3Cϕ) that are set by BFRECDT.FRECYC[8:0] has passed in the state of SCL = SDA = 1.
- After setting BCTL.BUSE to 1, when the number of cycles (I3Cϕ) that are set by BFRECDT.FRECYC[8:0] has passed in the state of SCL = SDA = 1.

[Clearing conditions]

- When SCL and SDA are other than high.
- When the BCTL.BUSE bit is set to 0.

BAVLF bit (Bus Available Detection Flag)

The Bus Available Condition is a period during which the Bus Free Condition is sustained continuously for a duration of at least tAVAL. A Slave can only issue a START Request (for an In-Band Interrupt, or for a Master Handoff Request) after a Bus Available Condition.

[Setting conditions]

- After a STOP condition is detected, when the number of cycles (I3Cϕ) that are set by BAVLCDT.AVLCYC[8:0] has passed in the state of SCL = SDA = 1.
- After setting BCTL.BUSE to 1, when the number of cycles (I3Cϕ) that are set by BAVLCDT.AVLCYC[8:0] has passed in the state of SCL = SDA = 1.

[Clearing conditions]

- When SCL and SDA are other than high.
- When the BCTL.BUSE bit is set to 0.

BIDLF bit (Bus Idle Detection Flag)

The I3C Bus Idle Condition is in order to help ensure Bus stability during Hot-Join events. The Bus Idle Condition is a period during which the Bus Available Condition is sustained continuously for a duration of at least tIDLE.

If a Hot-Join Device is powered up onto the I3C Bus at the same time as the Main Master, then the Hot-Join Device may pull SDA Low after 1 ms if (1) the Main Master has SCL and SDA pulled up, and (2) the Master does not act on the I3C Bus within the same Idle period.

[Setting conditions]

- After a STOP condition is detected, when the number of cycles (I3Cϕ) that are set by BIDLCDT.IDLCYC[17:0] has passed in the state of SCL = SDA = 1.
- After setting BCTL.BUSE to 1, when the number of cycles (I3Cϕ) that are set by BIDLCDT.IDLCYC[17:0] has passed in the state of SCL = SDA = 1.

[Clearing conditions]

- When SCL and SDA are other than high.
- When the BCTL.BUSE bit is set to 0.

26.2.51 Slave Status Register (SVST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	SVAF[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HOAF	—	—	—	—	—	—	—	—	DVIDF	HSMCF	—	—	—	—	GCAF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
18 to 16	SVAF[2:0]	000b	R/W*1	Slave Address Detection Flag [2:0] 0: Slave [2:0] does not detect 1: Slave [2:0] detect
15	HOAF	0b	R/W*1	Host Address Detection Flag 0: Host address does not detect. 1: Host address detects. • This bit set to 1 when the received slave address matches the host address (000_1000).
14 to 7	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
6	DVIDF	0b	R/W*1	Device-ID Address Detection Flag 0: Device-ID command does not detect. 1: Device-ID command detects. • This bit set to 1 when the first frame received immediately after a START condition is detected matches a value of (device ID (111_1100) + 0[W]).
5	HSMCF	0b	R/W*1	Hs-mode Master Code Detection Flag 0: Hs-mode Master Code does not detect. 1: Hs-mode Master Code detects.
4 to 1	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
0	GCAF	0b	R/W*1	General Call Address Detection Flag 0: General call address does not detect. 1: General call address detects.

Note: This register supports for I^2C mode.

Note 1. Clearing (to 0) condition: Writing 0 after the 1 state is read.

GCAF flag (General Call Address Detection Flag)

[Setting conditions]

- This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the first byte when all of the followings are satisfied.

  1. The SVCTL.GCAE bit = 1 (General call address detection is enabled).

  2. When the received slave address matches the general call address (000_0000 + 0 (write)).

[Clearing conditions]

- When 0 is written to the GCAF flag after reading GCAF flag to be 1.
- When a STOP condition is detected.
- When a Repeated START condition is detected.

HSMCF flag (Hs-mode Master Code Detection Flag)

[Setting conditions]

- This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the first byte when all of the followings are satisfied.

  1. The SVCTL.HSMCE bit = 1 (Hs-mode master code detection is enabled).

  2. When the first byte received immediately after a START condition is detected matches a value of Hs-mode master code (0000_1XXX) + 1 (NACK).

[Clearing conditions]

- When 0 is written to the HSMCF flag after reading HSMCF flag to be 1.
- When a STOP condition is detected.

DVIDF flag (Device-ID Address Detection Flag)

[Setting conditions]

- This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the first byte when all of the followings are satisfied.

  1. The SVCTL.DVIDE bit = 1 (Device-ID address detection is enabled).

  2. When the first byte received immediately after a START condition or Repeated START condition is detected matches a value of Device ID (111_1100) + 0 (write).

[Clearing conditions]

- When 0 is written to the DVIDF flag after reading DVIDF flag to be 1.
- When a STOP condition is detected.
- This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the first byte when the following 1 and 2 or 1 and 3 are satisfied.

  1. The SVCTL.DVIDE bit = 1 (Device-ID address detection is enabled).

  2. When the first byte received immediately after a START condition or Repeated START condition is detected does not match a value of Device ID (111_1100).

  3. When the first byte received immediately after a START condition or Repeated START condition is detected matches a value of (device ID (111_1100) + 0 [W]) and the second byte does not match any of slave addresses 0 to 2.

HOAF flag (Host Address Detection Flag)

[Setting conditions]

- This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the first byte when all of the followings are satisfied.

  1. The SVCTL.HOAE bit = 1 (Host address detection is enabled).

  2. When the received slave address matches the host address (000_1000).

[Clearing conditions]

- When 0 is written to the HOAF flag after reading HOAF flag to be 1.
- When a STOP condition is detected.
- When a Repeated START condition is detected.

SVAFy flags (Slave Address Detection Flag y) (y = 0 to 2)

[Setting conditions]

For 7-bit address format: SVDVADy.SADLG bit = 0.

- This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the first byte when all of the followings are satisfied.
  1. The SVCTL.SVAEy bit = 1 (Slave n enabled).
  2. When the received slave address matches the SVDVADy.SVAD[6:0] bits value.

For 10-bit address format: SVDVADy.SADLG bit = 1.

- This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the second byte when all of the followings are satisfied.
  1. The SVCTL.SVAEy bit = 1 (Slave n enabled).
  2. When the received slave address matches a value of 1_1110 + SVDVADy.SVAD[9:8] bits and the following address matches the SVDVADy.SVAD[7:0] value.

[Clearing conditions]

- When 0 is written to the SVAFy flag after reading SVAFy flag to be 1.
- When a STOP condition is detected.

For 7-bit address format: SVDVADy.SADLG bit = 0.

- This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the first byte when all of the followings are satisfied.
  1. The SVCTL.SVAEy bit = 1 (Slave n enabled).
  2. When the received slave address does not match SVDVADy.SVAD[6:0] bits value.

For 10-bit address format: SVDVADy.SADLG bit = 1.

- This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the first byte when all of the followings are satisfied.
  1. The SVCTL.SVAEy bit = 1 (Slave n enabled).
  2. When the received slave address does not match a value of 1_1110 + SVDVADy.SVAD[9:8] bits.
- This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the second byte when all of the followings are satisfied.
  1. The SVCTL.SVAEn bit = 1 (Slave n enabled).
  2. When the received slave address matches a value of 1_1110 + SVDVADy.SVAD[9:8] bits and the following address does not match the SVDVADy.SVAD[7:0] value.

26.2.52 Wake Up Unit Operating Status Register (WUST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WUASYNF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
0	WUASYNF	0b	R	Wake-Up function Asynchronous operation status Flag 0: This IP synchronous circuit enable condition 1: This IP asynchronous circuit enable condition

WUASYNF Flag (Wake-Up function asynchronous operation status flag)

This bit shows whether this IP is in the TCLK asynchronous operation (WUCTL.WUFE bit = 1).

[Setting condition]

The following 1 and 2 are satisfied.

1. The WUCTL.WUFE bit = 1 (Wake-up function is enabled)
2. When the BCST.BFREF flag = 1 after 0 is written to the WUCTL.WUFSYNE bit

[Clearing condition: I^2C slave]

The WUCTL.WUFE bit = 0 (Wake-up function is disabled)

All of the following 1 to 3 are satisfied.

1. The WUCTL.WUFE bit = 1 (Wake-up function is enabled)
2. Wake-Up event is detected
3. When 1 is written to the WUCTL.WUFSYNE bit during WUASYNF flag = 1

[Clearing condition: I3C slave]

The WUCTL.WUFE bit = 0 (Wake-up function is disabled)

All of the following 1 to 4 are satisfied.

1. The WUCTL.WUFE bit = 1 (Wake-up function is enabled)
2. Wake-Up event is detected
3. When 1 is written to the WUCTL.WUFSYNE bit during WUASYNF flag = 1
4. When a STOP condition is detected.

[Clearing condition: I^2C/I3C slave]

All of the following 1 to 5 are satisfied.

1. The WUCTL.WUFE bit = 1 (Wake-up function is enabled)
2. Wake-Up event is not detected
3. The WUASYNF flag = 1
4. The WUCTL.WUFSYNE bit = 1
5. When a STOP condition is detected.

[Clearing condition: I3C master]

The WUCTL.WUFE bit = 0 (Wake-up function is disabled)

All of the following 1 to 4 are satisfied.

1. The WUCTL.WUFE bit = 1 (Wake-up function is enabled)
2. Wake-Up event is detected
3. The WUASYNF flag = 1
4. The WUCTL.WUFSYNE bit = 1

26.2.53 MsyncCNT Counter Capture Register (MRCCPT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	n/a	All 0	R/W	MSyncCNT Counter Capture Used in Async Mode 1, not used in Async Mode0.

Note: This register supports for I3C secondary master mode and I3C slave mode.

MRCCPT[31:0] Bits

Async Mode 1 (Asynchronous Advanced Mode)

When ATCCNTE.ATCE is enabled, it starts counting. It captures as MSyncCNT for each aME (SDA falling edge of START condition) and store it in the capture register.

26.2.54 Device Address Table Basic Register m (DATBASm) (m = 0 to 7)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DVTYP	DVNACK[1:0]		—	—	—	—	—	DVDYAD[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DVIBITS	DVMRRJ	DVSIRRJ	DVIBIPL	—	—	—	—	—	DVSTAD[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	DVTYP	0b	R/W	Device Type 0: I3C Device 1: I^2C Device
30, 29	DVNACK[1:0]	00b	R/W	Device NACK Retry Count Device-specific retry count
28 to 24	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
23 to 16	DVDYAD[7:0]	All 0	R/W	Device I3C Dynamic Address Bit 23 is the parity bit, per the I3C specification, computed and updated by the software driver.
15	DVIBITS	0b	R/W	Device IBI Time-stamp 0: The Master shall not time-stamp IBIs from this Device with Master Time-stamps. 1: The Master shall time-stamp IBIs for this Device with Master Time-stamps.
14	DVMRRJ	0b	R/W	Device In-Band Master Request Reject 0: This Device shall ACK Master Requests. 1: This Device shall NACK Master Requests and send the auto-disable command.
13	DVSIRRJ	0b	R/W	Device In-Band Slave Interrupt Request Reject 0: This Device shall ACK the SIR. 1: This Device shall NACK the SIR and send the auto-disable CCC.
12	DVIBIPL	0b	R/W	Device IBI Payload 0: IBIs from this Device do not carry a Data Payload. 1: IBIs from this Device do carry a Data Payload.
11 to 7	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
6 to 0	DVSTAD[6:0]	All 0	R/W	Device Static Address I3C Static Address

Note: This register supports for I3C master mode and I3C secondary master mode.

DVIBIPL bit (Device IBI Payload)

Indicates whether IBIs from this Device have a Data Payload. This field reflects the IBI Payload bit in the Device's Bus Characteristics Register (BCR).

During IBI handling for this Device, the Master shall use this field to determine whether or not to drive reception of the IBI Data Payload. Data continuation is indicated by the T-Bit.

DVSIRRJ bit (Device In-Band Slave Interrupt Request Reject)

Controls whether this Device, when operating in the Master role, will accept vs. reject Slave Interrupt Requests from other Devices.

DVMRRJ bit (Device In-Band Master Request Reject)

Controls whether this Device, when operating in the Master role, will accept vs. reject Master requests from other Devices. This bit is only valid if I3C declares Non-Current Master Capability.

DVIBITS bit (Device IBI Time-stamp)

Enables or disables IBI time-stamping for a specific Device.

Note: The IBI Status Descriptor for each IBI event indicates whether or not the individual IBI event was actually time- stamped. Set to 0 except for Async mode 0 and Async mode 1 of timing control.

DVNACK[1:0] bits (Device NACK Retry Count)

These bits set the number of retries when a NACK response is received from the slave for the transaction set in the Command Descriptor.

Note: When ENTDAA is executed by Address Assign Command, the setting of this bit is ignored and the transaction ends when NACK is received once.

26.2.55 Slave Device Address Table Basic Register n (SDATBASn) (n = 0 to 2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	SDDYAD[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SDIBIPL	—	SDADLS	SDSTAD[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
22 to 16	SDDYAD[6:0] *1	All 0	R/W	Slave Device I3C Dynamic Address*5
15 to 13	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
12	SDIBIPL*1	0b	R	Slave Device IBI Payload*4 This bit is the mirror bit of the SVDCT.TBCR[2]. 0: IBIs from this device do not carry a data payload. 1: IBIs from this device carry a data payload.
11	—	0b	R	Reserved These bits are read as 0. The write value should be 0.
10	SDADLS	0b	R/W	Slave Device Address Length Selection*3 0: Slave device address length 7 bits selected. 1: Slave device address length 10 bits selected. (I^2C device only)
9 to 0	SDSTAD[9:0]	All 0	R/W	Slave Device Static Address*2 I3C / I^2C Static Address

Note: SW write to the SDATBAS register of the main master is prohibited.

Note 1. This bit is valid only in SDATBAS0 register.

Note 2. These bits support for I^2C, I3C secondary master, and I3C slave mode.

Note 3. This bit supports for I^2C mode.

Note 4. This bit supports for I3C secondary master mode and I3C slave mode.

Note 5. These bits support for I3C secondary master mode and I3C slave mode.

SDSTAD[9:0] bits (Slave Device Static Address)

When the 7-bit address format is selected (SDADLS bit is 0), the lower 7 bits of SDSTAD[9:0] function as the 7-bit address.

When the 10-bit address format is selected (SDADLS bit is 1), the SDSTAD[9:0] function as the 10-bit address. While the SVCTL.SVAEn bit is 0, the setting of this bit is ignored.

SDIBIPL bit (Slave Device IBI Payload)

Indicates whether IBIs from this Device have a Data Payload. This field reflects the IBI Payload bit in the Device's Bus Characteristics Register (BCR).

During IBI handling for this Device, the Master shall use this field to determine whether or not to drive reception of the IBI Data Payload. Data continuation is indicated by the T-Bit.

SDDYAD[6:0] bits (Slave Device I3C Dynamic Address)

[Update conditions]

- When writing Dynamic Address value.
- When Slave Address value is its own Static Address in receiving SETDASA CCC (Direct), these bits are updated to Dynamic Address value.*1
- When Dynamic Address Assignment procedure that starts by receiving ENTDAA CCC (Broadcast) is established.*1
- When receiving RSTDAA CCC (Broadcast), all bits are cleared to 0.*1
- When Slave Address value is its own Dynamic Address in receiving RSTDAA CCC (Direct), all bits are cleared to 0.*1
- When Slave Address value is its own Dynamic Address in receiving SETNEWDA CCC (Direct), these bits are updated to the Dynamic Address value.*1
- When receiving SETAASA CCC (Broadcast), these bits are updated to the value of SDSTAD[6:0] bits*2.

Note 1. See *the MIPI I3C Specification v1.0*.

Note 2. See *the MIPI I3C Basic Specification v1.0*.

26.2.56 Master Device Characteristic Table Register m (MSDCTm) (m = 0 to 7)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RBCR76[1:0]		–	RBCR4	RBCR3	RBCR2	RBCR1	RBCR0	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
15, 14	RBCR76[1:0]	00b	R/W	Device Role 0 0: I3C Slave 0 1: I3C Master*3 Others: Setting prohibited
13	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
12	RBCR4	0b	R/W	Bridge Identifier 0: Not a Bridge Device 1: Is a Bridge Device
11	RBCR3	0b	R/W	Offline Capable*2 0: Device will always respond to I3C bus commands. 1: Device will not always respond to I3C bus commands.
10	RBCR2	0b	R/W	IBI Payload 0: No data byte follows the accepted IBI. 1: Mandatory one or more data bytes follow the accepted IBI. Data byte continuation is indicated by T-Bit.
9	RBCR1	0b	R/W	IBI Request Capable 0: Not Capable 1: Capable
8	RBCR0	0b	R/W	Max Data Speed Limitation*1 0: No Limitation 1: Limitation
7 to 0	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.

Note: This register supports for I3C master mode and I3C secondary master mode.

Note 1. Master shall use the GETMXDS CCC to interrogate the Slave for specific limitation.

Note 2. Offline Capable Devices retain the Dynamic Address.

Note 3. For an I3C Device acting as I3C Main Master, the BCR Device Role bits will contain the value 01.

The DCT table captures the Device characteristics (PID, BCR, DCR) and assigned Dynamic Address of each Device on the I3C Bus that participates in the Dynamic Address Allocation (ENTDAA) procedure.

RBCRn bits (Received Bus Characteristic Register)

Each I3C Device that is connected to the I3C Bus shall have an associated read-only Bus Characteristics Register (BCR). This read-only register describes the I3C compliant Device's role and capabilities for use in Dynamic Address assignment and Common Command Codes.

Note: When RBCR[2] is 0 and when ACK response to Slave Interrupt Request from I3C Slave by DATBASm.DVSIRR J = 0, STOP Condition is issued after ACK response. When RBCR[2] is 1 and when ACK response to Slave Interrupt Request from I3C Slave by DATBASm.DVSIRRJ = 0, IBI Payload is received after ACK response. STOP Condition is issued after end of IBI Payload.

[Update condition]

- When receiving of Bus Characteristics Register (BCR) from Device in the Dynamic Address Assignment procedure starting by receiving ENTDAA CCC (Broadcast).*[1]

Note 1. See *the MIPI I3C Specification v1.0*.

26.2.57 Extended Device Address Table Basic Register (EXDATBAS)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EDTYP	EDNACK[1:0]		—	—	—	—	—	EDDYAD[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	EDSTAD[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	EDTYP	0b	R/W	Extended Device Type 0: I3C Device 1: I^2C Device
30, 29	EDNACK[1:0]	00b	R/W	Extended Device NACK Retry Count Device-specific retry count
28 to 24	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
23 to 16	EDDYAD[7:0]	All 0	R/W	Extended Device I3C Dynamic Address Bit 23 is the parity bit, per the I3C specification, computed and updated by the software driver.
15 to 7	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
6 to 0	EDSTAD[6:0]	All 0	R/W	Extended Device Static Address I3C / I^2C static address

Note: This register supports for I3C master mode and I3C secondary master mode.

26.2.58 Slave Device Characteristic Table Register (SVDCT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TBCR76[1:0]		—	TBCR4	TBCR3	TBCR2	TBCR1	TBCR0	TDCR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
15 to 14	TBCR76[1:0]	00b	R/W	Device Role 0 0: I3C Slave 0 1: I3C Master*3 1 0: Reserved for future definition by MIPI Sensor WG 1 1: Reserved for future definition by MIPI Sensor WG
13	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
12	TBCR4	0b	R/W	Bridge Identifier 0: Not a Bridge Device 1: Is a Bridge Device
11	TBCR3	0b	R/W	Offline Capable*2 0: Device will always respond to I3C bus commands. 1: Device will not always respond to I3C bus commands.
10	TBCR2	0b	R/W	IBI Payload 0: No data byte follows the accepted IBI. 1: Mandatory one or more data bytes follow the accepted IBI. Data byte continuation is indicated by T-Bit.
9	TBCR1	0b	R/W	IBI Request Capable 0: Not Capable 1: Capable
8	TBCR0	0b	R/W	Max Data Speed Limitation*1 0: No Limitation 1: Limitation
7 to 0	TDCR[7:0]	All 0	R/W	Transfar Device Characteristic Register 255 available codes for describing the type of sensor, or device. Examples: Accelerometer, gyroscope, composite devices Default value is 0: Generic Device

Note: This register supports for I3C secondary master mode and I3C slave mode.

Note 1. Master shall use the GETMXDS CCC to interrogate the Slave for specific limitation.

Note 2. Offline Capable Devices retain the Dynamic Address.

Note 3. For an I3C Device acting as I3C Main Master, the BCR Device Role bits will contain the value 01.

The DCT table captures the Device characteristics (PID, BCR, DCR) and assigned Dynamic Address of each device on the I3C bus that participates in the Dynamic Address Allocation (ENTDAA) procedure.

TDCR[7:0] bits (Transfar Device Characteristic Register)

Each I3C device that is connected to the I3C bus has an associated Device Characteristics Register (DCR). This register describes the I3C compliant device type such as accelerometer and gyroscope, for use in Dynamic Address assignment and Common Command Codes.

TBCRn bits (Transfar Bus Characteristic Register)

Each I3C device that is connected to the I3C bus has an associated Bus Characteristics Register (BCR). This register describes the role and capabilities of the I3C compliant device for use in Dynamic Address assignment and Common Command Codes.

When I3C Slave issues IBI by Command Descriptor, the condition of TBCR[7:0] is described as follows:

[Slave Interrupt Request: No IBI Payload follow the accepted IBI]

- TBCR1 = 1
- TBCR2 = 0

Note: Set DATA_LENGTH[15:0] of Command Descriptor to 0.

[Slave Interrupt Request: IBI Payload follow the accepted IBI]

- TBCR1 = 1
- TBCR2 = 1

Note: Set DATA_LENGTH[15:0] of Command Descriptor to any value.

[Mastership Request]

- TBCR1 = 1
- TBCR76[1:0] = 01b

[Hot-join Event]

- TBCR1 = 1

When I3C Slave receives CCC from I3C Master, it performs the following operations according to the setting of TBCR[7:0]:

- When TBCR2 = 1, CMRLG.IBIPSZ[7:0] is sent as the 3rd byte data to GETMRL CCC from I3C Master
- When TBCR0 = 0, NACK responses to GETMXDS CCC from I3C Master
- When TBCR0 = 1, ACK responses to GETMXDS CCC from I3C Master and sends data from CMDSPW, CMDSPR, and CMDSPT registers

26.2.59 Slave Device Characteristic Table Provisional ID Low Register (SDCTPIDL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	n/a	All 0	R/W	Transfar Device Provisional ID Low Bits 31 to 16 are read as 0. Bits 15 to 0 are bits [15:0] of device's I3C PID.

26.2.60 Slave Device Characteristic Table Provisional ID High Register (SDCTPIDH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	n/a	All 0	R/W	Transfar Device Provisional ID High Bits [47:16] of device's I3C PID.

Note: This register supports for I3C secondary master mode and I3C slave mode.

26.2.61 Slave Device Address Register n (SVDVADn) (n = 0 to 2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDYADV*4	SSTADV	—	—	SADLG	—	SVAD[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	SDYADV*4	0b	R	Slave Dynamic Address Valid*3 0: Dynamic Address is disabled. 1: Dynamic Address is enabled.
30	SSTADV	0b	R	Slave Static Address Valid*1 0: Slave address is disabled. 1: Slave address is enabled.
29, 28	—	00b	R	Reserved These bits are read as 0.
27	SADLG	0b	R	Slave Address Length*2 0: The 7-bit address format is selected. 1: The 10-bit address format is selected.
26	—	0b	R	Reserved This bit is read as 0.
25 to 16	SVAD[9:0]	All 0	R	Slave Address*1 A slave address is set. When rewriting SVAD, change to SVAE = 0 and rewrite.
15 to 0	—	All 0	R	Reserved These bits are read as 0.

Note 1. These bits support for I^2C, I3C secondary master, and I3C slave mode.

Note 2. This bit supports for I^2C mode.

Note 3. This bit supports for I3C secondary master mode and I3C slave mode.

Note 4. This bit is valid only in SVDVAD0 register.

SVAD[9:0] bits (Slave Address)

The SVAD[9:0] bits indicate a valid slave address.

[The SVDVAD0.SDYADV bit = 1]

Note: This condition is only for SVDVAD0.SVAD[9:0].

- The SVAD[9:7] bits = 0
- The SVAD[6:0] bits = the SDATBAS0.SDDYAD[6:0] bits

[The SVDVADy.SSTADV bit = 1 and the SVDVADy.SADLG bit = 0]

- The SVAD[9:7] bits = 0
- The SVAD[6:0] bits = the SDATBASy.SDSTAD[6:0] bits

[The SVDVADy.SSTADV bit = 1 and the SVDVADy.SADLG bit = 1]

- The SVAD[9:0] bits = the SDATBASy.SDSTAD[9:0] bits

SADLG bit (Slave Address Length)

[Setting conditions]

- All of the followings are satisfied:

1. The PRTS.PRTMD bit = 1 (I^2C Protocol mode)
2. The SVCTL.SVAEy bit = 1 (Slave y is enabled)
3. The SDATBASy.SDADLS bit = 1 (The address length is 10 bits)

[Clearing condition]

- [Setting condition] is not satisfied.

SSTADV bit (Slave Static Address Valid)

[Setting conditions]

- All of the followings are satisfied:

1. The SVCTL.SVAEy bit = 1 (Slave y is enabled)
2. The SVDVAD0.SDYADV bit = 0 (Dynamic Address is disabled)

   Note: This condition is only for SVDVAD0.SSTADV.
3. If the SVDVADy.SADLG bit = 0, the SDATBASy.SDSTAD[6:0] bits are not all 0
   If the SVDVADy.SADLG bit = 1, the SDATBASy.SDSTAD[9:0] bits are not all 0

[Clearing condition]

- [Setting condition] is not satisfied.

SDYADV*4 bit (Slave Dynamic Address Valid)

[Setting conditions]

- All of the followings are satisfied:

1. The PRTS.PRTMD bit = 0 (I3C Protocol mode)
2. The SVCTL.SVAEy bit = 1 (Slave y is enabled)
3. The SDATBAS0.SDDYAD[6:0] bits are not all 0

   Note: This condition is only for SVDVAD0.SDYADV.

[Clearing condition]

- [Setting condition] is not satisfied.

26.2.62 CCC Slave Events Command Register (CSECMD)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	HJEVE	—	MSRQE	SVIRQE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
3	HJEVE	0b	R/W	Hot-Join Event Enable 0: DISABLED: Slave-initiated Hot-Join is Disabled by the Master to control. 1: ENABLED: Slave-initiated Hot-Join is Enabled by the Master to control.
2	—	0b	R	Reserved This bit is read as 0. The write value should be 0.
1	MSRQE	0b	R/W	Mastership Requests Enable 0: DISABLED: Mastership requests from Secondary master's is Disabled by the Current Master to control. 1: ENABLED: Mastership requests from Secondary master's is Enabled by the Current Master to control.
0	SVIRQE	0b	R/W	Slave Interrupt Requests Enable 0: DISABLED: Slave-initiated Interrupts is Disabled by the Master to control. 1: ENABLED: Slave-initiated Interrupts is Enabled by the Master to control.

Note: This register supports for I3C secondary master mode and I3C slave mode.

SVIRQE bit (Slave Interrupt Requests Enable)

This bit allows the Master to control when Slave-initiated Interrupts are allowed on the I3C Bus.

These four Direct (ENEC/DISEC Format 1) or Broadcast (ENEC/DISEC Format 2) CCCs allows the Master to control when Slave- initiated traffic is (Enable) vs. is not (Disable) allowed on the I3C Bus. This control governs a Slave's attempts to request an Interrupt (ENI), to request Mastership (ENMR), or to signify a Hot-Join event (ENHJ).

[Setting conditions]

- When writing 1
- When receiving ENEC CCC (Broadcast) with ENINT bit = 1.*1
- When ENINT bit = 1 with own Slave Address in receiving ENEC CCC (Direct).*1

[Clearing conditions]

- When writing 0.
- When receiving DISEC CCC (Broadcast) with DISINT bit = 1.*1
- When DISINT bit = 1 with own Slave Address in receiving DISEC CCC (Direct).*1

MSRQE bit (Mastership Requests Enable)

This bit allows the Current Master to control when Mastership requests from Secondary masters are allowed on the I3C Bus.

[Setting conditions]

- When writing 1.
- When receiving ENEC CCC (Broadcast) with ENMR bit = 1.*1
- When ENMR bit = 1 with own Slave Address in receiving ENEC CCC (Direct).*1

[Clearing conditions]

- When writing 0.
- When receiving DISEC CCC (Broadcast) with DISMR bit = 1.*1
- When DISMR bit = 1 with own Slave Address in receiving DISEC CCC (Direct).*1

HJEVE bit (Hot-Join Event Enable)

This bit allows the Master to control when Slave-initiated Hot-Join is allowed on the I3C Bus.

[Setting conditions]

- When writing 1.
- When receiving ENEC CCC (Broadcast) with ENHJ bit = 1.*1
- When ENHJ bit = 1 with own Slave Address in receiving ENEC CCC (Direct).*1

[Clearing conditions]

- When writing 0.
- When receiving DISEC CCC (Broadcast) with DISHJ bit = 1.*1
- When DISHJ bit = 1 with own Slave Address in receiving DISEC CCC (Direct).*1

Note 1. See *the MIPI I3C Specification v1.0*.

26.2.63 CCC Enter Activity State Register (CEACTST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	ACTST[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
3 to 0	ACTST[3:0]	All 0	R/W	Activity State H'1: ENTAS0 (1 µs: Latency-free operation) H'2: ENTAS1 (100 µs) H'4: ENTAS2 (2 ms) H'8: ENTAS3 (50 ms: Lowest-activity operation) Others: Setting prohibited

Note: This register supports for I3C secondary master mode and I3C slave mode.

ACTST[3:0] bits (Activity State)

[Update conditions]

- When writing Activity State value.
- When receiving ENTAS0 CCC (Broadcast), these bits are updated to H'1.*1
- When receiving ENTAS1 CCC (Broadcast), these bits are updated to H'2.*1
- When receiving ENTAS2 CCC (Broadcast), these bits are updated to H'4.*1
- When receiving ENTAS3 CCC (Broadcast), these bits are updated to H'8.*1
- When Slave Address value is its own Slave Addres in receiving ENTAS0 CCC (Direct), these bits are updated to H'1.*1
- When Slave Address value is its own Slave Addres in receiving ENTAS1 CCC (Direct), these bits are updated to H'2.*1
- When Slave Address value is its own Slave Addres in receiving ENTAS2 CCC (Direct), these bits are updated to H'4.*1
- When Slave Address value is its own Slave Addres in receiving ENTAS3 CCC (Direct), these bits are updated to H'8.*1

Note 1. See *the MIPI I3C Specification v1.0*.

26.2.64 CCC Max Write Length Register (CMWLG)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MWLG[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
15 to 0	MWLG[15:0]	All 0	R/W	Max Write Length

Note: This register supports for I3C secondary master mode and I3C slave mode.

MWLG[15:0] bits (Max Write Length)

These bits use for the I3C Master to set or get a maximum data write length in bytes for one Slave Device.

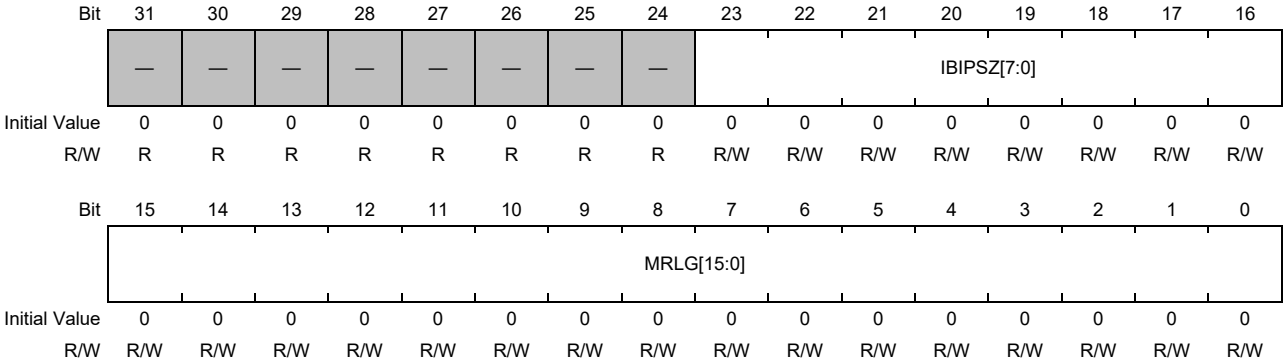
This Max Write Length does not affect data write lengths for Broadcast CCCs. The Set/Get Max Write Length value is transmitted over two bytes, with the most significant byte (MSB) transmitted first. The minimum value that Maxes Write Length can be set to is 8.

[Update conditions]

- When writing Max Write Length value.
- When receiving SETMWL CCC (Broadcast), these bits are updated to MWL value.*[1]
- When Slave Address value is its own Slave Address in receiving SETMWL CCC (Direct), these bits are updated to MWL value.*[1]

Note 1. See *the MIPI I3C Specification v1.0*.

26.2.65 CCC Max Read Length Register (CMRLG)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
23 to 16	IBIPSZ[7:0]	All 0	R/W	IBI Payload Size
15 to 0	MRLG[15:0]	All 0	R/W	Max Read Length

Note: This register supports for I3C secondary master mode and I3C slave mode.

MRLG[15:0] bits (Max Read Length)

These bits use for the I3C Master to set or get a maximum data read length for one Slave Device.

The Set/Get Max Read Length value is transmitted over the first two bytes, with most significant byte (MSB) transmitted first. The minimum value to which Max Read Length can be set is 16.

[Update conditions]

- When writing Max Read Length value.
- When receiving SETMRL CCC (Broadcast), these bits are updated to MRL value.*[1]
- When Slave Address value is its own Slave Address in receiving SETMRL CCC (Direct), these bits are updated to MRL value.*[1]

IBIPSZ[7:0] bits (IBI Payload Size)

These bits use for the I3C Master to set or get optionally a maximum IBI payload size.

For devices with BCR bit 2 set to 1, the Max IBI payload size value is added as a third byte, where a value of 0 indicates an unlimited payload size.

This CCC is optional for the Slave, with two exceptions:

1. This CCC is required if both (a) any private Read Request Message (s) and/or any extended Read Request CCC (s) implemented by the Slave support a variable limit on the maximum number of data bytes that the Slave may return per Message, and (b) this limit is greater than 16 bytes.
2. This CCC is required if the Slave both (a) supports an IBI Payload (as indicated with BCR bit 1), and (b) will transmit more than one byte of private payload.

[Update conditions]

- When writing Max IBI payload size value.
- When receiving SETMRL CCC (Broadcast), these bits are updated to IBI payload size value.*[1]

- When Slave Address value is its own Slave Address in receiving SETMRL CCC (Direct), these bits are updated to IBI payload size value.*[1]

Note 1. See *the MIPI I3C Specification v1.0*.

26.2.66 CCC Enter Test Mode Register (CETSTMD)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TSTMD[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0.
7 to 0	TSTMD[7:0]	All 0	R	Test Mode H'00: Exit Test Mode This value removes all I3C devices from Test Mode. H'01: Vendor Test Mode This value indicates that I3C devices shall return a random 32bit value in the provisional ID during the Dynamic Address Assignment procedure. Others: MIPI reserved Reserved for future use by the MIPI Alliance

Note: This register supports for I3C secondary master mode and I3C slave mode.

TSTMD[7:0] bits (Test Mode)

When these bits set to H'00, all I3C Devices remove from Test Mode.

When these bits set to H'01, I3C Devices shall return a random 32bit value in the Provisional ID during the Dynamic Address Assignment procedure.

The Broadcast CCC informs all I3C Devices that the Master is entering a specified Test Mode during manufacturing or Device test. The Enter Test Mode command Frame format includes a byte that specifies which Test Mode to enter.

Supporting I3C Devices shall enter the indicated Test Mode upon receipt of the Enter Test Mode CCC.

[Update condition]

- When receiving ENTTM CCC (Broadcast), these bits are updated to Test Mode Byte value.*[1]

Note 1. See *the MIPI I3C Specification v1.0*.

26.2.67 CCC Get Device Status Register (CGDVST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VDRSV[7:0]								ACTMD[1:0]		PRTE	—	PNDINT[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
15 to 8	VDRSV[7:0]	All 0	R/W	Vendor Reserved Reserved for vendor-specific meaning
7, 6	ACTMD[1:0]	00b	R/W	Slave Device's current Activity Mode 0 0: Activity Mode 0 0 1: Activity Mode 1 1 0: Activity Mode 2 1 1: Activity Mode 3
5	PRTE	0b	R	Protocol Error 0: The Slave has not detected a protocol error since the last Status read. 1: The Slave has detected a protocol error since the last Status read.
4	—	0b	R	Reserved This bit is read as 0. The write value should be 0.
3 to 0	PNDINT[3:0]	All 0	R/W	Pending Interrupt Contains the interrupt number of any pending interrupt, or 0 if no interrupts are pending. This encoding allows for up to 15 numbered interrupts. If more than one interrupt is set, then the highest priority interrupt shall be returned.

Note: This register supports for I3C secondary master mode and I3C slave mode.

PRTE bit (Protocol Error)

If this bit set to 1, then the Slave detects a protocol error since the last Status read.

The Slave checks for such errors. Note that this value self-clears by the hardware upon every successful completion of a Master read of the Slave's Status.

The Direct CCC is a Get request for one I3C Slave Device to return its current Status, in the two-byte format detailed. Note that byte 0 is the LSb, and byte 1 is the MSB.

[Setting condition]

- When the Slave detected a protocol error.*1

[Clearing condition]

- When transmission by own Slave Address is completed without error after receiving GETSTATUS CCC (Direct).*1

ACTMD[1:0] bits (Slave Device's current Activity Mode)

Contains the two-bit ID of the Slave Device's current Activity Mode (readiness to support data read of sensor or related information).

Note 1. See *the MIPI I3C Specification v1.0*.

26.2.68 CCC Max Data Speed W (Write) Register (CMDSPW)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	MSWDR[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
2 to 0	MSWDR[2:0]	000b	R/W	Maximum Sustained Write Data Rate 000: fscl Max (default value) 001: 8 MHz 010: 6 MHz 011: 4 MHz 100: 2 MHz Others: Setting prohibited

Note: This register supports for I3C secondary master mode and I3C slave mode.

26.2.69 CCC Max Data Speed R (Read) Register (CMDSPR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CDTTIM[2:0]			MSRDR[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
5 to 3	CDTTIM[2:0]	000b	R/W	Clock to Data Turnaround Time (TSCO) 000: 8 ns or less (default value) 001: 9 ns or less 010: 10 ns or less 011: 11 ns or less 100: 12 ns or less 111: TSCO is more than 12 ns and is reported by private agreement. Others: Setting prohibited
2 to 0	MSRDR[2:0]	000b	R/W	Maximum Sustained Read Data Rate 000: fscl Max (default value) 001: 8 MHz 010: 6 MHz 011: 4 MHz 100: 2 MHz Others: Setting prohibited

Note: This register supports for I3C secondary master mode and I3C slave mode.

26.2.70 CCC Max Data Speed T (Turnaround) Register (CMDSPT)

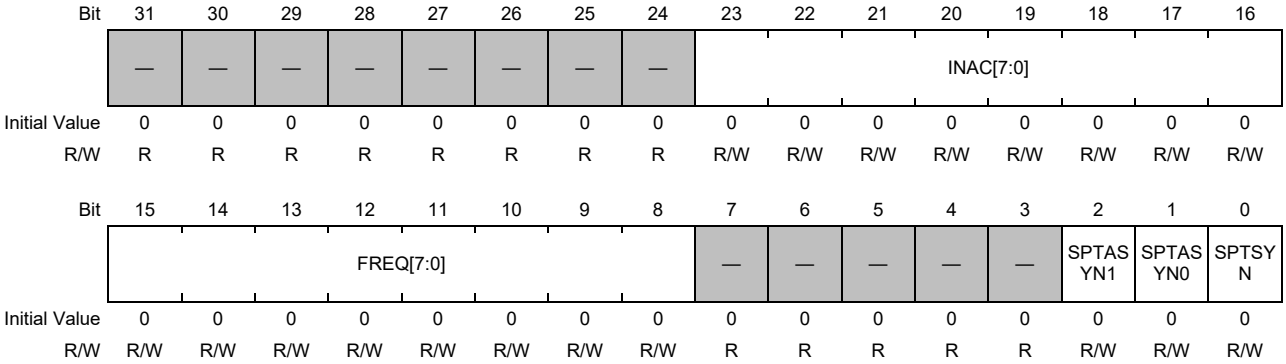
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MRTE	—	—	—	—	—	—	—	MRTTIM[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MRTTIM[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	MRTE	0b	R/W	Maximum Read Turnaround Time Enable 0: Disables transmission of the Maximum Read Turnaround Time. (GETMXDS Format 1: Without Turnaround) 1: Enables transmission of the Maximum Read Turnaround Time. (GETMXDS Format 2: With Turnaround)
30 to 24	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
23 to 0	MRTTIM [23:0]	All 0	R/W	Maximum Read Turnaround Time 24-bit field can encode turnaround times from 0.0 seconds to 16 seconds. H'F4_0000: 0 µs (minimum value) H'F4_0001: 1 µs (resolution) ⋮ H'F4_2400: 16 seconds (maximum value) H'F4_2401: Setting prohibited ⋮ H'FF_FFFF: Setting prohibited

Note: This register supports for I3C secondary master mode and I3C slave mode.

26.2.71 CCC Exchange Timing Support Information M (Mode) Register (CETSM)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
23 to 16	INAC[7:0]	All 0	R/W	Inaccuracy Byte H'00: 0.0% H'0F: 1.5% H'1F: 3.1% H'2F: 4.7% H'3F: 6.3% H'4F: 7.9% H'5F: 9.5% H'6F: 11.1% H'7F: 12.7% H'8F: 14.3% H'9F: 15.9% H'AF: 17.5% H'BF: 19.1% H'CF: 20.7% H'DF: 22.3% H'EF: 23.9% H'FF: 25.5%
15 to 8	FREQ[7:0]	All 0	R/W	Frequency Byte This byte represents the Slave's internal oscillator frequency in increments of 0.5 MHz (500 kHz), up to 127.5 MHz. H'00: 32.0 KHz H'0F: 7.5 MHz H'1F: 15.5 MHz H'2F: 23.5 MHz H'3F: 31.5 MHz H'4F: 39.5 MHz H'5F: 47.5 MHz H'6F: 55.5 MHz H'7F: 63.5 MHz Others: Setting prohibited
7 to 3	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
2	SPTASYN1	0b	R/W	Support Async Mode 1 0: Async Mode 1 is not supported. 1: Async Mode 1 is supported.

Bit	Bit Name	Initial Value	R/W	Description
1	SPTASYN0	0b	R/W	Support Async Mode 0 0: Async Mode 0 is not supported. 1: Async Mode 0 is supported.
0	SPTSYN	0b	R/W	Supports Sync Mode 0: Sync Mode is not supported. 1: Sync Mode is supported.

Note: This register supports for I3C secondary master mode and I3C slave mode.

The Directed CCC provides the framework for the Master to query the Exchange Timing capabilities supported by the I3C Slaves. The Get Exchange Timing Support Information CCC causes the addressed Slave to return four data bytes containing key information on supported current state, and internal oscillator/clock frequency and inaccuracy.

26.2.72 CCC Exchange Timing Support Information S (State) Register (CETSS)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ICOVF	—	—	—	—	ASYNE[1:0]		SYNE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7	ICOVF	0b	R/W	Internal Counter Overflow 0: Slave has not experienced a counter overflow since the most recent previous check. 1: Slave experienced a counter overflow since the most recent previous check.
6 to 3	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
2, 1	ASYNE[1:0]	00b	R/W	Async Mode Enabled 00: All Mode Disable 01: Async Mode 0 Enabled. 10: Async Mode 1 Enabled. Other Setting prohibited
0	SYNE	0b	R/W	Sync Mode Enabled 0: Sync Mode Disabled. 1: Sync Mode Enabled.

Note: This register supports for I3C secondary master mode and I3C slave mode.

ASYNE[0] Bit

Slave Timing Control Async Mode 0 is enabled.

[Setting condition]

- When writing “1”.
- When CETSM.SPTASYN[0] bit = 1 and either of the following 1 or 2 are satisfied.

1. When receiving SETXTIME CCC (Broadcast) with Defining byte value H’DF.
2. When Slave Address value is its own Slave Address in receiving SETXTIME CCC (Direct) with Defining byte value H’DF.

[Clearing condition]

- When writing “0”.
- When CETSM.SPTASYN[0] bit = 1 and either of the following 1 or 2 are satisfied.

1. When receiving SETXTIME CCC (Broadcast) with Defining byte value H’EF.
2. When Slave Address value is its own Slave Address in receiving SETXTIME CCC (Direct) with Defining byte value H’EF.

ASYNE[1] Bit

Slave Timing Control Async Mode 1 is enabled.

[Setting condition]

- When writing “1”.
- When CETSM.SPTASYN[1] bit = 1 and either of the following 1 or 2 are satisfied.

1. When receiving SETXTIME CCC (Broadcast) with Defining byte value H’EF.
2. When Slave Address value is its own Slave Address in receiving SETXTIME CCC (Direct) with Defining byte value H’EF.

[Clearing condition]

- When writing “0”.
- When CETSM.SPTASYN[1] bit = 1 and either of the following 1 or 2 are satisfied.

1. When receiving SETXTIME CCC (Broadcast) with Defining byte value H’DF.
2. When Slave Address value is its own Slave Address in receiving SETXTIME CCC (Direct) with Defining byte value H’DF.

ICOVF bit

Bit mask indicating which Timing Control Mode (if any) is currently enabled for the target Slave, and whether any counter overflows have occurred since the most recent previous check. If a Timing Control Mode bit is set (has value 1b), then that Slave has currently enabled the corresponding Timing Control Mode. If the Overflow bit is set (has value 1b), then that Slave experienced a counter overflow since the most recent previous check.

26.2.73 Bit Count Register (BITCNT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	BCNT[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are read as 0.
4 to 0	BCNT[4:0]	All 0	R	Bit Counter Indicates the number of bits remaining to be transferred. For details on the values, refer to **Table 26.8** and **Table 26.9**.

BCNT[4:0] bits (Bit Counter)

These bits function as a counter that indicates the number of bits remaining to be transferred at the detection of a sampling edge on the SCLn line.

Table 26.8 I^2C / Legacy I^2C transfer

	Master		Slave	
BCNT[4:0]	Address phase	Data phase	Address phase	Data phase
H'00	2 to 1 bits	2 to 1 bits	3 to 1 bits	2 to 1 bits
H'01	3 bits	3 bits	4 bits	3 bits
H'02	4 bits	4 bits	5 bits	4 bits
H'03	5 bits	5 bits	6 bits	5 bits
H'04	6 bits	6 bits	7 bits	6 bits
H'05	7 bits	7 bits	8 bits	7 bits
H'06	8 bits	8 bits	9 bits	8 bits
H'07	9 bits	9 bits	—	9 bits

Table 26.9 I3C transfer

	SDR*1		HDR-DDR		
BCNT[4:0]	Transmission	Reception	Command/Data	CRC	HDR-TS
H'00	1 bit	2 to 1 bits	19, 1 bits	11, 1 bits	1 Symbol
H'01	2 bits	3 bits	20, 2 bits	12, 2 bits	2 Symbols
H'02	3 bits	4 bits	3 bits	3 bits	3 Symbols
H'03	4 bits	5 bits	4 bits	4 bits	4 Symbols
H'04	5 bits	6 bits	5 bits	5 bits	5 Symbols
H'05	6 bits	7 bits	6 bits	6 bits	6 Symbols
H'06	7 bits	8 bits	7 bits	7 bits	7 Symbols
H'07	8 bits	9 bits	8 bits	8 bits	8 Symbols
H'08	9 bits	—	9 bits	9 bits	9 Symbols
H'09	—	—	10 bits	10 bits	10 Symbols
H'0A	—	—	11 bits	—	11 Symbols
H'0B	—	—	12 bits	—	12 Symbols
H'0C	—	—	13 bits	—	—
H'0D	—	—	14 bits	—	—
H'0E	—	—	15 bits	—	—
H'0F	—	—	16 bits	—	—
H'10	—	—	17 bits	—	—
H'11	—	—	18 bits	—	—

Note 1. The address phase is the same as in **Table 26.8**.

26.2.74 Normal Queue Status Level Register (NQSTLV)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	IBISCNT[4:0]					IBIQLV[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSPQLV[7:0]								CMDQFLV[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved These bits are read as 0.
28 to 24	IBISCNT[4:0]	All 0	R	Normal IBI Status Count*2 Number of IBI Status entries currently in the IBI Queue.
23 to 16	IBIQLV[7:0]	All 0	R	Normal IBI Queue Level*1 Number of buffer entries currently in the IBI Queue.
15 to 8	RSPQLV[7:0]	All 0	R	Normal Response Queue Level*1 Number of buffer entries currently in the Response Queue.
7 to 0	CMDQFLV [7:0]	H'04	R	Normal Command Queue Free Level*1 Number of free buffer entries currently in the Command Queue. Reset value is the depth of the Command Queue.

Note 1. These bits support for all I3C mode.

Note 2. These bits support for I3C master mode and I3C secondary master mode.

26.2.75 Normal Data Buffer Status Level Register 0 (NDBSTLV0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDBLV[7:0]								TDBFLV[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0.
15 to 8	RDBLV[7:0]	All 0	R	Normal Receive Data Buffer Level Indicates the number of Receive Data Buffer entries in the Receive Data Queue.
7 to 0	TDBFLV[7:0]	H'01	R	Normal Transmit Data Buffer Free Level Indicates the number of free Transmit Data Buffer entries in the Transmit Data Queue. Reset value is the depth of the Transmit Data Queue.

Note: This register supports for all I3C mode.

26.2.76 Normal Receive Status Queue Status Level Register (NRSQSTLV)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RSQLV[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0.
7 to 0	RSQLV[7:0]	All 0	R	Normal Receive Status Queue Level

Note: This register supports for I3C secondary master mode and I3C slave mode.

26.2.77 Present State Debug Register (PRSTDBG)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SDOLV	SCOLV	SDILV	SCILV
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	SDOLV	1b	R	SDA Output Level 0: I3C has driven the SDA pin low. 1: I3C has released the SDA pin.
2	SCOLV	1b	R	SCL Output Level 0: I3C has driven the SCL pin low. 1: I3C has released the SCL pin.
1	SDILV	1b	R	SDA Line Signal Level This bit is used to check the SDA Line level, in order to recover from errors and for debugging.
0	SCILV	1b	R	SCL Line Signal Level This bit is used to check the SCL Line level, in order to recover from errors and for debugging.

SDOLV bit (SDA Output Level) and SCOLV bit (SCL Output Level)

When reading these bits, the state of signals output from I3C can be read.

SCILV bit (SCL Line Signal Level)

This bit is used to check the SCL Line level, in order to recover from errors and for debugging.

SDILV bit (SDA Line Signal Level)

This bit is used to check the SDA Line level, in order to recover from errors and for debugging.

SCOLV bit (SCL Output Level)

This bit is used to select the output level of SCL pin.

SDOLV bit (SDA Output Level)

This bit is used to select the output level of SDA pin.

26.2.78 Master Error Counters Register (MSERRCNT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	M2ECNT[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
30 to 8	—	All 0	R	Reserved These bits are read as 0.
7 to 0	M2ECNT[7:0]	All 0	R	M2 Error Counter Counts I3C Type M2 errors on the I3C Bus. Cleared upon read out.

Note: This register supports for I3C master mode and I3C secondary master mode.

26.2.79 SC1 Capture monitor Register (SC1CPT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC1C[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0.
15 to 0	SC1C[15:0]	All 0	R	SC1 Capture

Note: This register supports for all I3C mode.

SC1C[15:0] Bits

- Async Mode 0 (Asynchronous Basic Mode)

  After enabling ATCCNTE.ATCE, SC1 Counter counts up from SC1 count trigger*1 to SCL rise edge next to ACK for the IBI and capture it as SC1.

- Async Mode 1 (Asynchronous Advanced Mode)

  After enabling ATCCNTE.ATCE, SC1 Counter counts up from SC1 count trigger*1 to the first aME and capture it as SC1.

Note 1. SW or external trigger can be selected by selection bits.

CAUTION

As the timing control specification, the SC1 counter value is included in the IBI frame as IBI data and is sent to I3C Master, therefore it is not necessary for the I3C Slave to read this register. If the I3C Slave needs to read this register, read it after completing the IBI frame.

26.2.80 SC2 Capture monitor Register (SC2CPT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC2C[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0.
15 to 0	SC2C[15:0]	All 0	R	SC2 Capture

Note: This register supports for all I3C mode.

SC2C[15:0] Bits

- Async Mode 0 (Asynchronous Basic Mode)

  After enabling ATCCNTE.ATCE, SC2 Counter counts up from SC2 count trigger*1 to SCL rise edge next to ACK for the IBI and capture it as SC2.

- Async Mode 1 (Asynchronous Advanced Mode)

  After enabling ATCCNTE.ATCE, SC2 Counter counts up from SC2 count trigger*1 to the first aME and capture it as SC2.

Note 1. SW or external trigger can be selected by selection bits.

CAUTION

As the timing control specification, the SC2 counter value is included in the IBI frame as IBI data and is sent to I3C Master, therefore it is not necessary for the I3C Slave to read this register. If the I3C Slave needs to read this register, read it after completing the IBI frame.

26.3 Operation

26.3.1 Data Structures

26.3.1.1 Command Descriptor

The write-only Command Descriptor structure is 64 bits in length. The Command Descriptor is put to the Command Queue with writes to the Command Queue Port.

Write to the Command Queue Port in the following order:

1. First write: The least significant DWORD (Command Descriptor Structure Low).
2. Second write: The most significant DWORD (Command Descriptor Structure High).

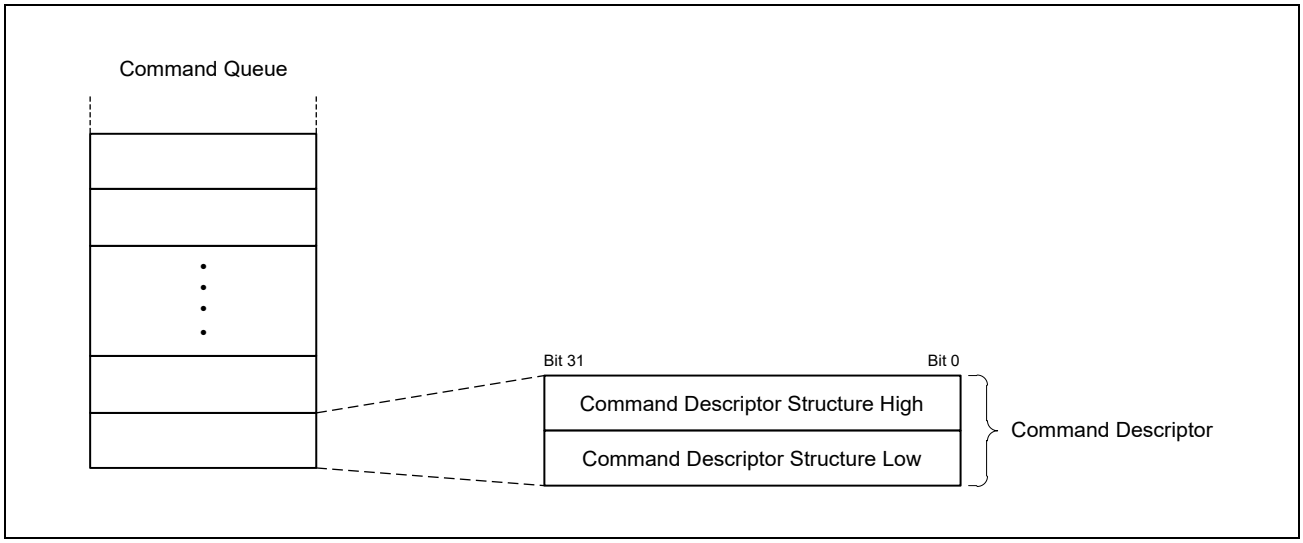


Figure 26.2 Command Descriptor Data Structure

I3C provides a Command Descriptor structure for each command type as follows:

- Address Assign Command
- Immediate Transfer Command
- Regular Transfer Command
- Combo Transfer Command
- Internal Control Command

Details are explained in the following sections.

(1) Address Assign Command

This command is used for address assignment (ENTDAA, SETDASA).

Remark When issuing SETAASA CCC, use the Immediate Transfer command.

The I3C provides an address assign command for the following mode:

- I3C Master Mode

Details of the Address Assign command structure are as follows.

Bit position	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
Bit field	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
Bit field	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field	TOC	ROC	DEV_COUNT[3:0]				—	—	—	—	EXT_DEVICE	DEV_INDEX[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field	—	CMD[7:0]								TID[3:0]				CMD_ATTR[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Bit Name	R/W	Description
63 to 32	—	W	The write value should be 0.
31	TOC	W	Terminate on Completion 0: RESTART: Issue Repeated START (Sr) at end of transfer 1: STOP: Issue STOP (P) at end of transfer
30	ROC	W	Response on Completion 0: NOT_REQUIRED: Response Status is not required. 1: REQUIRED: Response Status is required.
29 to 26	DEV_COUNT[3:0]	W	Device Count
25 to 22	—	W	The write value should be 0.
21	EXT_DEVICE	W	Extended Device Index 0: Use the DATBASm table indicated by DEV_INDEX[4:0]. 1: Use the EXDATBAS table.
20 to 16	DEV_INDEX[4:0]	W	Device Index
15	—	W	The write value should be 0.
14 to 7	CMD[7:0]	W	Transfer Command CCC Value
6 to 3	TID[3:0]	W	Transaction ID

Bit	Bit Name	R/W	Description
2 to 0	CMD_ATTR[2:0]	W	Command Attributes 000: XFER: Regular Transfer 001: IMMED_DATA_XFER: Immediate Data Transfer 010: ADDR_ASSGN_CMD: Address Assignment Command 011: WWR_COMBO_XFER: Write + Write/Read Combo Transfer 111: INTERNAL_CONTROL: Internal Control command Others: Setting prohibited

CMD_ATTR[2:0] bits (Command Attributes)

Command Type, defining the format of the other fields.

TID[3:0] bits (Transaction ID)

is reflected in the Response Descriptor.

CMD[7:0] bits (Transfer Command CCC Value)

Specifies CCC code indicating whether Address Assignment uses ENTDAA or SETDASA commands. The field comprises the entire command code (ENTDAA or SETDASA).

DEV_INDEX[4:0] bits (Device Index)

Indicates the DATBASm table index for the Slave device being addressed with the transfer. Static and device addressing related information are stored to this index in the DATBASm.

DEV_COUNT[3:0] bits (Device Count)

Indicates the number of devices that a dynamic address is assigned to.

ROC bit (Response on Completion)

Controls whether Response Status is sent after successful completion of the Transfer command. The successful completion is read from register NRSPQP. Upon unsuccessful transfer the Response Status is sent.

TOC bit (Terminate on Completion)

Controls what bus condition to issue after the Transfer command completes.

For ENTDAA, a STOP condition is issued regardless of the setting value of TOC. It is meaningful for SETDASA transfers.

When sending SETDASA CCC by TOC = 0 (RESTART), the next command must be set to SETDASA CCC with the Address Assign Command.

When the next command is not the same SETDASA CCC flame, it must be set to TOC = 1 (STOP).

(2) Immediate Transfer Command

This structure directly contains data (max 4 bytes) to be transferred, and as a result is only useful for Transfers/CCCs that write data. This structure shall not be used for Read operations (for example, to receive data).

When transmitting data of 4 bytes or less, use this Immediate Transfer Command to communicate.

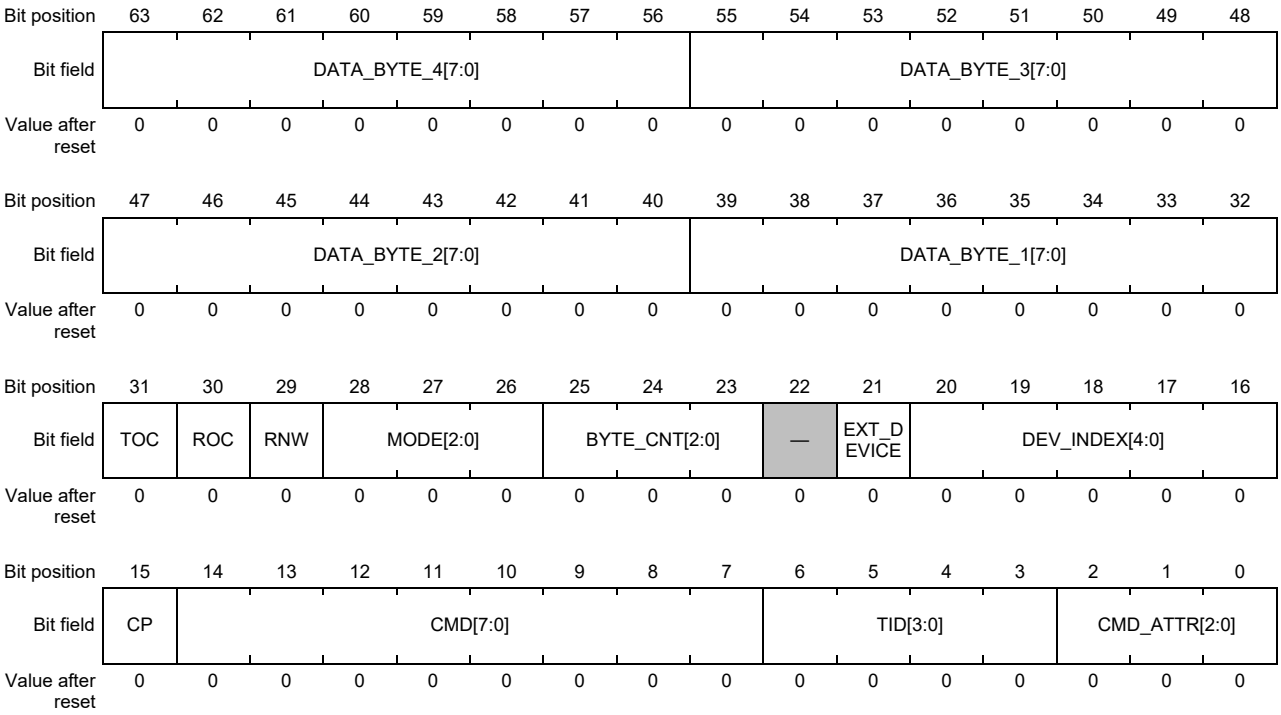
When transmitting data of 5 bytes or more, use the Regular Transfer Command to communicate.

For the Regular Transfer Command, refer to **Section 26.3.1.1(3), Regular Transfer Command**.

I3C provides an Immediate Transfer Command for the following mode:

- I3C Master Mode

Details of the Immediate Transfer Command Structure of each mode are shown in this section.



Bit	Bit Name	R/W	Description
63 to 56	DATA_BYTE_4[7:0]	W	Immediate Data Transfer Data Byte 4 Direct argument
55 to 48	DATA_BYTE_3[7:0]	W	Immediate Data Transfer Data Byte 3 Direct argument
47 to 40	DATA_BYTE_2[7:0]	W	Immediate Data Transfer Data Byte 2 Direct argument
39 to 32	DATA_BYTE_1[7:0]	W	Immediate Data Transfer Data Byte 1 Direct argument
31	TOC	W	Immediate Data Transfer Terminate on Completion 0: RESTART: Issue Repeated START (Sr) at end of data transfer 1: STOP: Issue STOP (P) at end of data transfer
30	ROC	W	Immediate Data Transfer Response on Completion 0: NOT_REQUIRED: Response Status is not required. 1: REQUIRED: Response Status is required.
29	RNW	W	Immediate Data Transfer R/W 0: WRITE: Write transfer 1: READ: Read transfer

Bit	Bit Name	R/W	Description
28 to 26	MODE[2:0]	W	Immediate Data Transfer Mode and Speed Values for I3C Mode: 000: I3C SDR0 Data rate: STDBR (e.g., 12.5 MHz) 001: I3C SDR1 Data rate: EXTBR (e.g., 8 MHz) 010: I3C SDR2 Data rate: STDBR × 2 (e.g., 6.25 MHz) 011: I3C SDR3 Data rate: EXTBR × 2 (e.g., 4 MHz) 100: I3C SDR4 Data rate: EXTBR × 4 (e.g., 2 MHz) Others: Setting prohibited Values for I^2C Mode: 000: Legacy I2C Message 0 Data rate: STDBR (e.g., 400 kHz) 001: Legacy I^2C Message 0 Data rate: EXTBR (e.g., 1 MHz) Others: Setting prohibited
25 to 23	BYTE_CNT[2:0]	W	Immediate Data Transfer Byte Count 000: No payload 001 to 100: N bytes are valid. Others: Setting prohibited
22	—	W	The write value should be 0.
21	EXT_DEVICE	W	Immediate Data Transfer Extended Device Index 0: Use the DATBASm table indicated by DEV_INDEX[4:0]. 1: Use the EXDATBAS table.
20 to 16	DEV_INDEX[4:0]	W	Immediate Data Transfer Device Index
15	CP	W	Immediate Data Transfer Command Present 0: TRANFSER: This structure describes an SDR transfer, so the CMD field is not valid. 1: CCC: This structure describes a CCC transfer, so the CMD field is valid.
14 to 7	CMD[7:0]	W	Immediate Data Transfer CCC Value For CCC: 8 bits
6 to 3	TID[3:0]	W	Immediate Data Transfer Transaction ID
2 to 0	CMD_ATTR[2:0]	W	Immediate Data Transfer Command Attribute 000: XFER: Regular Transfer 001: IMMED_DATA_XFER: Immediate Data Transfer 010: ADDR_ASSGN_CMD: Address Assignment Command 011: WWR_COMBO_XFER: Write + Write/Read Combo Transfer 111: INTERNAL_CONTROL: Internal Control command Others: Setting prohibited

CMD_ATTR[2:0] bits (Immediate Data Transfer Command Attribute)

Command Type, defining the format of the other fields.

TID[3:0] bits (Immediate Data Transfer Transaction ID)

Used as an identification tag for this command. This field shall be populated by the software Driver, and the same value shall be reflected in the Response Descriptor.

CP bit (Immediate Data Transfer Command Present)

Indicates whether CMD field is valid for CCC Transfer.

DEV_INDEX[4:0] bits (Immediate Data Transfer Device Index)

Indicates the DATBASm Table index for the Slave Device being addressed with the transfer. Static and Device addressing related information will be stored to this index in the DATBASm.

BYTE_CNT[2:0] bits (Immediate Data Transfer Byte Count)

Number of valid data bytes to use in this Immediate Data Transfer Descriptor.

This field must be set to non-zero value, except for CCCs that does not have payload defined.

MODE[2:0] bits (Immediate Data Transfer Mode and Speed Values)

Sets the mode and speed for the I3C or I^2C transfer.

Interpretation of this field depends on whether the Device is in I3C Mode vs. I^2C Mode (see the DEVICE field in the DATBASm Table entry indexed by field DEV_INDEX).

RNW bit (Immediate Data Transfer R/W)

Identifies direction of the transfer.

This field shall always be set to 0, because Immediate transfers are valid for Write transactions only.

ROC bit (Immediate Data Transfer Response on Completion)

Controls whether Response Status is required after successful completion of the data transfer command. The successful completion shall be read from NRSPQP register. Upon unsuccessful transfer the Response Status shall always be sent.

TOC bit (Immediate Data Transfer Terminate on Completion)

Controls what Bus condition is issued after completion of the data transfer.

When sending Direct CCC by TOC = 0 (RESTART), next command must be set to same Direct CCC.

When the next command is not the same Direct CCC, must be set to TOC = 1 (STOP).

(3) Regular Transfer Command

This structure does not contain data to be transferred.

For Master Mode, the data buffer is available through Transfer Data Queue Port (Receive Data Queue Port and Transmit Data Queue Port).

When transmitting data of 5 bytes or more, use this Regular Transfer Command to communicate.

When transmitting data of 4 bytes or less, use the Immediate Transfer Command to communicate.

For the Regular Transfer Command, refer to **Section 26.3.1.1(2), Immediate Transfer Command**.

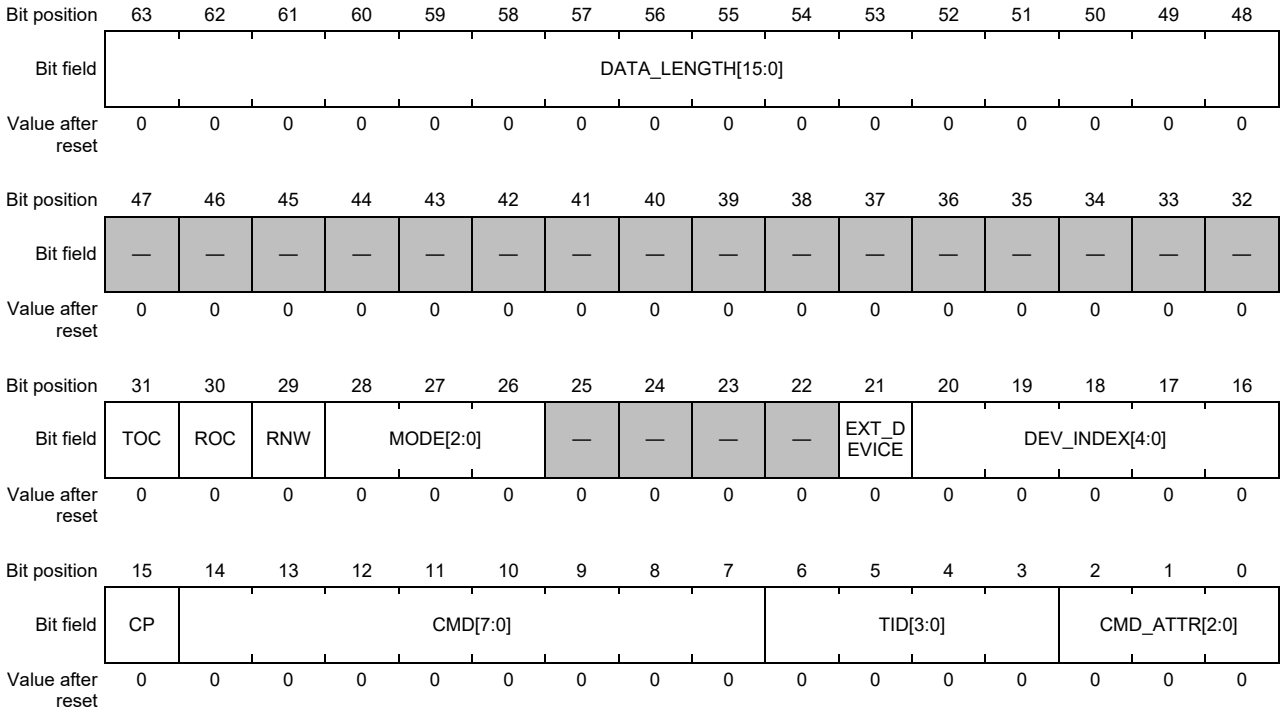
For I3C Slave Mode, the IBI Payload buffer is available through IBI Status Queue Port.

I3C provides a Regular Transfer Command for each mode below.

- I3C Master Mode
- I3C Slave Mode

Details of the regular transfer command structure of each mode are shown below.

(a) I3C Master Mode



Bit	Bit Name	R/W	Description
63 to 48	DATA_LENGTH[15:0]	W	Data Transfer Data Length Indicates the number of bytes to be transferred. This field must be set to non-zero value, except for CCCs that does not have payload defined.
47 to 32	—	W	The write value should be 0.
31	TOC	W	Data Transfer Terminate on Completion 0: RESTART: Issue Repeated START (Sr) at end of transfer 1: STOP: Issue STOP (P) at end of transfer

Bit	Bit Name	R/W	Description
30	ROC	W	Data Transfer Response on Completion 0: NOT_REQUIRED: Response Status is not required. 1: REQUIRED: Response Status is required.
29	RNW	W	Data Transfer R/W 0: WRITE: Write transfer 1: READ: Read transfer
28 to 26	MODE[2:0]	W	Data Transfer Speed and Mode Values for I3C Mode: 000: I3C SDR0 Data rate: STDBR (e.g., 12.5 MHz) 001: I3C SDR1 Data rate: EXTBR (e.g., 8 MHz) 010: I3C SDR2 Data rate: STDBR × 2 (e.g., 6.25 MHz) 011: I3C SDR3 Data rate: EXTBR × 2 (e.g., 4 MHz) 100: I3C SDR4 Data rate: EXTBR × 4 (e.g., 2 MHz) Others: Setting prohibited Values for I^2C Mode: 000: Legacy I^2C Message 0 Data rate: STDBR (e.g., 400 kHz) 001: Legacy I^2C Message 0 Data rate: EXTBR (e.g., 1 MHz) Others: Setting prohibited
25 to 22	—	W	The write value should be 0.
21	EXT_DEVICE	W	Data Transfer Extended Device Index 0: Use the DATBASm Table indicated by DEV_INDEX[4:0]. 1: Use the EXDATBAS table.
20 to 16	DEV_INDEX[4:0]	W	Data Transfer Device Index
15	CP	W	Data Transfer Command Present 0: TRANFSER: This structure describes an SDR transfer, so the CMD field is not valid. 1: CCC: This structure describes a CCC transfer, so the CMD field is valid.
14 to 7	CMD[7:0]	W	Data Transfer CCC Code Value Specifies the I3C Command code. For CCC: 8 bits
6 to 3	TID[3:0]	W	Data Transfer Transaction ID Identification tag for this command
2 to 0	CMD_ATTR[2:0]	W	Data Transfer Command Attribute Command Type, defining the format of the other fields. Values: 000: XFER: Regular Transfer 001: IMMED_DATA_XFER: Immediate Data Transfer 010: ADDR_ASSGN_CMD: Address Assignment Command 011: WWR_COMBO_XFER: Write + Write/Read Combo Transfer 111: INTERNAL_CONTROL: Internal Control command Others: Setting prohibited

CMD_ATTR[2:0] bits (Data Transfer Command Attribute)

Command Type, defining the format of the other fields.

TID[3:0] bits (Data Transfer Transaction ID Identification tag for this command)

Used as an identification tag for this command. This field shall be populated by the software Driver, and the same value shall be reflected in the Response Descriptor.

CP bit (Data Transfer Command Present)

Indicates whether he contents of the CMD field is valid for a CCC Transfer.

DEV_INDEX[4:0] bits (Data Transfer Device Index)

Indicates the DATBASm Table index for the Slave Device being addressed with the transfer. Static and Device addressing related information will be stored to this index in the DATBASm.

MODE[2:0] bits (Data Transfer Speed and Mode)

Sets the mode and speed for the I3C or I^2C transfer.

Interpretation of this field depends on whether the Device is in I3C Mode vs. I^2C Mode (see the DEVICE field in the DATBASm Table entry indexed by field DEV_INDEX).

RNW bit (Data Transfer R/W)

Identifies direction of the transfer.

ROC bit (Data Transfer Response on Completion)

Controls whether Response Status is required after successful completion of the transfer command. The successful completion shall be read from NRSPQP register. Upon unsuccessful transfer the Response Status shall always be sent.

TOC bit (Data Transfer Terminate on Completion)

Controls what Bus condition will be issued after completion of the transfer.

When sending Direct CCC by TOC = 0 (RESTART), next command must be set to same Direct CCC.

When the next command is not the same Direct CCC b , must be set to TOC = 1 (STOP).

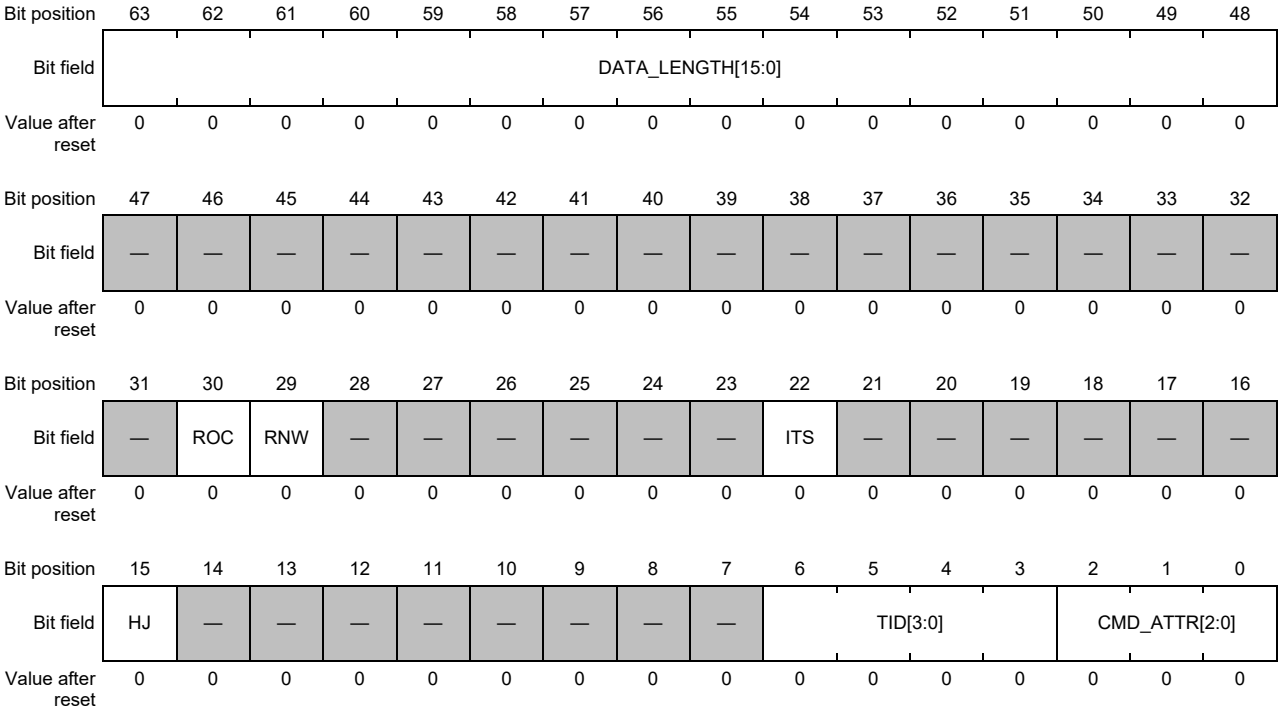
DATA_LENGTH[15:0] bits (Data Transfer Data Length)

Number of valid data bytes to use in this Regular Transfer Descriptor.

This field must be set to non-zero value, except for CCCs that does not have payload defined.

Length setting of GETMXDS command should be fixed to 5.

(b) I3C Slave Mode



Bit	Bit Name	R/W	Description
63 to 48	DATA_LENGTH[15:0]	W	Data Transfer Data Length Indicates the number of bytes to be transferred. This field must be set to non-zero value, except for CCCs that does not have payload defined.
47 to 31	—	W	The write value should be 0.
30	ROC	W	Data Transfer Response on Completion 0: NOT_REQUIRED: Response Status is not required. 1: REQUIRED: Response Status is required.
29	RNW	W	Data Transfer R/W 0: WRITE: Write transfer (Mastership Request) 1: READ: Read transfer (Slave Interrupt Request)
28 to 23	—	W	The write value should be 0.
22	ITS	W	Include timestamp for Async Mode 0: Do not include timestamp. 1: Include timestamp.
21 to 16	—	W	The write value should be 0.
15	HJ	W	Data Transfer Hot-Join Event 0: Slave Interrupt Request or Mastership Request, so the RNW field is valid. 1: Hot-Join Event, so the RNW field is not valid.
14 to 7	—	W	The write value should be 0.
6 to 3	TID[3:0]	W	Data Transfer Transaction ID Identification tag for this command
2 to 0	CMD_ATTR[2:0]	W	Data Transfer Command Attribute Command Type, defining the format of the other fields. Values: 000: XFER: Regular Transfer 001: IMMED_DATA_XFER: Immediate Data Transfer 010: ADDR_ASSGN_CMD: Address Assignment Command 011: WWR_COMBO_XFER: Write + Write/Read Combo Transfer 100 to 110: Reserved, do not use 111: INTERNAL_CONTROL: Internal Control command

CMD_ATTR[2:0] bits (Data Transfer Command Attribute)

Command Type, defining the format of the other fields.

TID[3:0] bits (Data Transfer Transaction ID Identification tag for this command)

Used as an identification tag for this command. This field shall be populated by the software Driver, and the same value shall be reflected in the Response Descriptor.

HJ bit (Data Transfer Hot-Join Event)

Indicates whether Hot-Join Event is valid in this IBI Data transfer.

RNW bit (Data Transfer R/W)

Identifies direction of the transfer.

ROC bit (Data Transfer Response on Completion)

Controls whether Response Status is required after successful completion of the transfer command. The successful completion shall be read from NRSPQP register. Upon unsuccessful transfer the Response Status shall always be sent.

(4) Combo Transfer Command

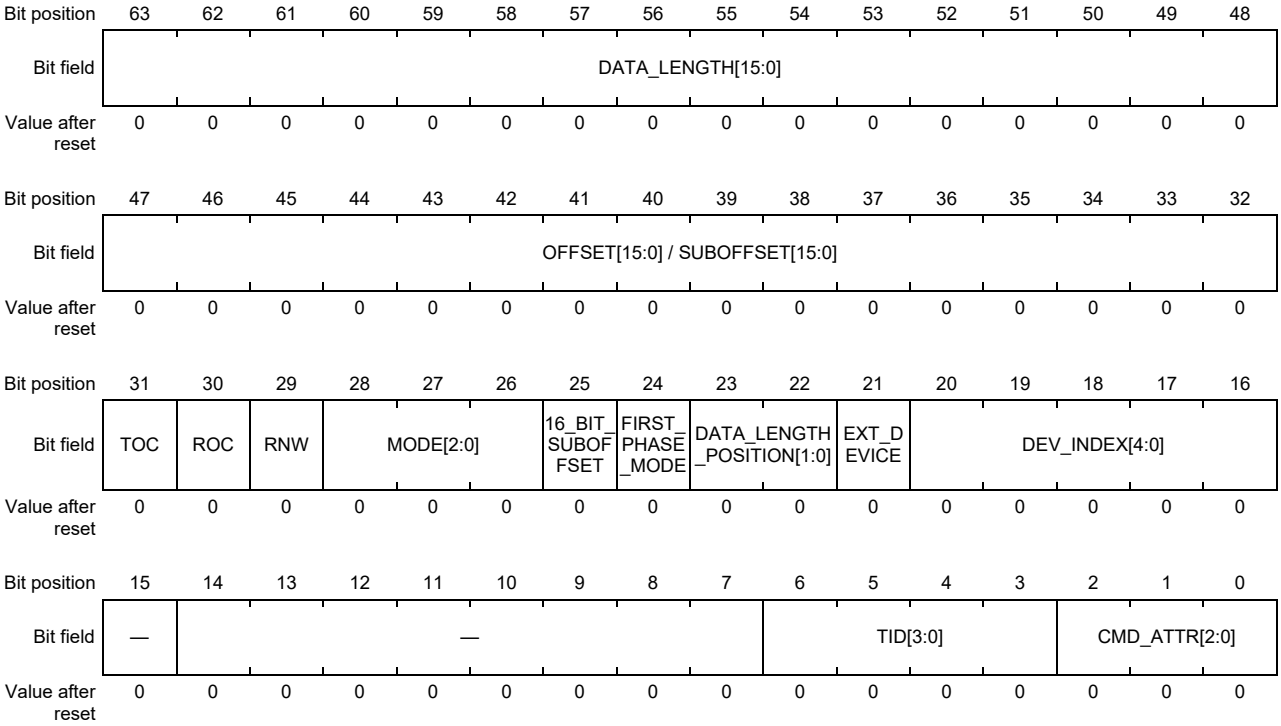
This structure contains a combined Write + Read/Write operation.

The data buffer is available through Transfer Data Queue Port (Receive Data Queue Port and Transmit Data Queue Port).

I3C provides a Combo Transfer Command for the following mode:

- I3C Master Mode

Details of the Combo Transfer Command Structure of each mode are as follows.



Bit	Bit Name	R/W	Description
63 to 48	DATA_LENGTH[15:0]	W	Combo Transfer Data Length Number of bytes to be transferred. This field must be set to non-zero value.
47 to 32	OFFSET[15:0]/ SUBOFFSET[15:0]	W	Combo Transfer Offset/Sub-Offset Offset of the target operation
31	TOC	W	Combo Transfer Terminate on Completion 0: RESTART: Issue Repeated START (Sr) at end of transfer 1: STOP: Issue STOP (P) at end of transfer
30	ROC	W	Combo Transfer Response on Completion 0: NOT_REQUIRED: Response Status is not required. 1: REQUIRED: Response Status is required.
29	RNW	W	Combo Transfer R/W Identifies direction of the transfer. 0: WRITE: Write transfer 1: READ: Read transfer

Bit	Bit Name	R/W	Description
28 to 26	MODE[2:0]	W	Combo Transfer Speed and Mode Values for I3C Mode: 000: I3C SDR0 Data rate: STDBR (e.g., 12.5 MHz) 001: I3C SDR1 Data rate: EXTBR (e.g., 8 MHz) 010: I3C SDR2 Data rate: STDBR × 2 (e.g., 6.25 MHz) 011: I3C SDR3 Data rate: EXTBR × 2 (e.g., 4 MHz) 100: I3C SDR4 Data rate: EXTBR × 4 (e.g., 2 MHz) Others: Setting prohibited
25	16_BIT_SUBOFFSET	W	Combo Transfer Sub Offset Size 0: 8_BIT_SUBOFFSET: Sub-offset is 8-bits long. Value is encoded in Lower Byte of OFFSET/SUBOFFSET field. 1: 16_BIT_SUBOFFSET: Sub-offset is 16-bits long.
24	FIRST_PHASE_MODE	W	Combo Transfer First Phase Mode 0: SDR: First phase is executed in SDR mode. 1: MODE: First phase is executed in the mode indicated by the MODE field.
23, 22	DATA_LENGTH_POSITION[1:0]	W	Data Length Field Position 0 0: NO: Do not put length field. 0 1: FIRST: Put length as first field. 1 0: SECOND: Put length as second field. Others: Setting prohibited
21	EXT_DEVICE	W	Combo Transfer Extended Device Index 0: Use the DAT table indicated by DEV_INDEX[4:0]. 1: Use the EXDATBAS table.
20 to 16	DEV_INDEX[4:0]	W	Combo Transfer Device Index Indicates the DAT table index for the Slave device being addressed with the transfer. Static and device addressing related information are stored to this index in the DAT.
15 to 7	—	W	The write value should be 0.
6 to 3	TID[3:0]	W	Combo Transfer Transaction ID Identification tag for the command
2 to 0	CMD_ATTR[2:0]	W	Combo Transfer Command Attribute Command Type, defining the format of the other fields. 000: XFER: Regular Transfer 001: IMMED_DATA_XFER: Immediate Data Transfer 010: ADDR_ASSGN_CMD: Address Assignment Command 011: WWR_COMBO_XFER: Write + Write/Read Combo Transfer 111: INTERNAL_CONTROL: Internal Control command Others: Setting prohibited

CMD_ATTR[2:0] bits (Combo Transfer Command Attribute)

Command Type, defining the format of the other fields.

TID[3:0] bits (Combo Transfer Transaction ID Identification tag for the command)

Used as an identification tag for this command. This field shall be populated by the software Driver, and the same value shall be reflected in the Response Descriptor.

DEV_INDEX[4:0] bits (Combo Transfer Device Index)

Indicates the DAT table index for the Slave Device being addressed with the transfer. Static and Device addressing related information will be stored to this index in the DAT.

DATA_LENGTH_POSITION[1:0] bits (Data Length Field Position)

Indicates whether and where to put Data Length (DATA_LENGTH) in the first phase of the transfer.

Whether 8-bit or 16-bit of Data Length field is used is indicated with 16_BIT_SUBOFFSET field. In case of 8-bit value, it is encoded in Lower Byte of DATA_LENGTH field.

FIRST_PHASE_MODE bits (Combo Transfer First Phase Mode)

Indicates whether the first phase of the Combo Transfer is executed in SDR Mode, vs. the Mode indicated by the MODE field.

MODE[2:0] bits (Combo Transfer Speed and Mode Values for I3C Mode)

Sets the mode and speed for the I3C or I^2C transfer.

Interpretation of this field depends on whether the Device is in I3C Mode vs. I^2C Mode (see the DEVICE field in the DAT Table entry indexed by field DEV_INDEX).

RNW bit (Combo Transfer R/W Identifies direction of the transfer)

Identifies direction of the transfer.

ROC bit (Combo Transfer Response on Completion)

Controls whether Response Status is required after successful completion of the data transfer command. The successful completion shall be read from NRSPQP register. Upon unsuccessful transfer the Response Status shall always be sent.

TOC bit (Combo Transfer Terminate on Completion)

Controls what Bus condition is issued after completion of the data transfer.

When the next command is SDR mode, must be set to TOC = 1 (STOP).

DATA_LENGTH[15:0] bit (Combo Transfer Data Length Number of bytes to be transferred. This field must be set to non-zero value.)

Number of valid data bytes to use in this Combo Transfer Descriptor.

This field must be set to non-zero value.

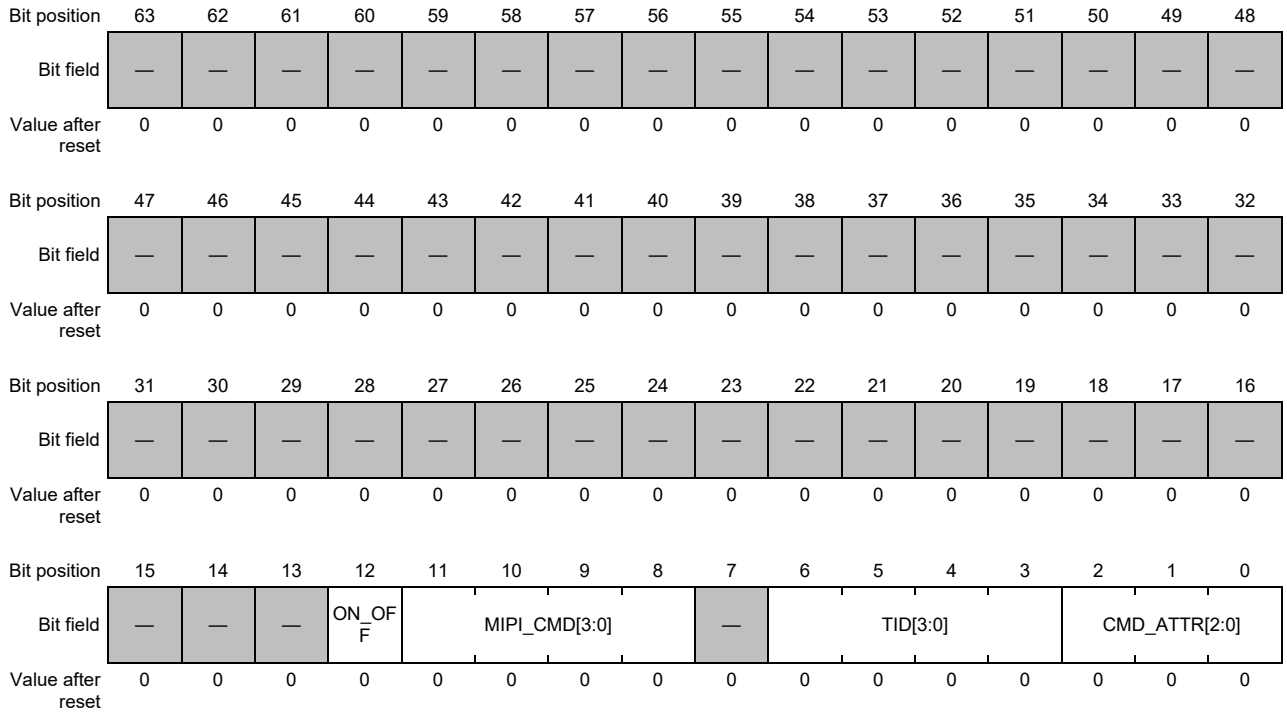
(5) Internal Control Command

This structure is used for controlling I3C itself (not for transfer commands).

I3C provides an Internal Control Command for the following mode:

- I3C Master Mode

Details of the Internal Control Command Structure are as follows:



Bit	Bit Name	R/W	Description
63 to 13	—	W	The write value should be 0.
12	ON_OFF	W	Bus Instance 7E On/Off*1 Enables or disables automatic transmission of the I3C Broadcast Header after every START condition on this I3C Bus instance. 0: IBA_INCLUDE off 1: IBA_INCLUDE on
11 to 8	MIPI_CMD[3:0]	W	MIPI Alliance Command H'0: NoOp, so the ON_OFF field is not valid. H'2: Include 7E (IBA), so the ON_OFF field is valid. Others: Setting prohibited
7	—	W	The write value should be 0.
6 to 3	TID[3:0]	W	Transaction ID Identification tag for the command
2 to 0	CMD_ATTR[2:0]	W	Command Attribute*2 Command Type, defining the format of the other fields. 000: XFER: Regular Transfer 001: IMMED_DATA_XFER: Immediate Data Transfer 010: ADDR_ASSGN_CMD: Address Assignment Command 011: WWR_COMBO_XFER: Write + Write/Read Combo Transfer 111: INTERNAL_CONTROL: Internal Control command Others: Setting prohibited

Note 1. The IBA_INCLUDE on state set by MIPI_CMD[3:0] = 0010b and ON_OFF = 1 is cleared by setting RSTCTL.INTLRST to 1.

Note 2. The Response descriptor is not stored when the Internal Control Command is executed.

26.3.1.2 Response Descriptor

The Response Descriptor is a read-only structure describing the success or failure of a command, and the amount of data transferred.

The Response Descriptor is read from Response Queue with reads from Response Queue Port.

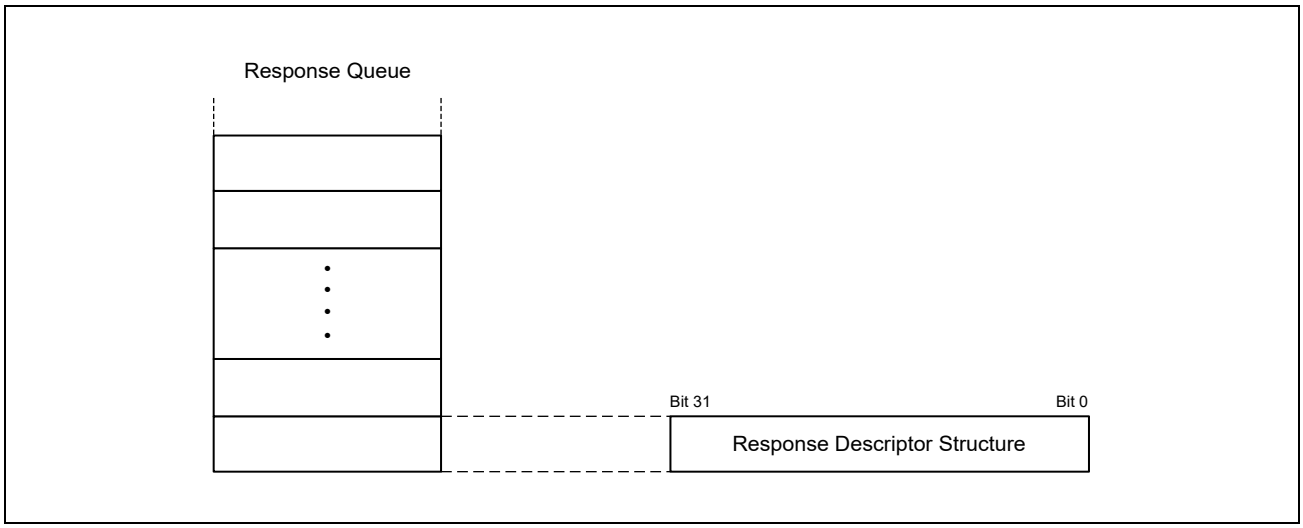


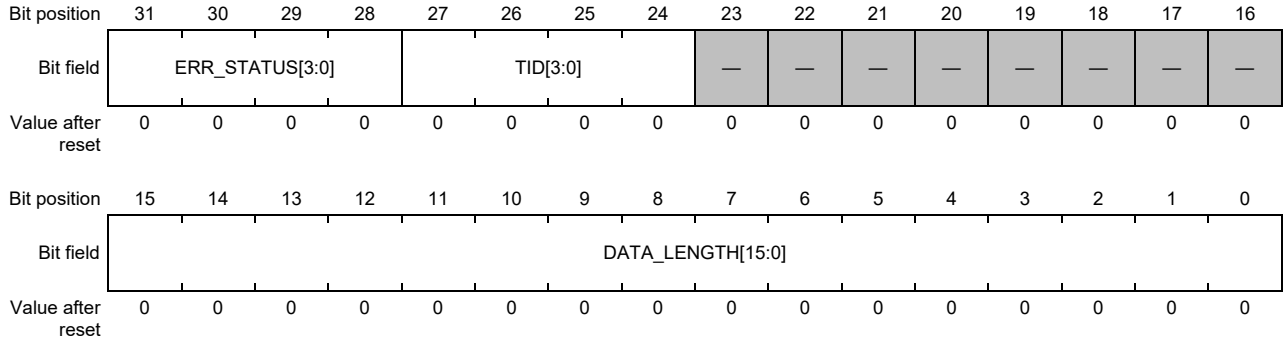
Figure 26.3 Response Descriptor Data Structure

I3C provides a Response Descriptor for the following modes:

- I3C Master Mode
- I3C Slave Mode

Details of the Response Descriptor structure of each mode are shown in the following sections.

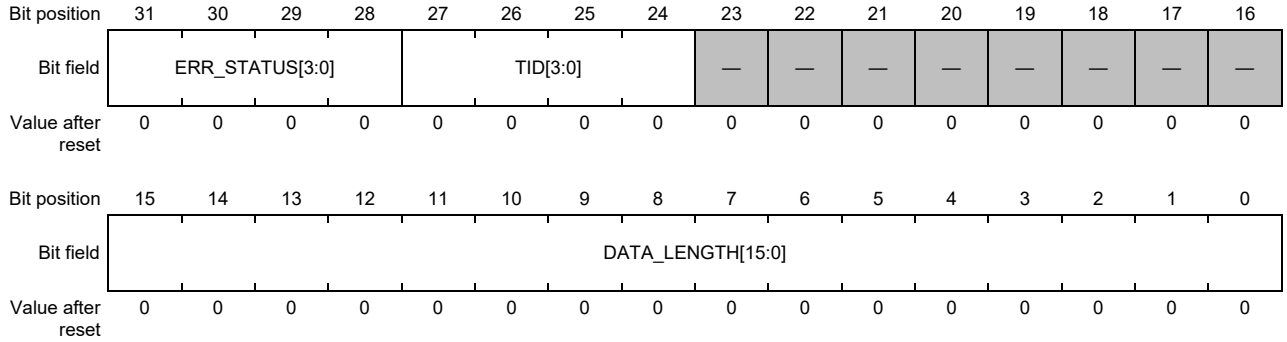
(a) I3C Master Mode



Bit	Bit Name	R/W	Description
31 to 28	ERR_STATUS[3:0]	R	MIPI Alliance Command H'0: SUCCESS: Transfer successful, no error H'1: CRC: CRC Error H'2: PARITY: Parity Error H'3: FRAME: Frame Error H'4: ADDR_HEADER: Address Header Error H'5: NACK: Address NACKed or Dynamic Address Assignment NACKed H'6: OVL: Receive Overflow or Transfer Underflow Error H'8: ABORTED: Aborted H'9: I^2C_WR_DATA_NACK: NACK received for the I^2C Write Data transfer H'A: NOT_SUPPORTED: Command with specific parameters not supported by I3C implementation (for example, specific Internal Control codes may not be supported) Others: Setting prohibited
27 to 24	TID[3:0]	R	Command/Response Transaction ID Identification tag for the command. This value shall match one of commands sent on the Bus. H'0 to H'7: Valid Transaction IDs Others: Setting prohibited
23 to 16	—	R	These bits are read as 0.
15 to 0	DATA_LENGTH[15:0]	R	Data Length/Device Count The meaning of this field depends on the context: For Write Transfer: Remaining data length (in bytes) For Read Transfer: Received data length (in bytes) For Address Assignment: Remaining Device count

Note: In I3C Master mode, when an abnormal command with a specific parameter that is not supported is stored in Command Descriptor, it is indicated as NOT_SUPPORTED (H'A) in ERR_STATUS[3:0].

(b) I3C Slave Mode



Bit	Bit Name	R/W	Description
31 to 28	ERR_STATUS[3:0]	R	Response Error Status H'0: SUCCESS: Transfer successful, no error. H'3: FRAME: Frame Error H'4: ADDR_HEADER: Address Header Error H'5: NACK: Address NACK'ed or Dynamic Address Assignment NACK'ed H'6: OVL: Receive Overflow or Transfer Underflow Error H'8: ABORTED: Aborted H'A: NOT_SUPPORTED: Command with specific parameters not supported by I3C implementation (for example, specific Internal Control codes may not be supported) Others: Setting prohibited
27 to 24	TID[3:0]	R	Command/Response Transaction ID Identification tag for the command. This value matches one of commands sent on the bus. H'0 to H'7: Valid Transaction IDs Others: Setting prohibited
23 to 16	—	R	These bits are read as 0.
15 to 0	DATA_LENGTH[15:0]	R	Data Length Remaining data length (in bytes) for Slave Interrupt Request

Note: In I3C Slave mode, it is indicated as NOT_SUPPORTED (H'A) in ERR_STATUS[3:0] in the following cases:
- When an abnormal command with a specific parameter that is not supported is stored in the Command Descriptor.
- When the IBI to be transmitted is disabled in the CSECMD register.
- After the normal command for IBI transmission is prepared in the Command Queue, when that IBI is disabled in the CSECMD register by the DISEC CCC frame from the I3C Master.

26.3.1.3 IBI Status Descriptor

The IBI Status Descriptor is a read-only structure describing an IBI event received from a Slave device on the I3C Bus.

The IBI Status Descriptor is read from IBI Status Queue with reads from IBI Status Queue Port.

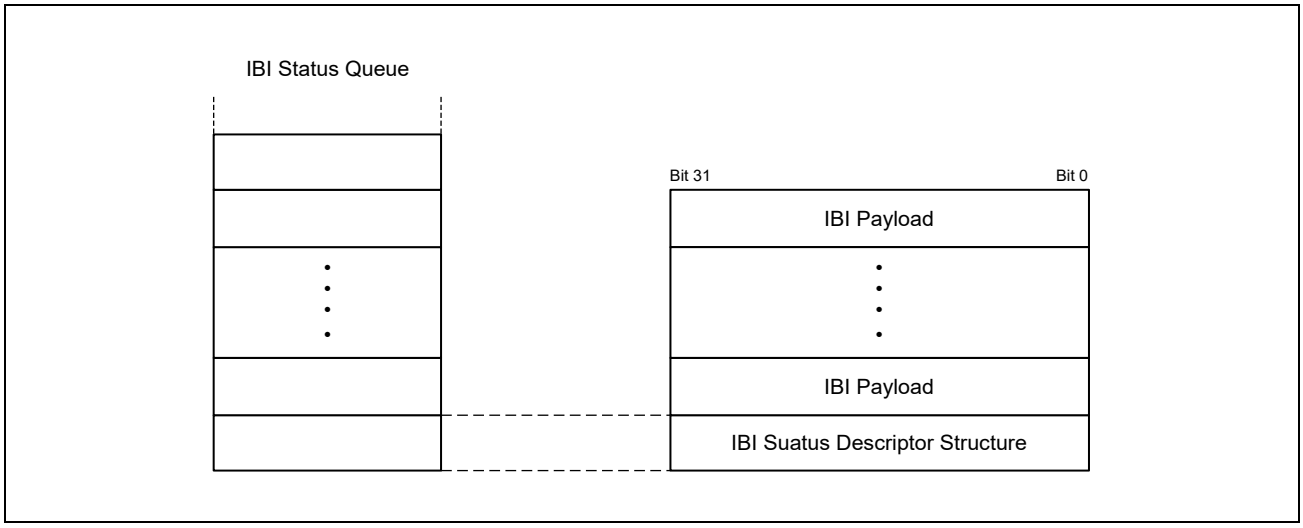


Figure 26.4 IBI Status Descriptor Data Structure

I3C provides a IBI Status Descriptor for the following mode:

- I3C Master Mode

Details of the IBI Status Descriptor Structure are as follows.

Bit position	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field	IBI_ST	—	—	ERR_STATUS[2:0]			TS	LAST_STATUS	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field	IBI_ID[7:0]								DATA_LENGTH[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Bit Name	R/W	Description
31	IBI_ST	R	IBI Received Status Indicates how the received IBI was handled. 0: The IBI was handled with ACK. 1: NACK: The IBI was handled with NACK, and then Auto-Disabled.
30, 29	—	R	These bits are read as 0. The write value should be 0.
28:26	ERR_STATUS[2:0]	R	IBI Error Status 000: SUCCESS 011: ERROR: FRAME (Frame Error) 100: ERROR: ADDR_HEADER (Address Header Error) 101: NACK: Address NACKed 111: ERROR: ABORT (Aborted to Master) Others: Setting prohibited

Bit	Bit Name	R/W	Description
25	TS	R	IBI Time-stamp Present Indicates whether a timestamp is available for the IBI. 0: OFF: IBI is not time-stamped. 1: ON: IBI is time-stamped.
24	LAST_STATUS	R	Last IBI Status Last IBI status for the IBI transaction.
23:16	—	R	These bits are read as 0. The write value should be 0.
15:8	IBI_ID[7:0]	R	IBI Received ID The meaning of this field depends on the context: For Slave Interrupt or Master Request: Bits 15:9 contain the Slave's Device Address, and bit 8 contains the R/W bit. For Hot-Join IBI: Bits 15:8 contain the Hot-Join ID for the IBI.
7:0	DATA_LENGTH[7:0]	R	IBI Data Length Number of data bytes in IBI Data.

LAST_STATUS bits (Last IBI Status)

Even if LAST_STATUS is set to 0, the software driver still evaluates the data payload length by examining the CHUNKS field.

26.3.1.4 Receive Status Descriptor

The Receive Status Descriptor is a read-only structure describing the success or failure of read/write operation from the master, and the amount of data transferred.

The Receive Status Descriptor is read from Receive Status Queue with reads from Receive Status Queue Port.

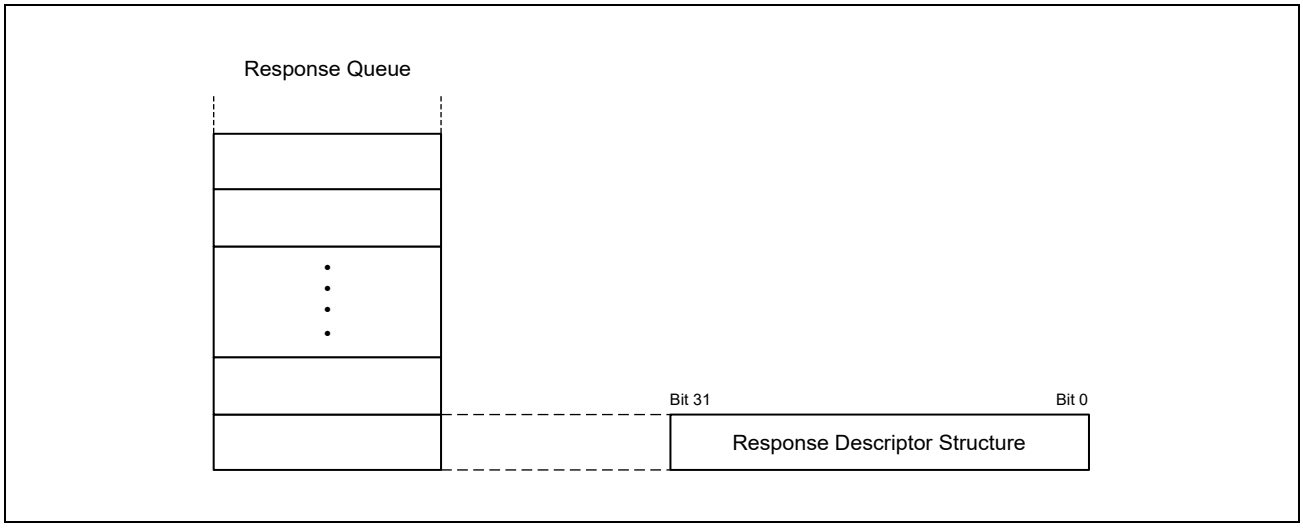


Figure 26.5 Receive Status Descriptor Data Structure

I3C provides a Receive Status Descriptor for the following mode:

- I3C Slave Mode

Details of the Receive Status Descriptor structure of each mode are as follows.

Bit position	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field	DEV_INDEX[2:0]			TRANSFER_TYPE[1:0]		ERR_STATUS[2:0]			CMD[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field	DATA_LENGTH[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Bit Name	R/W	Description
31 to 29	DEV_INDEX[2:0]	R	Device Index Indicates the SVDVADn index for the response with the transfer.
28, 27	TRANSFER_TYPE [1:0]	R	Transfer Type 00: I3C SDR/I^{2}C Message 01: I3C CCC 10: Setting prohibited 11: Setting prohibited

Bit	Bit Name	R/W	Description
26 to 24	ERR_STATUS[2:0]	R	Error Status 000: SUCCESS 001: ERROR: CRC (CRC Error) 010: ERROR: PARITY (Parity Error) 011: ERROR: FRAME (Frame Error) 100: ERROR: ADDR_HEADER (Address Header Error) 101: ERROR: NACK (Slave NACKed) 110: ERROR: OVL (FIFO Overflow/Underflow) 111: ERROR: ABORT (Aborted to Master)
23 to 16	CMD[7:0]	R	The contents are different depending on the operation mode. Details are are follows: [SDR Private Message Mode] Bit 23: R/W Type Bits 22 to 20: Setting prohibited Bit 19: I3C_I^2C Type Bits 18 to 16: Setting prohibited [SDR CCC Mode] CCC code[7:0]
15 to 0	DATA_LENGTH[15:0]	R	Data Length The meaning of this field depends on the context. For Write Transfer: Received data length (in bytes) For Read Transfer: Transmitted data length (in bytes)

26.3.2 Details of Function

26.3.2.1 Operation Mode

The support relationship between the mode selects (I3C mode/I²C mode) and operation mode (Master/Slave) on the I3C bus or the I²C bus is shown in **Table 26.10**.

Table 26.10 Support of Operating Mode

I3C/I²C Bus	I3C Mode		I²C Mode	
	Master	Slave	Master	Slave
I3C Bus	✓	✓	—	✓
I²C Bus	—	—	✓	✓

Note: ✓: Supported
— : Un-Supported

(1) Master Mode Operation

(a) I²C Master Operation

1) Data Write Transfer (Single Buffer transfer)

In master transmit operation, I3C outputs the SCL clock and transmitted data signals as the master device, and the slave device returns acknowledgments. **Figure 26.6** shows an example of usage of master transmission and **Figure 26.6** to **Figure 26.8** show the timing of operations in master transmission.

The following describes the procedure and operations for master transmission.

1. Initial settings. For details, refer to **Section 26.3.3.1, Initial Setting Flow**.
2. Read the BCST.BFREF flag to check that the bus is open, and then set the CNDCTL.STCND bit to 1 (START condition issuance request). Upon receiving the request, I3C issues a START condition. At the same time, the BFREF flag bit is automatically set to 0, the BST.STCNDDF flag is automatically set to 1 and the STCND bit is automatically set to 0. At this time, if the START condition is detected and the internal levels for the SDA output state and the levels on the SDAn line have matched while the STCND bit = 1, I3C recognizes that issuing of the START condition as requested by the STCND bit has been successfully completed, and bits CRMS and TRMD in the PRSST register are automatically set to 1, placing I3C in master transmit mode. The NTST.TDBEF0 flag is also automatically set to 1 in response to setting of the TRMD bit to 1.
3. Check that the NTST.TDBEF0 flag = 1, and then write the value for transmission (the slave address and the R/W# bit) to the NTDTBP0 register. Once the data for transmission are written to the NTDTBP0 register, the TDBEF0 flag is automatically set to 0, the data are transferred from the Normal Transmit Data Buffer 0 to the Shift Register, and the TDBEF0 flag is again set to 1. After the byte containing the slave address and R/W# bit has been transmitted, the value of the TRMD bit is automatically updated to select master transmit or master receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 0, I3C continues in master transmit mode. Because the BST.NACKDF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the CNDCTL.SPCND bit to issue a STOP condition. For data transmission with an address in the 10-bit format, start by writing 1_1110, the 2 higher-order bits of the slave address, and W to the NTDTBP0 register as the first address transmission. Then, as the second address transmission, write the 8 lower-order bits of the slave address to the NTDTBP0 register.
4. After confirming that the NTST.TDBEF0 flag = 1, write the data for transmission to the NTDTBP0 register. I3C automatically holds the SCLn line low until the data for transmission are ready or a STOP condition is issued.

5. After all bytes of data for transmission have been written to the NTDTBP0 register, wait until the value of the BST.TENDF flag returns to 1, and then set the CNDCTL.SPCND bit to 1 (STOP condition issuance request). Upon receiving a STOP condition issuance request, I3C issues the STOP condition.
6. Upon detecting the STOP condition, I3C automatically sets bits CRMS and TRMD in the PRSST register to 00 and enters slave receive mode. Furthermore, it automatically sets the TDBEF0 and TENDF flags to 0, and sets the BST.SPCNDDF flag to 1.
7. After checking that the BST.SPCNDDF flag = 1, set the BST.NACKDF and SPCNDDF flags to 0 for the next transfer operation.

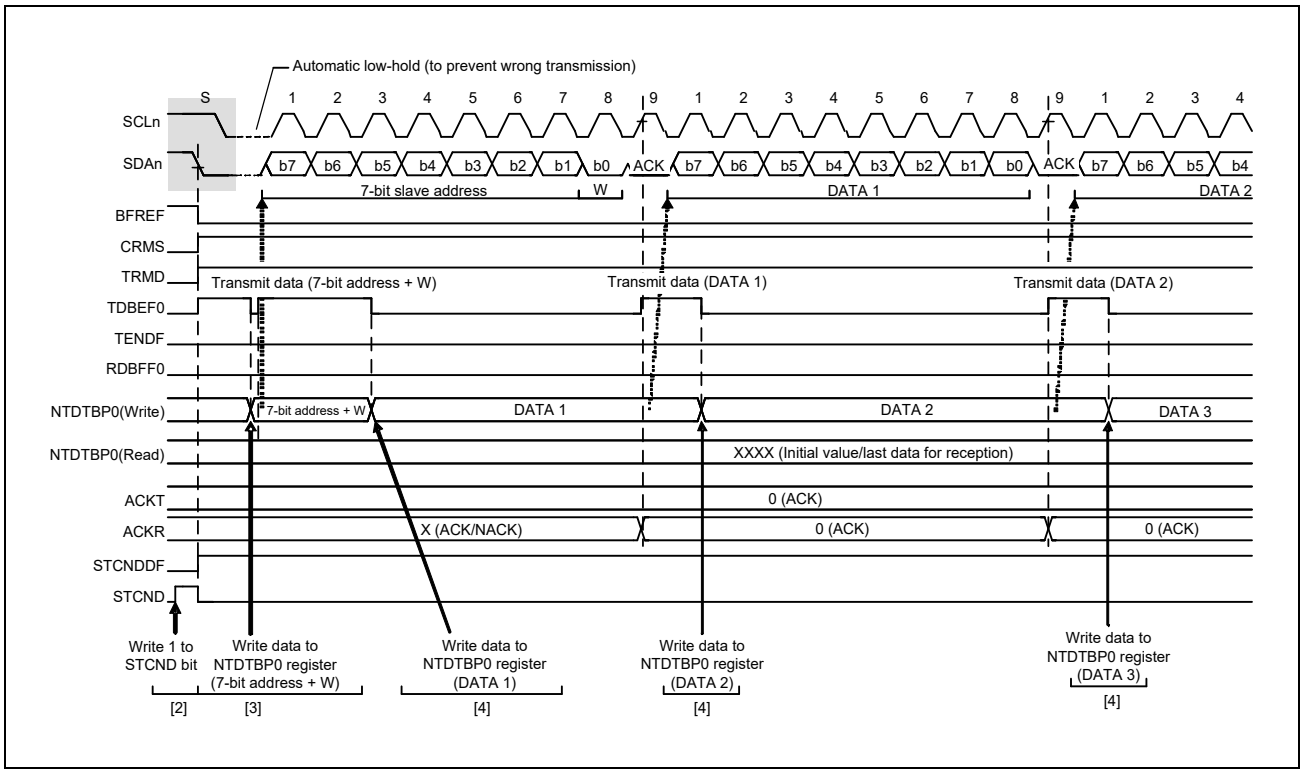


Figure 26.6 Master Transmit Operation Timing (1) (7-bit Address Format)

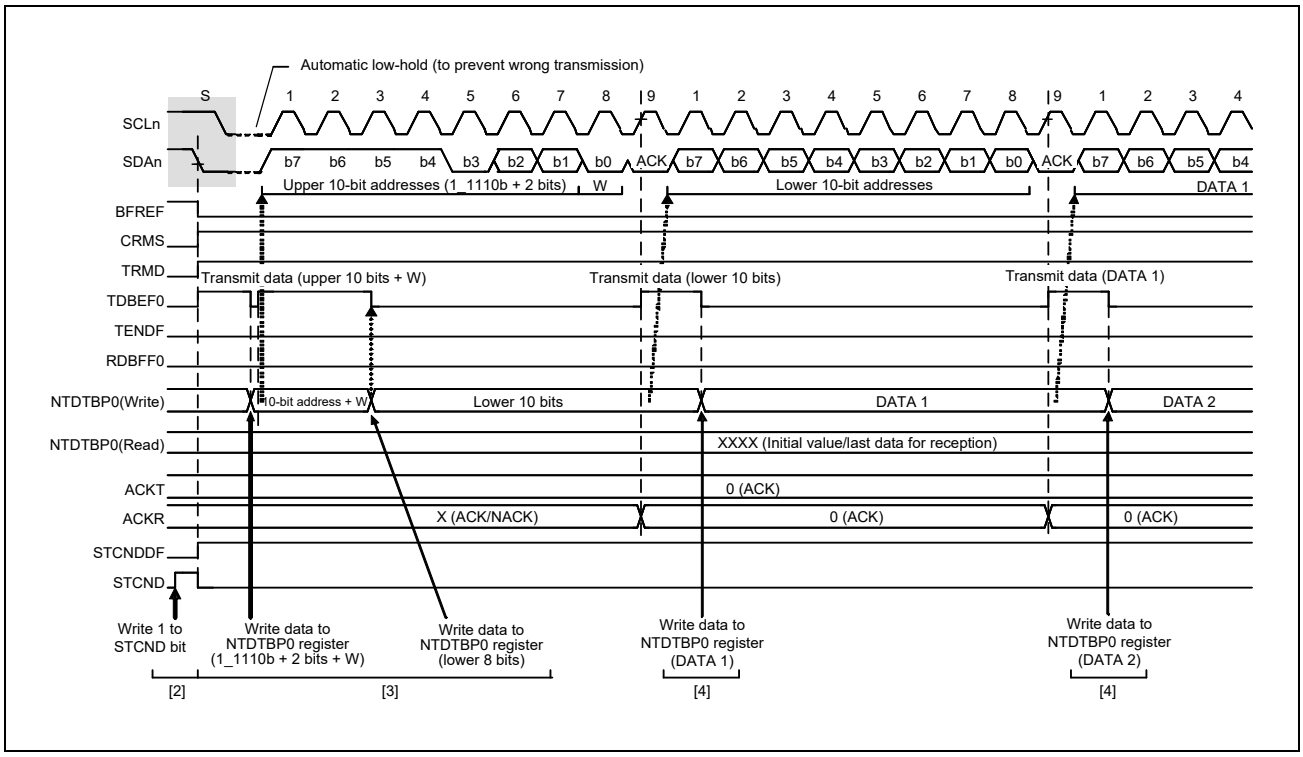


Figure 26.7 Master Transmit Operation Timing (2) (10-bit Address Format)

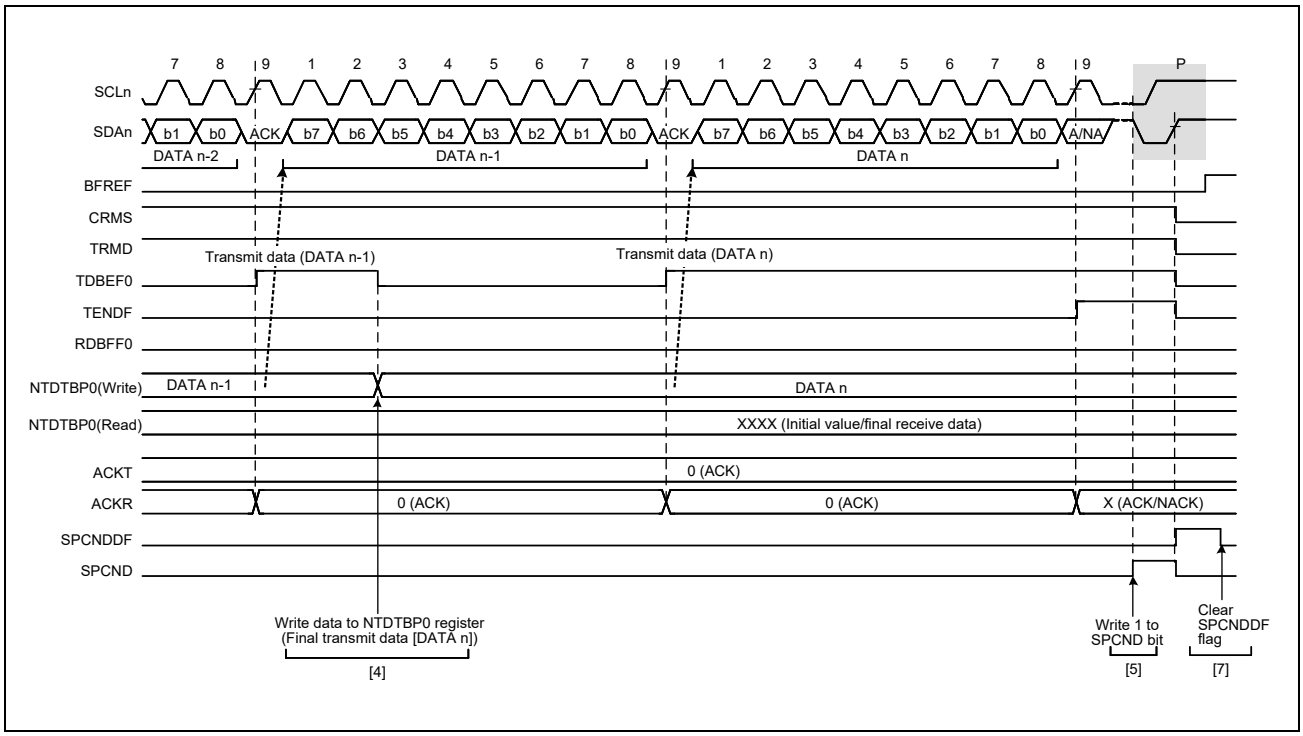


Figure 26.8 Master Transmit Operation Timing (3)

2) Data Read Transfer (Single Buffer transfer)

In master receive operation, I3C as a master device outputs the SCL clock, receives data from the slave device, and returns acknowledgments. Because I3C must start by sending a slave address to the corresponding slave device, this part of the procedure is performed in master transmit mode, but the subsequent steps are in master receive mode.

Figure 26.119 and **Figure 26.120** show examples of usage of master reception (7-bit address format) and **Figure 26.9** to **Figure 26.11** show the timing of operations in master reception.

The following describes the procedure and operations for master reception.

1. Initial settings. For details, refer to **Section 26.3.3.1, Initial Setting Flow**.
2. Read the BCST.BFREF flag to check that the bus is open, and then set the CNDCTL.STCND bit to 1 (START condition issuance request). Upon receiving the request, I3C issues a START condition. When I3C detects the START condition, the BFREF flag is automatically set to 0 and the BST.STCNDDF flag is automatically set to 1 and the STCND bit is automatically set to 0. At this time, if the START condition is detected and the levels for the SDA output and the levels on the SDAn line have matched while the STCND bit = 1, I3C recognizes that issuing of the START condition as requested by the STCND bit has been successfully completed, and bits CRMS and TRMD in the PRSST register are automatically set to 1, placing I3C in master transmit mode. The NTST.TDBEF0 flag is also automatically set to 1 in response to setting of the TRMD bit to 1.
3. Check that the NTST.TDBEF0 flag = 1, and then write the value for transmission (the first byte indicates the slave address and value of the R/W# bit) to the NTDTBP0 register. Once the data for transmission are written to the NTDTBP0 register, the TDBEF0 flag is automatically set to 0, the data are transferred from the Normal Transmit Data Buffer 0 to the Shift Register, and the TDBEF0 flag is again set to 1. Once the byte containing the slave address and R/W# bit has been transmitted, the value of the PRSST.TRMD bit is automatically updated to select transmit or receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 1, the TRMD bit is set to 0 on the rising edge of the ninth cycle of SCL clock, placing I3C in master receive mode. At this time, the TDBEF0 flag is set to 0. The NTST.RDBFF0 flag is automatically set to 1 when ACK response is received from the slave device. If the slave device is not recognized or a communication failure occurs, the BST.NACKDF flag will be set to 1. At this time, set 1 to the CNDCTL.SPCND bit to issue a STOP condition. For master reception from a device with a 10-bit address, start by using master transmission to issue the 10-bit address, and then issue a Repeated START condition. After that, transmitting 1_1110, the two higher-order bits of the slave address, and the R bit places I3C in master receive mode.
4. Dummy read the NTDTBP0 register after confirming that the NTST.RDBFF0 flag = 1; this makes I3C start output of the SCL clock and start data reception.
5. After 1 byte of data has been received, the NTST.RDBFF0 flag is set to 1 on the rising edge of the eighth or ninth cycle of SCL clock (the clock signal) as selected by the SCSTRCTL.ACKTWE bit. Reading the NTDTBP0 register at this time will produce the received data, and the RDBFF0 flag is automatically set to 0 at the same time. Furthermore, the value of the acknowledgment field received during the ninth cycle of SCL clock is returned as the value set in the ACKCTL.ACKT bit. Furthermore, if the next byte to be received is the next to last byte, set the SCSTRCTL.RWE bit to 1 (for wait insertion) before reading the NTDTBP0 register (containing the second byte from last). As well as enabling NACK output even in the case of delays in processing to set the ACKCTL.ACKT bit to 1 (NACK) in step 6, due to other interrupts, etc., this fixes the SCLn line to the low level on the falling edge of the ninth clock cycle in reception of the last byte, so the state is such that issuing a STOP condition is possible.
6. When the SCSTRCTL.ACKTWE bit = 0 and the slave device must be notified that it is to end transfer for data reception after transfer of the next (final) byte, set the ACKCTL.ACKT bit to 1 (NACK).
7. After reading the byte before last from the NTDTBP0 register, if the value of the NTST.RDBFF0 flag is confirmed to be 1, write 1 to the CNDCTL.SPCND bit (STOP condition issuance request) and then read the last

byte from the NTDTBP0 register. When 1 is written to the CNDCTL.SPCND bit, I3C is released from the wait state and issues the STOP condition after low-level output in the ninth clock cycle is completed or the SCLn line is released from the low- hold state.

8. Upon detecting the STOP condition, I3C automatically sets bits CRMS and TRMD in the PRSST register to 00 and enters slave receive mode. Furthermore, detection of the STOP condition leads to setting of the BST.SPCNDDF flag to 1.

9. After checking that the BST.SPCNDDF flag = 1, set the BST.NACKDF and SPCNDDF flags to 0 for the next transfer operation.

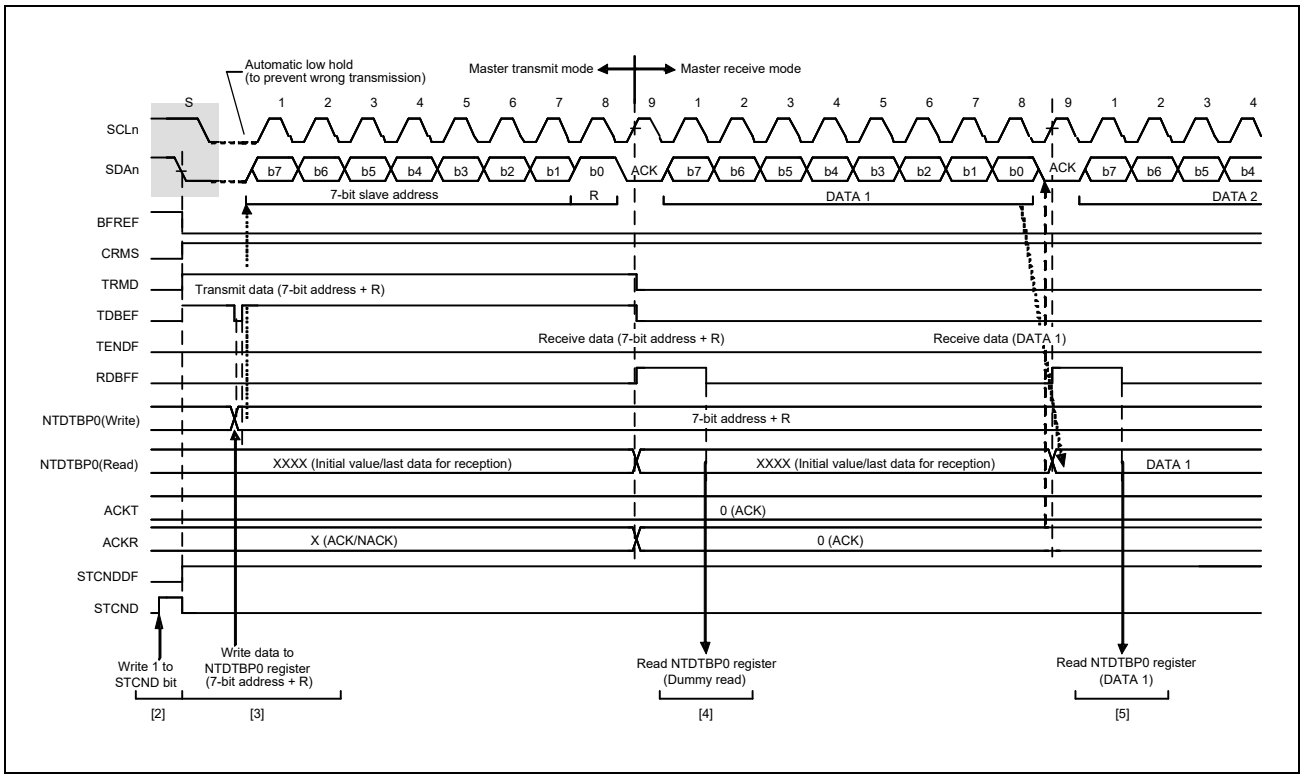


Figure 26.9 Master Receive Operation Timing (1) (7-bit Address Format, When ACKTWE = 0)

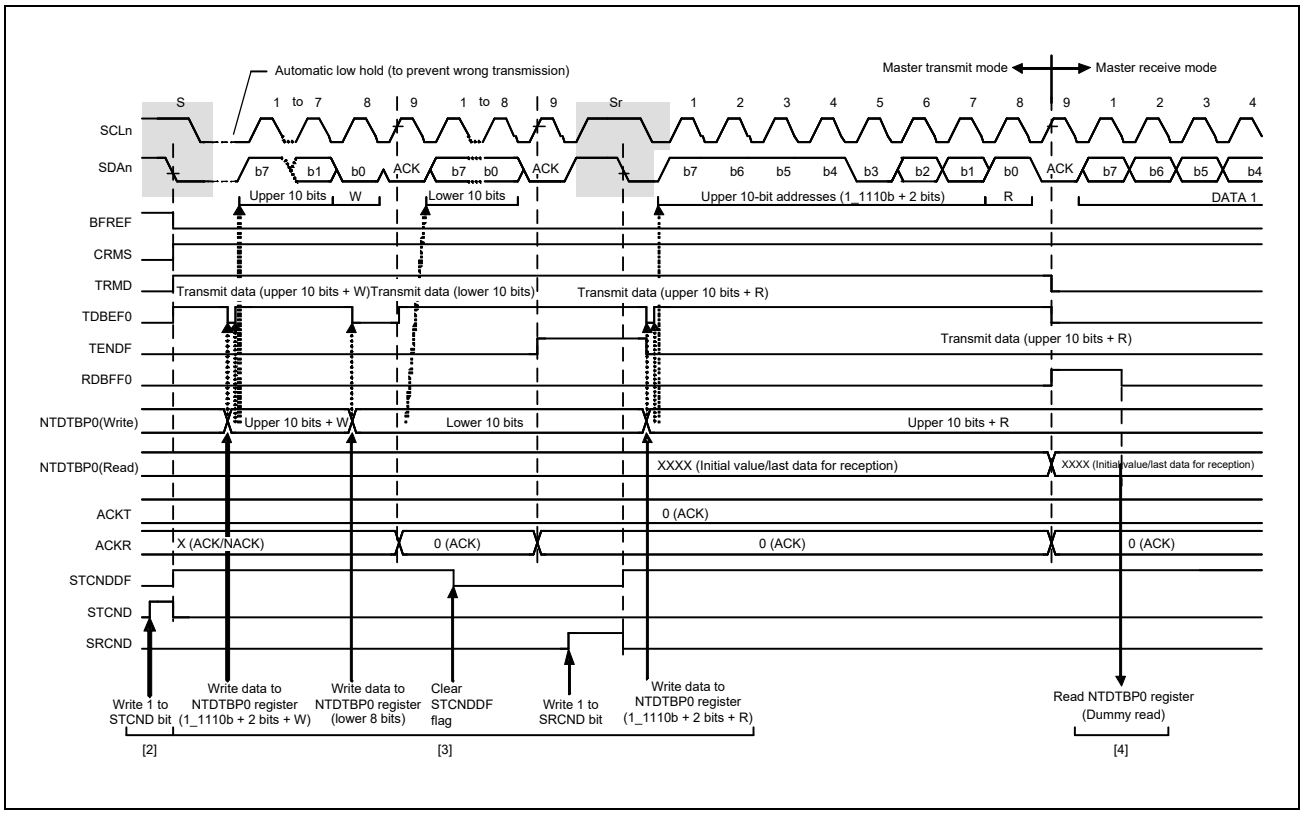


Figure 26.10 Master Receive Operation Timing (2) (10-bit Address Format, When ACKTWE = 0)

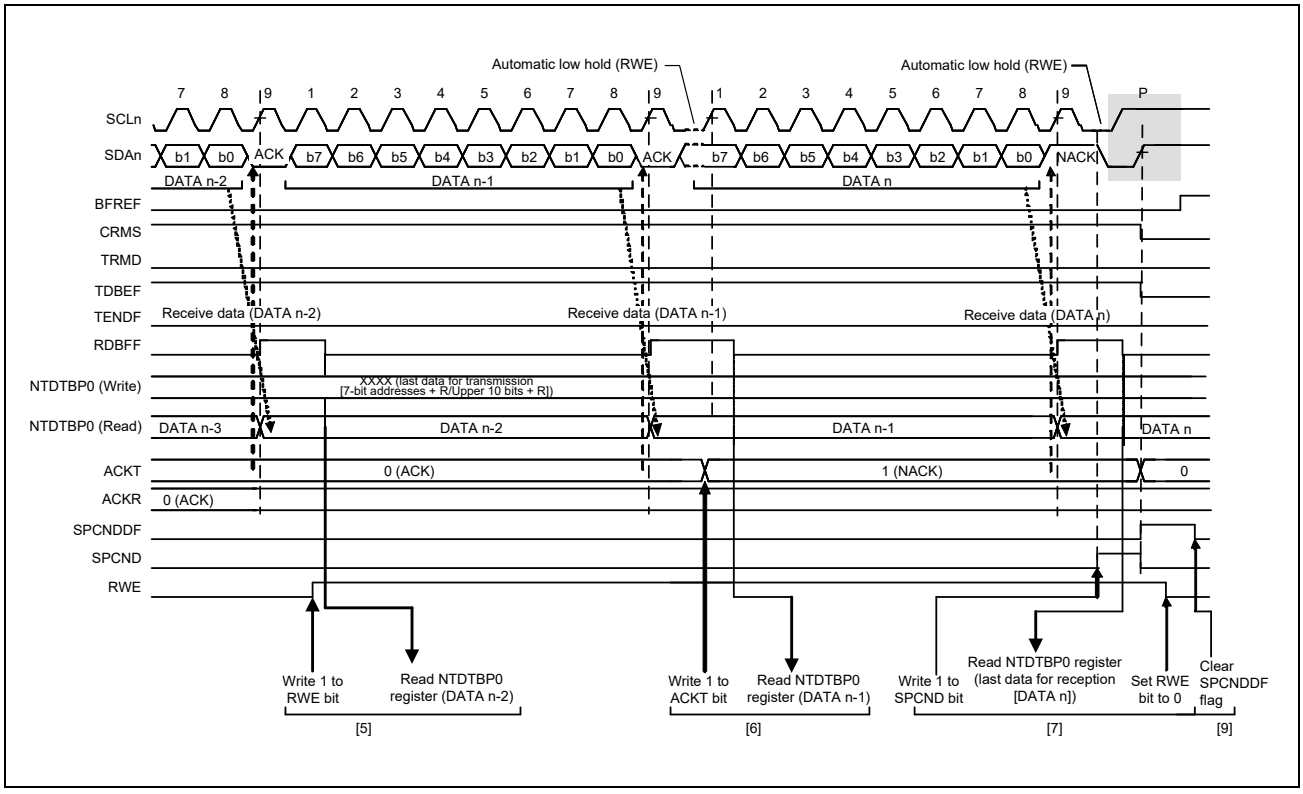


Figure 26.11 Master Receive Operation Timing (3) (When ACKTWE = 0)

(b) I3C Master Operation

1) Dynamic Address Assign Procedure

After initializing I3C, first execute Dynamic Address Assign Procedure for I3C Slave connected on the I3C Bus. The following describes the procedure.

1. Initial setting (refer to **Section 26.3.3.1(2), I3C Initial Setting Flow** for details)
2. Execute Dynamic Address Assign with ENTDAA or SETDASA Common Command Code (CCC) for I3C Slave set in DAT (DATBASm register).
   Write Command Descriptor (Address Assign Command) to Command Buffer via the NCMDQP register.
3. When Command Descriptor is written to Command Buffer, Transaction is issued on I3C Bus.
4. When ENTDAA is specified for CMD[7:0] of Address Assign Command:
   Execute Dynamic Address Assign for I3C Slave for the number of DATs specified by DEV_COUNT[3:0] starting with DAT specified by DEV_INDEX[4:0] of Address Assign Command.
   When SETDASA is specified for CMD[7:0] of Address Assign Command:
   Execute Dynamic Address Assign for I3C Slave indicated by DAT specified by DEV_INDEX[4:0] of Address Assign Command.
5. In case of ENTDAA, the Provisional ID, BCR, DCR transmitted from I3C Slave is stored in Receive Data Buffer (BCR is also automatically stored in the MSDCTm register).
   Read the Provisional ID, BCR, and DCR from the Receive Data Buffer via the NTDTBPn register with an interrupt by RDBFF0 = 1.
6. When execution of Dynamic Address Assign is completed, issue STOP condition and store the Response Descriptor into the Response Buffer.
7. Read the Response Descriptor via the NRSPQP register and check the status.
8. Check whether the value of the DATA_LENGTH[15:0] bits of the Response Descriptor match the value of DEV_COUNT[3:0] of the Address Assign Command.

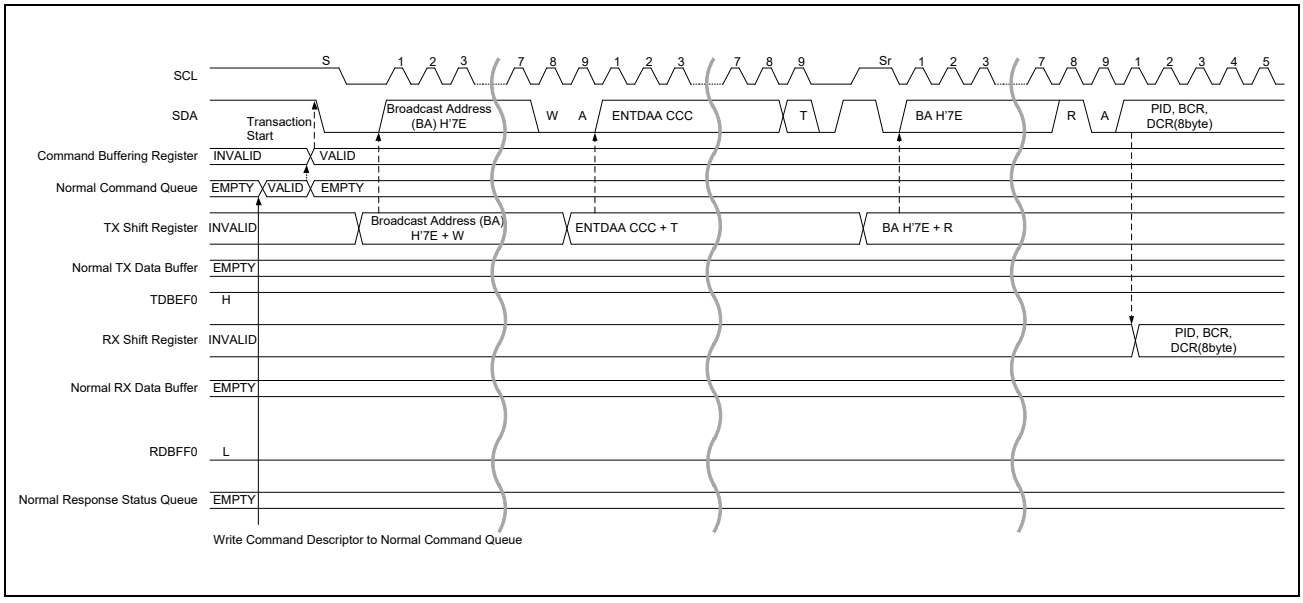


Figure 26.12 Dynamic Address Assign Procedure (ENTDAA CCC) Timing (1/3)

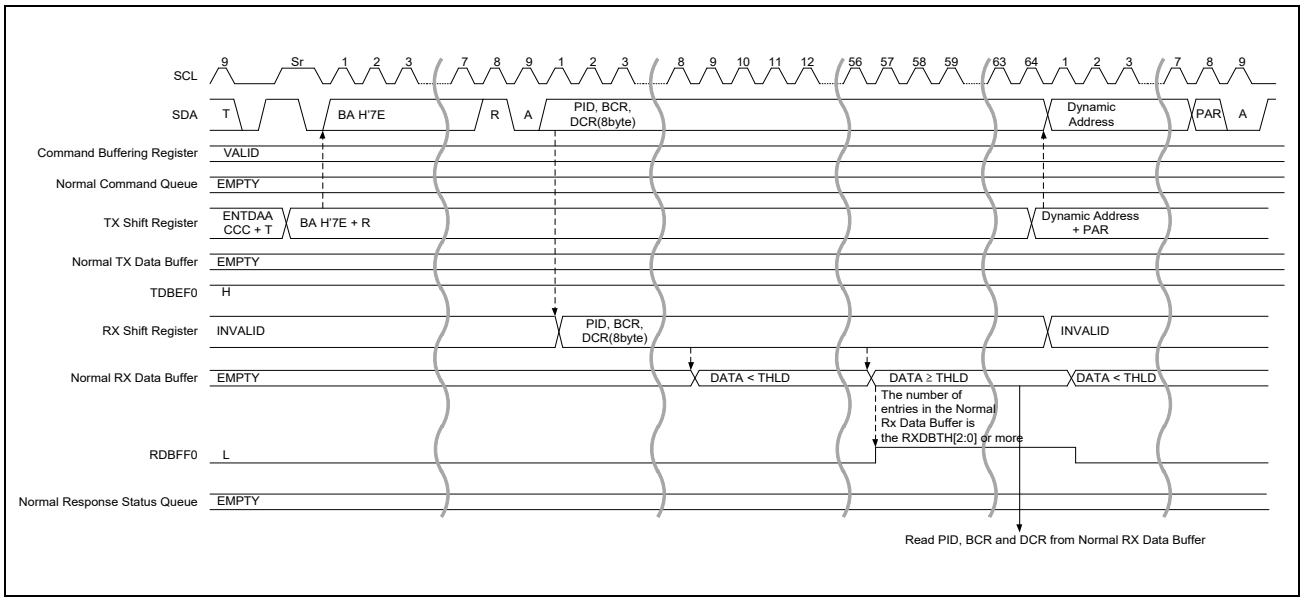


Figure 26.13 Dynamic Address Assign Procedure (ENTDAA CCC) Timing (2/3)

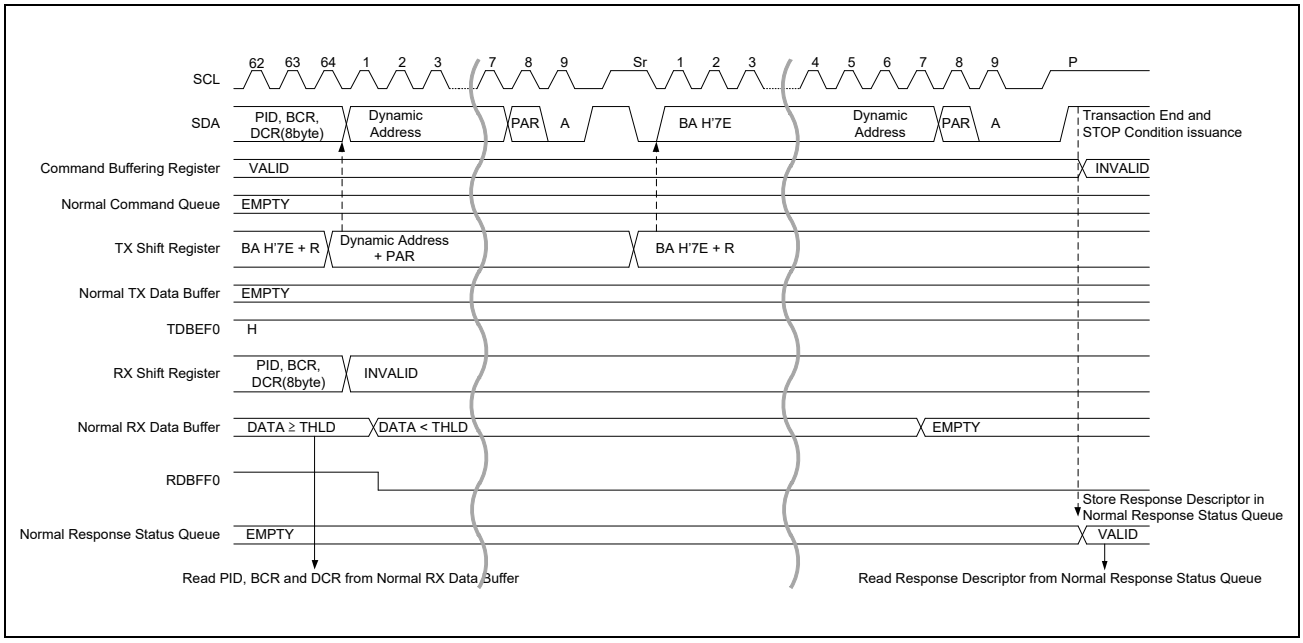


Figure 26.14 Dynamic Address Assign Procedure (ENTDAA CCC) Timing (3/3)

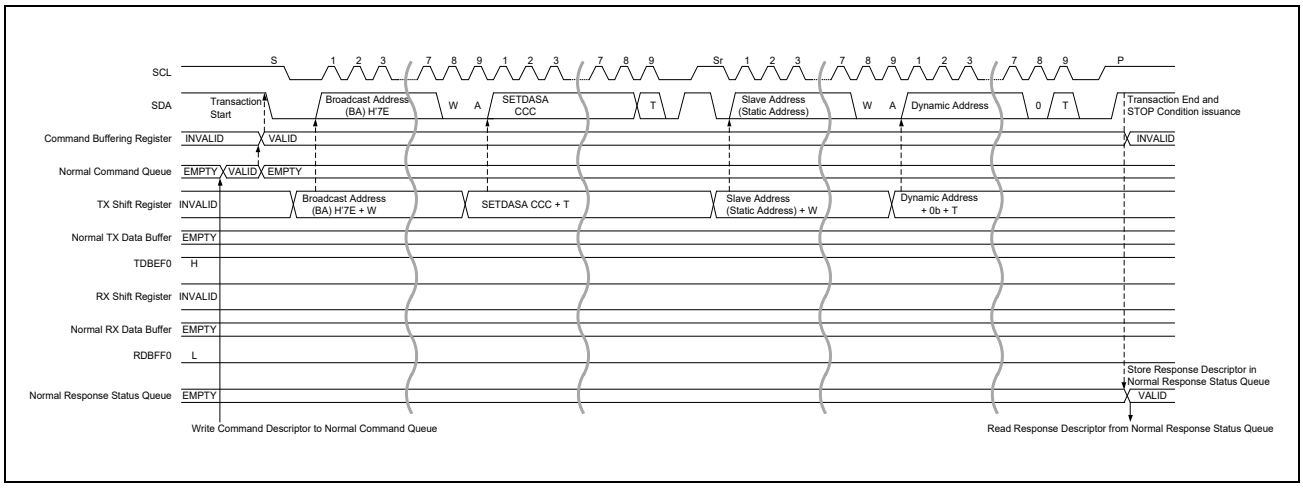


Figure 26.15 Dynamic Address Assign Procedure (SETDASA CCC) Timing

2) SDR Data Write Transfer

1. Write data for transmission to the Transmit Data Buffer via the NTDTBPn register.
2. Write a Command Descriptor (Immediate Transfer Command or Regular Transfer Command or Combo Transfer Command) for Data Transfer to the Command Buffer via the NCMDQP register.
3. When Command Descriptor is written to Command Buffer, transaction is issued on I3C Bus.
   When NACK is received with the Address Header, transaction of the same command is automatically issued according to the NACK Retry Count value (DATBASm.DVNACK) of DAT.
4. If data for transmission still remain, write data for transmission by an interrupt with TDBEF0 = 1 to the Transmit Data Buffer via the NTDTBPn register.
5. When data transmission for the number of Data Length specified by the DATA_LENGTH[15:0] bits of the Command Descriptor is completed, the Repeated START condition or STOP condition is issued and the Response Descriptor is stored in the Response Buffer.
6. Read the Response Descriptor via the NRSPQP register and check the status.
7. Check that the value of the DATA_LENGTH[15:0] bits of the Response Descriptor is 0.

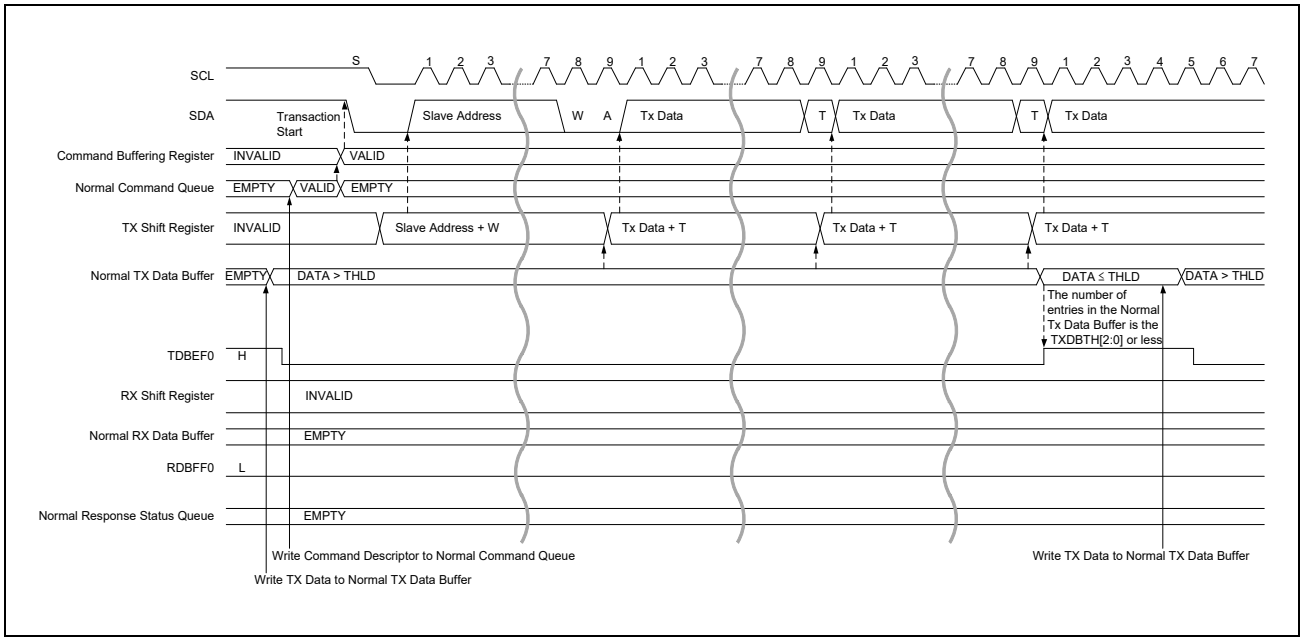


Figure 26.16 SDR Data Write Transfer Timing (1/2)

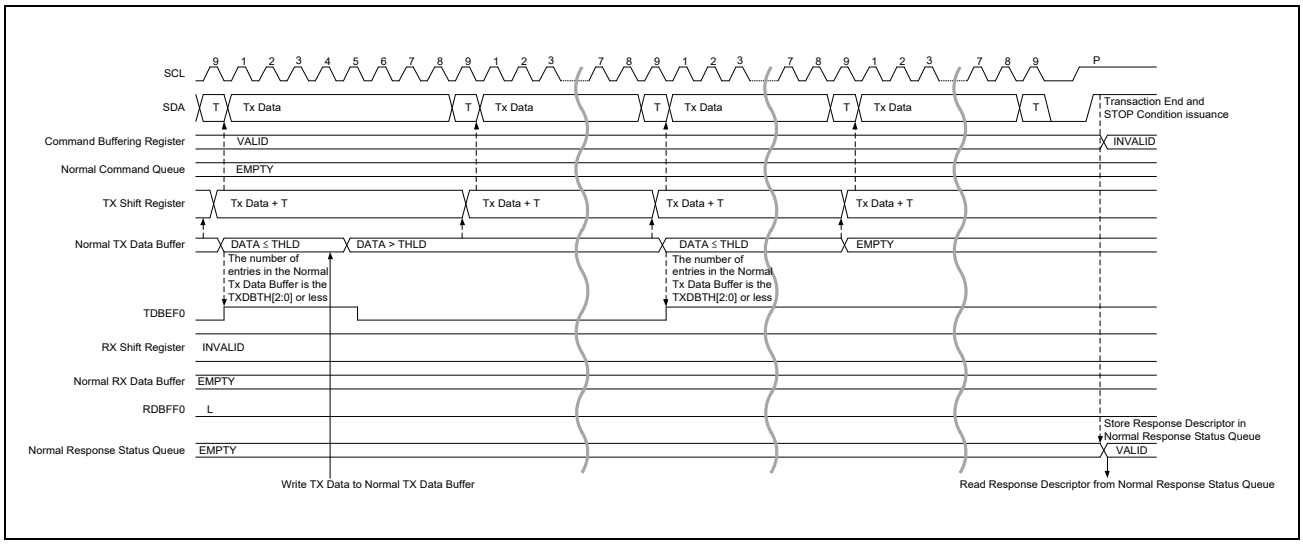


Figure 26.17 SDR Data Write Transfer Timing (2/2)

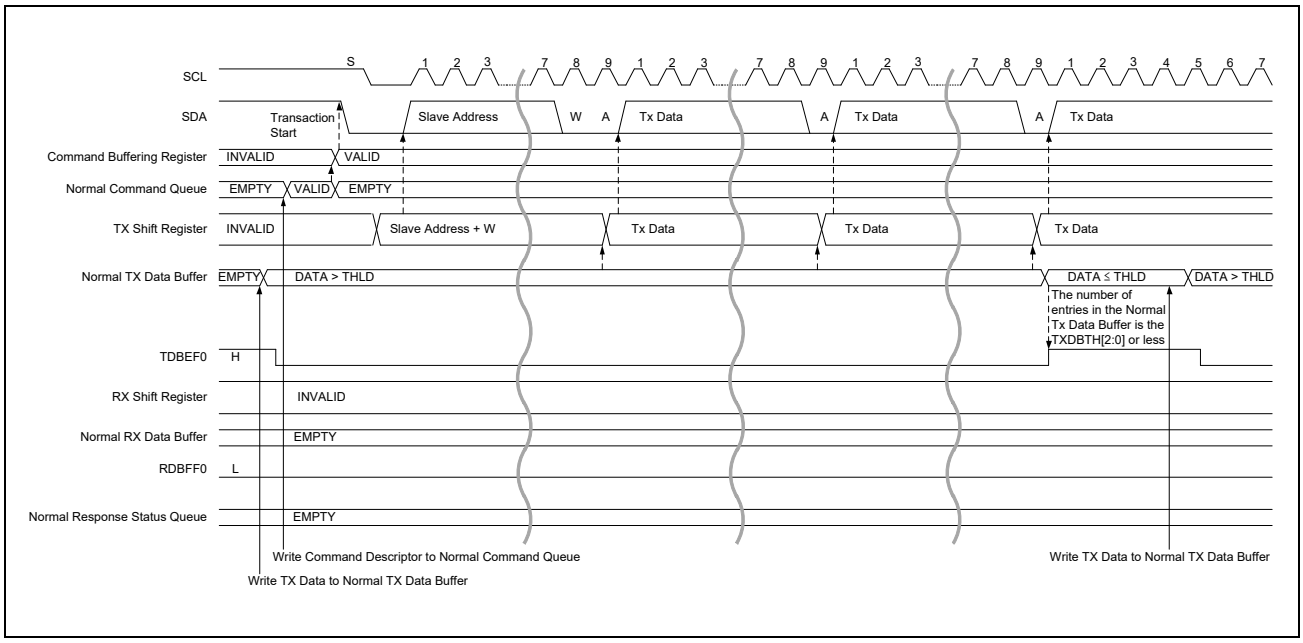


Figure 26.18 Legacy I²C Message Data Write Timing (1/2)

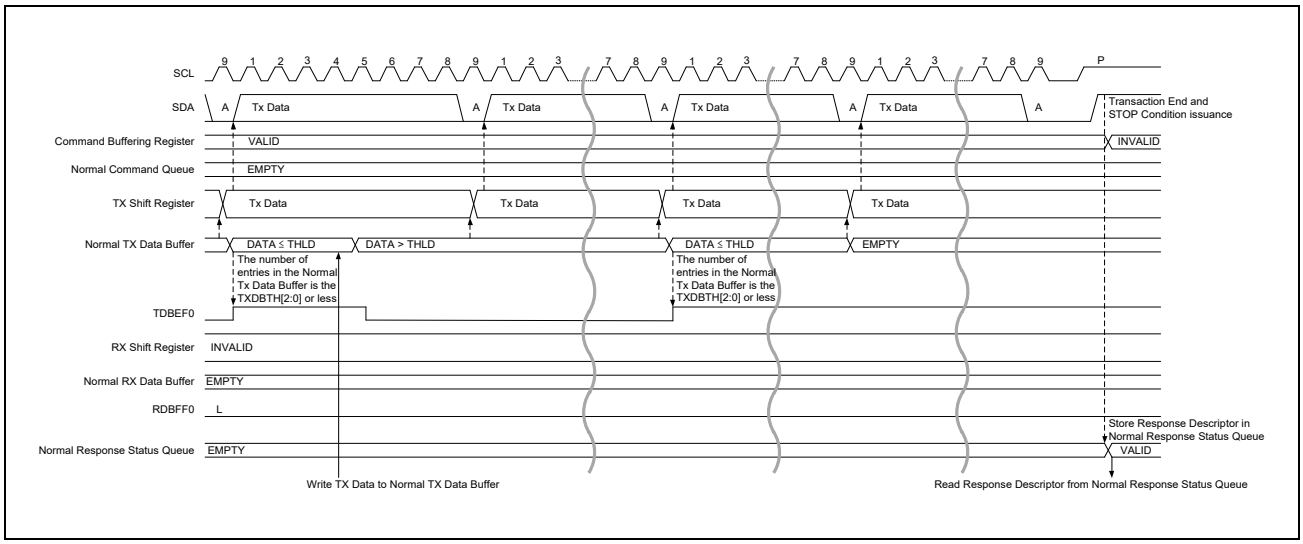


Figure 26.19 Legacy I²C Message Data Write Timing (2/2)

3) SDR Data Read Transfer

1. Write a Command Descriptor (Immediate Transfer Command or Regular Transfer Command or Combo Transfer Command) for Data Transfer to the Command Buffer via the NCMDQP register.
2. When Command Descriptor is written to Command Buffer, transaction is issued on I3C Bus.
   When NACK is received with the Address Header, transaction of the same command is automatically issued according to the NACK Retry Count value (DATBASm.DVNACK) of DAT.
3. Data received from the I3C Slave is stored in the Receive Data Buffer.
4. With the RDBFF0 = 1 interrupt, the received data is read from the Receive Data Buffer via the NTDTBPn register.
5. SDR:
   Detecting Low in T-bit or receiving Data for the number of Data Length specified by the DATA_LENGTH[15:0] bits of Command Descriptor is completed, issue Repeated START condition or STOP condition and store the Response Descriptor into the Response Buffer.
   Legacy I^2C Message:
   When data reception for the number of Data Length specified by the DATA_LENGTH[15:0] bits of Command Descriptor is completed, NACK is issued. After that, issue a Repeated START condition or STOP condition and store the Response Descriptor into the Response Buffer.
6. Read the Response Descriptor via the NRSPQP register and check the status.
7. Check whether the value of the DATA_LENGTH[15:0] bits of the Response Descriptor match the data length setting value of the Command Descriptor.

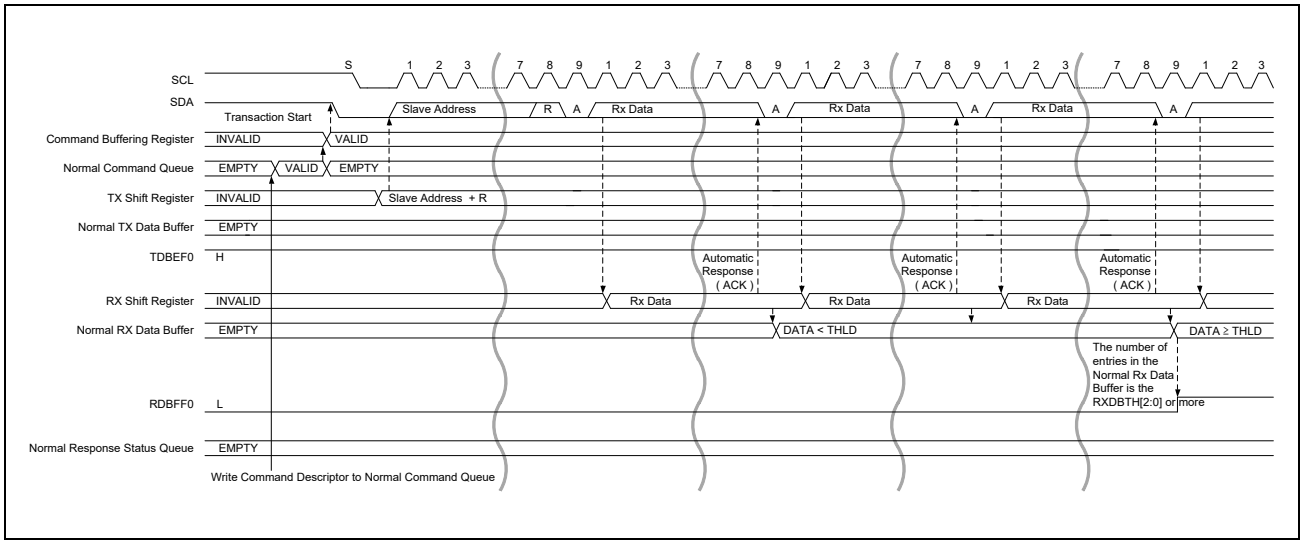


Figure 26.20 SDR Data Read Transfer Timing (1/2)

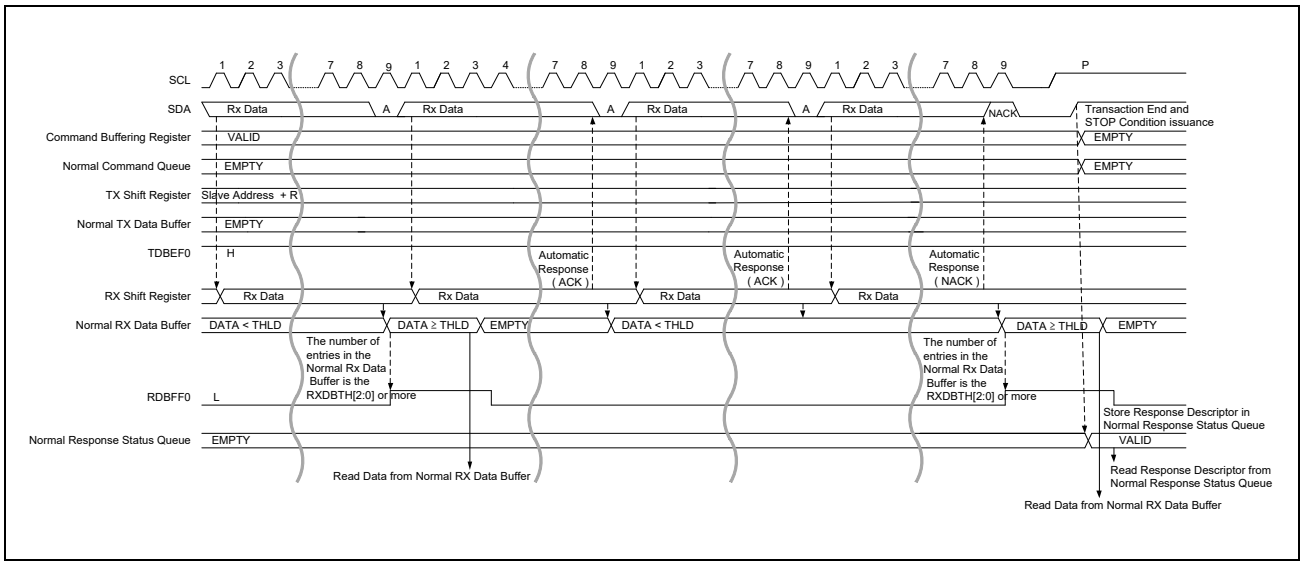


Figure 26.21 SDR Data Read Transfer Timing (2/2)

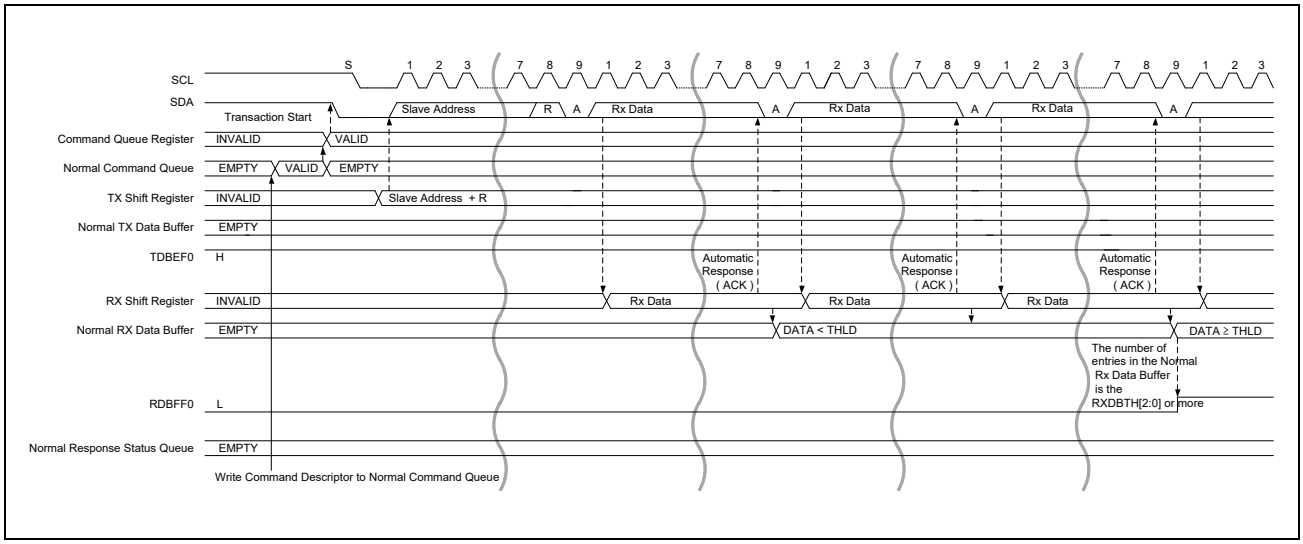


Figure 26.22 Legacy I²C Message Data Read Transfer Timing (1/2)

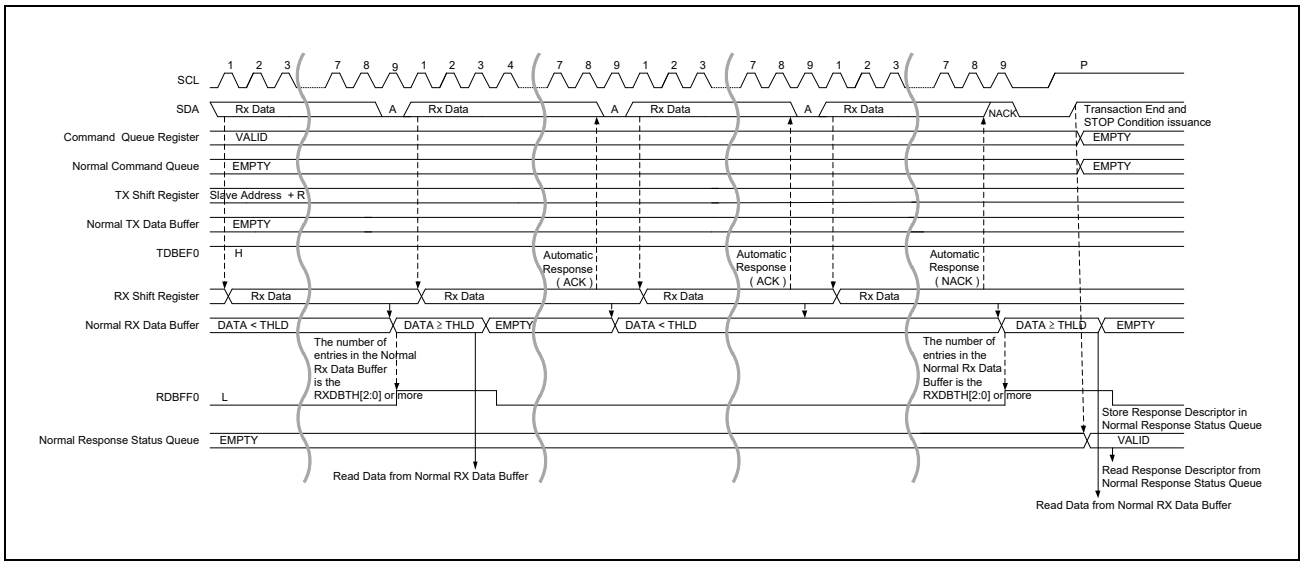


Figure 26.23 Legacy I²C Message Data Read Transfer Timing (2/2)

4) IBI Transfer

1. Write Command Descriptor to the Command Buffer and issue Transaction on I3C Bus.
   If START Request (SDA Low Drive) is issued from the slave device, I3C drives SCL to Low and completes START condition.
   Thereafter, the SCL is supplied, and In-Band Interrupt Request is received.
2. In Slave Address with RnW of the Address Header, if losing Arbitration by issuing In-Band Interrupt from I3C Slave, stop issuing Transaction.
3. According to **Section 26.3.2.3(8), In-Band Interrupt [I3C mode]**, detect In-Band Interrupt and process.
4. In the interrupt with IBIQEFF = 1, read the IBI Status Descriptor from the IBI Status Buffer via the NIBIQP register and check the status.
   When detected a Slave Interrupt Request and responded with ACK, Read the IBI Data for the Data Length indicated by the DATA_LENGTH[15:0] bits of the IBI Status Descriptor from the IBI Data Buffer via the NIBIQP register.
5. Restart issuing Transaction of Command of Step1.

An example of the processing procedure after detection of In-Band Interrupt is shown below.

Processing procedure for detecting Mastership Request and transferring master right to Secondary Master

1. If the I3C Secondary Master wins the Arbitration, issue a DEFSLVS CCC and notify Slave information to Secondary Master.
2. Issue a GETACCMST CCC and complete CCC by a STOP condition.

NOTE

- After transferring master right to Secondary Master, to get master right again, issue a Mastership Request according to (f) IBI Transfer of **Section 26.3.2.1(2)(b), I3C Slave Operation**.
- After Mastership Request is accepted by the Current Master, to get master right again at receiving the GETACCMST CCC and complete CCC by a STOP condition.

Processing procedure when Hot-Join Event is detected

1. Issue a Broadcast Command Code Enter Dynamic Address Assignment (ENTDAA) to start the Dynamic Address Assignment process.
2. Issue a DEFSLVS CCC and notify Slave information to Secondary Master.

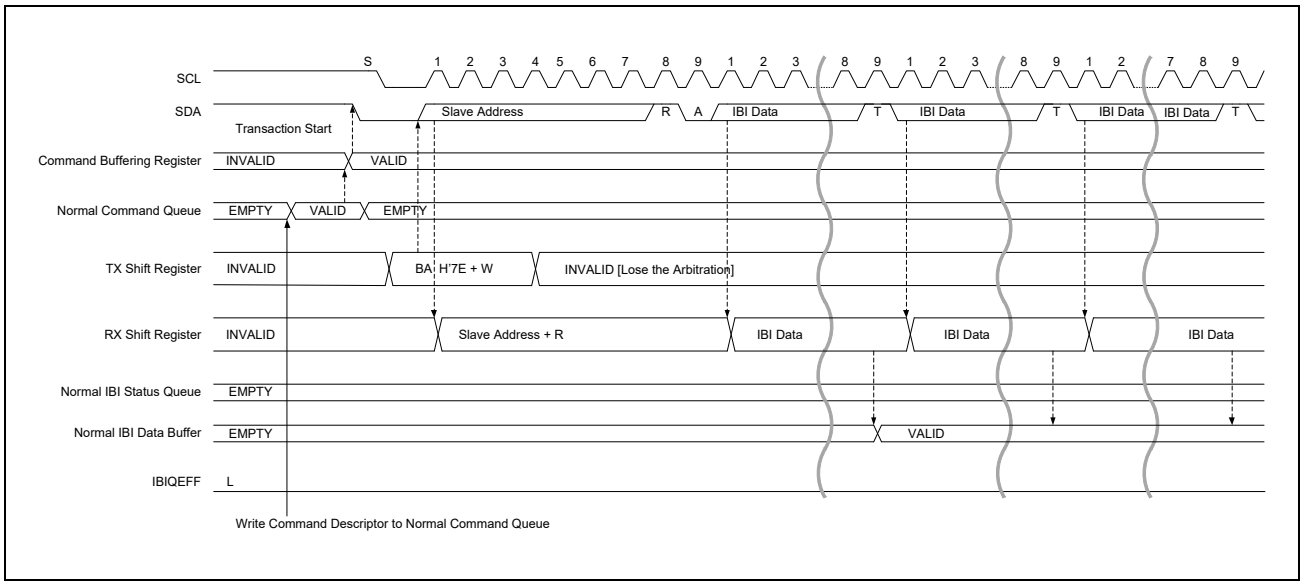


Figure 26.24 I3C Master IBI Transfer Timing (1/2)

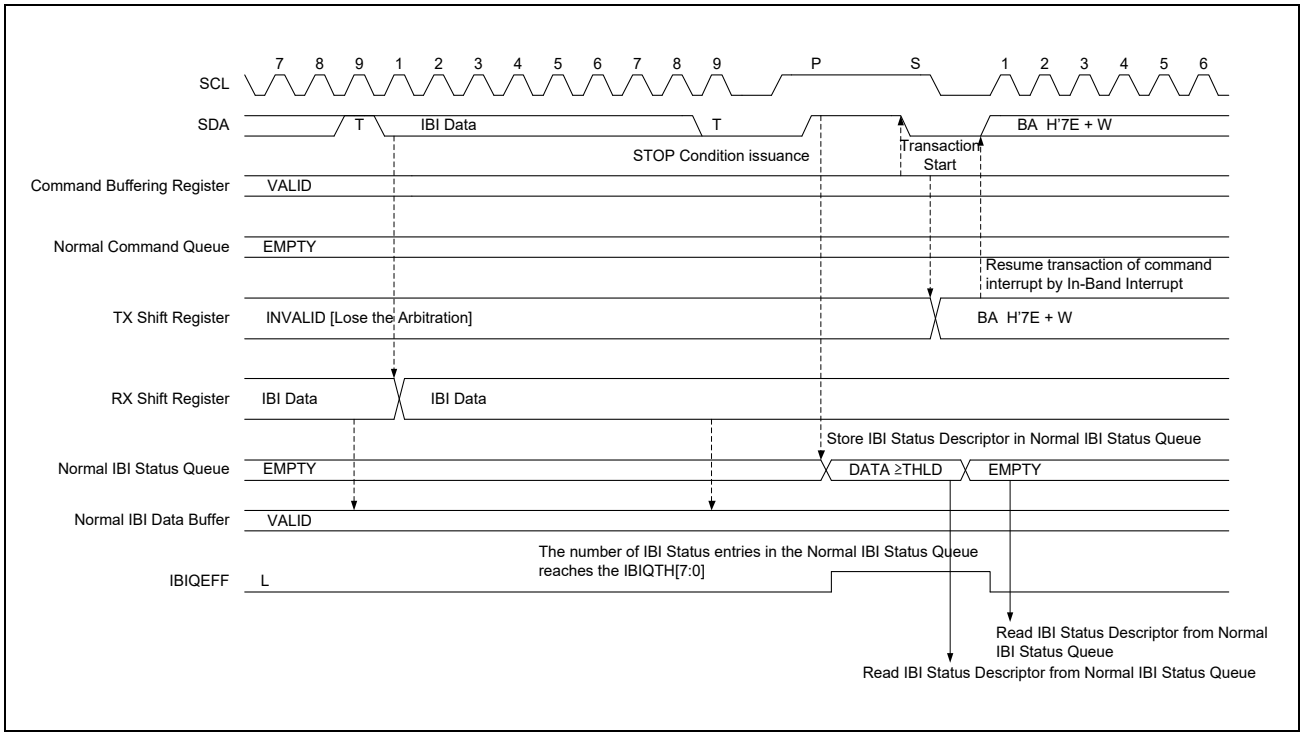
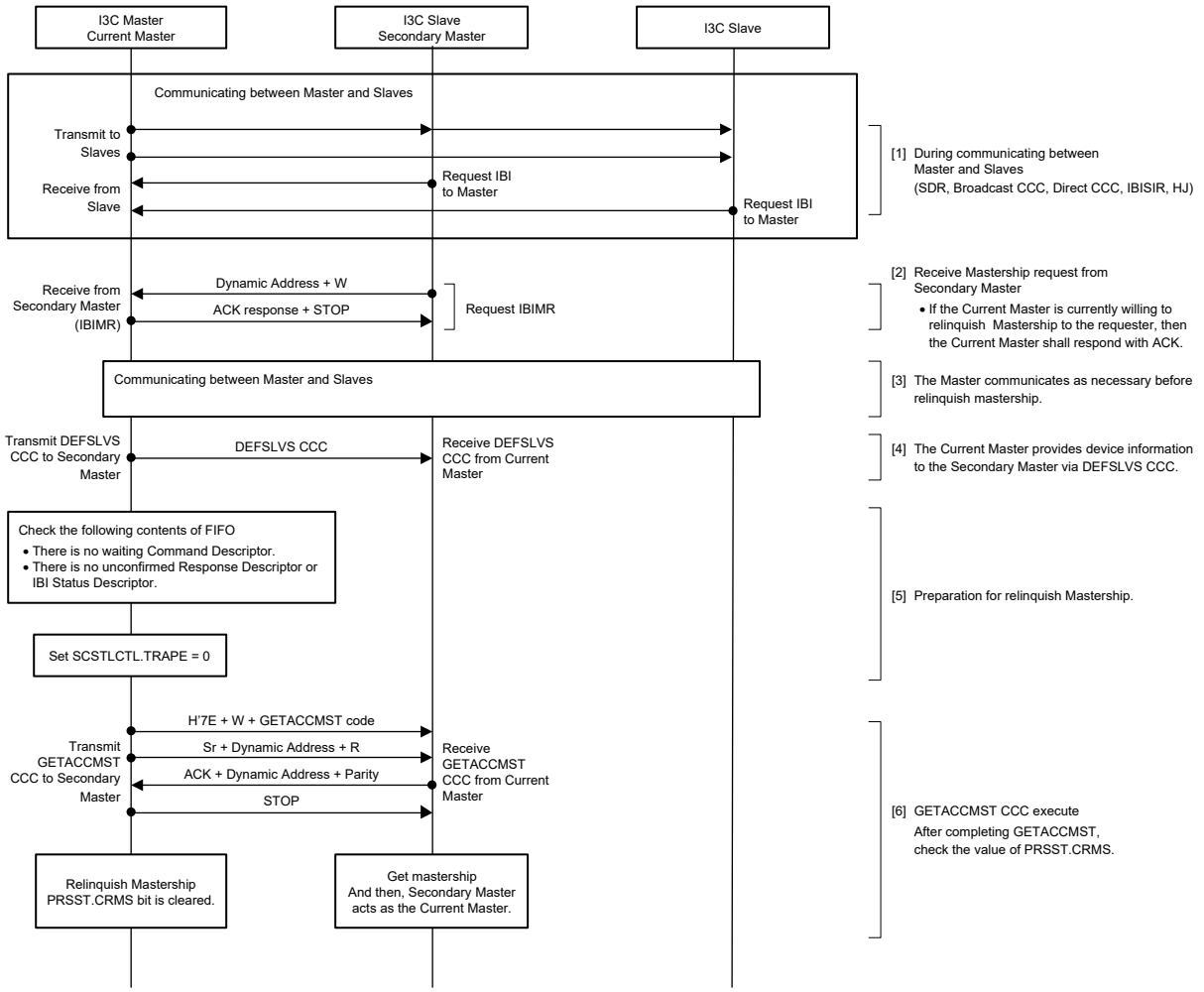


Figure 26.25 I3C Master IBI Transfer Timing (2/2)



Note: The Abort request during GETACCMST by BCTL.ABT setting are prohibited due to the transfer of Mastership.

Figure 26.26 I3C Master Mastership Processing Flow

(2) Slave Mode Operation

(a) I²C Slave Operation

1) Data Write Transfer (Single Buffer Transfer)

In slave receive operation, the master device outputs the SCL clock and transmit data, and I3C returns acknowledgments as a slave device.

Figure 26.126 shows an example of usage of slave reception and **Figure 26.27** and **Figure 26.28** show the timing of operations in slave reception.

The following describes the procedure and operations for slave reception.

1. Initial settings. For details, refer to **Section 26.3.3.1, Initial Setting Flow**. After initial settings, I3C will stay in the standby state until it receives a slave address that it matches.
2. After receiving a matching slave address, I3C sets one of the corresponding bits SVST.HOAF, GCAF, and SVAFy (y = 0 to 2) to 1 on the rising edge of the ninth cycle of SCL clock (the clock signal) and outputs the acknowledge bit (ACK) on the ninth cycle of SCL clock. If the value of the R/W# bit that was also received at this time is 0, I3C continues to place itself in slave receive mode and sets the NTST.RDBFF0 flag to 1.
3. After the BST.SPCNDDF flag is confirmed to be 0 and the NTST.RDBFF0 flag to be 1, dummy read the NTDTBP0 register (the dummy value consists of the slave address and R/W# bit when the 7-bit address format is selected, or the lower 8 bits when the 10-bit address format is selected).
4. When the NTDTBP0 register is read, I3C automatically sets the NTST.RDBFF0 flag to 0. If reading of the NTDTBP0 register is delayed and a next byte is received while the RDBFF0 flag is still set to 1, I3C holds the SCLn line low from one SCL cycle before the timing with which RDBFF0 should be set. In this case, reading the NTDTBP0 register releases the SCLn line from being held at the low level. When the BST.SPCNDDF flag = 1 and the NTST.RDBFF0 flag is also 1, read the NTDTBP0 register until all the data is completely received.
5. Upon detecting the STOP condition, I3C automatically clears bits SVST.HOAF, GCAF, and SVAFy (y = 0 to 2) to 0.
6. After checking that the BST.SPCNDDF flag = 1, set the BST.SPCNDDF flag to 0 for the next transfer operation.

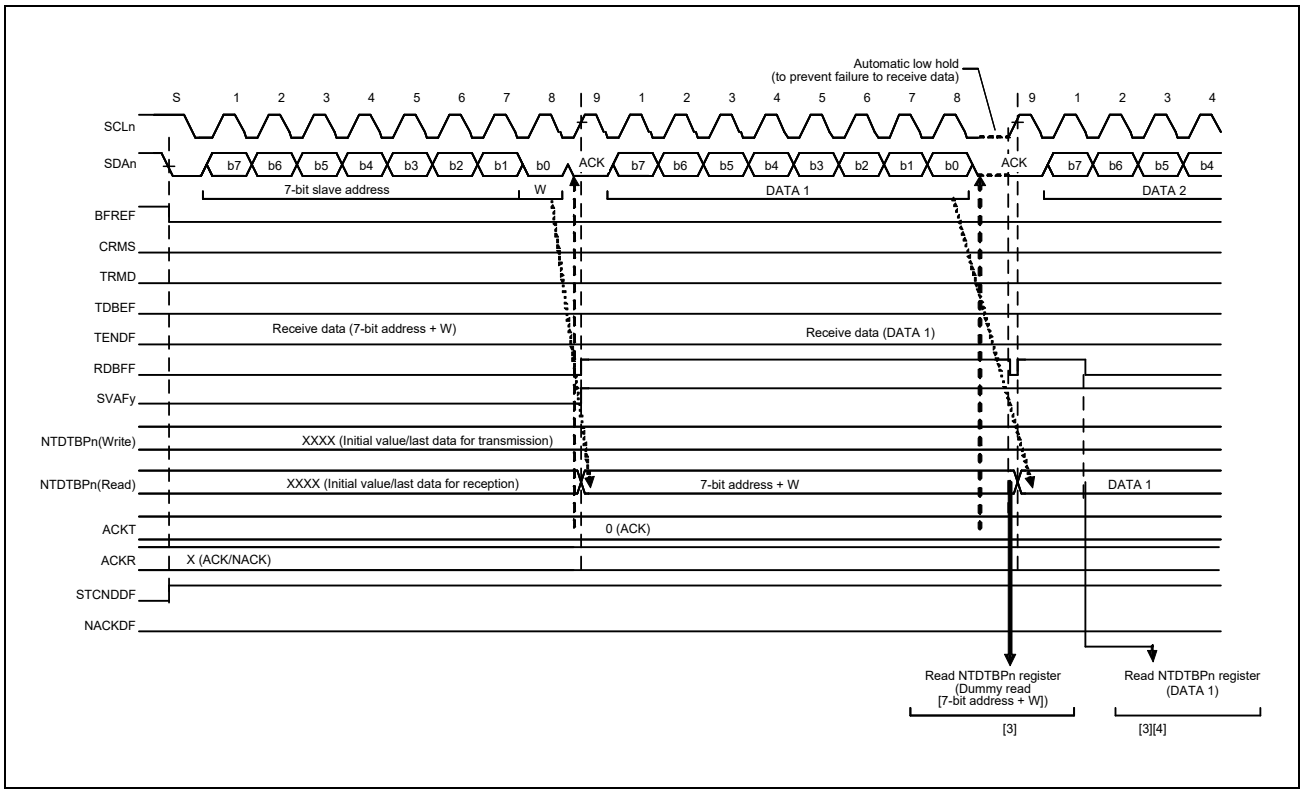


Figure 26.27 Slave Receive Operation Timing (1) (7-bit Address Format, When ACKTWE = 0)

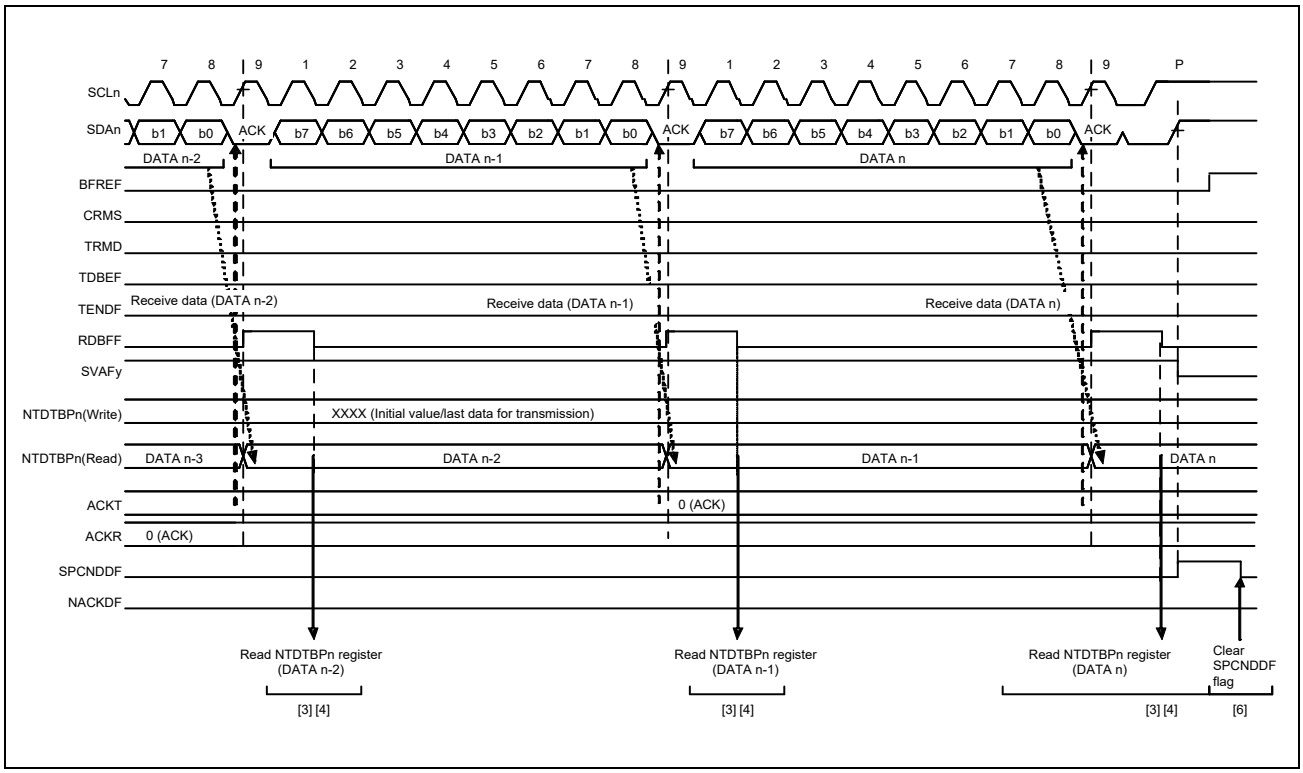


Figure 26.28 Slave Receive Operation Timing (2) (When ACKTWE = 0)

2) Data Read Transfer (Single Buffer transfer)

In slave transmit operation, the master device outputs the SCL clock, I3C transmits data as a slave device, and the master device returns acknowledgments.

Figure 26.125 shows an example of usage of slave transmission and **Figure 26.29** and **Figure 26.30** show the timing of operations in slave transmission.

The following describes the procedure and operations for slave transmission.

1. Initial settings. For details, refer to **Section 26.3.3.1, Initial Setting Flow**.
   After initial settings, I3C will stay in the standby state until it receives a slave address that it matches.
2. After receiving a matching slave address, I3C sets one of the corresponding bits SVST.HOAF, GCAF, and SVAFy (y = 0 to 2) to 1 on the rising edge of the ninth cycle of SCL clock (the clock signal) and outputs the acknowledge bit (ACK) on the ninth cycle of SCL clock. If the value of the R/W# bit that was also received at this time is 1, I3C automatically places itself in slave transmit mode by setting both the PRSST.TRMD bit and the NTST.TDBEF0 flag to 1.
3. After the NTST.TDBEF0 flag is confirmed to be 1, write the data for transmission to the NTDTBP0 register. At this time, if I3C does not receive acknowledge from the master device (receives an NACK signal) while the BSTE.NACKDE bit = 1, I3C aborts transfer of the next data.
4. Wait until the following (a) or (b) condition.
   (a) The BST.NACKDF flag is set to 1.
   (b) The BST.TENDF flag is set to 1 while the NTST.TDBEF0 flag = 1, after the last byte for transmission is written to the NTDTBP0 register.
   When the BST.NACKDF flag or the TENDF flag = 1, I3C drives the SCLn line low on the ninth falling edge of SCL clock.
5. When the BST.NACKDF flag or the BST.TENDF flag = 1, dummy read the NTDTBP0 register to complete the processing. This releases the SCLn line.
6. Upon detecting the STOP condition, I3C automatically sets bits SVST.HOAF, GCAF, and SVAFy (y = 0 to 2), flags NTST.TDBEF0 and BST.TENDF, and the PRSST.TRMD bit to 0, and enters slave receive mode.
7. After checking that the BST.SPCNDDF flag = 1, set the BST.NACKDF and SPCNDDF flags to 0 for the next transfer operation.

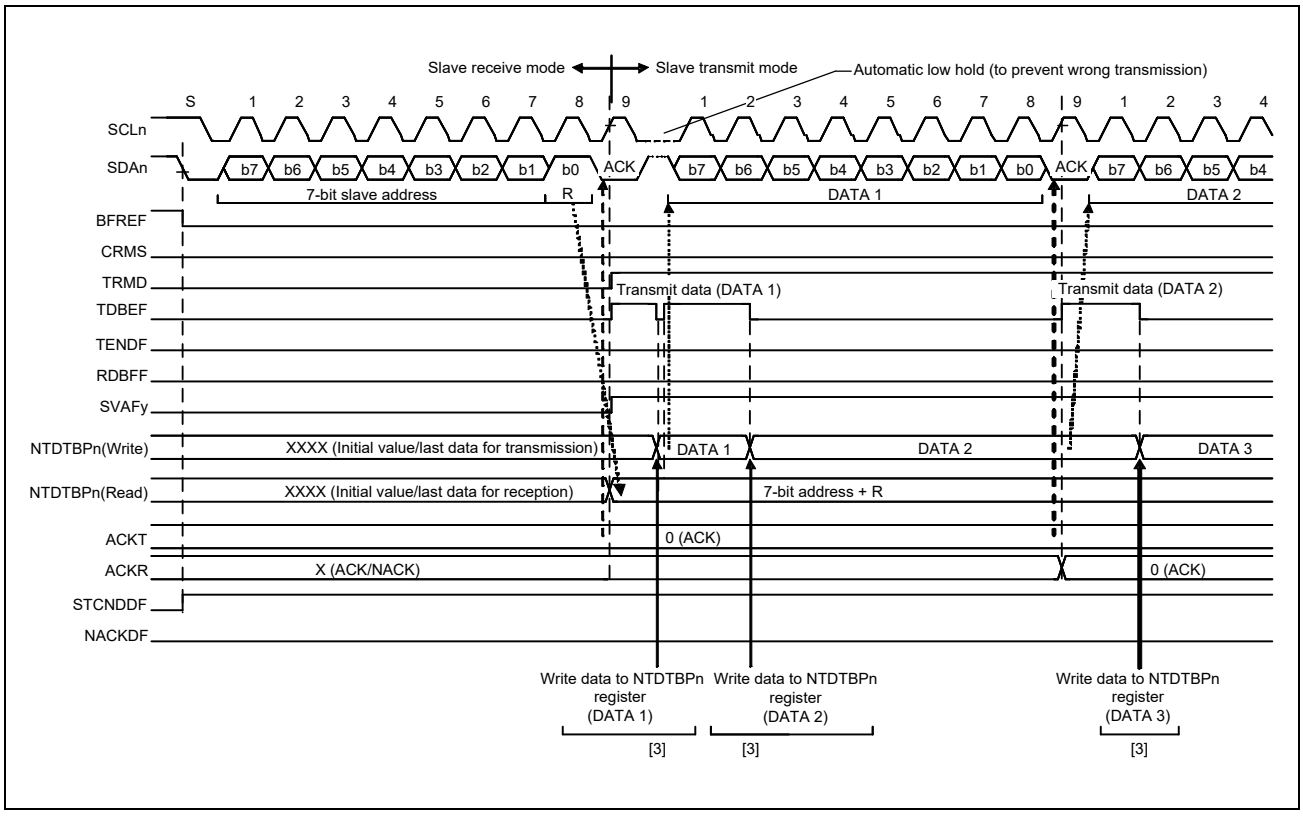


Figure 26.29 Slave Transmit Operation Timing (1) (7-bit Address Format)

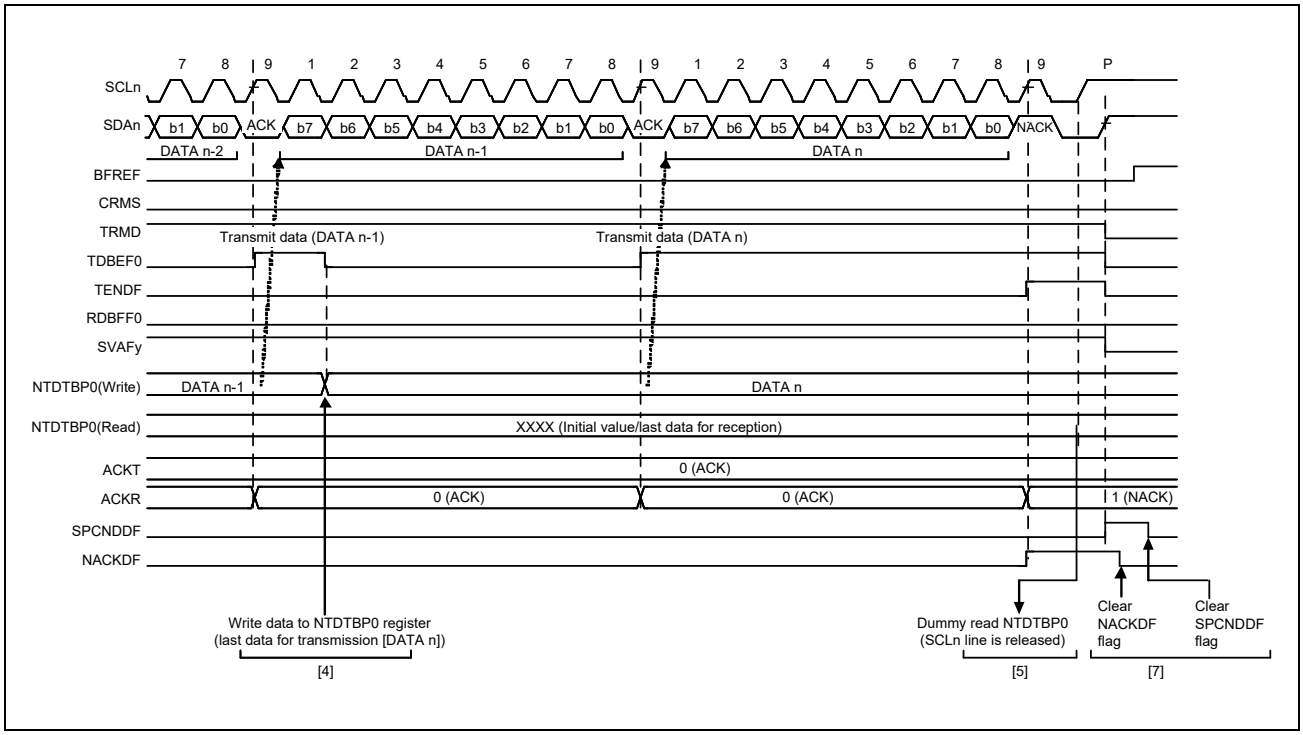
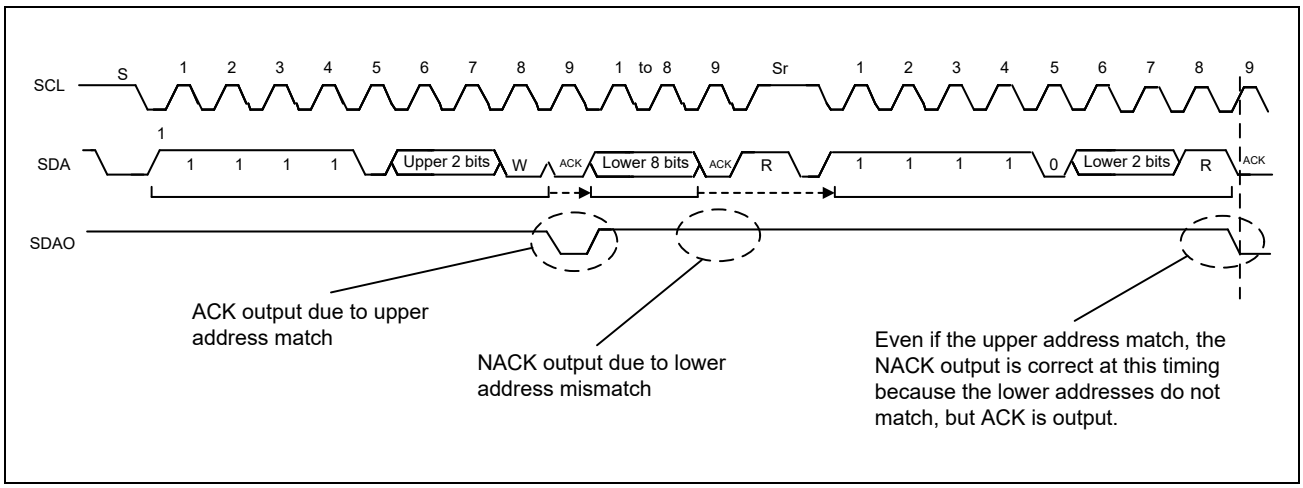


Figure 26.30 Slave Transmit Operation Timing (2)

NOTE

[10-bit address format Slave transmission]



When multiple I²C slaves are connected to the I²C bus and there is a possibility that an I²C Slave other than this module will NACK for the upper address/R after Repeated START, there are the following restrictions and workarounds.

- Work around: Set the upper 2 bits of the 10-bit address assigned to this module to a value different from other Slave. If the address is exhausted and cannot be set to a different value, use restriction (1).
- Restriction (1): 10-bit address not available.
- Restriction (2): After the ACK response in the red circle in the above figure, none of the slaves respond to data, so the SDA keeps the high level, and the I²C Master receives the H'FF data. In the case of a system that can handle H'FF as abnormal data, H'FF is read and discarded on the I²C Master side. If H'FF is valid data, use restriction (1).

3) Data Write Trans

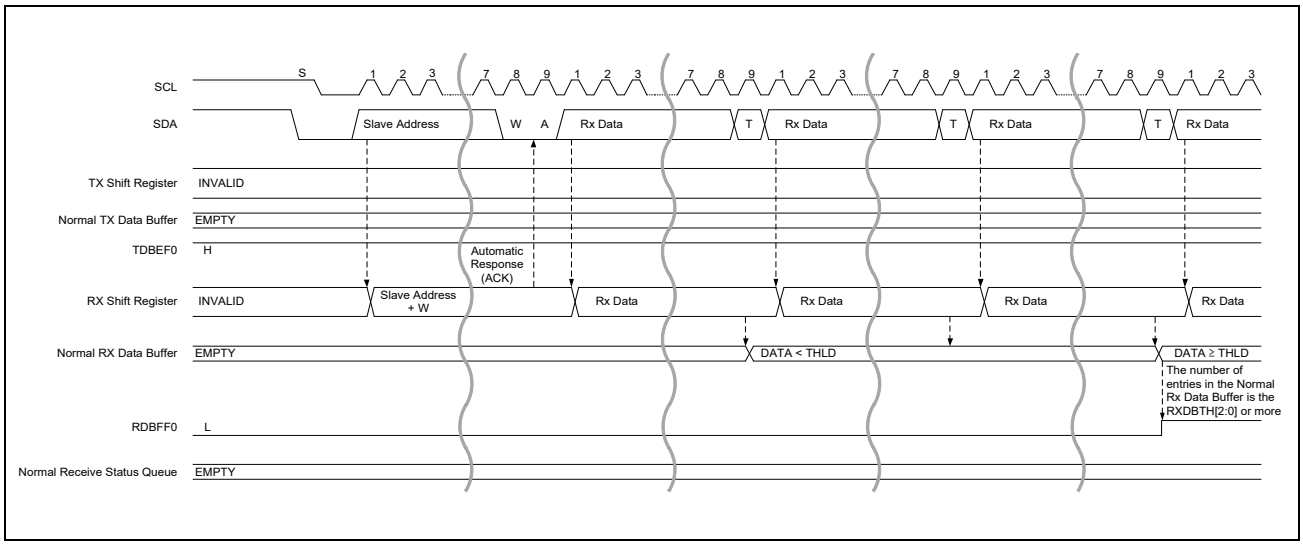


Figure 26.31 Data Write Transfer (FIFO Buffer Transfer) Timing (1)

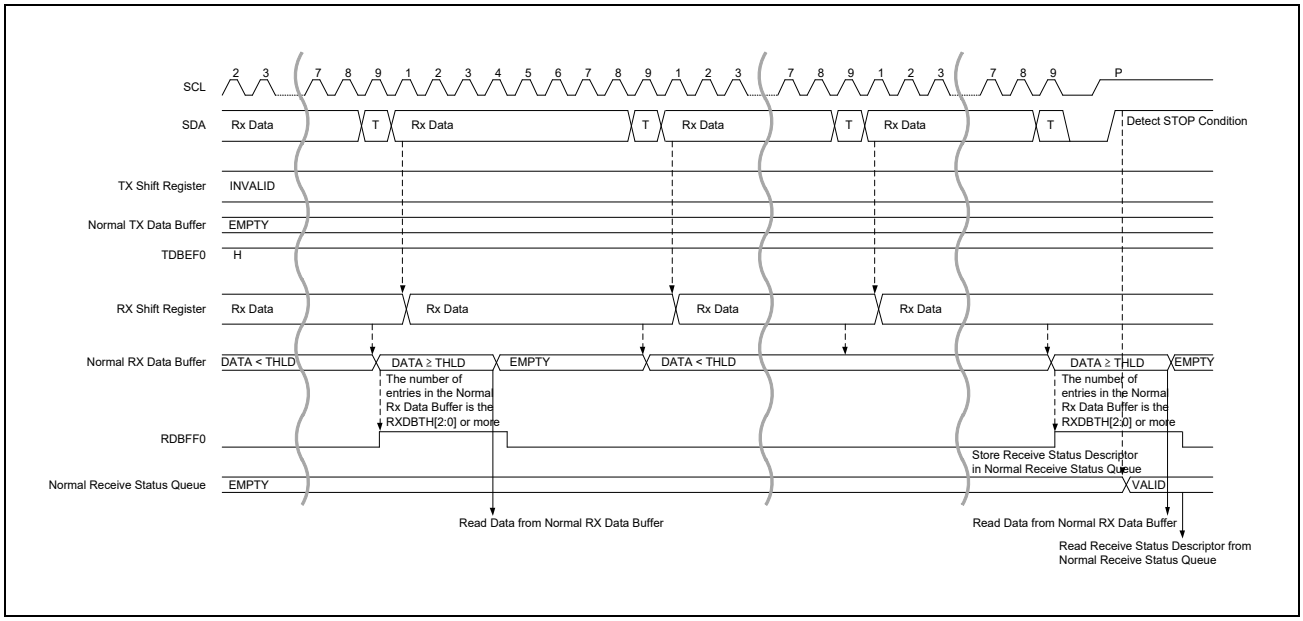


Figure 26.32 Data Write Transfer (FIFO Buffer Transfer) Timing (2)

(b) I3C Slave Operation

1) Dynamic Address Assign Procedure

After initializing I3C, the I3C master first performs Dynamic Address Assign Procedure.

The operation of R-I3 during the Dynamic Address Assign Procedure by ENTDAA CCC is described below.

1. Initial setting (For details, refer to **Section 26.3.3.1(2), I3C Initial Setting Flow**)
2. When ENTDAA CCC is received, I3C transmits Provisional ID (SDCTPIDH[31:0], SDCTPIDL[15:0]), BCR (SVDCT.TBCR[7:0]), DCR (SVDCT.TDCR[7:0]) until a dynamic address is assigned. (For details, refer to "In case of Broadcast CCC (ENTDAA)" of **Section 26.3.2.3(4)(e), CCC detection function [I3C mode]**.)
3. When ENTDAA CCC is completed and a STOP condition is detected, Receive Status Descriptor is stored in Receive Status Buffer.
4. Read Receive Status Descriptor via NRSQP register and check the status.
5. Read the data for the Data Length indicated by the DATA_LENGTH[15:0] bits of the Receive Status Descriptor from the Receive Data Buffer via the NTDTBP0 register.

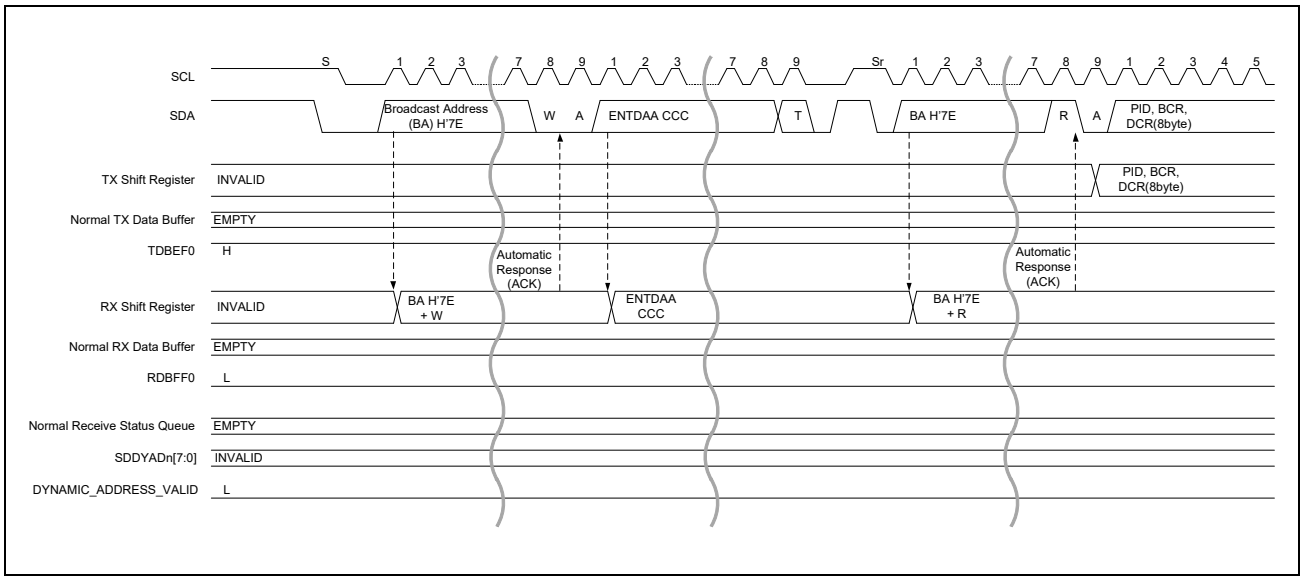


Figure 26.33 Dynamic Address Assign Procedure (ENTDAA CCC) Timing (1/3)

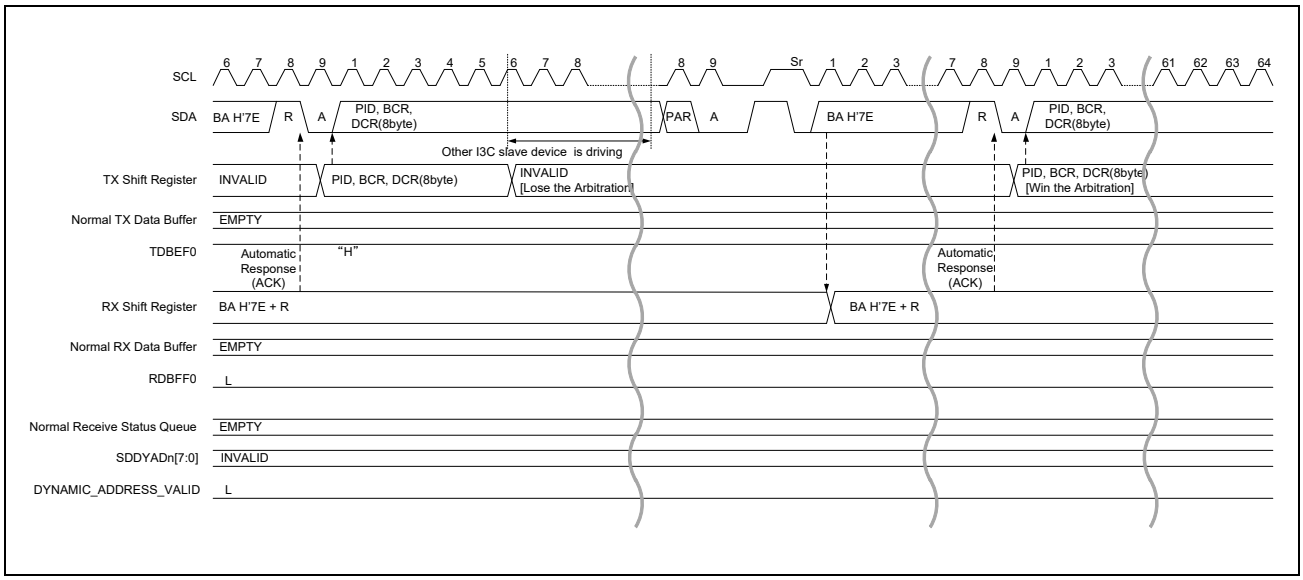


Figure 26.34 Dynamic Address Assign Procedure (ENTDAA CCC) Timing (2/3)

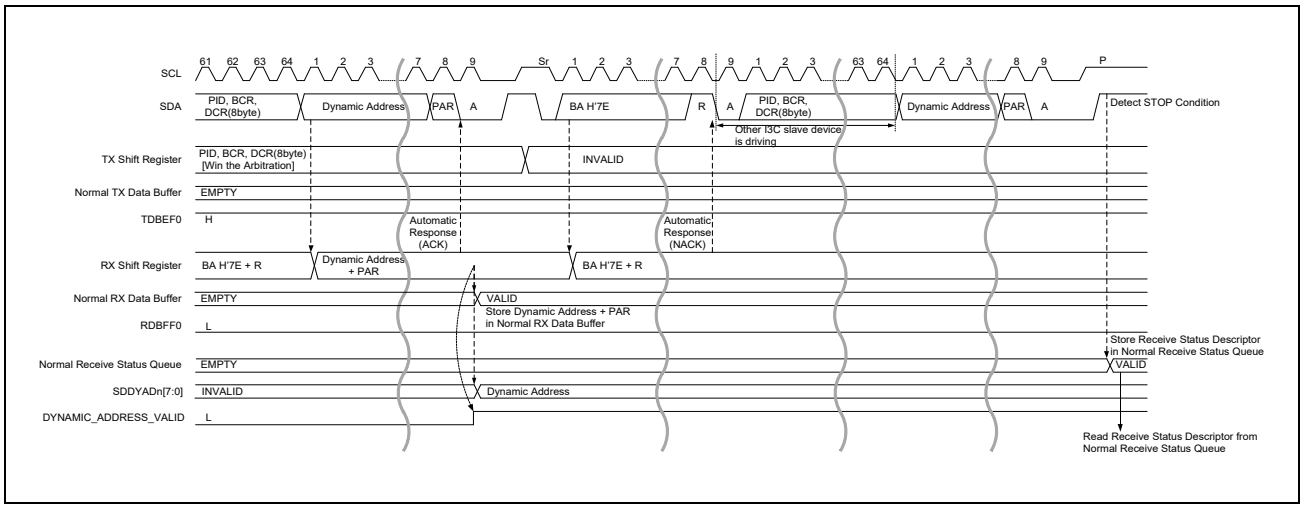


Figure 26.35 Dynamic Address Assign Procedure (ENTDAA CCC) Timing (3/3)

When communicating with a Static Address until the Dynamic Address is assigned from the I3C Master, by setting to the DVSTAD[6:0] bit of DAT (SDATBASn register), the SSTADV bit of the SVDVADn register is set to 1 and the Static Address Will be effective.

If the I3C Slave has a Static Address and the I3C Master executes the Dynamic Address Assign Procedure, it is possible to assign a Dynamic Address with SETDASA CCC.

The operation of I3C during SETDASA CCC Dynamic Address Assign Procedure is described below.

1. Initial setting (For details, refer to **Section 26.3.3.1(2), I3C Initial Setting Flow**)
2. When SETDASA CCC which agrees with its own Static Address is received, the SDDYAD[7:0] bit of DAT (SDATBAS0 register) is renewed and SDYADV bit of SVDVAD0 register is set in 1. (For details, refer to "In case of Direct Write CCC" of **Section 26.3.2.3(4)(e), CCC detection function [I3C mode]**.)
3. When SETDASA CCC is completed and a STOP condition is detected, Receive Status Descriptor is stored in Receive Status Buffer.

4. Read Receive Status Descriptor via NRSQP register and check the status.

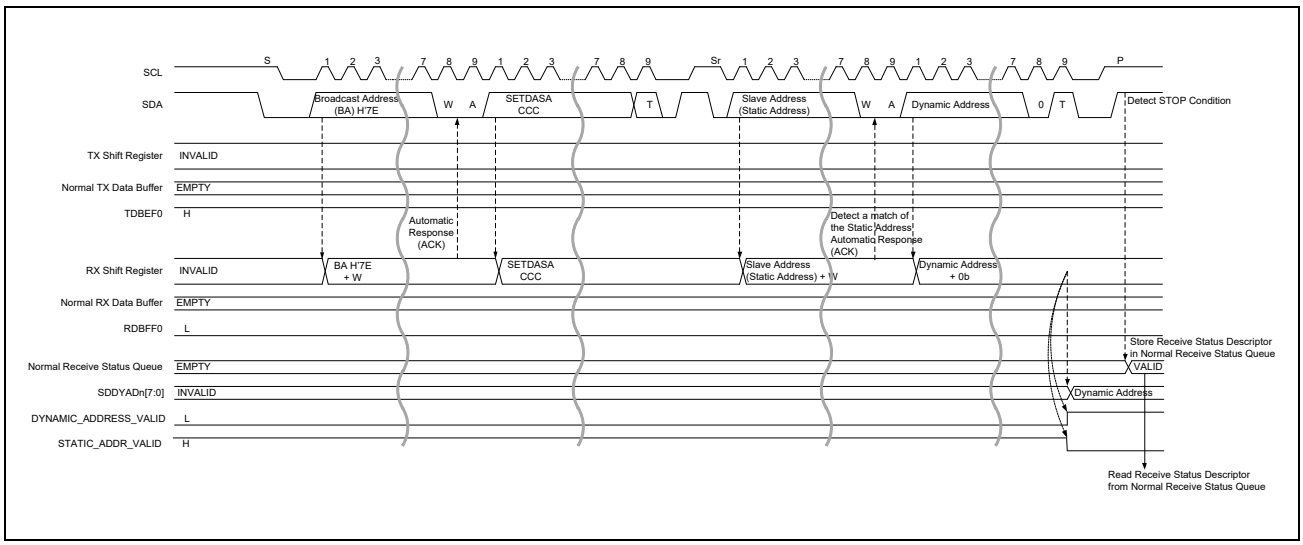


Figure 26.36 Dynamic Address Assign Procedure (SETDASA CCC) Timing

2) SDR Data Write Transfer

1. When Transaction is issued from the I3C Master, it compares the Slave Address of Address Header with its own Slave Address, and if it matches, I3C responds with ACK.
   When a Transaction is received, if the Receive Data Buffer is full, the I3C Slave will respond with NACK in the Address Header.
   In preparation for retrying the I3C Master, read the data from the Receive Data Buffer via the NTDTBPn register, and empty the Receive Data Buffer.
2. Data received from I3C Master is stored in the Receive Data Buffer.
3. With the RDBFF0 = 1 interrupt, the received data is read from the Receive Data Buffer via the NTDTBPn register.
4. When Repeated START condition or STOP condition is detected, the Receive Status Descriptor is stored in the Receive Status Buffer.
5. Read Receive Status Descriptor via NRSQP register and check the status.

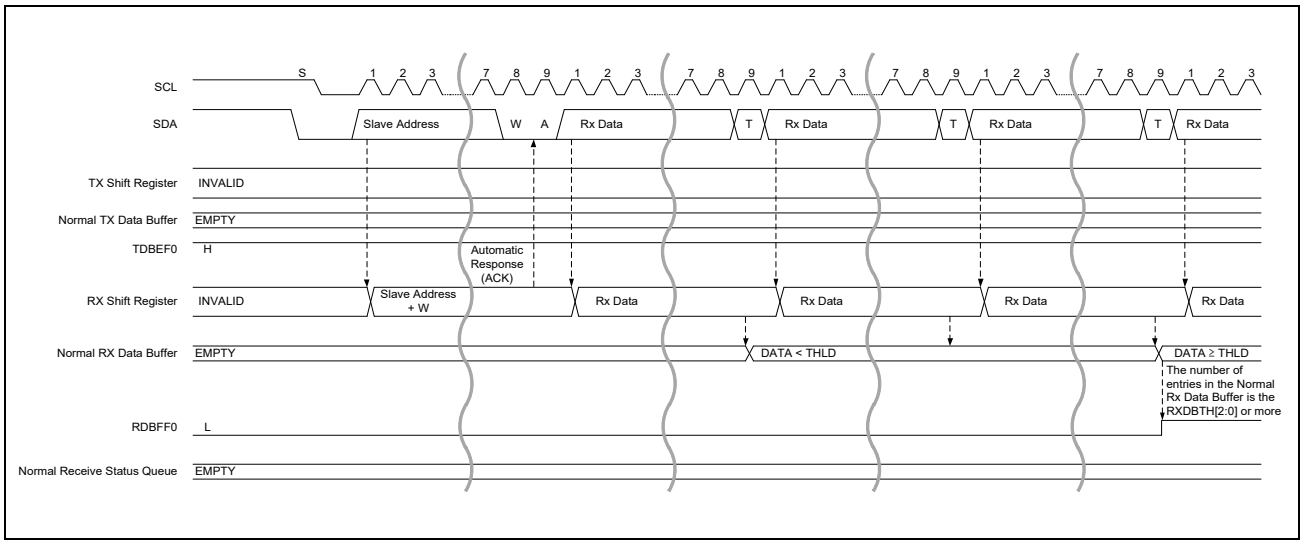


Figure 26.37 SDR Data Write Transfer Timing (1/2)

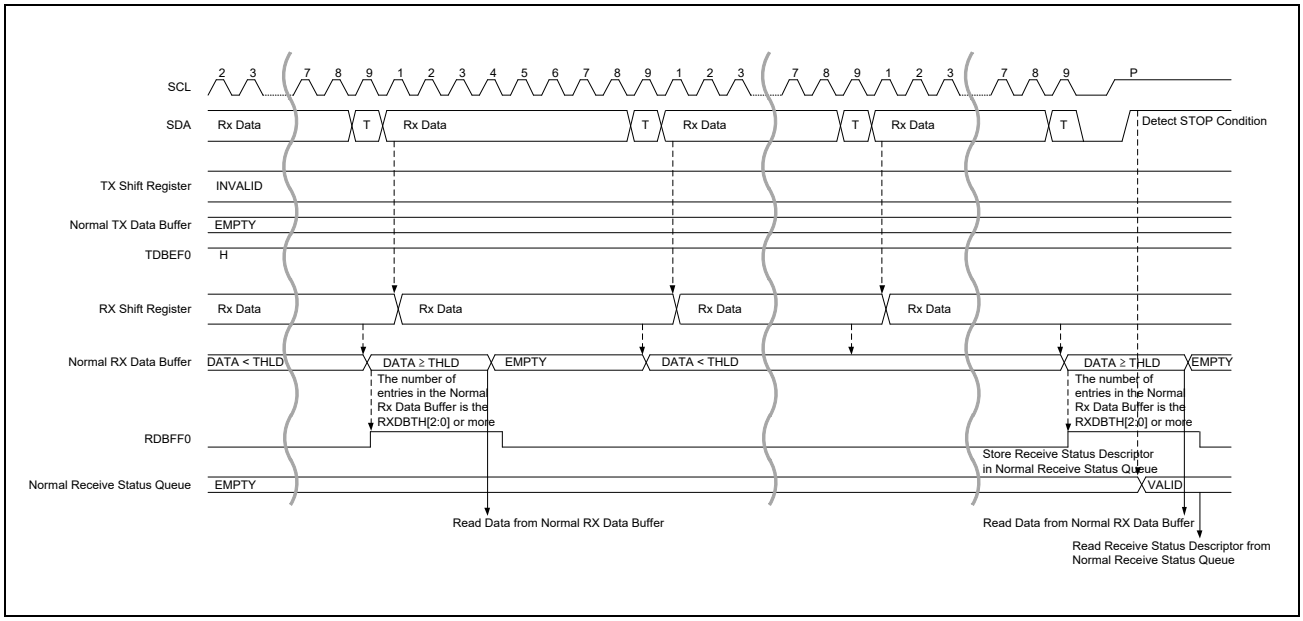


Figure 26.38 SDR Data Write Transfer Timing (2/2)

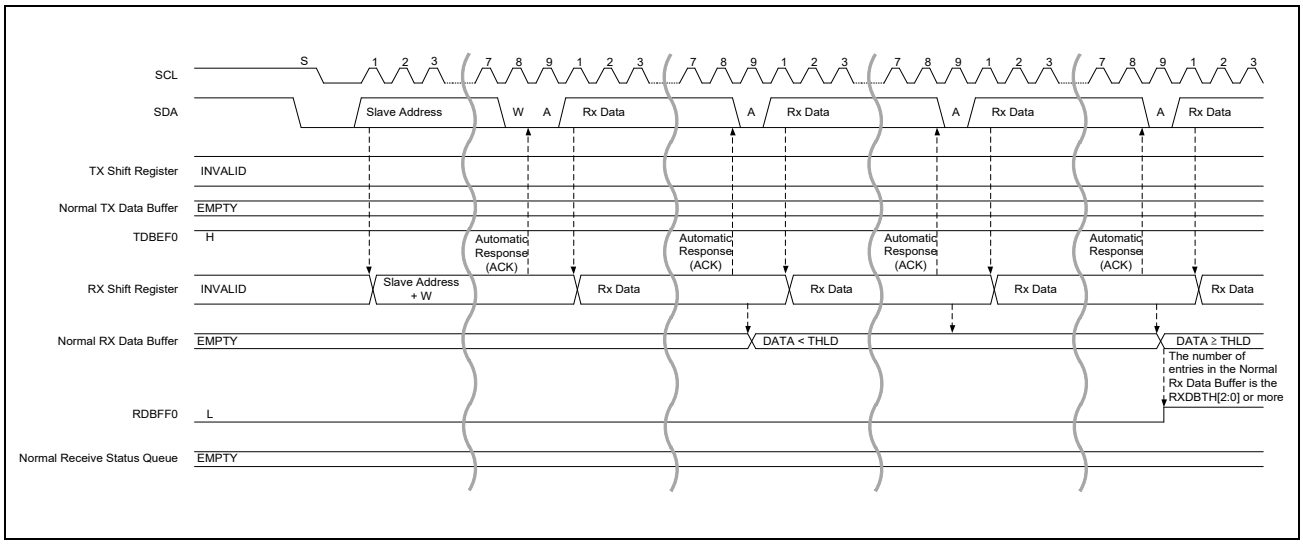


Figure 26.39 Legacy I²C Message Data Write Transfer Timing (1/2)

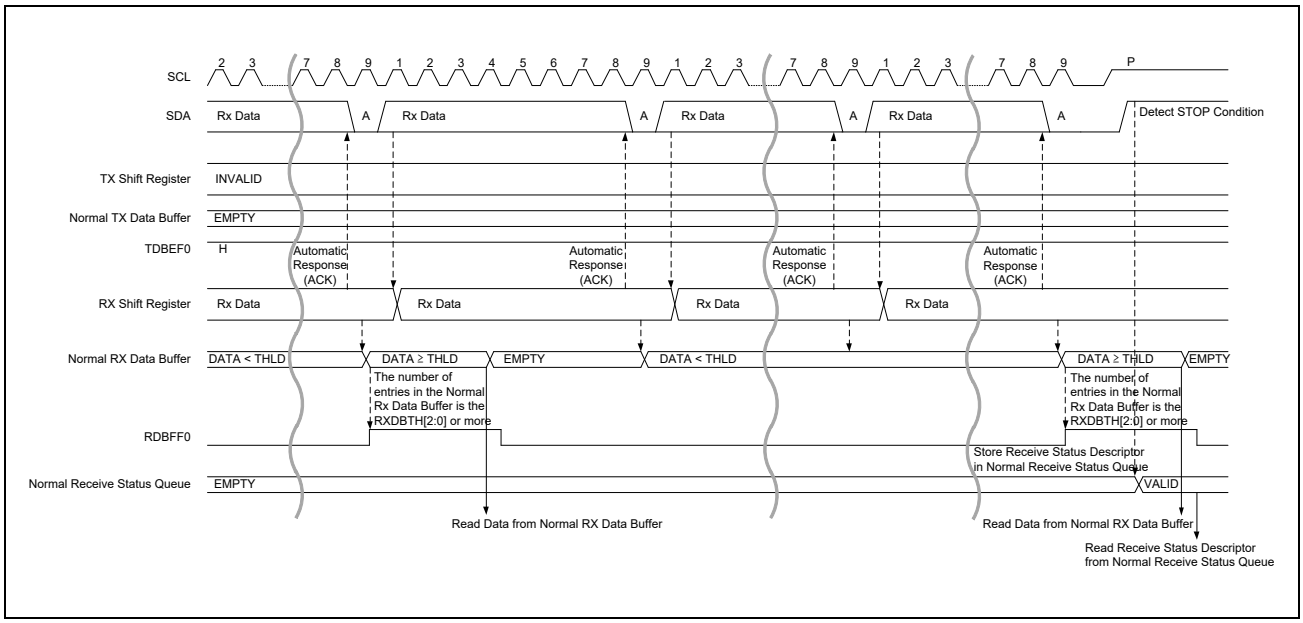


Figure 26.40 Legacy I²C Message Data Write Transfer Timing (2/2)

3) SDR Data Read Transfer

1. Write the data requested from the I3C Master to the Transmit Data Buffer via the NTDTBPn register.
2. When Transaction is issued from the I3C Master, it compares the Slave Address of Address Header with its own Slave Address, and if it matches, I3C responds with ACK.
   When a Transaction is received, if the Transmit Data Buffer is EMPTY, I3C Slave responds with NACK with the Address Header.
   In preparation for retrying the I3C Master, write data to the Transmit Data Buffer via the NTDTBPn register.
3. Transmit the data stored in the Transmit Data Buffer.
4. If data to be transmitted still remains, write the data to be transmitted with an interrupt by TDBEF0 = 1 to the Transmit Data Buffer via the NTDTBPn register.
5. SDR:
   When the transmission of the data stored in the Transmit Data Buffer is completed, Low is output to the T-bit following Data, and it is notified to the I3C Master that it is the final data.
   Legacy I^2C Message:
   When NACK is detected, data transmission is terminated.
6. When a Repeated START condition or STOP condition is detected, the Receive Status Descriptor is stored in the Receive Status Buffer.
7. Read the Receive Status Descriptor via NRSQP and check the status.
   If the data length does not match, set the RSTCTL.INTLRST bit to 1 and then reset the internal states of this module. For details, refer to **Section 26.3.2.4(6), Error Recovery Operation [I3C mode]**.

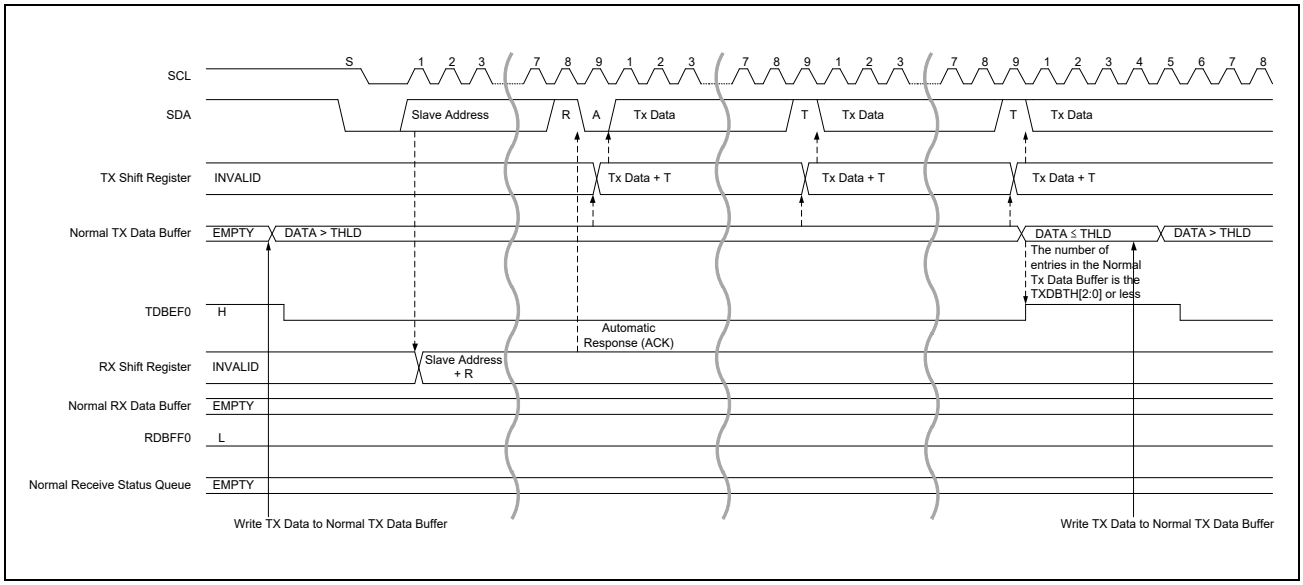


Figure 26.41 SDR Data Read Transfer Timing (1/2)

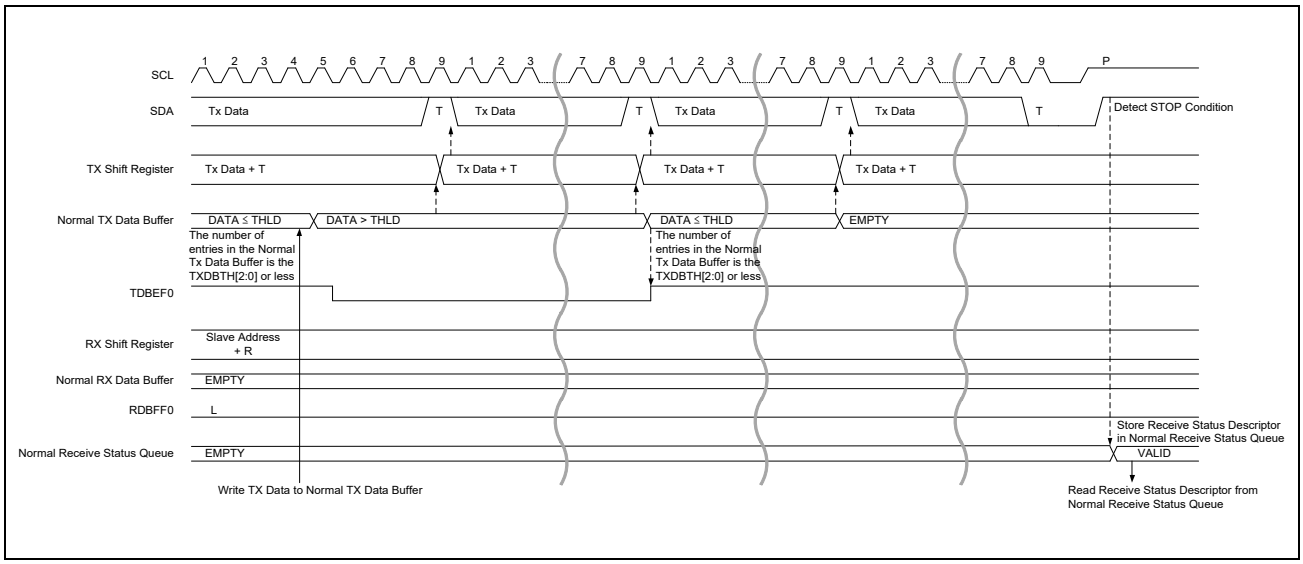


Figure 26.42 SDR Data Read Transfer Timing (2/2)

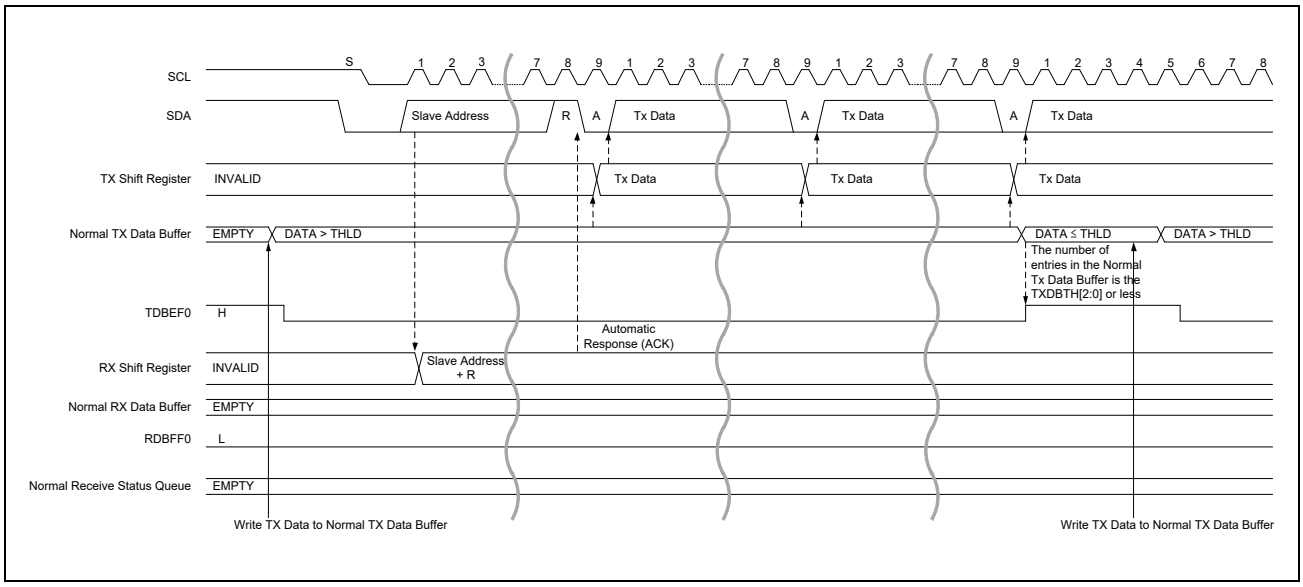


Figure 26.43 Legacy I²C Message Data Read Transfer Timing (1/2)

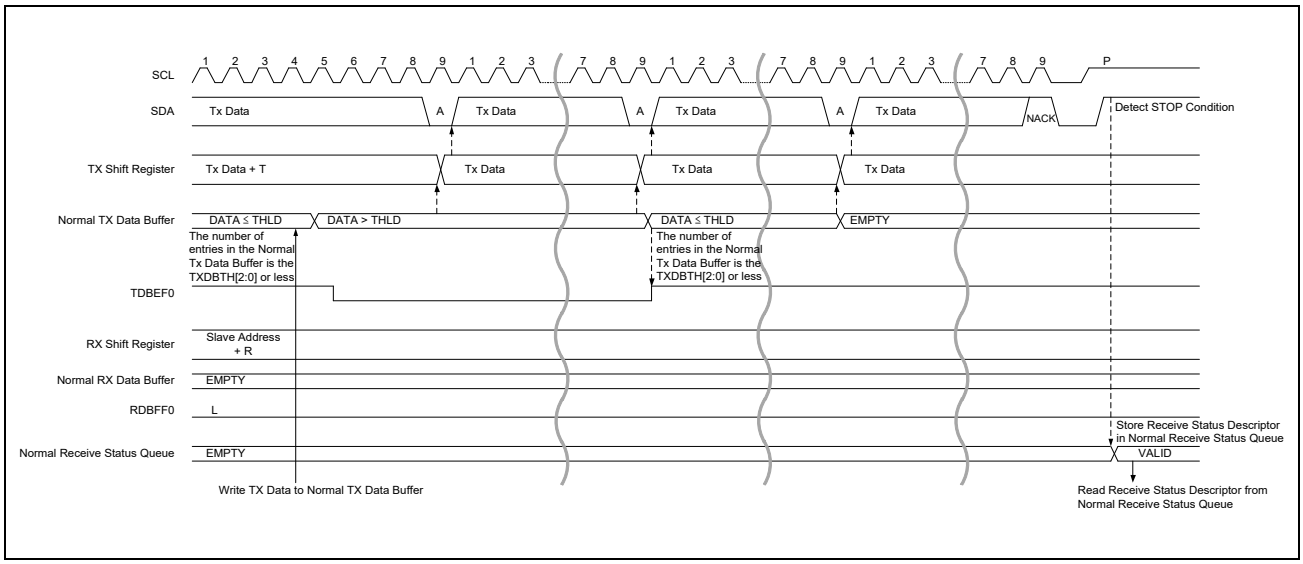


Figure 26.44 Legacy I²C Message Data Read Transfer Timing (2/2)

4) IBI Transfer

1. When sending Slave Interrupt Request.
   When transmitting IBI Data, write IBI Data to the IBI Data Buffer via the NIBIQP register.
2. Write Command Descriptor (Immediate Transfer Command or Regular Transfer Command) to the Command Buffer for IBI Transfer via the NCMDQP register.
3. When Command Descriptor is written to Command Buffer, IBI Transaction is issued under the following conditions.
   - Detect a START condition. (Does not apply a Repeated START condition)
   - If no START is forthcoming within the following Bus Condition, then this module issue a START Request by pulling the SDA line Low.
     a) Slave Interrupt Request, Mastership Request: Bus Available
     b) Hot-Join Event: Bus Idle
4. In Slave Address with RnW of the Address Header, if losing Arbitration by issuing a Transaction from I3C Master, stop issuing Transaction.
   When detecting Repeated START condition or STOP condition, store the Response Descriptor into the Response Buffer.
5. When sending Slave Interrupt Request:
   - When IBI data for transmission still remain, write IBI data with an interrupt by IBIQEFF = 1 to the IBI Data Buffer via the NIBIQP register.
   - When the transmission of IBI Data for the number of Data Length specified by the DATA_LENGTH[15:0] bits of the Command Descriptor is completed, output Low to the T-bit following IBI Data and notify the I3C Master that it is the final IBI Data.
6. When detecting Repeated START condition or STOP condition, store the Response Descriptor into the Response Buffer.
7. Read the Response Descriptor Form the Response Buffer with the NRSPQP register and check the status. If NACK is responded, repeat steps 1 to 7.
8. When sending Slave Interrupt Request:
   Check that the value of the DATA_LENGTH[15:0] bit of the Response Descriptor is 0.

The Mastership processing flow is shown in **Figure 26.47**. When joining the I3C Bus by Hot-Join after the I3C Bus has already been configured, issue the Hot-Join according to the flow shown in **Figure 26.131**.

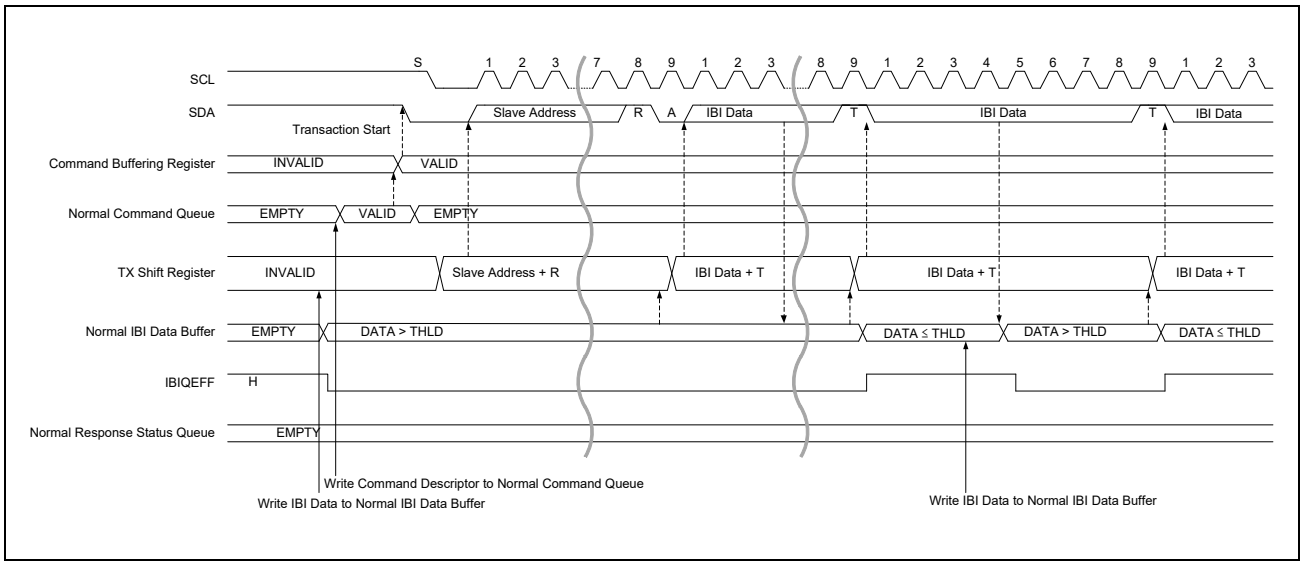


Figure 26.45 I3C Slave IBI Transfer Timing (1/2)

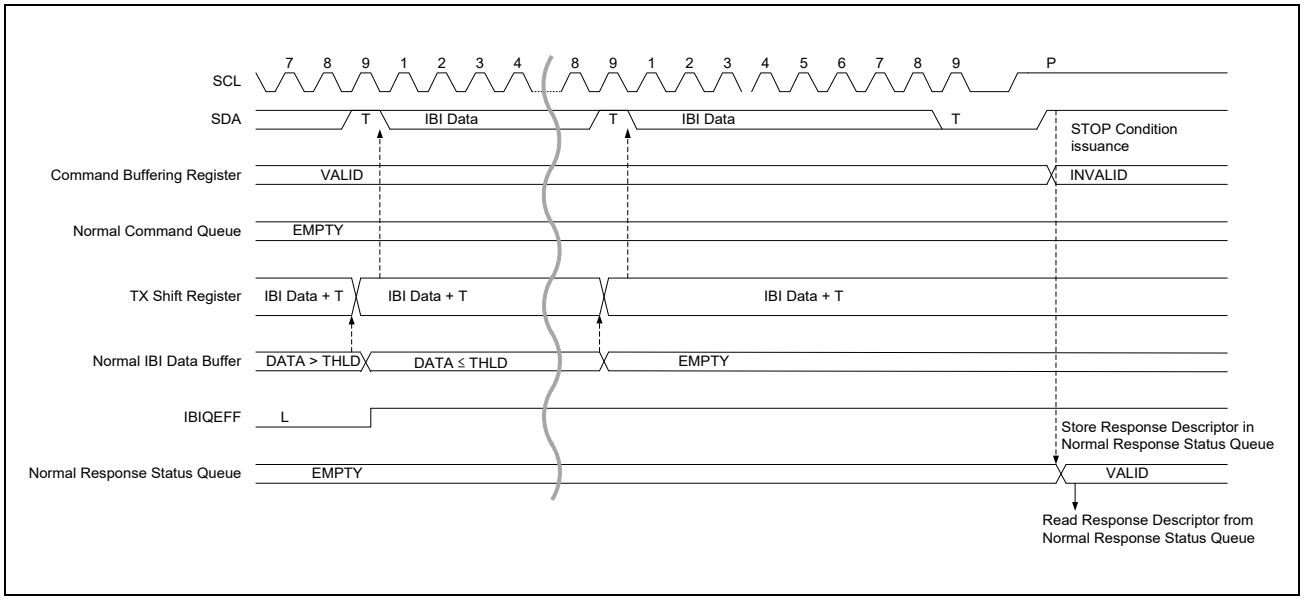


Figure 26.46 I3C Slave IBI Transfer Timing (2/2)

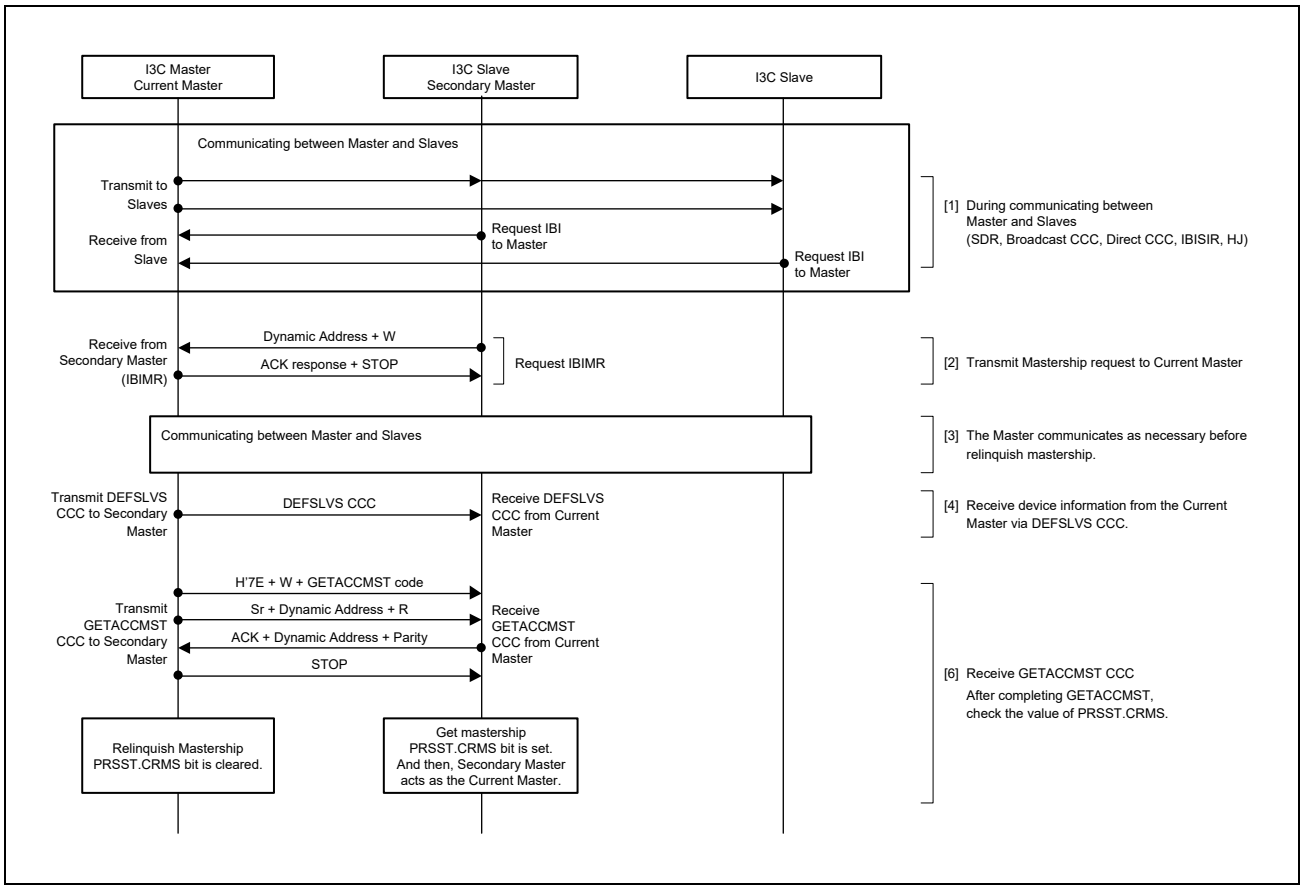


Figure 26.47 I3C Slave Mastership Processing Flow

26.3.2.2 Data Handler

The relationship between the transfer method and the queue is shown in **Table 26.11**.

Table 26.11 Transfer Method and Queue

Protocol	Transfer Method	Queue/Buffer	Size	Master	Slave	Secondary Master
I^2C Mode	Single buffer transfer	Normal Transmit Data	1 byte	✓	✓	—
		Normal Receive Data	1 byte	✓	✓	—
I3C Mode	Normal FIFO buffer transfer	Normal Command	8 QUEUEs	✓	✓	✓
		Normal Response Status	8 QUEUEs	✓	✓	✓
		Normal Transmit Data	32 DWORDs	✓	✓	✓
		Normal Receive Data	32 DWORDs	✓	✓	✓
		Normal Receive Status	8 QUEUEs	—	✓	✓
		Normal IBI Status	4 QUEUEs	✓	—	✓
		Normal IBI Data	8 DWORDs	✓	✓	✓

(1) Transfer Method in I²C Mode

(a) Single Buffer Transfer

Each process (condition issue, data transfer, ACK/NACK response) is controlled by software.

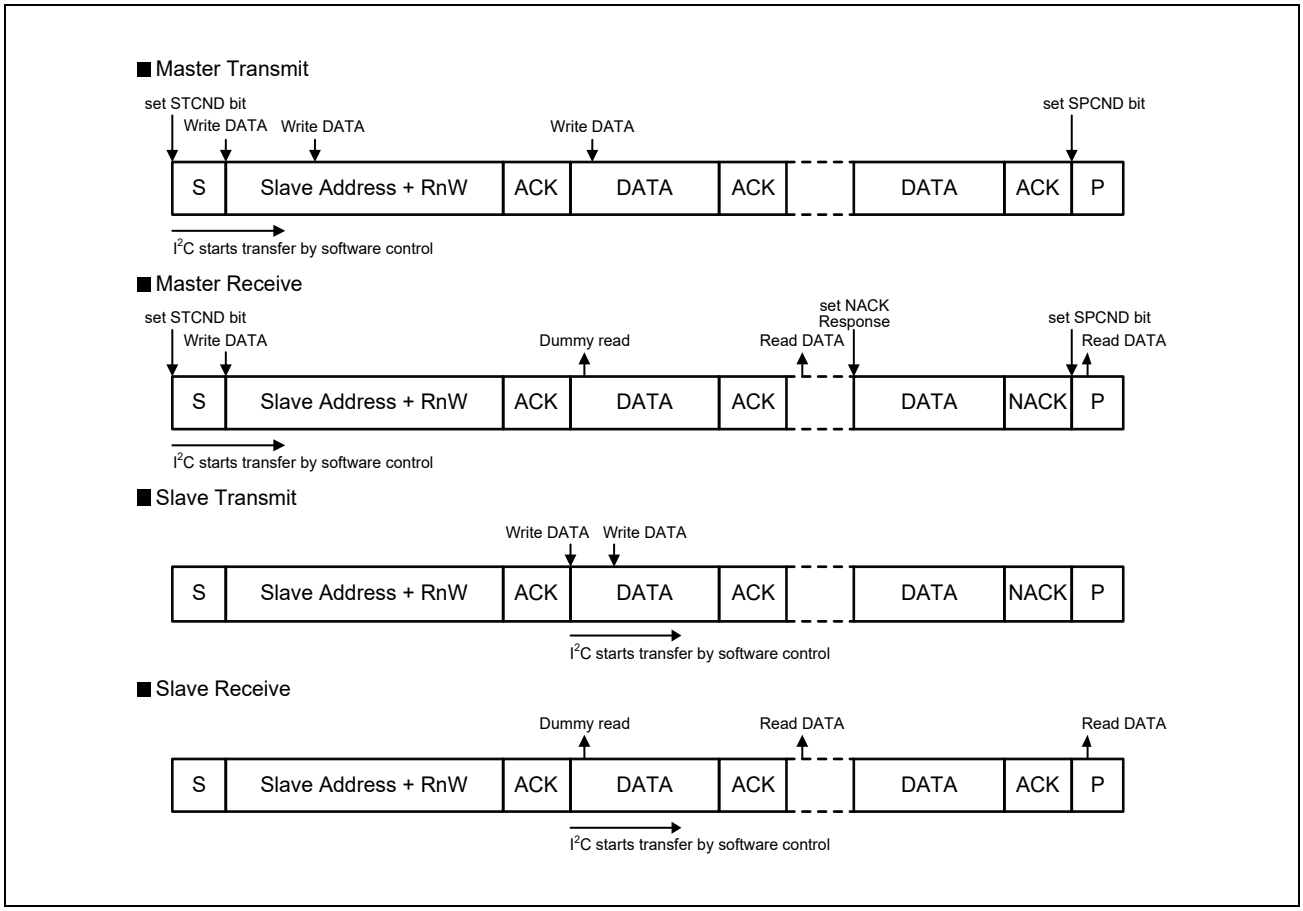


Figure 26.48 Data Handler with Single Buffer Transfer

(2) Transfer Method in I3C Mode

(a) Normal FIFO Buffer Transfer

I3C autonomously starts transfer when data and command are written.

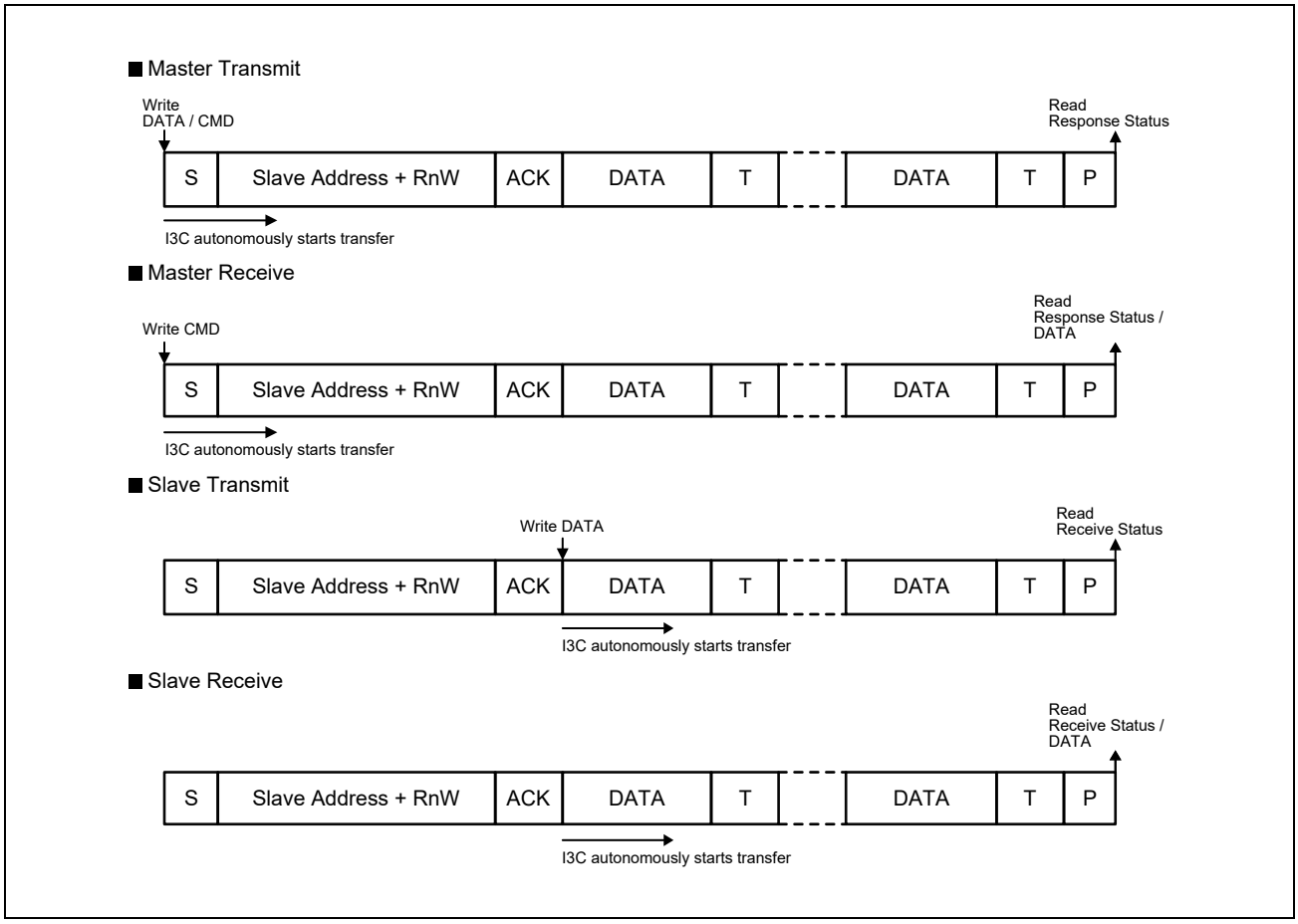


Figure 26.49 Data Handler with Normal FIFO Buffer Transfer

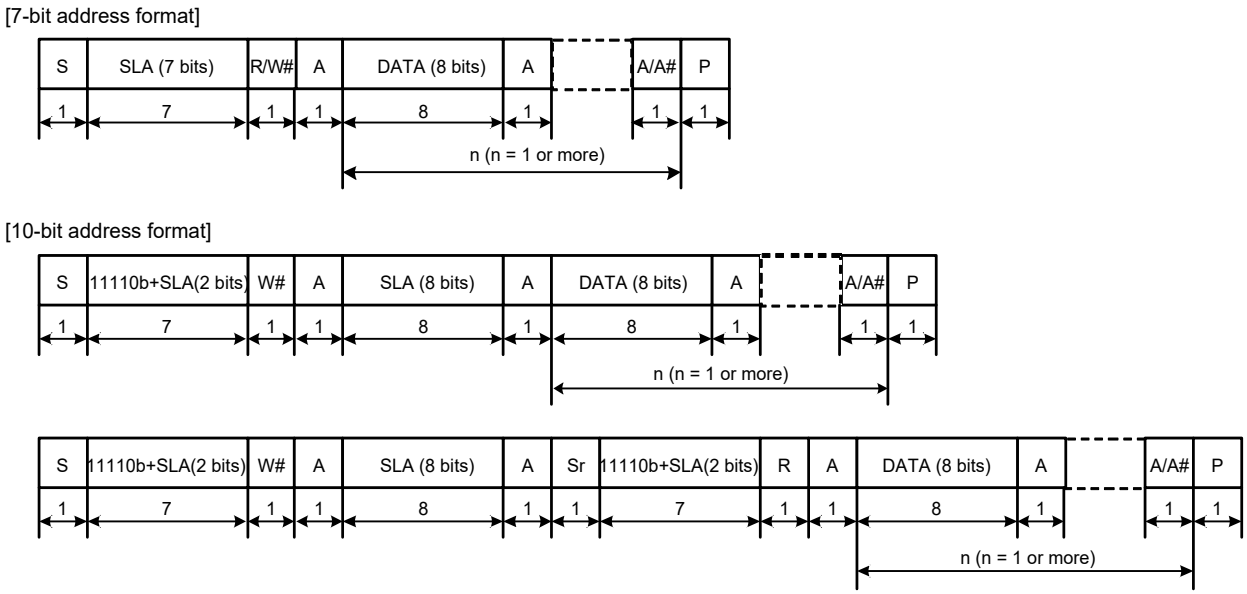
26.3.2.3 I²C/I3C Protocol

(1) Communication Protocol

(a) I²C Communication Data Format

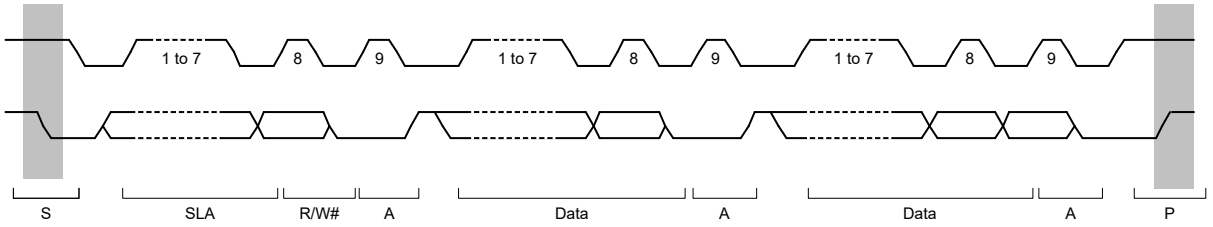
The I²C bus format consists of 8-bit data and 1-bit acknowledge. The frame following a START condition or Repeated START condition is an address frame used to specify a slave device with which the master device communicates. The specified slave is valid until a new slave is specified or a STOP condition is issued.

Figure 26.50 shows the I²C bus format, and **Figure 26.51** shows the I²C bus timing.



Note: n: Number of transmit DATA bytes

Figure 26.50 I²C Bus Format



S: START condition. The master device drives the SDAn line low from high level while the SCLn line is at a high level.

SLA: Slave address, by which the master device selects a slave device.

R/W#: Indicates the direction of data transfer: from the slave device to the master device when R/W = 1, or from the master device to the slave device when R/W = 0.

A: Acknowledge. The receive device drives the SDAn line low. (In master transmit mode, the slave device returns acknowledge. In master receive mode, the master device returns acknowledge.)

A#: Not Acknowledge. The receive device drives the SDAn line high.

Sr: Repeated START condition. The master device drives the SDAn line low from the high level after the setup time has elapsed with the SCLn line at the high level.

DATA: Transmitted or received data

P: STOP condition. The master device drives the SDAn line high from low level while the SCLn line is at a high level.

Figure 26.51 I^2C Bus Timing (SLA = 7 bits)

(b) I3C Communication Data Format

Figure 26.52 through illustrate a typical communication for each of the six I3C Protocols. While these diagrams do not exhaustively illustrate all possible I3C communications, they do serve as useful introductions to the signaling and transmission formatting used in each I3C Protocol.

Figure 26.52illustrates example communication using I3C Single Data Rate (SDR) coding with Broadcast (H'7E). It shows the Master reading a byte of data from the Slave at Address H'2B in SDR Mode. From the Bus Free Condition, the Master issues a START by driving the SDA line Low while keeping the SCL line High. It then issues the Broadcast Address (H'7E) followed by RnW (0 for Write). Then the Master turns on a pull-up resistor and goes to Open Drain.

All Slaves ACK by pulling the SDA line Low (in the Figure, pink fill means the Slave is in control of the SDA line at this time). The Master then issues a Repeated START, then the Address of the Slave (H'2B) it wants to read followed by RnW (1 for Read). The Master then turns on a pull-up resistor and goes to Open Drain, allowing the Slave to acknowledge by pulling the SDA line Low. At this point, the Master continues to toggle the SCL line and release the SDA line, allowing the Slave to drive SDA to send one byte of data (H'4A) followed by T. T = 1 informs the Master that there is additional data, whereas T = 0 signals the end. Here there is additional data, so the Slave drives SDA High until SCL goes High, at which time it releases SDA. The Master has the option of holding SDA High with a weak pullup, which signals to the Slave that the Master allows another byte to be transmitted, or to pull SDA Low (while SCL is High – hence a Repeated START), which would signal to the Slave that the Master has terminated the Read and is taking over. SDR Mode is backwards compatible with Legacy I^2C Devices, because the High time of an SCL pulse is always less than 50ns and therefore SCL will always appear to be Low because of the I^2C 50ns Spike Filter.

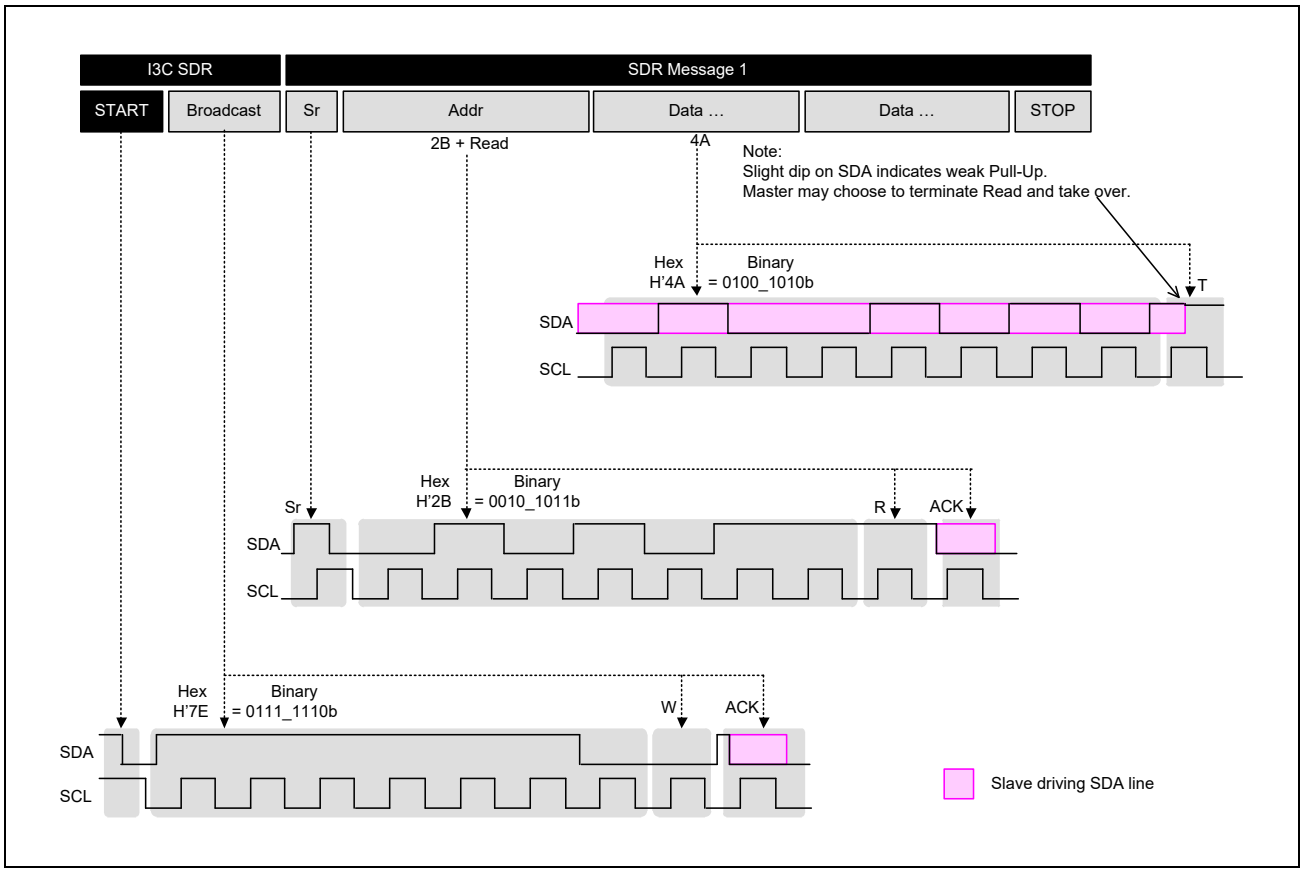


Figure 26.52 Example Communication Using I3C Coding SDR with Broadcast (H'7E)

Figure 26.53 illustrates example communication using I3C Single Data Rate (SDR) coding without Broadcast (H'7E). It shows the Master reading a byte of data from the Slave at Address H'2B in SDR Mode. From the Bus Free Condition, The Master then issues a START, then the Address of the Slave (H'2B) it wants to read followed by RnW (1 for Read).

The Master then turns on a pull-up resistor and goes to Open Drain, allowing the Slave to acknowledge by pulling the SDA line Low. At this point, the Master continues to toggle the SCL line and release the SDA line, allowing the Slave to drive SDA to send one byte of data (H'4A) followed by T. T = 1 informs the Master that there is additional data, whereas T = 0 signals the end. Here there is additional data, so the Slave drives SDA High until SCL goes High, at which time it releases SDA. The Master has the option of holding SDA High with a weak pull-up, which signals to the Slave that the Master allows another byte to be transmitted, or to pull SDA Low (while SCL is High – hence a Repeated START), which would signal to the Slave that the Master has terminated the Read and is taking over. SDR Mode is backwards compatible with Legacy I^2C Devices, because the High time of an SCL pulse is always less than 50ns and therefore SCL will always appear to be Low because of the I^2C 50ns Spike Filter.

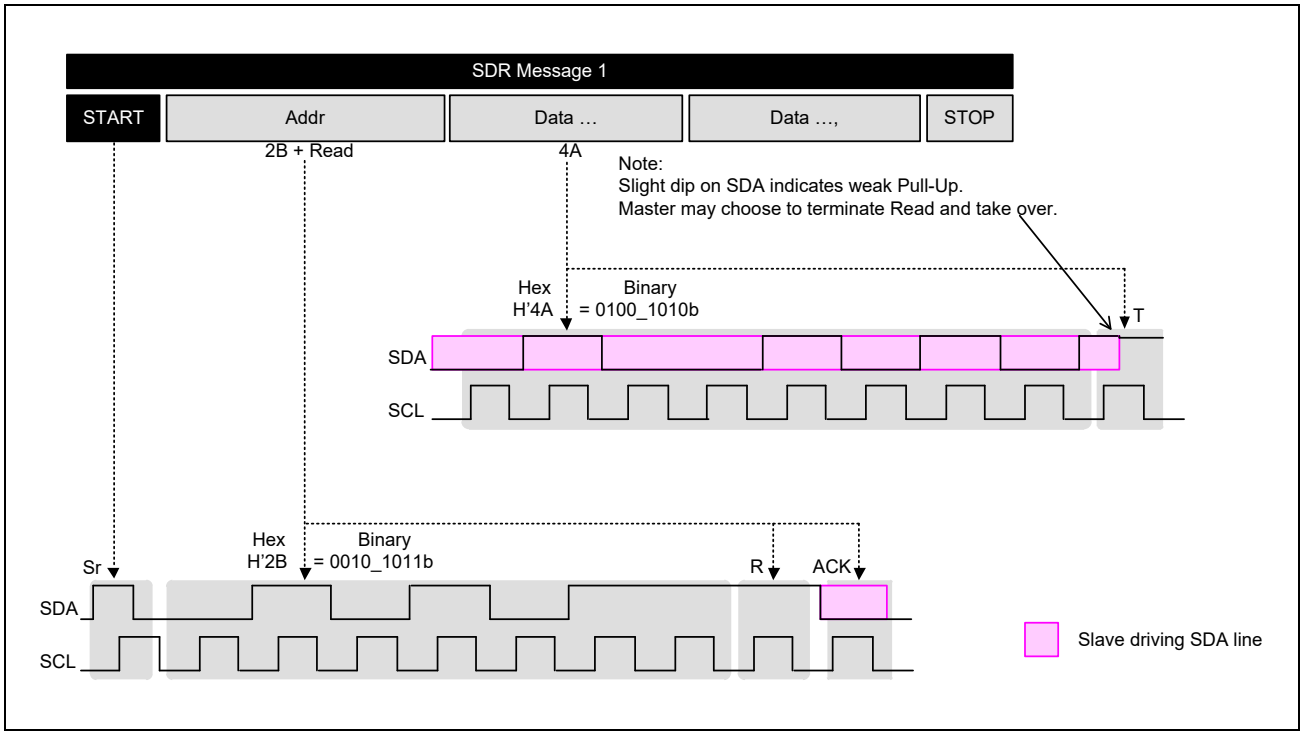


Figure 26.53 Example communication using I3C coding SDR without broadcast (H'7E)

Figure 26.54 shows the Master issuing a CCC Direct Command to a single Slave. This particular command (GETPID) reads the Provisional ID of a Slave.

From the Bus Free Condition, the Master issues a START by driving the SDA line Low while keeping the SCL line High. It then issues the Broadcast Address (H'7E) followed by RnW (0 for Write). Then the Master turns on a pull-up resistor and goes to Open Drain. All Slaves ACK by pulling SDA Low (in the Figure, pink fill means the Slaves are in control of SDA at this time). The Master then issues the Direct Common Command Code for GETPID (H'8C) followed by parity bit T (odd parity = 0 for H'8C) then the 7-bit Dynamic Address of the Slave (chosen arbitrarily here to be H'2B) followed by a RnW bit (1 for Read). Then the Master turns on a pull-up resistor and goes to Open Drain, allowing the Slave at Address H'2B to ACK by pulling SDA Low, which tells the Master that the Slave Acknowledges the command and will comply. (Alternatively, the Slave may NACK by not pulling SDA Low, which would inform the Master that the Slave will not comply – in this case, that an error occurred.) Following the ACK, the Slave outputs its 48-bit PID one byte at a time, and then the Master issues a Repeated START (this part of the waveform sequence is not shown in the Figure).

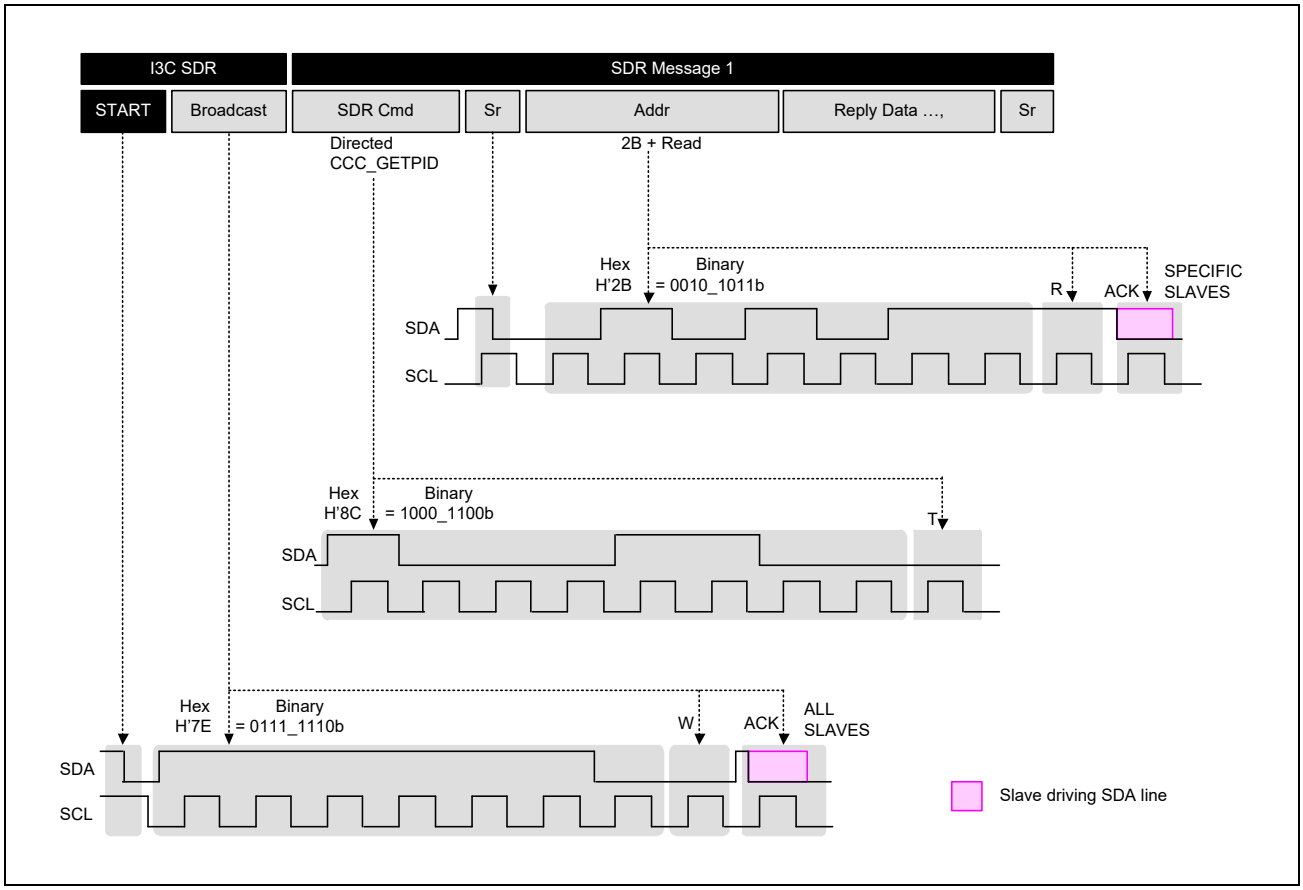


Figure 26.54 Example Communication Using I3C Coding SDR with CCC Direct Addressing

Figure 26.55 illustrates example SDR communication with a CCC Broadcast command. The command used in this example sets the Maximum Read Length of all Slaves to 43 bytes (H'2B).

From the Bus Free Condition, the Master issues a START by driving the SDA line Low while keeping the SCL line High. It then issues the Broadcast Address (H'7E) followed by RnW (0 for Write). Then the Master turns on a pull-up resistor and goes to Open Drain. All Slaves ACK by pulling SDA Low (in the Figure, pink fill means the Slaves are in control of SDA at this time). The Master then issues the Broadcast Common Command Code for SETMRL (H'09) followed by parity bit T (odd parity = 1 for H'09), and then 2 data bytes (MSB first) to define the maximum number of bytes which can be read from a Slave in a single read operation. Each data byte is followed by a T bit (parity bit – odd parity). After this the Master issues a Repeated START.

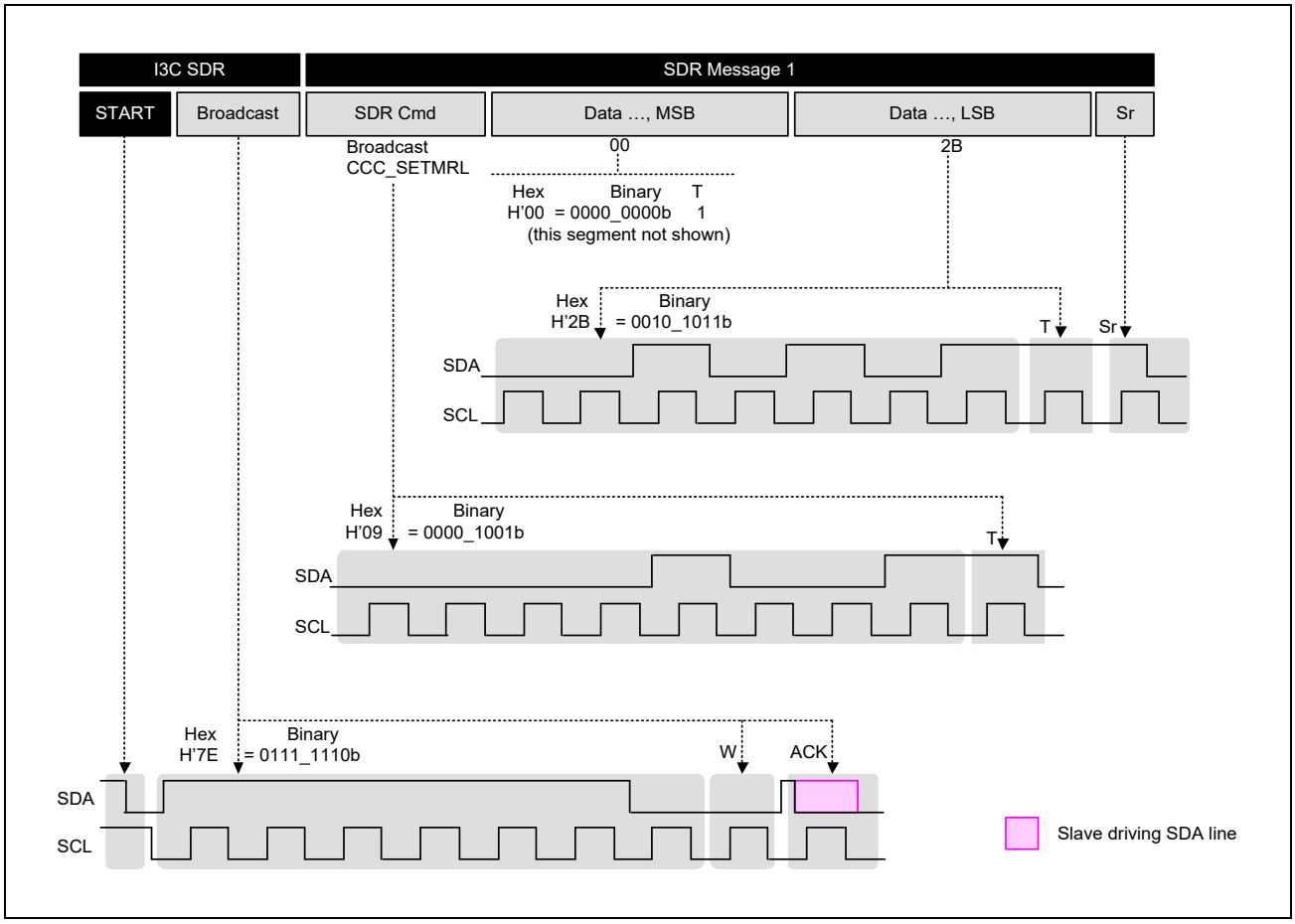


Figure 26.55 Example Communication Using I3C Coding SDR with CCC Broadcast

(2) Bus Conditions

I3C defines three distinct conditions in which the I3C Bus shall be considered inactive: Bus Free, Bus Available, and Bus Idle (refer to **Figure 26.56**).

(a) Bus Free Condition

State on the I3C Bus where both the SCL line and the SDA line are High for at least the period set by BFRECDT.FRECYC[8:0] bit.

(b) Bus Available Condition [I3C mode]

State on the I3C Bus where both the SCL line and the SDA line are High for at least the period set by BAVLCDT.AVLCYC[8:0] bit.

A Slave may only issue a START Request (For example, for an In-Band Interrupt, or for a Master Handoff Request) after a Bus Available Condition.

(c) Bus Idle Condition [I3C mode]

State on the I3C Bus where both the SCL line and the SDA line are High for at least the period set by BIDLCDT.IDLCYC[17:0] bit.

A Slave may only issue a START Request (For example, for a Hot-Join) after a Bus Idle Condition. Specifications are as follows. IDLE needs to be the largest.

BFRECDT.FRECYC[8:0] < BAVLCDT.AVLCYC[8:0] < BIDLCDT.IDLCYC[17:0]

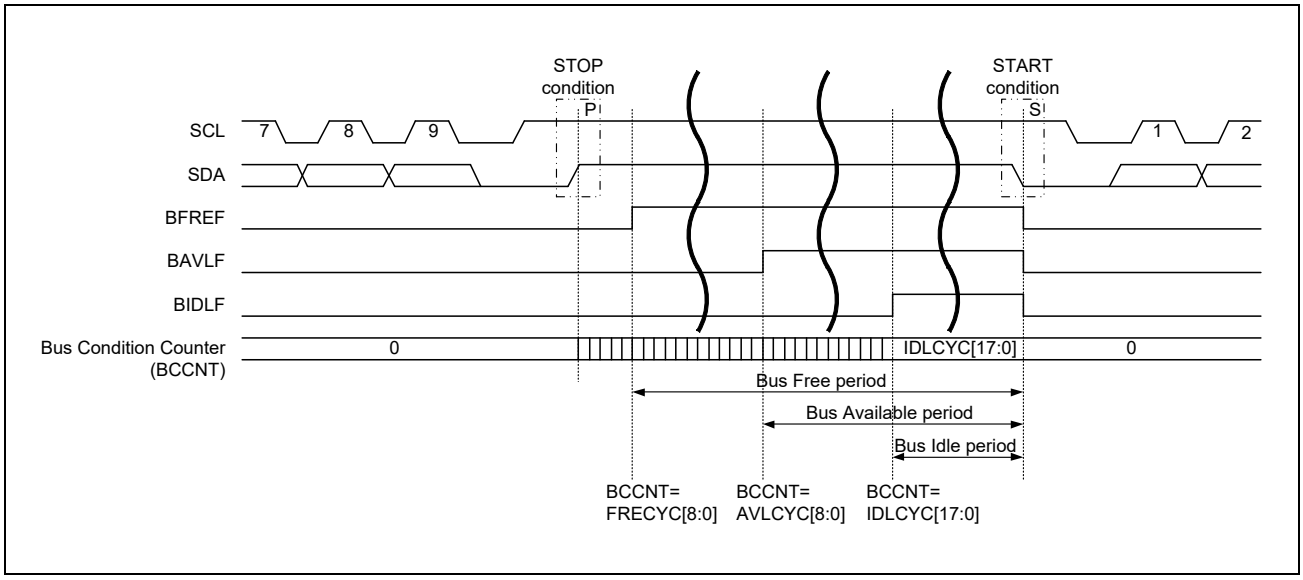


Figure 26.56 Bus Conditions

(3) START Condition / Repeated START Condition / STOP Condition Issuing Function

(a) Issuing a START Condition

I3C issues a START condition when the CNDCTL.STCND bit is set to 1.

Set the STCND bit to 1 (START condition issuance request) when the BCST.BFREF flag is set to 1 (bus free state).

I3C issues a START condition.

When a START condition is issued normally, I3C automatically shifts to the master transmit mode. A START condition is issued in the following sequence.

[START condition issuance]

- Drive the SDAn line low (high level to low level).
- Ensure the time set in STDBR.SBRHO[7:0] and the START condition hold time.
- Drive the SCLn line low (high level to low level).
- Detect low level of the SCLn line and ensure the low-level period of SCLn line set in STDBR.SBRLO[7:0].

(b) Issuing a Repeated START Condition

I3C issues a Repeated START condition when the CNDCTL.SRCND bit is set to 1.

When the SRCND bit is set to 1, a Repeated START condition issuance request is made and I3C issues a Repeated START condition when the BCST.BFREF flag = 0 (bus busy state) and the PRSST.CRMS bit = 1 (master mode).

A Repeated START condition is issued in the following sequence. [Repeated START condition issuance]

- Release the SDAn line.
- Ensure the low-level period of SCLn line set in STDBR.SBRLO[7:0].
- Release the SCLn line (low level to high level).
- Detect a high level of the SCLn line and ensure the time set in STDBR.SBRLO[7:0] and the Repeated START condition setup time.
- Drive the SDAn line low (high level to low level).
- Ensure the time set in STDBR.SBRHO[7:0] and the Repeated START condition hold time.
- Drive the SCLn line low (high level to low level).
- Detect a low level of the SCLn line and ensure the low-level period of SCLn line set in STDBR.SBRLO[7:0].

NOTE

When issuing Repeated START conditions request, please write the slave address to NTDTBP0 after confirming CNDCTL.SRCND = 0. Data written in the period of CNDCTL.SRCND = 1 is not forwarded because retransmission condition before the occurrence.

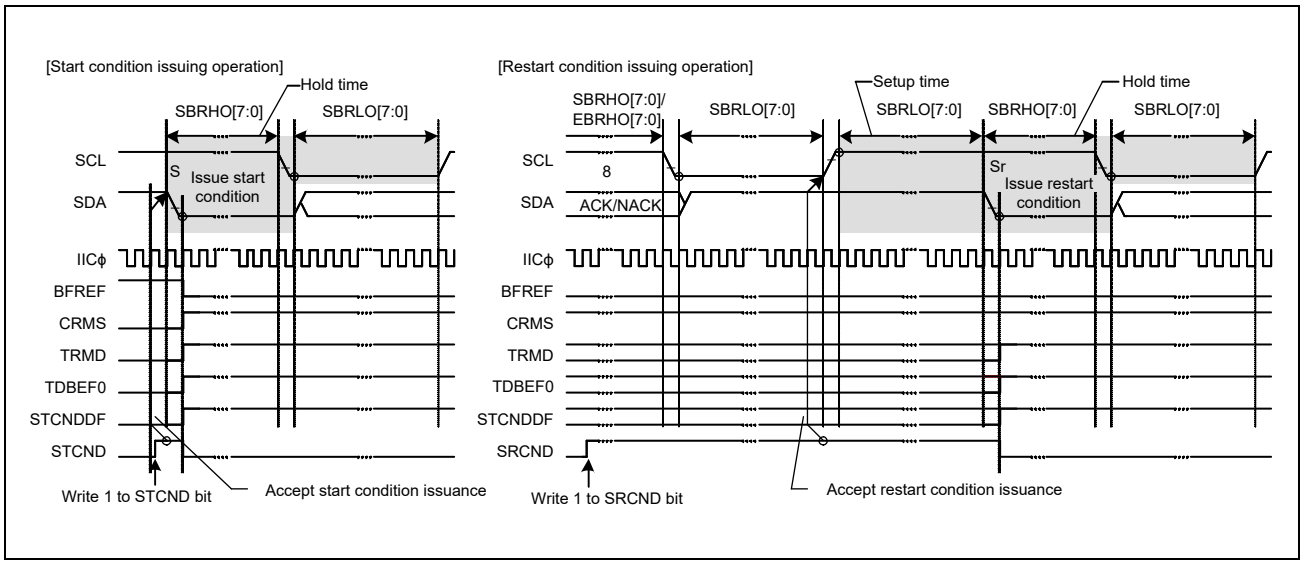


Figure 26.57 START Condition / Repeated START Condition Issue Timing (STCND and SRCND bits)

Figure 26.58 shows the operation to issue a Repeated START condition after the master transmission. [Repeated START condition issuance after the master transmission]

- Initial setting. For details, refer to **Section 26.3.3.1, Initial Setting Flow**.
- Read the BFREF flag in BCST to check that the bus is open, and then set the STCND bit in CNDCTL to 1 (START condition issuance request). Upon receiving the request, I3C issues a START condition. At the same time, the BFREF flag is automatically set to 0 and the STCNDDF flag in BST is automatically set to 1 and the STCND bit is automatically set to 0. At this time, if the START condition is detected and the internal levels for the SDA output state and the levels on the SDAn line have matched while the STCND bit = 1, I3C recognizes that issuing of the START condition as requested by the STCND bit has been successfully completed, and CRMS and TRMD bits in PRSST is automatically set to 1, placing I3C in master transmit mode. The NTST.TDBEF0 flag is also automatically set to 1 in response to setting of the TRMD bit to 1.
- Check that the NTST.TDBEF0 flag = 1, and then write the value for transmission (the slave address and the R/W# bit) to NTDTBP0. Once the data for transmission are written to NTDTBP0, the TDBEF0 flag is automatically set to 0, the data are transferred from NTDTBP0, and the TDBEF0 flag is again set to 1. After the byte containing the slave address and R/W# bit has been transmitted, the value of the TRMD bit is automatically updated to select master transmit or master receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 0, I3C continues in master transmit mode. Since the BST.NACKDF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to CNDCTL.SPCND bit to issue a STOP condition. For data transmission with an address in the 10-bit format, start by writing 1_1110, the 2 higher-order bits of the slave address, and W to NTDTBP0 as the first address transmission. Then, as the second address transmission, write the 8 lower-order bits of the slave address to NTDTBP0.
- After confirming that the NTST.TDBEF0 flag = 1, write the data for transmission to the NTDTBP0 register. I3C automatically holds the SCLn line low until the data for transmission are ready, a Repeated START condition is issued or a STOP condition is issued.
- After all bytes of data for transmission have been written to the NTDTBP0 register, wait until the value of the BST.TENDF flag returns to 1, and then, after check that the BST.STCNDDF flag = 1, set the BST.STCNDDF flag to 0.
- Set the SRCND bit in CNDCTL to 1 (Repeated START condition issuance request). Upon receiving the request, I3C issues a Repeated START condition.

- After check that the BST.STCNDDF flag = 1, write the value for transmission (the slave address and the R/W# bit) to NTDTBP0.

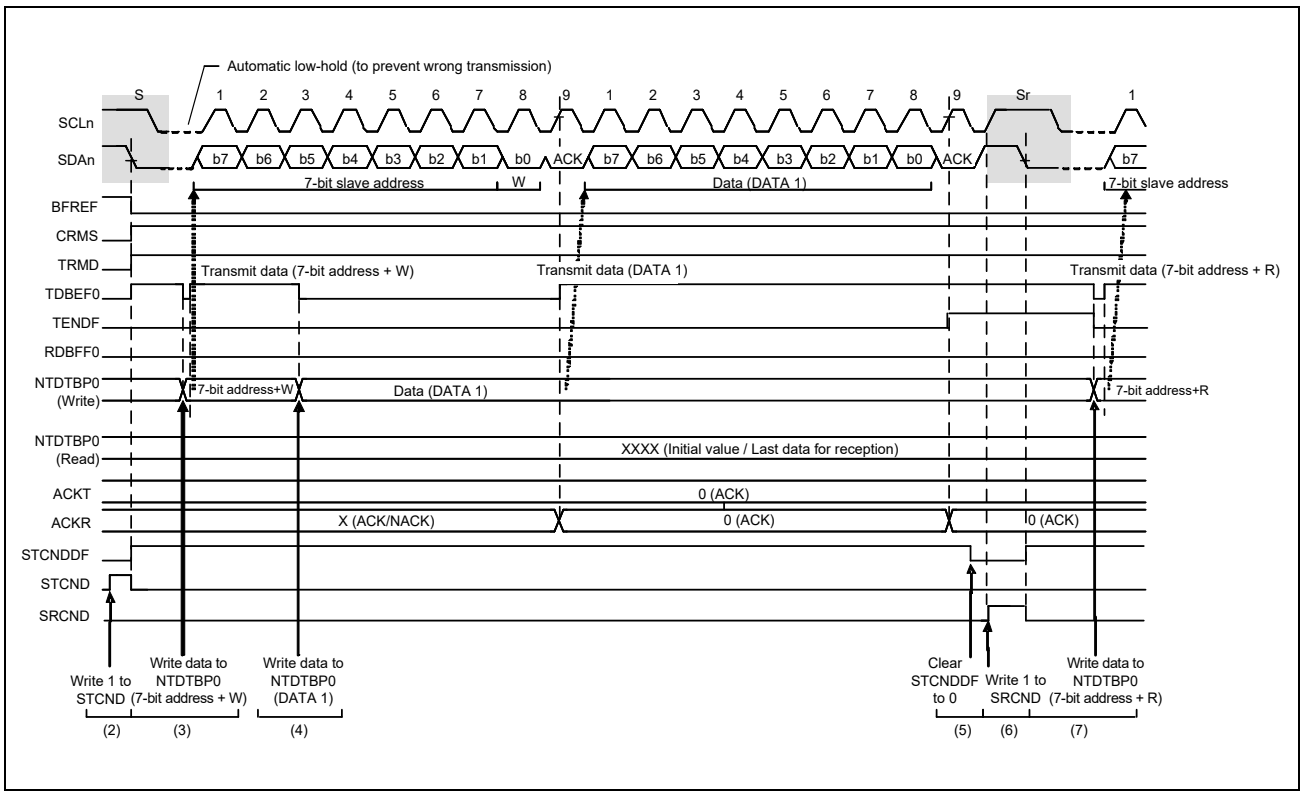


Figure 26.58 Repeated START Condition Issuance after the Master Transmission Timing

(c) Issuing a STOP Condition

I3C issues a STOP condition when the SPCND bit in CNDCTL is set to 1.

When the SPCND bit is set to 1, a STOP condition issuance request is made and I3C issues a STOP condition when the BCST.BFREF flag = 0 (bus busy state) and the PRSST.MST bit = 1 (master mode).

A STOP condition is issued in the following sequence.

[STOP condition issuance]

- Drive the SDAn line low (high level to low level).
- Ensure the low-level period of SCLn line set in STDBR.SBRLO[7:0].
- Release the SCLn line (low level to high level).
- Detect a high level of the SCLn line and ensure the time set in STDBR.SBRHO[7:0] and the STOP condition setup time.
- Release the SDAn line (low level to high level).
- Ensure the time set in STDBR.SBRLO[7:0] and the bus free time.
- Set the BFREF flag to 1 (to release the bus mastership).

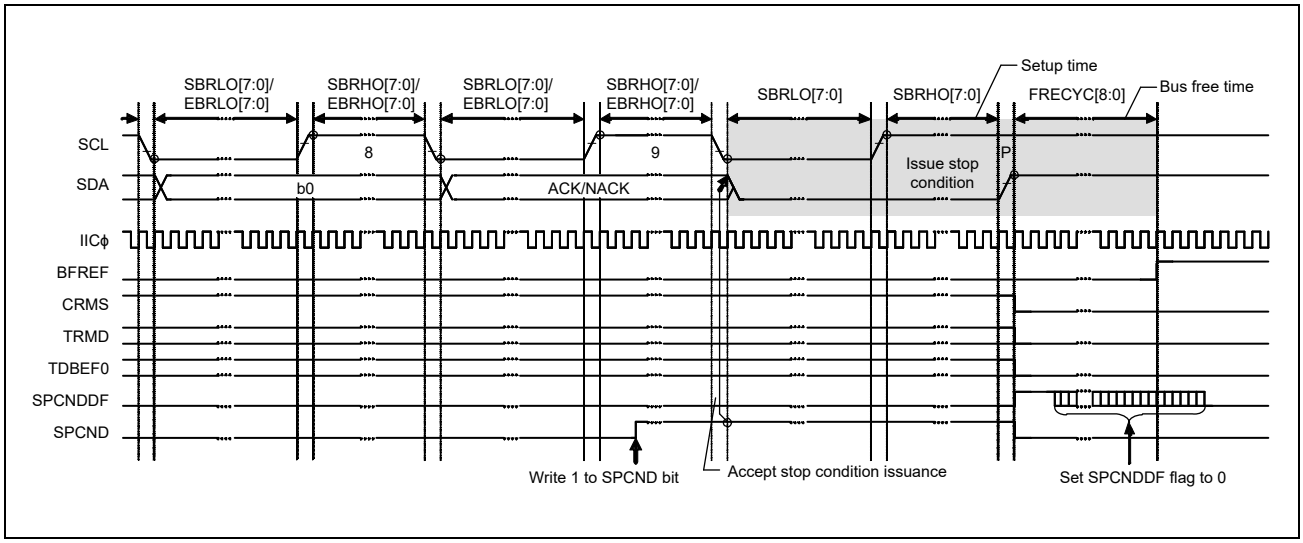


Figure 26.59 STOP Condition Issue Timing (SPCND bit)

(4) Address Match Detection

I3C can set three unique slave addresses in addition to the general call address and host address, and also can set 7- bit or 10- bit slave addresses.

(a) Slave-Address Match Detection [I²C mode]

I3C can set three unique slave addresses and has a slave address detection function for each unique slave address.

When the SVCTL.SVAEy bit (y = 0 to 2) is set to 1, the slave addresses set in the SVDVADy register (y = 0) can be detected.

When I3C detects a match of the set slave address, the corresponding SVST.SVAFy flag (y = 0 to 2) is set to 1 at the rising edge of the ninth SCL clock cycle, and the NTST.RDBFF0 flag or the NTST.TDBEF0 flag is set to 1 by the following R/W# bit. This causes a receive data full interrupt (I3Cn_RX) or transmit data empty interrupt (I3Cn_TX) to be generated. The SVAFy flag is used to identify which slave address has been specified.

Figure 26.60 to **Figure 26.62** show the SVAFy flag set timing in three cases.

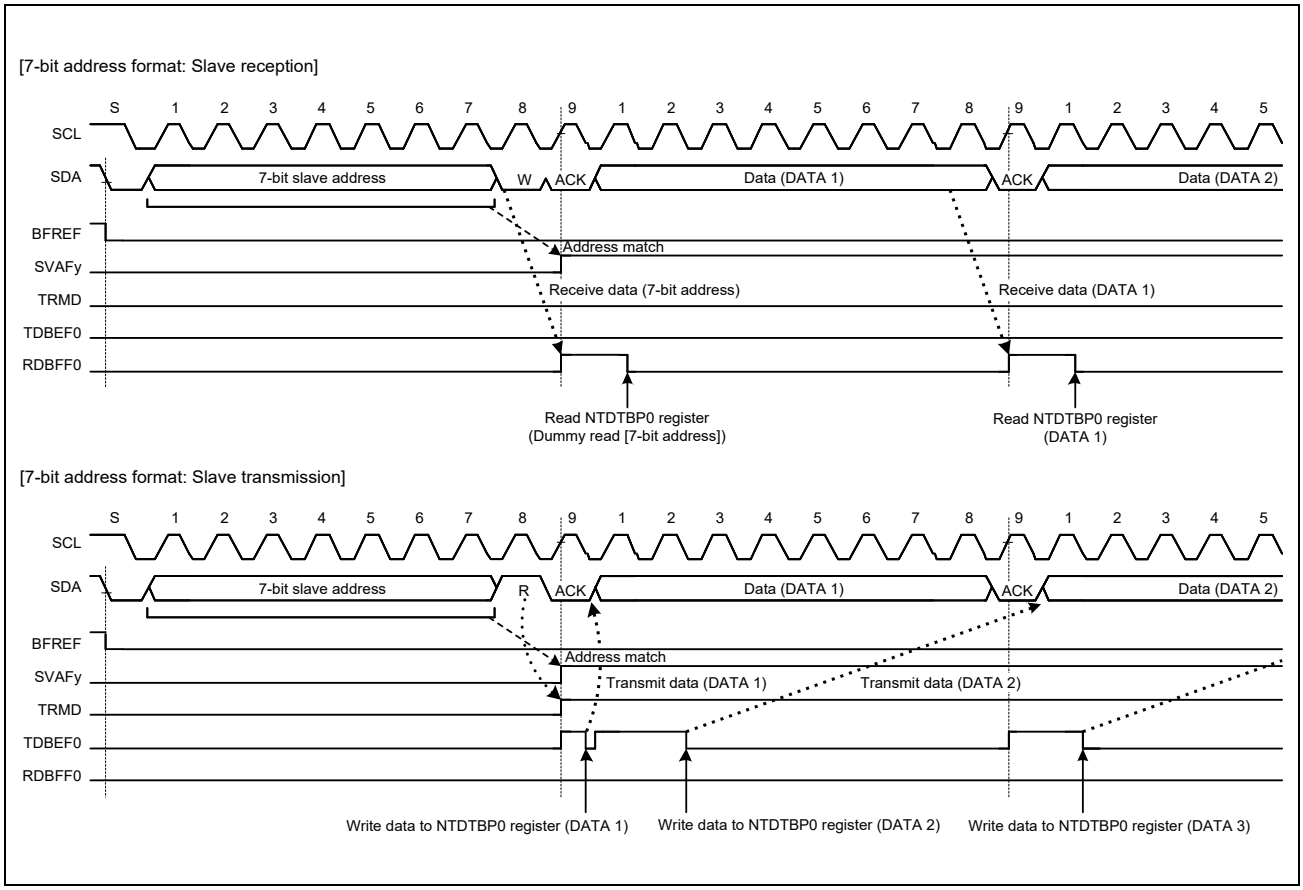


Figure 26.60 SVAFy Flag Set Timing with 7-bit Address Format Selected

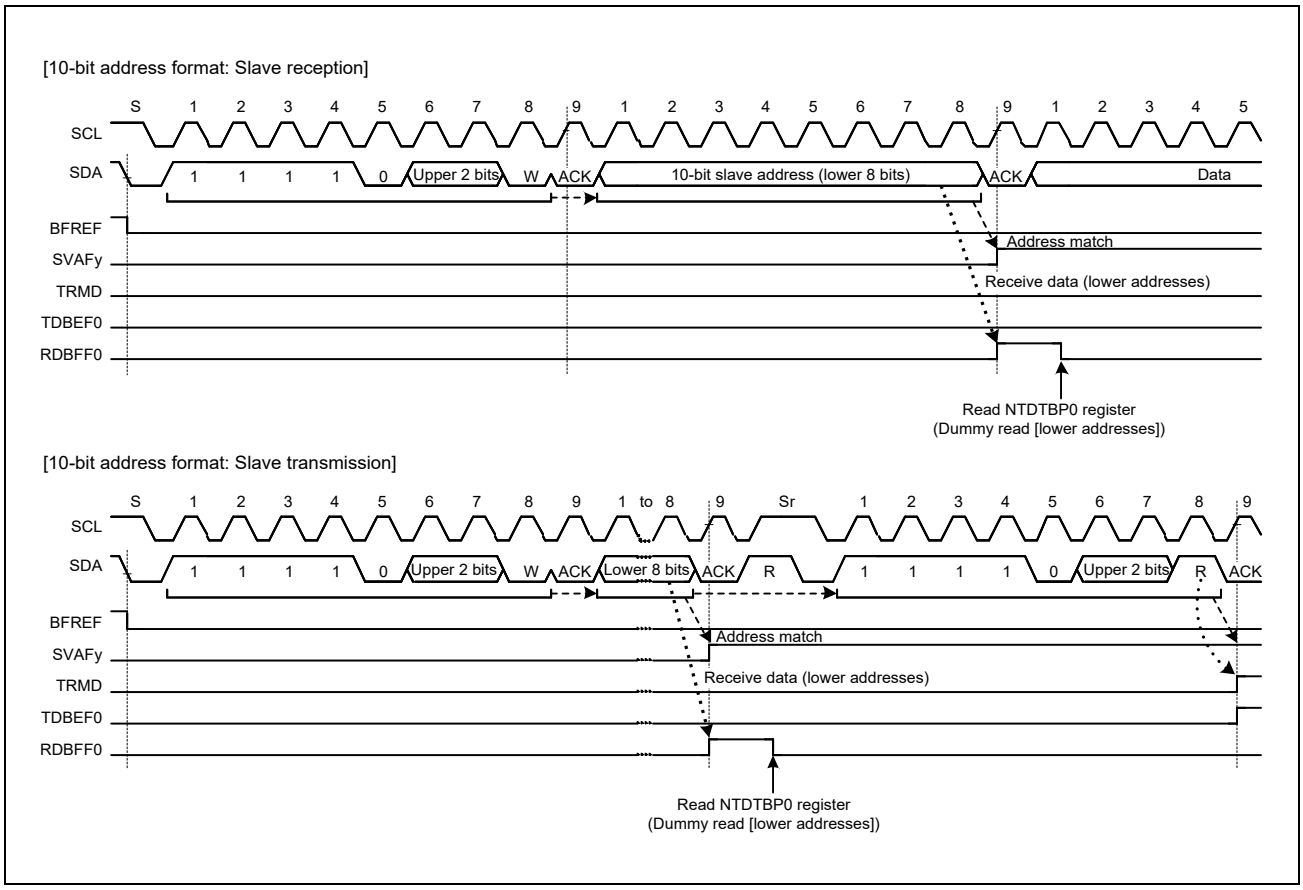


Figure 26.61 SVAFy Flag Set Timing with 10-bit Address Format Selected

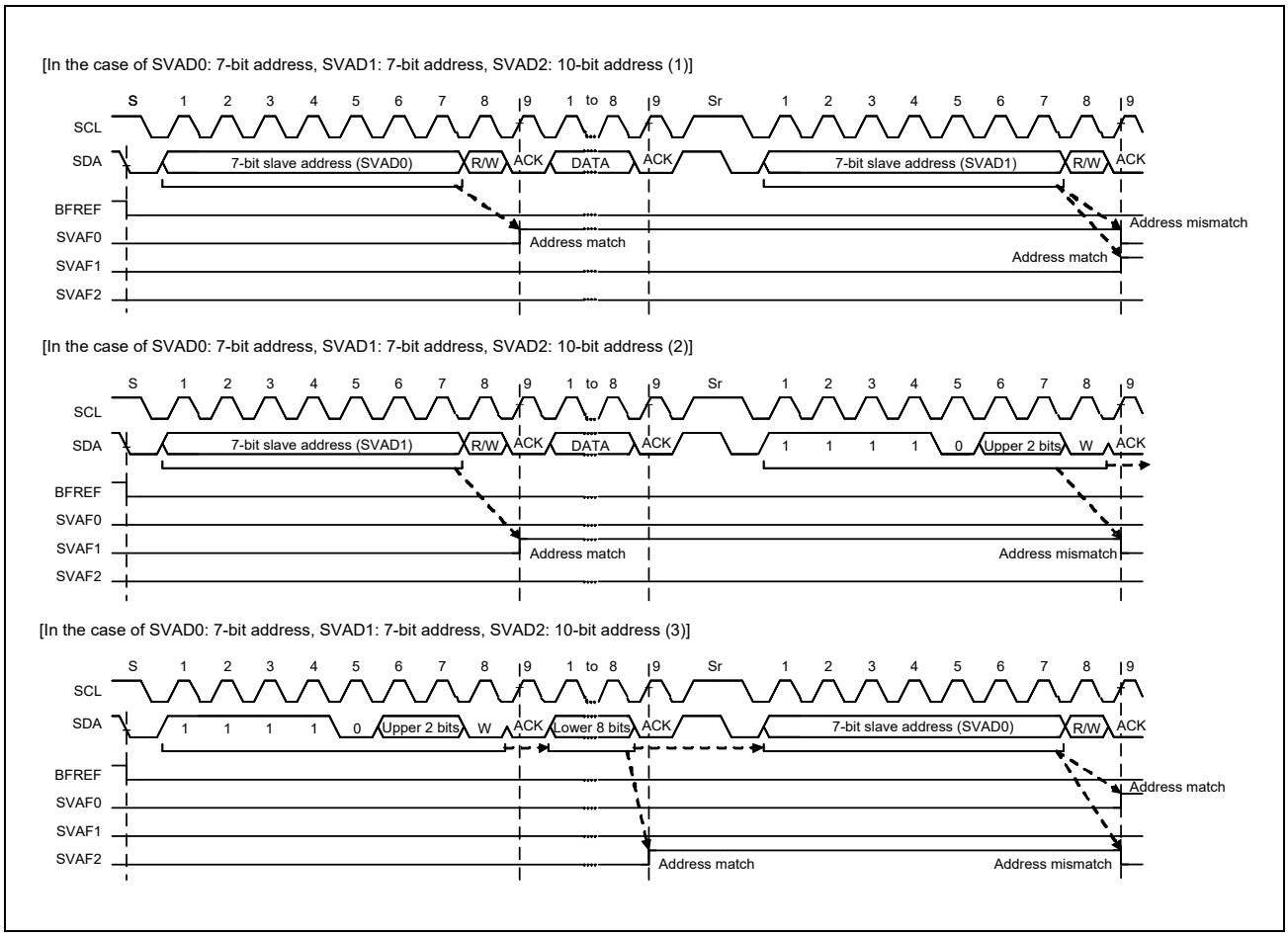


Figure 26.62 SVAFy Flag Set / Clear Timing with 7-bit/10-bit Address Formats Mixed

(b) Detection of the General Call Address [I^2C mode]

I3C has a facility for detecting the general call address (000_0000 + 0 (write)). This is enabled by setting the SVCTL.GCAE bit to 1.

If the address received after a START or Repeated START condition is issued is 000_0000 + 1 (read) (start byte), I3C recognizes this as the address of a slave device with an all-zero address but not as the general call address.

When I3C detects the general call address, both the SVST.GCAF flag and the NTST.RDBFF0 flag are set to 1 on the rising edge of the ninth cycle of SCL clock. This leads to the generation of a receive data full interrupt (I3Cn_RX). The value of the GCAF flag can be confirmed to recognize that the general call address has been transmitted.

Operation after detection of the general call address is the same as normal slave receive operation.

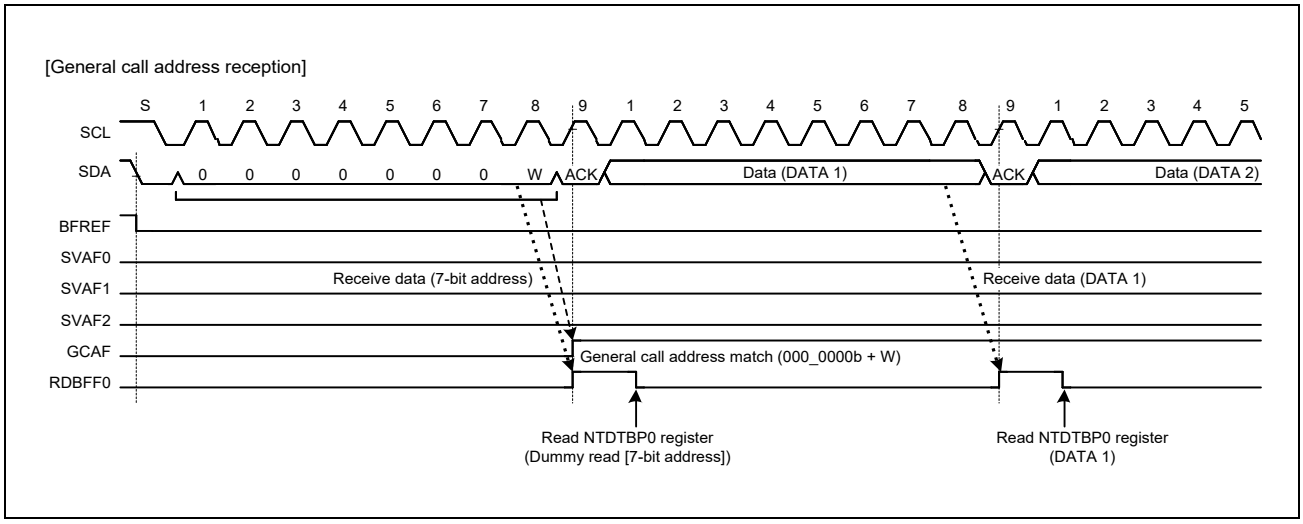


Figure 26.63 Timing of GCAF Flag Setting during Reception of General Call Address

(c) Device-ID Address Detection [I²C mode]

I3C module has a facility for detecting device-ID addresses conformant with the I²C-bus specification (Rev.03). When I3C receives 111_1100 as the first byte after a START condition or Repeated START condition was issued with the SVCTL.DVIDE bit set to 1, I3C recognizes the address as a device ID, sets the SVST.DVIDF flag to 1 on the rising edge of the ninth SCL clock cycle when the following R/W# bit = 0, and then compares the second and subsequent bytes with its own slave address. If the address matches the value in the slave address register, I3C sets the corresponding SVST.SVAFy flag (y = 0 to 2) to 1.

After that, when the first byte received after a START or Repeated START condition is issued matches the device ID address (111_1100) again and the following R/W# bit = 1, I3C does not compare the second and subsequent bytes and sets the NTST.TDBEF0 flag to 1.

In the device-ID address detection function, I3C sets the DVIDF flag to 0 if a match with I3C's own slave address is not obtained or a match with the device ID address is not obtained after a match with I3C's own slave address and the detection of a Repeated START condition. If the first byte after detection of a START or Repeated START condition matches the device ID address (111_1100) and the R/W# bit = 0, I3C sets the DVIDF flag to 1 and compares the second and subsequent bytes with I3C's slave address. If the R/W# bit = 1, the DVIDF flag holds the previous value and I3C does not compare the second and subsequent bytes. Therefore, the reception of a device-ID address can be checked by reading the DVIDF flag after confirming that TDBEF0 flag = 1.

Furthermore, prepare the device-ID fields (3 bytes: 12 bits indicating the manufacturer + 9 bits identifying the part + 3 bits indicating the revision) that must be sent to the host after reception of a continuous device-ID field as normal data for transmission. For details of the information that must be included in device-ID fields, contact NXP Semiconductors.

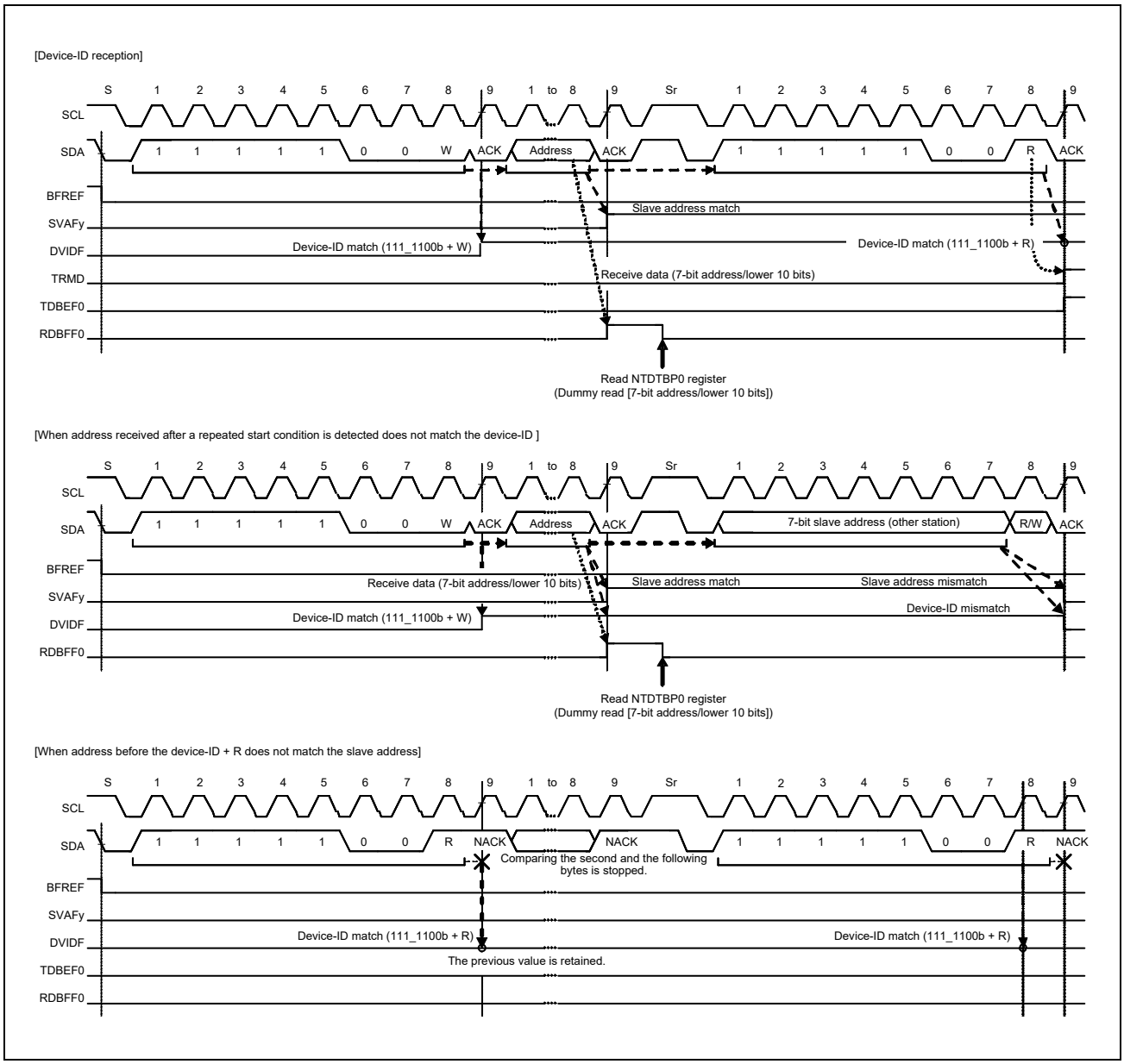


Figure 26.64 SVAFy/DVIDF Flag Set / Clear Timing during Reception of Device-ID

(d) Hs-Mode Master Code Detection [I²C mode]

IIC has a facility for detecting the Hs-mode master code (0000_1XXXb). When IIC receives the Hs-mode master code (0000_1XXXb) as the first byte after a START condition was issued with the SVCTL.HSMCE bit set to 1, this module recognizes the address as the Hs-mode master code, sets the SVST.HSMCF flag to 1 on the rising edge of the ninth SCL clock cycle. The first byte after Repeated START after NACK response to Hs-mode master code is recognized as a slave address and compared with the slave address set by SVDVADy.SVAD[9:0] (y = 0 to 2). When IIC detects a match of the set slave address, the corresponding SVST.SVAFy flag (y = 0 to 2) is set to 1 at the rising edge of the ninth SCL clock cycle, and the NTST.RDBFF0 flag or the NTST.TDBEF0 flag is set to 1 by the following R/W# bit. This causes a receive data full interrupt (IICn_RX) or transmit data empty interrupt (IICn_TX) to be generated. The SVAFy flag is used to identify which slave address has been specified. The SVST.HSMCF flag is cleared to 0 when the STOP condition is detected.

NOTE

If the Hs-mode master code (0000_1XXXb) is received with the SVCTL.HSMCE bit set to 0, other patterns are ignored until the STOP condition is detected.

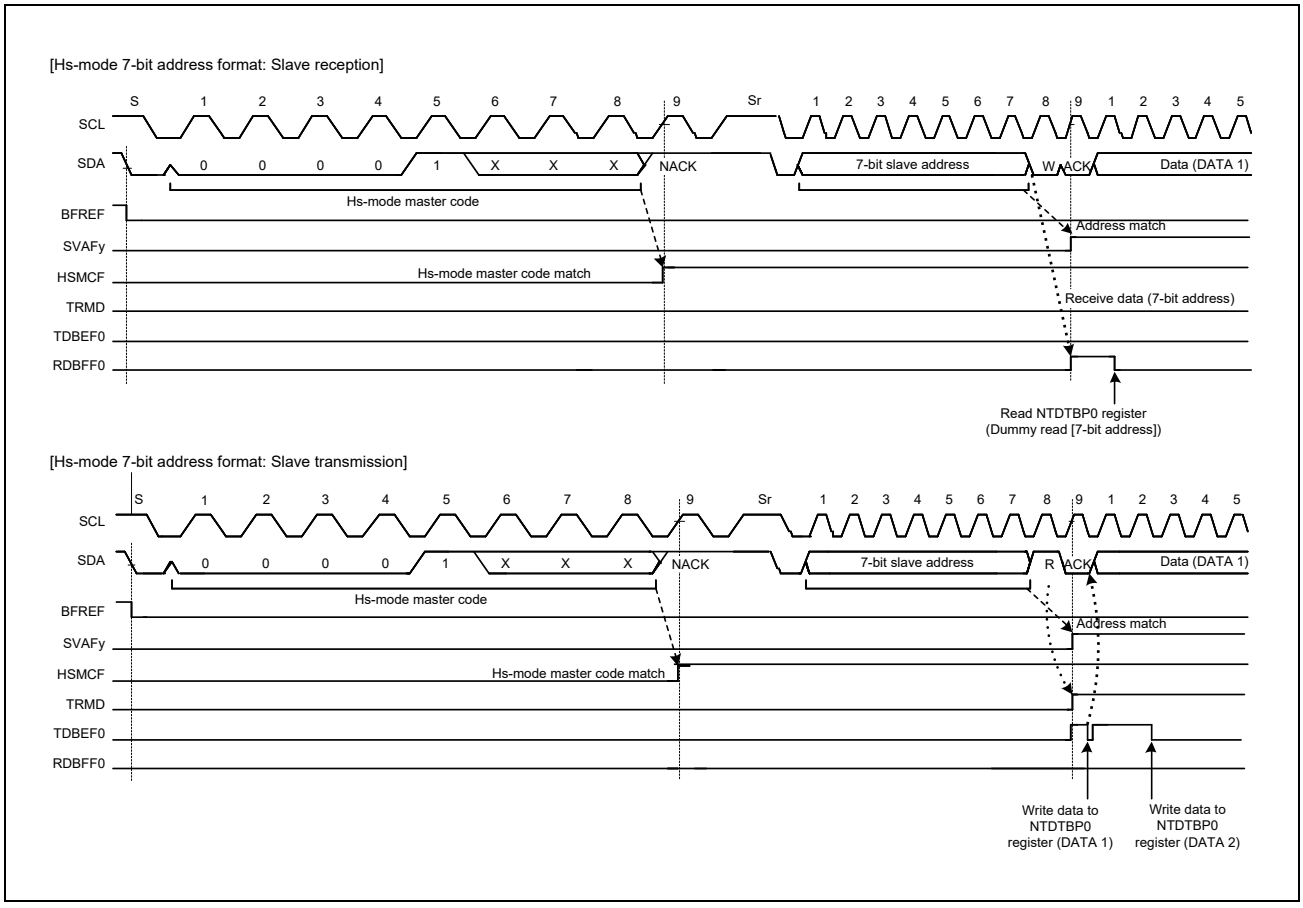


Figure 26.65 SVAFy/HSMCF Flag Set Timing during Reception of Hs-Mode Master Code

(e) CCC Detection Function [I3C mode]

- In case of Broadcast CCC

1. It receives Broadcast Address (H'7E) and R/W# = 0 after START condition or Repeated START.
2. Respond to ACK.
3. Receive Common Command Code (CCC).
4. In accordance with the CCC, the following data is stored. (Storage destination: refer to **Table 26.12**)
5. Store the Receive Status Descriptor into the Receive Status Queue.

- In case of Broadcast CCC (ENTDAA)

1. It receives Broadcast Address (H'7E) and R/W# = 0 after START condition.
2. Respond to ACK.
3. Receive ENTDAA.
4. If receives Broadcast Address (H'7E) and R/W# = 1 after Repeated START.
5. When the Dynamic Address is not assigned, ACK response is done.
6. This Provisional ID (SDCTPIDH[31:0], SDCTPIDL[15:0]), BCR (SVDCT.TBCR[7:0]) and DCR (SVDCT.TDCR[7:0]) are transmitted.
7. When winning the arbitration in a transmission of the above Step 6, the dynamic address following that is received. When losing arbitration in a transmission of the above Step 6, processing of Step 6 is repeated from Step 4.
8. When parity of the Dynamic Address is valid, ACK response is done.
9. When parity of the Dynamic Address is invalid, NACK replies, and repeat the process from Steps 4 to 7.
10. SDATBAS0.SDDYAD[7:0] is renewed and the SVDVAD0.SDYADV bit is set to 1.
11. Upon detecting the STOP condition, Store the Receive Status Descriptor into the Receive Status Queue.

- In case of Direct Write CCC

1. It receives Broadcast Address (H'7E) and R/W# = 0 after START condition or Repeated START.
2. Respond to ACK.
3. Receive Common Command Code (CCC).
4. Receive Dynamic Address and R/W# = 0 after Repeated START.
5. Compare the received Dynamic Address with the assigned Dynamic Address, and if it matches, I3C responds with ACK.
   If they do not match, it responds with NACK and waits for Repeated START or STOP.
6. In accordance with the CCC, the following data is stored. (Storage destination: refer to **Table 26.12**)
7. Store the Receive Status Descriptor into the Receive Status Queue.

- In case of Direct Read CCC

1. It receives Broadcast Address (H'7E) and R/W# = 1 after START condition or Repeated START.
2. Respond to ACK.
3. Receive Common Command Code (CCC).
4. Receive Dynamic Address and R/W# = 1 after Repeated START.

5. Compare the received Dynamic Address with the assigned Dynamic Address, and if it matches, I3C responds with ACK.
   If they do not match, it responds with NACK and waits for Repeated START or STOP.
6. Respond from SFR according to CCC. (Responding CCC: refer to **Table 26.12**)
7. Store the Receive Status Descriptor into the Receive Status Queue.

Table 26.12 Common Command Code Operation

Command Code	CCC Type	Command Name	With Data	Auto Response	Storage
H'00	Broadcast	ENEC	Yes	—	SFR
H'01	Broadcast	DISEC	Yes	—	SFR
H'02	Broadcast	ENTAS0	No	—	SFR
H'03	Broadcast	ENTAS1	No	—	SFR
H'04	Broadcast	ENTAS2	No	—	SFR
H'05	Broadcast	ENTAS3	No	—	SFR
H'06	Broadcast	RSTDAA	No	—	SFR
H'07	Broadcast	ENTDAA	Yes	Yes	SFR
H'08	Broadcast	DEFSLVS	Yes	—	FIFO
H'09	Broadcast	SETMWL	Yes	—	SFR
H'0A	Broadcast	SETMRL	Yes	—	SFR
H'0B	Broadcast	ENTTM	Yes	—	SFR
H'29	Broadcast	SETAASA	No	—	SFR
H'80	Direct Write	ENEC	Yes	—	SFR
H'81	Direct Write	DISEC	Yes	—	SFR
H'82	Direct Write	ENTAS0	No	—	SFR
H'83	Direct Write	ENTAS1	No	—	SFR
H'84	Direct Write	ENTAS2	No	—	SFR
H'85	Direct Write	ENTAS3	No	—	SFR
H'86	Direct Write	RSTDAA	No	—	SFR
H'87	Direct Write	SETDASA	Yes	—	SFR
H'88	Direct Write	SETNEWDA	Yes	—	SFR
H'89	Direct Write	SETMWL	Yes	—	SFR
H'8A	Direct Write	SETMRL	Yes	—	SFR
H'8B	Direct Read	GETMWL	—	Yes	SFR
H'8C	Direct Read	GETMRL	—	Yes	SFR
H'8D	Direct Read	GETPID	—	Yes	SFR
H'8E	Direct Read	GETBCR	—	Yes	SFR
H'8F	Direct Read	GETDCR	—	Yes	SFR
H'90	Direct Read	GETSTATUS	—	Yes	SFR
H'91	Direct Read	GETACCMST	—	Yes	SFR
H'94	Direct Read	GETMXDS	—	Yes	SFR

(5) Arbitration-Lost Detection [I²C mode]

In addition to the normal arbitration-lost detection function defined by the I²C-bus specification, the I3C has functions to prevent double-issue of a start condition, to detect arbitration-lost during transmission of NACK, and to detect arbitration- lost in slave transmit mode.

(a) Master Arbitration-Lost Detection (MALE Bit)

The I3C drives the SDAn line low to issue a start condition. However, if the SDAn line has already been driven low by another master device issuing a start condition, this module causes arbitration to be lost, so priority is given to transfer by the other master device. Similarly, if the CNDCTL.STCND bit is set to 1 while the BCST.BFREF flag is 0 (bus busy state), arbitration is lost, so priority is given to transfer by the other master device. No start condition is issued in this case.

When a start condition is issued successfully, if the data for transmission including the address bits (the internal SDA output level) and the level on the SDAn line do not match (the high output as the internal SDA output, that is, the SDA0 pin is in the high-impedance state) and the low level is detected on the SDAn line, the I3C loses in arbitration.

I3C detects master arbitration-lost when the following conditions are met while the BSTE.ALE bit = 1 and the BFCTL.MALE bit = 1 (master arbitration-lost detection enabled).

If arbitration of mastership is lost, I3C immediately enters slave receive mode.

If a slave address (including the general call address) matches its own address at this time, I3C continues in slave operation.

[Conditions for master arbitration-lost]

- Non-matching of the internal level for output on SDA and the level on the SDAn line after a START condition was issued by setting the CNDCTL.STCND bit to 1 while the BCST.BFREF flag was set to 1 (erroneous issuing of a START condition)
- Setting of the CNDCTL.STCND bit to 1 (START condition double-issue error) while the BFREF flag is set to 0

NOTE

I3C does not issue a START condition.

- When the transmit data excluding acknowledge (internal SDA output level) does not match the level on the SDAn line in master transmit mode (bits CRMS and TRMD in the PRSST register = 11)

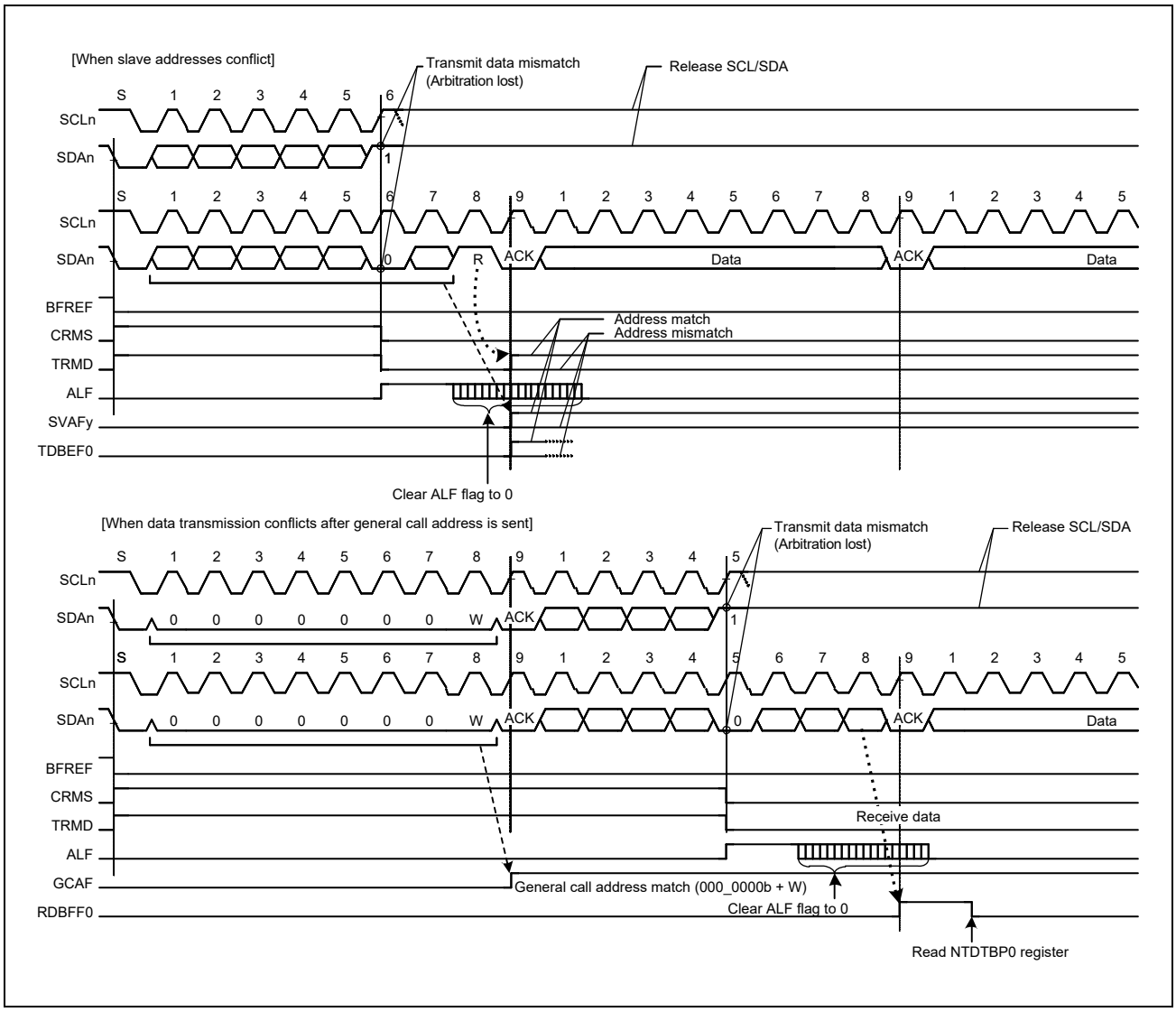


Figure 26.66 Examples of Master Arbitration-Lost Detection (MALE = 1)

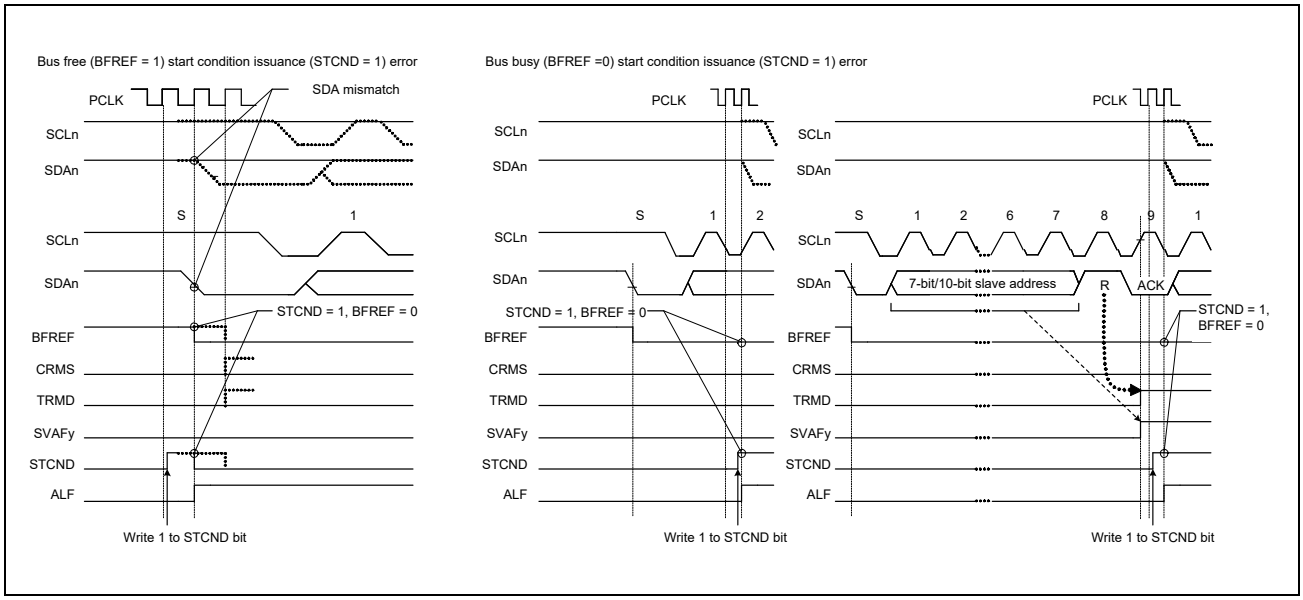


Figure 26.67 Arbitration-Lost Detection when a START Condition is Issued (MALE = 1)

(b) Arbitration-Lost Detection during NACK Transmission (NALE Bit)

The I3C has a function to cause arbitration to be lost if the internal SDA output level does not match the level on the SDAn line (the high output as the internal SDA output, i.e., the SDAn pin is in the high-impedance state) and the low level is detected on the SDAn line during transmission of NACK in receive mode. Arbitration is lost due to a conflict of NACK transmission and ACK transmission when two or more master devices receive data from the same slave device simultaneously in a multi-master system. Such conflict occurs when multiple master devices send/receive the same information through a single slave device.

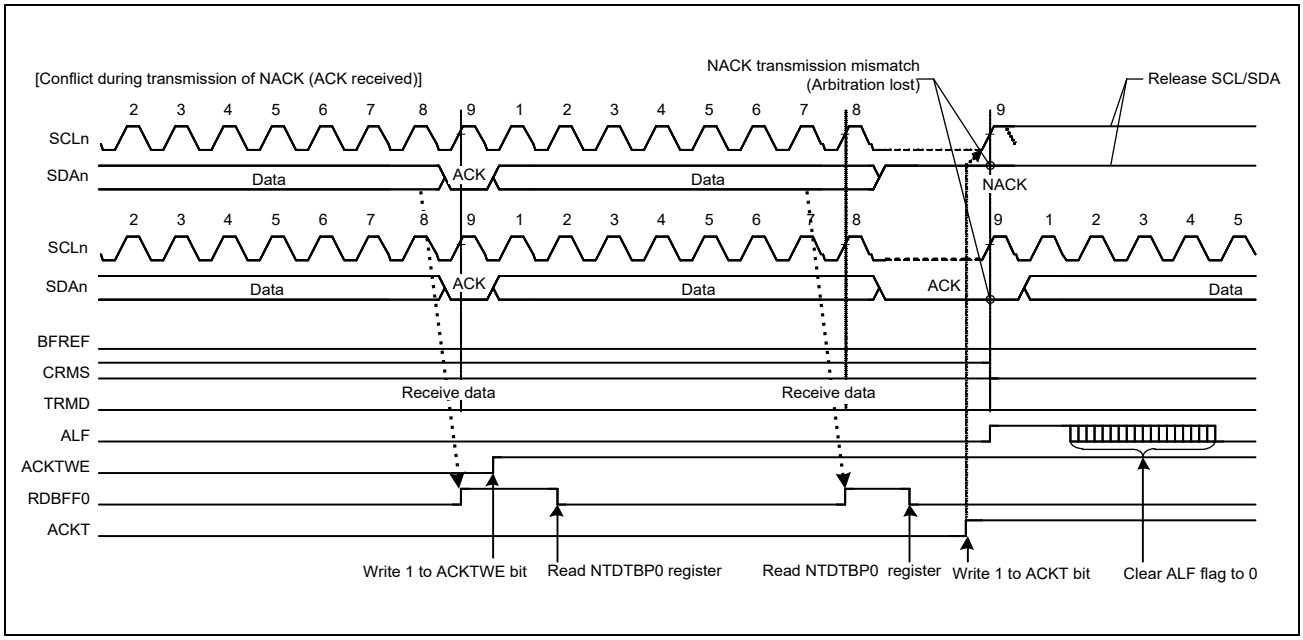


Figure 26.68 Example of Arbitration-Lost Detection during Transmission of NACK (NALE = 1)

The following section explains arbitration-lost detection using an example where two master devices (master A and master B) and a single slave device are connected through the bus. In this example, master A receives 2 bytes of data from the slave device, and master B receives 4 bytes of data from the slave device.

If master A and master B access the slave device simultaneously, because the slave address is identical, arbitration is not lost in both master A and master B during access to the slave device. Therefore, both master A and master B recognize that they have obtained the bus mastership and operate as such. In this example, master A sends NACK when it has received 2 final bytes of data from the slave device. Meanwhile, master B sends ACK because it has not received the necessary 4 bytes of data. At this time, the NACK transmission from master A and the ACK transmission from master B conflict. In general, if a conflict like this occurs, master A cannot detect ACK transmitted by master B and issues a stop condition. Therefore, the issuance of the stop condition conflicts with the SCL clock output of master B, which disturbs communication.

When this module receives ACK during transmission of NACK, it detects a defeat in conflict with other master devices and causes arbitration to be lost.

If arbitration is lost during transmission of NACK, this module is immediately released from the slave-matched state and enters slave receive mode. This prevents a stop condition from being issued, preventing a communication failure on the bus.

The I3C detects arbitration-lost during transmission of NACK when the following condition is met while the BSTE.ALE bit = 1 and the BFCTL.NALE bit = 1 (arbitration-lost detection during NACK transmission enabled).

[Condition for arbitration-lost during NACK transmission]

- When the internal SDA output level does not match the SDAn line (ACK is received) during transmission of NACK (ACKCTL.ACKT bit = 1)

(c) Slave Arbitration-Lost Detection (SALE Bit)

The I3C has a function to cause arbitration to be lost if the data for transmission (the internal SDA output level) and the level on the SDAn line do not match (the high output as the internal SDA output, that is, the SDAn pin is in the high impedance state) and the low level is detected on the SDAn line in slave transmit mode.

If arbitration is lost during transmission of DATA, this module is immediately released from the slave-matched state and enters slave receive mode.

The I3C detects slave arbitration-lost when the following condition is met while the BSTE.ALE bit = 1 and the BFCTL.SALE bit = 1 (slave arbitration-lost detection enabled).

[Condition for slave arbitration-lost]

- When the transmit data excluding acknowledge (internal SDA output level) does not match the level on the SDAn line in slave transmit mode (bits CRMS and TRMD in the PRSST register = 01).

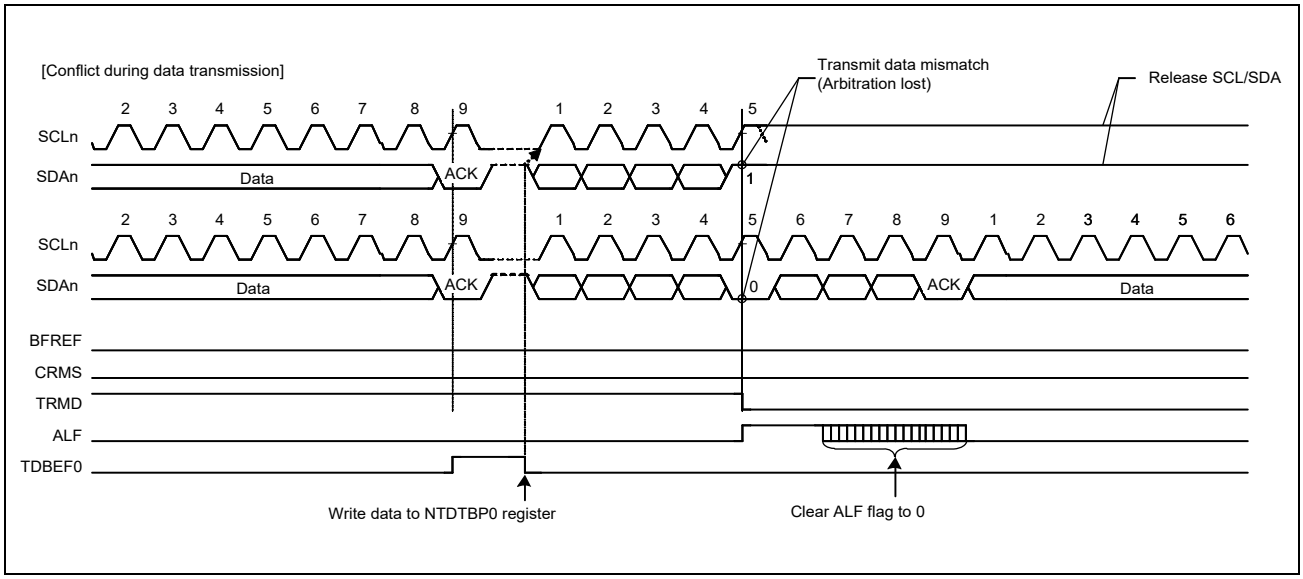


Figure 26.69 Example of Slave Arbitration-Lost Detection (SALE = 1)

(6) Clock Stretching [I²C mode]

(a) Function to Prevent Wrong Transmission of Transmit Data

When data have not been written to the I²C bus transmit data register (NTDTBP0) with I3C in transmission mode (PRSST.TRMD = 1), the SCLn line is automatically held at the low level over the intervals shown below. This low-hold period is extended until data for transmission have been written, which prevents the unintended transmission of erroneous data.

Master transmit mode

- Low-level interval after a START condition or Repeated START condition is issued
- Low-level interval between the ninth clock cycle of one transfer and the first clock cycle of the next

Slave transmit mode

- Low-level interval between the ninth clock cycle of one transfer and the first clock cycle of the next

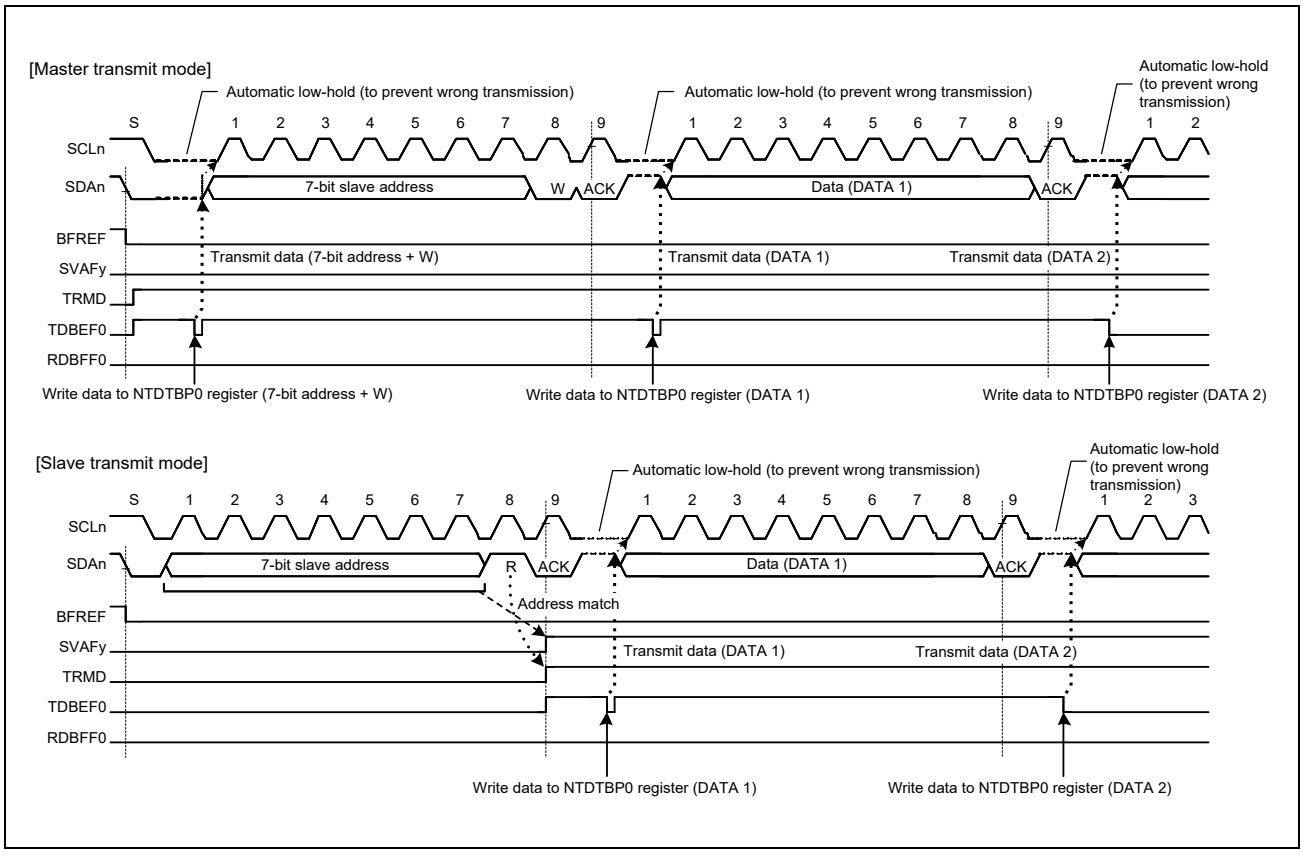


Figure 26.70 Automatic Low-Hold Operation in Transmit Mode

(b) NACK Reception Transfer Abort Function

I3C has a function to abort transfer operation when NACK is received in transmit mode (PRSST.TRMD = 1). This function is enabled when the BSTE.NACKDE bit is set to 1 (transfer abort enabled). If the next transmit data has already been written (NTST.TDBEF0 = 0) when NACK is received, next data transmission at the falling edge of the ninth SCL clock cycle is automatically aborted. This prevents the SDAn line output level from being held low when the MSB of the next transmit data is 0.

If the transfer operation is aborted by this function (BST.NACKDF = 1), transmit operation and receive operation are discontinued. To restore transmit/receive operation, be sure to set the NACKDF flag to 0. In master transmit mode, restore operation using either of the methods below:

- After issuing a Repeated START condition, set the NACKDF flag to 0
- After issuing a STOP condition, set the NACKDF flag to 0 and then issue a START condition

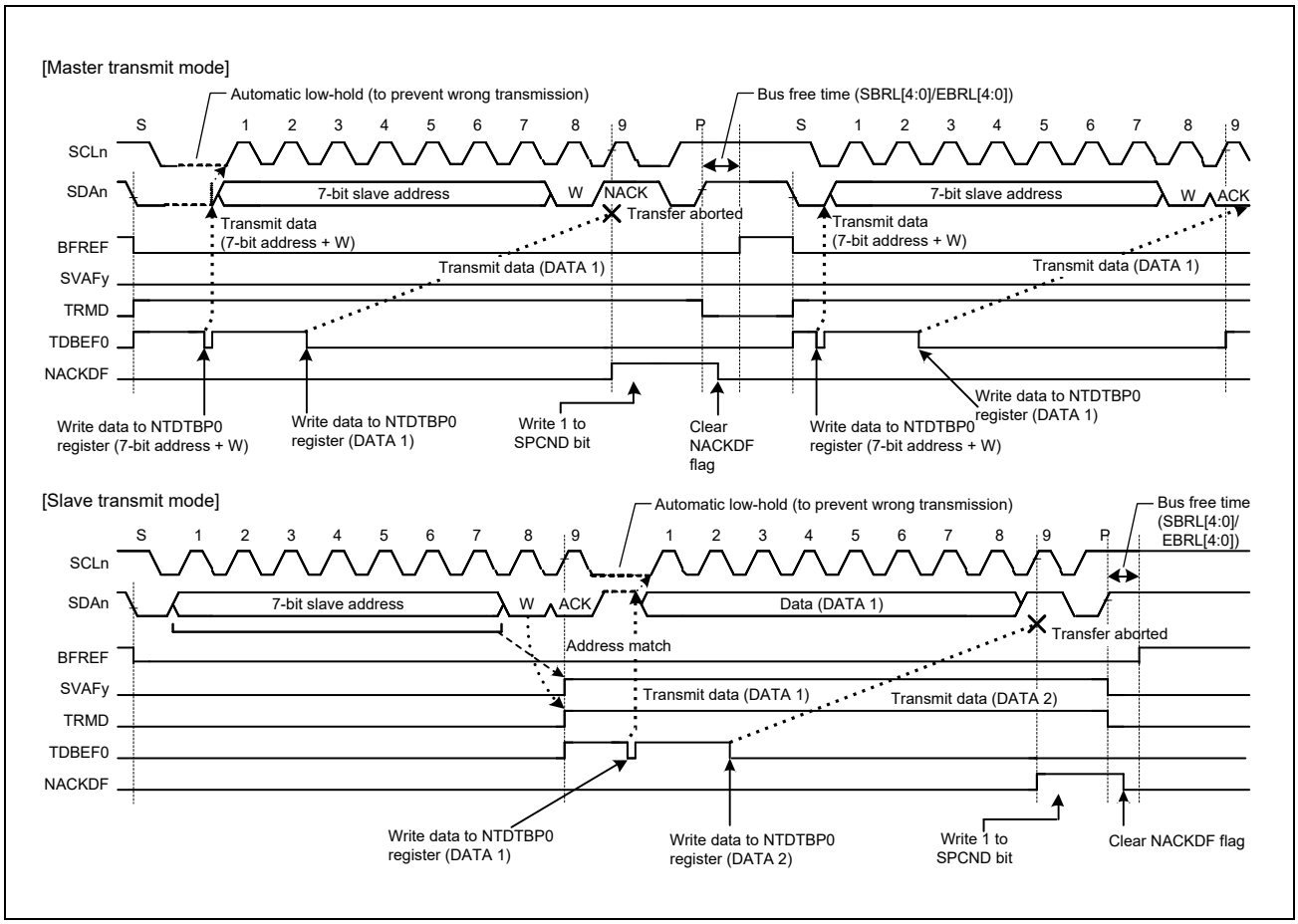


Figure 26.71 Abort of Data Transfer When NACK is Received (NACKE = 1)

(c) Function to Prevent Failure to Receive Data

If response processing is delayed when receive data (NTDTBP0) read is delayed for a period of one transfer frame or more with receive data full (NTST.RDBFF0 = 1) in receive mode (PRSST.TRMD = 0), I3C holds the SCLn line low automatically immediately before the next data is received to prevent failure to receive data.

This function to prevent failure to receive data using the automatic low-hold function is also enabled even if the read processing of the final receive data is delayed and, in the meantime, I3C's own slave address or another slave address is received after a STOP condition is issued.

Sections in which the SCLn line is held low can be selected with a combination of the RWE and ACKTWE bits in SCSTRCTL.

1) 1-Byte Receive Operation and Automatic Low-Hold Function Using the RWE Bit

When the SCSTRCTL.RWE bit is set to 1, I3C performs 1-byte receive operation using the RWE bit function.

Furthermore, when the SCSTRCTL.ACKTWE bit = 0, I3C automatically sends the ACKCTL.ACKT bit value for the acknowledge bit in the period from the falling edge of the eighth SCL clock cycle to the falling edge of the ninth SCL clock cycle, and automatically holds the SCLn line low at the falling edge of the ninth SCL clock cycle using the RWE bit function. This low hold is released by reading data from NTDTBP0, which enables bytewise receive operation.

The RWE bit function is enabled for receive frames after a match with I3C's own slave address (including the general call address and host address) is obtained in master receive mode or slave receive mode.

2) 1-Byte Receive Operation (ACK/NACK Transmission Control) and Automatic Low-Hold Function Using the ACKTWE Bit

When the SCSTRCTL.ACKTWE bit is set to 1, I3C performs 1-byte receive operation using the ACKTWE bit function.

When the ACKTWE bit is set to 1, the NTST.RDBFF0 flag (receive data full) is set to 1 at the rising edge of the eighth SCL clock cycle, and the SCLn line is automatically held low at the falling edge of the eighth SCL clock cycle. This lowhold is released by writing a value to the ACKCTL.ACKT bit, but cannot be released by reading data from NTDTBP0, which enables receive operation by the ACK/NACK transmission control according to the data received in byte units.

The ACKTWE bit function is enabled for receive frames after a match with I3C's own slave address (including the general call address and host address) is obtained in master receive mode or slave receive mode.

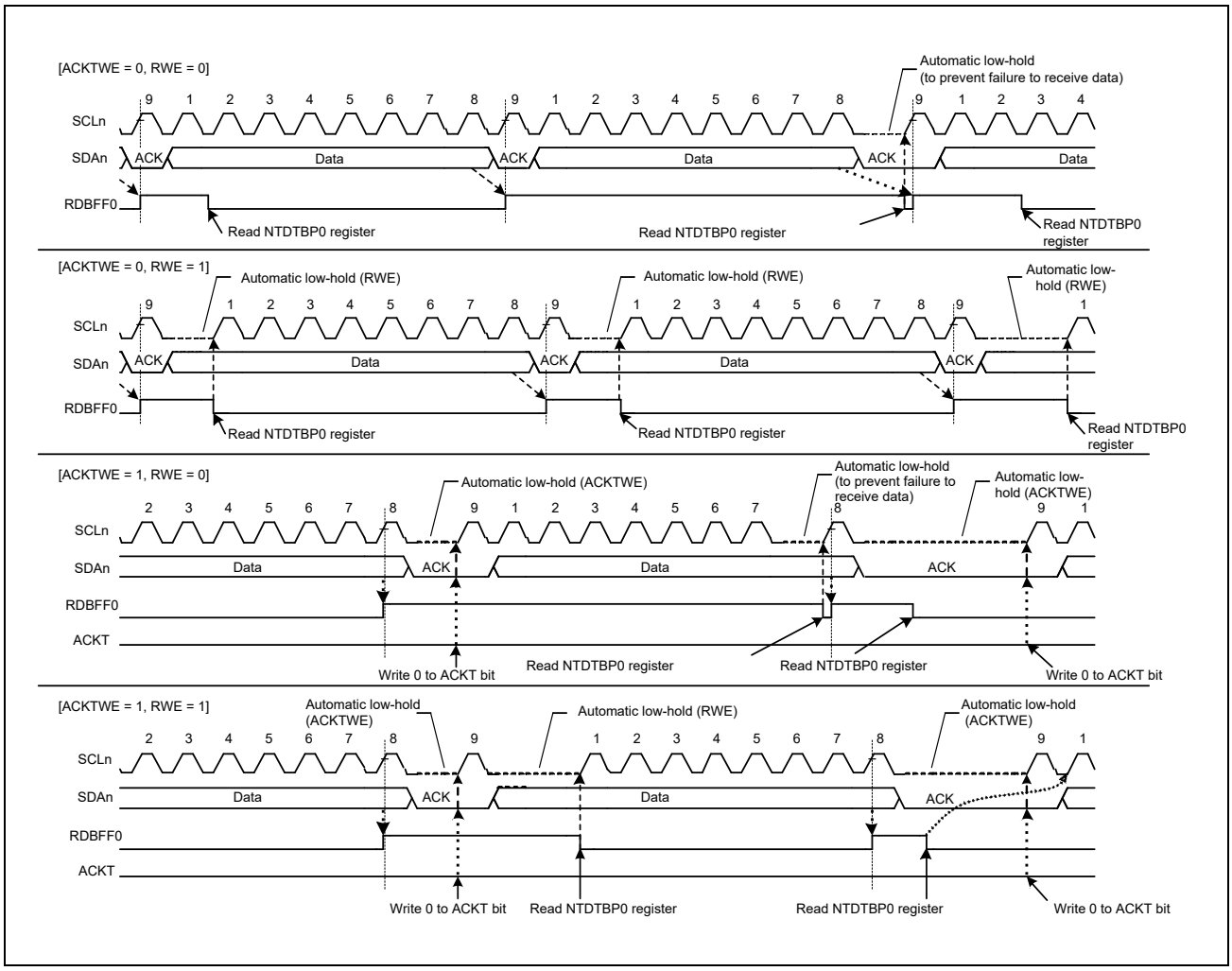


Figure 26.72 Automatic Low-Hold Operation in Receive Mode (using ACKTWE and RWE bits)

(7) Clock Stalling [I3C mode]

I3C has the function of stalling the SCL during the SCL Low period.

The SCL stall control is described in the table below.

Table 26.13 I3C clock stalling

Clock Stalling Condition	Clock Stalling Control	Clock Stalling Period
I3C Transfer, ACK/NACK Phase	SCSTLCTL.ACKPE bit setting	During the count period of SCSTLCTL.STLCYC[15: 0] value
	Transmit Data FIFO Empty	Until data is written to the TX FIFO
	Receive Data FIFO Full	Until data is read from the RX FIFO
I3C Write Data Transfer, Parity Bit	SCSTLCTL.PARPE bit setting	During the count period of SCSTLCTL.STLCYC[15: 0] value
	Transmit Data FIFO Empty	Until data is written to the TX FIFO
I3C Read Transfer, Transition Bit	Receive Data FIFO Full	Until data is read from the RX FIFO
Assigned Address Phase	SCSTLCTL.AAPE bit setting	During the count period of SCSTLCTL.STLCYC[15: 0] value
I3C Transfer, ACK/NACK Phase	SCSTLCTL.ACKPE bit setting	During the count period of SCSTLCTL.STLCYC[15: 0] value
	Transmit Data FIFO Empty	Until data is written to the TX FIFO
	Receive Data FIFO Full	Until data is read from the RX FIFO
I3C Write Data Transfer, Parity Bit	SCSTLCTL.PARPE bit setting	During the count period of SCSTLCTL.STLCYC[15: 0] value
	Transmit Data FIFO Empty	Until data is written to the TX FIFO
I3C Read Transfer, Transition Bit	Receive Data FIFO Full	Until data is read from the RX FIFO
Assigned Address Phase	SCSTLCTL.AAPE bit setting	During the count period of SCSTLCTL.STLCYC[15: 0] value

The following figure shows the stalling timing of each Condition.

(a) I3C Transfer, ACK/NACK Phase

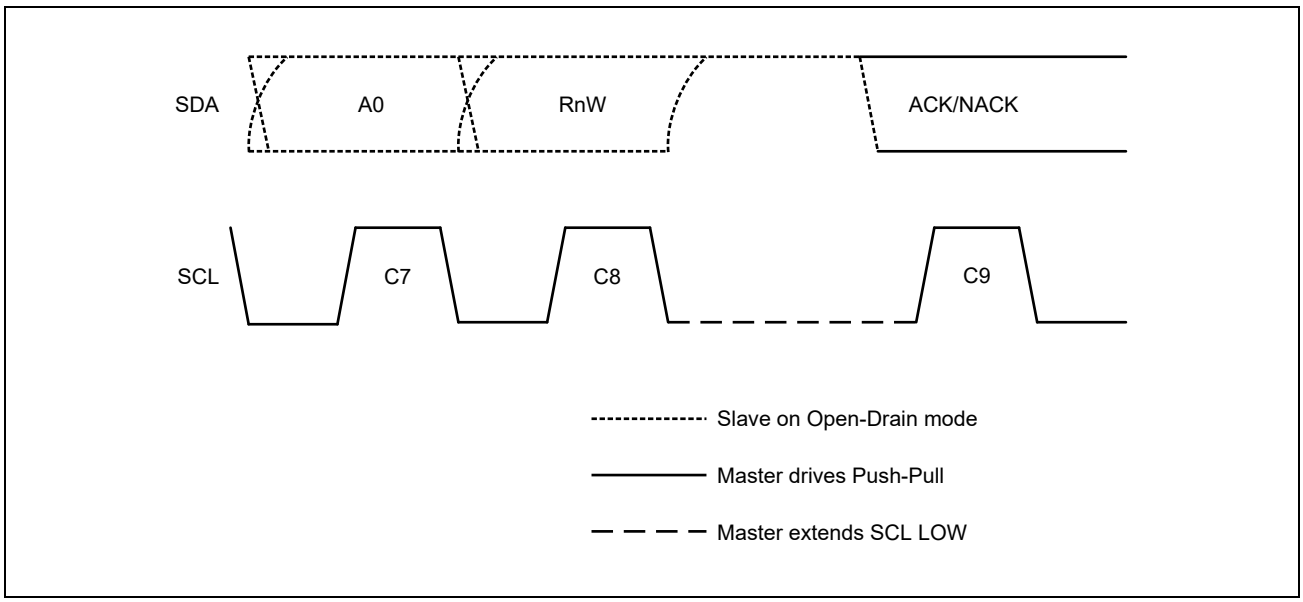


Figure 26.73 Master Clock Stalling in ACK Phase

(b) I3C Write Data Transfer, Parity Bit

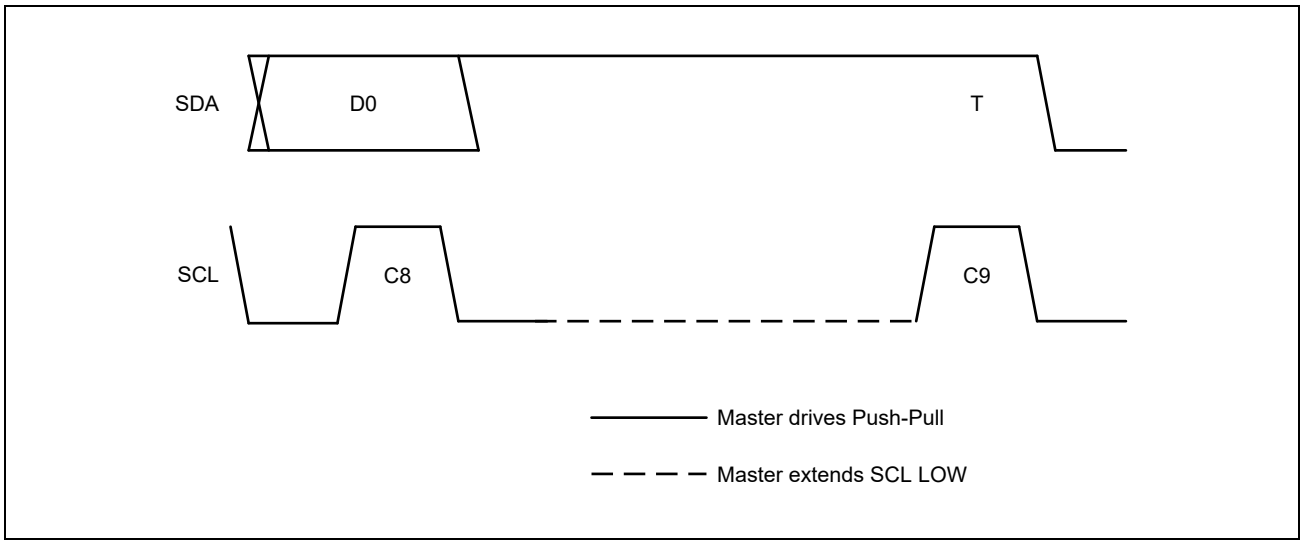


Figure 26.74 Master Clock Stalling in Write Parity Bit

(c) I3C Read Transfer, Transition Bit

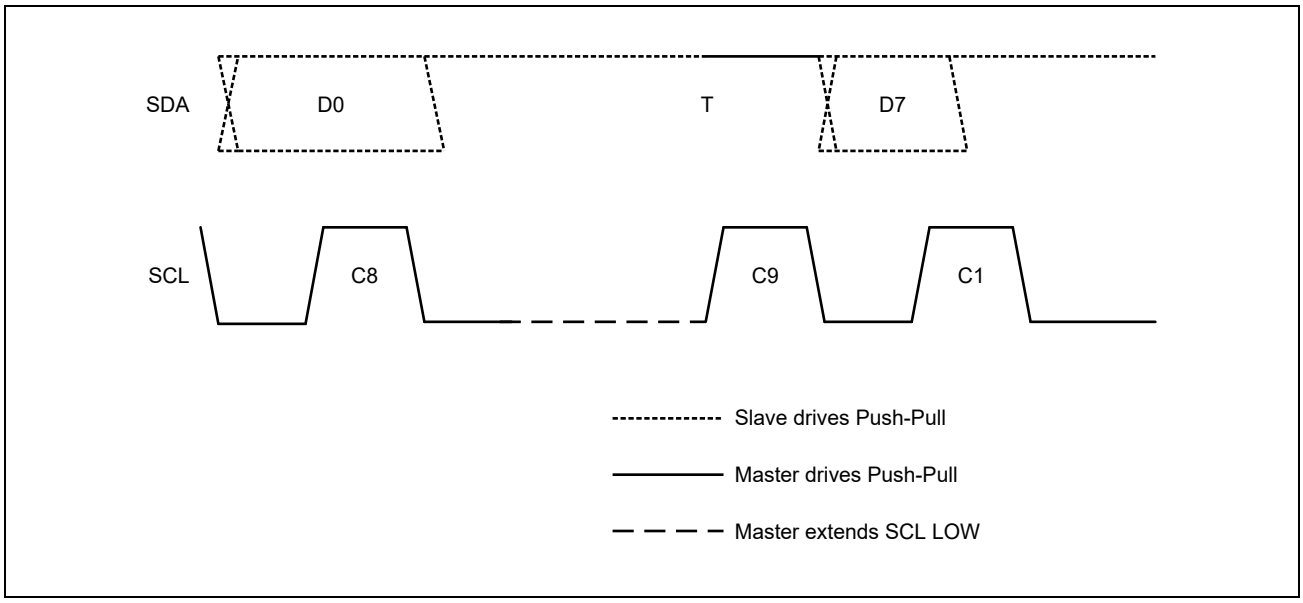


Figure 26.75 Master Clock Stalling in T-bit before Next Read Data

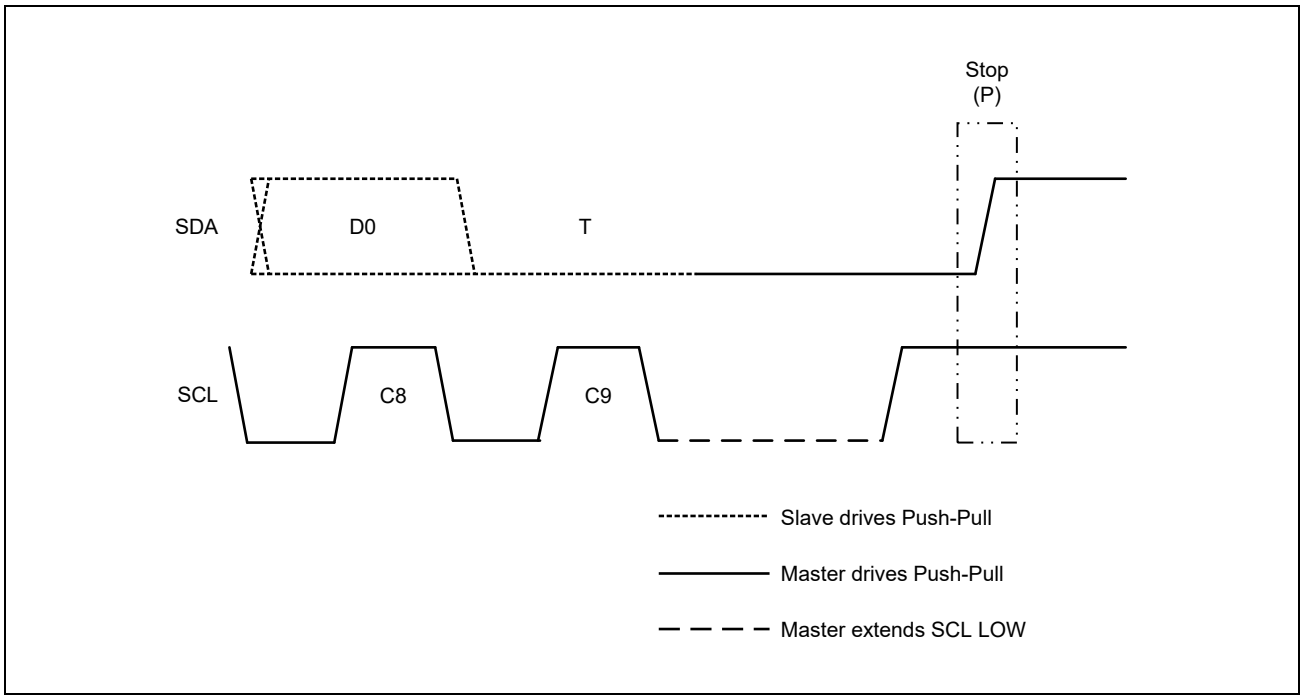


Figure 26.76 Master Clock Stalling in T-bit before STOP

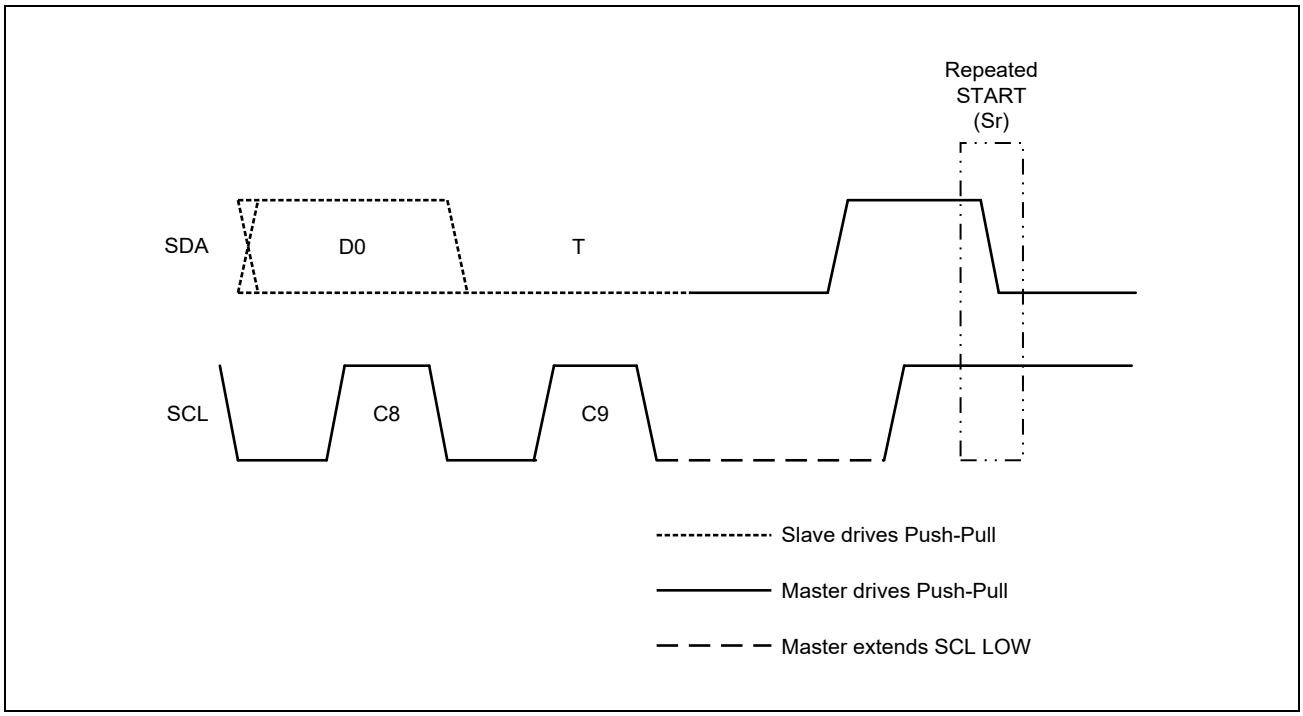


Figure 26.77 Master Clock Stalling in Low T-bit before Repeated START

(d) Dynamic Address Assignment, First Bit of Assigned Address

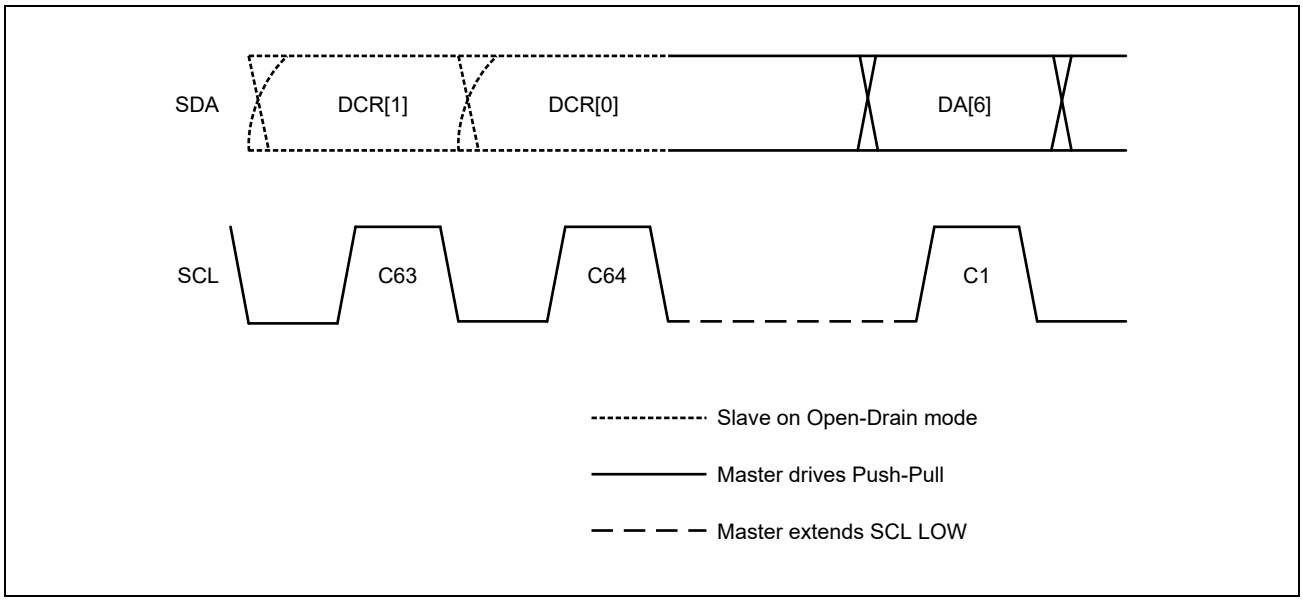


Figure 26.78 Master Clock Stalling in Dynamic Address First Bit

(8) In-Band Interrupt [I3C mode]

I3C detects In-Band Interrupt in the arbitrated Address Header following a START condition (but not following a Repeated START). If START Request (SDA Low Drive) is issued from Slave Device, I3C drives SCL low and completes START condition. After that, it supplies SCL and receives In-Band Interrupt Request.

The In-Band Interrupt to be detected is classified into the following three types.

- Slave Interrupt Request
- Mastership Request
- Hot-Join Event

The operation when detecting each In-Band Interrupt is described below.

(a) Slave Interrupt Request

1. Detect Slave Address with RnW bit High in Address Header.
2. Compare the detected Slave Address with the DVDYAD[7:0] in each DAT (DATBASm register).
3. When it does not match DAT.DVDYAD[7:0]:
   Responds NACK, then issues the STOP condition.
   When it matches the DAT.DVDYAD[7:0] bits and the DAT.DVSIRRJ bit = 1:
   It operates in the following order:
   (a) Responds NACK.
   (b) Issues Repeated START condition, then automatically issues Direct DISEC CCC to the detected Slave.
   (c) Issues the STOP condition.
   When it matches the DAT.DVDYAD[7:0] bits and the DAT.DVSIRRJ bit = 0:
   Responds ACK.
4. When DAT.DVIBIPL = 0:
   Issues the STOP condition.
   When DAT.DVIBIPL = 1:
   Drives the SCL to receive the IBI Data from the Slave following the ACK response and receives IBI Data.
   It stores the received IBI Data into the IBI Data Queue.
   Each time IBI Data of the size set by the NQTHCTL.IBIDSSZ[7:0] bits is received, the IBI Status Descriptor is stored in the IBI Status Queue.
5. After detection of Low of T-bit following IBI Data, issues STOP condition.
6. After issues of STOP condition NACK response:
   – If IBINCTL.NRSIRCTL = 0, the IBI Status Descriptor is not stored into the IBI Status Queue.
   – If IBINCTL.NRSIRCTL = 1, the IBI Status Descriptor is stored into the IBI Status Queue.
   ACK response:
   Stores the IBI Status Descriptor into the IBI Status Queue.

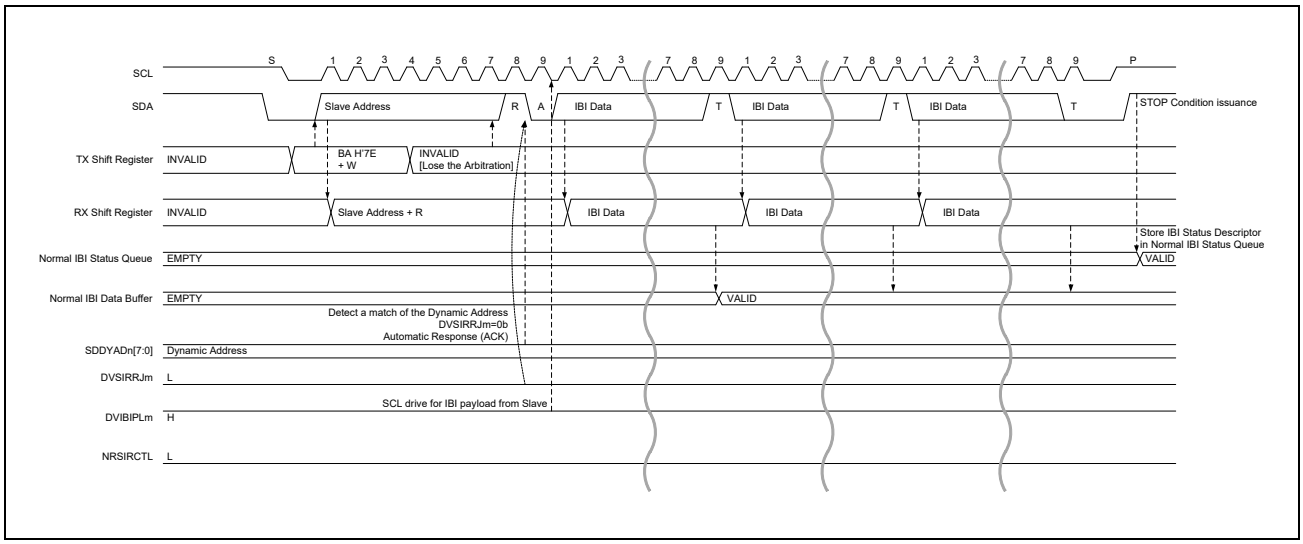


Figure 26.79 Slave Interrupt Request: ACK and DVIBIPL = 1

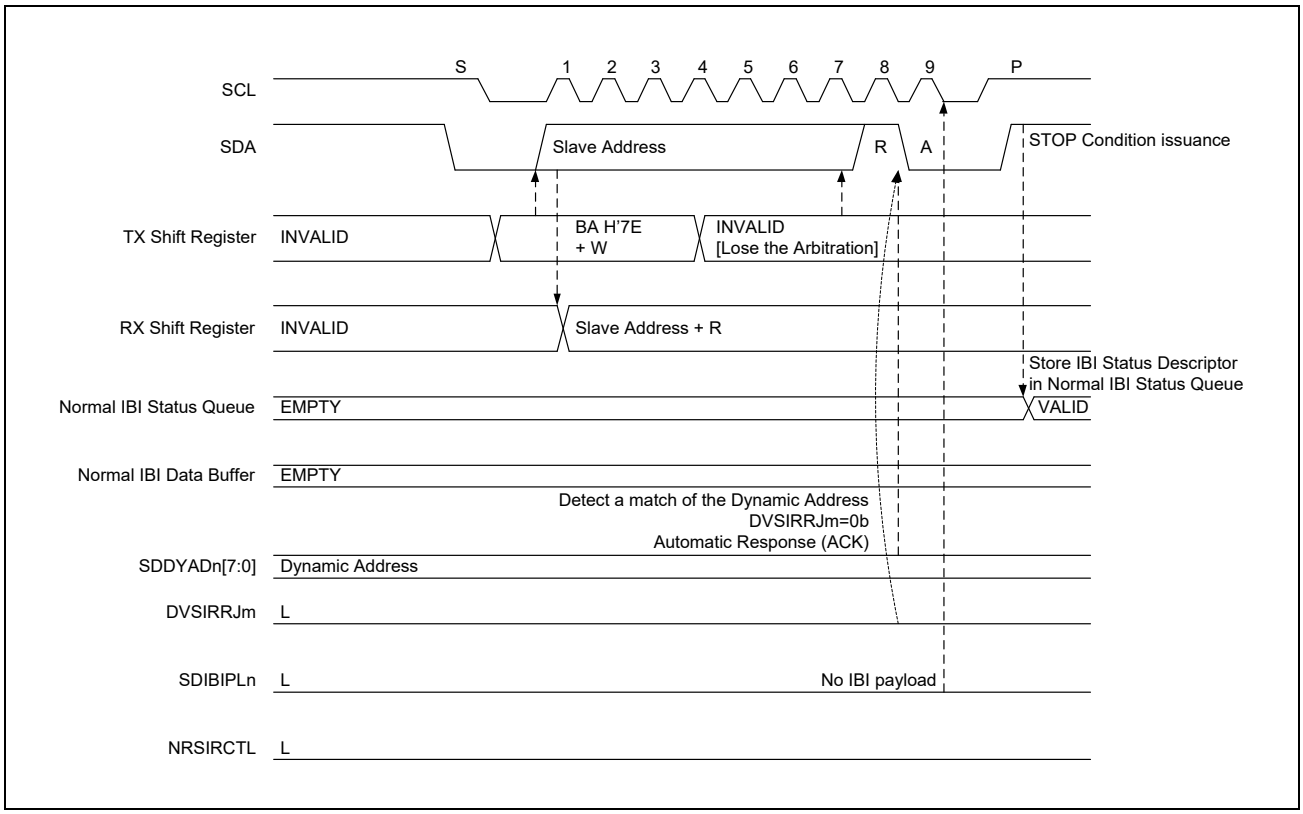


Figure 26.80 Slave Interrupt Request: ACK and DVIBIPL = 0

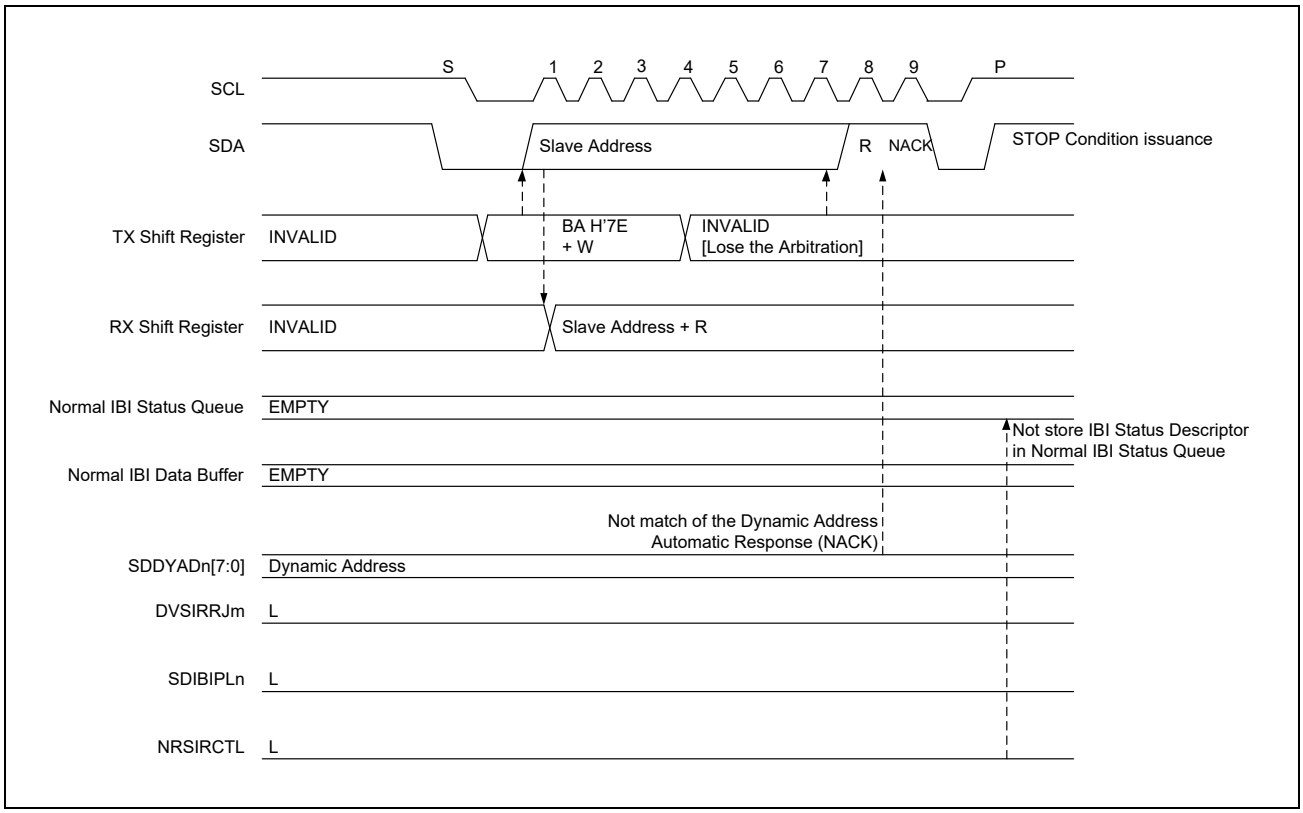


Figure 26.81 Slave Interrupt Request: NACK (not match the SDDYAD[7:0] of DAT) and NRSIRCTL = 0

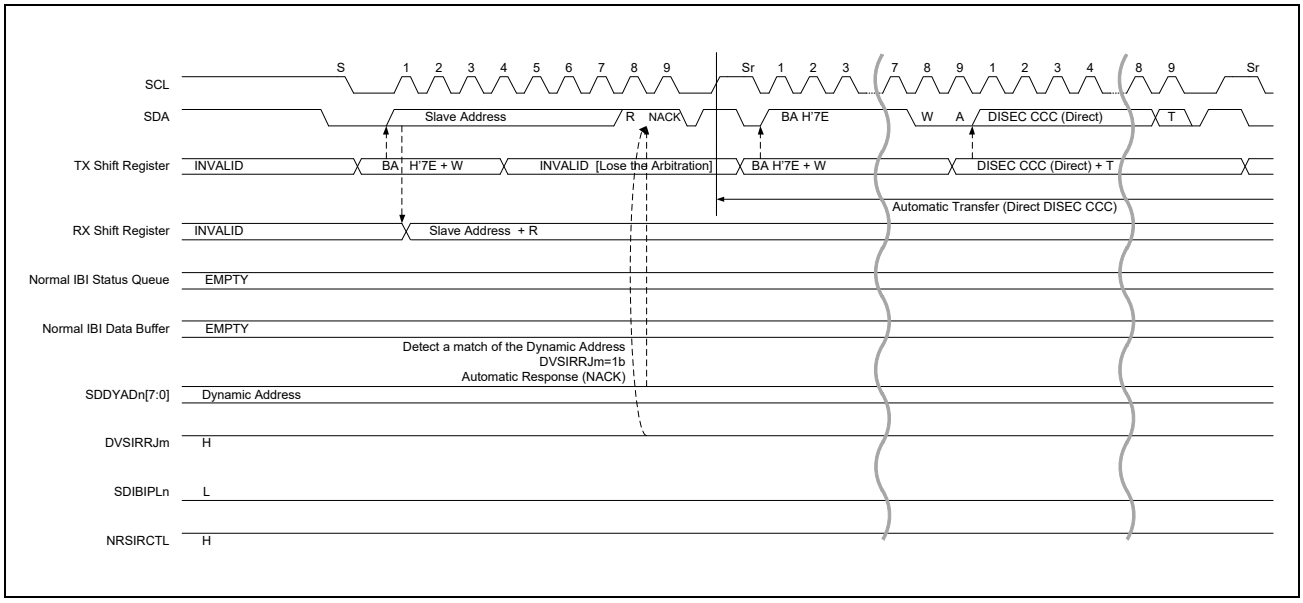


Figure 26.82 Slave Interrupt Request: NACK (DVSIRRJ = 1) and NRSIRCTL = 1 (1/2)

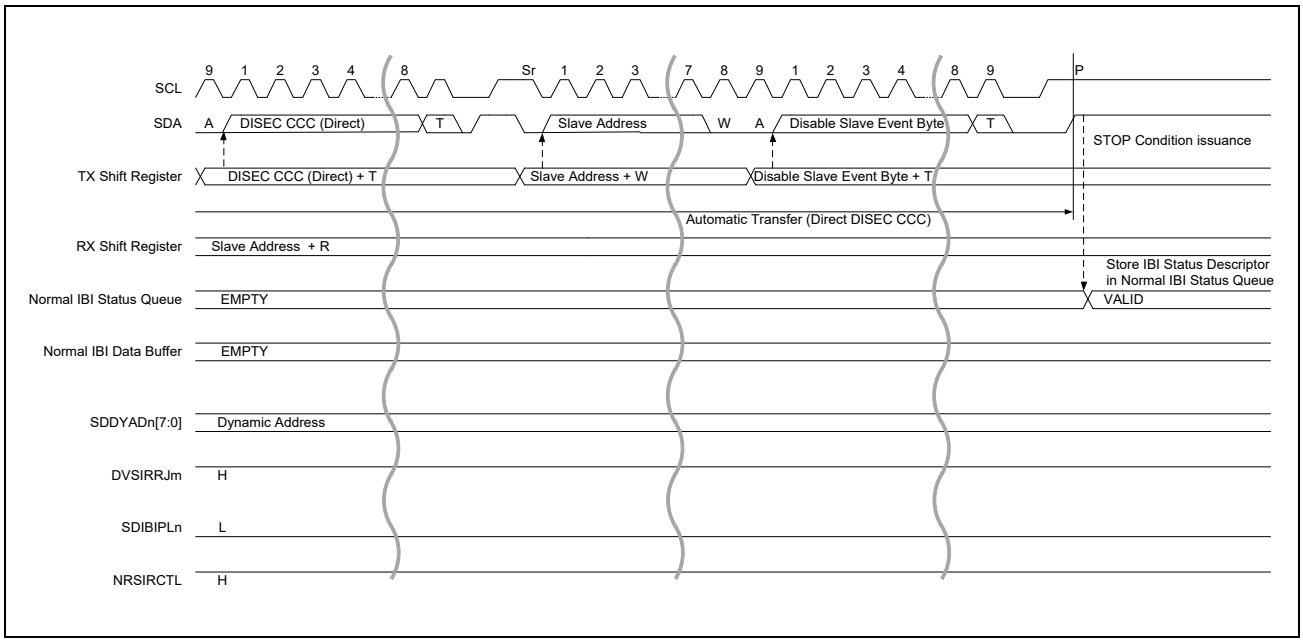


Figure 26.83 Slave Interrupt Request: NACK (DVSIRRJ = 1) and NRSIRCTL = 1 (2/2)

(b) Mastership Request

1. Detect Slave Address with RnW bit Low in Address Header.
2. Compare the detected Slave Address with the DVDYAD[7:0] in each DAT (DATBAS register).
3. When it does not match DAT.DVDYAD[7:0]:
   Rresponds NACK, then issues the STOP condition.
   When it matches the DAT.DVDYAD[7:0] bits and Device Role[1:0] in RBCR (MSDCTm) is other than I3C Master (01):
   Responds NACK, then issues the STOP condition.
   When it matches the DAT.DVDYAD[7:0] bits and Device Role[1:0] in RBCR (MSDCTm) is I3C Master (01):
   - When DAT.DVMRRJ = 1
     It operates in the following order.
     (a) Responds NACK.
     (b) Issued Repeated START condition and automatically issues Direct DISEC CCC to the detected Slave.
     (c) Issues the STOP condition.
   - When DAT.DVMRRJ = 0
     Responds ACK, then issues STOP condition.
4. After issues of STOP condition,
   NACK response:
   – If IBINCTL.NRMRCTL = 0, the IBI Status Descriptor is not stored into the IBI Status Queue.
   – If IBINCTL.NRMRCTL = 1, the IBI Status Descriptor is stored into the IBI Status Queue.
   ACK response:
   Stores the IBI Status Descriptor into the IBI Status Queue.

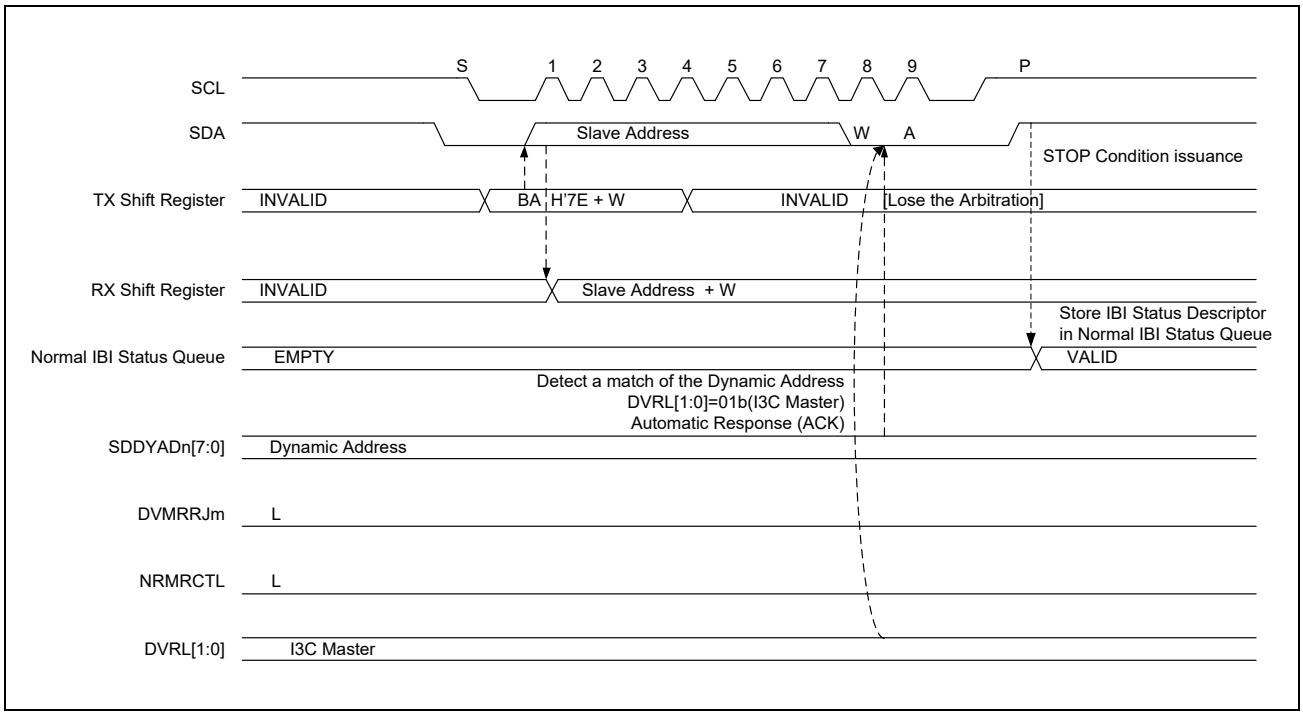


Figure 26.84 Mastership Request: ACK

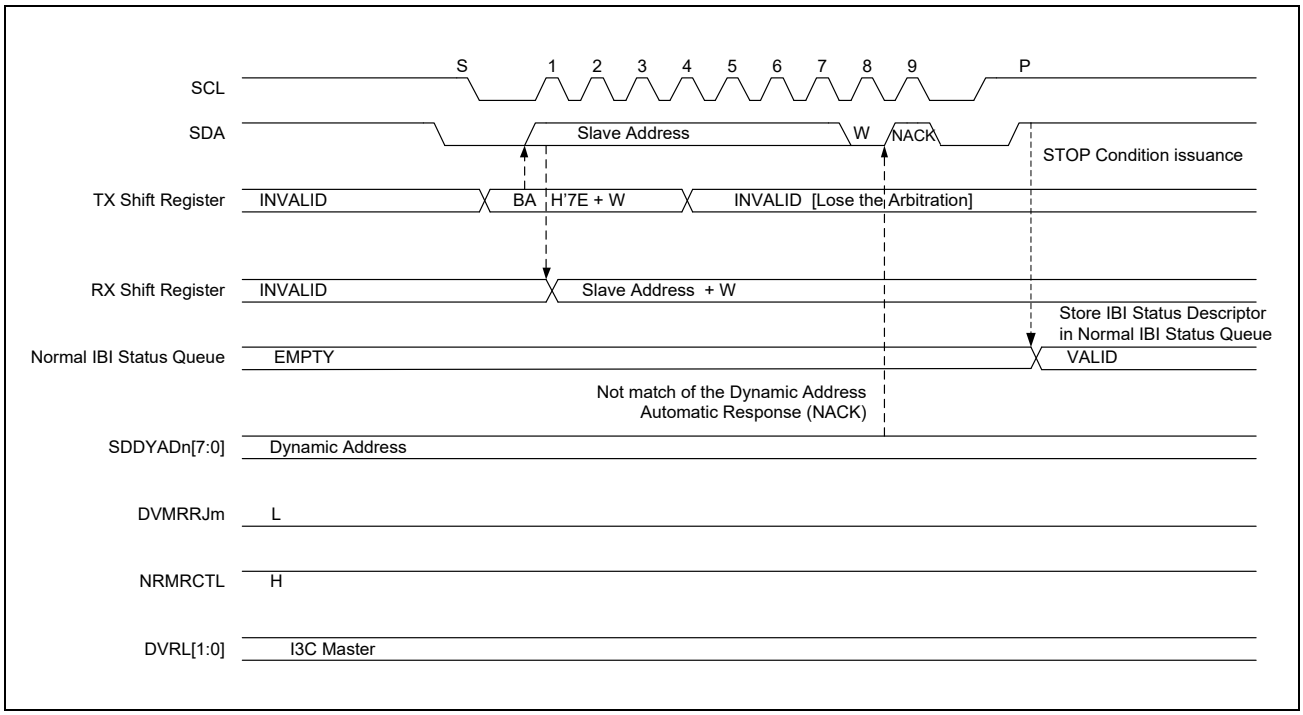


Figure 26.85 Mastership Request: NACK (not match the DVDYAD[7:0] of DAT) and NRMRCTL = 1

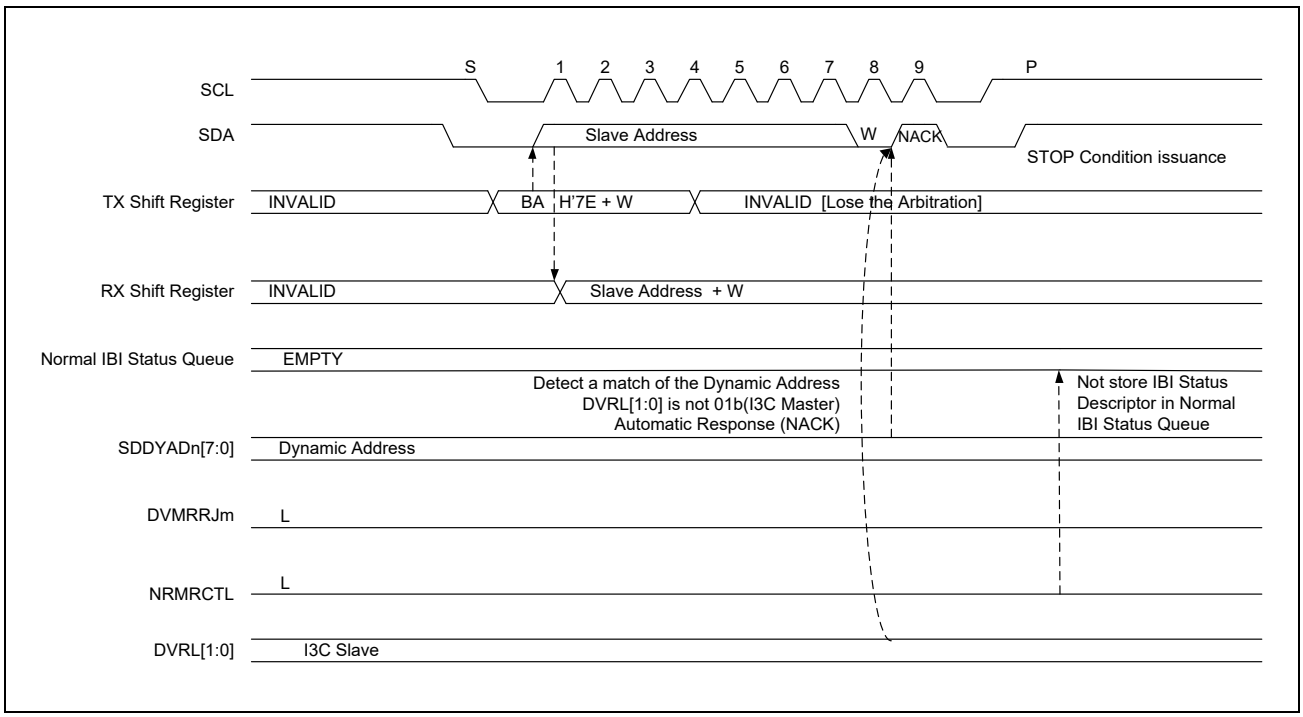


Figure 26.86 Mastership Request: NACK (Device Role[1:0] is not 01 (I3C master)) and NRMRCTL = 0

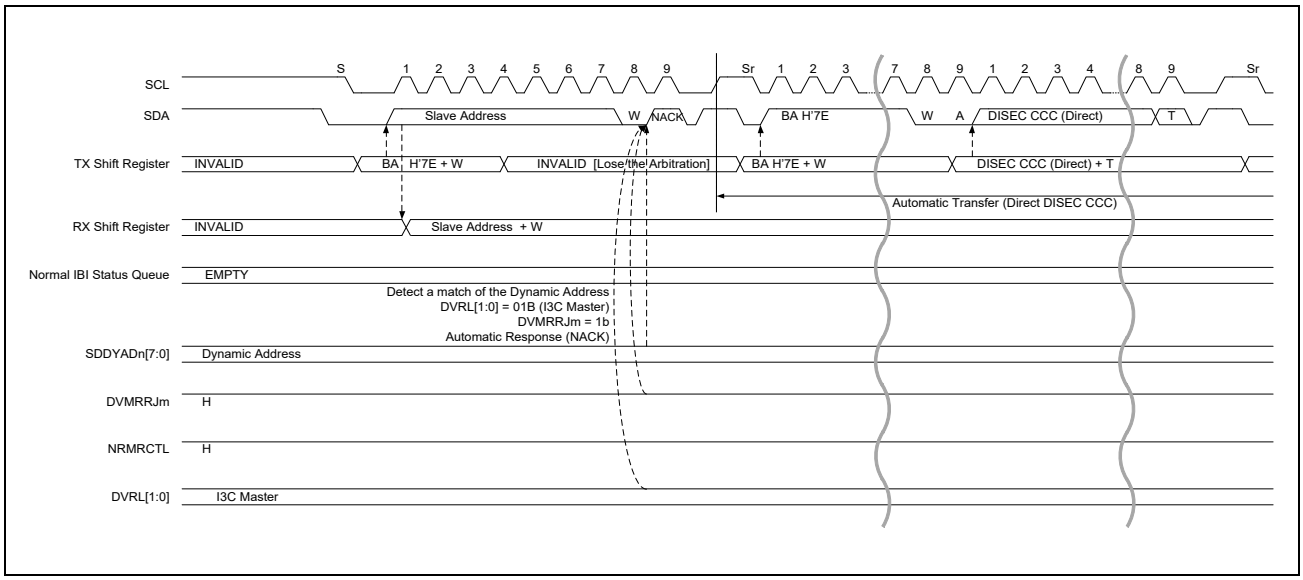


Figure 26.87 Mastership Request: NACK (DVMRRJ = 1) and NRMRCTL = 1 (1/2)

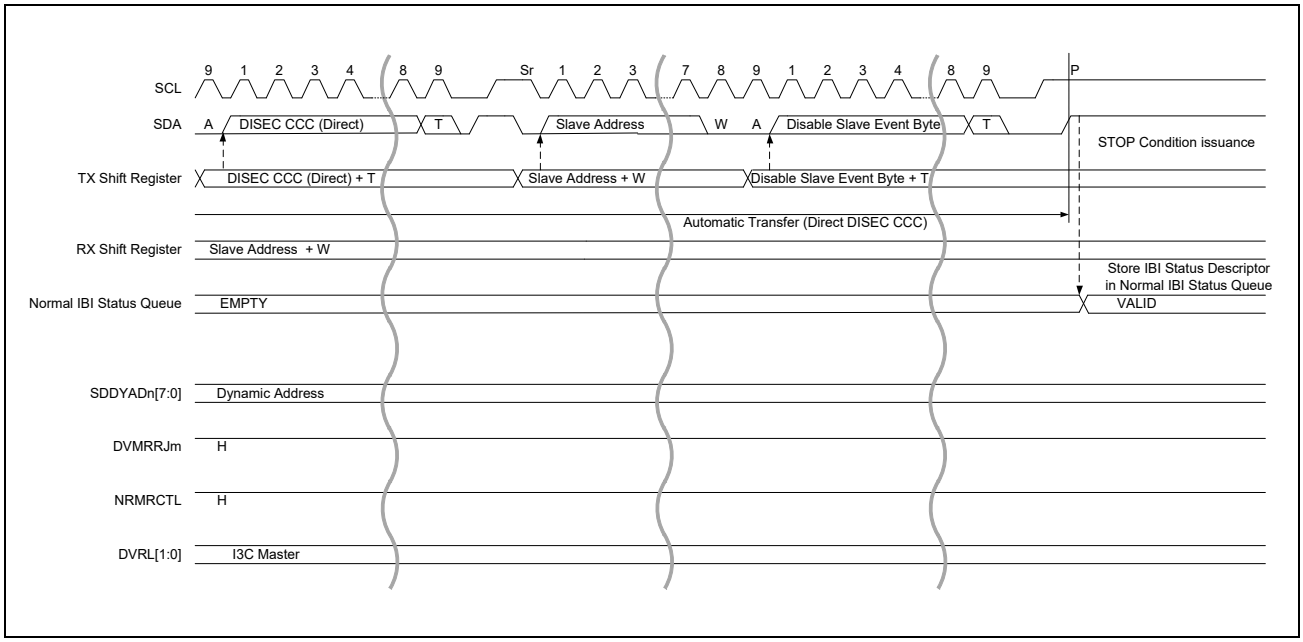


Figure 26.88 Mastership Request: NACK (DVMRRJ = 1) and NRMRCTL = 1 (2/2)

(c) Hot-Join Event

1. Detect the Hot-Join Address (H'02) with RnW bit Low in the Address Header.
2. When BCTL.HJACKCTL = 1,
   It operates in the following order.
   (a) Responds NACK.
   (b) Issues Repeated START condition and automatically issues Broadcast DISEC CCC.
   (c) Issues the STOP condition.
   When BCTL.HJACKCTL = 0,
   Responds ACK, then issues STOP condition.
3. After issues of STOP condition, NACK response:
   – If IBINCTL.NRHJCTL = 0, the IBI Status Descriptor is not stored into the IBI Status Queue.
   – If IBINCTL.NRHJCTL = 1, store the IBI Status Descriptor into the IBI Status Queue.
   ACK response:
   Stores the IBI Status Descriptor into the IBI Status Queue.

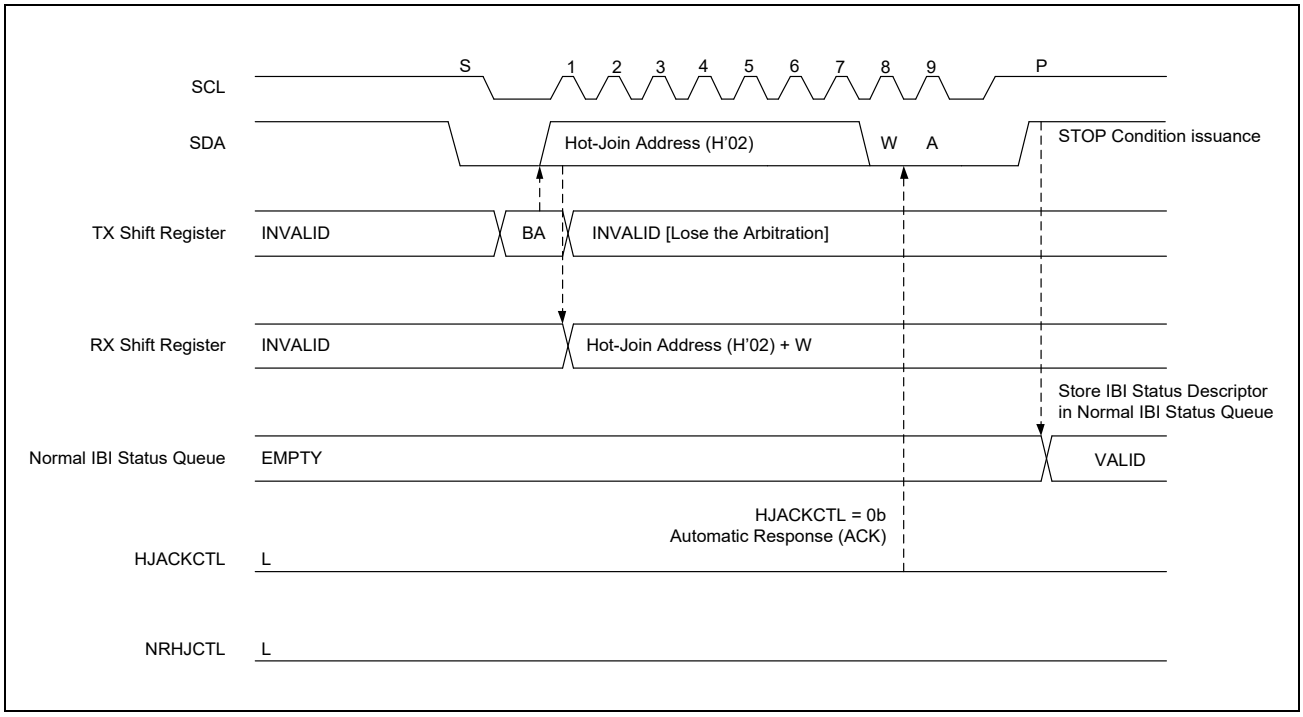


Figure 26.89 Hot-join Event: ACK

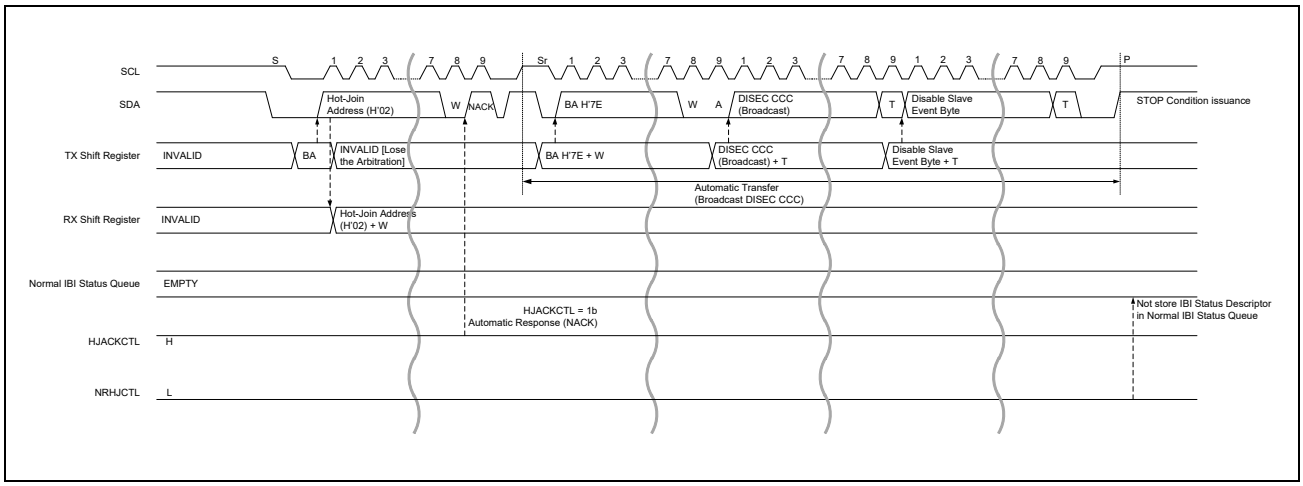


Figure 26.90 Hot-join Event: NACK (HJACKCTL = 1) and NRHJCTL = 0

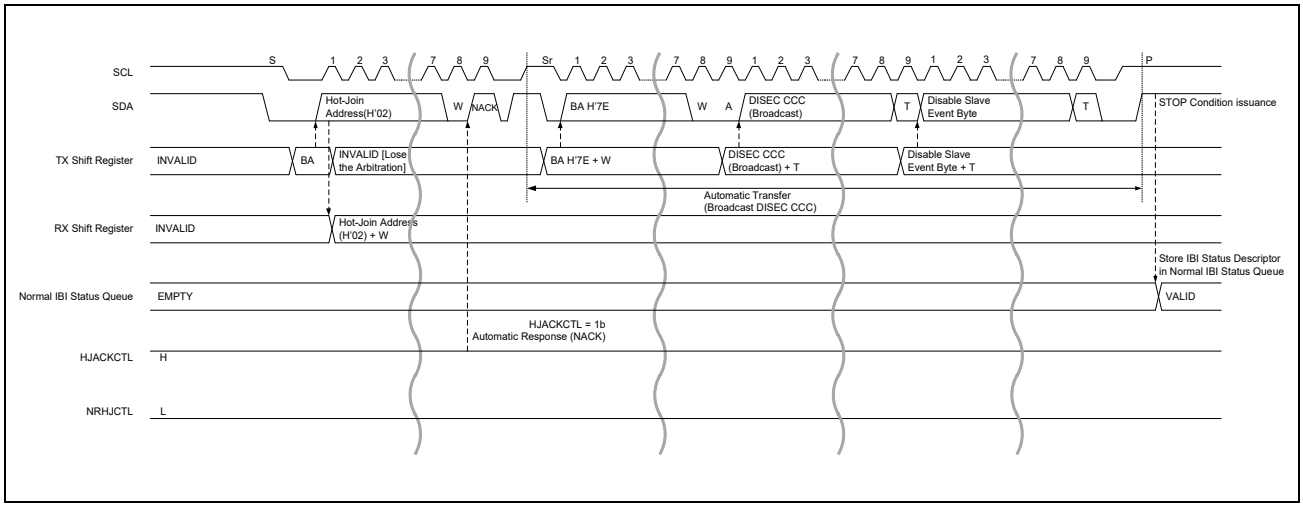


Figure 26.91 Hot-join Event: NACK (HJACKCTL = 1) and NRHJCTL = 1

(9) Timing Control [I3C mode]

I Timing Control is a function that enables Master to efficiently read data from Slave by controlling and grasping the timing at which the Slave Device samples the sensor value.

For details, refer to *5.1.8, Timing Control of MIPI I3C Specificaton v1.0*.

In this IP, timing Control supports the following three modes.

- Sync Mode
- Async Mode 0 (Asynchronous Basic Mode)
- Async Mode 1 (Asynchronous Advanced Mode)

The resources for realizing Timing Control in each mode are described below.

(a) Sync Mode

1) I3C Master

The measured value of Delay Time is sent as a DT message (SETXTIME CCC with DT subcommand) following the ST message.

2) I3C Slave

The sampling timing is recalculated by the corrected T_ph.

(b) Async Mode 0 (Asynchronous Basic Mode)

For timing control in Async Mode 0, set the ATCTL register if necessary.

1) I3C Master

This IP has counters of MREF (32bit) and MC2(16bit) for Async Mode 0.

- MREF Counter

  When ATCCNTE.ATCE is enabled, it starts counting.
  It captures as MREF on the SCL rise edge next to ACK for the IBI transmitted from the I3C Slave.

- MC2 Counter

  After enabling ATCCNTE.ATCE, it counts up from the SCL rise edge next to ACK for the IBI transmitted from the I3C Slave to the SCL rise edge next to the Tbit after Mandatory Byte and capture it as MC2.

The MREF and MC2 capture values are stored next to the IBI Status Descriptor when IBI is received from the I3C Slave with the DATBASm.DVIBITS bit set to 1.
The MREF counter implemented in this IP is 32-bit counter.

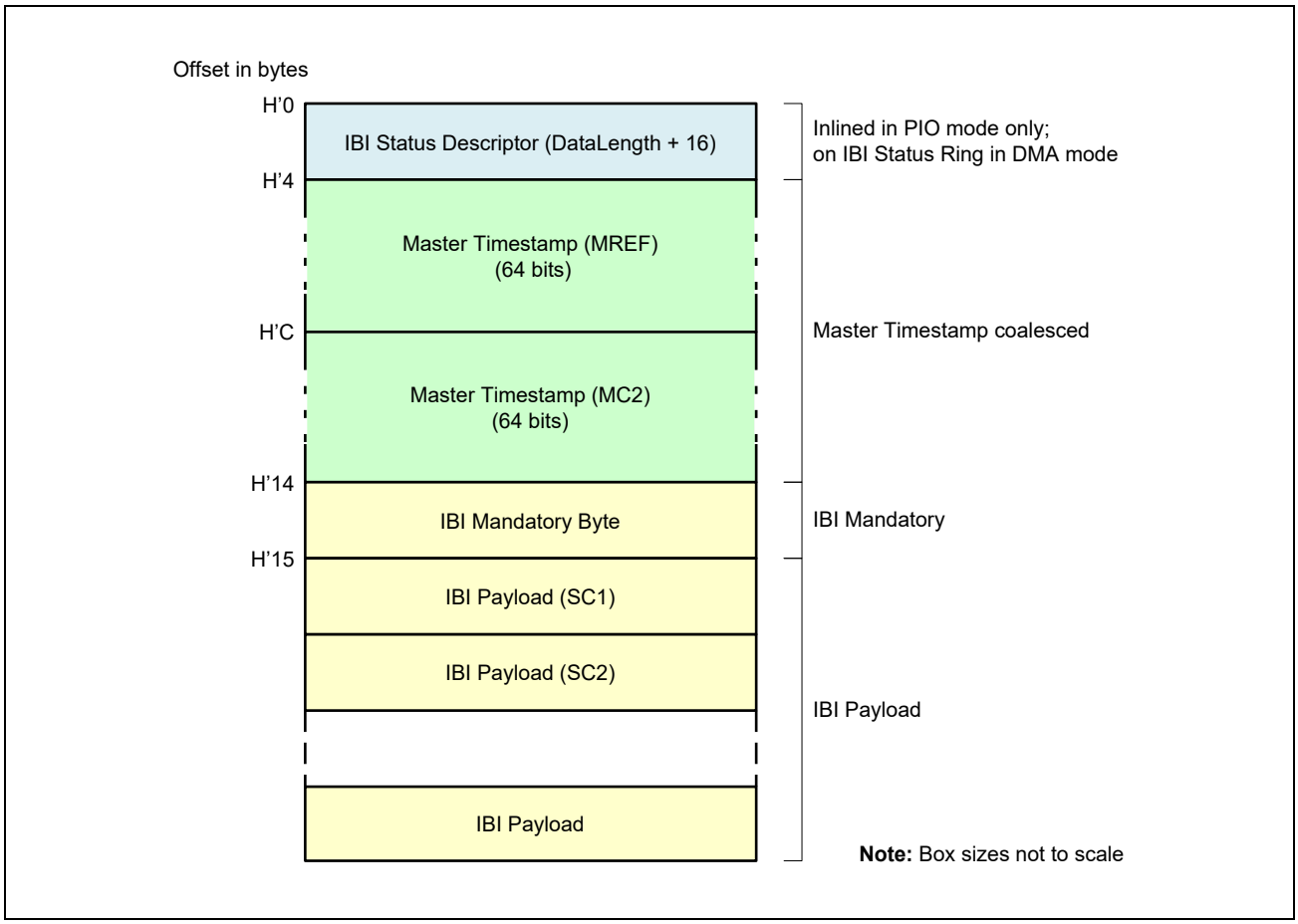


Figure 26.92 Master Timestamp Counters for IBI Event

2) I3C Slave

This IP has counters of SC1 (16 bits) and SC2 (8 bits) for Async Mode 0.

- SC1 Counter
  After enabling ATCCNTE.ATCE, it counts up from SC1 count trigger to SCL rise edge next to ACK for the IBI and capture it as SC1.
- SC2 Counter
  After enabling ATCCNTE.ATCE, it counts up from SCL rise edge next to ACK for the IBI transmitted from I3C Slave to SCL rise edge next to Tbit after Mandatory Byte and capture it as SC2.

When the CETSS.ASYNE[0] bit = 1 and the ITS bit in Command Descriptor for issuing IBI is 1, the SC1 and SC2 capture values are transmitted following the IBI Mandatory Byte as shown in the following figure.

If the SC1 and SC2 counters overflow, H'FFFF and H'FF are captured and CETSS.ICOVF is set 1.

The DATA_LENGTH[15:0] bits value of the Command Descriptor sets a value obtained by adding the number of data of SC1 and SC2 to the number of transmission data.

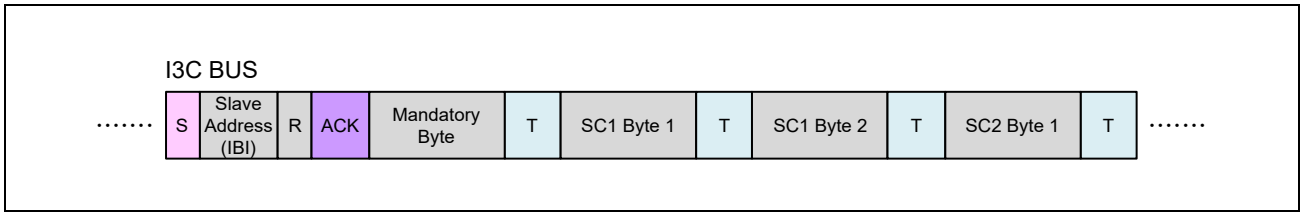


Figure 26.93 Example of Asynchronous Mode 0 Timestamp Data Transfer

(10) Async Mode 1 (Asynchronous Advanced Mode)

For timing control in Async Mode 1, set the ATCTL register if necessary.

(a) I3C Master

This IP has counters of MREF (32 bits), MSyncCNT (32 bits) and MC2 (16 bits) for Async Mode 1.

- MREF Counter

  When ATCCNTE.ATCE is enabled, it starts counting.
  It captures as MREF at the SCL rise edge next to ACK for the IBI transmitted from the I3C Slave.
  The MREF extension method described in "1) I3C Master" of **Section 26.3.2.3(9)(b), Async Mode 0 (Asynchronous Basic Mode)** is also possible in Async Mode 1.
- MSyncCNT Counter

  When ATCCNTE.ATCE is enabled, it starts counting.
  It captures as MSyncCNT for each aME (SDA falling edge of START condition) and store it in the capture register.
- MC2 Counter

  After enabling ATCCNTE.ATCE, it counts up from the SCL rise edge next to ACK for the IBI transmitted from the I3C Slave to the SCL rise edge next to the Tbit after Mandatory Byte and capture it as MC2.

The MREF and MC2 capture values are stored next to the IBI Status Descriptor when the IBI is received from the I3C Slave with the DATBASm.DVIBITS bit set to 1. (Same as Async Mode 0)

(b) I3C Slave

This IP has counters of SC1 (16 bits), SC2 (8 bits) and aME_TICK (8 bits) for Async Mode 1.

- SC1 Counter

  After enabling ATCCNTE.ATCE, it counts up from SC1 count trigger*1 to the first aME and capture it as SC1.

Note 1. SW or external trigger can be selected by selection bits.
The SC1 count trigger select method described in “2) I3C Slave” of **Section 26.3.2.3(9)(b), Async Mode 0 (Asynchronous Basic Mode)** is also possible in Async Mode 1.

- SC2 Counter

  After enabling ATCCNTE.ATCE, it counts up from SCL rise edge next to ACK for the IBI transmitted from I3C Slave to SCL rise edge next to Tbit after Mandatory Byte and capture it as SC2.

- aME_TICK Counter

  After enabling ATCCNTE.ATCE, it counts every aME, and capture it as aME_TICK at the SCL rise edge next to ACK for the IBI.
  The aME_TICK counter is cleared on the first aME after the SC1 count trigger.

When the CETSS.ASYNE[1] bit = 1 and the ITS bit in Command Descriptor for issuing IBI is 1, the SC1, SC2 and aME_TICK capture values are transmitted following the IBI Mandatory Byte as shown in the following figure.

If the SC1 and SC2 counters overflow, H’FFFF and H’FF are captured and CETSS.ICOVF is set 1.

The DATA_LENGTH[15:0] bits value of the Command Descriptor sets a value obtained by adding the number of data of SC1, SC2 and aME_TICK to the number of transmission data.

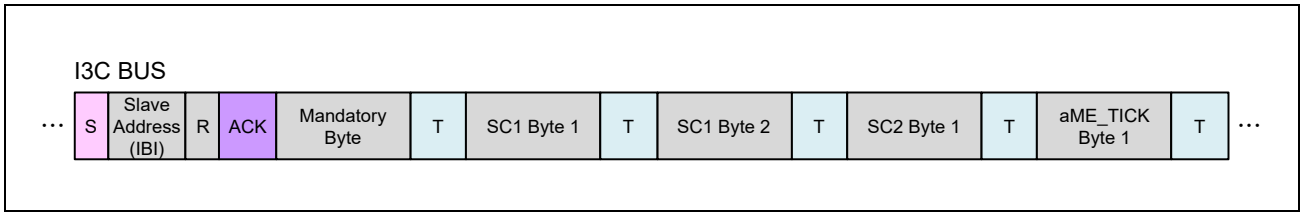


Figure 26.94 Example of Asynchronous Mode 1 Timestamp Data Transfer

(11) Port Control

(a) Extra SCL Clock Cycle Output Function

In master mode, I3C module has a facility for the output of extra SCL clock cycles to release the SDAn line of the slave device from being held at the low level due to the master being out of synchronization with the slave device.

This function is mainly used in master mode to release the SDAn line of the slave device from the state of being fixed to the low level by including extra cycles of SCL output from I3C with single cycles of the SCL clock as the unit in the case of a bus error where I3C cannot issue a Repeated START condition or a STOP condition because the slave device is holding the SDAn line at the low level. Do not use this facility in normal situations. Using it when communications are proceeding correctly will lead to malfunctions.

When the OUTCTL.EXCYC bit is set to 1, an additional clock pulse at the frequency set by the REFCKCTL.IREFCKS[2:0] bits and the STDBR.SBRHO[7:0] and STDBR.SBRLO[7:0] registers is output from the SCLn pin. After output of this clock pulse, the EXCYC bit automatically becomes 0. After confirming that the EXCYC bit is 0, wait for the setup time of the Repeated START condition or STOP condition, and then confirm the detection of the Repeated START condition or STOP condition. If the Repeated START condition or STOP condition is not detected, consecutive additional clock pulses can be output by writing 1 to the EXCYC bit again.

When I3C module is in master mode and the slave device is holding the SDAn line at the low level because synchronization with the slave device has been lost due to the effects of noise, etc., the output of a Repeated START condition or a STOP condition is not possible. The facility for output of an extra cycle of the SCL clock can be used to output extra cycles of SCL one by one to make the slave device release the SDAn line from being held at the low level, thus recovering the bus from an unusable state. Release of the SDAn line by the slave device can be monitored by reading the SDILV bit in PRSTDBG. After the SDAn line has been released by the slave device, the preset of a Repeated START condition or a STOP condition is issued.

Use this function with the BFCTL.MALE bit set to 0 (master arbitration-lost detection is disabled).

[Output conditions for using the EXCYC bit in OUTCTL]

- When the bus is free (BFREF flag in BCST = 1) or in master mode (CRMS bit = 1 in PRSST and BFREF flag = 0 in BCST)
- When the communication device does not hold the SCLn line low

Figure 26.95 shows the operation timing of the extra SCL clock cycle output function (EXCYC bit).

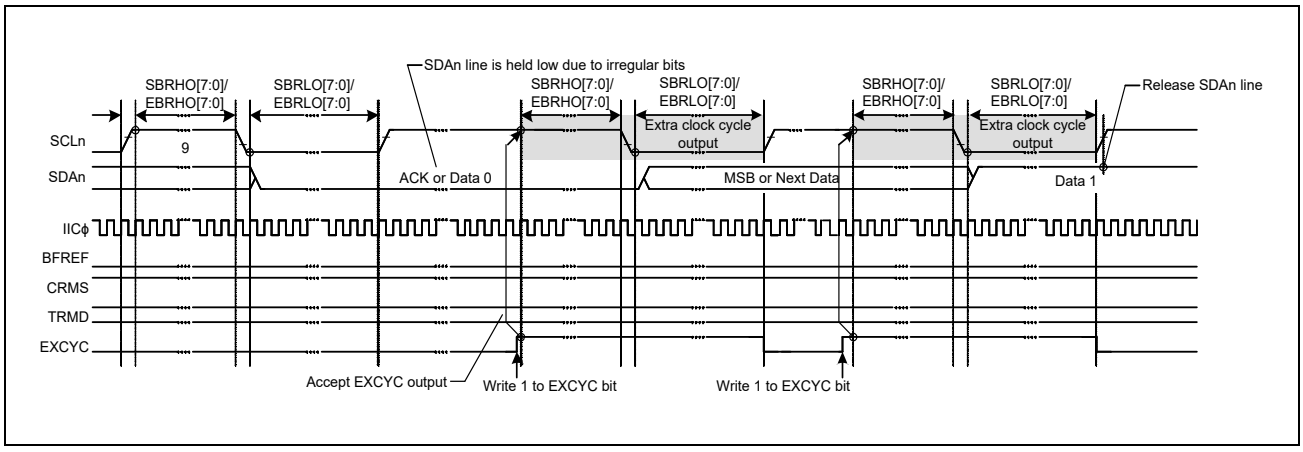


Figure 26.95 Extra SCL Clock Cycle Output Function (EXCYC Bit)

(12) Common Command Codes (CCC) [I3C mode]

Command Code H'E0 to H'FE Vendor Extension-Direct CCCs defined are not supported. The MIPI reserved area and Vendor Extension area of Command Code are not supported. Do not use an unsupported CCC when using this module with I3C Slave.

If the I3C Master must use an unsupported CCC, use the added CCC after using ENTASx CCC to put this module to Sleep mode.

26.3.2.4 Error Detection

(1) SDR Error Detection and Recovery Methods for I3C Slave Devices [I3C mode]

The seven error types summarized in **Table 26.14** are supported for all I3C slave devices. Each error type is further explained below the table.

Table 26.14 SDR Slave Error Types

Error Type	Description	Error Detection Method	Error Recovery Method
S0	Broadcast address/W (= H'7E/W) or Dynamic address/RW	Detect any of the following: H'3E/W H'5E/W H'6E/W H'76/W H'7A/W H'7C/W H'7F/W H'7E/R	Enable HDR EXIT Detector and ignore all other patterns
S1	CCC code	Parity check, using T-Bit	Enable HDR EXIT detector and neglect other patterns
S2	Write data	Parity check, using T-Bit	Enable STOP detector and neglect other patterns
S3	Assigned address during Dynamic address arbitration	Parity check, using PAR Bit	Generate NACK (after PAR), then wait for another Repeated START and 7E/R to re-transmit the Provisional ID
S4	H'7E/R after Sr during Dynamic address arbitration	Detect any value other than H'7E/R after Sr during Dynamic Address Arbitration	Generate NACK (after H'7E/R), then enable STOP Detector and ignore all other patterns
S5	Transaction after detecting CCC	Detect illegally formatted CCC	Generate NACK (after Slave Address), then enable STOP Detector and ignore all other patterns
S6 (optional)	Monitoring error	Slave detects (through monitoring) that transmitted Data differs from what it intended to transmit (Does not apply during Dynamic address arbitration)	Stop the transmission, then enable STOP Detector and ignore all other patterns

(2) SDR Error Detection and Recovery Methods for I3C Master Devices [I3C mode]

The two error types summarized in **Table 26.15** are supported for all I3C master devices. Each error type is further explained below the table.

Table 26.15 SDR Master Error Types

Error Type	Description	Error Detection Method	Error Recovery Method
M0	Transaction after sending CCC	Detect illegally formatted CCC	Stop the transmission, then send STOP and retry the transmission.
M1 (optional)	Monitoring error	Master detects (through monitoring) transmitted data different from what it intended to transmit (Does not apply during Dynamic address arbitration)	Stop the transmission, then send STOP and retry the transmission.
M2	No response to Broadcast address (H'7E)	Master detects NACK after Broadcast address (H'7E) transmission	Upon detection of NACK, master transmits HDR exit pattern followed by STOP

(3) Timeout Error Detection

I3C includes a timeout function for detecting when the SCLn line has been stuck longer than the predetermined time. I3C can detect an abnormal bus state by monitoring that the SCLn line is stuck low or high for a predetermined time.

The timeout function monitors the SCLn line state and counts the low-level period or high-level period using the internal counter. The timeout function resets the internal counter each time the SCLn line changes (rising or falling), but continues to count unless the SCLn line changes. If the internal counter overflows due to no SCLn line change, I3C can detect the timeout and report the bus hung state.

This timeout function is enabled when BSTE.TODE = 1. It detects a hung state that the SCLn line is stuck low or high during the following conditions: (When TMOCTL.TOMDS[1:0] = 00b)

- The bus is busy (BCST.BFREF = 0) in master mode (PRSST.CRMS = 1).
- I3C's own slave address is detected (SVST register is not H'0000) and the bus is busy (BCST.BFREF = 0) in slave mode (PRSST.CRMS = 0).
- The bus is free (BCST.BFREF = 1) while generation of a START condition is requested (CNDCTL.STCND = 1).

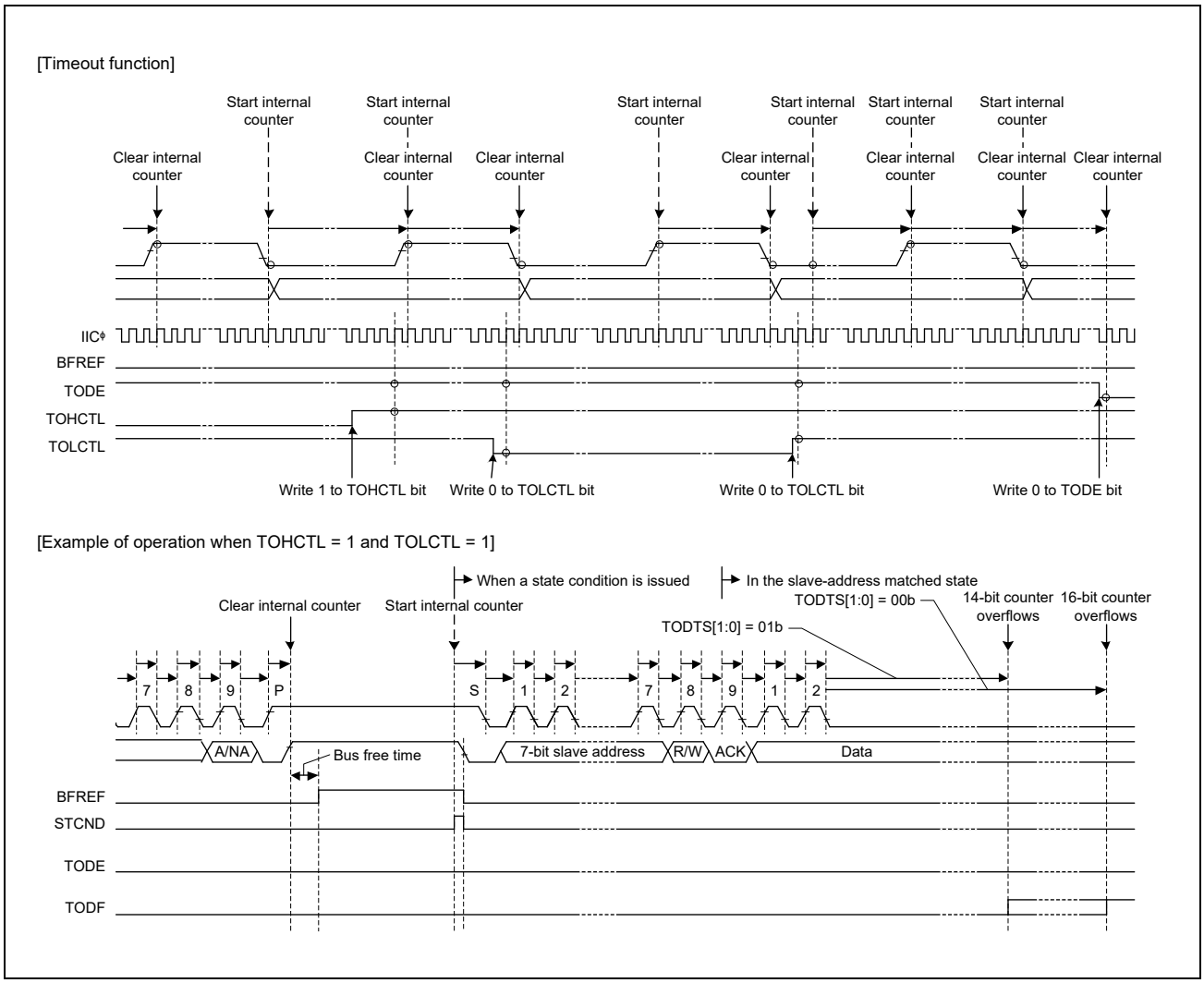


Figure 26.96 Timeout Error Detection (TODE, TODTS[1:0], TOHCTL, and TOLCTL bits)

(4) Resume Operation [I3C mode]

I3C enters the Halt state as a result of any type of error occurring in a transfer.

The error type is indicated by the field ERR_STATUS in Response Descriptor or Receive Status Descriptor. After I3C has entered the Halt state, the user must write the value 1 to the RSM bit to resume operation. I3C shall auto clear the RSM bit once it has initiated the next Command transfer or detected the START condition.

(5) Abort Operation [I3C mode]

When the BCTL.ABT bit is set to 1, I3C relinquish control of the bus before completing the currently issued transfer. In response to an abort request, I3C issues the STOP condition on the bus after the complete data byte is transferred or received. After I3C has aborted, the user shall clear the BCTL.ABT bit to allow operation on the bus.

NOTE

For Read transaction, when BCTL.ABT is set to 1, that receive data is stored in Receive data buffer.

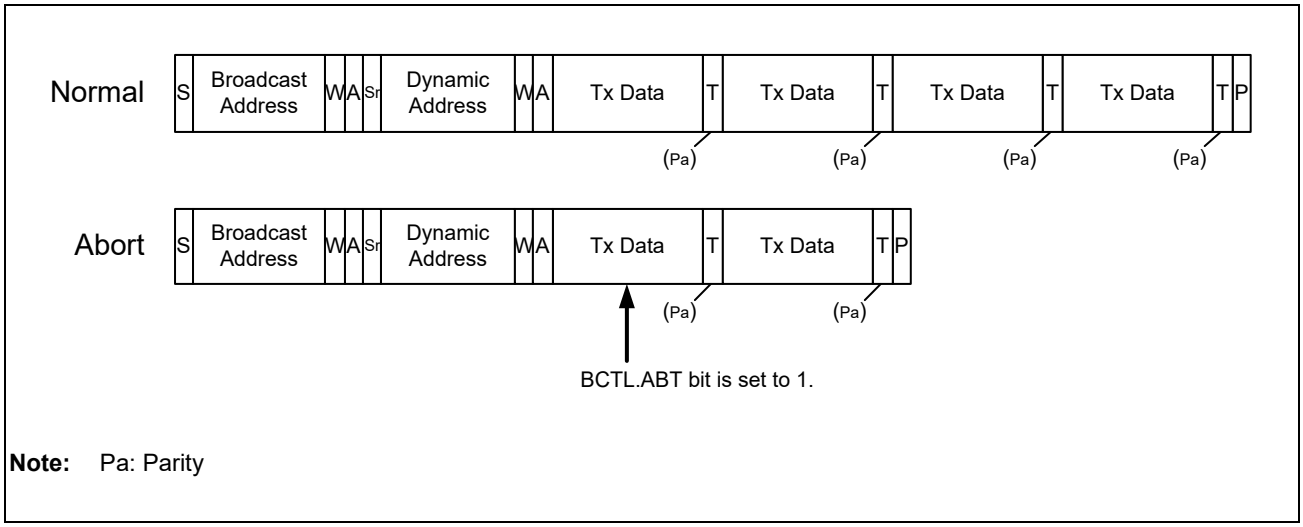


Figure 26.97 Abort Operation of SDR Write Transfer

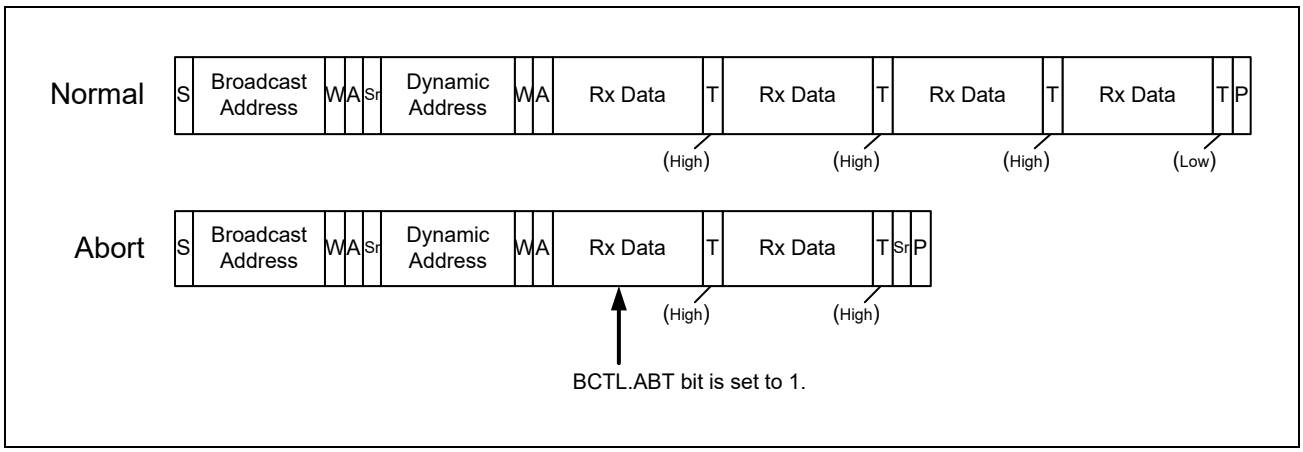


Figure 26.98 Abort Operation of SDR Read Transfer

(6) Error Recovery Operation [I3C mode]

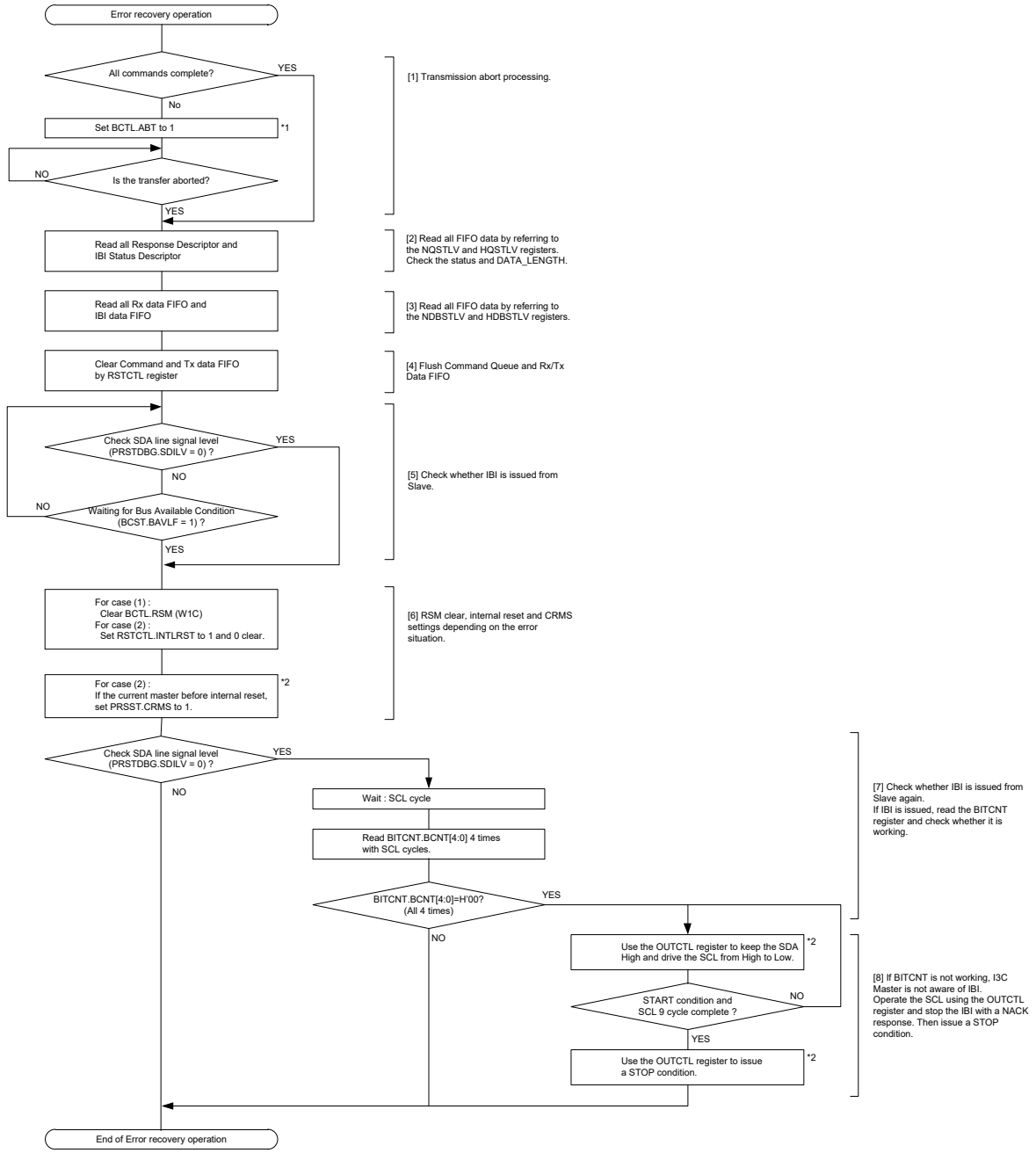
When an error occurs, the INST.INEF, NTST.TEF and NTST.TABTF flags are set to 1 according to the cause of the error, or the interrupts associated with each flag are asserted (when detection and interrupts are enabled.)

There is a possibility of communication error or internal module error.

The I3C master must perform an error recovery flow according to the following case:

- When TEF is detected.

Figure 26.99 and **Figure 26.100** show the error recovery flow.



Note 1. If the transfer is not aborted due to an error, set the BCTL.ABT bit to 1 and then abort the transfer.

Note 2. When writing to the CRMS bit, set PRSST.PRSSTWP bit to 1 at the same time.

Note 3. When writing to the SDOC and SCOC bit, set OUTCTL.SOCWP bit to 1 at the same time.

Figure 26.99 Example of Error Recovery Operation Flowchart for I3C Master

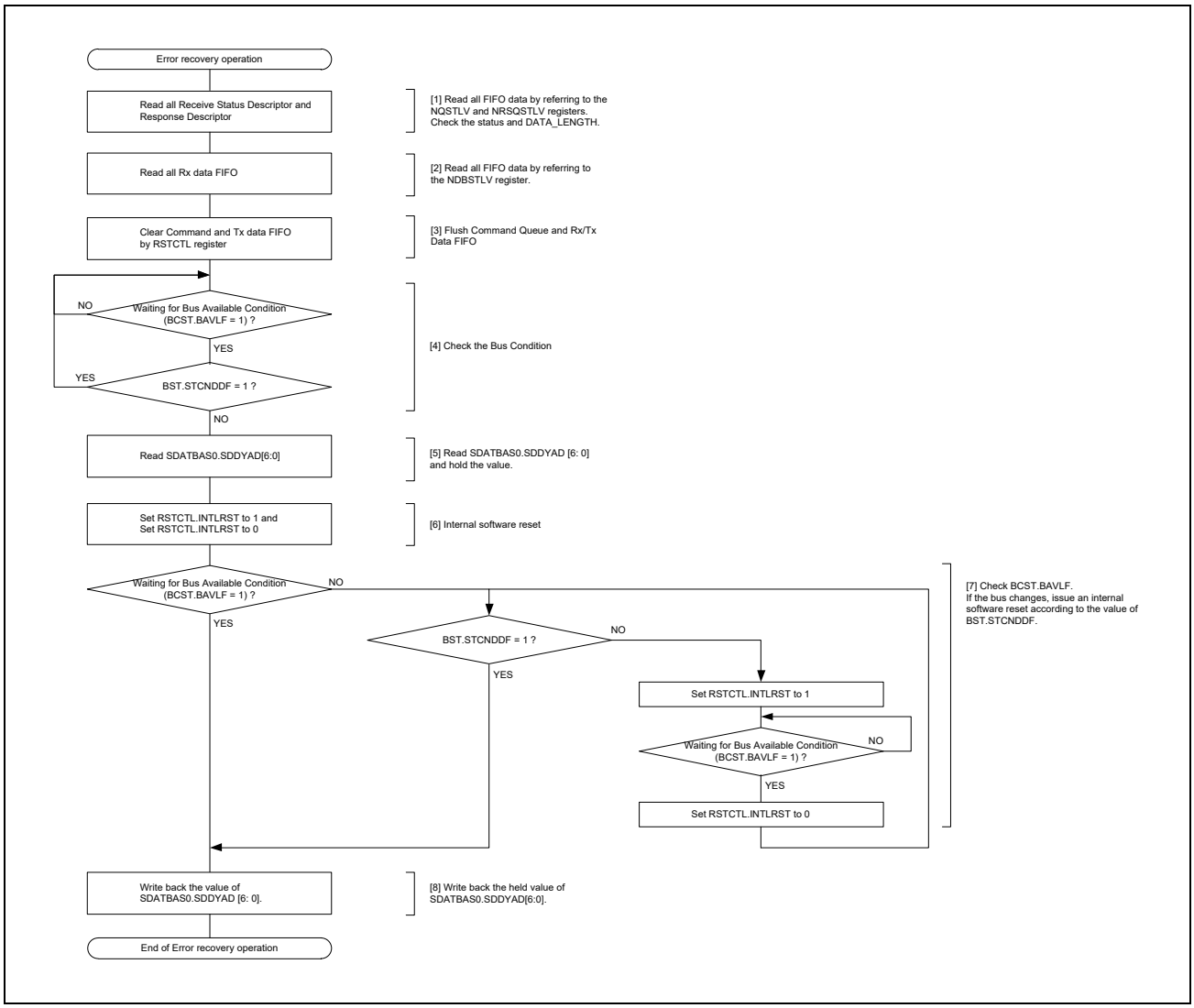


Figure 26.100 Example of Error Recovery Operation Flowchart for I3C Slave

26.3.2.5 Low Power Function [I²C/I3C common]

(1) Wake Up function [I²C mode]

This IP is equipped with the Wake-up function that causes the microcomputer to transition from low power consumption mode with system clock is stopped (software standby mode, snooze, etc.) to the normal operation. The Wake-up function is used to generate a Wake-Up interrupt signal when the received data matches the address set to Wake-Up interrupt factor also receives data in a state where the operating clock (PCLK/TCLK) is stopped (PCLK/TCLK asynchronous operation). This wake-up interrupt signal causes the microcomputer to transition to the normal operation. After Wake-Up interrupt occurs, switch this IP to PCLK/TCLK synchronous operation, it will be able to continue the communication operation.

The Wake-up function has four wake-up operation modes (normal WU mode 1, normal WU mode 2, command recovery mode, and EEP response mode). The table below describes the behavior in these four wake-up operation modes.

Table 26.16 Wake-Up Operation Mode

	ACK Response Timing	ACK Type Responded before Recovery to PCLK/TCLK Synchronous Operation	SCL State before Recovery to PCLK/TCLK Synchronous Operation
Normal WU mode 1	Before recovery to PCLK/TCLK synchronous operation*1	ACK	Fixed to “L”
Normal WU mode 2	After recovery to PCLK/TCLK synchronous operation*2	Before recovery: no response (NACK level retained) After recovery: ACK response	Fixed to “L”
Command recovery mode	Before recovery to PCLK/TCLK synchronous operation*1	ACK	Open
EEP response mode	Before recovery to PCLK/TCLK synchronous operation*1	NACK	Open

Note 1. Switching timing from PCLK/TCLK asynchronous operation to PCLK/TCLK Synchronous operation is the fall of the 9th clock of SCL.

Note 2. Switching timing from PCLK/TCLK asynchronous operation to PCLK/TCLK Synchronous operation is the fall of the 8th clock of SCL.

The following is able to select as Wake-Up interrupt factor.

- Host address detection (valid when SVCTL.HOAE = 1)
- General call address detection (valid when SVCTL.GCAE = 1)
- Slave address 0*1 detection (valid when SVCTL.SVAE0 = 1)
- Slave address 1*1 detection (valid when SVCTL.SVAE1 = 1)
- Slave address 2*1 detection (valid when SVCTL.SVAE2 = 1)

Note 1. 7-bit address only can be set. Please set SDADLS bit to “0” in SDATBASn.

(a) Normal Wake-Up mode 1

This section describes the behavior, the timing, and a use case of normal WU mode 1.

1. A wake-up interrupt triggered by the match of the slave address makes the transition to the normal operation in the manner described below. Also, the detailed timing is provided in **Figure 26.103**.

   Before Wake-Up recovery: ACK is sent in response to the data received with its own slave address.

   During Wake-Up recovery: ACK response is made at the 9th clock cycle of SCL, and the SCL is held low afterwards.*1

   After Wake-Up recovery: Normal operation continues.

   If the slave address does not match, the SCL line is not held low after the fall of the 9th clock cycle of SCL, and the slave operation continues.

Note 1. Between ninth clock cycle and first clock cycle during Wake-Up recovery, SCSTRCTL.RWE = 1 does not work.

Refer to **Figure 26.101** “Use Case of Normal WU Mode 1” below for a use case.

2. A wake-up interrupt is not generated at the transition to the normal operation if the transition is triggered by a cause (other recovery causes (IRQ)) other than a wake-up interrupt signal generated by a slave address match. BST.WUCNDDF is not set in this case. Carry out the following processing according to **Figure 26.102**.

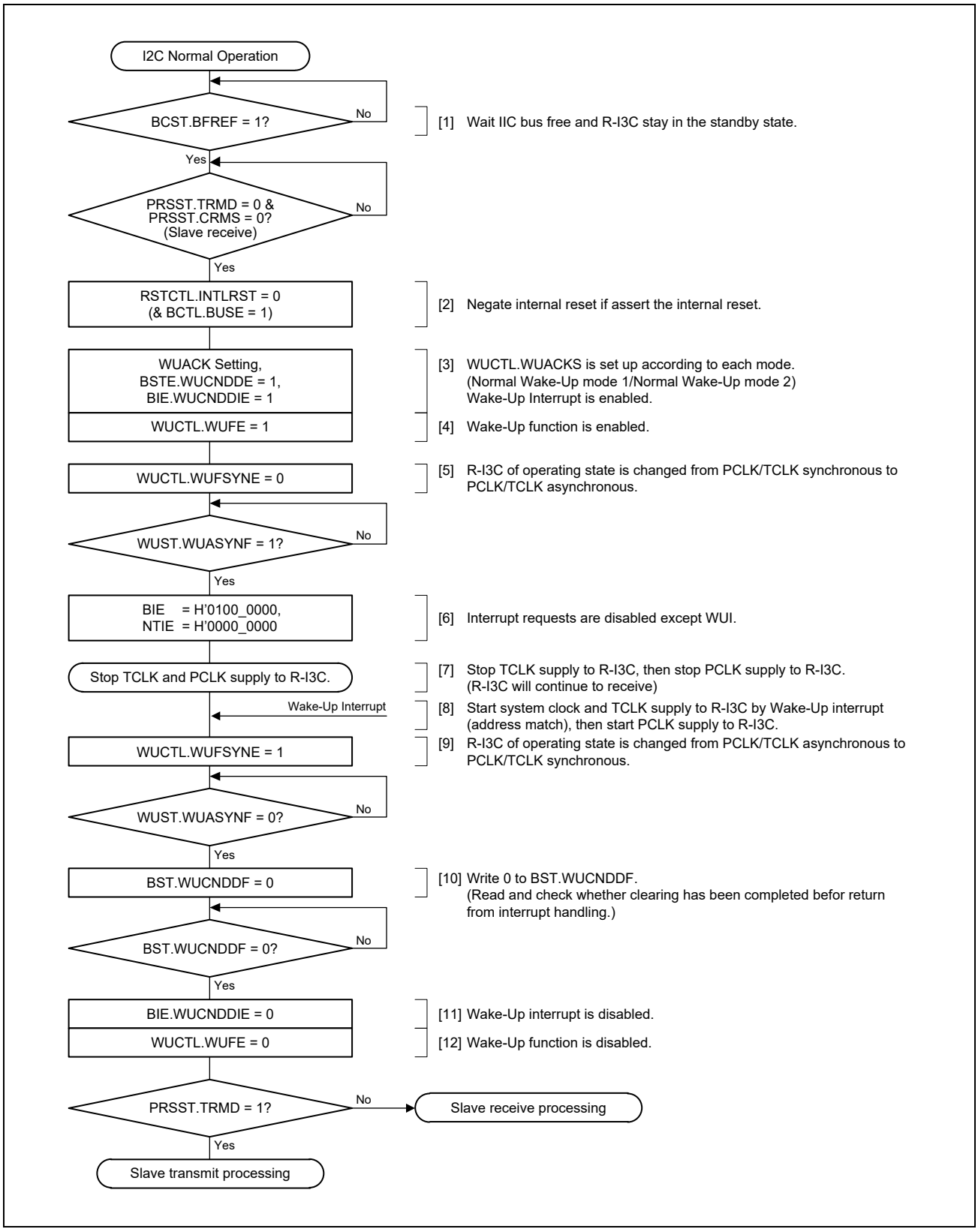


Figure 26.101 Use Case of Normal WU Mode 1
(Wake-Up Recovery by a Wake-Up Interrupt Triggered by the Match of the Slave Address)

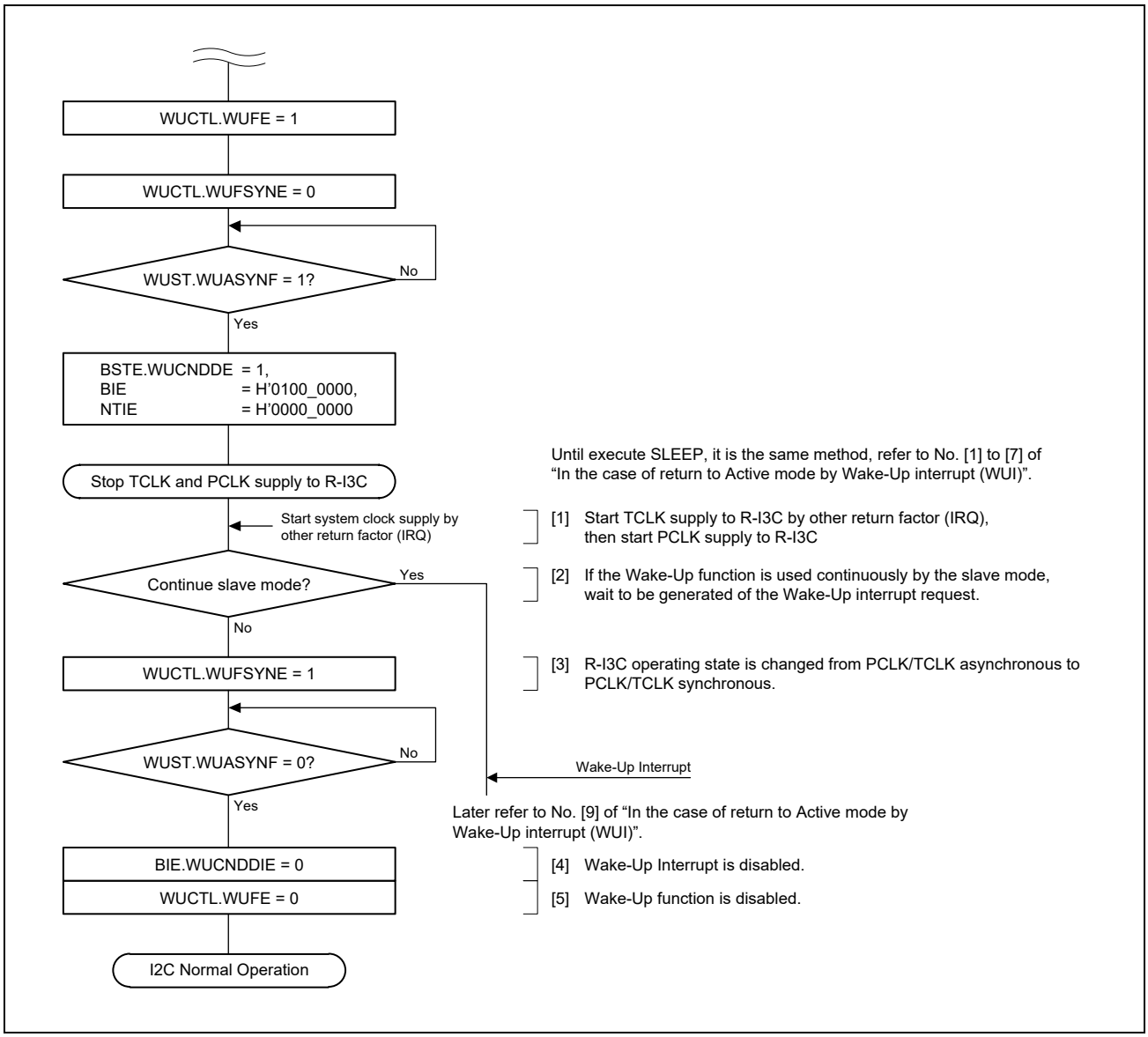


Figure 26.102 Use Case of Normal WU Modes 1 and 2
(Wake-Up Recovery by Other Recovery Causes (IRQ))

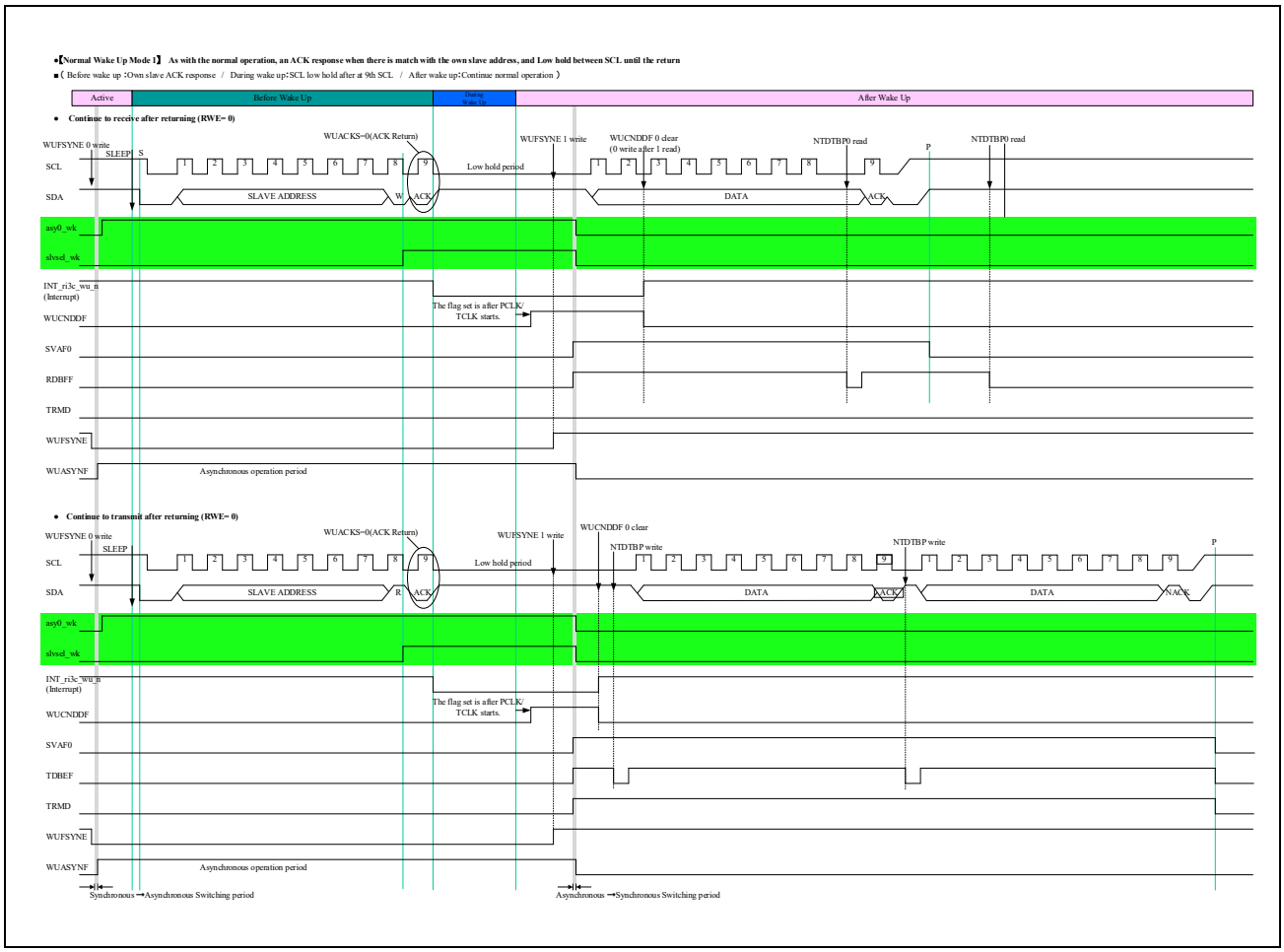


Figure 26.103 Timing of Normal Wake Up Mode 1

(b) Normal Wake Up mode 2

This section describes the behavior, the timing, and a use case of normal WU mode 2.

1. A wake-up interrupt triggered by the match of the slave address makes the transition to the normal operation in the manner described below. Also, the detailed timing is provided in **Figure 26.105**.

   Before Wake-Up recovery: No response to the data received with its own slave address (until 8th SCL cycle end)

   During Wake-Up recovery: Holding the SCL line low during the 8th and 9th clock cycles

   After Wake-Up recovery: Returning ACK at the 9th clock cycle of SCL, and continuing the normal operation

If the slave address does not match, the SCL line is not held low after the fall of the 8th SCL v clock cycle. The slave operation continues.

Refer to **Figure 26.104** “Use Case of Normal WU Mode 2” below for a use case.

2. A wake-up interrupt is not generated at the transition to the normal operation if the transition is triggered by a cause (other recovery causes (IRQ)) other than a wake-up interrupt signal generated by a slave address match. BST.WUCNDDF is not set in this case. Carry out the following processing according to **Figure 26.102**.

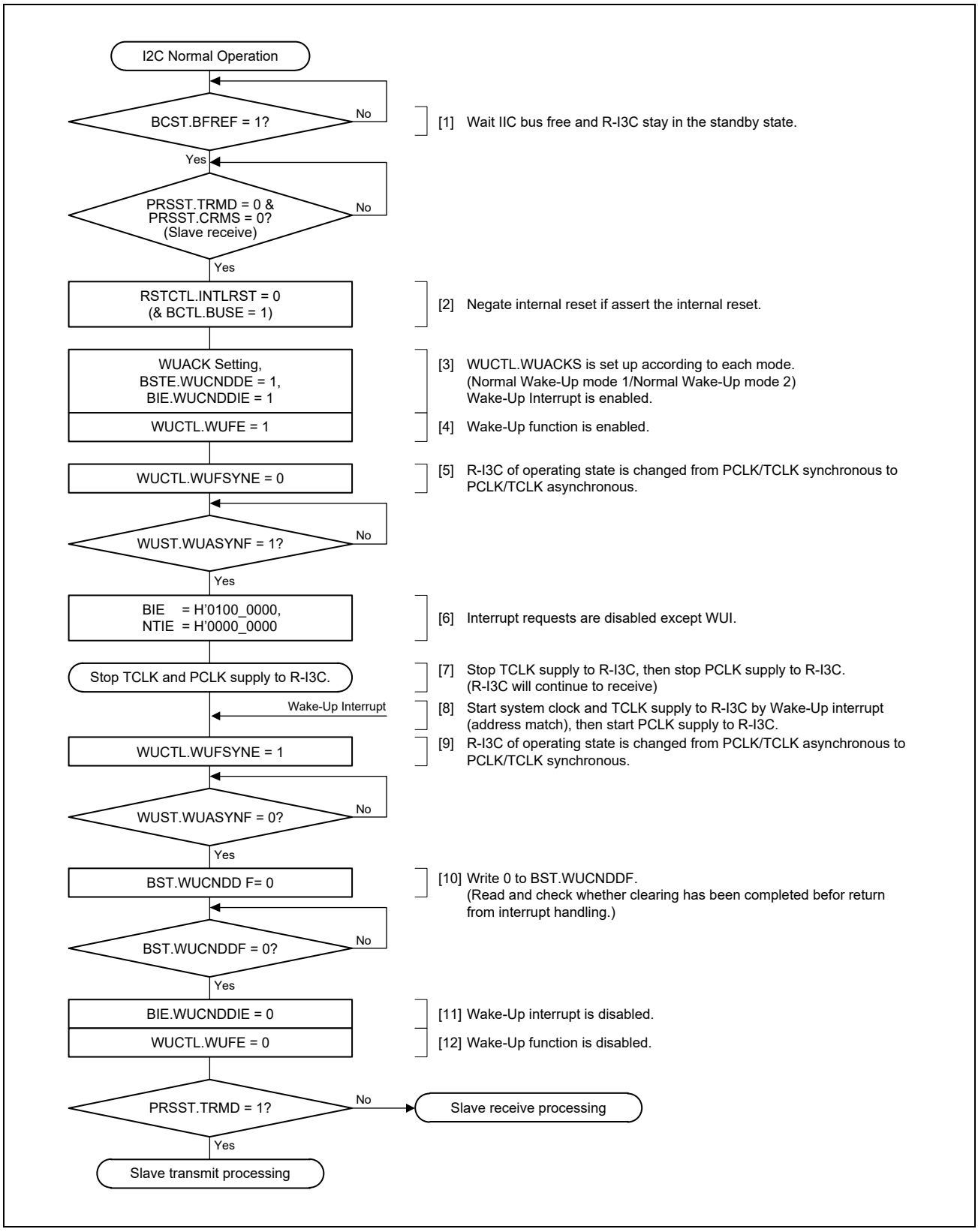


Figure 26.104 Use Case of Normal WU Mode 2
(Wake-Up Recovery by a Wake-Up Interrupt Triggered by the Match of the Slave Address)

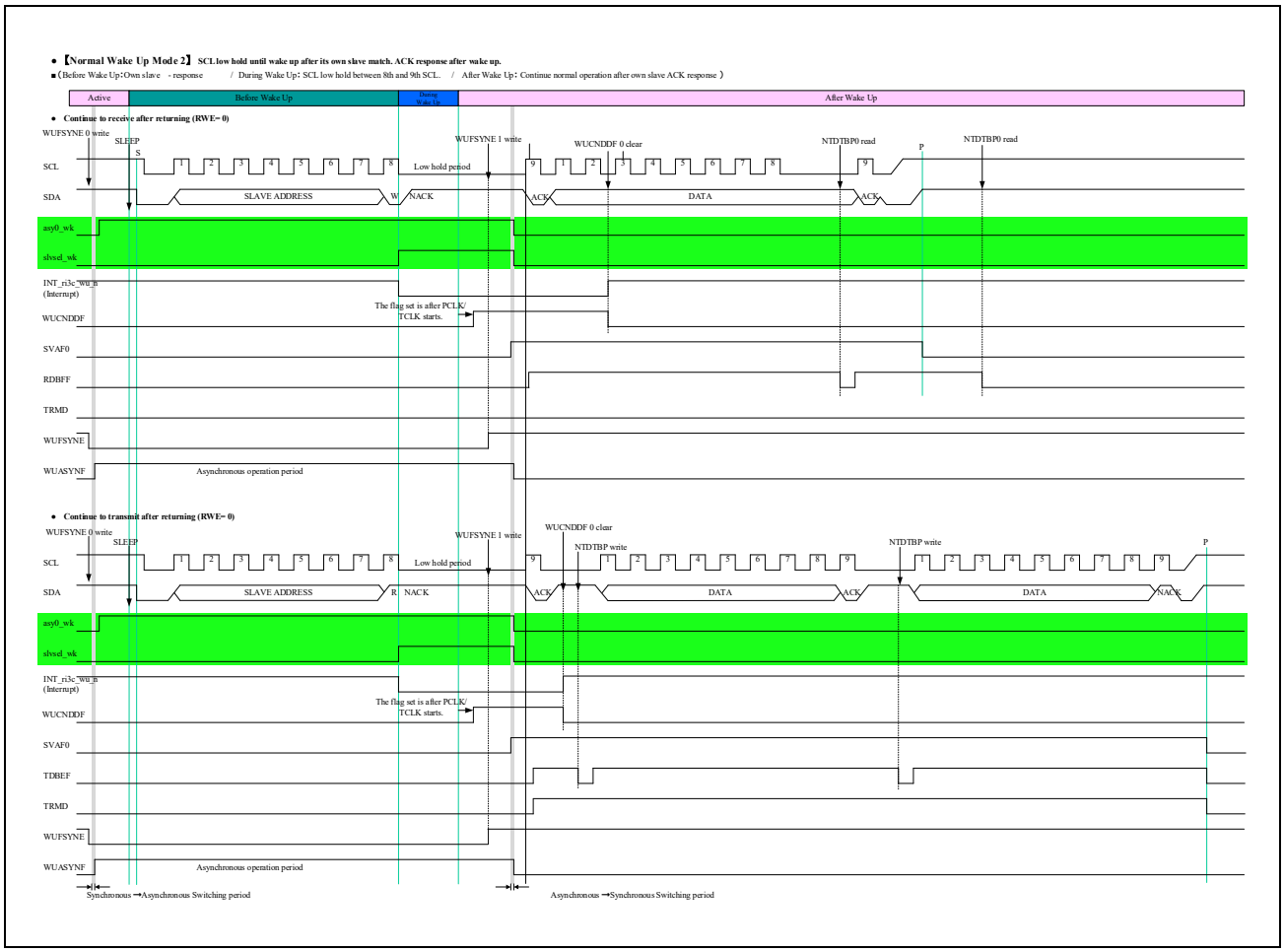


Figure 26.105 Timing of Normal Wake Up Mode2

(c) Command Recovery mode/EEP Response mode (Special Wake Up mode)

In the command recovery mode and EEP response mode, the SCL line is not held low during the wake-up recovery period (after the rise of the 9th clock cycle of SCL), so other I^2C/I3C devices can use the I^2C bus during this period.

This section describes the behavior, the timing, and use cases of the command recovery mode and the EEP response mode.

1. A wake-up interrupt triggered by the match of the slave address makes the transition to the normal operation in the manner described below. Also, the detailed timing is provided in **Figure 26.108**.

   Before Wake-Up recovery: In response to the data received with its own slave address, ACK (command recovery mode) or NACK (EEP response mode) is returned.

   During Wake-Up recovery: The SCL line is not held low.

   After Wake-Up recovery: Normal operation continues after this IP initial setting.

If the slave address does not match, the slave operation continues.

NOTES

1. Because the SCL line is not held low during wake-up recovery, the transmission/reception of the data that follows the slave address is not possible.
2. The command recovery mode and the EEP response mode are internal reset (RSTCTL.INTLRST= 1) states. Therefore, the match of the slave address does not set the SVST flags (HOAF, GCAF, and SVAF2, SVAF1, SVAF0).

Refer to **Figure 26.106** “Use Case of Command Recovery Mode and EEP Response Mode” below for a use case.

2. A wake-up interrupt is not generated at the transition to the normal operation if the transition is triggered by a cause (other recovery causes (IRQ)) other than a wake-up interrupt signal generated by a slave address match. BST.WUCNDDF is not set in this case. Carry out the following processing according to **Figure 26.107**.

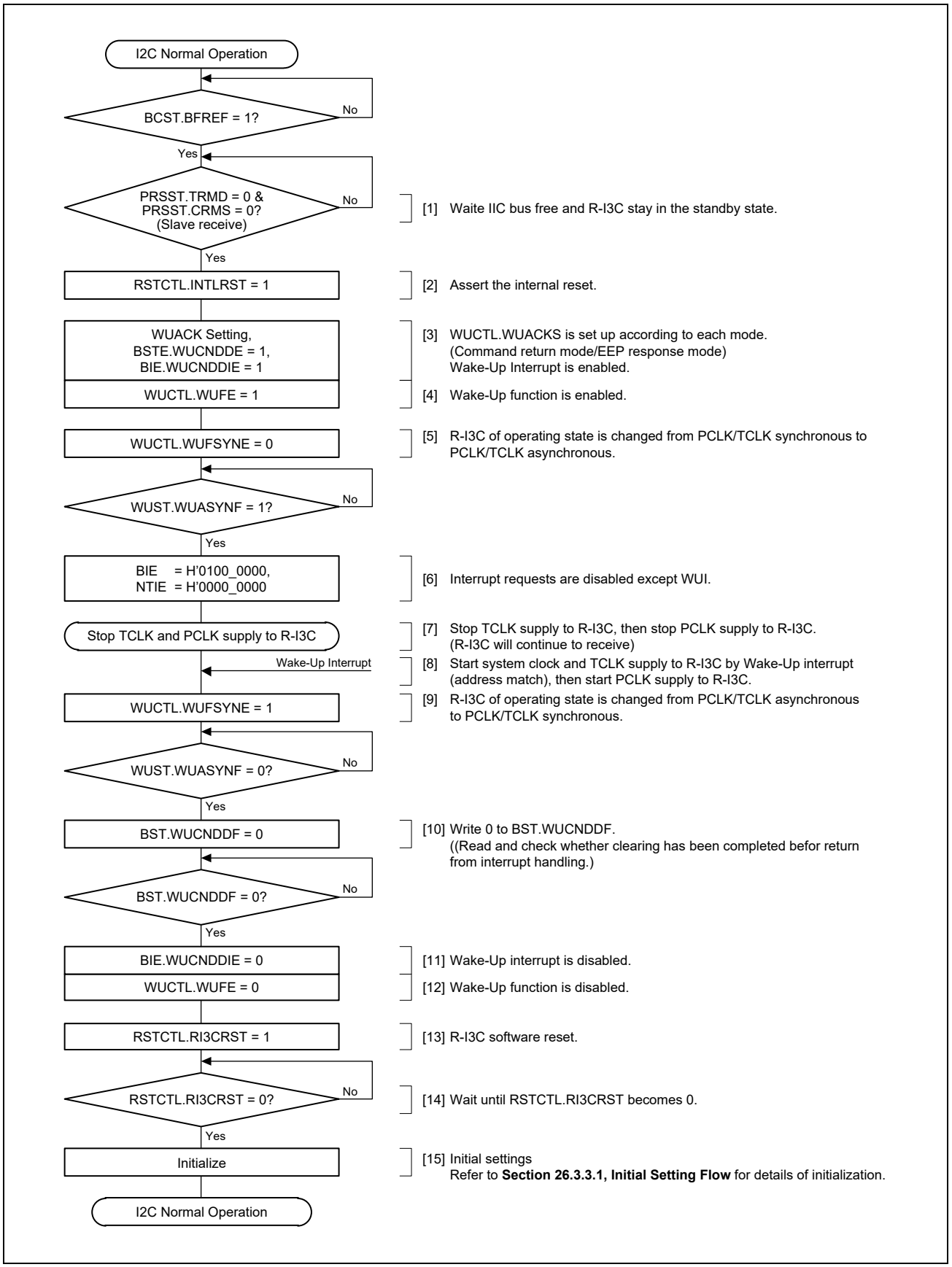


Figure 26.106 Use Case of Command Recover Mode and EEP Response Mode
(Wake-Up Recovery by a Wake-Up Interrupt Triggered by the Match of the Slave Address)

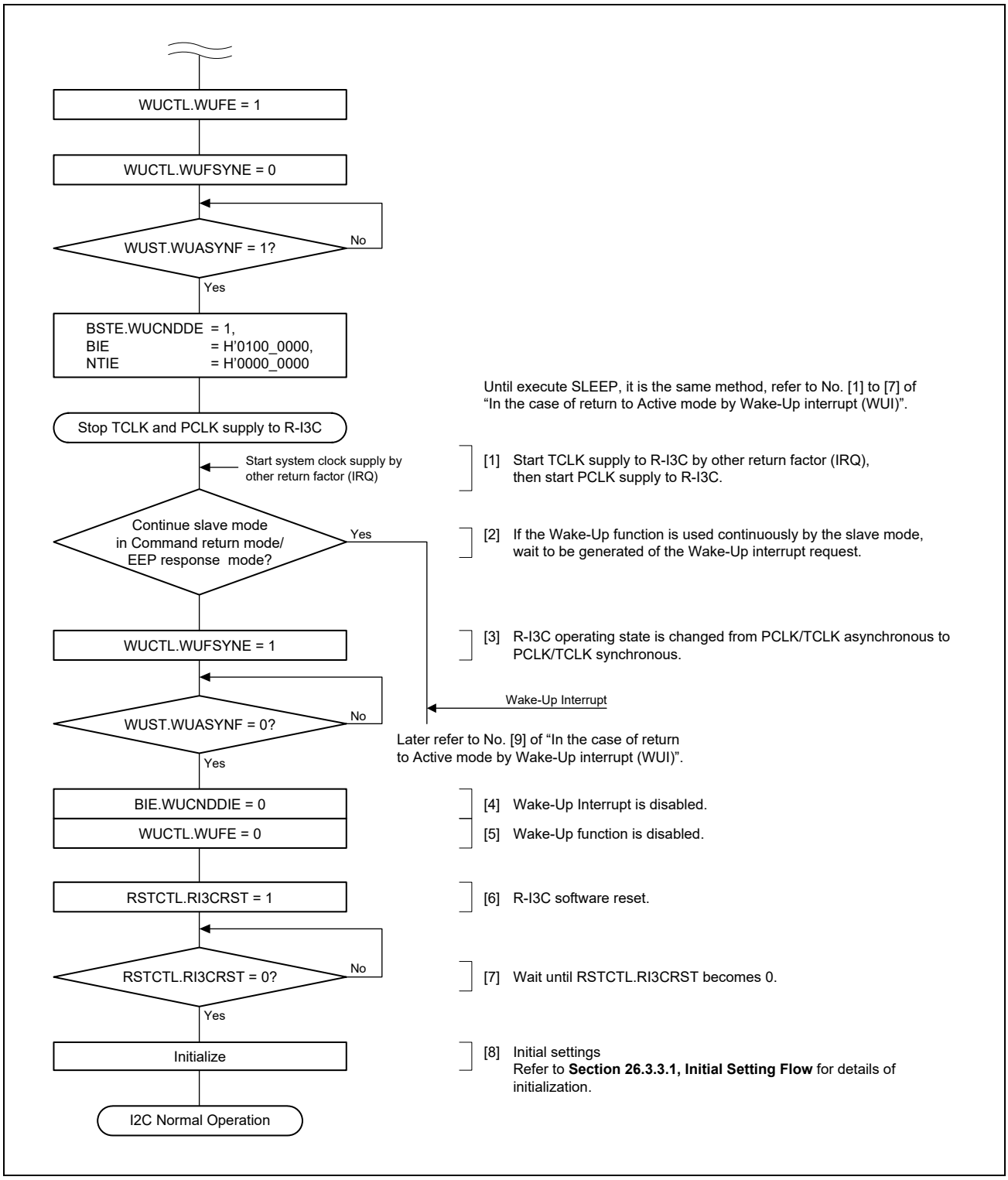


Figure 26.107 Use Case of Command Recover Mode and EEP Response Mode
(Wake-Up Recovery by Other Recovery Causes (IRQ))

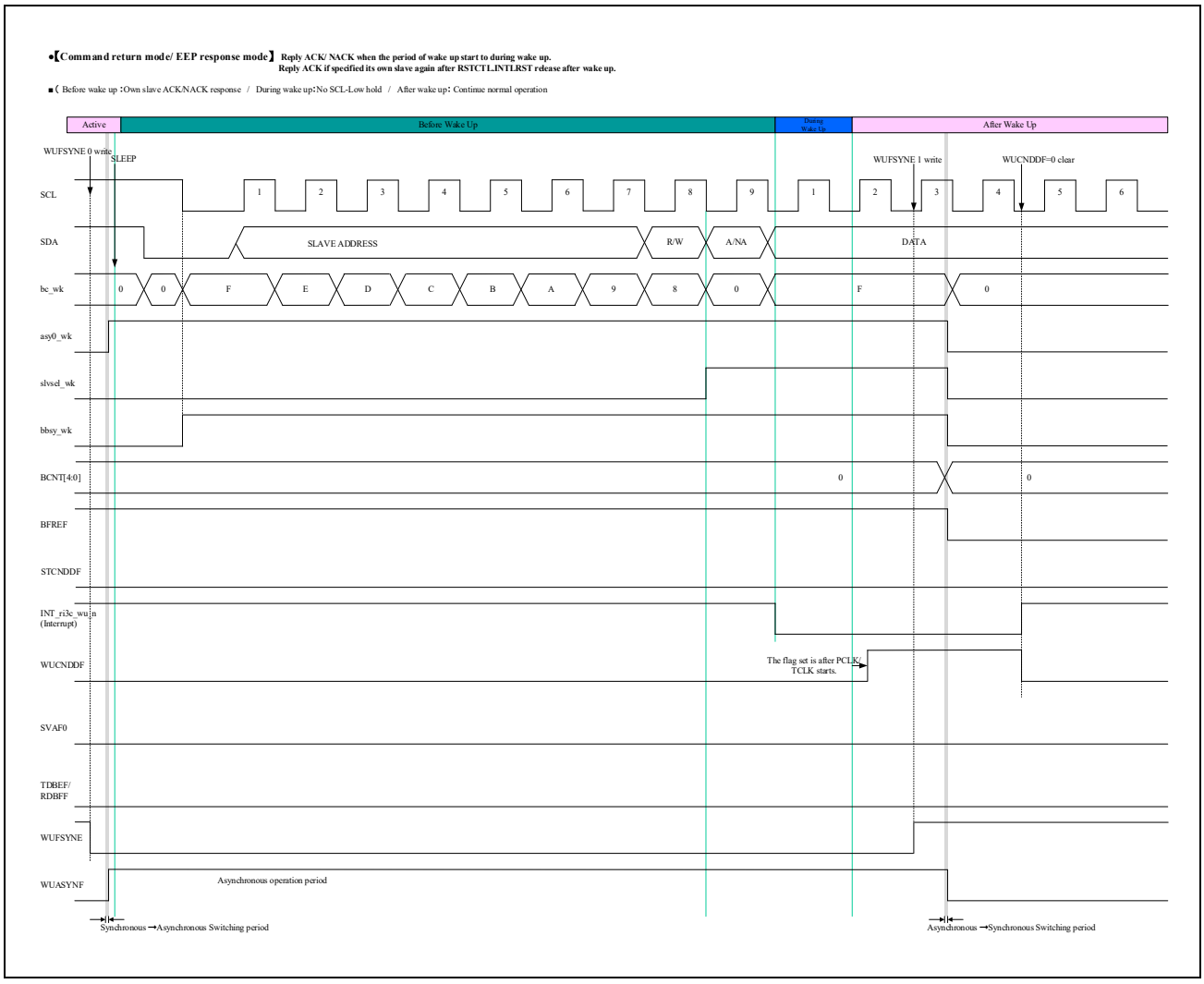


Figure 26.108 Timing of Command Recovery Mode/EEP Response Mode

(d) Precautions on the Use of the Wake-Up Function

Precautions on the use of the Wake-Up function is shown below.

1. Do not change the registers in this IP except the WUCTL.WUFSYNE bit while the WUST.WUASYNF flag = 1 (while PCLK/TCLK asynchronous operation).
2. Set WUCTL.WUFE = BSTE.WUCNDDE = BIE.WUCNDDIE = 1 and PRSST.CRMS = PRSST.TRMD = 0 (slave reception mode) before switching PCLK/TCLK asynchronous mode.
3. Can not select the device ID and the 10-bit slave address for Wake-Up interrupt factor. Set the DVIDE bit in SVCTL and SDADLS bit in SDATBAS2 to 0.
4. Sets all bits in BIE (TENDIE, NACKDIE, SPCNDDIE, STCNDDIE, ALIE, TODIE) to 0 (Interrupt disabled) before switching the asynchronous operation.
5. Do not use the timeout function while the Wake-up function is enabled (WUCTL.WUFE = 1).
6. Wake-up interrupt is generated while PCLK/TCLK asynchronous operation (when WUST.WUASYNF = 1). In case of detecting slave address matching, the case of detect slave address match in PCLK/TCLK synchronous mode (WUST.WUASYNF = 0), does not occur Wake-Up interrupt, and BST.WUCNDDF flag will be not set also.
7. If WUCTL.WUFSYNE bit to “0” write timing and START condition of detecting a conflict, this IP might start the next reception in PCLK/TCLK synchronous operation mode. In this case, WUST.WUASYNF flag becomes 1 (switch to PCLK/TCLK asynchronous mode) when data communication is finished and detected STOP condition and starts the Wake-Up event detection.
8. If you want to switch from PCLK/TCLK asynchronous operation to PCLK/TCLK synchronous operation without address match detection, it will switch in the STOP condition detection. When the WUCTL.WUFSYNE bit was set to 1 in a bus free state, it is continued PCLK/TCLK asynchronous operation (Reception operation: waiting communication frame). WUST.WUASYNF flag becomes to 0 when this IP detect the STOP condition of the next communication frame, and this IP switches to PCLK/TCLK synchronous operation.
9. After writing 0 to WUFSYNE bit in WUCTL, do not change this IP operation mode setting register (BFCTL, SCSTRCTL, ACKCTL, INCTL, SVCTL, SDATBASn) until switched to the PCLK/TCLK asynchronous operation from PCLK/TCLK synchronous operation (while WUST.WUASYNF flag = 1). If register value changes by the interrupt processing etc. in this period, this IP might malfunction without succeeding to the setting to the asynchronous operation.
10. During PCLK/TCLK asynchronous operation (WUST.WUASYNF = 1), do not refer to each flag of SVST, BST register and BCST.BFREF flag.
11. Do not set ACKCTL.ACKT = 1 in order to make an ACK response in the synchronization unit when Wake-Up is performed by slave address match in Normal Wake-Up Mode 2.

(2) Wake Up function [I3C mode]

(a) I3C Master Wake-Up

Wake-Up interrupt causes of I3C Master are shown below.

1. SDA Low detection (IBI request from I3C Slave)

   The operation when transitioning to active mode (normal operation) by Wake-Up interrupt of SDA Low detection is shown below.

   Before Wake-Up recovery: SDA Low Drive is detected and the INTWU interrupt is asserted.

   During Wake-Up recovery: Keep SCL Line High.

   After Wake-Up recovery: Drive SCL Low and complete START condition. SCL is supplied on the I3C Bus and IBI from I3C Slave is received.

If transition to active mode (normal operation) due to other factors, disable the Wake-Up function as necessary.

After confirming PRSTDBG.SDILV = 1, set WUCTL.WUFE = 0.

Do not use the timeout function while the Wake-up function is enabled (WUCTL.WUFE = 1).

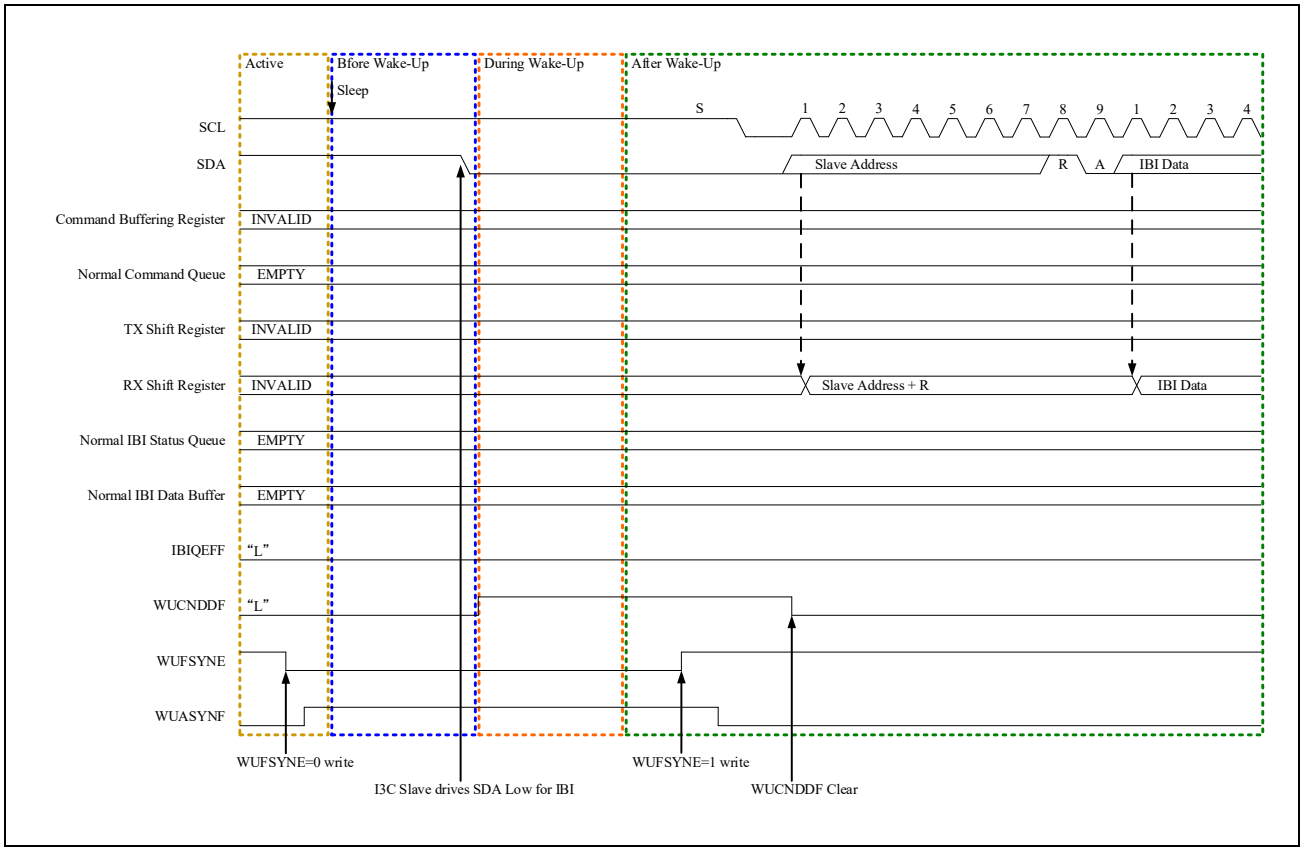


Figure 26.109 I3C Master Wake-Up Operation

(b) I3C Slave Wake-Up

Wake-Up interrupt causes of I3C Slave are shown below.

1. Broadcast Address (H'7E) and detects its own Slave Address match

   The operation when the Broadcast Address (H'7E) and transition to active mode (normal operation) by Wake-Up interrupt by detecting its own Slave Address match is shown.

   Before Wake-Up recovery:

   [Step 1]
   If this IP detects BA (H'7E/W) following a START (or Repeated START) Condition, then this IP shall generate ACK (after H'7E/W).

   [Step 2]

   If this IP detects its own Dynamic Address after a Repeated START condition following Step1, then this IP shall generate NACK (after its own Dynamic Address) and then issues a INTWU interrupt.

   During Wake-Up recovery: This IP always generates NACK.

   After Wake-Up recovery: Normal operation continues.

If transition to active mode (normal operation) due to other factors, disable the Wake-Up function as necessary.

Do not use the timeout function while the Wake-up function is enabled (WUCTL.WUFE = 1).

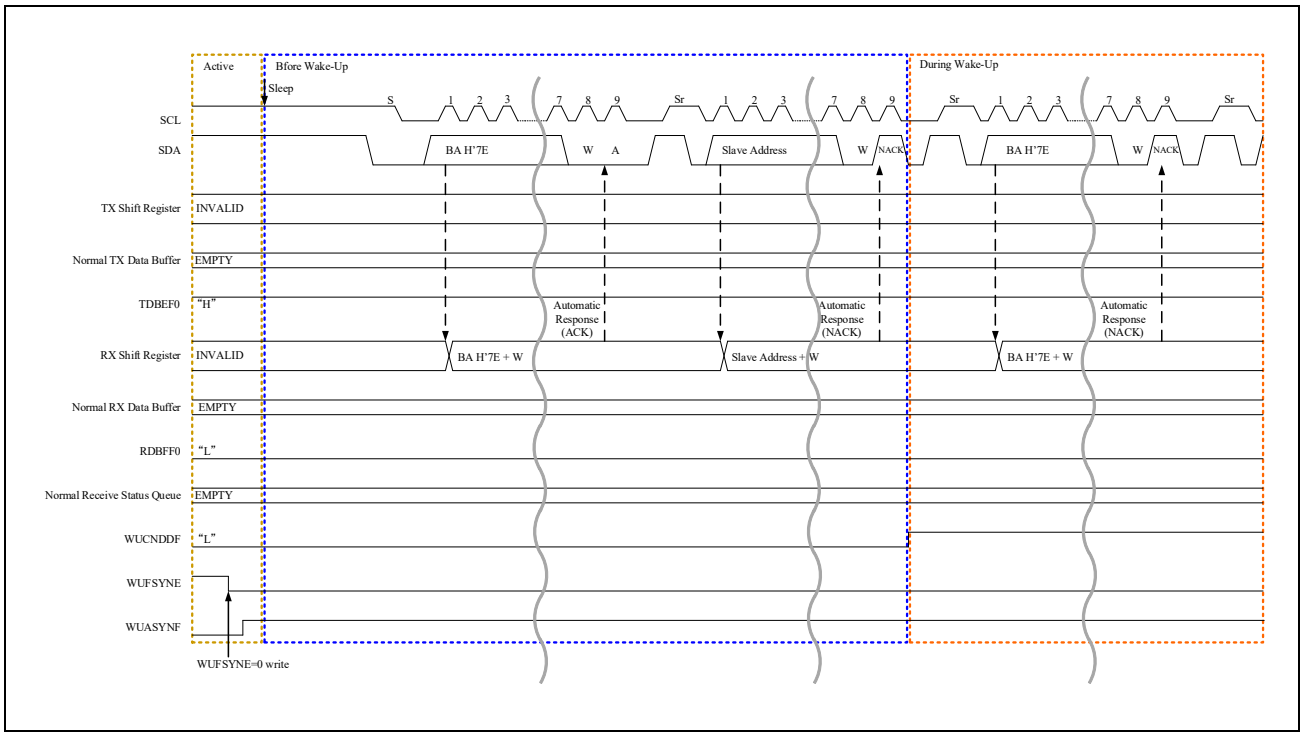


Figure 26.110 I3C Slave Wake-Up Operation (1/2)

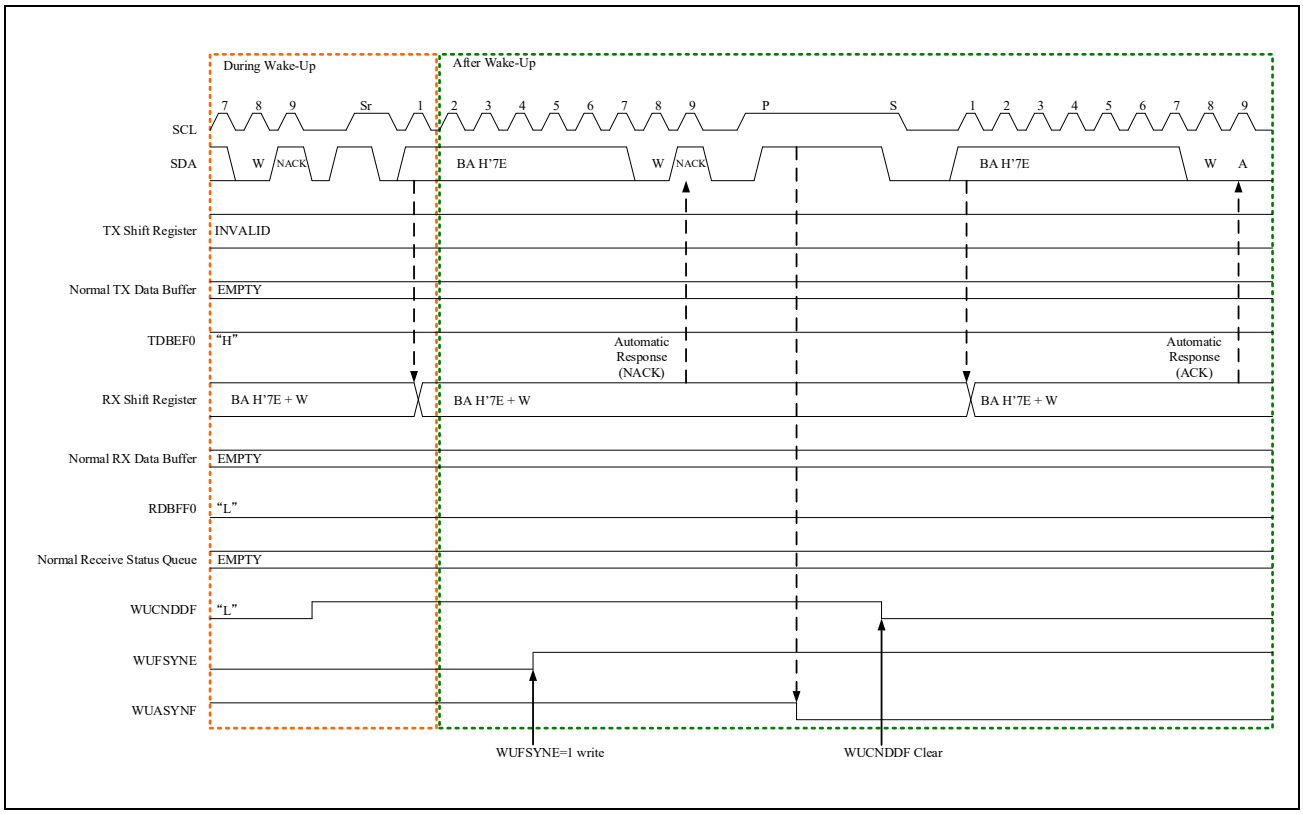


Figure 26.111 I3C Slave Wake-Up Operation (2/2)

26.3.2.6 Other

(1) SCL Synchronization Circuit [I²C mode]

This function is enabled while the PRTS.PRTMD bit is set to 1.

In generation of the SCL clock, I3C starts counting out the value for width at high level specified in STDBR.SBRHO[7:0] when it detects a rising edge on the SCLn line and drives the SCLn line low once counting of the width at high level is complete.

When I3C detects the falling edge of the SCLn line, it starts counting out the width at low level period specified in STDBR.SBRLO[7:0], and then stops driving the SCLn line (releases the line) once counting of the width at low level is complete. The SCL clock is thus generated.

If multiple master devices are connected to the I²C bus, a collision of SCL signals may arise due to contention with another master device. In such cases, the master devices have to synchronize their SCL signals. Since this synchronization of SCL signals must be bit by bit, I3C is equipped with a facility (the SCL synchronization circuit) to obtain bit-by-bit synchronization of the SCL clock signals by monitoring the SCLn line while in master mode.

When I3C has detected a rising edge on the SCLn line and thus started counting out the width at high level specified in STDBR.SBRHO[7:0], and the level on the SCLn line falls because an SCL signal is being generated by another master device, I3C stops counting when it detects the falling edge, drives the level on the SCLn line low, and starts counting out the width at low level specified in STDBR.SBRLO[7:0]. When I3C finishes counting out the width at low level, it stops driving the SCLn line to the low level (releases the line). At this time, if the width at low level of the SCL clock signal from the other master device is longer than the width at low level set in this module, the width at low level of the SCL signal will be extended. Once the width at low level for the other master device has ended, the SCL signal rises because the SCLn line has been released. When I3C finishes outputting the low-level period of the SCL clock, the SCLn line is released and the SCL clock rises. That is, in cases of contention of SCL signals from more than one master, the width at high level of the SCL signal is synchronized with that of the clock having the narrower width, and the width at low level of the SCL signal is synchronized with that of the clock having the broader width. However, such synchronization of the SCL signal is only enabled when the SCSYNE bit in BFCTL is set to 1.

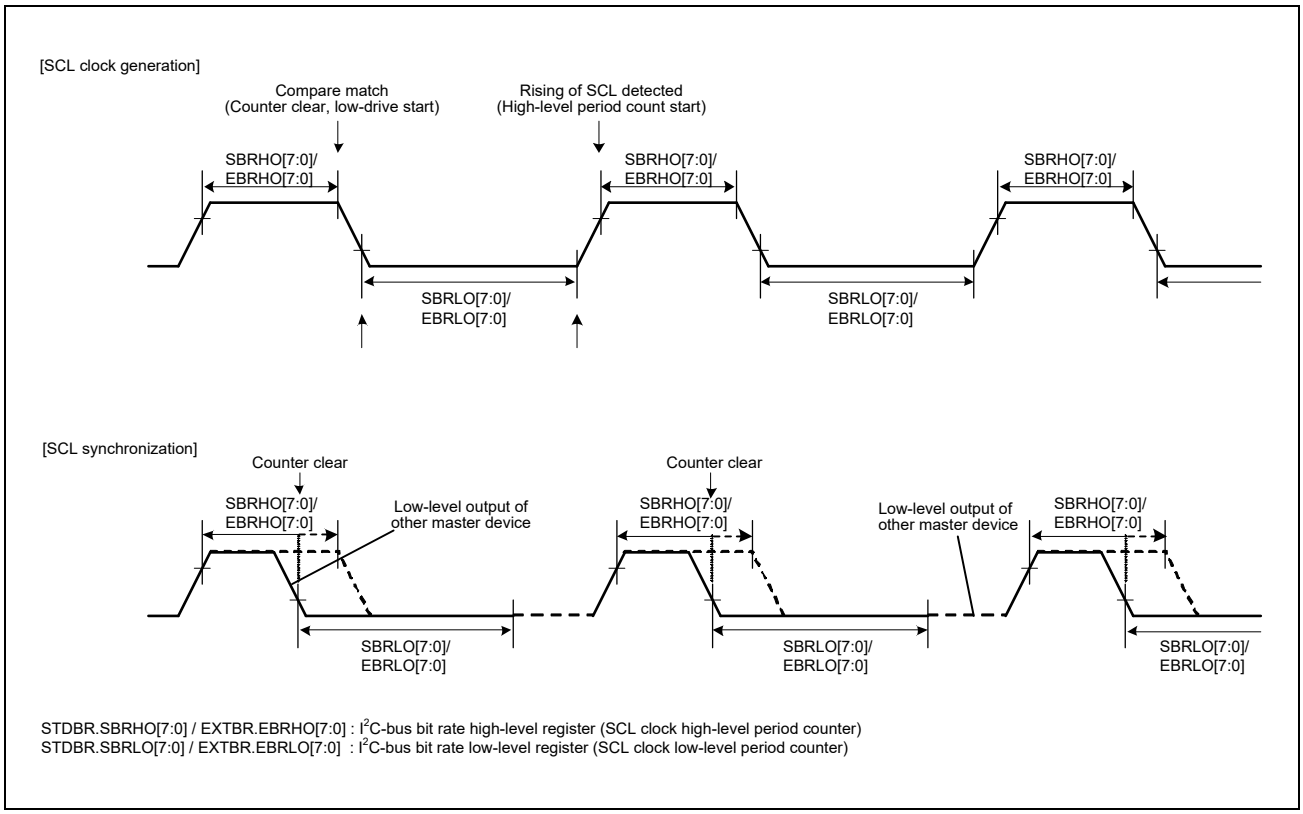


Figure 26.112 Generation and Synchronization of the SCL Signal

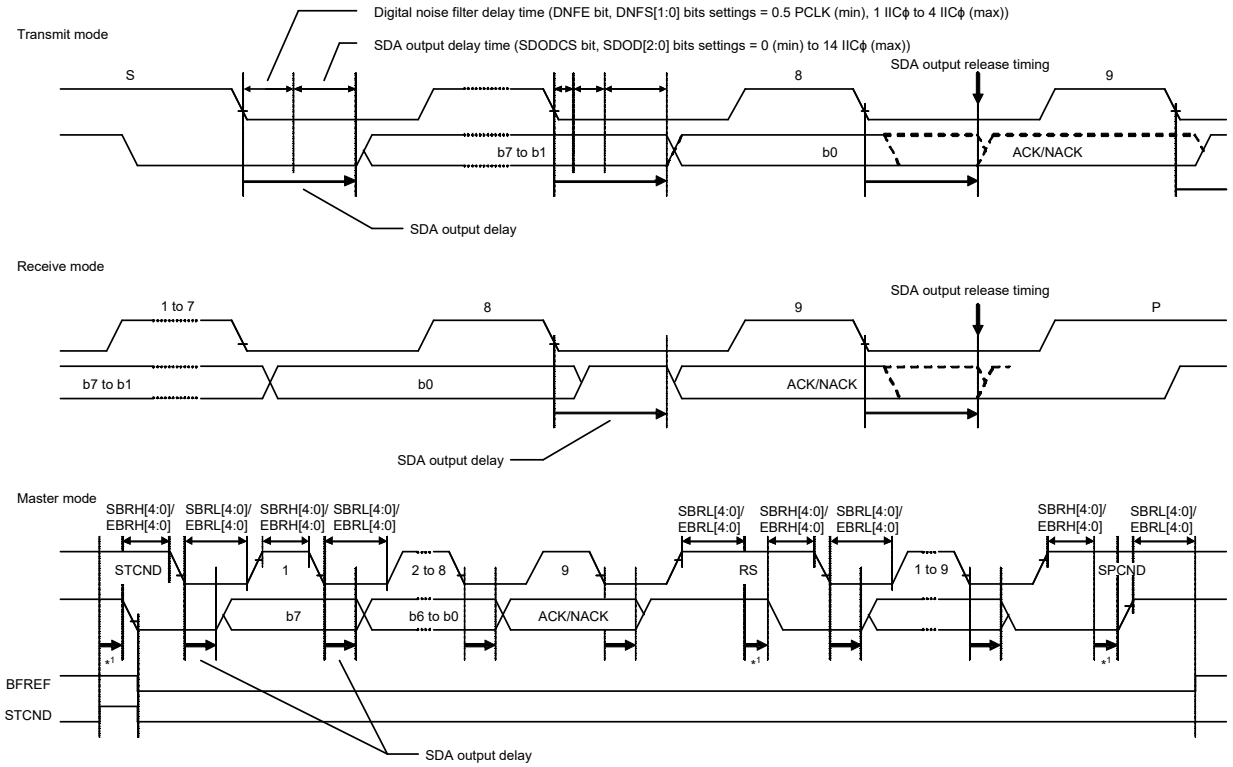
(2) Facility for Delaying SDA Output [I²C mode]

I3C module incorporates a facility for delaying output on the SDA line. The delay can be applied to all output (issuing of the START, Repeated START, and STOP conditions, data, and the ACK and NACK signals) on the SDA line.

With the SDA output delay facility, SDA output is delayed from detection of a falling edge of the SCL signal to ensure that the SDA signal is output within the interval over which the SCL clock is at the low level.

The output delay facility is enabled by setting the SDOD[2:0] bits in OUTCTL to any value other than 000b, and disabled by setting the same bits to 000b.

While the SDA output delay facility is enabled (while the SDOD[2:0] bits in OUTCTL are set to any value other than 000b), the SDODCS bit in OUTCTL selects the clock source for counting by the SDA output delay counter as the internal base clock (I3Cϕ) for I3C module or as a clock signal derived by dividing the frequency of the internal base clock by two (I3Cϕ/2). The counter counts the number of cycles set in the SDOD[2:0] bits in OUTCTL. After counting of the set number of cycles of delay is completed, I3C module places the required output (START, Repeated START, or STOP condition, data, or an ACK or NACK signal) on the SDA line.



Note: PCLK = PCLKB

Note 1. The output delay function is set by the SDODCS and SDOD[2:0] bits when a start (STCND), repeated start (SRCND), or stop (SPCND) condition is issued.

Figure 26.113 SDA Output Delay Facility

(3) Digital Noise-Filter Circuits [I²C mode]

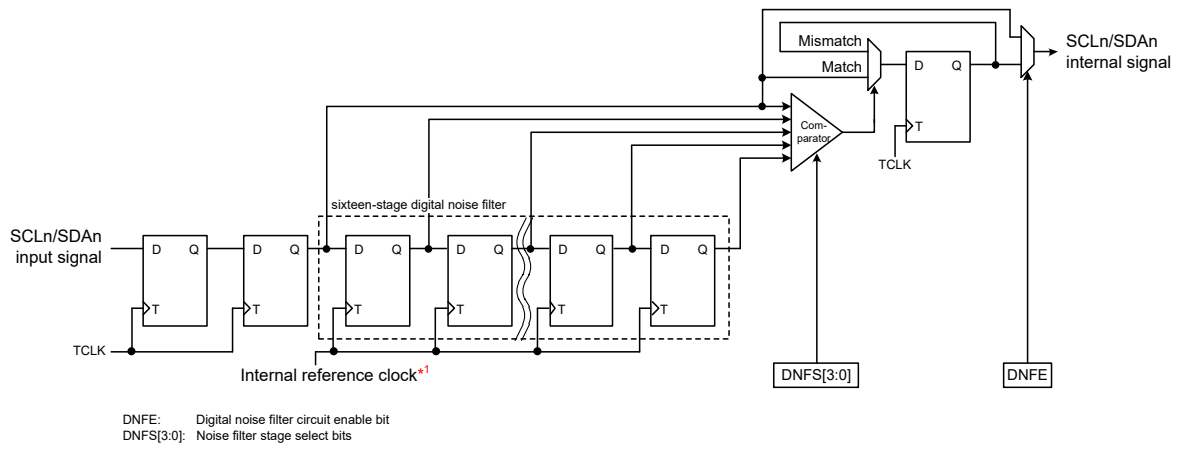
The states of the SCLn and SDAn pins are conveyed to the internal circuitry through digital noise-filter circuits. **Figure 26.114** is a block diagram of the digital noise-filter circuit.

The on-chip digital noise-filter circuit of I3C consists of 16 flip-flop circuit stages connected in series and a match detection circuit. When HS mode is selected, only the first four flip-flop circuits stages are enabled.

The number of effective stages in the digital noise filter is selected by the INCTL.DNFS[3:0] bits. The selected number of effective stages determines the noise-filtering capability as a period from one to sixteen I3Cϕ cycles.

The input signal to the SCLn pin (or SDAn pin) is sampled on rising edges of the I3Cϕ signal. When the input signal level matches the output level of the number of effective flip-flop circuit stages as selected by the INCTL.DNFS[3:0] bits, the signal level is conveyed to the subsequent stage. If the signal levels do not match, the previous value is retained.

If the ratio between the frequency of the internal operating clock (PCLKD) and the transfer rate is small (For example, data transfer at 400 kbps with PCLKD = 4 MHz), the characteristics of the digital noise filter may lead to the elimination of needed signals as noise.



Note: TCLK = PCLKD

Note 1. Internal reference clock = I3Cϕ

Figure 26.114 Block Diagram of Digital Noise Filter Circuit

26.3.3 Operation

26.3.3.1 Initial Setting Flow

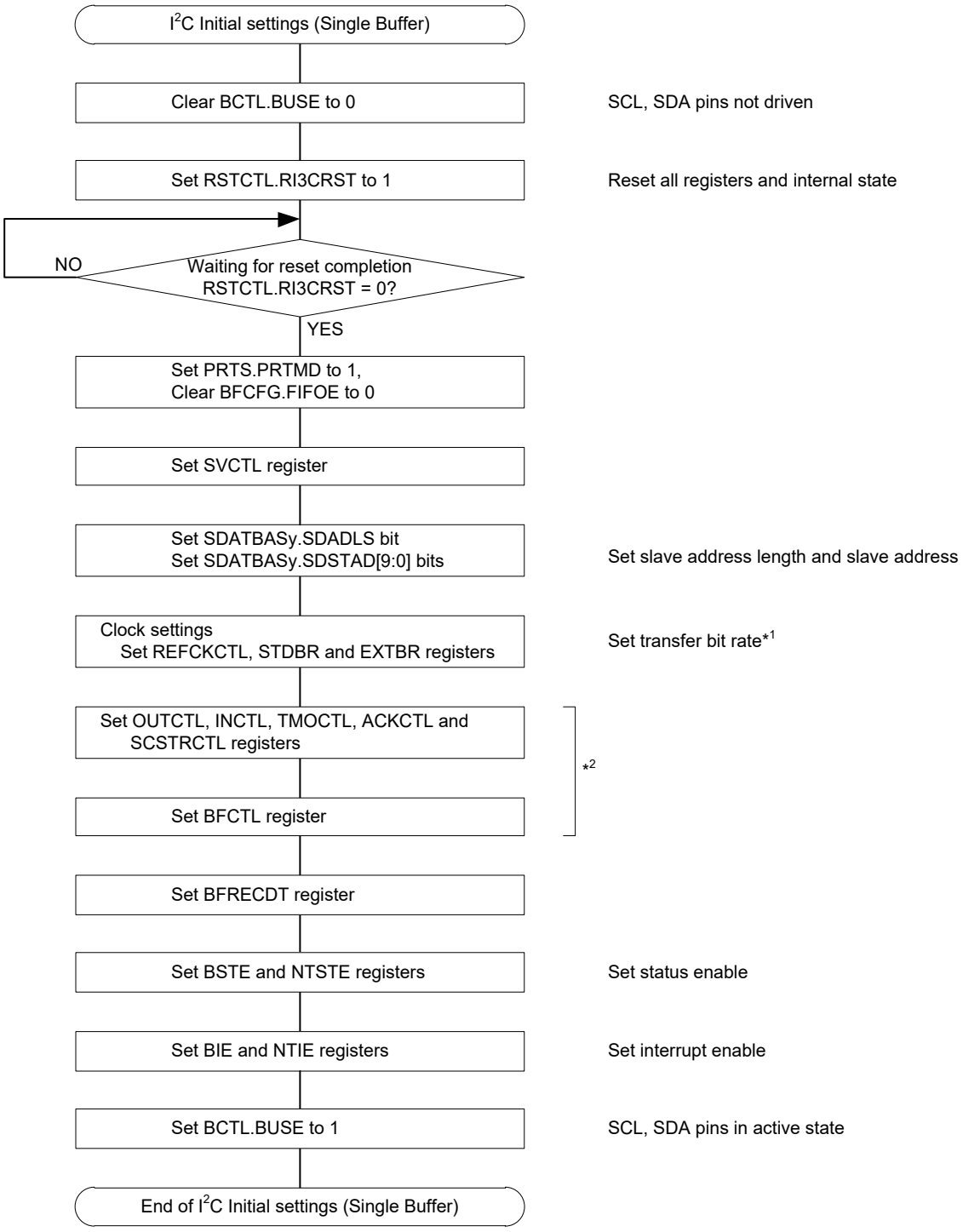
(1) I²C Initial Setting Flow (Single Buffer Transfer)

Before starting data transmission and reception, initialize I3C according to the procedure in **Figure 26.115**. First, set the BCTL.BUSE bit to 0 (SCLn, SDAn pins not driven).

Next, set the RSTCTL.RI3CRST bit to 1 (I3C reset). This initializes all registers and internal state. Then, waits for RI3CRST to become 0.

This initializes the various flags and some registers. Refer to **Section 26.5, Reset Descriptions**.

After that, set registers SDATBAS.SDADLS, SDATBAS.SDATAD[9:0], STDBR, INCTL, OUTCTL, TMOCTL, TMOCNT, SCSTRCTL, ACKCTL, and BFCTL, then set the other registers as necessary (for initial settings of I3C, refer to **Figure 26.115**).



Note: n = 0

Note 1. When the I3C is used only in slave mode, set the STDBR register to a value longer than the data setup time.

Note 2. Set these registers as necessary.

Figure 26.115 Block Diagram of Digital Noise Filter Circuit

(2) I3C Initial Setting Flow

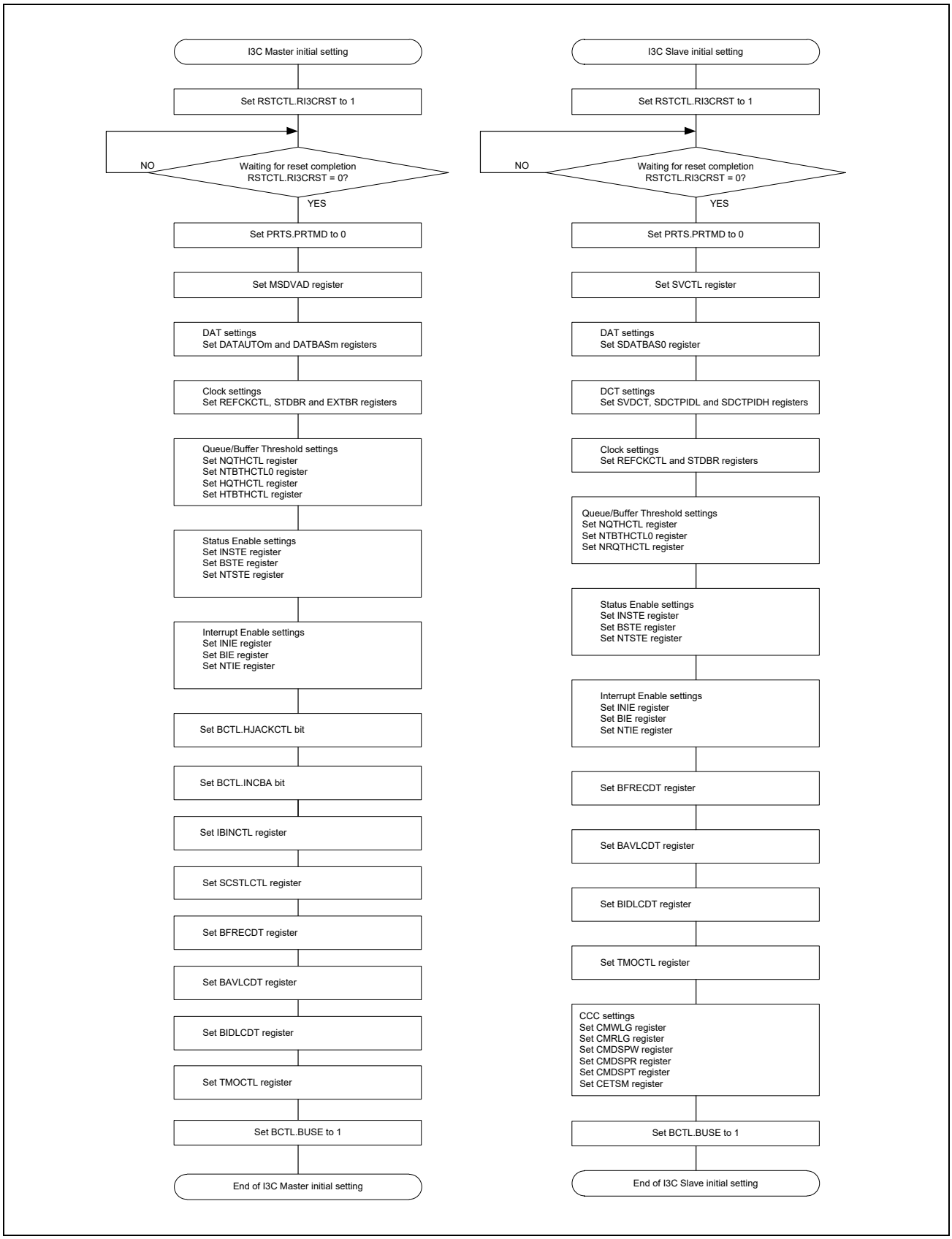


Figure 26.116 Example of I3C Initialization Flowchart

26.3.3.2 I3C Communication Flow

Figure 26.117 illustrates how I3C communication is initiated:

- All I3C communication occurs within a frame. The frame begins with a START, followed by one or more transfers, and a STOP.

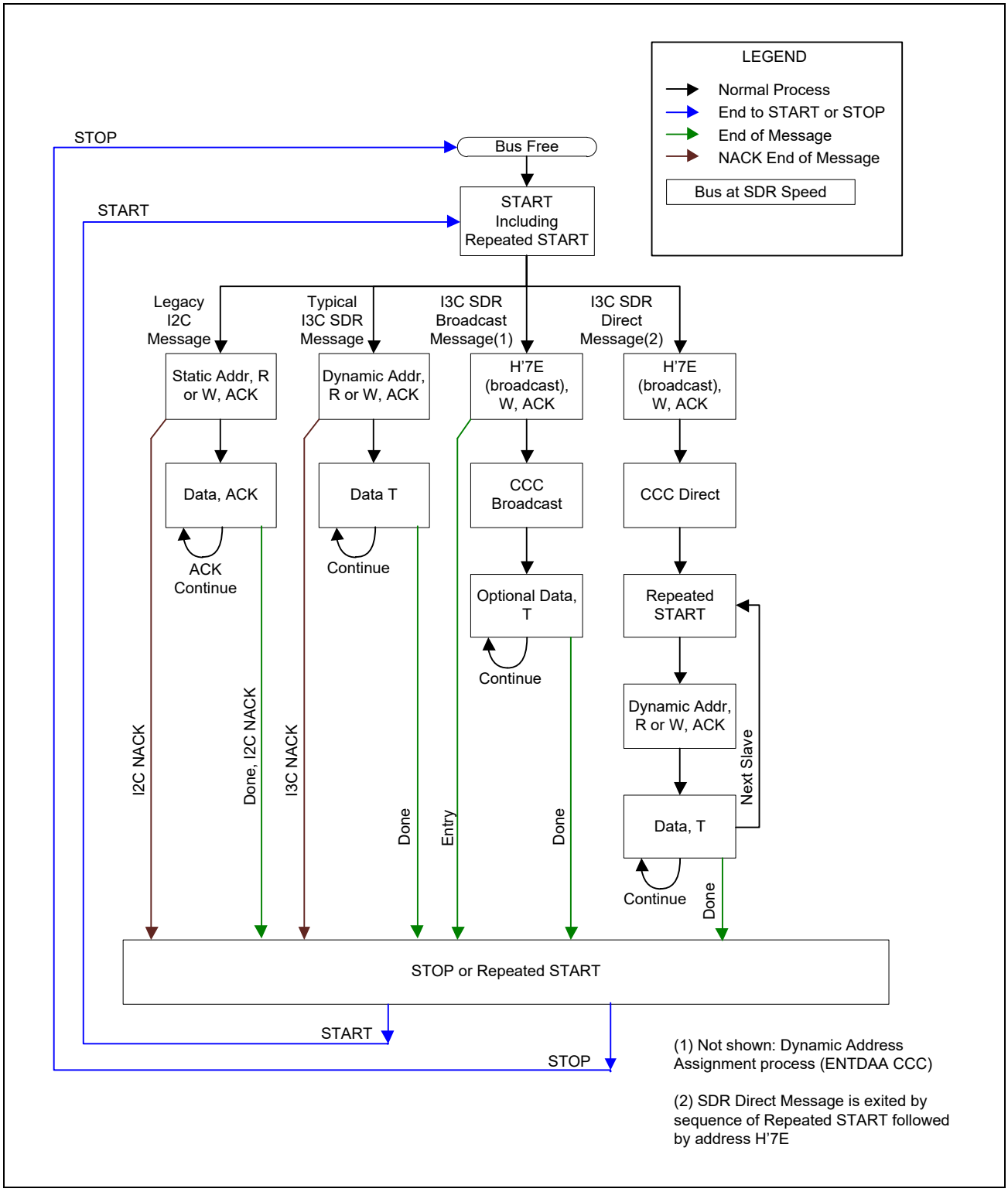


Figure 26.117 I3C Communication Flow

I3C is based on a frame encapsulation approach. A frame includes a data payload. The transfer protocol for the data payload is either SDR. Frames are bordered by I^2C-like bus management.

The I3C frame always includes at least the START, the Header, the Data, and the STOP. The Header following a START allows for Bus Arbitration. The Master uses the Header to address Slave device (s). Slave devices (s) may use the Header Arbitration for multiple purposes: for In-Band Interrupt, for Hot-Join, and for Secondary Master functionality.

I3C allows only one Master to have control of the I3C bus at a time. Mechanisms for handoff of the Master role from one device to another device are provided.

26.3.3.3 Master Mode Communication Flow

(1) I²C Master Transmission Flow (Single Buffer Transfer)

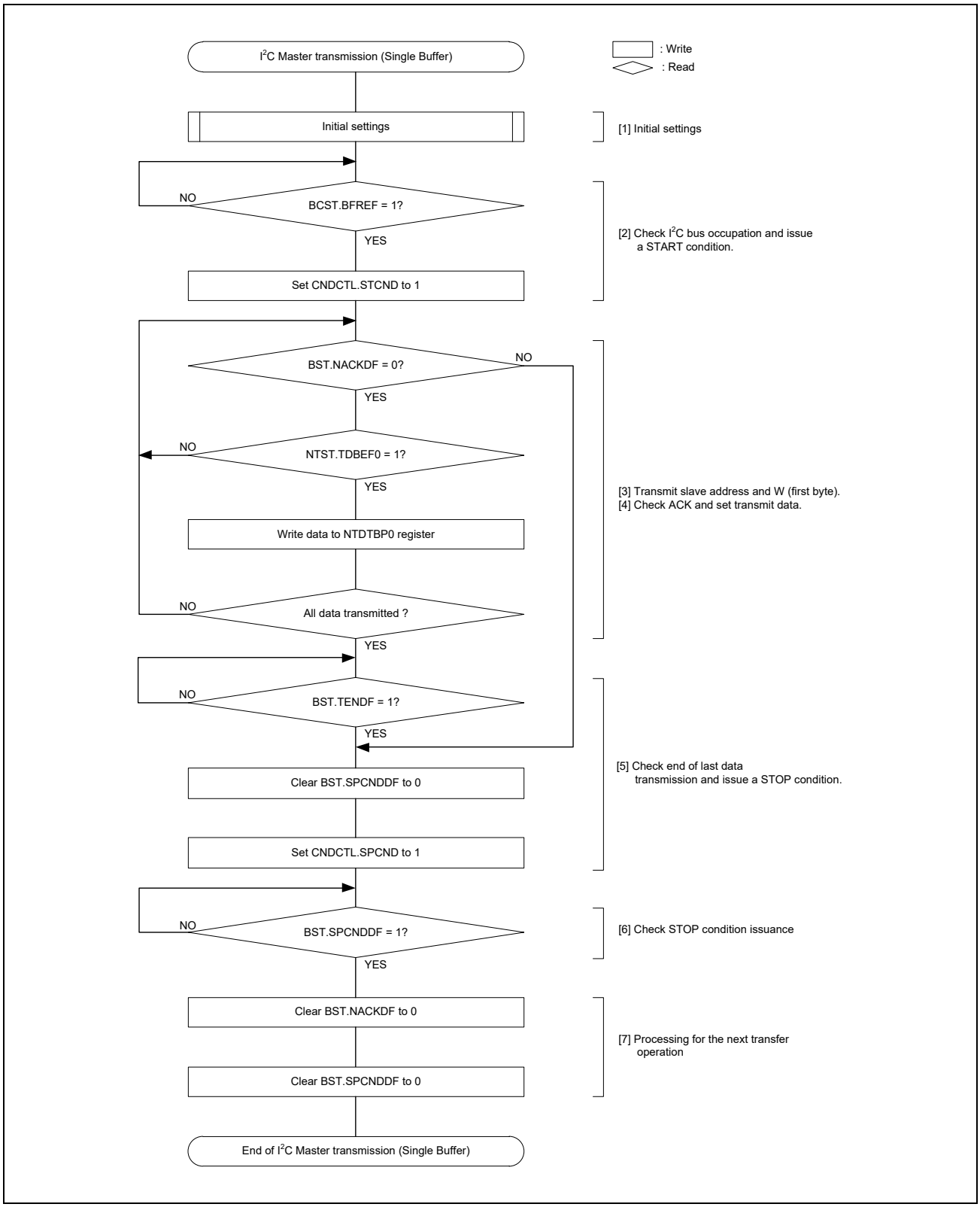


Figure 26.118 Example of I²C Master Transmission Flowchart (Single Buffer Transfer)

NOTE

The following processing is required when checking NTST.TDBEF0 = 1 in Steps [3] and [4] of the I^2C master transmission flowchart as shown in **Figure 26.118**.

- When sending a slave address:
  After confirming NTST.TDBEF0 = 1, confirm that PRSTDBG.SCILV = 0 (check the status of SCL) before writing the transmission data.
- When sending data:
  - If BITCNT.BCNT = other than 0 after confirming NTST.TDBEF0 = 1, write the transmission data immediately
  - If BITCNT.BCNT = 0 after confirming NTST.TDBEF0 = 1, check that BST.TENDF = 1 and PRSTDBG.SCILV = 0 (check the status of SCL) before writing the transmission data.

(2) I²C Master Reception Flow (Single Buffer Transfer)

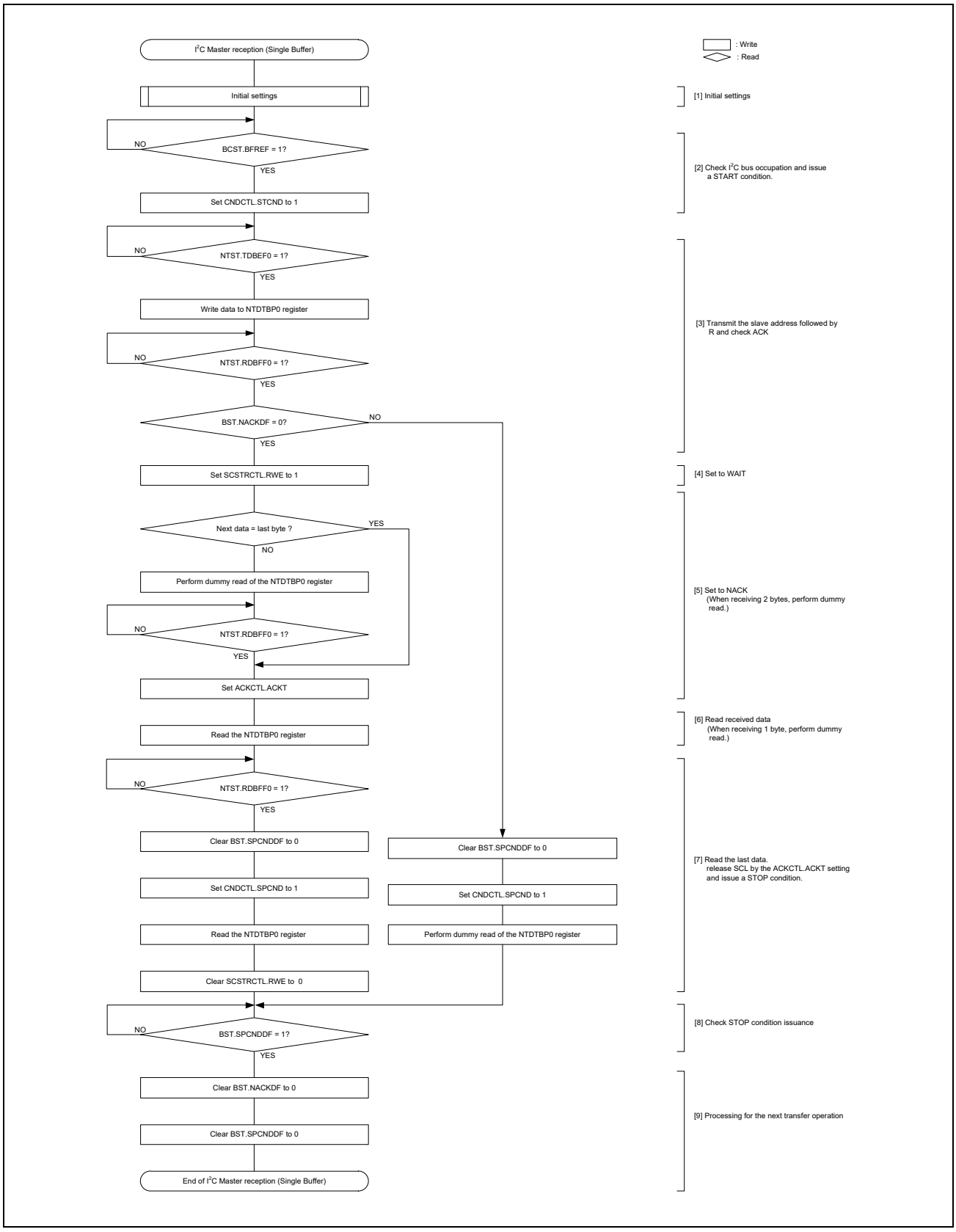


Figure 26.119 Example of I²C Master Reception Flowchart (7-bit Address Format, 1 or 2 bytes)

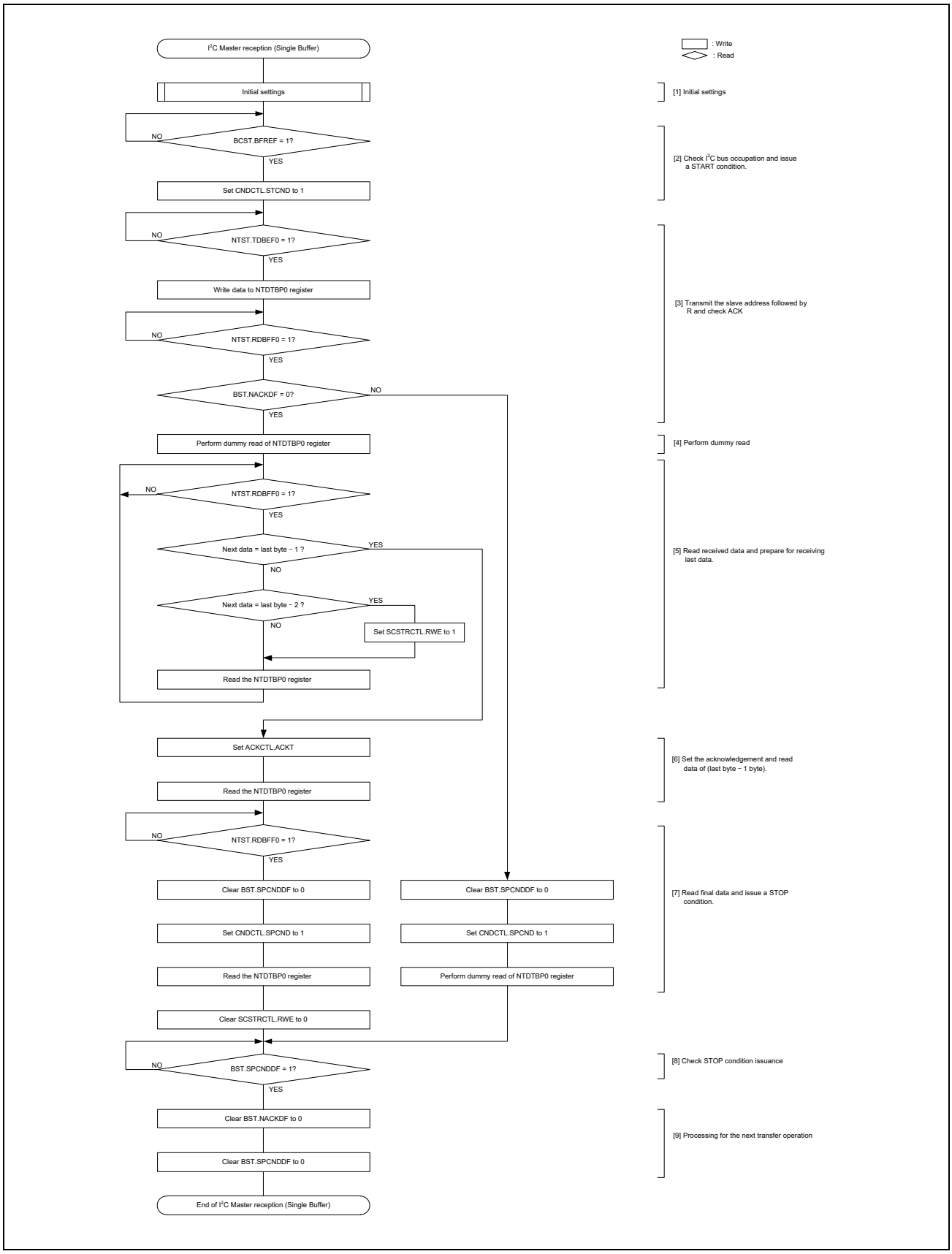
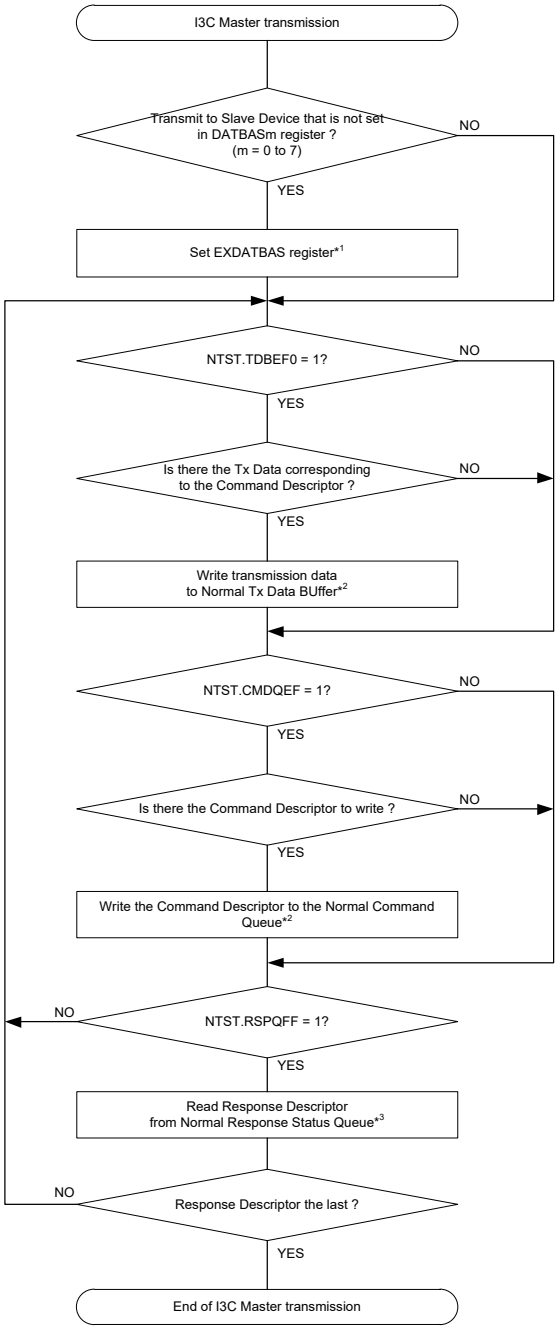


Figure 26.120 Example of I²C Master Reception Flowchart (7-bit Address Format, 3 bytes or more)

(3) I3C Master Transmission Flow (Normal FIFO Buffer Transfer)

Master transmission flow in I3C normal FIFO buffer transfer is common to Legacy I^2C, SDR (Private Transfer, Broadcast CCC, Direct CCC).



Note 1. Changing the EXDATBAS register is prohibited until all response descriptors of command for the Slave device set in the EXDATBAS register are stored.

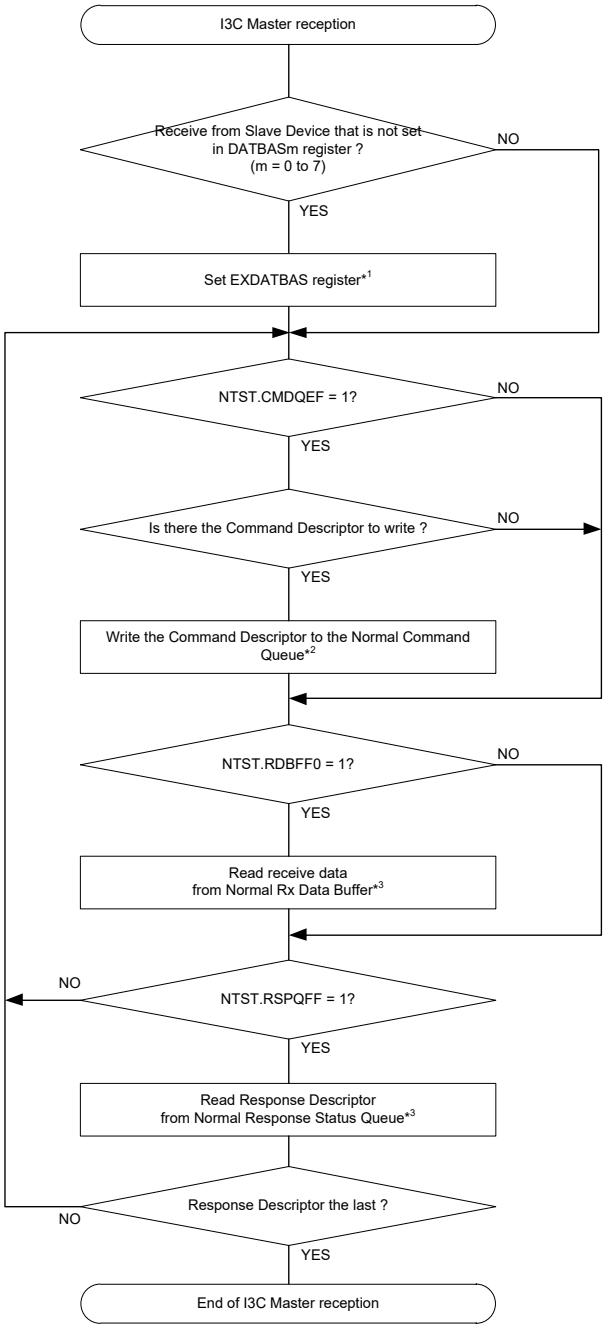
Note 2. In the case the register is write by away other than DTC, clear corresponding flags after write access.

Note 3. In the case the register is read by away other than DTC, clear corresponding flags after read access.

Figure 26.121 Example of I3C Master Transmission Flowchart (Normal FIFO Buffer Transfer)

(4) I3C Master Reception Flow (Normal FIFO Buffer Transfer)

Master reception flow in I3C normal FIFO buffer transfer is common to Legacy I^2C, SDR (Private Transfer, Broadcast CCC, Direct CCC).



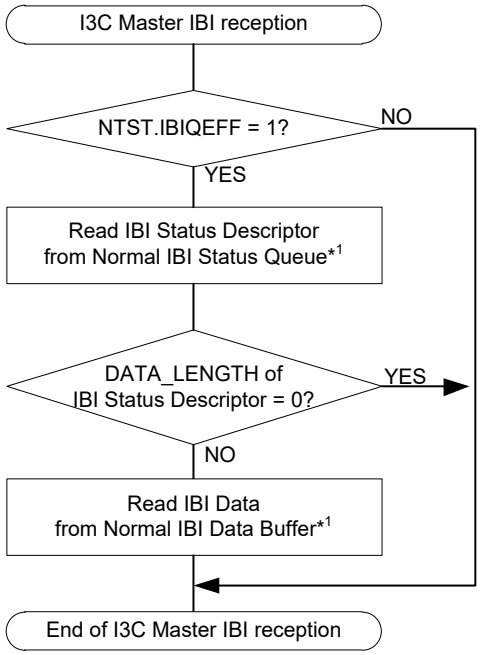
Note 1. Changing the EXDATBAS register is prohibited until all response descriptors of command for the Slave device set in the EXDATBAS register are stored.

Note 2. In the case the register is write by away other than DTC, clear corresponding flags after write access.

Note 3. In the case the register is read by away other than DTC, clear corresponding flags after read access.

Figure 26.122 Example of I3C Master Reception Flowchart (Normal FIFO Buffer Transfer)

(5) I3C Master IBI Reception Flow



Note 1. In the case the register is read by a way other than DTC, clear corresponding flags after read access.

Figure 26.123 Example of I3C Master IBI Reception Flowchart

(6) I3C Master Wake-Up Flow

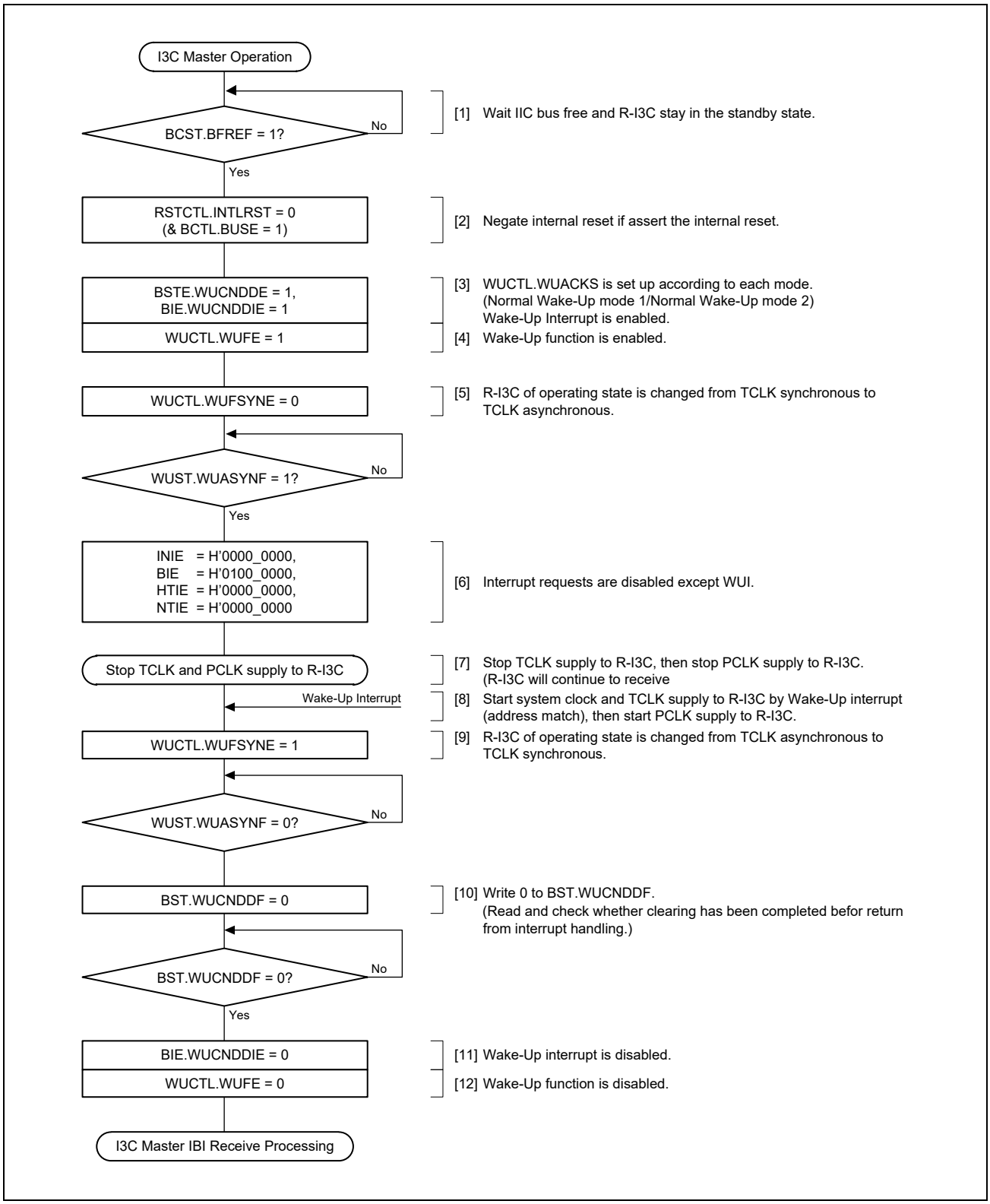


Figure 26.124 Use Case of I3C Master Wake-Up
(Wake-Up Recovery by a Wake-Up Interrupt Triggered by the Match of the Slave Address)

26.3.3.4 Slave Mode Communication Flow

(1) I²C Slave Transmission Flow (Single Buffer Transfer)

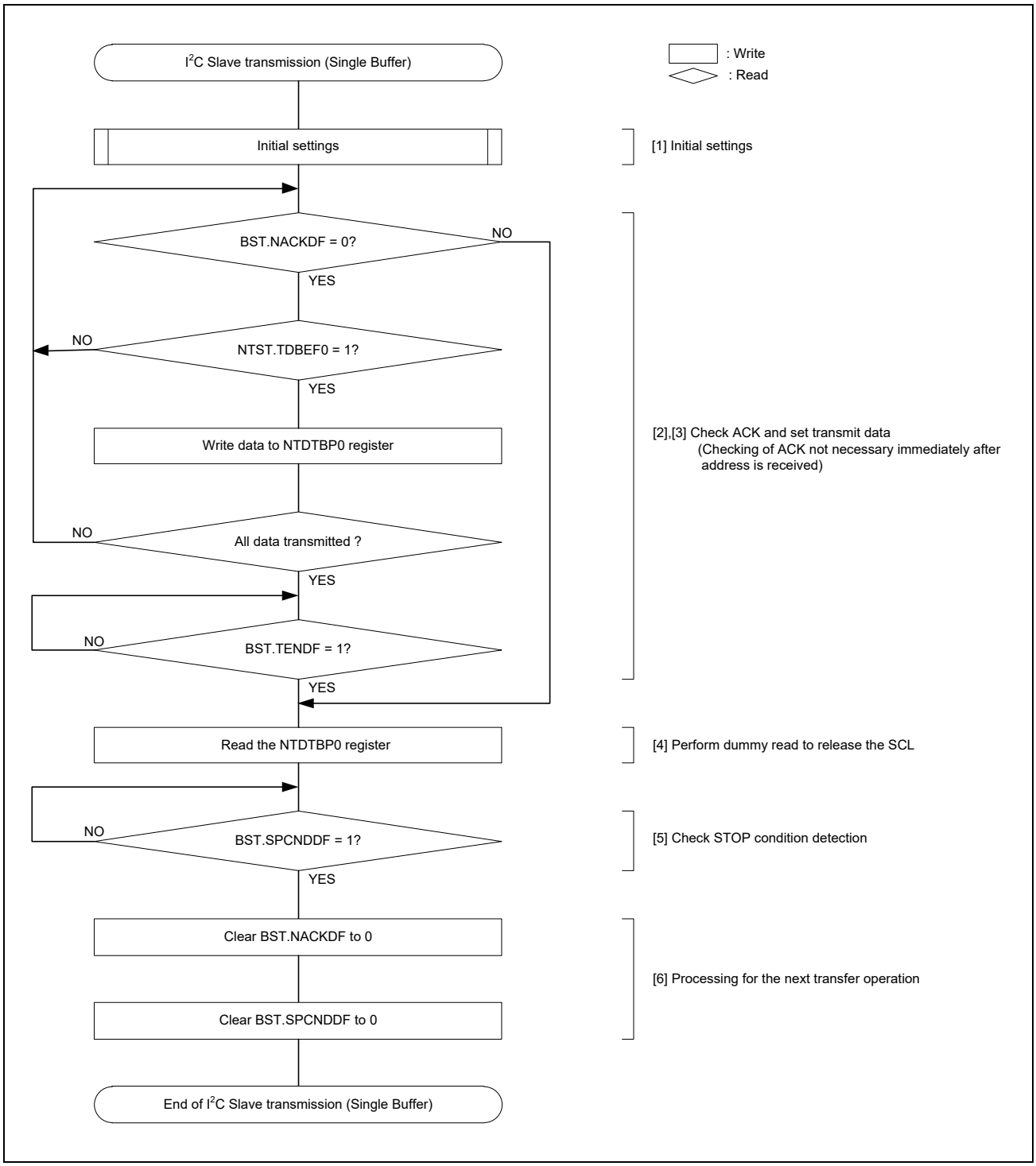


Figure 26.125 Example of I²C Slave Transmission Flowchart (Single Buffer Transfer)

(2) I²C Slave Reception Flow (Single Buffer Transfer)

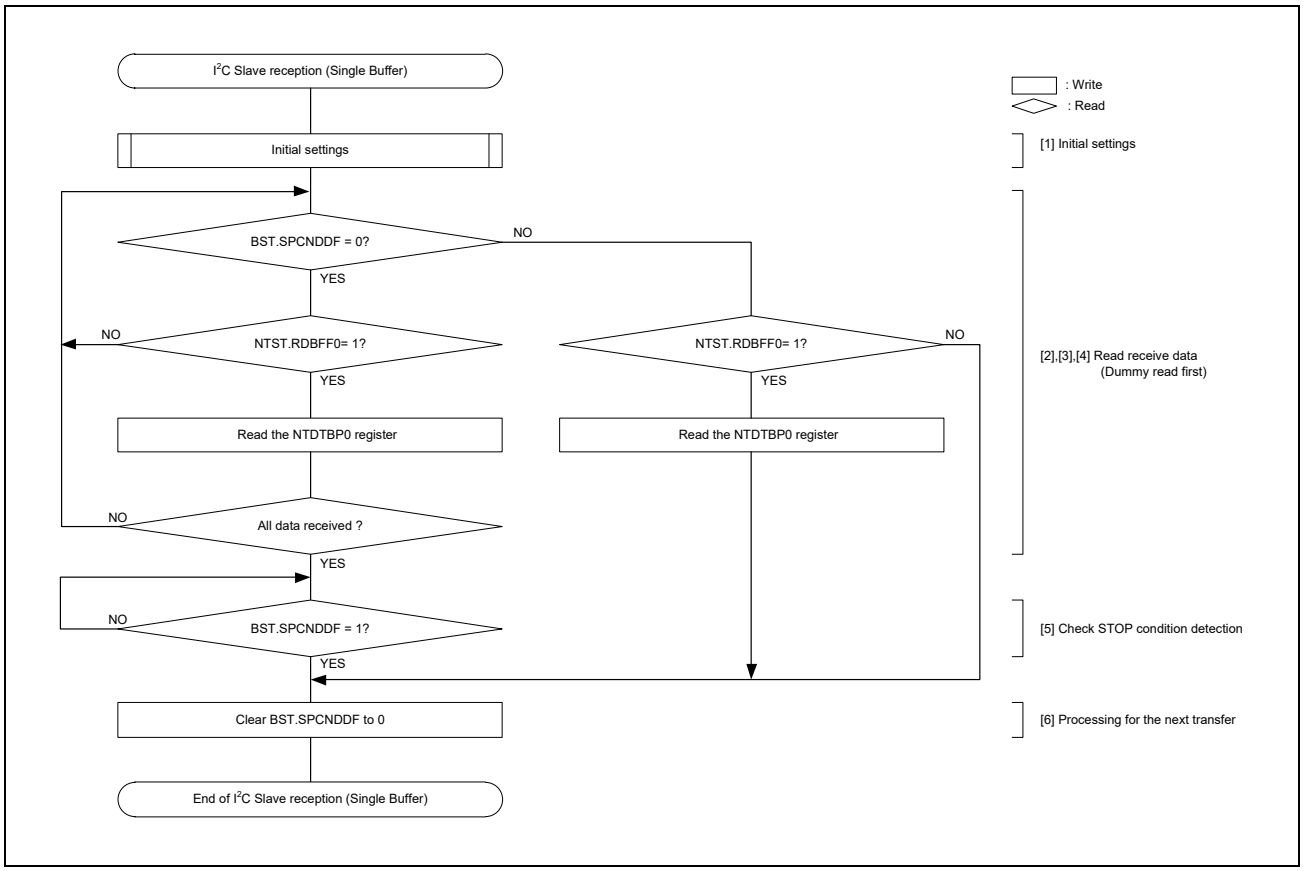
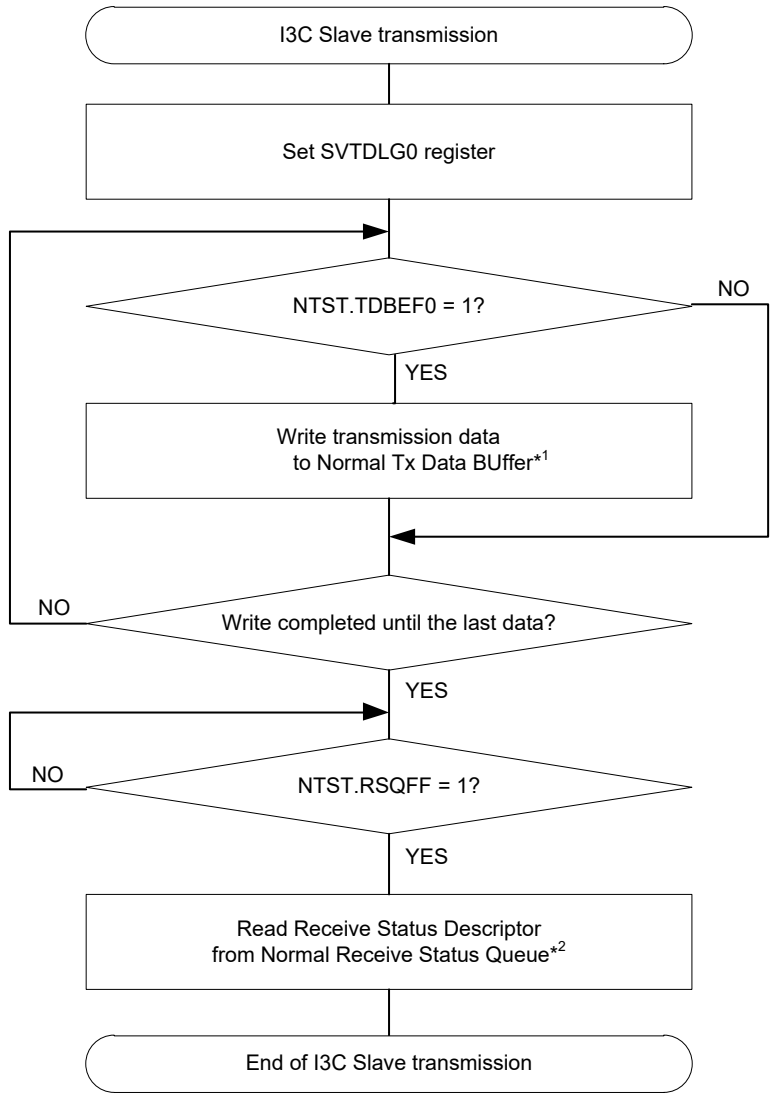


Figure 26.126 Example of I²C Slave Reception Flowchart (Single Buffer Transfer)

(3) I3C Slave Transmission Flow (Normal FIFO Buffer Transfer)

Slave Transmission Flow in I3C normal FIFO buffer transfer is common to Legacy I^2C, SDR (Private Transfer, Broadcast CCC, Direct CCC).

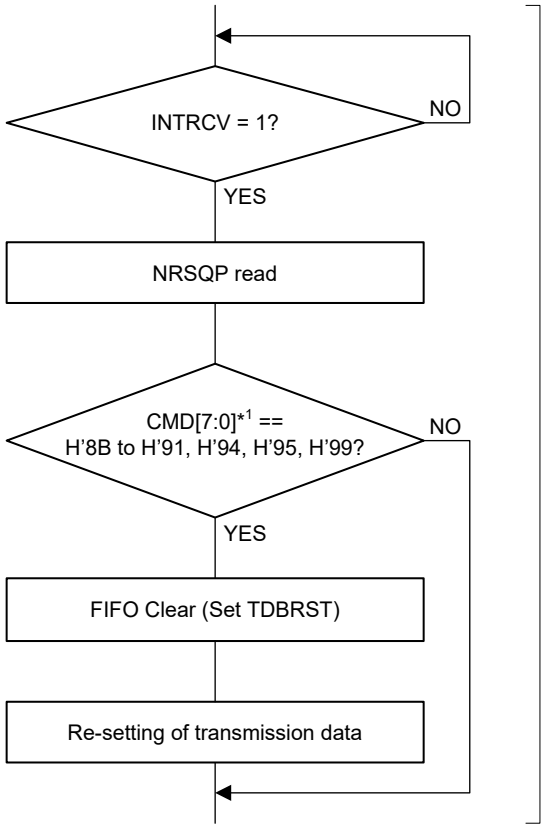


Note 1. In the case the register is write by away other than DTC, clear corresponding flags after write access.

Note 2. In the case the register is read by away other than DTC, clear corresponding flags after read access.

Figure 26.127 Example of I3C Slave Transmission Flowchart (Normal FIFO Buffer Transfer)

When using the I3C as an I3C slave, if I3C slave receives GET CCC while data exists by writing from the NTDTBP0 register to the transmission buffer, follow the flow below.

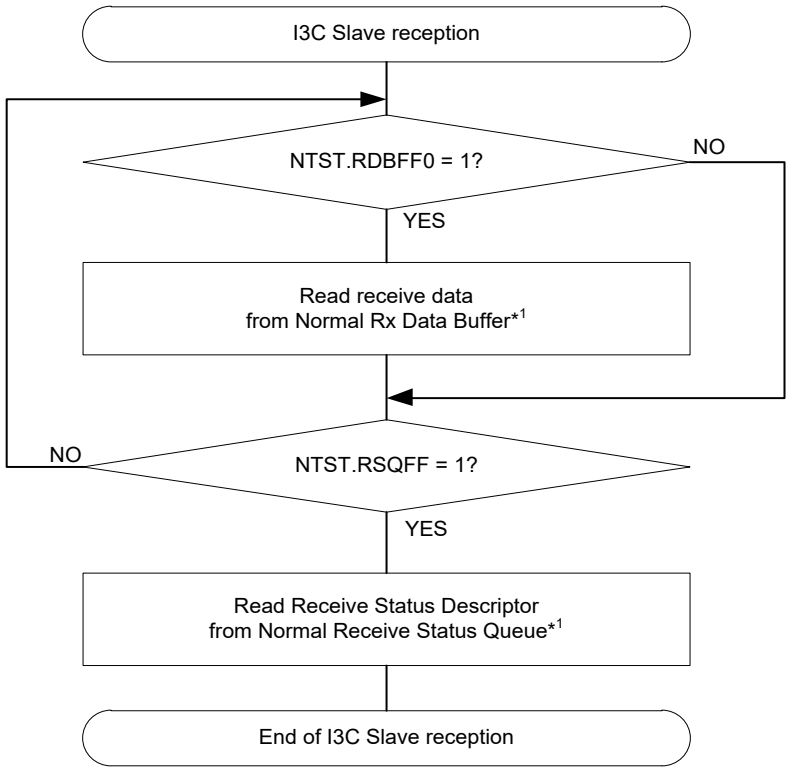


Note 1. CMD[7:0] is the bit field [23:16] of NRSQP (Receive Status Descriptor). GET CCC is when CMD[7:0] is H'8B to H'91, H'95, or H'99.

Figure 26.128 I3C Slave Receives GET CCC while Data Exists by Writing from the NTDTBP0 Register to the Transmission Buffer

(4) I3C Slave Reception Flow (Normal FIFO Buffer Transfer)

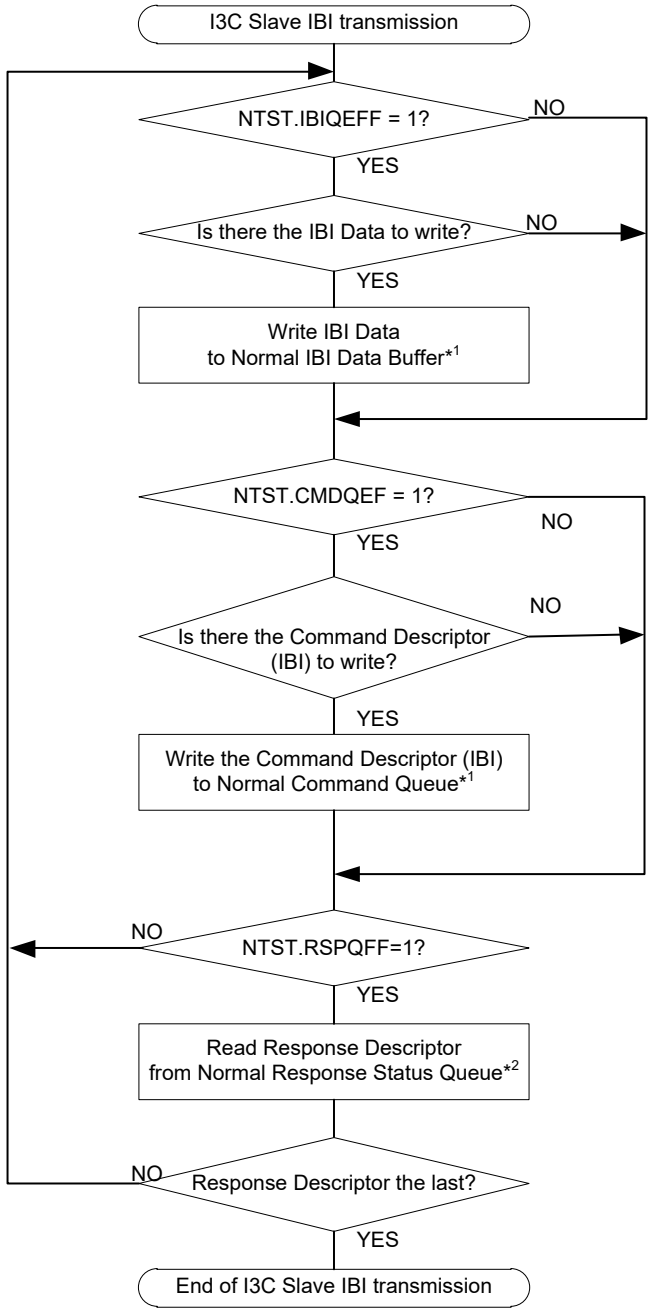
Slave Reception Flow in I3C normal FIFO buffer transfer is common to Legacy I^2C, SDR (Private Transfer, Broadcast CCC, Direct CCC).



Note 1. In the case the register is read by away other than DTC, clear corresponding flags after read access.

Figure 26.129 Example of I3C Slave Reception Flowchart (Normal FIFO Buffer Transfer)

(5) I3C Slave IBI Transmission Flow



Note 1. In the case the register is write by away other than DTC, clear corresponding flags after write access.

Note 2. In the case the register is read by away other than DTC, clear corresponding flags after read access.

Figure 26.130 Example of I3C Slave IBI Transmission Flowchart

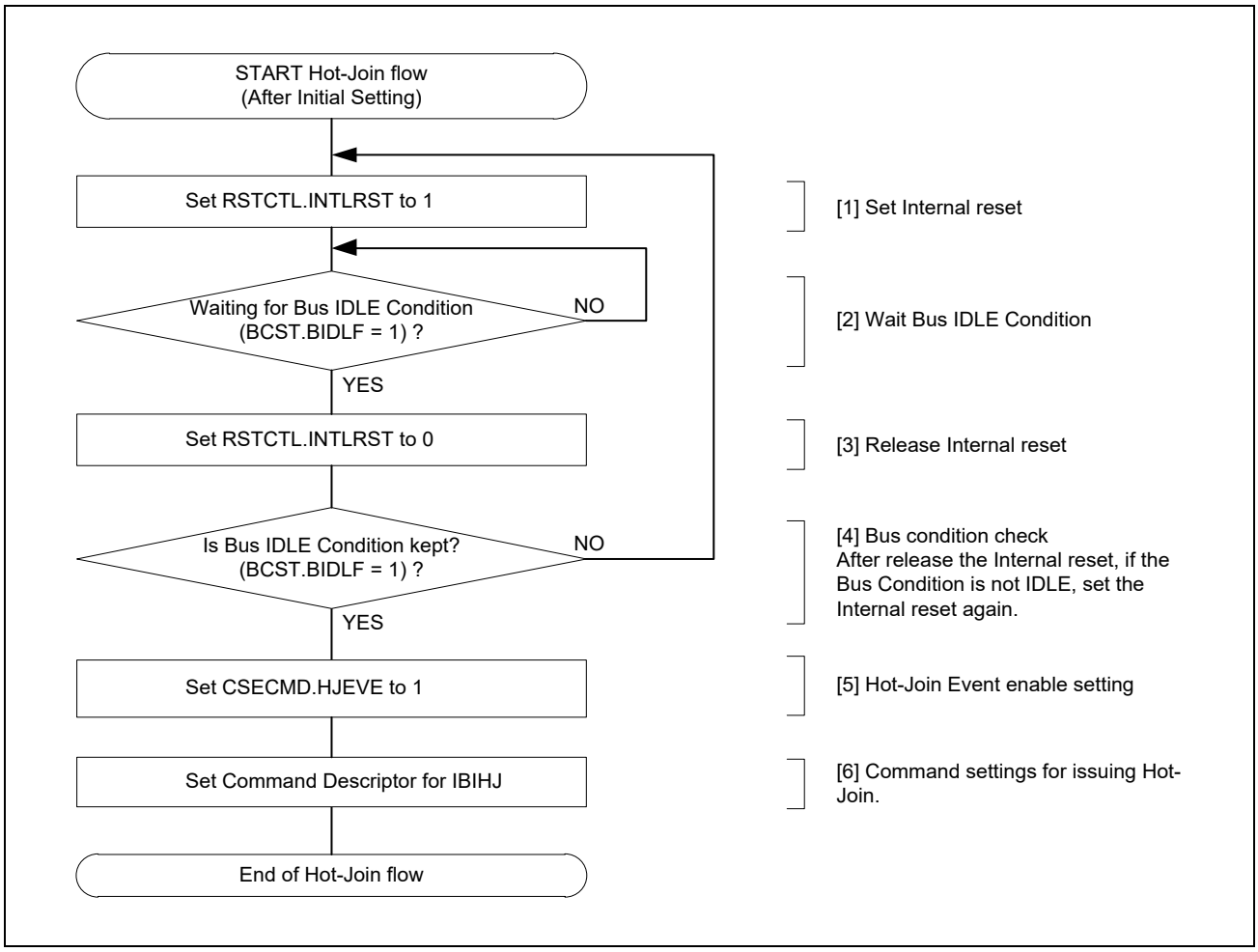


Figure 26.131 Hot-Join Flowchart after the I3C Bud has already been Configured

(6) I3C Slave Wake-Up Flow

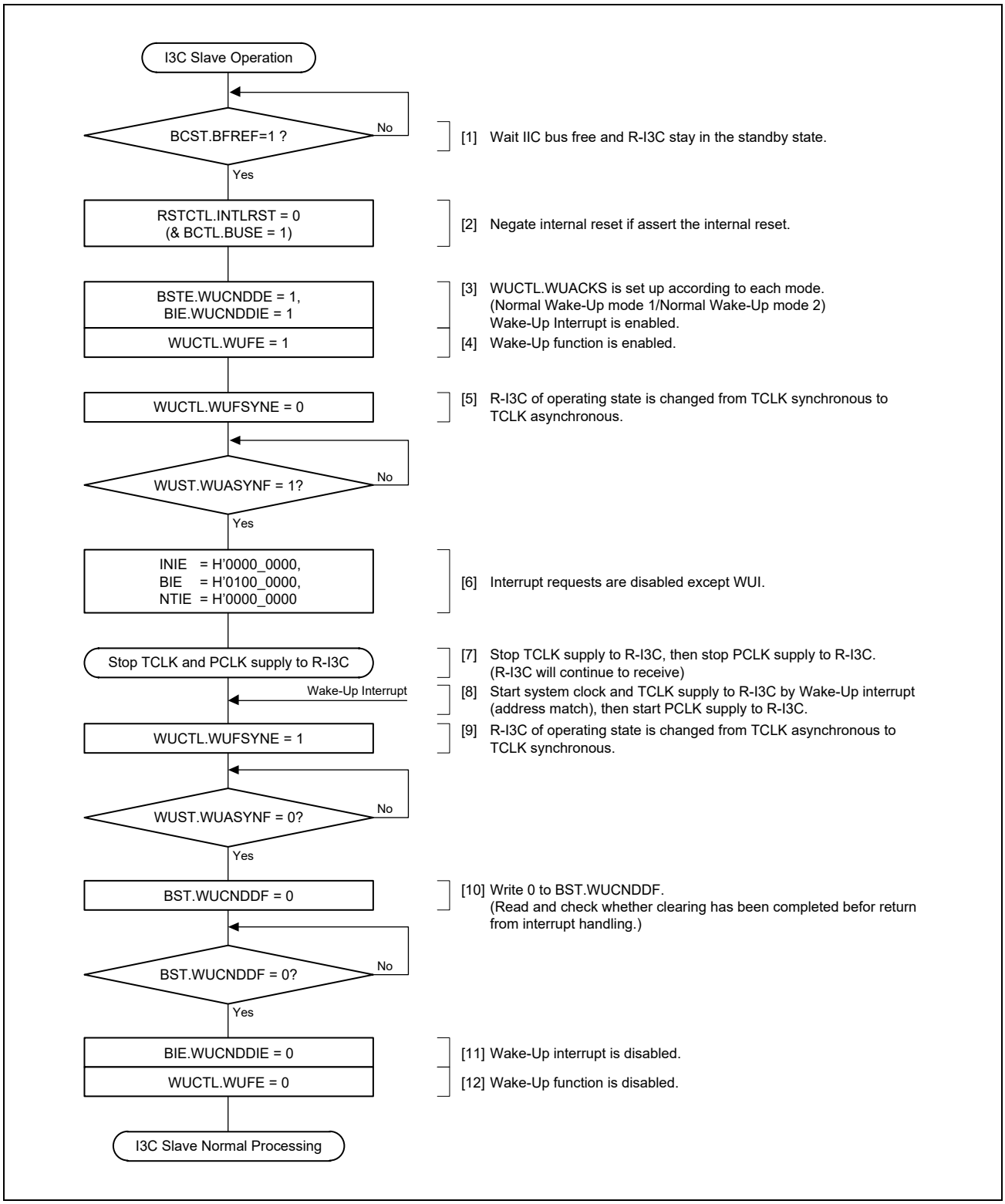


Figure 26.132 Use Case of I3C Slave Wake-Up
(Wake-Up Recovery by a Wake-Up Interrupt Triggered by the Match of the Slave Address)

26.4 Interrupt Sources

I3C can generate the following interrupt requests:

26.4.1 Overview

The I3C has the interrupt factors shown in **Table 26.17**.

The interrupt indicated by Possible in the DTC Activation column are capable of activating data transfer by the DTC.

Table 26.17 Interrupt Generation

Symbol	Interrupt Source		Interrupt Flag	Support			
				I^2C	I3CM	I3C2M	I3CS
I3C_RESP	Normal response status buffer full		NTST.RSPQFF	—	✓	✓	✓
I3C_CMD	Normal command buffer empty		NTST.CMDQEF	—	✓	✓	✓
I3C_IBI	Normal IBI status buffer empty/full		NTST.IBIQEFF	—	✓	✓	✓
I3C_RX	Normal receive data buffer empty/full		NTST.RDBEF0	✓	✓	✓	✓
I3C_TX	Normal transmit data buffer empty		NTST.TDBEF0	✓	✓	✓	✓
I3C_RCV	Normal receive status buffer full		NTST.RSQFF	—	—	✓	✓
I3C_TEND	Transmit end		BST.TENDF	✓	—	—	—
I3C_EEI	Transfer error or event occurrence	Start condition detection interrupt	BST.STCNDDF	✓	✓	✓	✓
		STOP condition detection interrupt	BST.SPCNDDF	✓	✓	✓	✓
		HDR Exit Pattern detection interrupt	BST.HDREXDF	—	✓	✓	✓
		NACK detection interrupt	BST.NACKDF	✓	—	—	—
		Arbitration lost interrupt	BST.ALF	✓	—	—	—
		Timeout detection interrupt	BST.TODF	✓	✓	✓	✓
		Wake-Up Condition detection interrupt	BST. WUCNDDF	✓	✓	✓	✓
		Non-recoverable internal error interrupt	INST.INEF	—	✓	✓	✓
		Transfer Error interrupt	NTST.TEF	—	✓	✓	✓
		Transfer Abort interrupt	NTST.TABTF	—	✓	✓	✓

Note: ✓: Support
—: Not support

Note: I^2C: I^2C Master/Slave (Single Buffer)
I3CM: I3C Master
I3C2M: I3C Secondary Master
I3CS: I3C Slave

For I3C Protocol mode:

For details, refer to the detailed explanation of each flag bit.

- The INTCMD, INTTX0 and INTIBI (I3C Slave) interrupts are cleared under the following conditions.
  On completaion of the last write accsess by DMAC/DTC. Write 0 to this bit after 1 state is read by CPU.

- The INTRESP, INTIBI (I3C Master), INTRX0 and INTRCV interrupts are cleared under the following conditions. On completaion of the last read accsess by DMAC/DTC. Write 0 to this bit after 1 state is read by CPU.
- The INTCMD, INTTX0 and INTIBI (I3C Slave) interrupts are cleared when the number of empty buffers is less than the threshold corresponding to each flag.
- The INTRESP, INTIBI (I3C Master), INTRX0 and INTRCV interrupts are cleared when the number of buffer entries is less than the threshold corresponding to each flag.

26.5 Reset Descriptions

Table 26.18 Register States When Issuing Each Condition (1/11)

Register symbol	Register bit name	System reset	RSTCTL Register											
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST	HCMDQRST	HRSPQRST	HTDBRST	HRDBRST
PRTS	PRTMD	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BCTL	BUSE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RSM	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ABT	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HJACKCTL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	INCBA	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
MSDVAD	MDYADV	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	MDYAD[6:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
RSTCTL	INTLRST	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HRDBRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HTDBRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HRSPQRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HCMDQRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RSQRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	IBIQRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RDBRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TDBRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RSPQRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	CMDQRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RI3CRST	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
PRSST	PRSSTWP	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TRMD	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	CRMS	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
INST	INEF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Table 26.18 Register States When Issuing Each Condition (2/11)

Register symbol	Register bit name	System reset	RSTCTL Register											
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST	HCMDQRST	HRSPQRST	HTDBRST	HRDBRST
INSTE	INEE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
INIE	INEIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
INSTFC	INEFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
DVCT	IDX[4:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
IBINCTL	NRSIRCTL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	NRMRCTL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	NRHJCTL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BFCTL	HSME	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	FMPE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SCSYNE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SALE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	NALE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	MALE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SVCTL	SVAE2	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SVAE1	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SVAE0	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HOAE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVIDE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HSMCE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	GCAE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
REFCKCTL	IREFCKS[2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Table 26.18 Register States When Issuing Each Condition (3/11)

Register symbol	Register bit name	System reset	RSTCTL Register											
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST	HCMDQRST	HRSPQRST	HTDBRST	HRDBRST
STDBR	DSBRPO	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SBRHP [5:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SBRLP [5:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SBRHO [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SBRLO [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
EXTBR	EBRHP [5:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	EBRLP [5:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	EBRHO [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	EBRLO [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BFRECDT	FRECYC [8:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BAVLCDT	AVLCYC [8:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BIDLCDT	IDLCYC [17:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
OUTCTL	SDODCS	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SDOD [2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	EXCYC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SOCWP	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SCOC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SDOC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
INCTL	SDID [1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DNFE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DNFS [3:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
TMOCTL	TOMDS [1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TOHCTL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TOLCTL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TODTS [1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Table 26.18 Register States When Issuing Each Condition (4/11)

Register symbol	Register bit name	System reset	RSTCTL Register											
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST	HCMDQRST	HRSPQRST	HTDBRST	HRDBRST
WUCTL	WUFE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	WUFSYNE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	WUANFS	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	WUACKS	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
ACKCTL	ACKTWP	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ACKT	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ACKR	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SCSTRCTL	RWE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ACKTWE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SCSTLCTL	ACKPE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	PARPE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	AAPE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	STLCYC [15:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SVTDLG0	STDLG [15:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
STCTL	STOE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
ATCTL	CDIV [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
ATTRG	ATSTRG	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
ATCCNTE	ATCE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CNDCTL	SPCND	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SRCND	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	STCND	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
NCMDQP	NCMDQP [31:0]	In reset	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
NRSPQP	NRSPQP [31:0]	In reset	In reset	In reset	Saved	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
NTDTBP0	NTDTBP0[31:0]	In reset	In reset	In reset	Saved	Saved	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
NIBIQP	NIBIQP [31:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	In reset	Saved	Saved	Saved	Saved	Saved

Table 26.18 Register States When Issuing Each Condition (5/11)

Register symbol	Register bit name	System reset	RSTCTL Register											
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST	HCMDQRST	HRSPQRST	HTDBRST	HRDBRST
NRSQP	NRSQP [31:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	In reset	Saved	Saved	Saved	Saved
NQTHCTL	IBIQTH [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	IBIDSSZ [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RSPQTH [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	CMDQTH [7: 0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
NTBTHCTL0	RXSTTH [2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TXSTTH [2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RXDBTH [2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TXDBTH [2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
NRQTHCTL	RSQTH [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BST	WUCNDDF	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TODF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ALF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TENDF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	NACKDF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HDREXDF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SPCNDDF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	STCNDDF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Table 26.18 Register States When Issuing Each Condition (6/11)

Register symbol	Register bit name	System reset	RSTCTL Register											
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST	HCMDQRST	HRSPQRST	HTDBRST	HRDBRST
BSTE	WUCNDDE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TODE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ALE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TENDE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	NACKDE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HDREXDE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SPCNDDE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	STCNDDE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BIE	WUCNDDIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TODIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ALIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TENDIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	NACKDIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HDREXDIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SPCNDDIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	STCNDDIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BSTFC	WUCNDDFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TODFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ALFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TENDFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	NACKDFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HDREXDFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SPCNDDFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	STCNDDFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Table 26.18 Register States When Issuing Each Condition (7/11)

Register symbol	Register bit name	System reset	RSTCTL Register											
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST	HCMDQRST	HRSPQRST	HTDBRST	HRDBRST
NTST	RSQFF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	In reset	Saved	Saved	Saved	Saved
	TEF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TABTF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RSPQFF	In reset	In reset	In reset	Saved	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	CMDQEF	In reset	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	IBIQEFF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	In reset	Saved	Saved	Saved	Saved	Saved
	RDBFF0	In reset	In reset	In reset	Saved	Saved	Saved	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	TDBEF0	In reset	In reset	In reset	Saved	Saved	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
NTSTE	RSQFE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TEE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TABTE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RSPQFE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	CMDQEE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	IBIQEFE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RDBFE0	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TDBEE0	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
NTIE	RSQFIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TEIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TABTIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RSPQFIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	CMDQEIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	IBIQEFIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RDBFIE0	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TDBEIE0	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Table 26.18 Register States When Issuing Each Condition (8/11)

Register symbol	Register bit name	System reset	RSTCTL Register											
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST	HCMDQRST	HRSPQRST	HTDBRST	HRDBRST
NTSTFC	RSQFFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TEFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TABTFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RSPQFFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	CMDQEFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	IBIQEFFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RDBFFC0	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TDBEFC0	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BCST	BIDLF	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	BAVLF	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	BFREF	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SVST	SVAF2	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SVAF1	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SVAF0	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HOAF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVIDF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HSMCF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	GCAF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
WUST	WUASYNF	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
MRCCPT	MRCCPT [31:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Table 26.18 Register States When Issuing Each Condition (9/11)

Register symbol	Register bit name	System reset	RSTCTL Register											
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST	HCMDQRST	HRSPQRST	HTDBRST	HRDBRST
DATBASm (m = 0 to 7)	DVTYP	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVNACK [1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVDYAD [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVIBITS	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVMRRJ	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVSIRRJ	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVIBIPL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVSTAD [9:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SDATBASy (y = 0 to 2)	SDDYAD [6:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SDIBIPL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SDADLS	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SDSTAD [9:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
MSDCTm (m = 0 to 7)	RBCR [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
EXDATBAS	EDTYP	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	EDNACK [1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	EDDYAD [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	EDSTAD [9:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SVDCT	TBCR [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TDCR [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SDCTPIDL	SDCTPIDL [31:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SDCTPIDH	SDCTPIDH [31:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SVDVADy (y = 0 to 2)	SDYADV	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SSTADV	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SADLG	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SVAD [9:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Table 26.18 Register States When Issuing Each Condition (10/11)

Register symbol	Register bit name	System reset	RSTCTL Register											
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST	HCMDQRST	HRSPQRST	HTDBRST	HRDBRST
CSECMD	HJEVE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	MSRQE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SVIRQE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CEACTST	ACTST [3:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CMWLG	MWLG[15:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CMRLG	IBIPSZ [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	MRLG [15:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CETSTMD	TSTMD [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CGDVST	VDRSV [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ACTMD [1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	PRTE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	PNDINT [3:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CMDSPW	MSWDR [2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CMDSPR	CDTTIM [2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	MSRDR [2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CMDSPT	MRTE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	MRTTIM [23:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CETSM	INAC [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	FREQ [7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SPTASYN [1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SPTSYN	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CETSS	ICOVF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ASYNE [1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SYNE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Table 26.18 Register States When Issuing Each Condition (11/11)

Register symbol	Register bit name	System reset	RSTCTL Register											
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST	HCMDQRST	HRSPQRST	HTDBRST	HRDBRST
BITCNT	BCNT [4:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
NQSTLV	IBISCNT [4:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	In reset	Saved	Saved	Saved	Saved	Saved
	IBIQLV [7:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	In reset	Saved	Saved	Saved	Saved	Saved
	RSPQLV [7:0]	In reset	In reset	In reset	Saved	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	CMDQFLV [7:0]	In reset	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
NDBSTLV0	RDBLV [7:0]	In reset	In reset	In reset	Saved	Saved	Saved	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	TDBFLV [7:0]	In reset	In reset	In reset	Saved	Saved	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
NRSQSTLV	RSQLV [7:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	In reset	Saved	Saved	Saved	Saved
PRSTDBG	SDOLV	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SCOLV	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SDILV	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SCILV	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
MSERRCNT	M2ECNT [7:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SC1CPT	SC1C [15:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SC2CPT	SC2C [15:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Note: In reset: To be reset (The FIFO corresponding to this register is cleared)

26.6 Usage Notes

26.6.1 Settings for the Operating Clock

The clock frequency ratio of PCLKD and PCLKB must be 2:1 or 1:1 when using the I3C module. Operation is not guaranteed for other settings.

27. Renesas Serial Peripheral Interface

This LSI circuit includes three independent Renesas serial peripheral interfaces.

This module is capable of full-duplex synchronous serial communication.

27.1 Features

This module has the following features.

SPI transfer functions

- Use of MOSI (master out/slave in), MISO (master in/slave out), SSL (slave select), and RSPCK (SPI clock) signals allows for serial communications through SPI operation (four-wire method).
- Capable of serial communications in master/slave mode
- Supports mode fault error detection (only in SPI slave mode)
- Supports overrun error detection (only in SPI slave mode)
- Switching of the polarity of the serial transfer clock
- Switching of the clock phase of serial transfer

Data format

- MSB-first/LSB-first selectable
- Transfer bit-length is selectable as 8, 16, or 32 bits.

Bit rate

- RSPCK can be divided by a maximum of 4096 in master mode
- RSPCK can be generated by dividing P0ϕ by the on-chip baud rate generator.
- An externally input clock can be used as a serial clock.

Buffer configuration

- 8 bytes for transmission and 32 bytes for reception

SSL control function

- One SSL signal for each channel
  - In master mode, outputs SSL signal.
  - In slave mode, inputs SSL signal.
- Controllable delay from SSL output assertion to RSPCK operation (RSPCK delay)
  - Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)
- Controllable delay from RSPCK stoppage to SSL output negation (SSL negation delay)
  - Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)
- Controllable wait for next-access SSL output assertion (next-access delay)
  - Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)

- Function for changing SSL polarity

Control in master transfer

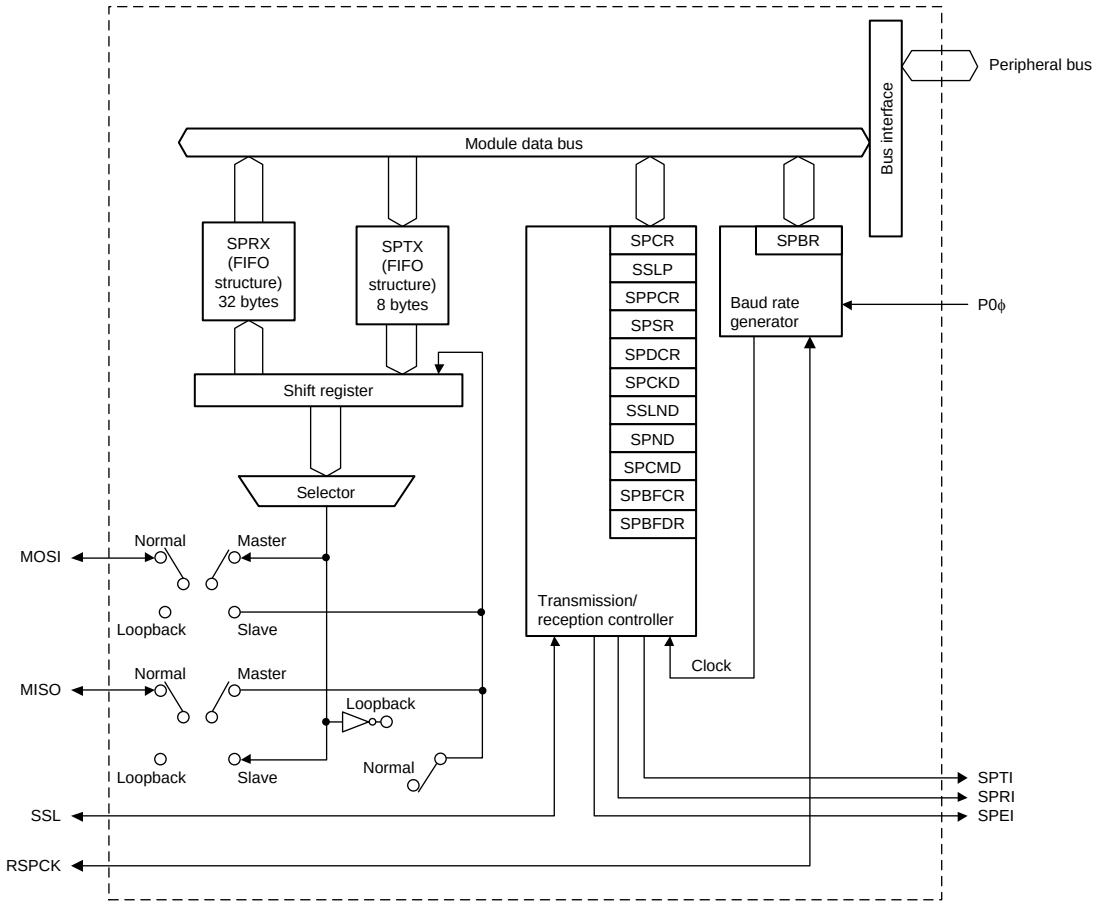
- A transfer of up to four commands can be executed sequentially in looped execution.
- For each command, the following can be set:
  - SSL signal value, bit rate, RSPCK polarity/phase, transfer data length, LSB/MSB first, burst, RSPCK delay, SSL negation delay, and next-access delay.
  - A transfer can be initiated by writing to the transmit buffer.
  - A transfer can be initiated by clearing the SPTEF bit.
  - MOSI signal value specifiable in SSL negation

Interrupt sources

- Maskable interrupt sources:
  - Receive interrupt (receive buffer full)
  - Transmit interrupt (transmit buffer empty)
  - Error interrupt (mode fault, overrun)

Others

- Provides loop back mode
- Provides a function for disabling (initializing) this module



[Legend]

SPCR:	Control register	SPCMD:	Command register
SSLP:	Slave select polarity register	SPBR:	Bit rate register
SPPCR:	Pin control register	SPTX:	Transmission buffer (Data register write side)
SPSR:	Status register	SPRX:	Receive buffer (Data register read side)
SPSCR:	Sequence control register	SPBFCR:	Buffer control register
SPSSR:	Sequence status register	SPBFDR:	Buffer data count setting register
SPDCR:	Data control register	SPTI:	Transmit interrupt
SPCKD:	Clock delay register	SPRI:	Receive interrupt
SSLND:	Slave select negate delay register	SPEI:	Error interrupt
SPND:	Next-access delay register		

Figure 27.1 Block Diagram (for One Channel)

27.2 Input/Output Pins

Table 27.1 shows the pin configuration. This module automatically switches the input/output direction of the SSL pin. SSL is set as an output in master mode and as an input in slave mode. Pins RSPCK, MOSI, and MISO are automatically set as inputs or outputs according to the setting of master or slave and the level input on SSL (see **Section 27.4.2, Pin Control**).

Table 27.1 Pin Configuration

Channel	Pin Name	Pin Name	I/O	Function
0	Clock pin	RSPI0_CK	I/O	Clock input/output
	Master transmit data pin	RSPI0_MOSI	I/O	Master transmit data
	Slave transmit data pin	RSPI0_MISO	I/O	Slave transmit data
	Slave select 0 pin	RSPI0_SSL	I/O	Slave selection
1	Clock pin	RSPI1_CK	I/O	Clock input/output
	Master transmit data pin	RSPI1_MOSI	I/O	Master transmit data
	Slave transmit data pin	RSPI1_MISO	I/O	Slave transmit data
	Slave select 0 pin	RSPI1_SSL	I/O	Slave selection
2	Clock pin	RSPI2_CK	I/O	Clock input/output
	Master transmit data pin	RSPI2_MOSI	I/O	Master transmit data
	Slave transmit data pin	RSPI2_MISO	I/O	Slave transmit data
	Slave select 0 pin	RSPI2_SSL	I/O	Slave selection
3	Clock pin	RSPI3_CK	I/O	Clock input/output
	Master transmit data pin	RSPI3_MOSI	I/O	Master transmit data
	Slave transmit data pin	RSPI3_MISO	I/O	Slave transmit data
	Slave select 0 pin	RSPI3_SSL	I/O	Slave selection
4	Clock pin	RSPI4_CK	I/O	Clock input/output
	Master transmit data pin	RSPI4_MOSI	I/O	Master transmit data
	Slave transmit data pin	RSPI4_MISO	I/O	Slave transmit data
	Slave select 0 pin	RSPI4_SSL	I/O	Slave selection

Note: In the description of the pins, the channel is omitted, and pin names are described as RSPCK, MOSI, MISO, and SSL.

27.3 Register Descriptions

Table 27.2 shows the register configuration. The address of the RSPI register is represented by the offset address from the base address. RSPI base address is as follows:

Base address: H'0_100A_0000 (Overall Address Space)
Base address: H'400A_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
Base address: H'500A_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 27.2 Register Configuration (1/3)

Channel	Register Name	Abbreviation*1	R/W	Initial Value	Offset Address	Access Size
0	Control register_0	SPCR_0	R/W	H'00	H'A000	8
	Slave select polarity register_0	SSLP_0	R/W	H'00	H'A001	8
	Pin control register_0	SPPCR_0	R/W	H'00	H'A002	8
	Status register_0	SPSR_0	R/(W)*2	H'60	H'A003	8
	Data register_0	SPDR_0	R/W	Undefined	H'A004	8, 16, 32
	Sequence control register_0	SPSCR_0	R/W	H'00	H'A008	8
	Sequence status register_0	SPSSR_0	R	H'00	H'A009	8
	Bit rate register_0	SPBR_0	R/W	H'FF	H'A00A	8
	Data control register_0	SPDCR_0	R/W	H'20	H'A00B	8
	Clock delay register_0	SPCKD_0	R/W	H'00	H'A00C	8
	Slave select negation delay register_0	SSLND_0	R/W	H'00	H'A00D	8
	Next-access delay register_0	SPND_0	R/W	H'00	H'A00E	8
	Command register0_0	SPCMD0_0	R/W	H'070D	H'A010	16
	Command register1_0	SPCMD1_0	R/W	H'070D	H'A012	16
	Command register2_0	SPCMD2_0	R/W	H'070D	H'A014	16
	Command register3_0	SPCMD3_0	R/W	H'070D	H'A016	16
	Buffer control register_0	SPBFCR_0	R/W	H'00	H'A020	8
	Buffer data count setting register_0	SPBFDR_0	R	H'0000	H'A022	16

Table 27.2 Register Configuration (2/3)

Channel	Register Name	Abbreviation*1	R/W	Initial Value	Offset Address	Access Size
1	Control register_1	SPCR_1	R/W	H'00	H'A400	8
	Slave select polarity register_1	SSLP_1	R/W	H'00	H'A401	8
	Pin control register_1	SPPCR_1	R/W	H'00	H'A402	8
	Status register_1	SPSR_1	R/(W)*2	H'60	H'A403	8
	Data register_1	SPDR_1	R/W	Undefined	H'A404	8, 16, 32
	Sequence control register_1	SPSCR_1	R/W	H'00	H'A408	8
	Sequence status register_1	SPSSR_1	R	H'00	H'A409	8
	Bit rate register_1	SPBR_1	R/W	H'FF	H'A40A	8
	Data control register_1	SPDCR_1	R/W	H'20	H'A40B	8
	Clock delay register_1	SPCKD_1	R/W	H'00	H'A40C	8
	Slave select negation delay register_1	SSLND_1	R/W	H'00	H'A40D	8
	Next-access delay register_1	SPND_1	R/W	H'00	H'A40E	8
	Command register0_1	SPCMD0_1	R/W	H'070D	H'A410	16
	Command register1_1	SPCMD1_1	R/W	H'070D	H'A412	16
	Command register2_1	SPCMD2_1	R/W	H'070D	H'A414	16
	Command register3_1	SPCMD3_1	R/W	H'070D	H'A416	16
	Buffer control register_1	SPBFCR_1	R/W	H'00	H'A420	8
	Buffer data count setting register_1	SPBFDR_1	R	H'0000	H'A422	16
2	Control register_2	SPCR_2	R/W	H'00	H'A800	8
	Slave select polarity register_2	SSLP_2	R/W	H'00	H'A801	8
	Pin control register_2	SPPCR_2	R/W	H'00	H'A802	8
	Status register_2	SPSR_2	R/(W)*2	H'60	H'A803	8
	Data register_2	SPDR_2	R/W	Undefined	H'A804	8, 16, 32
	Sequence control register_2	SPSCR_2	R/W	H'00	H'A808	8
	Sequence status register_2	SPSSR_2	R	H'00	H'A809	8
	Bit rate register_2	SPBR_2	R/W	H'FF	H'A80A	8
	Data control register_2	SPDCR_2	R/W	H'20	H'A80B	8
	Clock delay register_2	SPCKD_2	R/W	H'00	H'A80C	8
	Slave select negation delay register_2	SSLND_2	R/W	H'00	H'A80D	8
	Next-access delay register_2	SPND_2	R/W	H'00	H'A80E	8
	Command register0_2	SPCMD0_2	R/W	H'070D	H'A810	16
	Command register1_2	SPCMD1_2	R/W	H'070D	H'A812	16
	Command register2_2	SPCMD2_2	R/W	H'070D	H'A814	16
	Command register3_2	SPCMD3_2	R/W	H'070D	H'A816	16
	Buffer control register_2	SPBFCR_2	R/W	H'00	H'A820	8
	Buffer data count setting register_2	SPBFDR_2	R	H'0000	H'A822	16

Table 27.2 Register Configuration (3/3)

Channel	Register Name	Abbreviation*1	R/W	Initial Value	Offset Address	Access Size
3	Control register_3	SPCR_3	R/W	H'00	H'AC00	8
	Slave select polarity register_3	SSLP_3	R/W	H'00	H'AC01	8
	Pin control register_3	SPPCR_3	R/W	H'00	H'AC02	8
	Status register_3	SPSR_3	R/(W)*2	H'60	H'AC03	8
	Data register_3	SPDR_3	R/W	Undefined	H'AC04	8, 16, 32
	Sequence control register_3	SPSCR_3	R/W	H'00	H'AC08	8
	Sequence status register_3	SPSSR_3	R	H'00	H'AC09	8
	Bit rate register_3	SPBR_3	R/W	H'FF	H'AC0A	8
	Data control register_3	SPDCR_3	R/W	H'20	H'AC0B	8
	Clock delay register_3	SPCKD_3	R/W	H'00	H'AC0C	8
	Slave select negation delay register_3	SSLND_3	R/W	H'00	H'AC0D	8
	Next-access delay register_3	SPND_3	R/W	H'00	H'AC0E	8
	Command register0_3	SPCMD0_3	R/W	H'070D	H'AC10	16
	Command register1_3	SPCMD1_3	R/W	H'070D	H'AC12	16
	Command register2_3	SPCMD2_3	R/W	H'070D	H'AC14	16
	Command register3_3	SPCMD3_3	R/W	H'070D	H'AC16	16
	Buffer control register_3	SPBFCR_3	R/W	H'00	H'AC20	8
	Buffer data count setting register_3	SPBFDR_3	R	H'0000	H'AC22	16
4	Control register_4	SPCR_4	R/W	H'00	H'B000	8
	Slave select polarity register_4	SSLP_4	R/W	H'00	H'B001	8
	Pin control register_4	SPPCR_4	R/W	H'00	H'B002	8
	Status register_4	SPSR_4	R/(W)*2	H'60	H'B003	8
	Data register_4	SPDR_4	R/W	Undefined	H'B004	8, 16, 32
	Sequence control register_4	SPSCR_4	R/W	H'00	H'B008	8
	Sequence status register_4	SPSSR_4	R	H'00	H'B009	8
	Bit rate register_4	SPBR_4	R/W	H'FF	H'B00A	8
	Data control register_4	SPDCR_4	R/W	H'20	H'B00B	8
	Clock delay register_4	SPCKD_4	R/W	H'00	H'B00C	8
	Slave select negation delay register_4	SSLND_4	R/W	H'00	H'B00D	8
	Next-access delay register_4	SPND_4	R/W	H'00	H'B00E	8
	Command register0_4	SPCMD0_4	R/W	H'070D	H'B010	16
	Command register1_4	SPCMD1_4	R/W	H'070D	H'B012	16
	Command register2_4	SPCMD2_4	R/W	H'070D	H'B014	16
	Command register3_4	SPCMD3_4	R/W	H'070D	H'B016	16
	Buffer control register_4	SPBFCR_4	R/W	H'00	H'B020	8
	Buffer data count setting register_4	SPBFDR_4	R	H'0000	H'B022	16

Note 1. In the description of the register names, the channel is omitted.

Note 2. Only 0 can be written to clear the flag.

27.3.1 Control Register (SPCR)

SPCR sets the operating mode. If the MSTR and MODFEN bits are changed while the function of this module is enabled by setting the SPE bit to 1, subsequent operations cannot be guaranteed.

Bit	7	6	5	4	3	2	1	0
	SPRIE	SPE	SPTIE	SPEIE	MSTR	MODFEN	—	—
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	SPRIE	0	R/W	Receive Interrupt Enable Enables or disables generation of receive interrupt requests (SPRI) when the number of receive data units in the receive buffer (SPRX) is equal to or greater than the specified receive buffer data triggering number and the SPRF flag in SPSR is set to 1. 0: Disables the generation of receive interrupt requests. 1: Enables the generation of receive interrupt requests.
6	SPE	0	R/W	Function Enable Setting this bit to 1 enables the module function. When the MODF bit in the status register (SPSR) is 1, the SPE bit cannot be set to 1 (see **Section 27.4.6, Error Detection**). Setting the SPE bit to 0 disables the module function, and initializes a part of the module function (see **Section 27.4.7, Initialization**). 0: Disables the module function. 1: Enables the module function.
5	SPTIE	0	R/W	Transmit Interrupt Enable Enables or disables generation of transmit interrupt requests (SPTI) when the number of transmit data units in the transmit buffer (SPTX) is equal to or less than the specified transmit buffer data triggering number and the SPTEF flag in SPSR is set to 1. 0: Disables the generation of transmit interrupt requests. 1: Enables the generation of transmit interrupt requests.
4	SPEIE	0	R/W	Error Interrupt Enable Enables or disables the generation of error interrupt requests when this module detects a mode fault error and sets the MODF bit in the status register (SPSR) to 1, or when this module detects an overrun error and sets the OVRF bit in SPSR to 1 (see **Section 27.4.6, Error Detection**). 0: Disables the generation of error interrupt requests. 1: Enables the generation of error interrupt requests. *Note*: This bit is valid only in SPI slave mode.
3	MSTR	0	R/W	Master/Slave Mode Select Selects master/slave mode. According to MSTR bit settings, this module determines the direction of the RSPCK, MOSI, MISO, and SSL pins. 0: Slave mode 1: Master mode
2	MODFEN	0	R/W	Mode Fault Error Detection Enable Enables or disables the detection of a mode fault error (see **Section 27.4.6, Error Detection**). 0: Disables the detection of a mode fault error. 1: Enables the detection of a mode fault error. *Note*: This bit is valid only in SPI slave mode. When master mode is specified with the MSTR bit, this bit should always be cleared to 0.
1, 0	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

27.3.2 Slave Select Polarity Register (SSLP)

SSLP sets the polarity of the SSL signal. If the contents of SSL0P are changed while the function of this module is enabled by setting the SPE bit in the control register (SPCR) to 1, subsequent operations cannot be guaranteed.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SSL0P
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
0	SSL0P	0	R/W	SSL Signal Polarity Setting Sets the polarity of the SSL signal. The value of SSL0P indicates the active polarity of the SSL signal. 0: SSL signal 0-active 1: SSL signal 1-active

27.3.3 Pin Control Register (SPPCR)

SPPCR sets the modes of the pins. If the contents of this register are changed while the function of this module is enabled by setting the SPE bit in the control register (SPCR) to 1, subsequent operations cannot be guaranteed.

Bit	7	6	5	4	3	2	1	0
	—	—	MOIFE	MOIFV	—	—	—	SPLP
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
5	MOIFE	0	R/W	MOSI Idle Value Fixing Enable Fixes the MOSI output value when this module in master mode is in an SSL negation period (including the SSL retention period during a burst transfer). When MOIFE is 0, this module outputs the last output value from the previous serial transfer during the SSL negation period to the MOSI pin. (The value is undefined when CPHA is 0). When MOIFE is 1, this module outputs the fixed value set in the MOIFV bit to the MOSI pin. 0: MOSI output value equals the last output value from previous transfer. (The value is undefined when CPHA is 0). 1: MOSI output value equals the value set in the MOIFV bit.
4	MOIFV	0	R/W	MOSI Idle Fixed Value If the MOIFE bit is 1 in master mode, this module, according to MOIFV bit settings, determines the MOSI signal value during the SSL negation period (including the SSL retention period during a burst transfer). 0: MOSI Idle fixed value equals 0. 1: MOSI Idle fixed value equals 1.
3 to 1	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
0	SPLP	0	R/W	Loopback When the SPLP bit is set to 1, this module shuts off the path between the MISO pin and the shift register, and between the MOSI pin and the shift register, and connects (reverses) the input path and the output path for the shift register. 0: Normal mode 1: Loopback mode

27.3.4 Status Register (SPSR)

SPSR indicates the operating status.

Bit	7	6	5	4	3	2	1	0
	SPRF	TEND	SPTEF	—	—	MODF	—	OVRF
Initial Value	0	1	1	0	0	0	0	0
R/W	R	R	R	R	R	R/(W)*1	R	R/(W)*1

Note 1. Only 0 can be written to clear the flag after reading 1.

Bit	Bit Name	Initial Value	R/W	Description
7	SPRF	0	R	Receive Buffer Full Flag Indicates that the number of receive data units in the receive buffer (SPRX) is equal to or greater than the receive buffer data triggering number specified in the buffer control register (SPBFCR). 0: The number of receive data units in the receive buffer is less than the receive buffer data triggering number. 1: The number of receive data units in the receive buffer is equal to or greater than the receive buffer data triggering number. [Clearing conditions] • The receive buffer data is read until the number of data units in the receive buffer becomes less than the specified receive buffer data triggering number. • Receive buffer data reset is enabled. • Power-on reset [Setting condition] • The number of data units in the receive buffer is equal to or greater than the specified receive buffer data triggering number.
6	TEND	1	R	Transmit End This bit is set to 1 when transmission is completed, and this bit is 0 when transmission is not completed. [Clearing condition] • When transmit data are transferred from the transmit register to the shift register. [Setting condition] • When the number of data units in the transmit buffer (SPTX) is zero when a serial transfer is completed. *Note*: This bit is valid only in SPI master mode.
5	SPTEF	1	R	Transmit Buffer Empty Flag Indicates that the number of transmit data units in the transmit buffer (SPTX) is equal to or less than the transmit buffer data triggering number specified in the buffer control register (SPBFCR). 0: The number of transmit data units in the transmit buffer is equal to or greater than the specified transmit buffer data triggering number. 1: The number of transmit data units in the transmit buffer is less than the specified transmit buffer data triggering number. [Clearing condition] • When data is written to the transmit buffer until the number of transmit data units in the transmit buffer exceeds the specified transmit buffer data triggering number. [Setting conditions] • When the number of transmit data units in the transmit buffer is less than the specified transmit buffer data triggering number. • When transmit buffer data reset is enabled. • Power-on reset
4, 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

Bit	Bit Name	Initial Value	R/W	Description
2	MODF	0	R/(W) *1	Mode Fault Error Flag Indicates the occurrence of a mode fault error. If the MODFEN bit is set to 1 when this module is in slave mode and the SSL pin is negated before the RSPCK cycle necessary for data transfer ends, this module detects a mode fault error. The active level of the SSL signal is determined by the SSL0P bit in the slave select polarity register (SSLP). [Clearing conditions] • SPSR is read when the MODF bit is 1, and then 0 is written to the MODF bit. • Power-on reset 0: No mode fault error occurred 1: A mode fault error occurred *Note*: This bit is valid only in SPI slave mode.
1	—	0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
0	OVRF	0	R/(W) *1	Overrun Error Flag Indicates the occurrence of an overrun error. If a serial transfer ends when there is not enough space for receiving the specified length of data in the receive buffer (SPRX), this module detects an overrun error, and sets the OVRF bit to 1. [Clearing conditions] • SPSR is read when the OVRF bit is 1, and then 0 is written to the OVRF bit. • Power-on reset 0: No overrun error occurred 1: An overrun error occurred *Note*: This bit is valid only in SPI slave mode.

Note 1. Only 0 can be written to clear the flag after reading 1.

27.3.5 Data Register (SPDR)

SPDR is a buffer that holds data for transmission and reception.

The transmit buffer (SPTX) and receive buffer (SPRX) are independent and are mapped to SPDR.

SPDR should be read or written to in byte, word, or longword units according to the access width specification bit (SPLW) in the data control register (SPDCR).

The bit length to be used is determined by the data length specification bits (SPB3 to SPB0) in the command register (SPCMD).

The access width set by SPDCR must agree with the data length set by SPCMD.

When data is written to SPDR, the data will be written to the transmit buffer from SPDR if the transmit buffer has a space equal to or more than the SPDR access width. If there is not enough space, data will not be written to the transmit buffer. Even if an attempt is made to write data to the buffer, the data is ignored.

When data is read from SPDR, receive data in the receive buffer will be read. If SPDR is read when there is no receive data in the receive buffer, the read value is undefined.

When SPDR is written to with the longword-, word-, or byte-access width, the transmit data should be written to address 0 irrespective of the access width. If data is written to the other addresses, the data is not guaranteed.

When SPDR is read with the longword-, word-, or byte-access width, the receive data should be read from address 0. If data is read from the other addresses, the data is not guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24	SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
Initial Value	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
Initial Value	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note 1. Undefined

27.3.6 Sequence Control Register (SPSCR)

SPSCR sets the sequence control method when this module operates in master mode. If the contents of SPSCR are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPSLN1	SPSLN0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
1	SPSLN1	0	R/W	Sequence Length Specification These bits specify a sequence length when this module in master mode performs sequential operations. This module in master mode changes command registers 0 to 3 (SPCMD0 to SPCMD3) to be referenced and the order in which they are referenced according to the sequence length that is set in the SPSLN1 and SPSLN0 bits. The relationship among the setting of bits SPSLN1 and SPSLN0, sequence length, and SPCMD0 to SPCMD3 referenced by this module is shown below. In slave mode, SPCMD0 is always referenced. Sequence Length / Referenced SPCMD # 00: 1 — 0 → 0 → … 01: 2 — 0 → 1 → 0 → … 10: 3 — 0 → 1 → 2 → 0 → … 11: 4 — 0 → 1 → 2 → 3 → 0 → …
0	SPSLN0	0	R/W	

27.3.7 Sequence Status Register (SPSSR)

SPSSR indicates the sequence control status when this module operates in master mode.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPCP1	SPCP0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
1	SPCP1	0	R	Command Pointer During sequence control, these bits indicate one of the command registers 0 to 3 (SPCMD0 to SPCMD3) that is currently pointed to by the pointer. The relationship between the setting of SPCP1 and SPCP0 and SPCMD0 to SPCMD3 is shown below. For the sequence control, see **Section 27.4.8(1)(c), Sequence Control**. 00: SPCMD0 01: SPCMD1 10: SPCMD2 11: SPCMD3
0	SPCP0	0	R	

27.3.8 Bit Rate Register (SPBR)

SPBR sets the bit rate in master mode. If the contents of SPBR are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

Bit	7	6	5	4	3	2	1	0
	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0
Initial Value	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When this module is used in slave mode, the bit rate depends on the bit rate of the input clock regardless of the settings of SPBR and BRDV.

The bit rate is determined by combinations of SPBR settings and the bit settings in the BRDV1 and BRDV0 bits in the command registers (SPCMD0 to SPCMD3). The equation for calculating the bit rate is given below. In the equation, n denotes an SPBR setting (0, 1, 2, …, 255), and N denotes a BRDV1 and BRDV0 bit setting (0, 1, 2, 3).

$$\text{Bit rate} = \frac{\text{f (P0}\phi\text{)}}{2 \times (\text{n} + 1) \times 2^{\text{N}}}$$

Table 27.3 shows examples of the relationship between the SPBR register and BRDV1 and BRDV0 bit settings.

Table 27.3 Relationship between SPBR and BRDV1 and BRDV0 Settings

SPBR (n)	BRDV[1:0] (N)	Division Ratio	Bit Rate*1
			P0ϕ = 100 MHz
0	0	2	Not available
1	0	4	25.00 Mbps
2	0	6	16.67 Mbps
3	0	8	12.50 Mbps
4	0	10	10.00 Mbps
5	0	12	8.33 Mbps
5	1	24	4.17 Mbps
5	2	48	2.08 Mbps
5	3	96	1.04 Mbps
255	3	4096	24.41 Kbps

Note 1. Decide the bit rate to be actually used in the system considering timing specifications.

27.3.9 Data Control Register (SPDCR)

SPDCR selects the width to access SPDR from longword-, word-, and byte-width, and enables or disables dummy data transmission for the master mode operation.

If the contents of SPDCR are changed while bit TEND in the status register (SPSR) indicates that transmission is not completed, the subsequent operation cannot be guaranteed.

Bit	7	6	5	4	3	2	1	0
	TXDMY	SPLW1	SPLW0	—	—	—	—	—
Initial Value	0	0	1	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	TXDMY	0	R/W	Dummy Data Transmission Enable Enables or disables dummy data transmission. When communication is performed with this bit set to 1, dummy data is transmitted from the MOSI pin and a serial communication can be performed even if there is no transmit data in the transmit buffer. Specifically, if there is no transmit data in the transmit buffer and this bit is set to 1, dummy data is transferred to the shift register. Data previously transmitted from the pin is used as dummy data. If this bit is set to 1 after the initialization and a transfer is performed, the transmitted dummy data is undefined. 0: Disables dummy data transmission. 1: Enables dummy data transmission. *Note*: This bit is valid only in the master mode.
6	SPLW1	0	R/W	Access Width Specification Specifies the width for accessing the data register (SPDR). If the length of data transferred to SPDR does not agree with these bit settings, operation is not guaranteed.*1 00: Setting prohibited 01: SPDR is accessed in bytes (8 bits). 10: SPDR is accessed in words (16 bits). 11: SPDR is accessed in longwords (32 bits).
5	SPLW0	1	R/W	
4 to 0	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

Note 1. The data length is specified by the SPB3 to SPB0 bits in the command register (SPCMD).
See **Section 27.3.5, Data Register (SPDR)**.

27.3.10 Clock Delay Register (SPCKD)

SPCKD sets a period from the beginning of SSL signal assertion to RSPCK oscillation (RSPCK delay) when the SCKDEN bit in the command register (SPCMD) is 1. If the contents of SPCKD are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

When using this module in slave mode, set 000b to SCKDL2 to SCKDL0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	SCKDL2	SCKDL1	SCKDL0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2	SCKDL2	0	R/W	RSPCK Delay Setting These bits set an RSPCK delay value when the SCKDEN bit in SPCMD is 1. The relationship between the setting of SCKDL2 to SCKDL0 and the RSPCK delay value is shown below. 000: 1 RSPCK 001: 2 RSPCK 010: 3 RSPCK 011: 4 RSPCK 100: 5 RSPCK 101: 6 RSPCK 110: 7 RSPCK 111: 8 RSPCK
1	SCKDL1	0	R/W	
0	SCKDL0	0	R/W	

27.3.11 Slave Select Negation Delay Register (SSLND)

SSLND sets a period (SSL negation delay) from the transmission of a final RSPCK edge to the negation of the SSL signal during a serial transfer by this module in master mode. If the contents of SSLND are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

When using this module in slave mode, set 000b to SLNDL2 to SLNDL0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLNDL2	SLNDL1	SLNDL0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2	SLNDL2	0	R/W	SSL Negation Delay Setting These bits set an SSL negation delay when the SLNDEN bit in SPCMD is 1. The relationship between the setting of SLNDL2 to SLNDL0 and the SSL negation delay value is shown below. 000: 1 RSPCK 001: 2 RSPCK 010: 3 RSPCK 011: 4 RSPCK 100: 5 RSPCK 101: 6 RSPCK 110: 7 RSPCK 111: 8 RSPCK
1	SLNDL1	0	R/W	
0	SLNDL0	0	R/W	

27.3.12 Next-Access Delay Register (SPND)

SPND sets a non-active period (next-access delay) after termination of a serial transfer when the SPNDEN bit in the command register (SPCMD) is 1. If the contents of SPND are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

When using this module in slave mode, set 000b to SPNDL2 to SPNDL0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	SPNDL2	SPNDL1	SPNDL0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2	SPNDL2	0	R/W	Next-Access Delay Setting These bits set a next-access delay when the SPNDEN bit in SPCMD is 1. The relationship between the setting of SPNDL2 to SPNDL0 and the next-access delay value is shown below. 000: 1 RSPCK + 2 PΦ 001: 2 RSPCK + 2 PΦ 010: 3 RSPCK + 2 PΦ 011: 4 RSPCK + 2 PΦ 100: 5 RSPCK + 2 PΦ 101: 6 RSPCK + 2 PΦ 110: 7 RSPCK + 2 PΦ 111: 8 RSPCK + 2 PΦ
1	SPNDL1	0	R/W	
0	SPNDL0	0	R/W	

27.3.13 Command Register (SPCMD)

Each channel has four command registers (SPCMD0 to SPCMD3). SPCMD0 to SPCMD3 are used to set a transfer format for master mode operation. Some of the bits in SPCMD0 are used to set a transfer mode for slave mode operation. In master mode, this module sequentially references SPCMD0 to SPCMD3 according to the settings in bits SPSLN1 and SPSLN0 in the sequence control register (SPSCR), and executes the serial transfer that is set in the referenced SPCMD. While bit TEND in the status register (SPSR) indicates that transmission is not completed, correct operation of this module cannot be guaranteed if SPCMD is changed that is referred by this module. SPCMD referenced by this module in master mode can be checked by means of bits SPCP1 and SPCP0 in the sequence status register (SPSSR). When the function of this module in slave mode is enabled, operation cannot be guaranteed if the value set in SPCMD0 is changed.

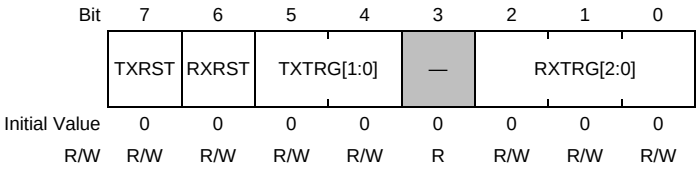
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0	SSLKP	—	—	—	BRDV1	BRDV0	CPOL	CPHA
Initial Value	0	0	0	0	0	1	1	1	0	0	0	0	1	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	SCKDEN	0	R/W	RSPCK Delay Setting Enable Sets the period from the point this module in master mode activates the SSL signal until the RSPCK starts oscillation (RSPCK delay). If the SCKDEN bit is 0, this module sets the RSPCK delay to 1 RSPCK. If the SCKDEN bit is 1, this module starts the oscillation of RSPCK at an RSPCK delay in compliance with the clock delay register (SPCKD) settings. To use this module in slave mode, the SCKDEN bit should be set to 0. 0: An RSPCK delay of 1 RSPCK 1: An RSPCK delay equal to SPCKD settings.
14	SLNDEN	0	R/W	SSL Negation Delay Setting Enable Sets the period from the point this module in master mode stops RSPCK oscillation until this module sets the SSL signal inactive (SSL negation delay). If the SLNDEN bit is 0, this module sets the SSL negation delay to 1 RSPCK. If the SLNDEN bit is 1, this module negates the SSL signal at an SSL negation delay in compliance with the slave select negation delay register (SSLND) settings. To use this module in slave mode, the SLNDEN bit should be set to 0. 0: An SSL negation delay of 1 RSPCK 1: An SSL negation delay equal to SSLND settings.
13	SPNDEN	0	R/W	Next-Access Delay Enable Sets the period from the point this module in master mode terminates a serial transfer and sets the SSL signal inactive until this module enables the SSL signal assertion for the next access (next-access delay). If the SPNDEN bit is 0, this module sets the next-access delay to 1 RSPCK + 2P0ϕ. If the SPNDEN bit is 1, this module inserts a next-access delay in compliance with the next-access delay register (SPND) settings. To use this module in slave mode, the SPNDEN bit should be set to 0. 0: A next-access delay of 1 RSPCK + 2 P0ϕ 1: A next-access delay equal to SPND settings.
12	LSBF	0	R/W	LSB First Sets the data format in master mode or slave mode to MSB first or LSB first. 0: MSB first 1: LSB first

Bit	Bit Name	Initial Value	R/W	Description
11	SPB3	0	R/W	Data Length Setting These bits set a transfer data length in master mode or slave mode. 0100 to 0111: 8 bits 1111: 16 bits 0010, 0011: 32 bits Others: Setting prohibited
10	SPB2	1	R/W	
9	SPB1	1	R/W	
8	SPB0	1	R/W	
7	SSLKP	0	R/W	SSL Signal Level Keeping When this module in master mode performs a serial transfer, this bit specifies whether the SSL signal level for the current command is to be kept or negated between the SSL negation timing associated with the current command and the SSL assertion timing associated with the next command. To use this module in slave mode, the SSLKP bit should be set to 0. 0: Negates the SSL signal upon completion of transfer. 1: Keeps the SSL signal level from the end of the transfer until the beginning of the next access.
6 to 4	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
3	BRDV1	1	R/W	Bit Rate Division Setting These bits are used to determine the bit rate. A bit rate is determined by combinations of bits BRDV1 and BRDV 0 and the settings in the bit rate register (SPBR) (see **Section 27.3.8, Bit Rate Register (SPBR)**). The settings in SPBR determine the base bit rate. The settings in bits BRDV1 and BRDV0 are used to select a bit rate which is obtained by dividing the base bit rate by 1, 2, 4, or 8. In the bits SPCMD0 to SPCMD3, different BRDV1 and BRDV0 settings can be specified. This permits the execution of serial transfers at a different bit rate for each command. 00: Select the base bit rate. 01: Select the base bit rate divided by 2. 10: Select the base bit rate divided by 4. 11: Select the base bit rate divided by 8.
2	BRDV0	1	R/W	
1	CPOL	0	R/W	RSPCK Polarity Setting Sets an RSPCK polarity in master or slave mode. When data communication is performed between the Renesas serial peripheral interface module and the other modules, the same RSPCK polarity should be set for both modules. 0: RSPCK = 0 when idle 1: RSPCK = 1 when idle
0	CPHA	1	R/W	RSPCK Phase Setting Sets an RSPCK phase in master or slave mode. When data communication is performed between the Renesas serial peripheral interface module and the other modules, the same RSPCK phase should be set for both modules. 0: Data sampling on odd edge, data variation on even edge 1: Data variation on odd edge, data sampling on even edge

27.3.14 Buffer Control Register (SPBFCR)

SPBFCR resets the number of data units in the transmit buffer (SPTX) or receive buffer (SPRX) and sets the number of triggering data units.



Bit	Bit Name	Initial Value	R/W	Description
7	TXRST	0	R/W	Transmit Buffer Data Reset Resets the transmit buffer to an empty state. Transmit data in the transmit buffer becomes invalid when this bit is set to 1. 0: Disables the reset operation.*1 1: Enables the reset operation *Note 1.* The reset operation is performed after a power-on reset.
6	RXRST	0	R/W	Receive Buffer Data Reset Resets the receive buffer to an empty state. Receive data in the receive buffer becomes invalid when this bit is set to 1. 0: Disables the reset operation.*1 1: Enables the reset operation *Note 1.* The reset operation is performed after a power-on reset.
5, 4	TXTRG[1:0]	00	R/W	Transmit Buffer Data Triggering Number*2 Specifies the timing at which the transmit buffer empty state is determined, that is when the SPTEF flag in the status register is set. When the number of bytes of data in the transmit buffer (SPTX) is equal to or less than the specified triggering number, the SPTEF flag is set to 1. 00: 7 bytes (1)*1 01: 6 bytes (2)*1 10: 4 bytes (4)*1 11: 0 bytes (8)*1 *Note 1.* The value in the parenthesis shows the number of available bytes in the transmit buffer (SPTX). *Note 2.* When transferring the data by using DMA, set the following values according to the data transfer length. (The data transfer length is set in SPBn (n = 0-3) of the command register (SPCMD)). 1) Data transfer length is 1 byte SPBFCR (TXTRG): 01 (6 bytes) or 10 (4 bytes) or 11 (0 byte) 2) Data transfer length is 2 bytes SPBFCR (TXTRG): 10 (4 bytes) or 11 (0 byte) 3) Data transfer length is 4bytes SPBFCR (TXTRG): 11 (0 byte)
3	—	0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	RXTRG[2:0]	000	R/W	Receive Buffer Data Triggering Number Specifies the timing at which the receive buffer full state is determined, that is when the SPRF flag in the status register is set. When the number of bytes of data in the receive buffer (SPRX) is equal to or greater than the specified triggering number, the SPRF flag is set to 1. 000: 1 byte (31)*1 001: 2 bytes (30)*1 010: 4 bytes (28)*1 011: 8 bytes (24)*1 100: 16 bytes (16)*1 101: 24 bytes (8)*1 110: 32 bytes (0)*1 111: 5 bytes (27)*1 *Note 1.* The value in the parenthesis shows the number of available bytes in the receive buffer (SPRX).

27.3.15 Buffer Data Count Setting Register (SPBFDR)

SPBFDR indicates the number of data units stored in the transmit buffer (SPTX) and receive buffer (SPRX). The upper eight bits indicate the number of transmit data units in SPTX and the lower eight bits indicate the number of receive data units in SPRX.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	T[3:0]				—	—	R[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
11 to 8	T[3:0]	0000	R	Indicates the number of bytes of data to be transmitted in SPTX. 0000b indicates that SPTX is empty. 1000b indicates that SPTX is full.
7, 6	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
5 to 0	R[5:0]	000000	R	Shows the number of bytes of received data in SPRX. 000000b indicates that SPRX is empty. 100000b indicates that SPRX is full.

27.4 Operation

In this section, the serial transfer period means a period from the beginning of driving valid data to the fetching of the final valid data.

27.4.1 Overview of Operations

This module is capable of serial transfers in slave mode and master mode. A particular mode of this module can be selected by using the MSTR bit in the control register (SPCR). **Table 27.4** gives the relationship between the modes and SPCR settings, and a description of each mode.

Table 27.4 Relationship between Modes and SPCR and Description of Each Mode

Mode	Slave (SPI Operation)	Master (SPI Operation)
MSTR bit setting	0	1
MODFEN bit setting	0 or 1	0
RSPCK signal	Input	Output
MOSI signal	Input	Output
MISO signal	Output/Hi-Z	Input
SSL signal	Input	Output
SSL polarity modification function	Supported	Supported
Transfer rate	Up to P0ϕ/8	Up to P0ϕ/2 (Up to P0ϕ/4 when P0ϕ is 100 MHz)
Clock source	RSPCK input	On-chip baud rate generator
Clock polarity	Two	Two
Clock phase	Two	Two
First transfer bit	MSB/LSB	MSB/LSB
Transfer data length	8, 16, or 32 bits	8, 16, or 32 bits
Burst transfer	Possible (CPHA = 1)	Possible (CPHA = 0, 1)
RSPCK delay control	Not supported	Supported
SSL negation delay control	Not supported	Supported
Next-access delay control	Not supported	Supported
Transfer activation method	SSL input active or RSPCK oscillation	Transmit buffer is written when SPE = 1
Sequence control	Not supported	Supported
Transmit buffer empty detection	Supported	Supported
Receive buffer full detection	Supported	Supported
Overrun error detection	Supported	Not Supported
Mode fault error detection	Supported (MODFEN = 1)	Not supported

27.4.2 Pin Control

According to the MSTR bit in the control register (SPCR), this module can automatically switch pin directions and output modes. **Table 27.5** shows the relationship between pin states and bit settings.

Table 27.5 Relationship between Pin States and Bit Settings

Mode	Pin	Pin State*1
Master mode (SPI operation) (MSTR = 1)	RSPCK	CMOS output
	SSL	CMOS output
	MOSI	CMOS output
	MISO	Input
Slave mode (SPI operation) (MSTR = 0)	RSPCK	Input
	SSL	Input
	MOSI	Input
	MISO*1	CMOS output/Hi-Z

Note 1. When SSL is at the non-active level or the SPE bit in SPCR is cleared to 0, the pin state is Hi-Z.

This module in master mode (SPI operation) determines MOSI signal values during the SSL negation period (including the SSL retention period during a burst transfer) according to MOIFE and MOIFV bit settings in SPPCR, as shown in **Table 27.6**.

Table 27.6 MOSI Signal Value Determination during SSL Negation Period

MOIFE	MOIFV	MOSI Signal Value during SSL Negation Period
0	0, 1	Last output value from previous transfer (The value is undefined when CPHA is 0)
1	0	Always 0
1	1	Always 1

27.4.3 System Configuration Example

(1) Master/Slave (with This LSI Acting as Master)

Figure 27.2 shows a master/slave system configuration example when this LSI is used as a master. In master/slave configuration, the SSL output of this LSI (master) is not used. The SSL input of the slave is fixed to the low level, and the slave is always maintained in a selected state. In the transfer format corresponding to the case where the CPHA bit in the control register (SPCR) is 0, there are slave devices for which the SSL signal cannot be fixed to the active level. In situations where the SSL signal cannot be fixed, the SSL output of this LSI should be connected to the SSL input of the slave device.

This LSI (master) always drives the RSPCK and MOSI. The slave always drives the MISO.

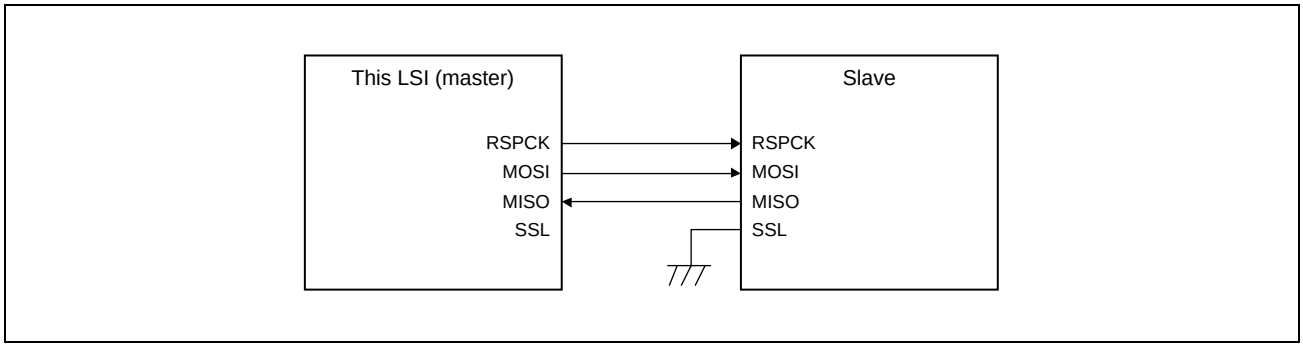


Figure 27.2 Master/Slave Configuration Example (This LSI = Master)

(2) Master/Slave (with This LSI Acting as Slave)

Figure 27.3 shows a master/slave system configuration example when this LSI is used as a slave. When this LSI is to operate as a slave, the SSL pin is used as SSL input. The master always drives the RSPCK and MOSI. This LSI (slave) always drives the MISO. When SSL is at the non-active level, the pin state is Hi-Z.

In the slave configuration in which the CPHA bit in the command register (SPCMD) is set to 1, the SSL input of this LSI (slave) is fixed to the 0 level, this LSI (slave) is always maintained in a selected state, and in this manner it is possible to execute serial transfer (**Figure 27.4**).

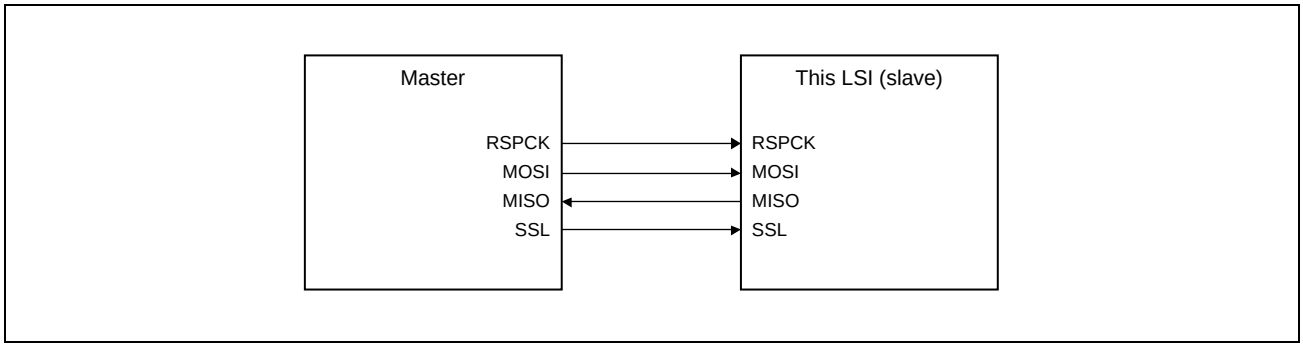


Figure 27.3 Master/Slave Configuration Example (This LSI = Slave)

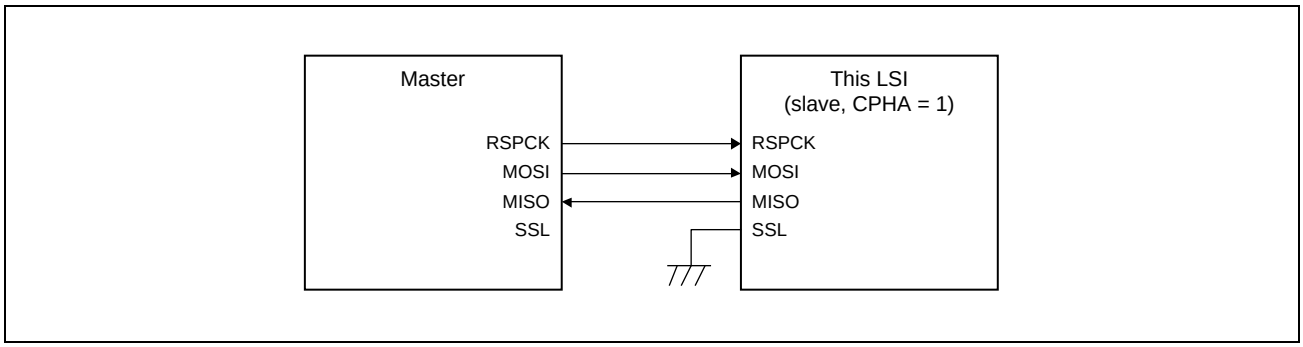


Figure 27.4 Master/Slave Configuration Example (This LSI = Slave, CPHA = 1)

(3) Master/Multi-Slave (with This LSI Acting as Slave)

Figure 27.5 shows a master/multi-slave system configuration example when this LSI is used as a slave. In the example of **Figure 27.5**, the system is comprised of a master and two LSIs (slave X and slave Y).

The RSPCK and MOSI outputs of the master are connected to the RSPCK and MOSI inputs of the LSIs (slave X and slave Y). The MISO outputs of the LSIs (slave X and slave Y) are all connected to the MISO input of the master. SSLX and SSLY outputs of the master are connected to the SSL inputs of the LSIs (slave X and slave Y), respectively.

The master always drives RSPCK, MOSI, SSLX, and SSLY. Of the LSIs (slave X and slave Y), the slave that receives low level input into the SSL0 input drives MISO.

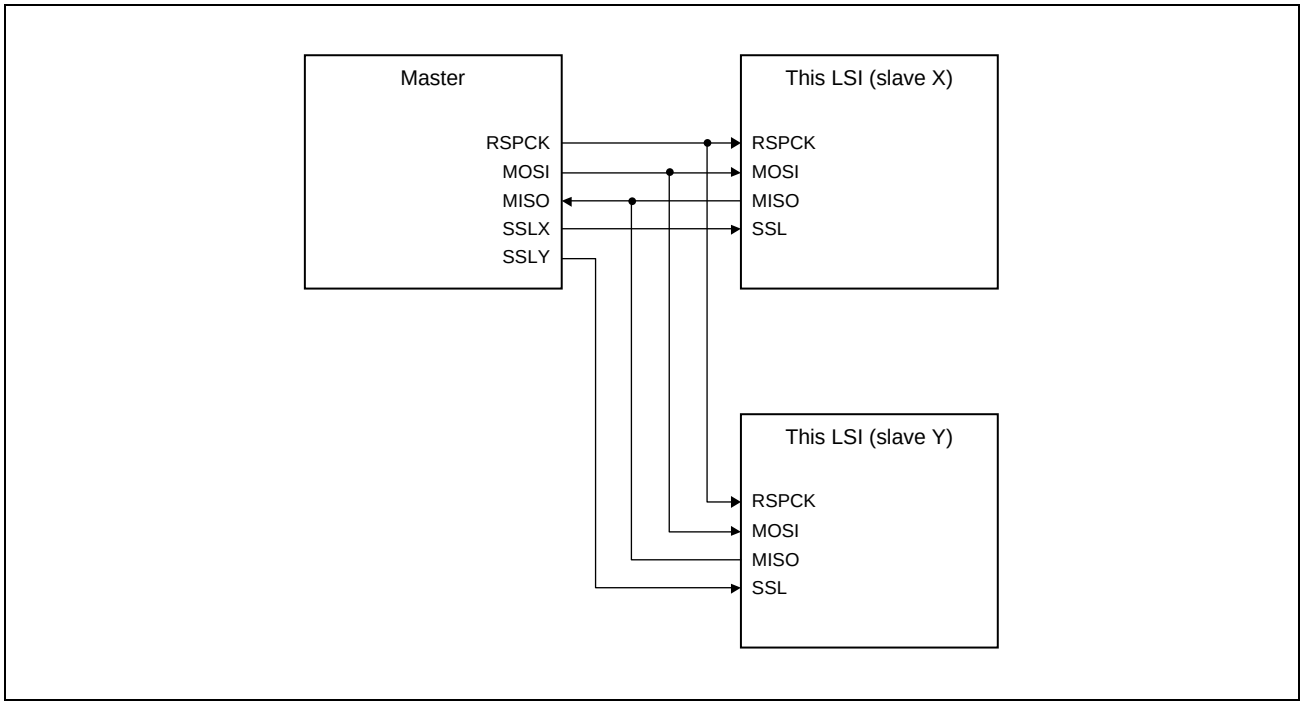


Figure 27.5 Master/Multi-Slave Configuration Example (This LSI = Slave)

27.4.4 Transfer Format

(1) CPHA = 0

Figure 27.6 shows a sample transfer format for the serial transfer of 8-bit data when the CPHA bit in the command register (SPCMD) is 0. In **Figure 27.6**, RSPCK (CPOL = 0) indicates the RSPCK signal waveform when the CPOL bit in SPCMD is 0; RSPCK (CPOL = 1) indicates the RSPCK signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which this module fetches serial transfer data into the shift register. The input/output directions of the signals depend on the settings of this module. For details, see **Section 27.4.2, Pin Control**.

When the CPHA bit is 0, the driving of valid data to the MOSI and MISO signals commences at an SSL signal assertion timing. The first RSPCK signal change timing that occurs after the SSL signal assertion becomes the first transfer data fetching timing. After this timing, data is sampled at every 1 RSPCK cycle. The change timing for MOSI and MISO signals is always 1/2 RSPCK cycle after the transfer data fetch timing. The settings in the CPOL bit do not affect the RSPCK signal operation timing; they only affect the signal polarity.

t1 denotes a period from an SSL signal assertion to RSPCK oscillation (RSPCK delay). t2 denotes a period from the cessation of RSPCK oscillation to an SSL signal negation (SSL negation delay). t3 denotes a period in which SSL signal assertion is suppressed for the next transfer after the end of serial transfer (next-access delay). t1, t2, and t3 are controlled by a master device running on the system. For a description of t1, t2, and t3 when this module is in master mode, see **Section 27.4.3(1), Master/Slave (with This LSI Acting as Master)**.

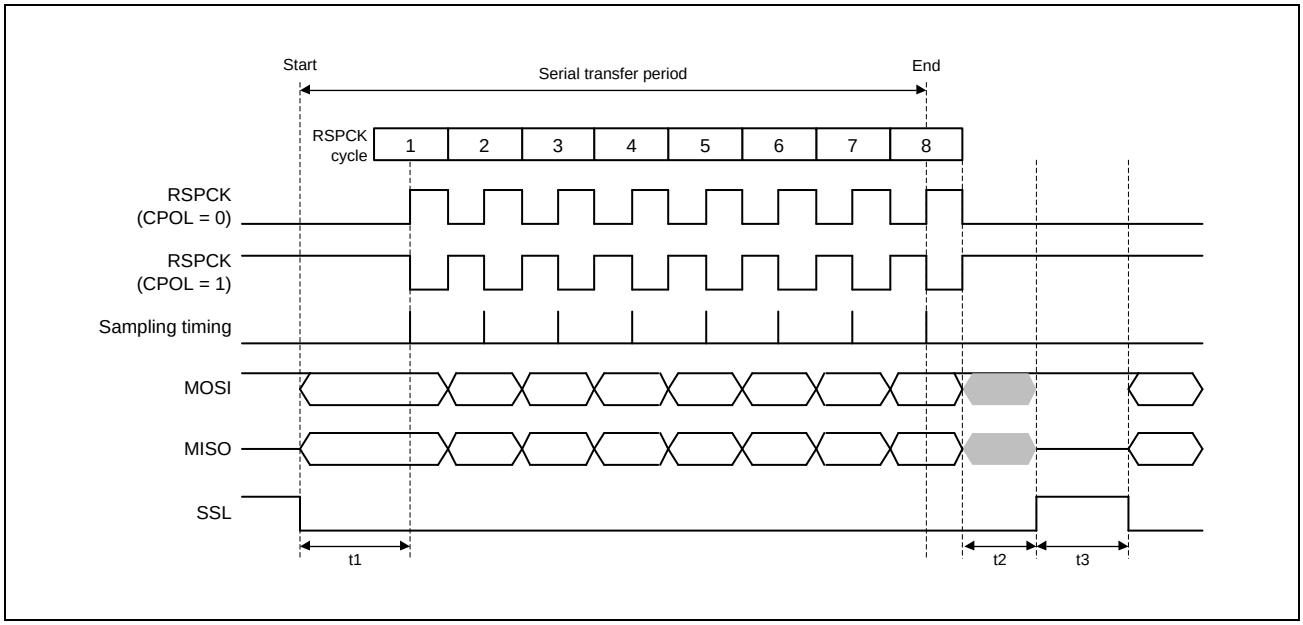


Figure 27.6 Transfer Format (CPHA = 0)

(2) CPHA = 1

Figure 27.7 shows a sample transfer format for the serial transfer of 8-bit data when the CPHA bit in the command register (SPCMD) is 1. In **Figure 27.7**, RSPCK (CPOL = 0) indicates the RSPCK signal waveform when the CPOL bit in SPCMD is 0; RSPCK (CPOL = 1) indicates the RSPCK signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which this module fetches serial transfer data into the shift register. The input/output directions of the signals depend on the modes (master or slave). For details, see **Section 27.4.2, Pin Control**.

When the CPHA bit is 1, the driving of invalid data to the MOSI and MISO signals commences at an SSL signal assertion timing. The driving of valid data to the MOSI and MISO signals commences at the first RSPCK signal change timing that occurs after the SSL signal assertion. After this timing, data is updated at every 1 RSPCK cycle. The transfer data fetch timing is always 1/2 RSPCK cycle after the data update timing. The settings in the CPOL bit do not affect the RSPCK signal operation timing; they only affect the signal polarity.

t1, t2, and t3 are the same as those in the case of CPHA = 0. For a description of t1, t2, and t3 when this module is in master mode, see **Section 27.4.3(1), Master/Slave (with This LSI Acting as Master)**.

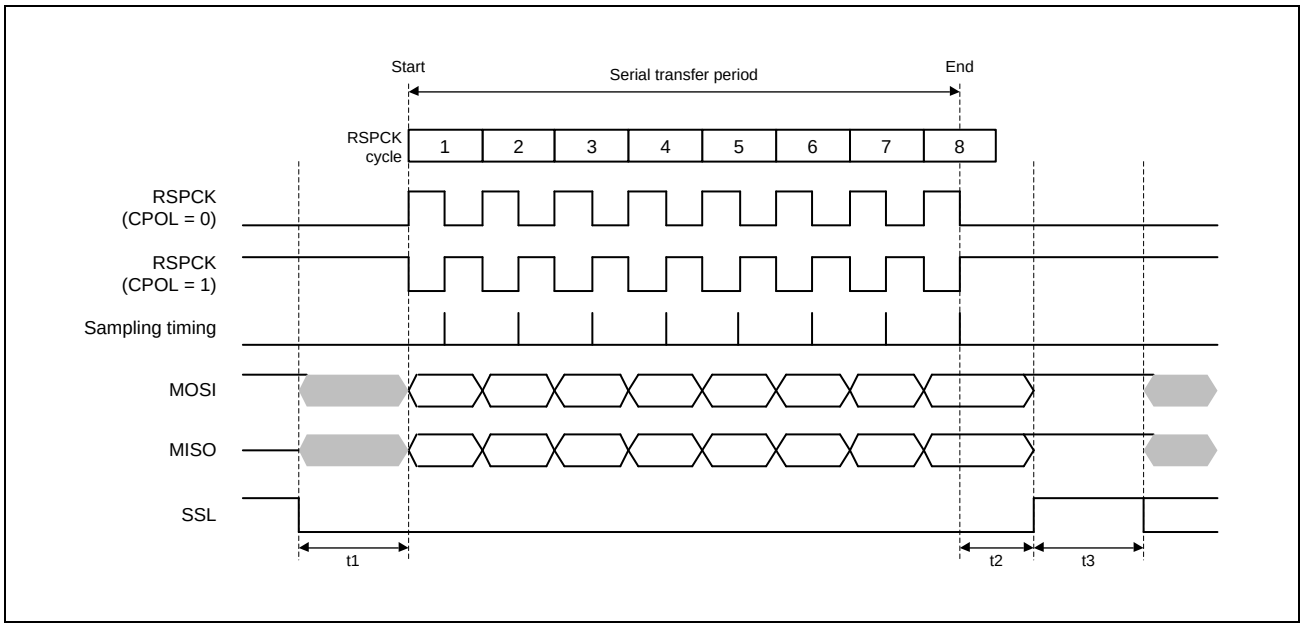


Figure 27.7 Transfer Format (CPHA = 1)

27.4.5 Data Format

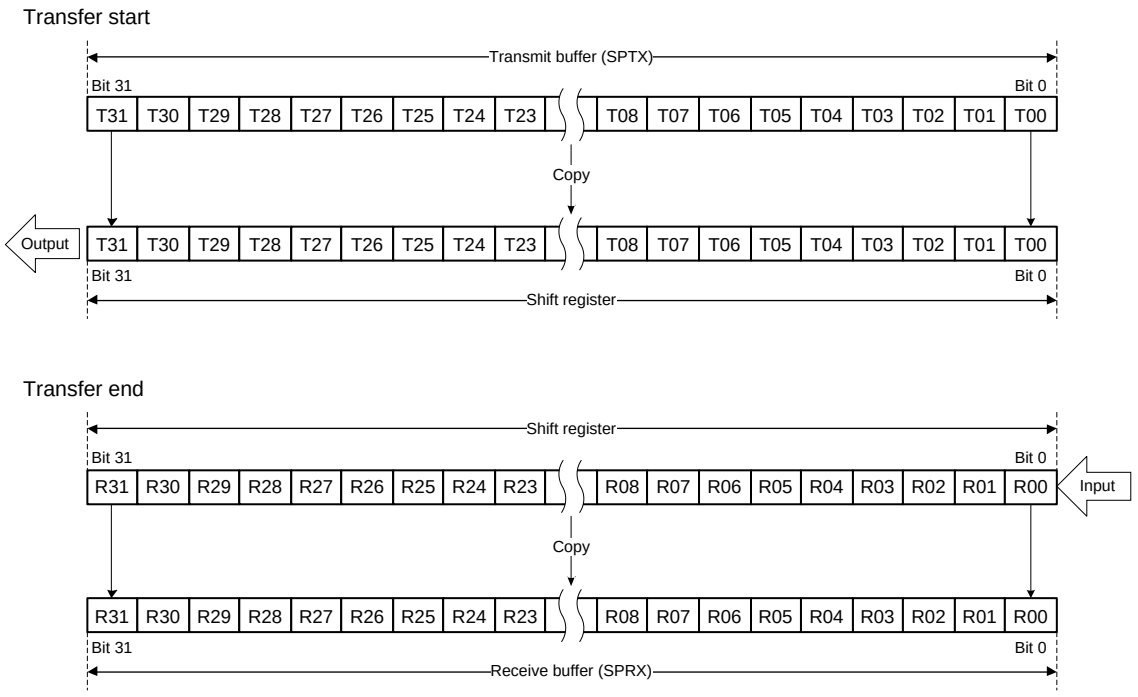
The data format depends on the settings in the command register (SPCMD). Irrespective of MSB/LSB first, this module treats the range from the LSB of the data register (SPDR) to the assigned data length as transfer data.

(1) MSB First Transfer (32-Bit Data)

Figure 27.8 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 32-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T31 to T00 to the transmit buffer of SPDR. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R31 to R00 is stored in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after the receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R31 to R00 is shifted out from the shift register.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

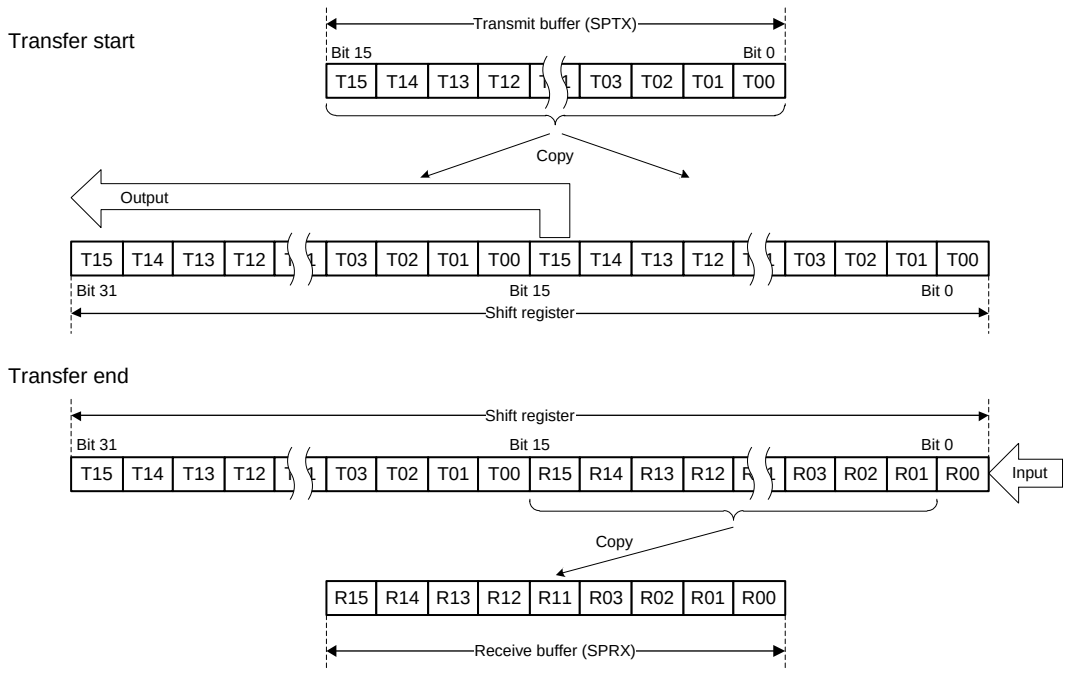
Figure 27.8 MSB First Transfer (32-Bit Data)

(2) MSB First Transfer (16-Bit Data)

Figure 27.9 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 16-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T15 to T00 to the transmit buffer. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from bit 15 in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 16 bits has passed, received data R15 to R00 is stored in bits 15 to 0 in the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 31 to 16 in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R15 to R00 is shifted out from the shift register.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

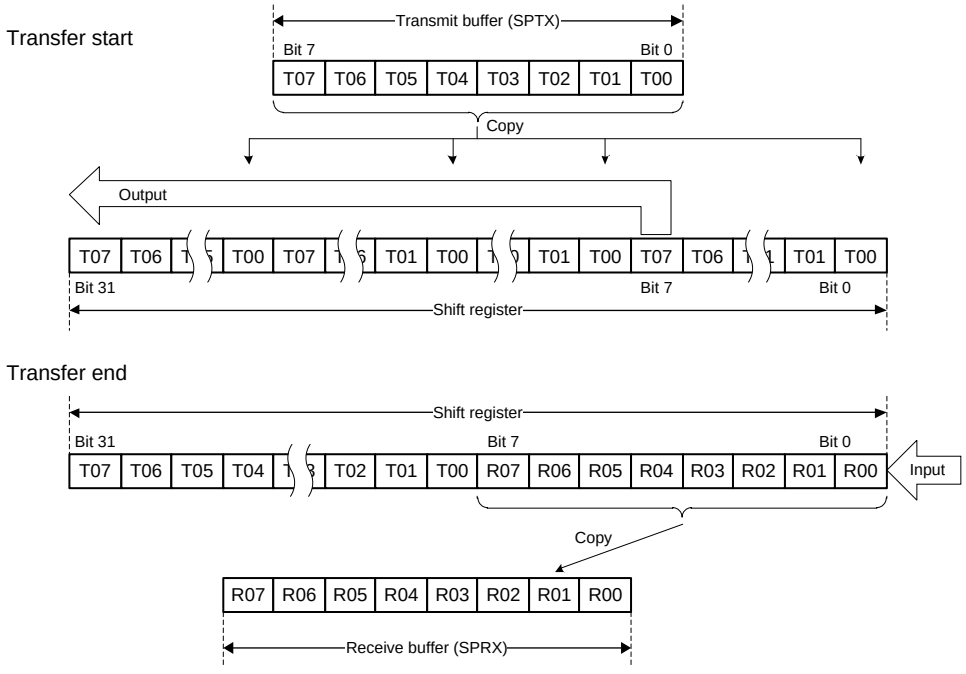
Figure 27.9 MSB First Transfer (16-Bit Data)

(3) MSB First Transfer (8-Bit Data)

Figure 27.10 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs an 8-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T07 to T00 to the transmit buffer. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from bit 7 in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 8 bits has passed, received data R07 to R00 is stored in bits 7 to 0 in the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 31 to 8 in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary area in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R07 to R00 is shifted out from the shift register.



Note: Output = MOSI (master)/MISO (slave), input = MOSI (master)/MISO (slave)

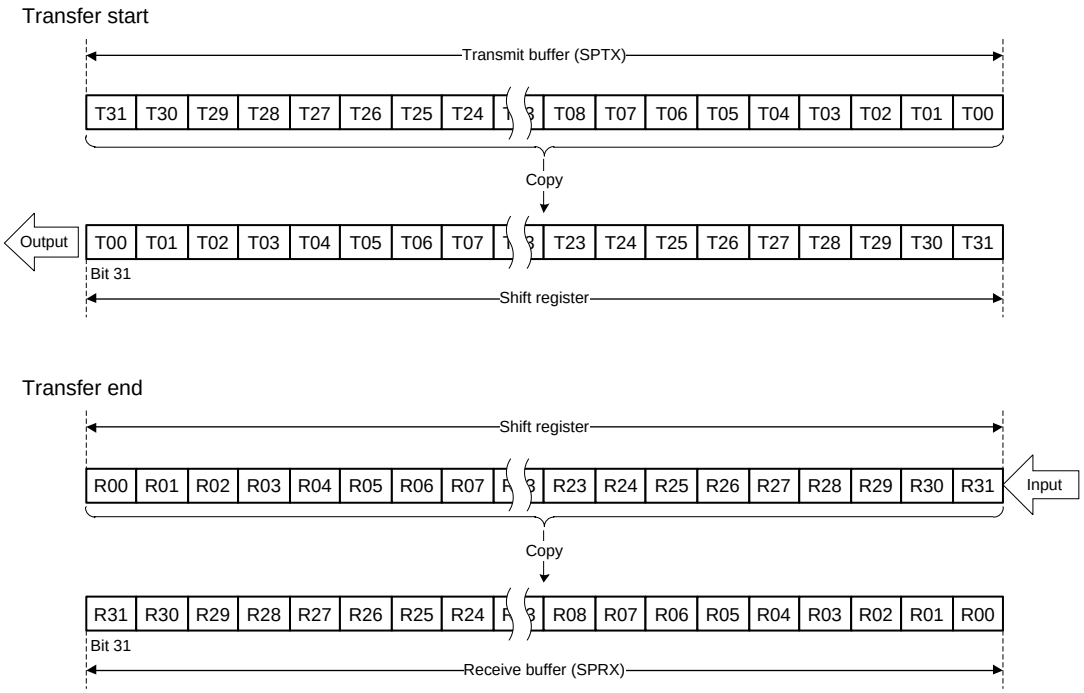
Figure 27.10 MSB First Transfer (8-Bit Data)

(4) LSB First Transfer (32-Bit Data)

Figure 27.11 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 32-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T31 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R00 to R31 is stored in the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of the SPDR, received data R00 to R31 is shifted out from the shift register.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

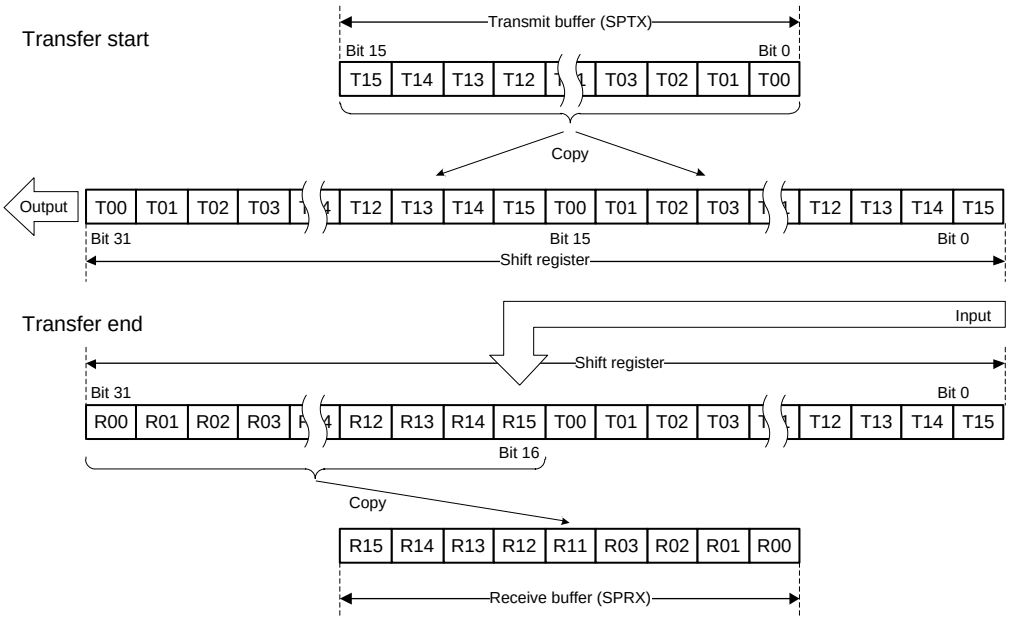
Figure 27.11 LSB First Transfer (32-Bit Data)

(5) LSB First Transfer (16-Bit Data)

Figure 27.12 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 16-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T15 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from bit 16 in the shift register. When the RSPCK cycle required for the serial transfer of 16 bits has passed, received data R00 to R15 is stored in bits 31 to 16 in the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 15 to 0 in the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of SPDR, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of SPDR, received data R00 to R15 is shifted out from the shift register.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

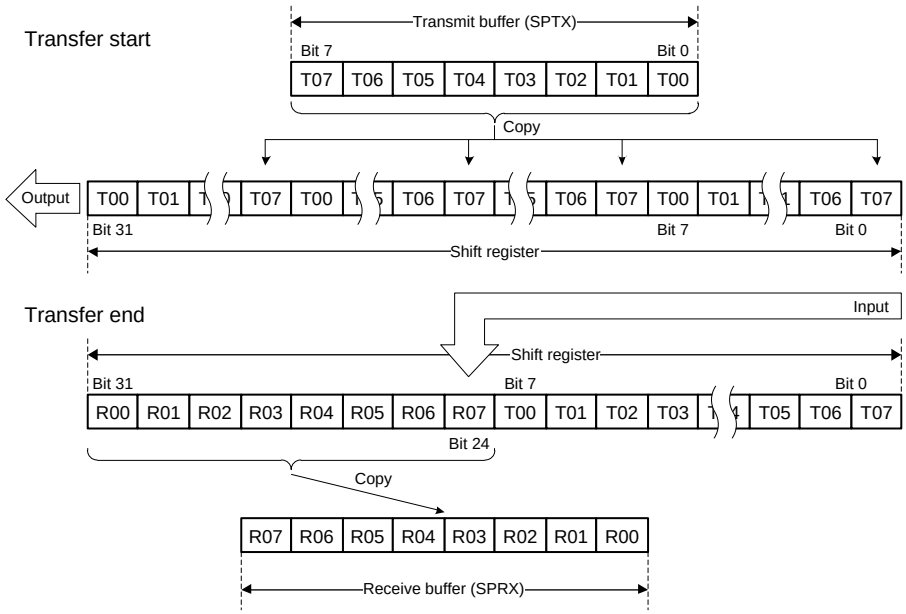
Figure 27.12 LSB First Transfer (16-Bit Data)

(6) LSB First Transfer (8-Bit Data)

Figure 27.13 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs an 8-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T07 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from bit 24 in the shift register. When the RSPCK cycle required for the serial transfer of 8 bits has passed, received data R00 to R07 is stored in bits 31 to 24 of the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 23 to 0 in the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of SPDR, and empties the shift register. If the receive buffer does not have a space for the receive data length after the receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of SPDR, received data R00 to R07 is shifted out from the shift register.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 27.13 LSB First Transfer (8-Bit Data)

27.4.6 Error Detection

In the normal serial transfer, the data written to the transmit buffer of the data register (SPDR) is serially transmitted, and the serially received data can be read from the receive buffer of SPDR. If access is made to SPDR, depending on the status of the transmit buffer/receive buffer or the status at the beginning or end of serial transfer, in some cases non-normal transfers can be executed.

If a non-normal transfer operation occurs, this module detects the event as an overrun error or a mode fault error. **Table 27.7** shows the relationship between non-normal transfer operations and the error detection function.

Table 27.7 Relationship between Non-Normal Transfer Operations and Error Detection Function

	Occurrence Condition	Operation	Error Detection
A	SPDR is written when the transmit buffer is full.	Missing write data.	None
B	Serial transfer is started in slave mode when transmit data is still not loaded on the shift register.	Data received in previous serial transfer is serially transmitted.	None
C	SPDR is read when the receive buffer is empty.	The output data is undefined.	None
D	Serial transfer terminates when the receive buffer is full.	Missing serial receive data.	Overrun error (only in slave mode)
E	The SSL input signal is negated during serial transfer in slave mode.	Serial transfer suspended. Missing send/receive data. Operation disabled.	Mode fault error

On operation A shown in **Table 27.7**, this module does not detect an error. Whether SPDR can be written to or not can be checked using the T[3:0] bits in the buffer data count setting register (SPBFDR).

Likewise, this module does not detect an error on operation B. In a serial transfer that was started before the shift register was updated, this module sends the data that was received in the previous serial transfer, and does not treat the operation indicated in B as an error. Note that the received data from the previous serial transfer is retained in the receive buffer of SPDR, thus it can be correctly read.

Similarly, this module does not detect an error on operation C. To prevent extraneous data from being read, the number of receive data units stored in the receive buffer should be read from the R[5:0] bits in the buffer data count setting register (SPBFDR).

An overrun error shown in D is described in **Section 27.4.6(1), Overrun Error**. A mode fault error shown in E is described in **Section 27.4.6(2), Mode Fault Error**.

(1) Overrun Error

If serial transfer ends when the receive buffer of the data register (SPDR) is full, this module detects an overrun error, and sets the OVRF bit in SPSR to 1. When the OVRF bit is 1, this module does not copy data from the shift register to the receive buffer so that the data prior to the occurrence of the error is retained in the receive buffer. To reset the OVRF bit in SPSR to 0, either perform a power-on reset, or write a 0 to the OVRF bit after SPSR has been read with the OVRF bit set to 1.

Figure 27.14 shows an example of operation of the SPRF and OVRF bits in SPSR. The SPSR and SPDR accesses shown in **Figure 27.14** indicate the condition of accesses to SPSR and SPDR, respectively, where I denotes an idle cycle, W a write cycle, and R a read cycle. In the example of **Figure 27.14**, this module performs an 8-bit serial transfer in which the CPHA bit in the command register (SPCMD) is 1, and CPOL is 0. The numbers given under the RSPCK waveform represent the number of RSPCK cycles (i.e., the number of transferred bits).

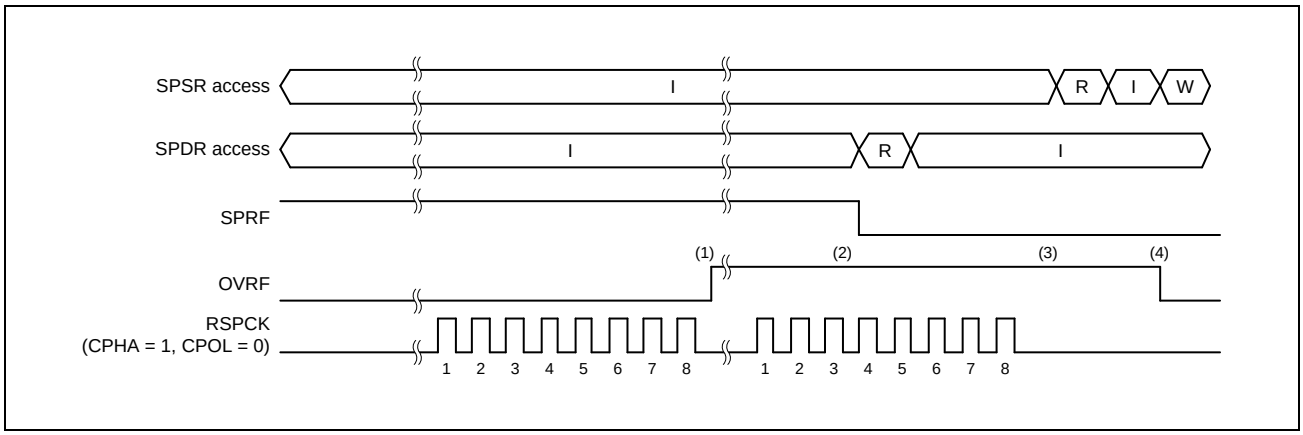


Figure 27.14 SPRF and OVRF Bit Operation Example

The operation of the flags at the timing shown in steps (1) to (4) in the figure is described below.

1. If a serial transfer terminates when the receive buffer does not have a space for the receive data length, this module detects an overrun error, and sets the OVRF bit to 1. This module does not copy the data in the shift register to the receive buffer.
2. The OVFR bit is not cleared even when SPDR is read and thus the number of data bytes in the receive buffer becomes less than the number of the receive buffer data triggering number specified by the RXTRG bits.
3. If the serial transfer terminates in an overrun error state, this module determines that the shift register is empty; in this manner, data transfer is enabled from the transmit buffer to the shift register.
4. If 0 is written to the OVRF bit after SPSR is read with OVRF = 1, this module clears the OVRF bit.

The occurrence of an overrun can be checked either by reading SPSR or by using an error interrupt and reading SPSR. When using an error interrupt, set the SPEIE bit in the control register (SPCR) to 1. When executing a serial transfer without using an error interrupt, measures should be taken to ensure the early detection of overrun errors, such as reading SPSR immediately after SPDR is read.

The OVRF bit is cleared to 0 under the following conditions:

- After SPSR is read in a condition in which the OVRF bit is set to 1, 0 is written to the OVRF bit.
- Power-on reset

NOTE

When the receive buffer has area enough to store receive data with an overrun error, this module receives receive data.

(2) Mode Fault Error

When the MSTR bit is 0, this module operates in slave mode. This module detects a mode fault error if the SSL input signal is negated during the serial transfer period (from the time the driving of valid data is started to the time the final valid data is fetched) when the MODFEN bit is 1 in slave mode.

Upon detecting a mode fault error, this module stops driving of the output signals and clears the SPE bit in SPCR to 0. When the SPE bit is cleared to 0, the function of this module is disabled, and this module stops driving external signals. For details of disabling the function of this module by clearing the SPE bit to 0, see **Section 27.4.7, Initialization**.

The occurrence of a mode fault error can be checked either by reading SPSR or by using an error interrupt and reading SPSR. When using an error interrupt, set the SPEIE bit in the control register (SPCR) to 1. To detect a mode fault error without using an error interrupt, it is necessary to poll SPSR.

When the MODF bit is 1, writing 1 to the SPE bit is ignored. To enable the function of this module after the detection of a mode fault error, the MODF bit must be set to 0. The MODF bit is cleared to 0 under the following conditions:

- After SPSR is read in a condition where the MODF bit has turned 1, 0 is written to the MODF bit.
- Power-on reset

27.4.7 Initialization

If 0 is written to the SPE bit in the control register (SPCR) or this module clears the SPE bit to 0 because of the detection of a mode fault error, this module disables the module function, and initializes a part of the module function. When a power-on reset is generated, this module initializes all of the module function. An explanation follows of initialization by the clearing of the SPE bit.

(1) Initialization by Clearing SPE Bit

When the SPE bit in SPCR is cleared, this module performs the following initialization:

- Suspending any serial transfer that is being executed
- Stopping the driving of output signals (Hi-Z) in slave mode
- Initializing the internal state
- Initializing the TEND bit in SPSR

Initialization by the clearing of the SPE bit does not initialize the control bits of this module. For this reason, this module can be started in the same transfer mode as prior to the initialization if the SPE bit is re-set to 1.

27.4.8 SPI Operation

(1) Multi-Master Mode Operation

This section explains the operation in multi-master mode.

(a) Starting Serial Transfer

A serial transfer is started when transmit data is copied from the transmit buffer to the shift register, the shift register becomes full, and the receive buffer has a space for the receive data length. If transmit data has already been written to the shift register, data is not copied from the transmit buffer to the shift register.

For details of the transfer format, see **Section 27.4.4, Transfer Format**.

(b) Terminating Serial Transfer

Irrespective of the CPHA bit in the command register (SPCMD), this module terminates the serial transfer after transmitting an RSPCK edge corresponding to the final sampling timing. After the serial transfer is completed, receive data is copied from the shift register to the receive buffer. If the receive buffer does not have a space for the receive data length after receive data is copied from the shift register to the receive buffer, another serial transfer will not be performed. In order to perform another serial transfer, data for the receive data length should be read from the receive buffer to secure the space for the receive data.

It should be noted that the final sampling timing varies depending on the bit length of transfer data. In master mode, the data length depends on the settings in bits SPB3 to SPB0 in SPCMD. For details on the transfer format, see **Section 27.4.4, Transfer Format**.

(c) Sequence Control

The transfer format that is employed in master mode is determined by the sequence control register (SPSCR), command registers 0 to 3 (SPCMD0 to SPCMD3), the bit rate register (SPBR), the clock delay register (SPCKD), the slave select negation delay register (SSLND), and the next-access delay register (SPND).

SPSCR is a register used to determine the sequence configuration for serial transfers that are executed by this module in master mode. The following items are set in command registers SPCMD0 to SPCMD3: SSL output signal value, MSB/LSB first, data length, some of the bit rate settings, RSPCK polarity/phase, whether SPCKD is to be referenced, whether SSLND is to be referenced, and whether SPND is to be referenced. SPBR holds some of the bit rate settings; SPCKD, a clock delay value; SSLND, an SSL negation delay; and SPND, a next-access delay value.

According to the sequence length that is assigned to SPSCR, this module makes up a sequence comprised of a part or all of SPCMD0 to SPCMD3. This module contains a pointer to the SPCMD that makes up the sequence. The value of this pointer can be checked by reading bits SPCP1 and SPCP0 in the sequence status register (SPSSR). When the SPE bit in the control register (SPCR) is set to 1 and the function of this module is enabled, this module loads the pointer to the commands in SPCMD0, and incorporates the SPCMD0 settings into the transfer format at the beginning of serial transfer. This module increments the pointer each time the next-access delay period for a data transfer ends. Upon completion of the serial transfer that corresponds to the final command comprising the sequence, this module sets the pointer in SPCMD0, and in this manner the sequence is executed repeatedly.

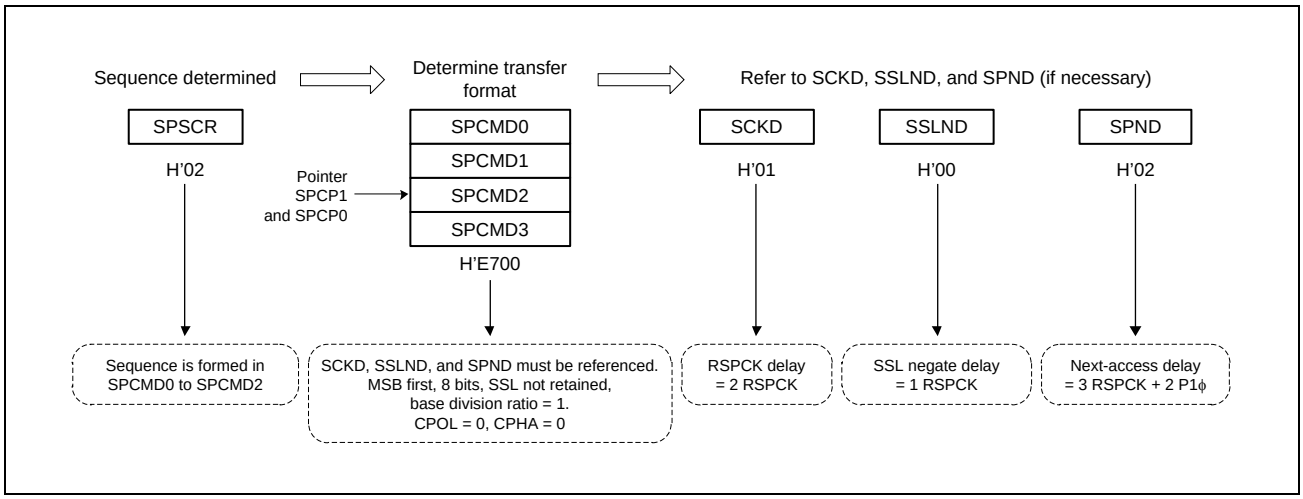


Figure 27.15 Determination Procedure of Serial Transfer Mode in Master Mode

(d) Burst Transfer

If the SSLKP bit in the command register (SPCMD) that this module references during the current serial transfer is 1, this module keeps the SSL signal level during the serial transfer until the beginning of the SSL signal assertion for the next serial transfer. If the SSL signal level for the next serial transfer is the same as the SSL signal level for the current serial transfer, this module can execute continuous serial transfers while keeping the SSL signal assertion status (burst transfer).

Figure 27.16 shows an example of an SSL signal operation for the case where a burst transfer is implemented using SPCMD0 and SPCMD1 settings. The text below explains operations (1) to (7) as depicted in **Figure 27.16**. It should be noted that the polarity of the SSL output signal depends on the settings in the slave select polarity register (SSLP).

1. Based on SPCMD0, this module asserts the SSL signal and inserts RSPCK delays.
2. Serial transfers are executed according to SPCMD0.
3. SSL negation delays are inserted.
4. Because the SSLKP bit in SPCMD0 is 1, this module keeps the SSL signal value on SPCMD0. This period is sustained, at the shortest, for a period equal to the next-access delay of SPCMD0. If the shift register is empty after the passage of a minimum period, this period is sustained until such time as the transmit data is stored in the shift register for another transfer.
5. Based on SPCMD1, this module asserts the SSL signal and inserts RSPCK delays.
6. Serial transfers are executed according to SPCMD1.
7. Because the SSLKP bit in SPCMD1 is 0, this module negates the SSL signal. In addition, a next-access delay is inserted according to SPCMD1.

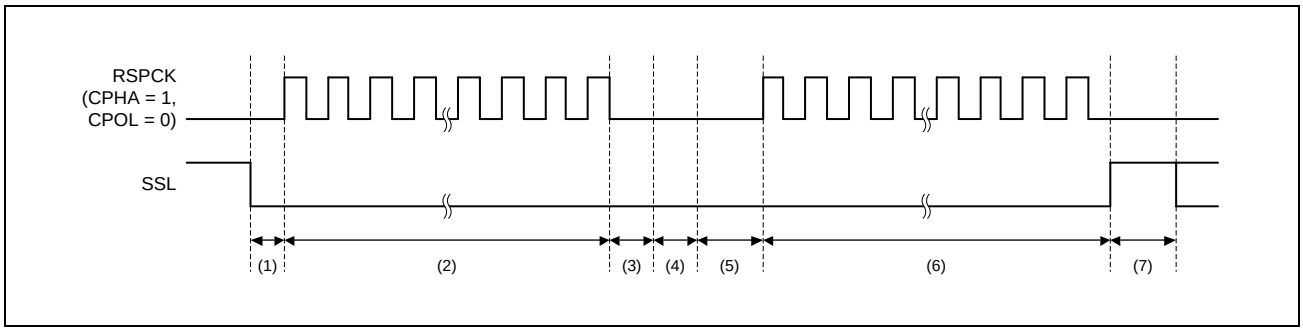


Figure 27.16 Example of Burst Transfer Operation using SSLKP Bit

If the SSL signal settings in the SPCMD in which 1 is assigned to the SSLKP bit are different from the SSL signal output settings in the SPCMD to be used in the next transfer, this module switches the SSL signal status to SSL signal assertion ((5) in **Figure 27.16**) corresponding to the command for the next transfer. Notice that if such an SSL signal switching occurs, the slaves that drive the MISO signal compete, and the possibility arises of the collision of signal levels.

This module in master mode references within the module the SSL signal operation for the case where the SSLKP bit is not used. Even when the CPHA bit in SPCMD is 0, this module can accurately start serial transfers by asserting the SSL signal for the next transfer. For this reason, burst transfers in master mode can be executed irrespective of CPHA bit settings (see **Section 27.4.8(2), Slave Mode Operation**).

(e) RSPCK Delay (t1)

The RSPCK delay value in master mode depends on SCKDEN bit settings in the command register (SPCMD) and on clock delay register (SPCKD) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines an RSPCK delay value during serial transfer by using the SCKDEN bit in the selected SPCMD and SPCKD, as shown in **Table 27.8**. For a definition of RSPCK delay, see **Section 27.4.4, Transfer Format**.

Table 27.8 Relationship among SCKDEN and SPCKD Settings and RSPCK Delay Values

SCKDEN	SPCKD	RSPCK Delay Value
0	000 to 111	1 RSPCK
1	000	1 RSPCK
	001	2 RSPCK
	010	3 RSPCK
	011	4 RSPCK
	100	5 RSPCK
	101	6 RSPCK
	110	7 RSPCK
	111	8 RSPCK

(f) SSL Negation Delay (t2)

The SSL negation delay value in master mode depends on SLNDEN bit settings in the command register (SPCMD) and on SSL negation delay register (SSLND) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines an SSL negation delay value during serial transfer by using the SLNDEN bit in the selected SPCMD and SSLND, as shown in **Table 27.9**. For a definition of SSL negation delay, see **Section 27.4.4, Transfer Format**.

Table 27.9 Relationship among SLNDEN and SSLND Settings and SSL Negation Delay Values

SLNDEN	SSLND	SSL Negation Delay Value
0	000 to 111	1 RSPCK
1	000	1 RSPCK
	001	2 RSPCK
	010	3 RSPCK
	011	4 RSPCK
	100	5 RSPCK
	101	6 RSPCK
	110	7 RSPCK
	111	8 RSPCK

(g) Next-Access Delay (t3)

The next-access delay value in master mode depends on SPNDEN bit settings in the command register (SPCMD) and on next-access delay register (SPND) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines a next-access delay value during serial transfer by using the SPNDEN bit in the selected SPCMD and SPND, as shown in **Table 27.10**. For a definition of next-access delay, see **Section 27.4.4, Transfer Format**.

Table 27.10 Relationship among SPNDEN and SPND Settings and Next-Access Delay Values

SPNDEN	SPND	Next-Access Delay Value
0	000 to 111	1 RSPCK + 2 P0ϕ
1	000	1 RSPCK + 2 P0ϕ
	001	2 RSPCK + 2 P0ϕ
	010	3 RSPCK + 2 P0ϕ
	011	4 RSPCK + 2 P0ϕ
	100	5 RSPCK + 2 P0ϕ
	101	6 RSPCK + 2 P0ϕ
	110	7 RSPCK + 2 P0ϕ
	111	8 RSPCK + 2 P0ϕ

(h) Initialization Flowchart

Figure 27.17 is a flowchart illustrating an example of initialization in SPI operation when this module is used in master mode. For a description of how to set up the interrupt controller, direct memory access controller, and input/output ports, see the descriptions given in the individual blocks.

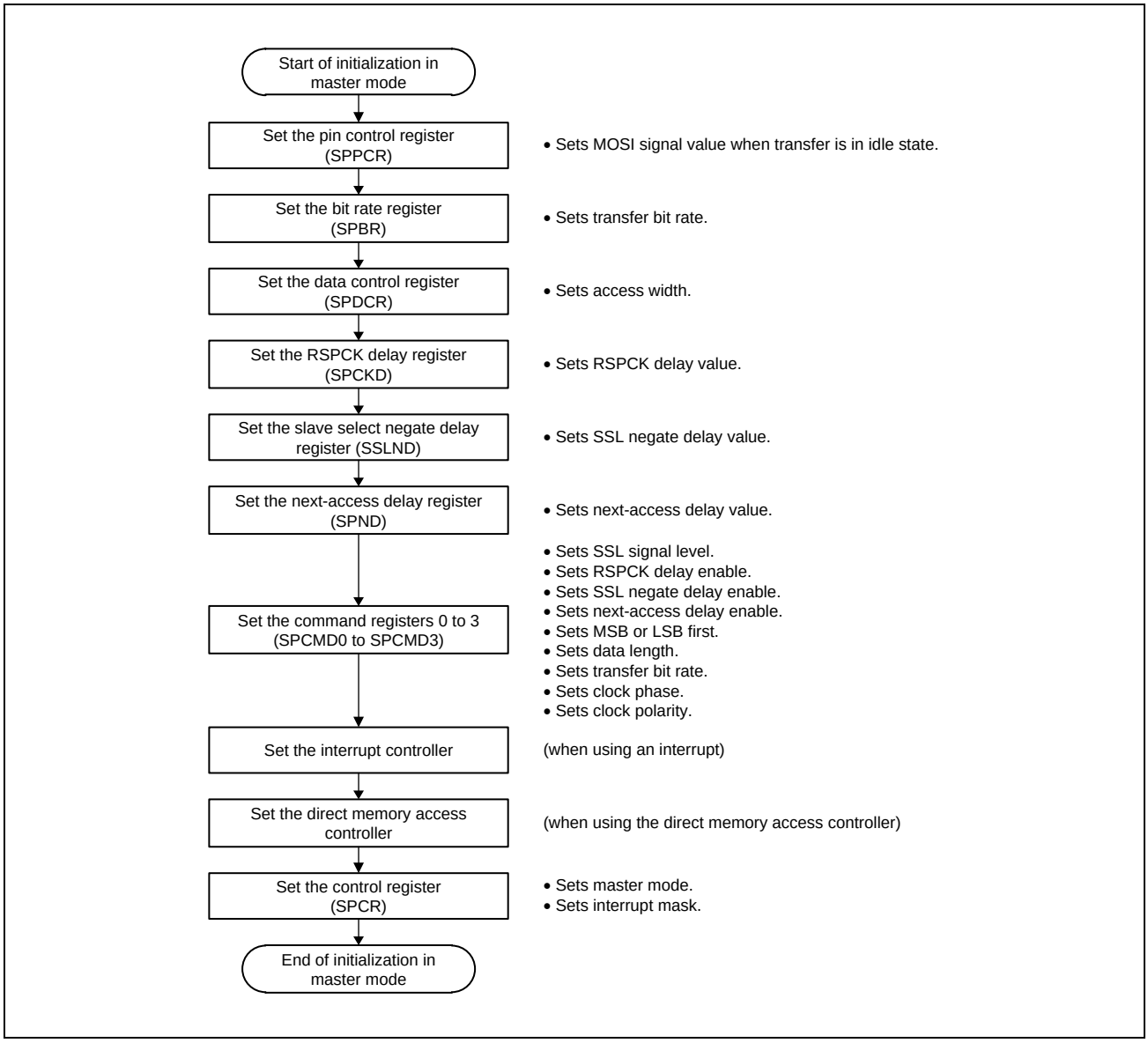


Figure 27.17 Example of Initialization Flowchart in Master Mode

(i) Transfer Operation Flowchart

Figure 27.18 is a flowchart illustrating a transfer in SPI operation when this module is used in master mode.

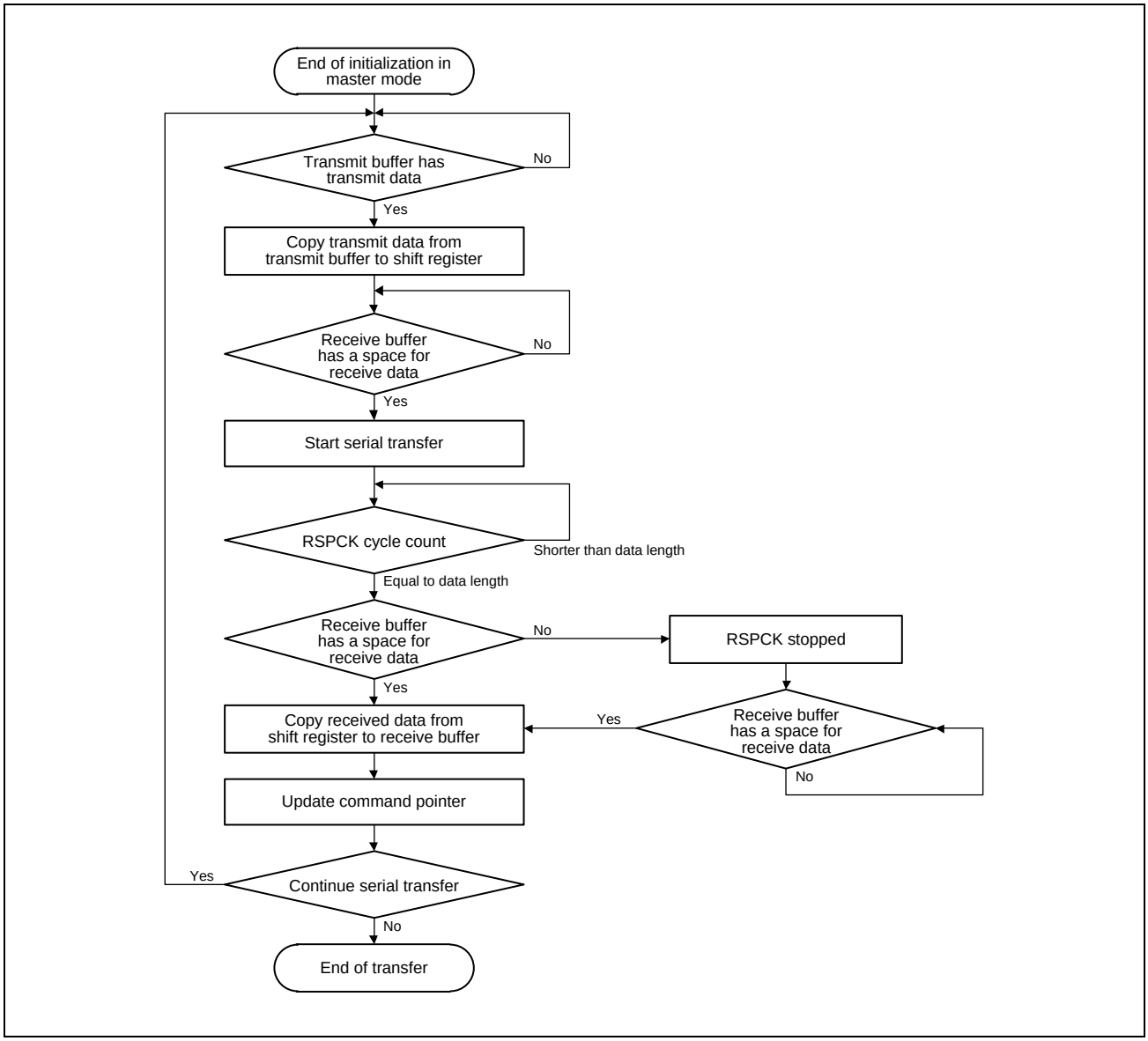


Figure 27.18 Transfer Operation Flowchart in Master Mode

(2) Slave Mode Operation

(a) Starting Serial Transfer

If this module detects an SSL input signal assertion when the CPHA bit in the command register 0 (SPCMD0) is 0, this module is required to start driving valid data to the MISO output signal. For this reason, when the CPHA bit is 0, the asserting of the SSL input signal triggers the start of a serial transfer.

If this module detects the first RSPCK edge in an SSL signal asserted condition when the CPHA bit is 1, this module is required to start driving valid data to the MISO output signal. For this reason, when the CPHA bit is 1, the first RSPCK edge in an SSL signal asserted condition triggers the start of a serial transfer.

When detecting the start of a serial transfer in a condition in which the shift register is empty, this module changes the status of the shift register to “full”, so that data cannot be copied from the transmit buffer to the shift register when serial transfer is in progress. If the shift register was full before the serial transfer started, this module leaves the status of the shift register intact, in the full state.

Irrespective of CPHA bit settings, this module starts driving MISO output signals at the SSL signal assertion timing. Whether the data output from this module is valid or invalid differs depending on CPHA bit settings.

For details on the transfer format, see **Section 27.4.4, Transfer Format**. The polarity of the SSL input signal depends on the setting of the SSL0P bit in the slave select polarity register (SSLP).

(b) Terminating Serial Transfer

Irrespective of the CPHA bit in the command register 0 (SPCMD0), this module terminates the serial transfer after detecting an RSPCK edge corresponding to the final sampling timing. When the receive buffer has an enough space for receive data, this module copies received data from the shift register to the receive buffer of the data register (SPDR) upon termination of the serial transfer. Irrespective of the value of the SPRF bit, this module changes the status of the shift register to “empty” upon termination of the serial transfer. If this module detects an SSL input signal negation from the beginning of serial transfer to the end of serial transfer, a mode fault error occurs (see **Section 27.4.6, Error Detection**).

The final sampling timing changes depending on the bit length of the transfer data. In slave mode, the data length depends on the settings in bits SPB3 to SPB0 bits in SPCMD0. The polarity of the SSL input signal depends on the setting in the SSL0P bit in the slave select polarity register (SSLP). For details on the transfer format, see **Section 27.4.4, Transfer Format**.

(c) Notes on Slave Operations

If the CPHA bit in the command register 0(SPCMD0) is 0, this module starts serial transfers when it detects the assertion edge for an SSL input signal. In the type of configuration shown in **Figure 27.4** as an example, if this module is used in single-slave mode, the SSL signal is always fixed at active state. Therefore, when the CPHA bit is set to 0, this module cannot correctly start a serial transfer. To correctly execute send/receive operation in a configuration in which the SSL input signal is fixed at active state, the CPHA bit should be set to 1. When it is necessary to set the CPHA bit to 0, the SSL input signal should not be fixed.

(d) Burst Transfer

If the CPHA bit in the command register 0 (SPCMD0) is 1, continuous serial transfer (burst transfer) can be executed while retaining the assertion state for the SSL input signal. If the CPHA bit is 1, the period from the first RSPCK edge to the sampling timing for the reception of the final bit in an SSL signal active state corresponds to a serial transfer period. Even when the SSL input signal remains at the active level, this module can accommodate burst transfers because it can detect the start of access.

If the CPHA bit is 0, for the reason given in **Section 27.4.8(2)(c), Notes on Slave Operations**, second and subsequent serial transfers during the burst transfer cannot be executed correctly.

(e) Initialization Flowchart

Figure 27.19 is a flowchart illustrating an example of initialization in SPI operation when this module is used in slave mode. For a description of how to set up the interrupt controller, direct memory access controller, and input/output ports, see the descriptions given in the individual blocks.

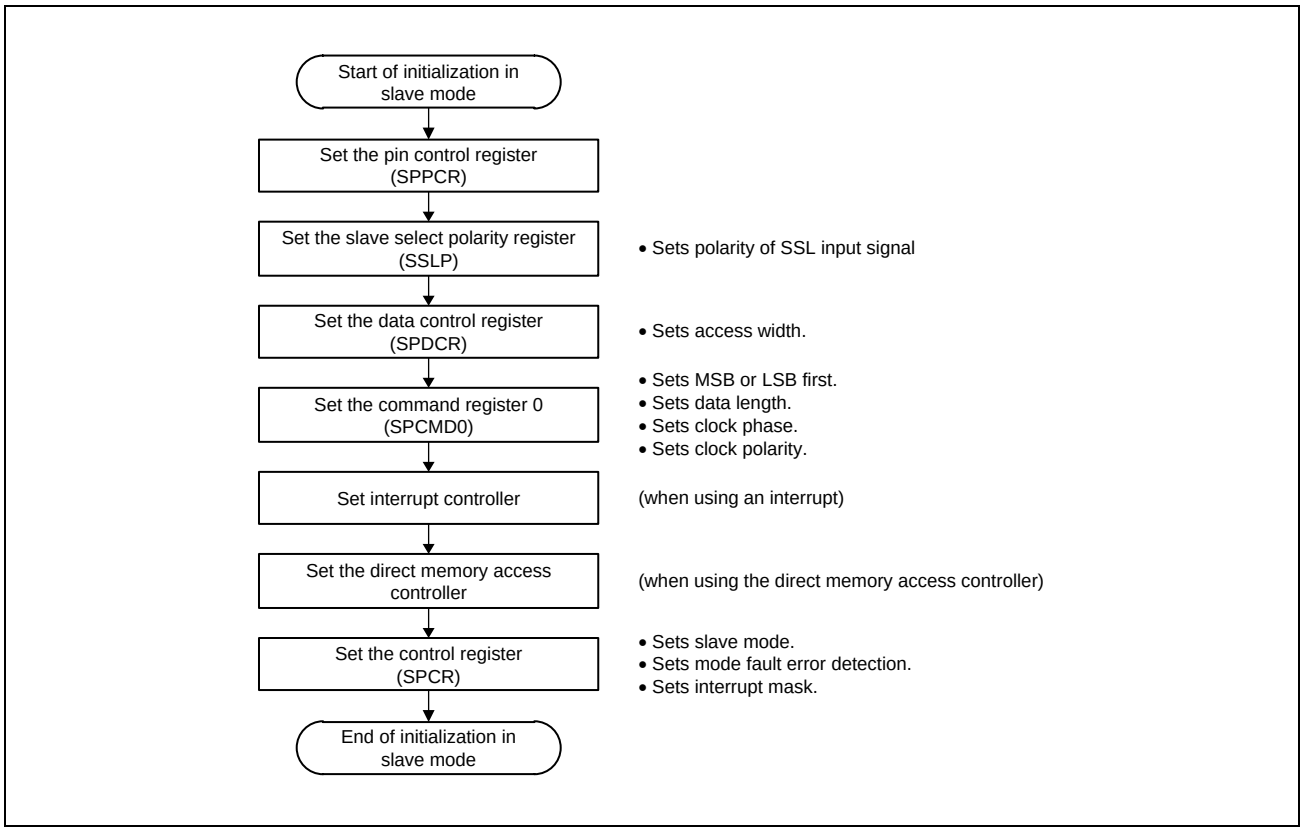


Figure 27.19 Example of Initialization Flowchart in Slave Mode

(f) Transfer Operation Flowchart (CPHA = 0)

Figure 27.20 is a flowchart illustrating a transfer in SPI operation when this module is used in slave mode with the CPHA bit in the command register 0 (SPCMD0) set to 0.

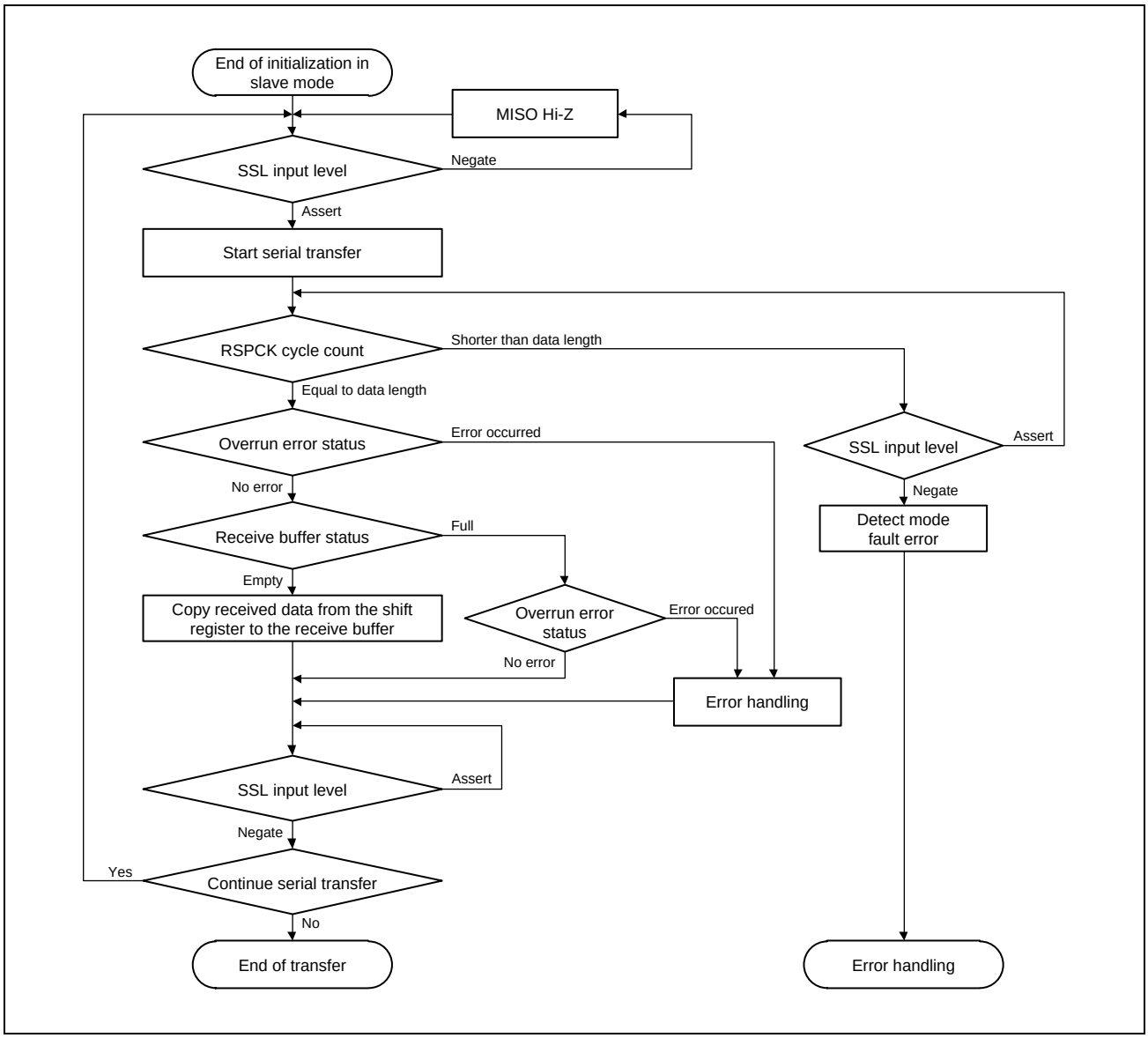


Figure 27.20 Transfer Operation Flowchart in Slave (CPHA = 0)

(g) Transfer Operation Flowchart (CPHA = 1)

Figure 27.21 is a flowchart illustrating a transfer in SPI operation when this module is used in slave mode with the CPHA bit in the command register 0 (SPCMD0) and the MODFEN bit in the control register (SPCR) set to 1, respectively. The subsequent operation is not guaranteed when the serial transfer is started with the MODFEN bit set to 0 and the SSL input level is negated with the number of RSPCK cycles shorter than the data length.

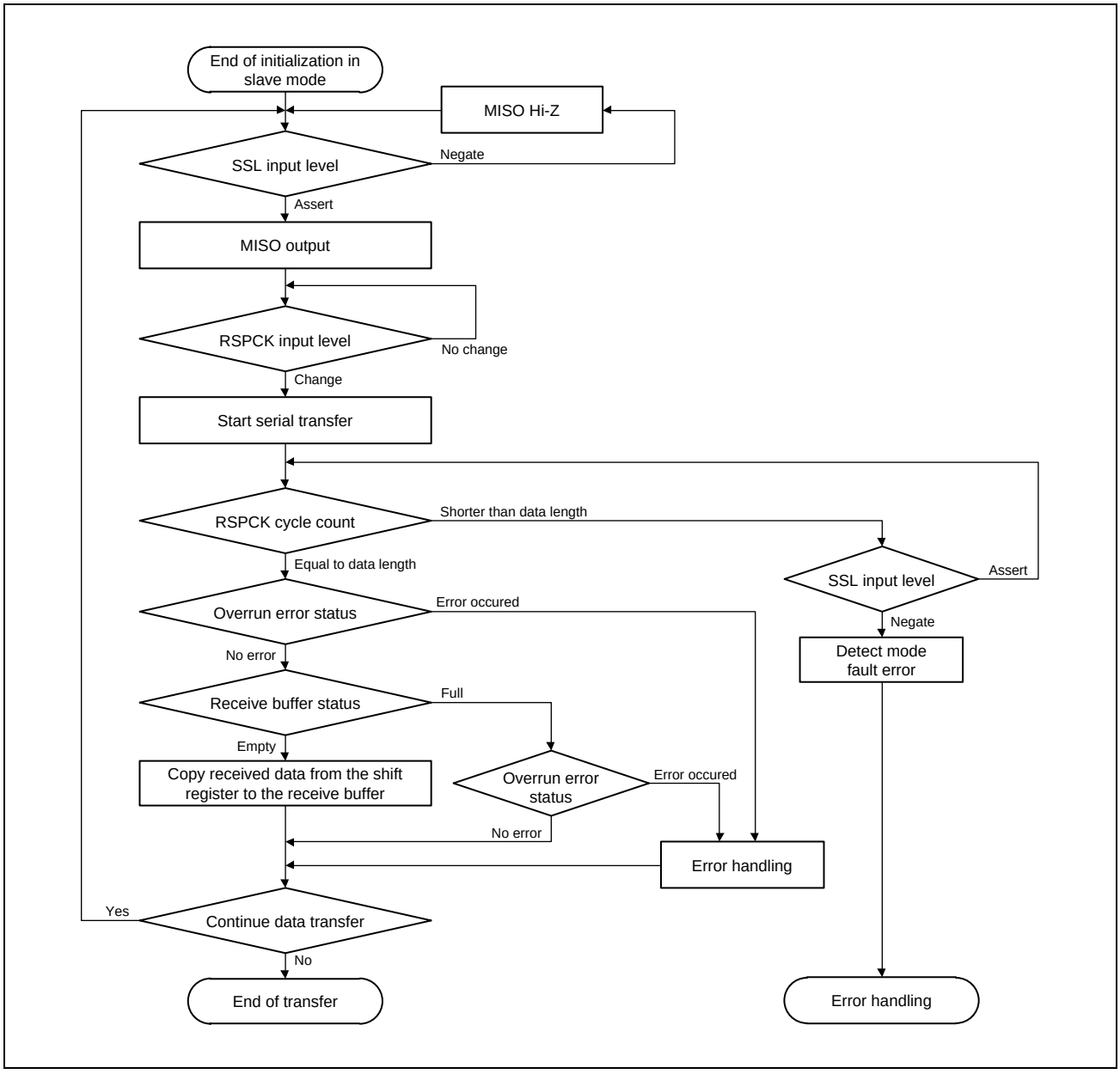


Figure 27.21 Transfer Operation Flowchart in Slave Mode (CPHA = 1)

27.4.9 Error Handling

Figure 27.22 and **Figure 27.23** show the error handling. The following error handling is used to return from the error state after an error in master or slave mode.

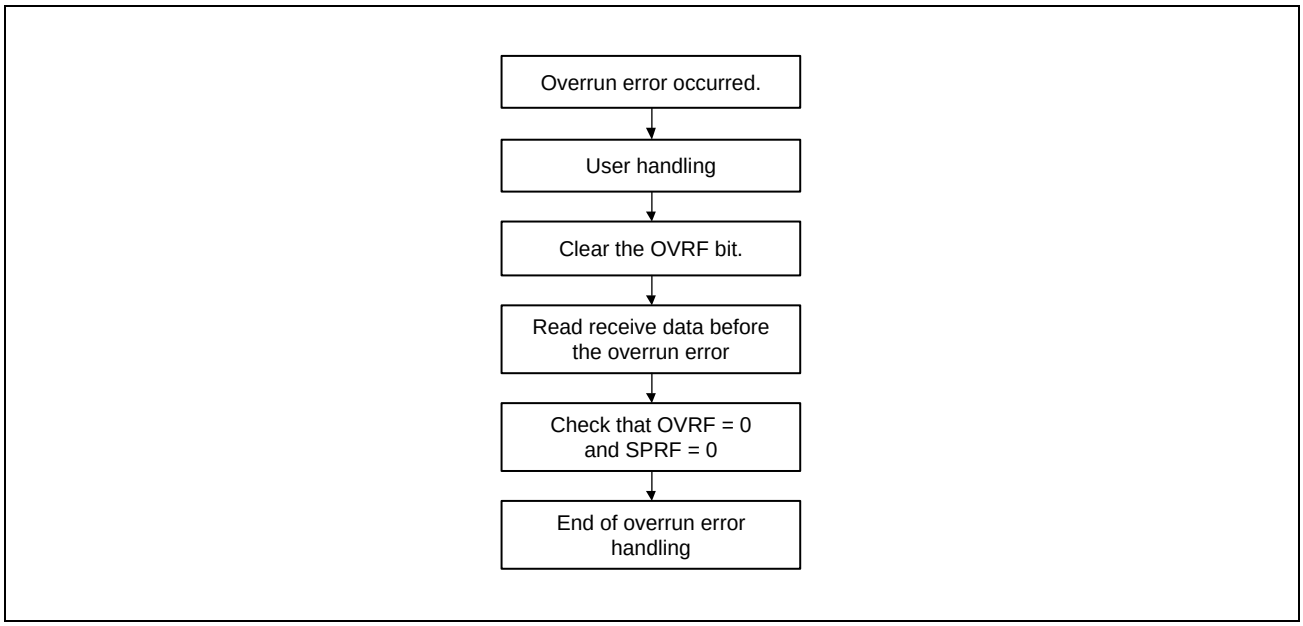


Figure 27.22 Error Handling (Overrun Error)

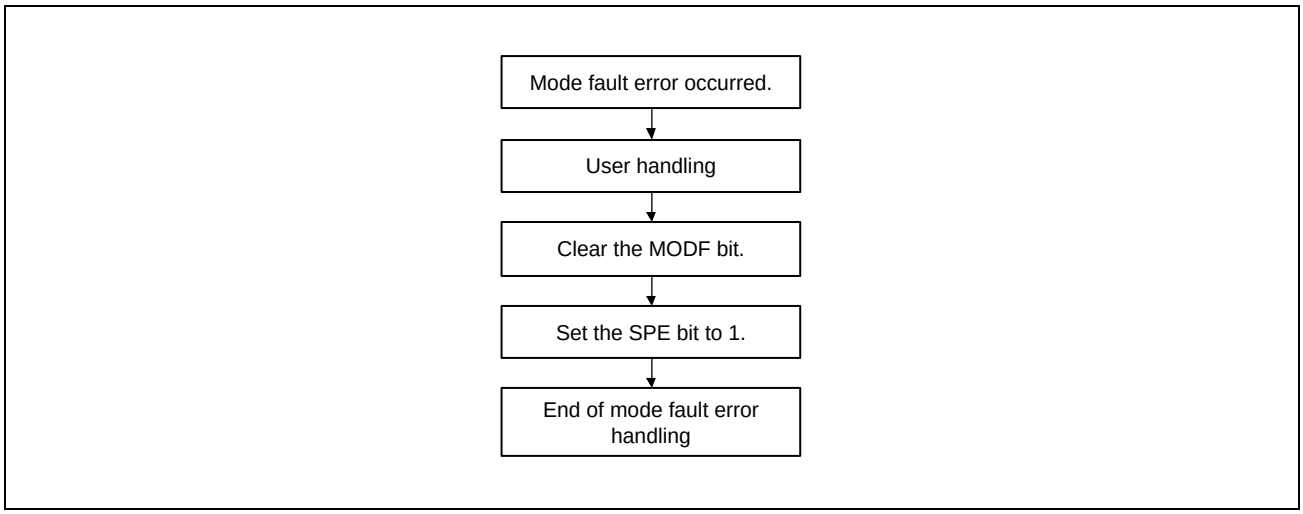


Figure 27.23 Error Handling (Mode Fault Error)

27.4.10 Loopback Mode

When 1 is written to the SPLP bit in the pin control register (SPPCR), this module shuts off the path between the MISO pin and the shift register, and between the MOSI pin and the shift register, and connects the input path and the output path (reversed) of the shift register. This is called loopback mode. When a serial transfer is executed in loopback mode, the transmit data becomes the received data. **Figure 27.24** shows the configuration of the shift register input/output paths for the case where this module in master mode is set in loopback mode.

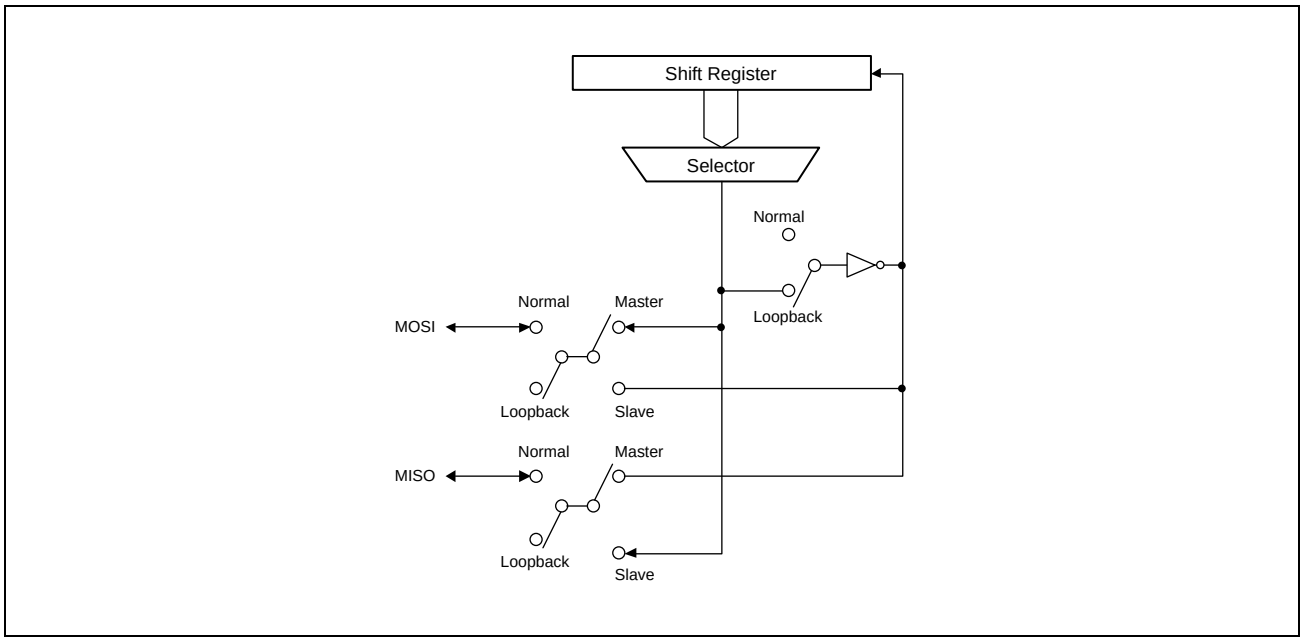


Figure 27.24 Configuration of Shift Register Input/Output Paths in Loopback Mode (Master Mode)

27.4.11 Interrupt Sources

This module has interrupt sources of receive buffer full, transmit buffer empty, mode fault, and overrun. In addition, the direct memory access controller can be activated by the receive buffer full or transmit buffer empty interrupt for data transfer.

Table 27.11 shows the interrupt sources.

When any of the interrupt conditions in **Table 27.11** is met, an interrupt is generated. The interrupt sources should be cleared with data transfer by the CPU or direct memory access controller.

Table 27.11 Interrupt Sources

Name	Interrupt Source	Abbreviation	Interrupt Condition	Activation of Direct Memory Access Controller
SPRI	Receive buffer full	RXI	(SPRIE = 1) • (SPRF = 1)	Possible
SPTI	Transmit buffer empty	TXI	(SPTIE = 1) • (SPTEF = 1)	Possible
SPEI	Mode fault	MOI	(SPEIE = 1) • (MODF = 1)	—
	Overrun	OVI	(SPEIE = 1) • (OVRF = 1)	—

28. Expanded Serial Peripheral Interface (xSPI)

28.1 Overview

This LSI has one unit of Expanded Serial Peripheral Interface (xSPI). The xSPI protocol specifies the interface for Memory Devices, which provides high data throughput, low signal count, and limited backward compatibility with legacy SPI devices. The electrical interface can deliver up to 266 MB per second raw data throughput.

Table 28.1 lists the xSPI specifications.

Table 28.1 xSPI Specifications

Item	Description
Number of channels	1 channel
Protocol	Compliant for the xSPI protocol
Data transmission and reception	Issue the transaction for up to 2 Slave as Master
Transfer speed	Support the transfer at xSPI200 or xSPI266
Mode	• Support Protocol modes below – 1/4/8pin with SDR/DDR (1S-1S-1S, 8D-8D-8D) – 2/4pin with SDR (1S-4S-4S, 4S-4S-4S) • Configurable address length • Configurable initial access latency cycle • Support XiP mode
xSPI function	• Support Write Data Mask • Support In-band Reset • Memory-mapping – Support up to 256 MBytes address space – Prefetch function for burst-read with low latency – Outstanding buffer for burst-write with high throughput • Manual command – Configurable up to 4 commands – Status Register Polling function – Automatic command issuing after released reset: up to 4 commands • Output Clock port timing shift • Input Strobe port timing shift

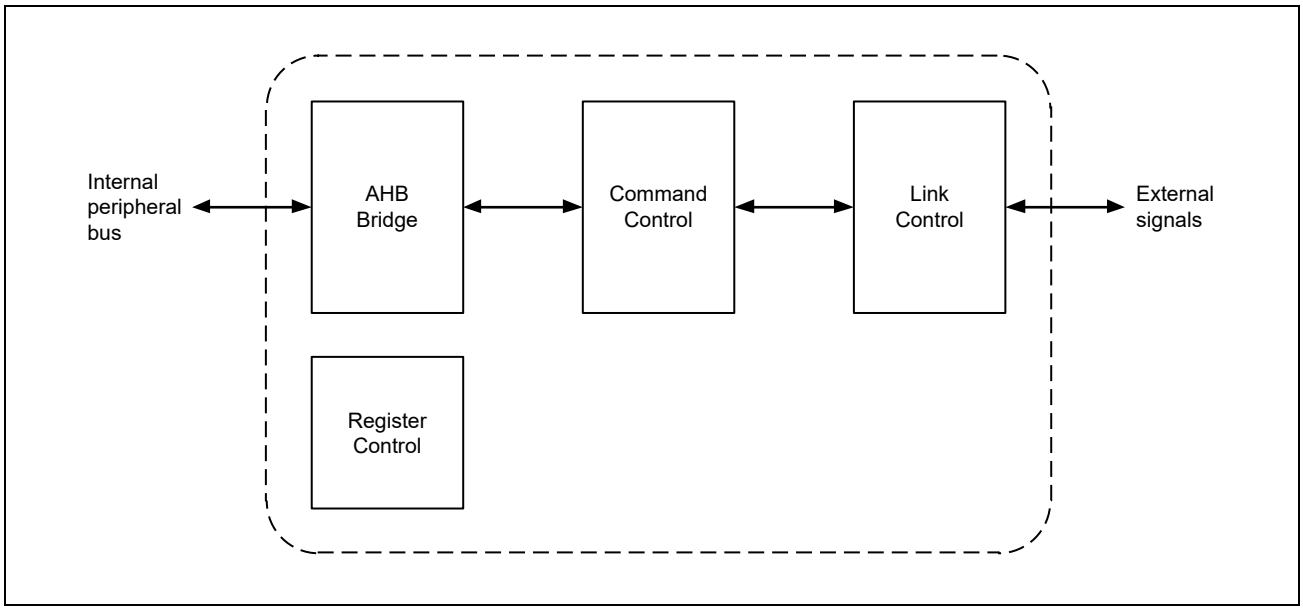


Figure 28.1 Block Diagram

28.2 Input/Output Pins

Table 28.2 xSPI I/O Pins

Pin Name	I/O	Function
XSPI_SPCLK	Output	SPI Clock
XSPI_CS0#	Output	Chip Select for slave0
XSPI_CS1#	Output	Chip Select for slave1
XSPI_DS	I/O	Read Data Strobe / Write Data Mask
XSPI_IO0	I/O	Data 0 input/output
XSPI_IO1	I/O	Data 1 input/output
XSPI_IO2	I/O	Data 2 input/output
XSPI_IO3	I/O	Data 3 input/output
XSPI_IO4	I/O	Data 4 input/output
XSPI_IO5	I/O	Data 5 input/output
XSPI_IO6	I/O	Data 6 input/output
XSPI_IO7	I/O	Data 7 input/output
XSPI_RESET0#	Output	Master reset status for slave0
XSPI_RESET1#	Output	Master reset status for slave1
XSPI_WP0#	Output	Write Protect for slave0
XSPI_WP1#	Output	Write Protect for slave1

Table 28.3 Address Map

Channel	Internal Address	Space
xSPI	H'2000_0000 to H'2FFF_FFFF	CS0
	—*1	CS1

Note 1. By default, CS1 does not have an address space. The address space of CS0 and CS1 can be set by registers in Section SYSC. When setting the address spaces of CS0 and CS1, overlapping addresses is prohibited.

28.3 Register Descriptions

Table 28.4 shows the register configuration. The address of the xSPI register is represented by the offset address from the base address. xSPI base address is as follows:

Base address: H'0_1006_0000 (Overall Address Space)
Base address: H'4006_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
Base address: H'5006_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above is in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 28.4 Register Configuration (1/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
xSPI Wrapper Configuration Register	WRAPCFG	R/W	H'0000_0000	H'000	32
xSPI Common Configuration Register	COMCFG	R/W	H'0000_0000	H'004	32
xSPI Bridge Map Configuration Register	BMCFG	R/W	H'0000_0000	H'008	32
xSPI Command Map Configuration Register 0 CS0	CMCFG0CS0	R/W	H'0000_0000	H'010	32
xSPI Command Map Configuration Register 0 CS1	CMCFG0CS1	R/W	H'0000_0000	H'020	32
xSPI Command Map Configuration Register 1 CS0	CMCFG1CS0	R/W	H'0008_0000	H'014	32
xSPI Command Map Configuration Register 1 CS1	CMCFG1CS1	R/W	H'0008_0000	H'024	32
xSPI Command Map Configuration Register 2 CS0	CMCFG2CS0	R/W	H'0008_0000	H'018	32
xSPI Command Map Configuration Register 2 CS1	CMCFG2CS1	R/W	H'0008_0000	H'028	32
xSPI Link I/O Configuration Register CS0	LIOCFGCS0	R/W	H'0007_0000	H'050	32
xSPI Link I/O Configuration Register CS1	LIOCFGCS1	R/W	H'0007_0000	H'054	32
xSPI Bridge Map Control Register 0	BMCTL0	R/W	H'0000_00FF	H'060	32
xSPI Bridge Map Control Register 1	BMCTL1	R/W	H'0000_0000	H'064	32
xSPI Command Map Control Register	CMCTL	R/W	H'0000_0000	H'068	32
xSPI Command Manual Control Register 0	CDCTL0	R/W	H'0000_0000	H'070	32
xSPI Command Manual Control Register 1	CDCTL1	R/W	H'0000_0000	H'074	32
xSPI Command Manual Control Register 2	CDCTL2	R/W	H'0000_0000	H'078	32
xSPI Command Manual Type Buf 0	CDTBUF0	R/W	H'0000_0000	H'080	32
xSPI Command Manual Type Buf 1	CDTBUF1	R/W	H'0000_0000	H'090	32
xSPI Command Manual Type Buf 2	CDTBUF2	R/W	H'0000_0000	H'0A0	32
xSPI Command Manual Type Buf 3	CDTBUF3	R/W	H'0000_0000	H'0B0	32
xSPI Command Manual Address Buf 0	CDABUF0	R/W	H'0000_0000	H'084	32
xSPI Command Manual Address Buf 1	CDABUF1	R/W	H'0000_0000	H'094	32
xSPI Command Manual Address Buf 2	CDABUF2	R/W	H'0000_0000	H'0A4	32
xSPI Command Manual Address Buf 3	CDABUF3	R/W	H'0000_0000	H'0B4	32
xSPI Command Manual Data 0 Buf 0	CDD0BUF0	R/W	H'0000_0000	H'088	32
xSPI Command Manual Data 0 Buf 1	CDD0BUF1	R/W	H'0000_0000	H'098	32
xSPI Command Manual Data 0 Buf 2	CDD0BUF2	R/W	H'0000_0000	H'0A8	32
xSPI Command Manual Data 0 Buf 3	CDD0BUF3	R/W	H'0000_0000	H'0B8	32
xSPI Command Manual Data 1 Buf 0	CDD1BUF0	R/W	H'0000_0000	H'08C	32
xSPI Command Manual Data 1 Buf 1	CDD1BUF1	R/W	H'0000_0000	H'09C	32

Table 28.4 Register Configuration (2/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
xSPI Command Manual Data 1 Buf 2	CDD1BUF2	R/W	H'0000_0000	H'0AC	32
xSPI Command Manual Data 1 Buf 3	CDD1BUF3	R/W	H'0000_0000	H'0BC	32
xSPI Link Pattern Control 0 register	LPCTL0	R/W	H'0000_0000	H'100	32
xSPI Link Pattern Control 1 register	LPCTL1	R/W	H'0000_0000	H'104	32
xSPI Link I/O Control register	LIOCTL	R/W	H'0003_0003	H'108	32
xSPI Command Calibration Control Register 0 CS0	CCCTL0CS0	R/W	H'1F00_0000	H'130	32
xSPI Command Calibration Control Register 0 CS1	CCCTL0CS1	R/W	H'1F00_0000	H'150	32
xSPI Command Calibration Control Register 1 CS0	CCCTL1CS0	R/W	H'0000_0000	H'134	32
xSPI Command Calibration Control Register 1 CS1	CCCTL1CS1	R/W	H'0000_0000	H'154	32
xSPI Command Calibration Control Register 2 CS0	CCCTL2CS0	R/W	H'0000_0000	H'138	32
xSPI Command Calibration Control Register 2 CS1	CCCTL2CS1	R/W	H'0000_0000	H'158	32
xSPI Command Calibration Control Register 3 CS0	CCCTL3CS0	R/W	H'0000_0000	H'13C	32
xSPI Command Calibration Control Register 3 CS1	CCCTL3CS1	R/W	H'0000_0000	H'15C	32
xSPI Command Calibration Control Register 4 CS0	CCCTL4CS0	R/W	H'0000_0000	H'140	32
xSPI Command Calibration Control Register 4 CS1	CCCTL4CS1	R/W	H'0000_0000	H'160	32
xSPI Command Calibration Control Register 5 CS0	CCCTL5CS0	R/W	H'0000_0000	H'144	32
xSPI Command Calibration Control Register 5 CS1	CCCTL5CS1	R/W	H'0000_0000	H'164	32
xSPI Command Calibration Control Register 6 CS0	CCCTL6CS0	R/W	H'0000_0000	H'148	32
xSPI Command Calibration Control Register 6 CS1	CCCTL6CS1	R/W	H'0000_0000	H'168	32
xSPI Command Calibration Control Register 7 CS0	CCCTL7CS0	R/W	H'0000_0000	H'14C	32
xSPI Command Calibration Control Register 7 CS1	CCCTL7CS1	R/W	H'0000_0000	H'16C	32
xSPI Version Register	VERSTT	R	H'0000_0000	H'180	32
xSPI Common Status Register	COMSTT	R	H'0000_0000	H'184	32
xSPI Calibration Status Register CS0	CASTTCS0	R	H'0000_0000	H'188	32
xSPI Calibration Status Register CS1	CASTTCS1	R	H'0000_0000	H'18C	32
xSPI Interrupt Status Register	INTS	R	H'0000_0000	H'190	32
xSPI Interrupt Clear Register	INTC	W	H'0000_0000	H'194	32
xSPI Interrupt Enable Register	INTE	R/W	H'0000_0000	H'198	32

28.3.1 xSPI Configuration Registers

These registers configure xSPI Master function. These registers should be configured in the initialization phase before issuing xSPI transaction.

28.3.1.1 xSPI Wrapper Configuration Register (WRAPCFG)

This register has functions to configure xSPI Master function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	DSSFTCS1[4:0]					—	—	—	CKSFTCS1[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	DSSFTCS0[4:0]					—	—	—	CKSFTCS0[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
28 to 24	DSSFTCS1 [4:0]	All 0	R/W	DS shift for slave1 The function is same as one of slave0.
23 to 21	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
20 to 16	CKSFTCS1 [4:0]	All 0	R/W	CK shift for slave1 The function is same as one of slave0.
15 to 13	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
12 to 8	DSSFTCS0 [4:0]	All 0	R/W	DS shift for slave0 These bits configure the number of delay cell for XSPI_DS pin. It is used to adjust the DS sampling timing. 1 cell delay is around 125 ps. When enabled automatic calibration, it can be updated automatically. H'00: No shift H'01: Add a delay of 1 cell ⋮ H'1E: Add a delay of 30 cells H'1F: Add a delay of 31 cells
7 to 5	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
4 to 0	CKSFTCS0 [4:0]	All 0	R/W	CK shift for slave0 This field configures the number of delay cell for a XSPI_SPCLK port. It is used to adjust the CK driving timing. 1 cell delay is around 125 ps. When enabled automatic calibration, it can be updated automatically. H'00: No shift H'01: Add a delay of 1 cell ⋮ H'1E: Add a delay of 30 cells H'1F: Add a delay of 31 cells

28.3.1.2 xSPI Common Configuration Register (COMCFG)

This register has functions to configure xSPI Master function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OENEGEX	OEASTEX
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
17	OENEGEX	0b	R/W	Output Enable Negating extension This bit extends 1 cycle output enable of Data and DS during output enable negating. This bit should not be used in case of no latency cycle. Because SPI output data could be conflicted with SPI input data. 0: No extend 1 cycle Output enable 1: Extend 1 cycle Output enable
16	OEASTEX	0b	R/W	Output Enable Asserting extension This bit extends 1 cycle output enable of Data and DS during output enable asserting. When set to 1, CS asserting should be extended (LIOCFGCSn.CSASTEX = 1 (n = 0, 1)). This bit shall not be used in case of no latency cycle. Because xSPI output data could be conflicted with xSPI input data. 0: No extend 1 cycle Output enable 1: Extend 1 cycle Output enable
15 to 0	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

28.3.1.3 xSPI Bridge Map Configuration Register (BMCFG)

This register has functions to configure xSPI Master function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMBTIM[7:0]								—	—	—	—	—	—	—	PREEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

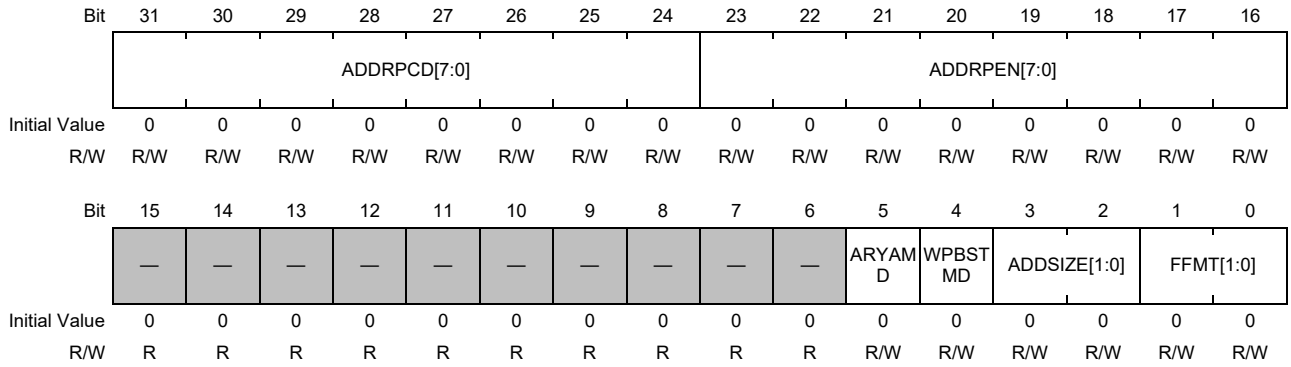
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MWRSIZE[7:0]								MWRCOMB	—	—	—	—	—	—	WRMD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CMBTIM[7:0]	All 0	R/W	Combination timer This field specifies expiration period of combination timer.H'00 means disabling the combination timer. When the timer is expired, the data in the combination buffer is pushed to memory device.
23 to 17	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
16	PREEN	0b	R/W	Prefetch enable This bit enables prefetch function for read transaction in memory-mapping mode. It could reduce the latency for AHB read transaction with incremental address. 0: Disable prefetch function 1: Enable prefetch function
15 to 8	MWRSIZE [7:0]	All 0	R/W	Memory Write Size These bits select the size to combine incremental address in memory-mapping mode. It transmits an xSPI frame with the data combined up to the configured size while the address is incremental. When detected non-incremental address or a read transaction before reaching to the target size, it transmits the pending data into xSPI bus. H'00: Combine incremental address up to 4 bytes H'01: Combine incremental address up to 8 bytes ⋮ H'0E: Combine incremental address up to 60 bytes H'0F: Combine incremental address up to 64 bytes Others: Setting prohibited
7	MWRCOMB	0b	R/W	Memory Write Combination mode This field selects to combine the xSPI data in write access of memory mapping-mode. When this field is set to "0", xSPI data size depends on system bus's burst type and size. When this field is set to "1", the data size depends on MWRSIZE field. *Note*: When this field is set to "1", any write transaction could be held in this xSPI master temporarily. 0: Disable combination mode 1: Enable combination mode
6 to 1	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	WRMD	0b	R/W	System bus Write Response mode This field selects the timing of system bus write response in memory-mapping mode. When this field is set to “1”, it returns the response after transmitting a frame on xSPI bus. *Note*: When this mode is enabled, Memory Write Combination mode shall be disabled. 0: Return response after storing to Internal Write Buffer 1: Return response after issuing write transaction to xSPI bus

28.3.1.4 xSPI Command Map Configuration Register 0 CSn (CMCFG0CSn) (n = 0, 1)

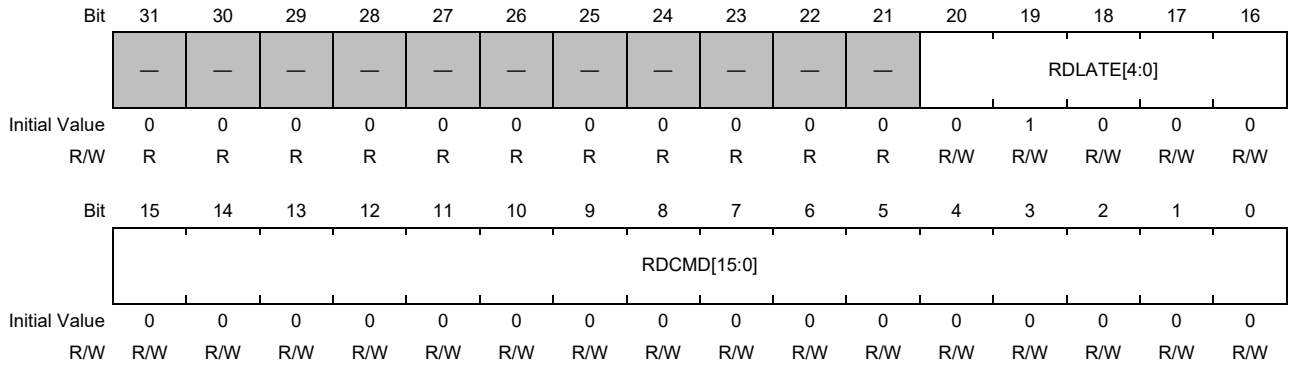
This register has functions to configure xSPI Master function.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ADDRPCD [7:0]	All 0	R/W	Address Replace Code These bits configure the code to replace the MSByte of system bus address in memory-mapping mode. It replaces the corresponding bits when Address Replace Enable bit is set to 1.
23 to 16	ADDRPEN [7:0]	All 0	R/W	Address Replace Enable These bits select the bits to replace for MSByte of system bus address in memory-mapping mode. 0: No replacement (xSPI frame address field is same as system bus address) 1: Replacement
15 to 6	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
5	ARYAMD	0b	R/W	Array address mode 0: Normal address mode 1: Array address mode When this field is set to 1, address for memory is mapped as {A [25:10], A [9:4], 6{RSV}, A [3:0]} where A [25:0] is normal address, and RSV is reserved value(0b). This field is effective only when FFMT [1:0] = 01b.
4	WPBSTMD	0b	R/W	Wrapping burst mode 0: Separate xSPI transfer at the wrapping address boundary 1: Not separate xSPI transfer at the wrapping address boundary
3, 2	ADDSIZE[1:0]	00b	R/W	Address size These bits configure the number of address byte in memory-mapping mode. In case of 8D-8D-8D profile 2.0, it should be configured to 4 bytes. 0 0: 1 byte (256 bytes address space) 0 1: 2 bytes (64 Kbytes address space) 1 0: 3 bytes (16 Mbytes address space) 1 1: 4 bytes (4 Gbytes address space)
1, 0	FFMT[1:0]	00b	R/W	Frame format These bits configure xSPI frame format in memory-mapping mode. Please see **Table 28.10** for detail. 0 0: Normal format: Command 1 byte, Address ADDSIZE, Data to AHB transaction. 0 1: 8D-8D-8D profile 1.0 format: Command 2 bytes, Address ADDSIZE, Data to AHB transaction 1 0: 8D-8D-8D profile 2.0 Command Modifier format: Command & Modifier 6 bytes, Data to AHB transaction 1 1: 8D-8D-8D profile 2.0 Commands with Extended Command Modifier format: Command & Modifier 6 bytes, Data to AHB transaction

28.3.1.5 xSPI Command Map Configuration Register 1 CSn (CMCFG1CSn) (n = 0, 1)

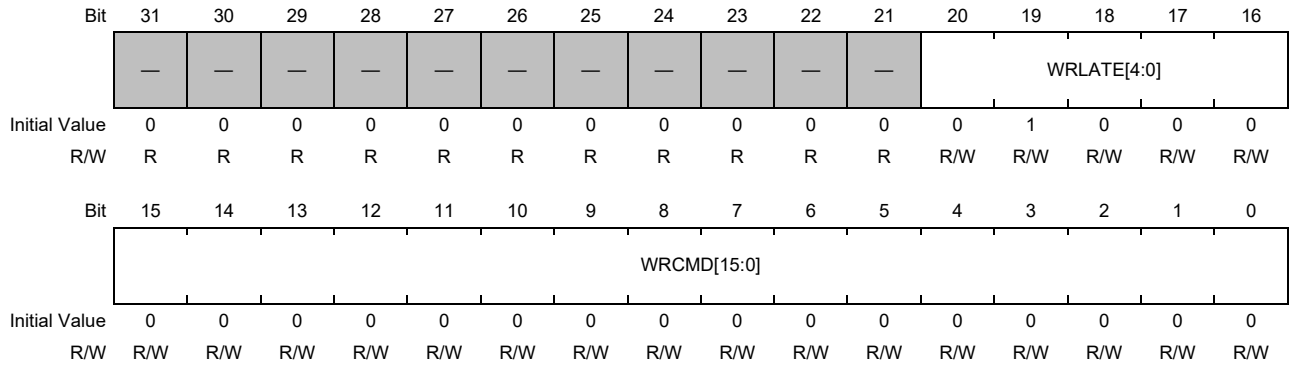
This register has functions to configure xSPI Master function.



Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
20 to 16	RDLATE[4:0]	H'08	R/W	Read latency cycle These bits configure the latency cycle of read transaction in memory-mapping mode. H'00: No latency H'01: 1 cycle ⋮ H'1E: 30 cycles H'1F: 31 cycles
15 to 0	RDCMD[15:0]	All 0	R/W	Read command These bits configure the command field of read transaction in memory-mapping mode. Normal format and 8D-8D-8D profile 2.0 format use only upper 1 byte. 8D-8D-8D profile 1.0 format uses 2 bytes.

28.3.1.6 xSPI Command Map Configuration Register 2 CSn (CMCFG2CSn) (n = 0, 1)

This register has functions to configure xSPI Master function.



Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
20 to 16	WRLATE[4:0]	H'08	R/W	Write latency cycle These bits configure the latency cycle of write transaction in memory-mapping mode. H'00: No latency H'01: 1 cycle ⋮ H'1E: 30 cycles H'1F: 31 cycles
15 to 0	WRCMD [15:0]	All 0	R/W	Write command These bits configure the command field of write transaction in memory-mapping mode. Normal format and 8D-8D-8D profile 2.0 format use only upper 1 byte. 8D-8D-8D profile 1.0 format uses 2 bytes.

28.3.1.7 xSPI Link I/O Configuration Register CSn (LIOCFGCSn) (n = 0, 1)

This register has functions to configure xSPI Master function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DDRSMPEX[3:0]				SDRSMPSFT[3:0]				SDRSMPMD	SDRDRV	CSNEGEX	CSASTEX	CSMIN[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	WRMSKMD	LATEMD	PRTMD[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	DDRSMPEX[3:0]	All 0	R/W	DDR sampling window extend These bits configure the cycle of extending the sampling window in DDR. In DDR, the input data is sampled during the expected cycle soon after latency cycle. The input data out of range is ignored. It can be configured depending on DS propagation delay. H'0: Expand no cycle H'1: Expand 1 cycle ⋮ H'6: Expand 6 cycles H'7: Expand 7 cycles Others: Setting prohibited
27 to 24	SDRSMPSFT[3:0]	All 0	R/W	SDR Sampling window shift These bits shift the timing of CK sampling in SDR. In case of using DS in SDR, there is no influence on the behavior. In case of DDR or using DS in SDR, it should be set to 0. H'0: Sample without delay H'1: Sample at 1 cycle delay ⋮ H'6: Sample at 6 cycle delay H'7: Sample at 7 cycle delay Others: Setting prohibited
23	SDRSMPMD	0b	R/W	SDR Sampling mode This bit selects the edge of sampling in SDR. In DDR, regardless of this setting, it samples data input ports with both edges of DS. When set to 1, it samples at rising-edge before falling-edge. 0: Samples data input at falling-edge 1: Samples data input at rising-edge
22	SDRDRV	0b	R/W	SDR driving timing This bit configures the timing of data output in SDR. This bit should not be used in case of no latency cycle. Because SPI output data could be conflicted with SPI input data. 0: Drive at 1/2 cycle before CK rising-edge 1: Drive at CK rising-edge
21	CSNEGEX	0b	R/W	CS negating extension This bit extends 1 cycle chip select pins when negating. 0: No extension 1: Extend 1 cycle

Bit	Bit Name	Initial Value	R/W	Description
20	CSASTEX	0b	R/W	CS asserting extension This bit extends 1 cycle chip select pins when asserting. 0: No extension 1: Extend 1 cycle
19 to 16	CSMIN[3:0]	H'7	R/W	CS minimum idle term This bit configures the minimum cycle between xSPI frames. H'0: 1 cycle H'1: 2 cycles ⋮ H'E: 15 cycles H'F: 16 cycles
15 to 12	—	0b	R	Reserved These bits are read as 0. The write value should be 0.
11	WRMSKMD	0b	R/W	Write mask mode This bit selects to use Data strobe port as write data mask. It could be useful for write access of odd byte. It is used only for 8D-8D-8D protocol mode. 0: Write mask disable 1: Write mask enable
10	LATEMD	0b	R/W	Latency mode This bit selects the behavior of initial access latency phase for both direct-manual mode and memory-mapping mode. When set to 0, the latency cycle is equal to each configured cycle from transmitting address field. When set to 1, the latency cycle is incremented from the last byte-pair of Address field and is extended 2 times of each configured cycle depending on data strobe port. It is used only for profile 2.0 frame format of 8D-8D-8D protocol mode with 6 bytes command/address field. Please refer to xSPI protocol for detail.
9 to 0	PRTMD[9:0]	All 0	R/W	Protocol mode These bits configure the protocol mode and the pin to sample data inputs. In case of using not SPI clock but Data strobe for sampling in SDR mode, it is required to set PRTMD[9] to 1. H'000: 1S-1S-1S H'3FF: 8D-8D-8D H'090: 1S-4S-4S H'092: 4S-4S-4S Others: Setting prohibited

Value: Function	Frame format	Usage
0: Configurable latency	profile 2.0	The configurable latency cycle should be set as minus 1.
	Others	Latency cycle increments after address field.
1: Variable latency	profile 2.0	Latency cycle increments from address [23:16]. And it should not be set to 1.
	Others	Not support

28.3.2 xSPI Control Registers

These registers control xSPI Master function.

28.3.2.1 xSPI Bridge Map Control Register 0 (BMCTL0)

This register has functions to control xSPI Master function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CS1ACC[1:0]		CS0ACC[1:0]	
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7 to 4	—	All 1	R	Reserved These bits are read as 1. The write value should be 1.
3, 2	CS1ACC[1:0]	11b	R/W	System bus channel to slave1 memory area access enable These bits enable the access from system bus channel to CS1 memory. 0 0: Read/Write disable 0 1: Read enable, Write disable 1 0: Read disable, Write enable 1 1: Read/Write enable
1, 0	CS0ACC[1:0]	11b	R/W	System bus channel to slave0 memory area access enable These bits enable the access from system bus channel to CS0 memory. 0 0: Read/Write disable 0 1: Read enable, Write disable 1 0: Read disable, Write enable 1 1: Read/Write enable

28.3.2.2 xSPI Bridge Map Control Register 1 (BMCTL1)

This register has functions to control xSPI Master function.

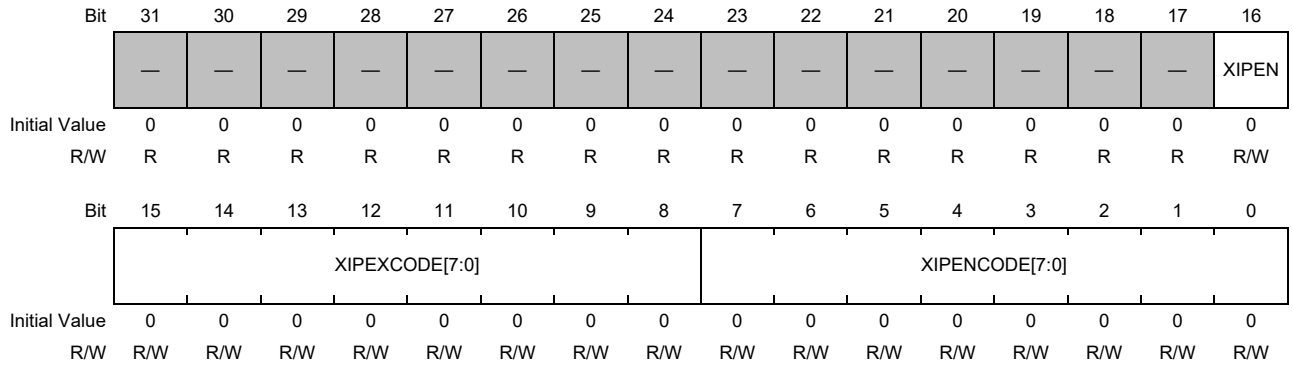
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PBUFCLR	—	MWRPUSH	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	W	R	W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved The write value should be 0.
10	PBUFCLR	0b	W	Prefetch Buffer clear This bit requests to clear the prefetch buffer when enabled the prefetch function. It should not be set during memory access (COMSTT.MEMACC = 1). 0: No command 1: Clear request
9	—	0b	R	Reserved The write value should be 0.
8	MWRPUSH	0b	W	Memory Write Data Push This bit requests to push the pending data in combination mode. 0: No command 1: Push request
7 to 0	—	All 0	R	Reserved The write value should be 0.

28.3.2.3 xSPI Command Map Control Register (CMCTL)

This register has functions to control xSPI Master function.



Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
16	XIPEN	0b	R/W	XiP mode enable This bit enables XiP mode in memory-mapping mode. When set to 1, XiP enter code is inserted in the latency field, and the command field in next transaction is omitted. When set to 0, XiP exit code is inserted in the latency field. And it is set to 0 automatically when transmitting XiP disable pattern. It should not be used for 8D-8D-8D protocol mode profile 2.0 frame format. 0: Disable XiP mode 1: Enable XiP mode
15 to 8	XIPEXCODE [7:0]	All 0	R/W	XiP mode exit code These bits configure the code to exit XiP mode in memory-mapping mode.
7 to 0	XIPENCODE [7:0]	All 0	R/W	XiP mode enter code These bits configure the code to enter XiP mode in memory-mapping mode.

28.3.2.4 xSPI Command Manual Control Register 0 (CDCTL0)

This register has functions to control xSPI Master function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	PERREP[3:0]				—	—	—	PERITV[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	TRNUM[1:0]		CSSEL	—	PERMD	TRREQ
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
27 to 24	PERREP[3:0]	All 0	R/W	Periodic transaction repeat These bits configure the number of transaction repetitions in periodic manual-command mode. H'0: 1 (= 2^0) time H'1: 2 (= 2^1) times ⋮ H'E: 16384 (= 2^{14}) times H'F: 32768 (= 2^{15}) times
23 to 21	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
20 to 16	PERITV[4:0]	All 0	R/W	Periodic transaction interval These bits configure the interval of transaction in periodic manual-command mode. Too short interval compared with CPU bus cycle could result in no store into command buffer0. The interval should be longer than 4 times of CPU bus cycle. H'00: 2 (= 2^1) cycles H'01: 4 (= 2^2) cycles ⋮ H'1E: 2,147,483,648 (= 2^{31}) cycles H'1F: 4,294,967,296 (= 2^{32}) cycles
15 to 6	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
5, 4	TRNUM[1:0]	00b	R/W	Transaction number These bits configure the number of transactions in normal manual-command mode. In periodic manual-command, regardless of this setting, it issues the transaction using only command buffer 0. 0 0: Issue 1 command (using command buffer 0) 0 1: Issue 2 commands (using command buffer 0-1) 1 0: Issue 3 commands (using command buffer 0-2) 1 1: Issue 4 commands (using command buffer 0-3)
3	CSSEL	0b	R/W	Chip select This bit selects a target memory to issue manual-command. 0: CS0 1: CS1
2	—	0b	R	Reserved This bit is read as 0. The write value should be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	PERMD	0b	R/W	Periodic mode This bit enables the periodic transaction mode. When set to 1, it repeats a transaction periodically and compares the read value with the expected value. It alternates the status polling operation for external memory. 0: Direct manual-command mode 1: Periodic manual-command mode
0	TRREQ	0b	R/W	Transaction request This bit requests to issue the transaction of manual-command. When set to 1, it starts the transaction. It is cleared to 0 when the transaction completed. The transaction is canceled by clearing to 0 while the transaction is ongoing. 0: No transaction 1: Request transaction

28.3.2.5 xSPI Command Manual Control Register 1 (CDCTL1)

This register has functions to control xSPI Master function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PEREXP[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PEREXP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PEREXP [31:0]	All 0	R/W	Periodic transaction expected value These bits configure the expected value to compare with the read value in periodic manual- command mode. For example, in case of comparing 1 byte, the lower byte should be configured.

28.3.2.6 xSPI Command Manual Control Register 2 (CDCTL2)

This register has functions to control xSPI Master function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PERMSK[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PERMSK[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PERMSK [31:0]	All 0	R/W	Periodic transaction masked value These bits configure the masked value for the expected value in periodic manual-command mode. When set 1 to any bit, the corresponding bit configured as expected value (CDCTL1.PEREXP[31:0]) is ignored. In 8D-8D-8D, the data bytes are transferred only in byte pairs on xSPI bus. It means the dummy read data could be stored. It should be masked for unused bits. For example, in case of read lower 1 byte, it should be configured to H'FFFF_FF00.

28.3.2.7 xSPI Command Manual Type Buf n (CDTBUFn) (n = 0 to 3)

This register has functions to control xSPI Master function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRTYPE	—	LATE[4:0]					DATASIZE[3:0]				ADDSIZE[2:0]			CMDSIZE[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	CMD[15:0]	All 0	R/W	Command (1-2 bytes) These bits configure the command field in manual-command mode. The number of bytes configured in Command Size bit is transferred. 1S-1S-1S: CMD[15:8] is command field, CMD[7:0] is not used. 8D-8D-8D profile 1.0: CMD[15:8] is command field, CMD[7:0] is extension field. 8D-8D-8D profile 2.0: CMD[15:0] is upper 2 bytes of command & modifier field. (bit 47-32 in xSPI protocol)
15	TRTYPE	0b	R/W	Transaction Type This bit selects the type of transaction. 0: Read transaction (Readout data from slave device) 1: Not read transaction
14	—	0b	R	Reserved This bit is read as 0. The write value should be 0.
13 to 9	LATE[4:0]	All 0	R/W	Latency cycle These bits configure the latency cycle in manual-command mode. H'0: No latency H'1: 1 cycle ⋮ H'1E: 30 cycles H'1F: 31 cycles
8 to 5	DATASIZE[3:0]	All 0	R/W	Write/Read Data Size These bits configure the size of data field. In 8D-8D-8D, the data bytes are transferred only in byte pairs on xSPI bus. For example, even if configuring 1 byte for read, 2 bytes data is received. The last byte should be ignored. The 0 bytes must not be configured for read transaction. H'0: 0 bytes (No data phase) H'1: 1 byte : H'7: 7 bytes H'8: 8 bytes Others: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
4 to 2	ADDSIZE[2:0]	All 0	R/W	Address size These bits configure the size of address field. 0 0 0: 0 bytes (No address phase) 0 0 1: 1 byte 0 1 0: 2 bytes 0 1 1: 3 bytes 1 0 0: 4 bytes Others: Setting prohibited
1, 0	CMDSIZE [1:0]	All 0	R/W	Command Size These bits configure the size of command field. In case of 8D-8D-8D, it should be fixed to 10b. It should not be configured both command size and address size to zero. 0 0: 0 bytes (No command phase) 0 1: 1 byte 1 0: 2 bytes Others: Setting prohibited

28.3.2.8 xSPI Command Manual Address Buf n (CDABUFn) (n = 0 to 3)

This register has functions to control xSPI Master function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADD[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ADD[31:0]	All 0	R/W	Address These bits configure the address field in manual-command mode. 1S-1S-1S, 8D-8D-8D profile 1.0: It is address field. 8D-8D-8D profile 2.0: It is lower 4 bytes of command & modifier field. (bit 31-0 in xSPI protocol)

28.3.2.9 xSPI Command Manual Data 0 Buf n (CDD0BUFn) (n = 0 to 3)

This register has functions to control xSPI Master function.

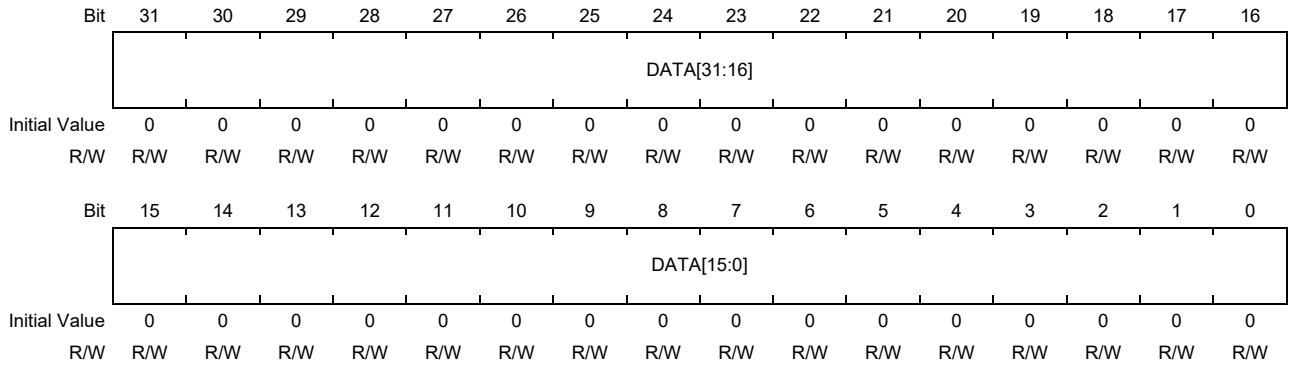
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	All 0	R/W	Write/Read Data These bits configure the data field in manual-command mode. In case of write transaction, the write data should be configured. In case of read transaction, the read data is stored.

28.3.2.10 xSPI Command Manual Data 1 Buf n (CDD1BUFn) (n = 0 to 3)

This register has functions to control xSPI Master function.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	All 0	R/W	Write/Read Data These bits configure the data field in manual-command mode. In case of write transaction, the write data should be configured. In case of read transaction, the read data is stored.

28.3.2.11 xSPI Link Pattern Control Register 0 (LPCTL0)

This register has functions to control xSPI Master function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	XD2VAL	—	—	XD2LEN[4:0]					XD1VAL	—	—	XD1LEN[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

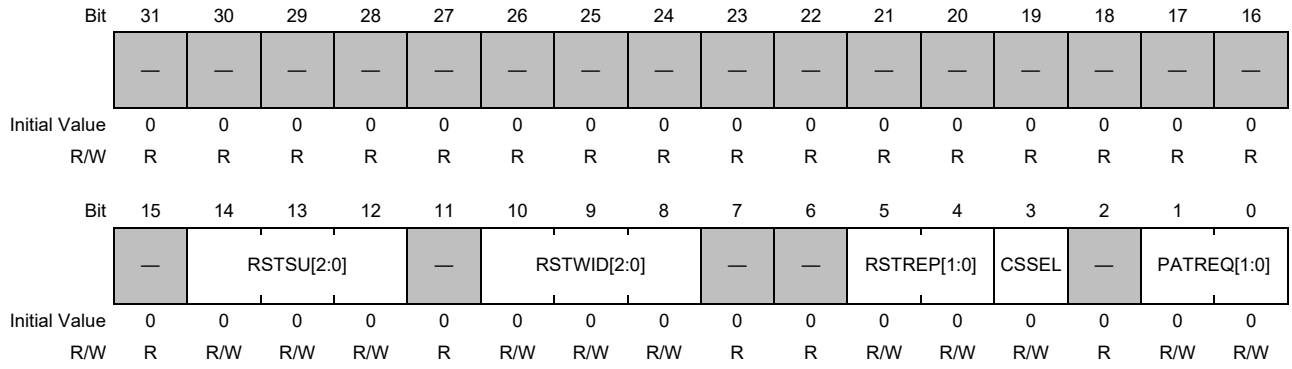
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	XDPIN[1:0]		CSSEL	—	—	PATREQ
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	XD2VAL	0b	R/W	XiP Disable pattern 2nd phase value This bit selects the value of 2nd phase in XiP disable pattern. 0: Low drive 1: High drive
30, 29	—	00b	R	Reserved These bits are read as 0. The write value should be 0.
28 to 24	XD2LEN[4:0]	All 0	R/W	XiP Disable pattern 2nd phase length These bits select the length of 2nd phase in XiP disable pattern. H'00: 0 cycles H'01: 1 cycle ⋮ H'1E: 30 cycles H'1F: 31 cycles
23	XD1VAL	0b	R/W	XiP Disable pattern 1st phase value This bit selects the value of 1st phase in XiP disable pattern. 0: Low drive 1: High drive
22, 21	—	00b	R	Reserved These bits are read as 0. The write value should be 0.
20 to 16	XD1LEN[4:0]	All 0	R/W	XiP Disable pattern 1st phase length These bits select the length of 1st phase in XiP disable pattern. The pattern with zero-length both 1st phase and 2nd phase should not be configured. H'0: 0 cycles H'1: 1 cycle ⋮ H'1E: 30 cycles H'1F: 31 cycles
15 to 6	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
5, 4	XDPIN[1:0]	00b	R/W	XiP Disable pattern pin These bits select the data output pins to transmit XiP Disable pattern. 0 0: 1 pin 0 1: 2 pins 1 0: 4 pins 1 1: 8 pins

Bit	Bit Name	Initial Value	R/W	Description
3	CSSEL	0b	R/W	Chip select This bit selects a target memory to issue a pattern. 0: slave0 (CS0) 1: slave1 (CS1)
2, 1	—	00b	R	Reserved These bits are read as 0. The write value should be 0.
0	PATREQ	0b	R/W	Pattern request This bit requests to issue the pattern. When set to 1, it starts the pattern. It is cleared to 0 when the pattern completed. 0: No request XiP Disable pattern 1: Request XiP Disable pattern

28.3.2.12 xSPI Link Pattern Control Register 1 (LPCTL1)

This register has functions to control xSPI Master function.



Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
14 to 12	RSTSU[2:0]	000b	R/W	Reset pattern data output setup time These bits configure the number of setup cycle for data output based on the edge of CS in reset pattern. It needs enough setup time because xSPI slave samples any data at the rising edge of CS. This cycle of setup time should be less than the cycle of reset pattern width (RSTWID[2:0]). 0 0 0: 1 cycle 0 0 1: 2 cycles ⋮ 1 1 0: 7 cycles 1 1 1: 8 cycles
11	—	0b	R	Reserved This bit is read as 0. The write value should be 0.
10 to 8	RSTWID[2:0]	000b	R/W	Reset pattern width These bits configure the width of cycle in reset pattern and CS only pattern. It toggles CS with the configured cycle. 0 0 0: 2 (= 2^1) cycles 0 0 1: 4 (= 2^2) cycles ⋮ 1 1 0: 128 (= 2^7) cycles 1 1 1: 256 (= 2^8) cycles
7, 6	—	00b	R	Reserved These bits are read as 0. The write value should be 0.
5, 4	RSTREP[1:0]	00b	R/W	Reset pattern repeat These bits select the repeating time to toggle CS from LOW to HIGH. 0 0: 4 times (Specified on Reset Signaling Protocol) 0 1: 5 times 1 0: 6 times 1 1: 7 times
3	CSSEL	0b	R/W	Chip select This bit selects a target memory to issue a pattern. 0: slave0 (CS0) 1: slave1 (CS1)

Bit	Bit Name	Initial Value	R/W	Description
2	—	0b	R	Reserved This bit is read as 0. The write value should be 0.
1, 0	PATREQ[1:0]	00b	R/W	Pattern request These bits request to issue the pattern. When set to 01b or 10b, it starts the pattern. It is cleared to 00b when the pattern completed. 0 0: No request 0 1: Request Reset pattern 1 0: Request CS only pattern 1 1: Setting prohibited

28.3.2.13 xSPI Link I/O Control Register (LIOCTL)

This register has functions to control xSPI Master function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RSTCS1	RSTCS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WPCS1	WPCS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved This bit is read as 0. The write value should be 0.
17	RSTCS1	1b	R/W	Reset drive for slave1 This function is same as RSTCS0. 0: Drive Low level 1: Drive High level
16	RSTCS0	1b	R/W	Reset drive for slave0 This bit controls the value of xSPI reset port. It can be useful only for xSPI slave with reset port. 0: Drive Low level 1: Drive High level
15 to 2	—	All 0	R	Reserved This bit is read as 0. The write value should be 0.
1	WPCS1	1b	R/W	WP drive for slave1 This function is same as WPCS0. 0: Drive Low level 1: Drive High level
0	WPCS0	1b	R/W	WP drive for slave0 This bit controls the value of Write Protect port. It can be useful only for xSPI slave with write protect port. 0: Drive Low level 1: Drive High level

28.3.2.14 xSPI Command Calibration Control Register 0 CSn (CCCTL0CSn) (n = 0, 1)

This register has functions to control xSPI Master function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	CASFTEND[4:0]					—	—	—	CASFTSTA[4:0]				
Initial Value	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CAITV[4:0]					—	—	—	—	—	—	CANOWR	CAEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
28 to 24	CASFTEND [4:0]	All 1	R/W	Calibration DS shift end value These bits configure the end value of DS shift. It should be equal or more than the start value (CASFTSTA).
23 to 21	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
20 to 16	CASFTSTA [4:0]	All 0	R/W	Calibration DS shift start value These bits configure the start value of DS shift.
15 to 13	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
12 to 8	CAITV[4:0]	All 0	R/W	Calibration interval These bits configure the interval between calibration patterns. H'00: 2 (= 2^1) cycle wait H'01: 4 (= 2^2) cycle wait ⋮ H'1E: 2,147,483,648 (= 2^{31}) cycle wait H'1F: 4,294,967,296 (= 2^{32}) cycle wait
7 to 2	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
1	CANOWR	0b	R/W	Calibration no write mode This bit selects to omit write command in calibration sequence. It can be used for any slave device with fixed calibration pattern data. 0: Calibration sequence with write command 1: Calibration sequence without write command
0	CAEN	0b	R/W	Automatic Calibration Enable This bit enables the automatic calibration. When set to 1, it transmits the calibration sequence periodically and adjusts the value of phase shift. When set to 0 during the calibration sequence, it stops after completed ongoing calibration sequence, and then this bit is cleared. 0: Disable automatic calibration 1: Enable automatic calibration

28.3.2.15 xSPI Command Calibration Control Register 1 CSn (CCCTL1CSn) (n = 0, 1)

This register has functions to control xSPI Master function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	CARDLATE[4:0]					—	—	—	CAWRLATE[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CADATASIZE[3:0]				CAADDSIZE[2:0]			CACMDSIZE [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
28 to 24	CARDLATE [4:0]	All 0	R/W	Read Latency cycle These bits configure the latency cycle in calibration frame. H'00: No latency H'01: 1 cycle ⋮ H'1E: 30 cycles H'1F: 31 cycles
23 to 21	—	000b	R	Reserved These bits are read as 0. The write value should be 0.
20 to 16	CAWRLATE [4:0]	All 0	R/W	Write Latency cycle These bits configure the latency cycle in calibration frame. H'00: No latency H'01: 1 cycle ⋮ H'1E: 30 cycles H'1F: 31 cycles
15 to 9	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
8 to 5	CADATASIZE [3:0]	All 0	R/W	Write/Read Data Size These bits configure the size of data field. In 8D-8D-8D, it should be configured with even byte. H'0: 1 byte H'1: 2 bytes ⋮ H'E: 15 bytes H'F: 16 bytes
4 to 2	CAADDSIZE [2:0]	000b	R/W	Address size These bits configure the size of address field. 0 0 0: 0 bytes (No address phase) 0 0 1: 1 byte 0 1 0: 2 bytes 0 1 1: 3 bytes 1 0 0: 4 bytes Others: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
1, 0	CACMDSIZE [1:0]	00b	R/W	Command Size These bits configure the size of command field. In case of 8D-8D-8D, it should be fixed to 10b. It should not be configured both command size and address size to zero. 0 0: 0 bytes (No command phase) 0 1: 1 byte 1 0: 2 bytes 1 1: Setting prohibited

28.3.2.16 xSPI Command Calibration Control Register 2 CSn (CCCTL2CSn) (n = 0, 1)

This register has functions to control xSPI Master function. It should be updated while Automatic Calibration Enable is disabled.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CARDCMD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CAWRCMD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	CARDCMD [15:0]	All 0	R/W	Calibration pattern read command These bits configure the calibration pattern read command.
15 to 0	CAWRCMD [15:0]	All 0	R/W	Calibration pattern write command These bits configure the calibration pattern write command.

28.3.2.17 xSPI Command Calibration Control Register 3 CSn (CCCTL3CSn) (n = 0, 1)

This register has functions to control xSPI Master function. It should be updated while Automatic Calibration Enable is disabled.

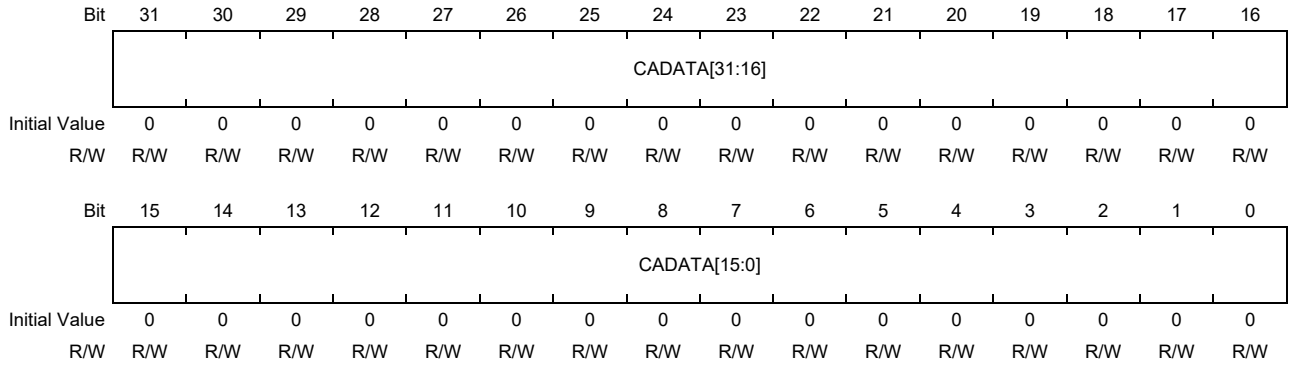
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAADD[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CAADD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CAADD[31:0]	All 0	R/W	Calibration pattern address These bits configure the calibration pattern address.

28.3.2.18 xSPI Command Calibration Control Register 4 CSn (CCCTL4CSn) (n = 0, 1)

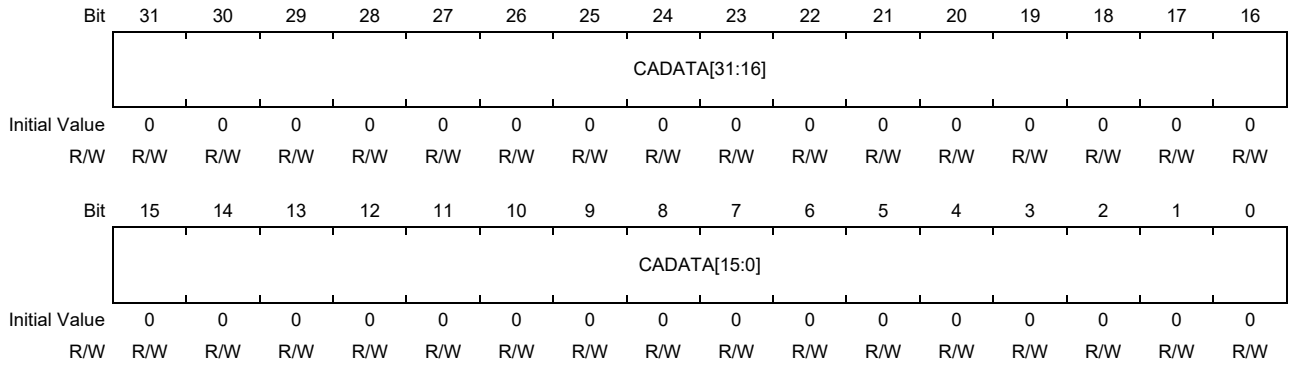
This register has functions to control xSPI Master function. It should be updated while Automatic Calibration Enable is disabled.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CADATA [31:0]	All 0	R/W	Calibration pattern data These bits configure the calibration pattern data.

28.3.2.19 xSPI Command Calibration Control Register 5 CSn (CCCTL5CSn) (n = 0, 1)

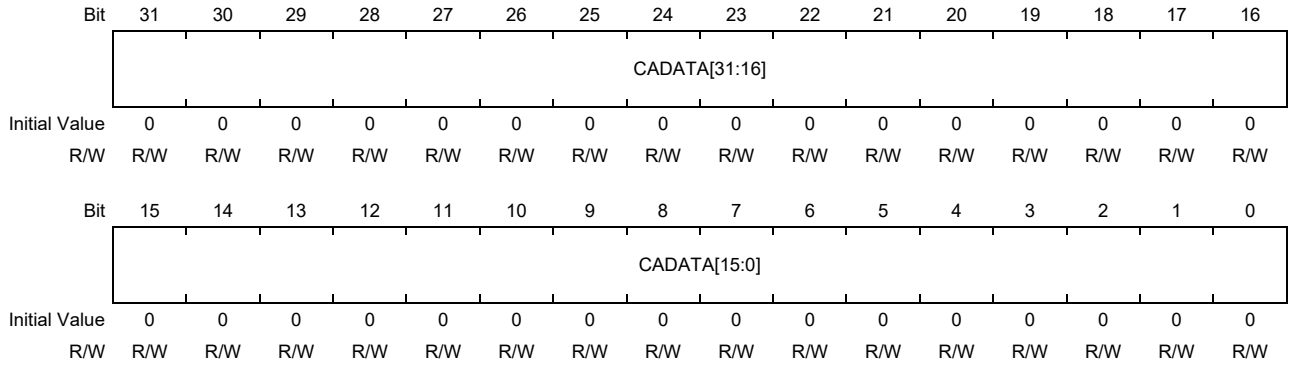
This register has functions to control xSPI Master function. It should be updated while Automatic Calibration Enable is disabled.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CADATA [31:0]	All 0	R/W	Calibration pattern data These bits configure the calibration pattern data.

28.3.2.20 xSPI Command Calibration Control Register 6 CSn (CCCTL6CSn) (n = 0, 1)

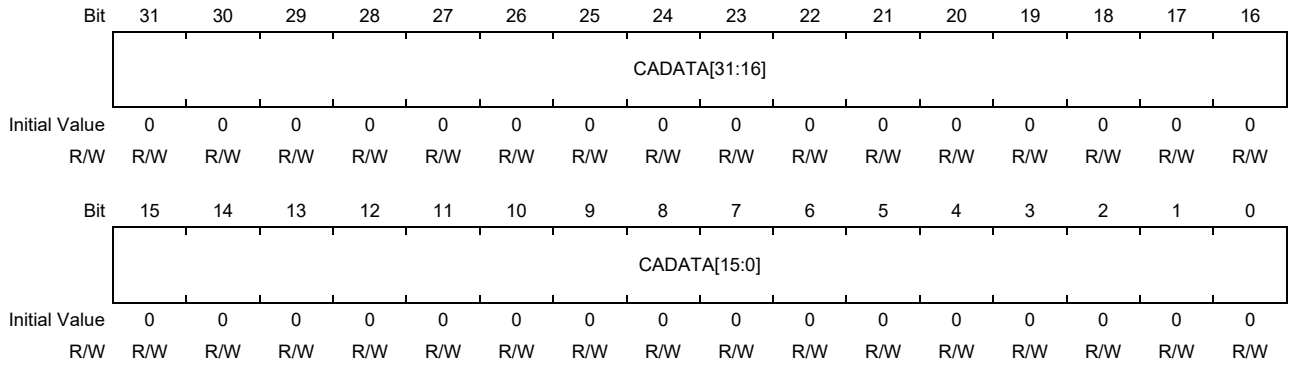
This register has functions to control xSPI Master function. It should be updated while Automatic Calibration Enable is disabled.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CADATA [31:0]	All 0	R/W	Calibration pattern data These bits configure the calibration pattern data.

28.3.2.21 xSPI Command Calibration Control Register 7 CSn (CCCTL7CSn) (n = 0, 1)

This register has functions to control xSPI Master function. It should be updated while Automatic Calibration Enable is disabled.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CADATA [31:0]	All 0	R/W	Calibration pattern data These bits configure the calibration pattern data.

28.3.3 xSPI Status Registers

These registers monitor the status of xSPI Master.

28.3.3.1 xSPI Version Register (VERSTT)

This register indicates the status of xSPI Master.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VER[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VER[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	VER[31:0]	All 0	R	Version These bits indicate this IP version.

28.3.3.2 xSPI Common Status Register (COMSTT)

This register indicates the status of xSPI Master.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	WRBUFNE	—	PBUFNE	—	—	—	MEMACC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
6	WRBUFNE	0b	R	Write Buffer Not Empty 0: Empty 1: Not empty
5	—	0b	R	Reserved This bit is read as 0.
4	PBUFNE	0b	R	Prefetch Buffer Not Empty 0: Empty 1: Not empty
3 to 1	—	000b	R	Reserved These bits are read as 0.
0	MEMACC	0b	R	Memory access ongoing 0: System bus channel is not accessing to memory. 1: System bus channel is accessing to memory.

28.3.3.3 xSPI Calibration Status Register CSn (CASTTCSn) (n = 0, 1)

This register indicates the status of xSPI Master.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CASUC[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CASUC[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CASUC[31:0]	All 0	R	Calibration Success These bits indicate the calibration success for each DS shift value. It is updated when completed each calibration sequence. CASUC[x] indicates calibration success in DS shift value = x.

28.3.4 xSPI Interrupt Registers

These registers control the interrupt function of xSPI Master.

28.3.4.1 xSPI Interrupt Status Register (INTS)

This register indicates the status of interrupt. The bits in this register are cleared to 0 when writing 1 on the corresponding bit of INTC register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CASUCCS1	CASUCCS0	CAFAILCS1	CAFAILCS0	—	—	—	—	—	—	—	BUSERR	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	DSTOCS1	DSTOCS0	PERTO	INICMP	PATCMP	CMDCMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	CASUCCS1	0b	R	Calibration success for slave1 This function is same as CASUCCS0. 0: No detection 1: Detection
30	CASUCCS0	0b	R	Calibration success for slave0 This bit is set to 1 when success calibration. 0: No detection 1: Detection
29	CAFAILCS1	0b	R	Calibration failed for slave1 This function is same as CAFAILCS0. 0: No detection 1: Detection
28	CAFAILCS0	0b	R	Calibration failed for slave0 This bit is set to 1 when failed calibration. 0: No detection 1: Detection
27 to 21	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
20	BUSERR	0b	R	System bus error This field is set to “1” when system bus responds error.
19 to 6	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
5	DSTOCS1	0b	R	DS timeout for slave1 This function is same as DSTOCS0. 0: No detection 1: Detection

Bit	Bit Name	Initial Value	R/W	Description
4	DSTOCS0	0b	R	DS timeout for slave0 This bit is set to 1 when lost DS in read transaction with using DS. It means not receiving the data during expected read phase. In this case, xSPI master stops the read transaction and the following transaction. 0: No detection 1: Detection
3	PERTO	0b	R	Periodic transaction timeout This bit is set to 1 when the read value does not match with the expected value in periodic manual-command mode. 0: No detection 1: Detection
2	INICMP	0b	R	Initial Sequence Completed This bit is set to 1 when completed the initial sequence. 0: No detection 1: Detection
1	PATCMP	0b	R	Pattern Completed This bit is set to 1 when completed the requested pattern. 0: No detection 1: Detection
0	CMDCMP	0b	R	Command Completed This bit is set to 1 when completed the requested manual-command. In direct manual-command, it means all transactions completed. In periodic manual-command, it means the read data matches with the expected data. 0: No detection 1: Detection

28.3.4.2 xSPI Interrupt Clear Register (INTC)

This register clears the status of interrupt.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CASUCCS1C	CASUCCS0C	CAFAILCS1C	CAFAILCS0C	—	—	—	—	—	—	—	BUSERRCH0C	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	R	R	R	R	R	R	R	W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	DSTOCS1C	DSTOCS0C	PERTOC	INICMPC	PATCMPC	CMDCMPC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	W	W	R	R	W	W	R	R	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31	CASUCCS1C	0b	W	Calibration success for slave1 interrupt clear 0: No change interrupt status 1: Clear interrupt status
30	CASUCCS0C	0b	W	Calibration success for slave0 interrupt clear 0: No change interrupt status 1: Clear interrupt status
29	CAFAILCS1C	0b	W	Calibration failed for slave1 interrupt clear 0: No change interrupt status 1: Clear interrupt status
28	CAFAILCS0C	0b	W	Calibration failed for slave0 interrupt clear 0: No change interrupt status 1: Clear interrupt status
27 to 21	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
20	BUSERRC	0b	W	System bus error for interrupt clear 0: No change interrupt status 1: Clear interrupt status
19 to 6	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
5	DSTOCS1C	0b	W	DS timeout for slave1 interrupt clear 0: No change interrupt status 1: Clear interrupt status
4	DSTOCS0C	0b	W	DS timeout for slave0 interrupt clear 0: No change interrupt status 1: Clear interrupt status
3	PERTOC	0b	W	Periodic transaction timeout interrupt clear 0: No change interrupt status 1: Clear interrupt status
2	INICMPC	0b	W	Initial Sequence Completed interrupt clear 0: No change interrupt status 1: Clear interrupt status
1	PATCMPC	0b	W	Pattern Completed interrupt clear 0: No change interrupt status 1: Clear interrupt status

Bit	Bit Name	Initial Value	R/W	Description
0	CMDCMPC	0b	W	Command Completed interrupt clear 0: No change interrupt status 1: Clear interrupt status

28.3.4.3 xSPI Interrupt Enable Register (INTE)

This register enables the interrupt.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CASUCCS1E	CASUCCS0E	CAFAILCS1E	CAFAILCS0E	—	—	—	—	—	—	—	BUSERRE	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	DSTOCS1E	DSTOCS0E	PERTOE	INICMPE	PATCMPE	CMDCMPE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	CASUCCS1E	0b	R/W	Calibration success for slave1 interrupt enable 0: Disabled 1: Enabled
30	CASUCCS0E	0b	R/W	Calibration success for slave0 interrupt enable 0: Disabled 1: Enabled
29	CAFAILCS1E	0b	R/W	Calibration failed for slave1 interrupt enable 0: Disabled 1: Enabled
28	CAFAILCS0E	0b	R/W	Calibration failed for slave0 interrupt enable 0: Disabled 1: Enabled
27 to 21	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
20	BUSERRE	0b	R/W	System bus error for interrupt enable 0: Disabled 1: Enabled
19 to 6	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
5	DSTOCS1E	0b	R/W	DS timeout for slave1 interrupt enable 0: Disabled 1: Enabled
4	DSTOCS0E	0b	R/W	DS timeout for slave0 interrupt enable 0: Disabled 1: Enabled
3	PERTOE	0b	R/W	Periodic transaction timeout interrupt enable 0: Disabled 1: Enabled
2	INICMPE	0b	R/W	Initial Sequence Completed interrupt enable 0: Disabled 1: Enabled
1	PATCMPE	0b	R/W	Pattern Completed interrupt enable 0: Disabled 1: Enabled

Bit	Bit Name	Initial Value	R/W	Description
0	CMDCMPE	0b	R/W	Command Completed interrupt enable 0: Disabled 1: Enabled

28.4 Operation

xSPI Master interface has the functions to issue the transaction for external memory with xSPI Slave interface. It allows to write to registers in external memory or read from it.

This xSPI Master mainly has two modes to issue the transaction. One is a manual-command mode; Software configures all fields of xSPI frame and starts the transaction by software request. The other is a memory-mapping mode; it automatically converts system bus access for pre-configured memory area into xSPI transaction. It enables to access from system bus to external memory area outside of chip via xSPI bus.

This section describes the xSPI bus operation, the direct control of xSPI frame (manual-command), the control of memory access (memory-mapping), the error operation, and the flow to operation.

28.4.1 xSPI Bus

This section describes the xSPI bus operation.

28.4.1.1 Supported Protocol Mode

This xSPI Master supports various protocol modes. It is configured by Protocol mode bits (LIOCFGCSn.PRTMD[9:0]). The table below shows the summary of protocol modes.

Table 28.5 Supported Protocol Mode

Protocol Mode	Function	PRTMD [9:0]	Note
1S-1S-1S	Command, Address, and Data fields are transferred at SDR with using 1 pin. Read data is sampled with CK.	H'000	Specified by xSPI protocol
8D-8D-8D	Command, Address, and Data fields are transferred at DDR with using 8 pins. Read data is sampled with DS.	H'3FF	Specified by xSPI protocol
1S-4S-4S	Command field is transferred at SDR with using 1 pin. Address and Data fields are transferred at SDR with using 4 pins. Read data is sampled with CK.	H'090	—
4S-4S-4S	Command, Address, and Data fields are transferred at SDR with using 4 pins. Read data is sampled with CK.	H'092	—

Note: In case of XiP mode enable, XiP code is inserted in Latency field. It is valid only for memory-mapping mode.

The table below shows a description of the main internal signals.

Table 28.6 Internal Signals

Signal Name	I/O	Function
clk_spi	I	xSPI control clock
spi_ck	O	xSPI Clock (Same as XSPI_SPCLK pin)
spi_cs0	O	xSPI Chip Select for slave0 (Same as XSPI_CS0# pin)
spi_cs1	O	xSPI Chip Select for slave1 (Same as XSPI_CS1# pin)
spi_doe[7:0]	O	xSPI Data Output Enable
spi_do[7:0]	O	xSPI Data Output
spi_di[7:0]	I	xSPI Data Input
spi_dsoe	O	xSPI Data Strobe Output Enable
spi_dso	O	xSPI Data Strobe Output
spi_dsi	I	xSPI Data Strobe Input

The bytes of Command and Address fields are transferred in highest order to lowest order sequence. The sequential bytes of Data field are transferred in lowest address to highest address order. In case of using multiple pins, the least significant bit of each byte is placed on spi_do/di[0] with each higher order bit on the successively higher numbered spi_do/di signals.

The figure below shows timing-chart for 1S-1S-1S protocol mode. The spi_do[0] signal is used for output data and the spi_di[1] signal is used for input data.

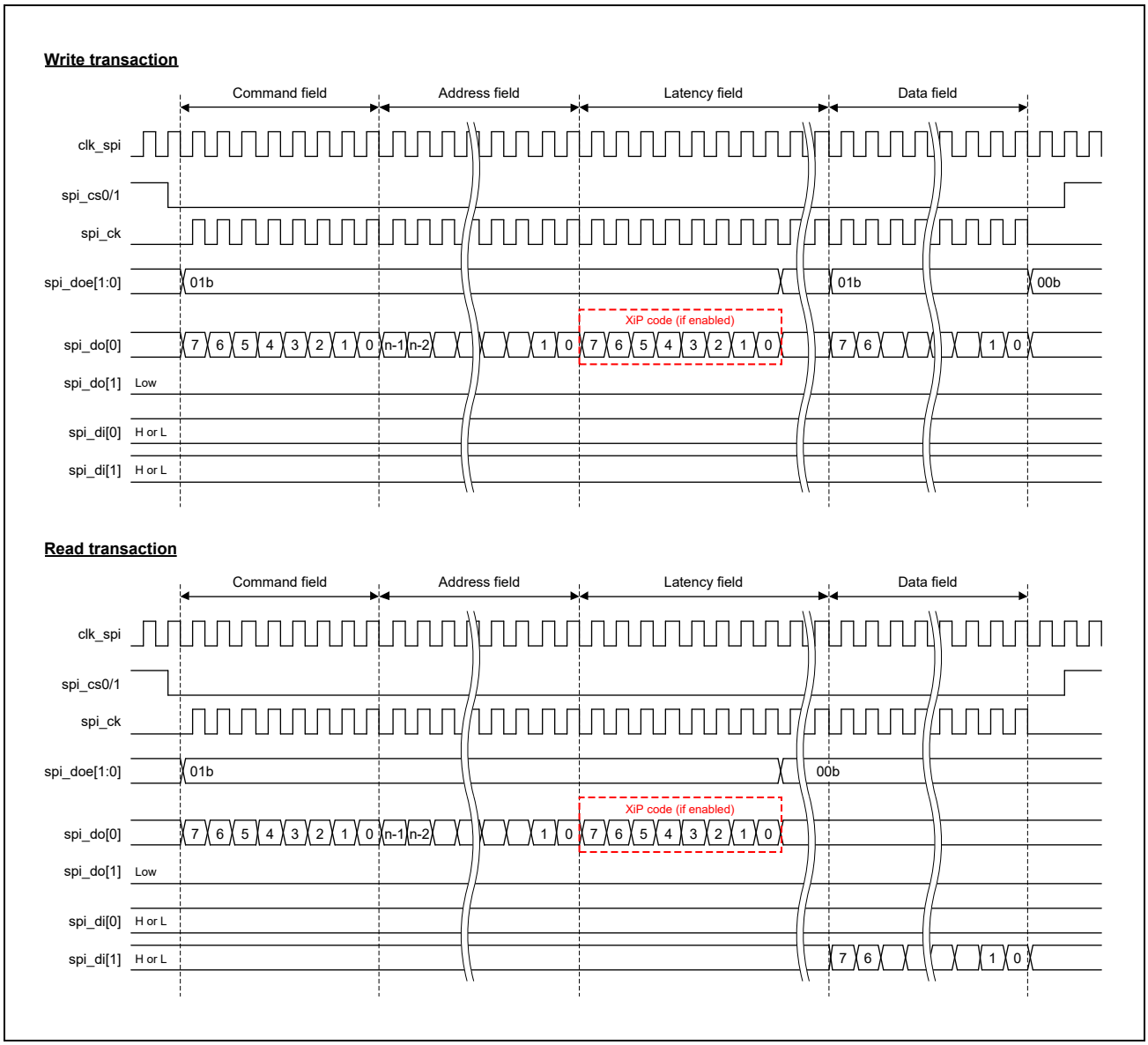


Figure 28.2 1S-1S-1S Timing-Chart

The figure below shows timing-chart for 8D-8D-8D profile 1.0 protocol mode. In 8D-8D-8D, the data is always transmitted with byte-pair. In case each field is odd byte, last one byte is padded with invalid data.

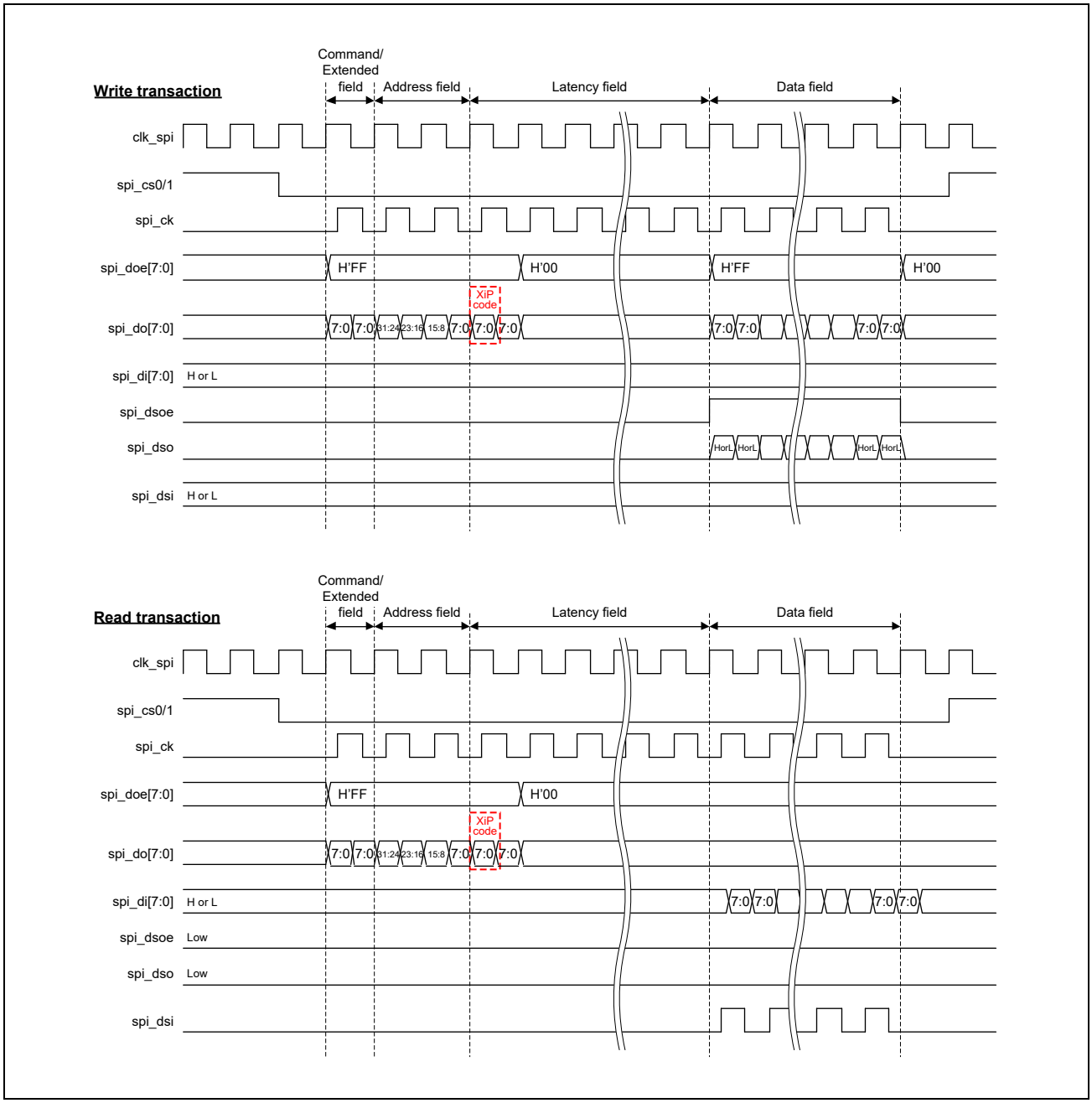


Figure 28.3 8D-8D-8D profile 1.0 Timing-Chart

The figure below shows timing-chart for 8D-8D-8D profile 2.0 protocol mode. In 8D-8D-8D, the data is always transmitted with byte-pair. In case each field is odd byte, last one byte is padded with invalid data.

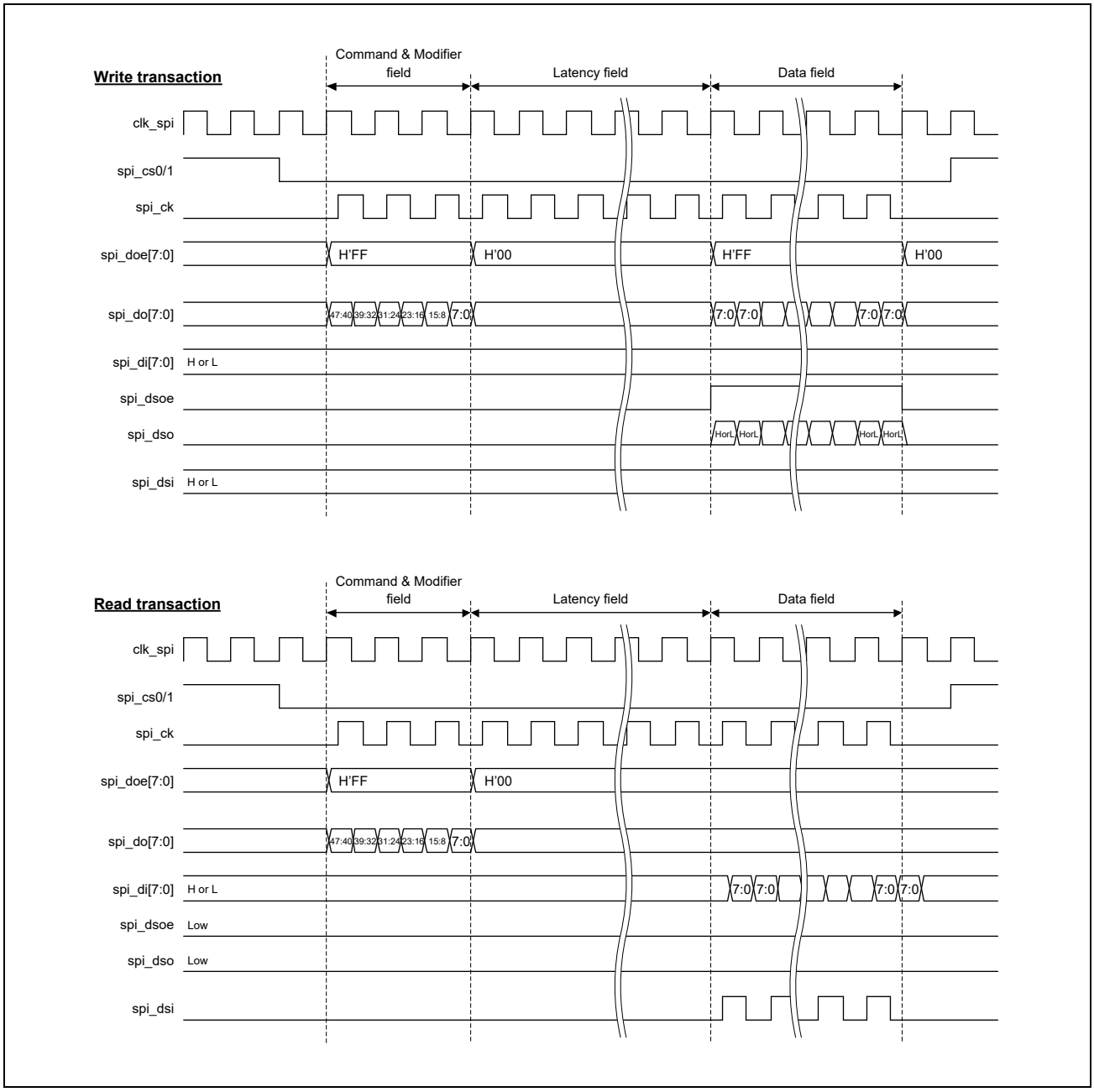


Figure 28.4 8D-8D-8D Profile 2.0 Timing-Chart

28.4.1.2 xSPI Frame Interval

The interval between xSPI frames is configured with CS minimum idle term bits (LIOCFGCSn.CSMIN[3:0]). It depends on the specification of xSPI slave device.

28.4.1.3 xSPI Signals Timing Control

This xSPI Master supports both SDR and DDR. And it is possible to sample input data with Data-Strobe (DS) signal at SDR. For the various mode support and easy implementation, this xSPI Master could adjust the timing to drive/sample xSPI interface signals statically. The table below shows the summary of xSPI Interface signal timing control.

Table 28.7 Summary of xSPI Signals Timing Control

Signal	Mode	Default Operation	Timing Control (n = 0, 1)
CS (spi_cs) drive	—	Asserting 1 cycle before the rising-edge of first spi_ck	1 cycle extension for asserting with LIOCFGCSn.CSASTEX bit
		Negating 1.5 cycle after the falling-edge of last spi_ck	1 cycle extension for negating with LIOCFGCSn.CSNEGEX bit
CK (spi_ck) drive	SDR without DS	Reference point	—
	SDR with DS		
	DDR with DS	Reference point	0 to 1 cycle phase shift with WRAPCFG.CKSFTCSx bit
DO/DOE (spi_do/doe) drive	SDR without DS	Falling edge of clk_spi	0 or 0.5 cycle shift with LIOCFGCSn.SDRDRV bit
	SDR with DS		
	DDR with DS	Both edges of clk_spi	—
DI (spi_di) sample	SDR without DS	Falling edge of spi_ck on expected data size	0 to 7 cycle shift (1 cycle unit) with LIOCFGCSn.SDRSMPSFT[3:0] bits 0 or 0.5 cycle shift with LIOCFGCSn.SDRSMPMD bit
	SDR with DS	Falling edge of spi_dsi signal on expected data size	Sample at rising edge with LIOCFGCSn.SDRSMPMD bit 0 to 1 cycle phase shift with WRAPCFG.DSSFTCSn[4:0] bits 0 to 7 cycle extension with LIOCFGCSn.DDRSMPEX[3:0] bits
	DDR with DS	Both edges of spi_dsi signal on expected data size	0 to 1 cycle phase shift with WRAPCFG.DSSFTCSn[4:0] bits 0 to 7 cycle extension with LIOCFGCSn.DDRSMPEX[3:0] bits

Note: In DDR on xSPI protocol, CK or DS should be aligned for center of data. It means to shift the phase by 0.25 cycle (90 degrees). This xSPI master supports to adjust this phase depending on the usage conditions.

The figure below shows the default operation and timing control for SDR without DS.

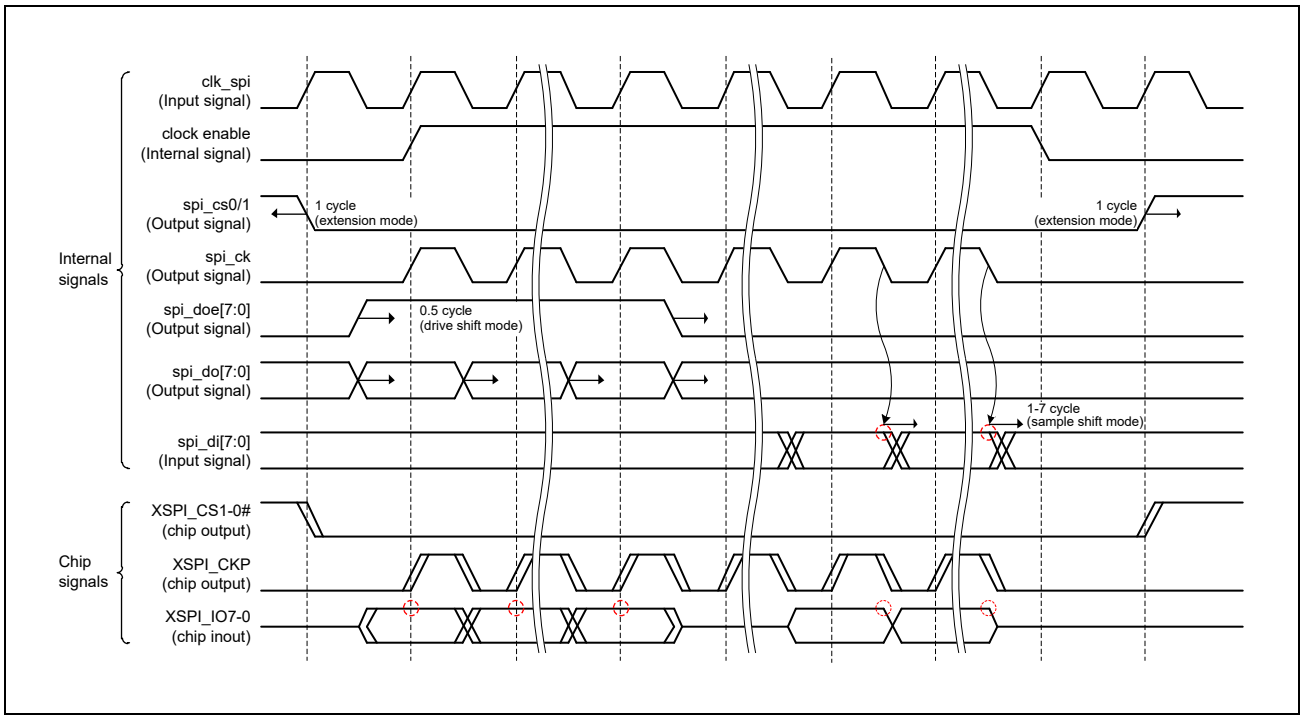


Figure 28.5 Timing Control for SDR without DS

The figure below shows the default operation and timing control for SDR with DS.

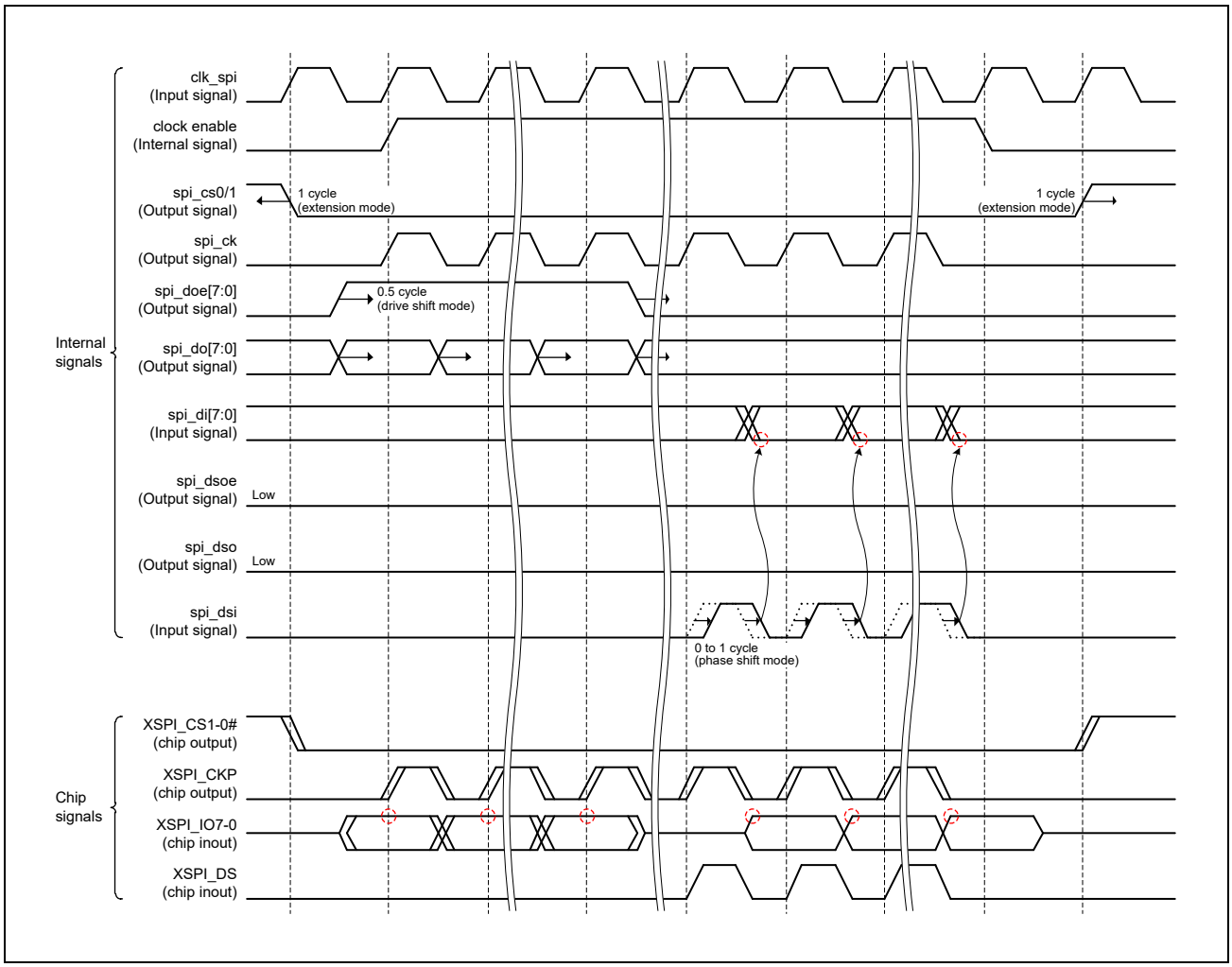


Figure 28.6 Timing Control for SDR with DS

The figure below shows the default operation and timing control for DDR with DS.

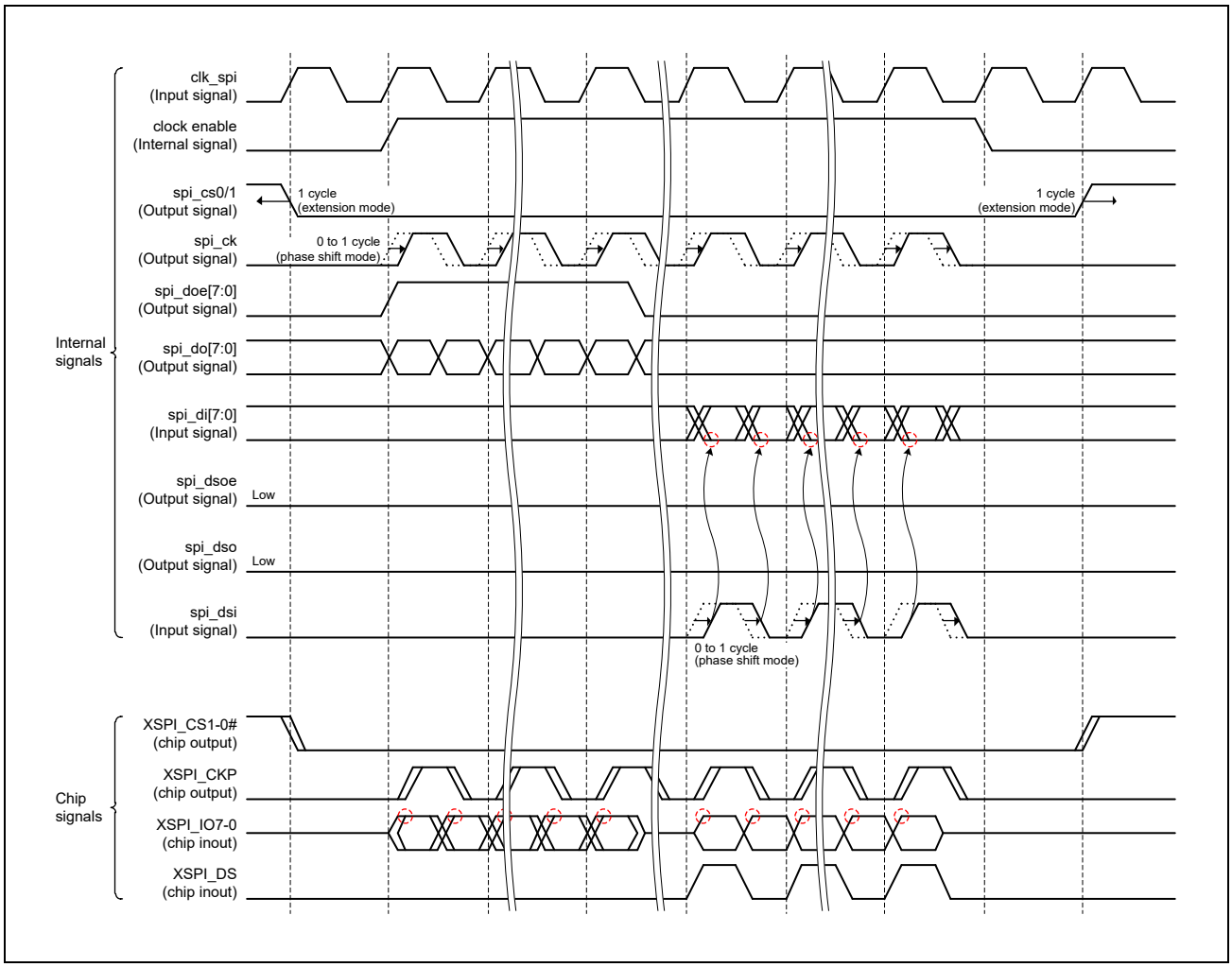


Figure 28.7 Timing Control for DDR with DS

28.4.2 Manual-command

This section describes the manual-command mode. The manual-command has two functional modes: direct mode and periodic mode.

28.4.2.1 Direct Mode

This mode sequentially can issue up to four xSPI transactions configured and requested by Software. A series of transaction can be issued by a transaction request (CDCTL0.TRREQ = 1 with PERMD = 0). The number of transactions (CDCTL0.TRNUM[1:0]) can be configured up to 4. It can be used to change the mode or read the status of xSPI slave device.

The table below shows the configured register bits for direct manual-command. The operating flow is illustrated in **Figure 28.20**.

Table 28.8 Manual-Command Configuration for Direct Mode

Transaction	Transaction Type	Command	Command Size	Address	Address Size	Data (x = 0, 1)	Data Size	Latency Cycle
1st Transaction	CDTBUF0.TRTYPE	CDTBUF0.CMD[15:0]	CDTBUF0.CMDSIZE[1:0]	CDABUF0.ADD[31:0]	CDTBUF0.ADDSIZE[2:0]	CDDxBUF0.DATA[31:0]	CDTBUF0.DATASIZE[3:0]	CDTBUF0.LATE[4:0]
2nd Transaction	CDTBUF1.TRTYPE	CDTBUF1.CMD[15:0]	CDTBUF1.CMDSIZE[1:0]	CDABUF1.ADD[31:0]	CDTBUF1.ADDSIZE[2:0]	CDDxBUF1.DATA[31:0]	CDTBUF1.DATASIZE[3:0]	CDTBUF1.LATE[4:0]
3rd Transaction	CDTBUF2.TRTYPE	CDTBUF2.CMD[15:0]	CDTBUF2.CMDSIZE[1:0]	CDABUF2.ADD[31:0]	CDTBUF2.ADDSIZE[2:0]	CDDxBUF2.DATA[31:0]	CDTBUF2.DATASIZE[3:0]	CDTBUF2.LATE[4:0]
4th Transaction	CDTBUF3.TRTYPE	CDTBUF3.CMD[15:0]	CDTBUF3.CMDSIZE[1:0]	CDABUF3.ADD[31:0]	CDTBUF3.ADDSIZE[2:0]	CDDxBUF3.DATA[31:0]	CDTBUF3.DATASIZE[3:0]	CDTBUF3.LATE[4:0]

28.4.2.2 Periodic Mode

This mode periodically issues an xSPI read transaction configured and requested by Software. And it can compare the read value up to 4 bytes with expected value. The transaction is issued by a transaction request (CDCTL0.TRREQ = 1 with PERMD = 1). It can be used to alternate the status polling operation of xSPI slave device.

The periodic term is configured in Periodic transaction interval bits (CDCTL0.PERITV[4:0]). The expected value is configured in Periodic transaction expected and masked value bits (CDCTL1.PEREXP[31:0] and CDCTL2.PERMSK[31:0]). The table below shows the configured register bits for periodic manual-command. The operating flow is illustrated in **Figure 28.21**.

Table 28.9 Manual-Command Configuration for Periodic Mode

Transaction	Transaction Type	Command	Command Size	Address	Address Size	Data (x = 0, 1)	Data Size	Latency Cycle
Read Transaction	CDTBUF0.TRTYPE = 0	CDTBUF0.CMD[15:0]	CDTBUF0.CMDSIZE[1:0]	CDABUF0.ADD[31:0]	CDTBUF0.ADDSIZE[2:0]	CDDxBUF0.DATA[31:0]	CDTBUF0.DATASIZE[3:0]	CDTBUF0.LATE[4:0]

28.4.3 Memory-mapping

This section describes the memory-mapping mode. This mode automatically converts system bus access for pre-configured memory area into xSPI transaction.

28.4.3.1 Configuration

In this operation, the payload of address and data field are delivered from system bus signals. The information of command field and size are delivered from the configured register bits. The table below shows the register bits configured for memory- mapping.

Table 28.10 Memory-Mapping Configuration for Memory Area Access (n = 0, 1)

AHB Transaction	Format Change Mode	Command	Command Size	Address Size	Latency Cycle
Write for slave n memory area	Normal	CMCFG2CSn.WRCMD [15:8]	1 byte	CMCFG0CSn.ADDSIZE[1:0]	CMCFG2CSn.WRLATE[4:0]
	8D-8D-8D profile 1.0	CMCFG2CSn.WRCMD [15:0]	2 bytes		
	8D-8D-8D profile 2.0 Command Modifier	CMCFG2CSn.WRCMD [15:8]	1 byte	5 bytes	
	8D-8D-8D profile 2.0 Extended Command Modifier	CMCFG2CSn.WRCMD [15:13]	3 bits	45 bits	
Read for slave n memory area	Normal	CMCFG1CSn.RDCMD [15:8]	1 byte	CMCFG0CSn.ADDSIZE[1:0]	CMCFG1CSn.RDLATE[4:0]
	8D-8D-8D profile 1.0	CMCFG1CSn.RDCMD [15:0]	2 bytes		
	8D-8D-8D profile 2.0 Command Modifier	CMCFG1CSn.RDCMD [15:8]	1 byte	5 bytes	
	8D-8D-8D profile 2.0 Extended Command Modifier	CMCFG1CSn.RDCMD [15:13]	3 bits	45 bits	

Note: The MSByte of Address can be replaced with Address Replace Enable and Code bits (CMCFG0CSn.ADDRPEN[7:0]/ADDRPCD[7:0]).

28.4.3.2 Write Access Operation

At accepting write access for memory area from system bus, this xSPI Master stores all payload data in internal bridge buffer and then issues a write transaction into xSPI slave. The figure below shows the operation summary.

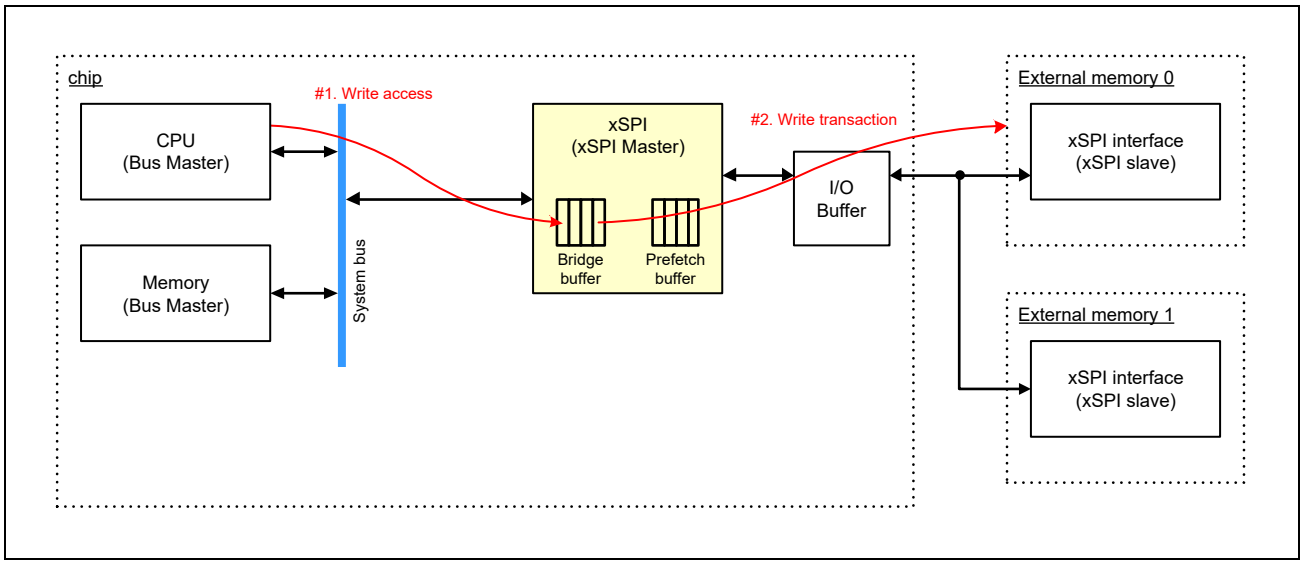


Figure 28.8 Write Access for Memory Area

The operation of xSPI bus changes depending on System bus's burst type. When HBURST is SINGLE or INCR, one AHB transaction occurs for one xSPI frame. When HBURST is INCR4, INCR8, or INCR16, AHB transactions equal to burst size occur for one xSPI frame. When HBURST is WRAP4, WRAP8, or WRAP16, AHB transactions occur for two xSPI frames. The figure below shows the relationship between AHB and xSPI frames.

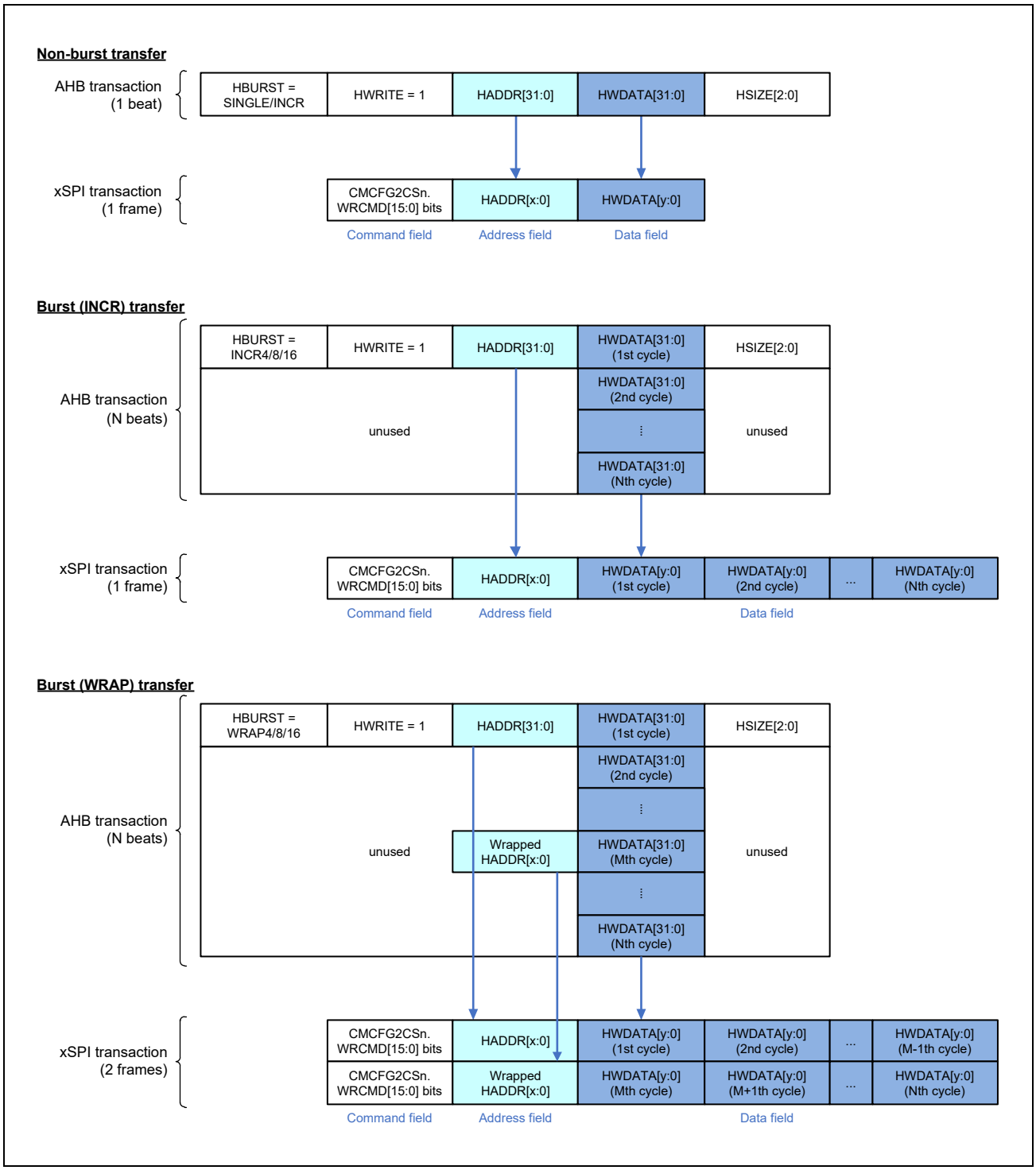


Figure 28.9 xSPI Frame Format in Write Access (Normal Format)

28.4.3.3 Combination Function

At the system bus write access for memory area, this xSPI master has the function to combinate the write data for high throughput on xSPI bus. When this function is enabled (BMCFG.MWRCOMB = 1), this xSPI master transmits a xSPI frame with the selected size while the sequential address is incremental. When non-incremental address access is detected, or when the access for different slave is detected, or when a read transaction is detected, even though reaching to the target size (BMCFG.MWRSIZE), this xSPI master transmits the pending data into xSPI bus. And also, when Memory Write Data Push bit (BMCTL1.MWRPUSH) is set, this xSPI master transmits the pending data into xSPI bus.

This function could be useful for any slave device to request a chunk of data at a time. In the case, system bus master shall continue to provide the fixed data size with incremental address. Ex. there is any device to request to write in page unit. The figure below shows the operation when enable the combination function.

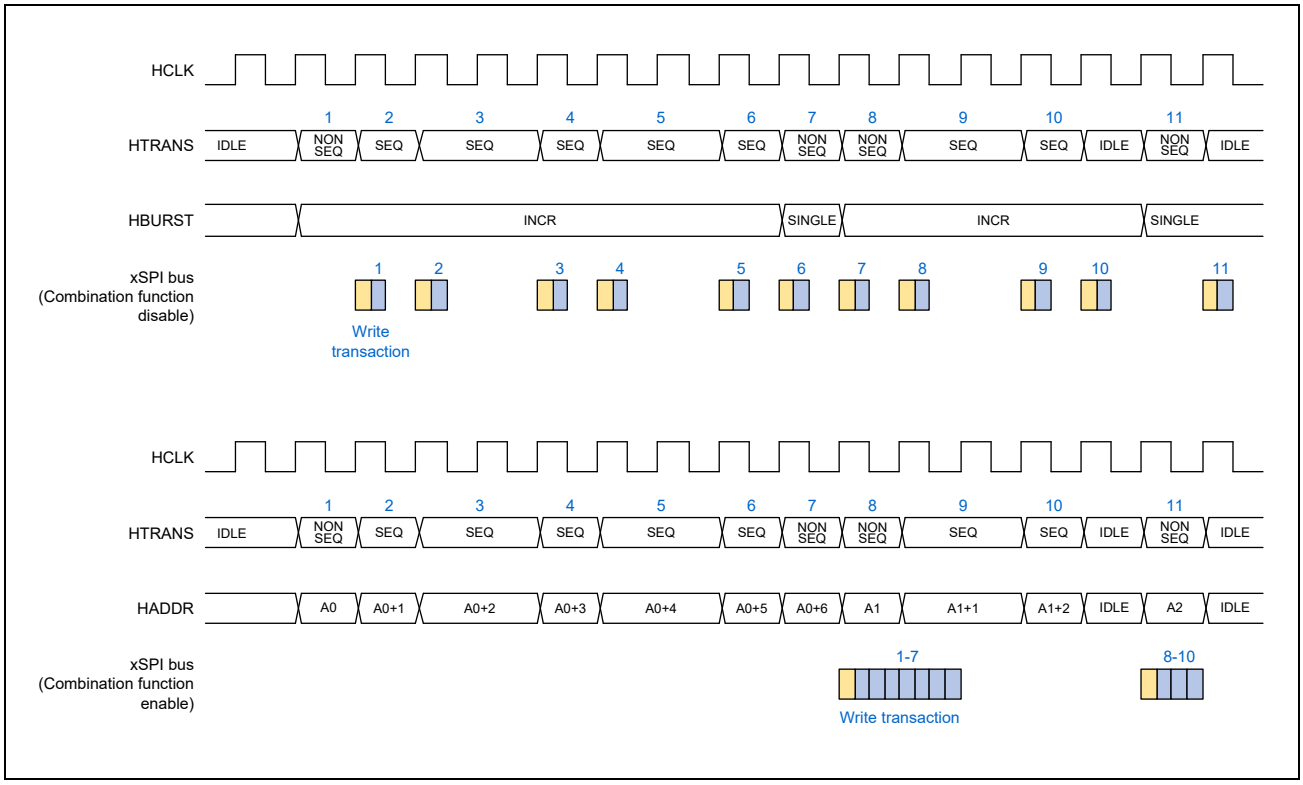


Figure 28.10 Combination Function

28.4.3.4 Read Access Operation

At the read access for memory area, soon after detected the read access, This xSPI Master issues a read transaction into xSPI slave. The figure below shows the operation summary.

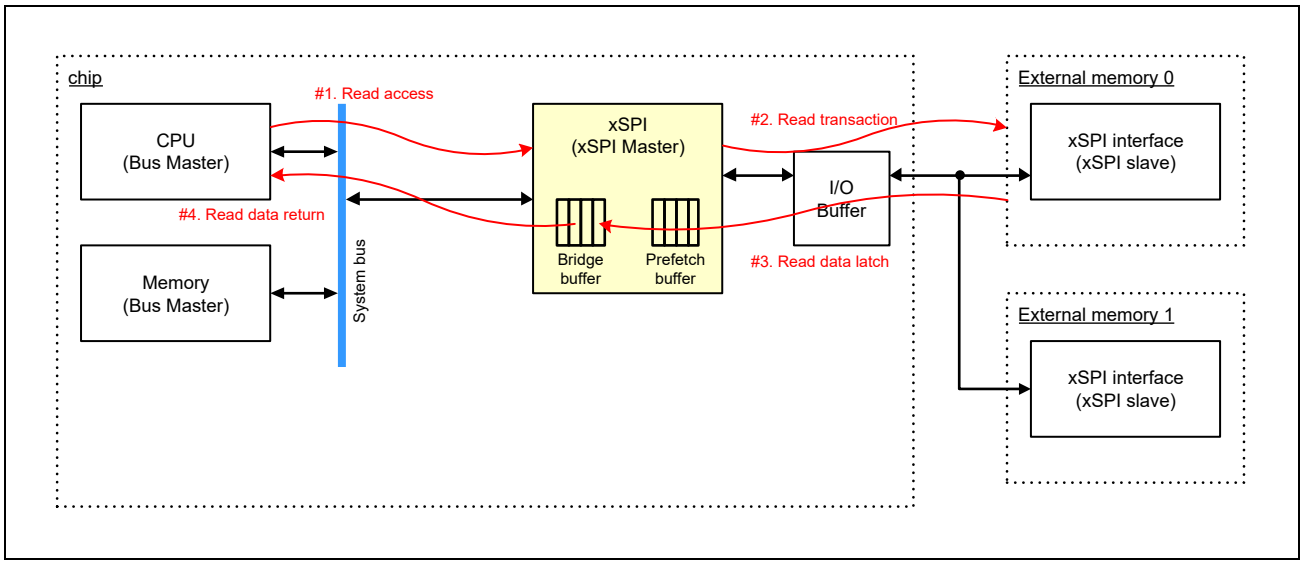


Figure 28.11 Read Access for Memory Area

The operation of xSPI bus changes depending on burst type. When the type is single or increment type, one system bus's read transaction triggers one xSPI frame. When the type is wrap type and the CMCFG0CSx.WPBSTMD is 0, one system bus read transactions triggers two xSPI frame. The figure below shows the relationship between AHB and xSPI frame.

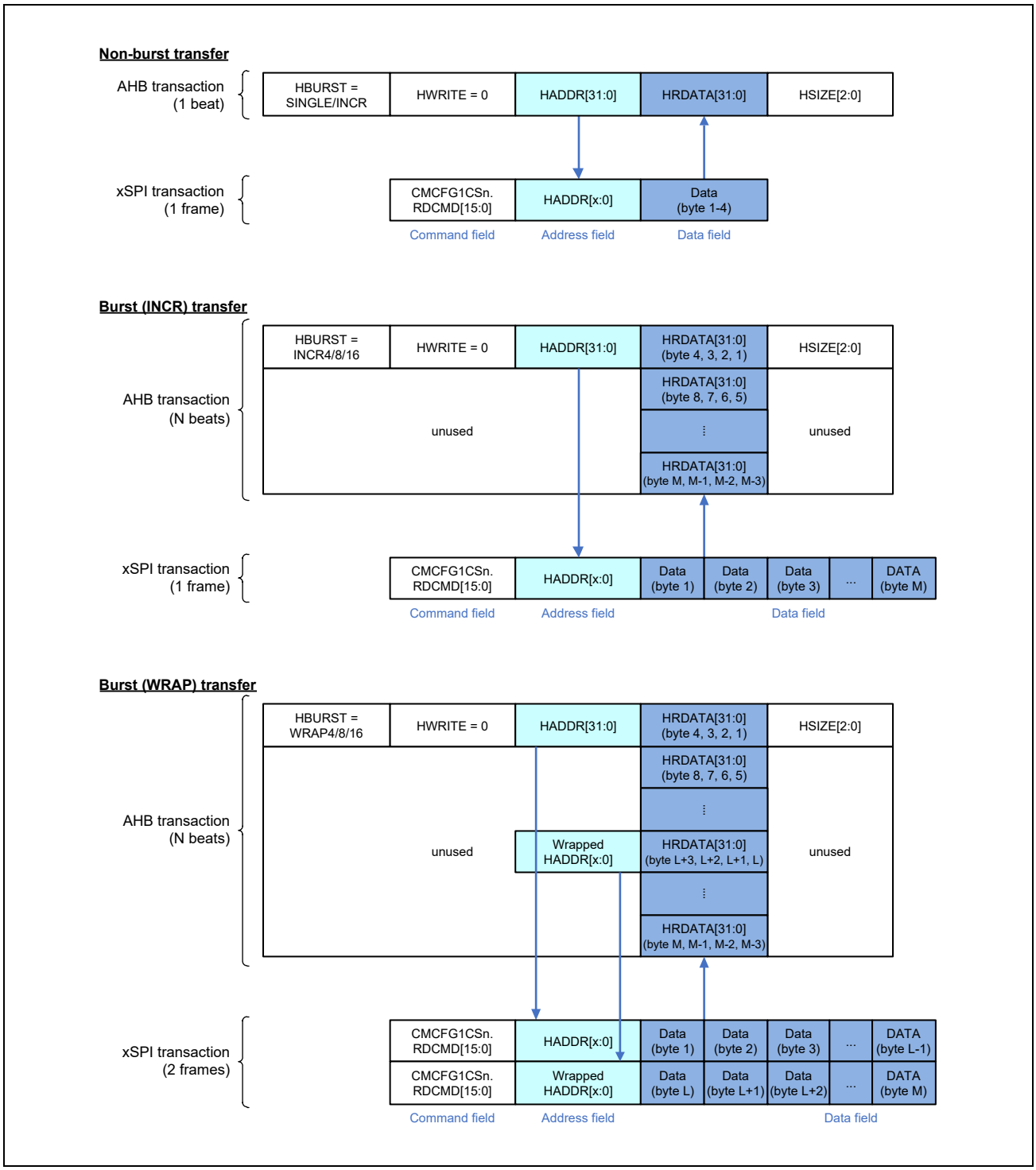


Figure 28.12 xSPI Frame Format in Read Access (Normal Format)

28.4.3.5 Prefetch Function

At the read access for memory area from system bus, this xSPI Master has the function to prefetch the read data for reducing the latency. When this function is enabled (BMCFG.PREEN = 1), this xSPI Master continues to read the incremental address and store the read data from xSPI slave in internal prefetch buffer. And this xSPI master searches in prefetch buffer for the following read access from system bus. If the target read data is found in prefetch buffer, this xSPI Master returns the data from prefetch buffer. If it is not found, this xSPI Master clears the prefetch buffer and newly issues a read transaction into xSPI slave. This function is effective in application such as the consecutive read addresses are close. But it is not effective in application such as the consecutive read addresses are not incremental because xSPI read frame for prefetch uses xSPI bus. The figure below shows the operation summary.

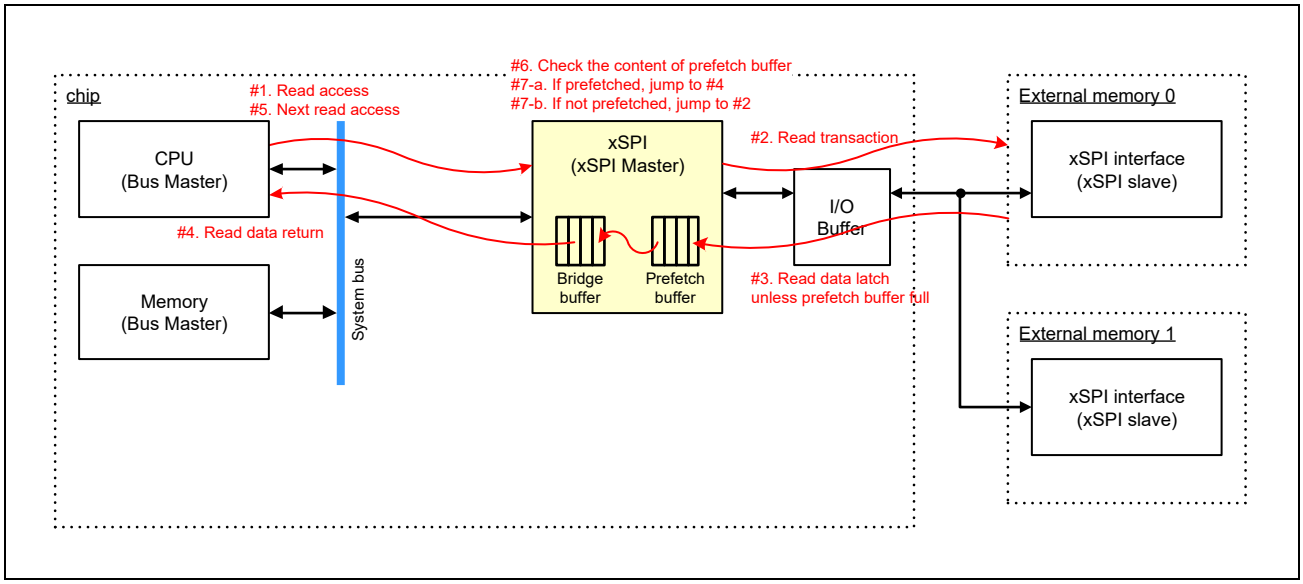


Figure 28.13 Read Access for Memory Area with Prefetch Enabled

NOTE

When this prefetch function is enabled, Bus Master could read from not a slave device but the internal prefetch buffer. When accessed to the same address from multiple Bus Masters, this xSPI Master does not guarantee to read the latest data. If Bus Master intend to read the latest data from a slave device, the Master shall clear the prefetch buffer (BMCTL1.PBUFCLR) before issuing read.

28.4.3.6 XiP Mode

Some slave devices have a mode (XiP mode) in which the command phase is not required for lower latency. While in this mode, the xSPI master skips sending the command and the slave device implicitly performs the command that was executed in the previous transaction. When enabled XiP mode bit (CMCTL.XIPEN = 1), this xSPI master inserts XiP enter code (CMCTL.XIPENCODE[7:0]) in latency field. When disabled XiP mode bit (CMCTL.XIPEN = 0), this xSPI master inserts XiP exit code (CMCTL.XIPEXCODE[7:0]) in latency field. This function is available only for memory-mapping mode.

And when this xSPI master transmits XiP disable pattern, this master clears XiP mode bit and disables XiP mode configured for both channels. Note that it is not possible to disable XiP mode for only one channel by transmitting XiP disable pattern.

NOTE

- When enough latency cycle does not exist for XiP code, this xSPI Master could not insert XiP code.
- XiP mode could be used only for unidirectional access to a slave. The write transaction and read transaction shall be separated.

28.4.4 Pattern Control

This xSPI Master has the function to transmit 3 type of patterns which is not xSPI frame format. The pattern is triggered by setting trigger bit (LPCTL0-1.PATREQ).

28.4.4.1 XiP Disable Pattern

XiP Disable pattern‘s length and value are configured by LPCTL0.XD1LEN / XD1VAL / XD2LEN / XD2VAL. It uses spi_ck, spi_doe/do signals. The number of output pin could be configured by XiP Disable pattern pin bit (LPCTL0.XDPIN). It may be used to disable XiP mode for legacy SPI. The figure below shows the timing-chart.

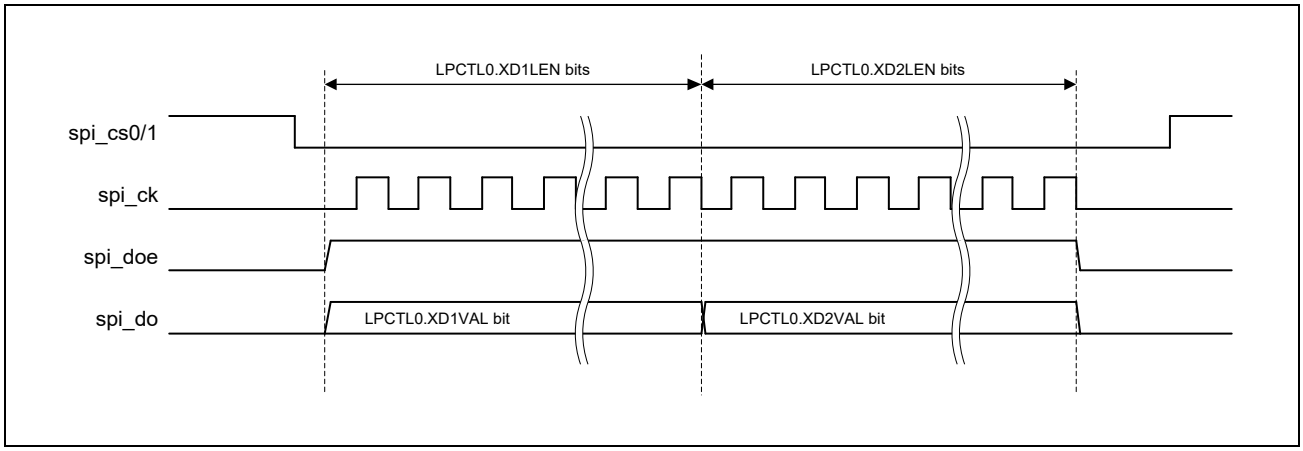


Figure 28.14 XiP Disable Pattern

NOTE

When transmitting XiP Disable pattern, spi_ck shall not be shifted (WRAPCFG.CKSFTCSx = 0).

28.4.4.2 Reset Pattern

Reset pattern is specified in Serial Flash Reset Signaling Protocol. CS Low/High width is configured with Reset Pattern Length bit (LPCTL1.RSTWID). xSPI slave will sample the data input at the rising edge of CS. Setup time for data output is configured with Reset pattern data output setup time bit (LPCTL1.RSTSU). The setup time shall be always less than Reset pattern width. The figure below shows the timing-chart.

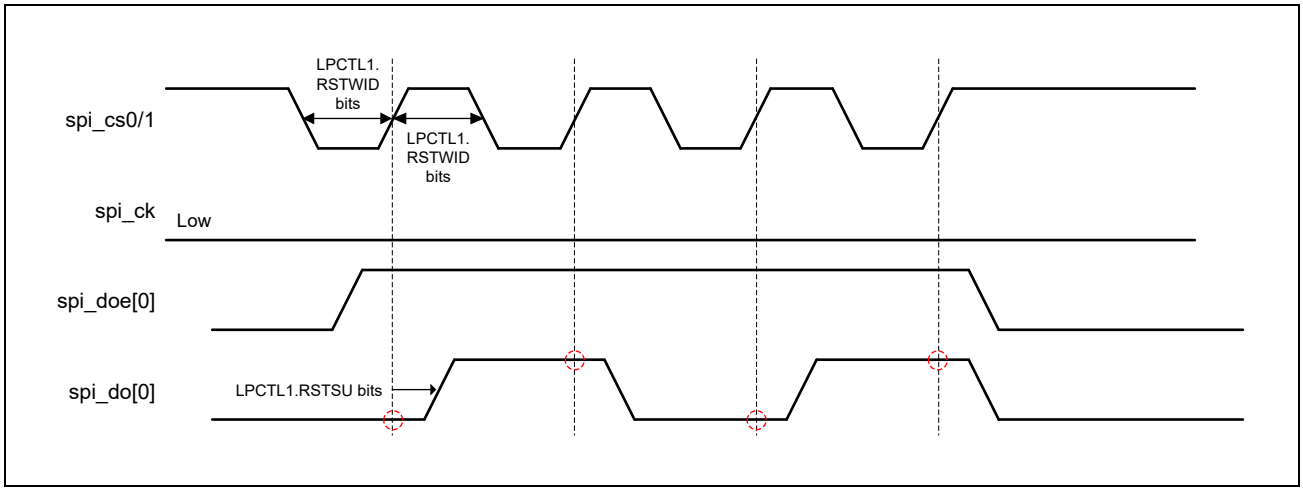


Figure 28.15 Reset Pattern

NOTE

In the protocol specification, CS Low/High width is defined as minimum 500 ns, and Setup time is defined as minimum 6 ns.

28.4.4.3 CS Only Pattern

CS Only pattern activates CS port with the configured length bit (LPCTL1.RSTWID). It may be used to resume from Deep Power Down state. The figure below shows the timing-chart.

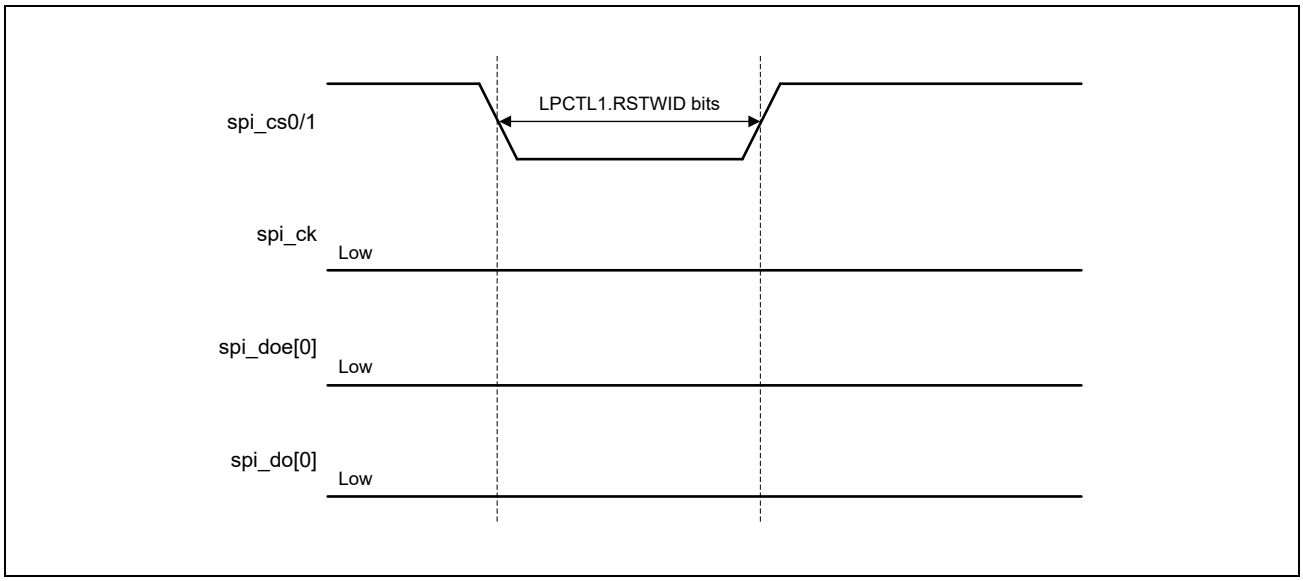


Figure 28.16 CS Only Pattern

28.4.5 Integrity Checking

This xSPI Master can detect some errors. The table below shows the error list and the detail behavior.

When reset of xSPI master is needed due to fatal error, set module reset register (MRCTLA) to reset xSPI master. After confirming it is reset surely, release module reset.

Table 28.11 Error List

Error Type	Event	Flag Bit	Note (Action)
Calibration failed	When the read data did not match the expected value during automatic calibration	INTS.CAFAILCS0-1	It results in writing unexpected data to xSPI slave.
System bus error	When an error response occurred on system bus slave interface for memory-mapping	INTS.BUSERR	This is a fatal error. This xSPI master shall be initialized. When detected this error in system bus write transaction, the internal bridge buffer is cleared. It means the pending write data is lost.
DS timeout	When DS does not toggle in read transaction with using DS	INTS.DSTOCS0-1	Both xSPI master and xSPI slave should be reset for fatal error.
Periodic transaction timeout	When the read value does not match with the expected value in periodic manual-command mode	INTS.PERTO	Depending on the status of function

28.4.6 Interrupts

This xSPI Master has an interrupt port.

It can monitor with Interrupt Status Register (INTS). In case of initialization phase, it can be programmable with Interrupt Enable register (INTE). The table below shows the related register bit.

Table 28.12 xSPI Interrupt Sources

Name	Interrupt Sources
XSPI_INT (int_spi_level)	Interrupt
XSPI_INTERR (int_spi_err_level)	Error interrupt

Table 28.13 Interrupt Register Bit

Flag Bit (INTS)	Enable Bit (INTE)	Clear Bit (INTC)	Interrupt Sources
CASUCCS1	CASUCCS1E	CASUCCS1C	XSPI_INT
CASUCCS0	CASUCCS0E	CASUCCS0C	
CAFAILCS1	CAFAILCS1E	CAFAILCS1C	XSPI_INTERR
CAFAILCS0	CAFAILCS0E	CAFAILCS0C	
BUSERR	BUSERRE	BUSERRC	XSPI_INTERR
DSTOCS1	DSTOCS1E	DSTOCS1C	XSPI_INTERR
DSTOCS0	DSTOCS0E	DSTOCS0C	
PERTO	PERTOE	PERTOC	XSPI_INTERR
INICMP	INICMPE	INICMPC	XSPI_INT
PATCMP	PATCMPE	PATCMPC	XSPI_INT
CMDCMP	CMDCMPE	CMDCMPC	XSPI_INT

28.4.7 Flows of Operations

28.4.7.1 Flow of Configuration

The figure below shows flow of configuration.

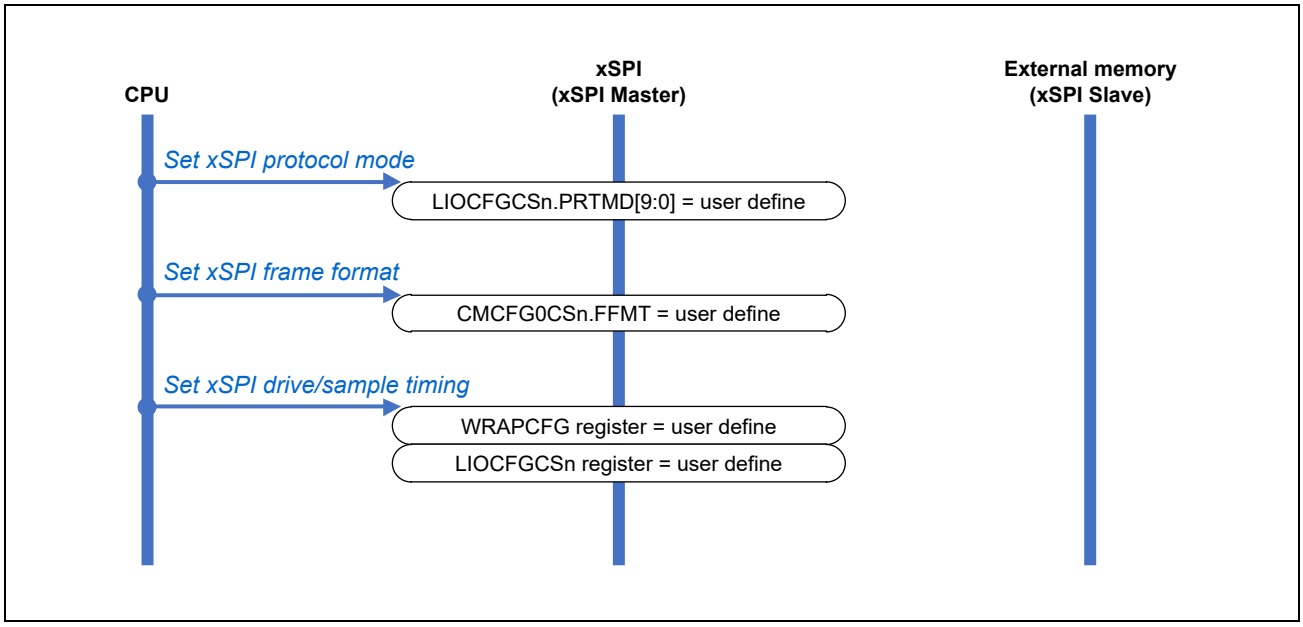


Figure 28.17 Flow of Configuration

28.4.7.2 Flow of Communication Stop

The figure below shows flow of communication stop.

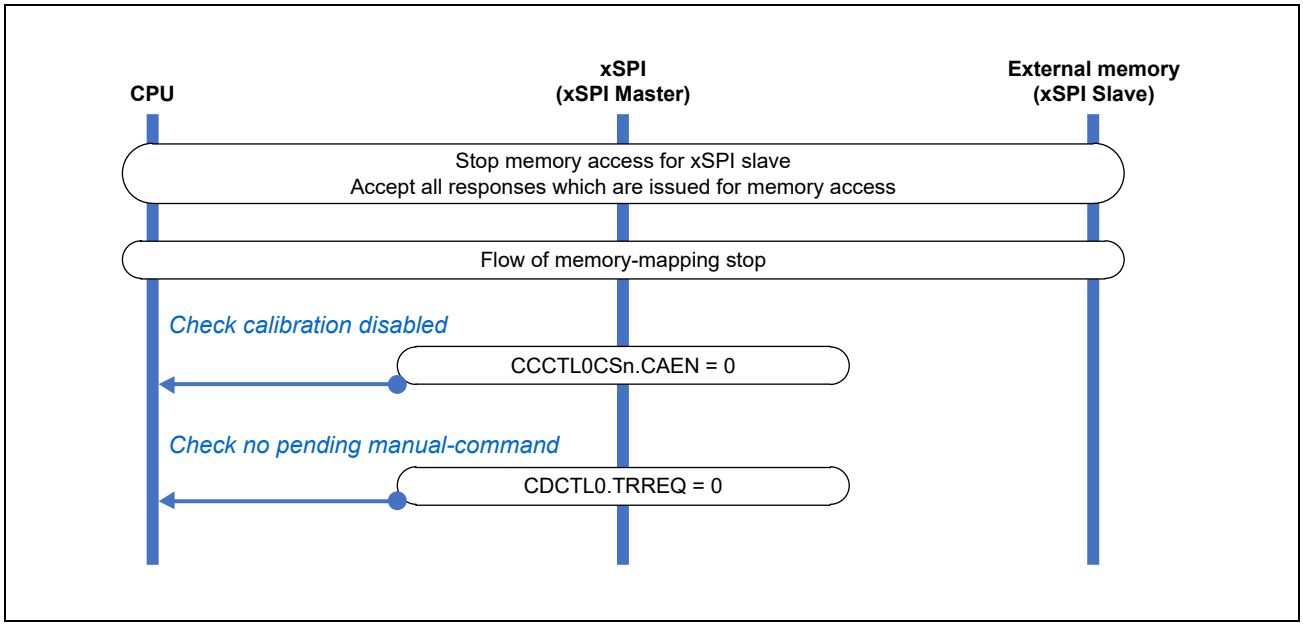


Figure 28.18 Flow of Communication Stop

NOTE

In case of re-config of any configuration register, all communication with xSPI slave shall be stopped completely. It means that the automatic calibration is disabled, and there is no pending manual-command and memory-mapping access.

28.4.7.3 Flow of Automatic Calibration

The figure below shows flow of automatic calibration.

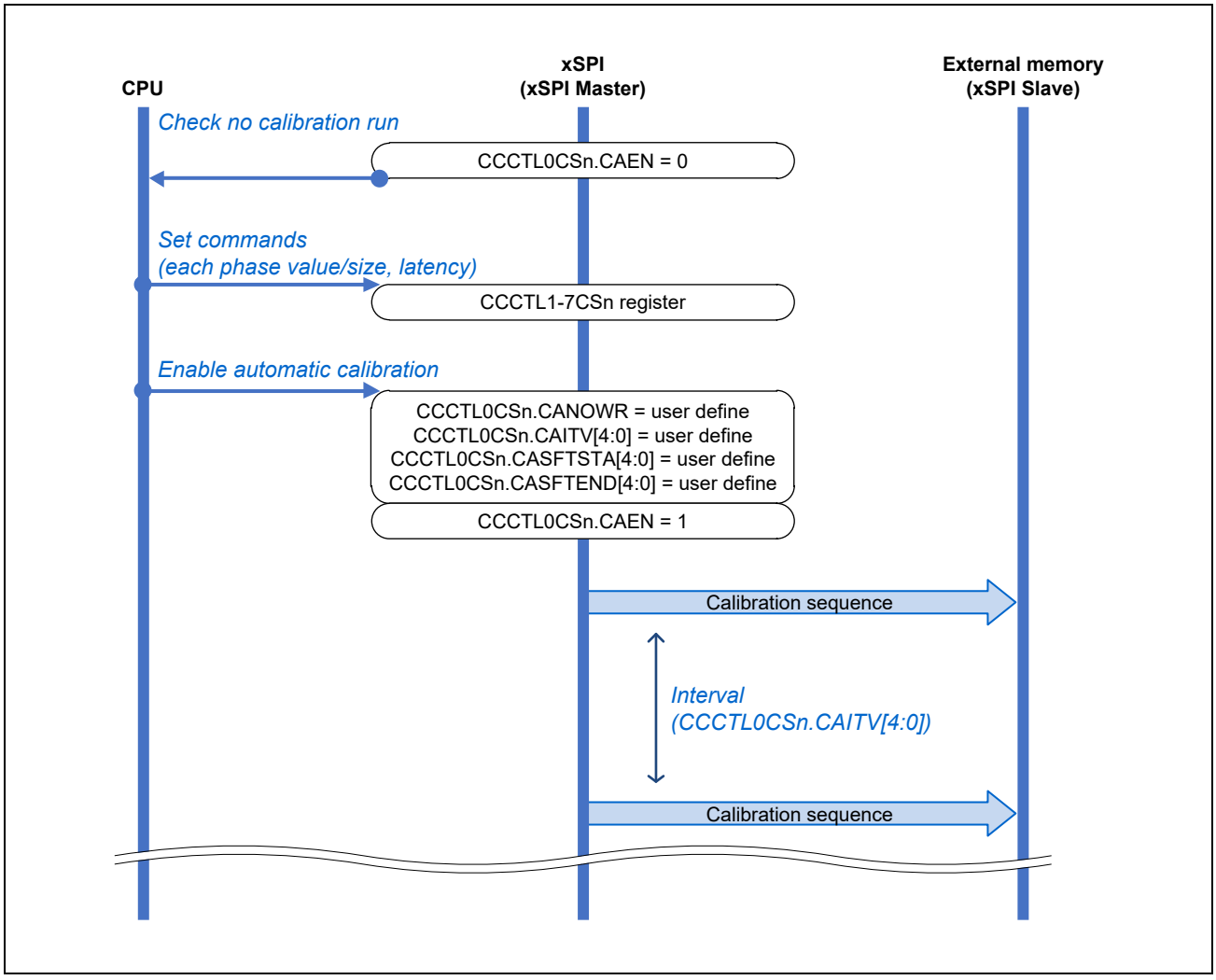


Figure 28.19 Flow of Automatic Calibration

28.4.7.4 Flow of Manual-command Procedure

The figure below shows manual-command procedure for direct mode.

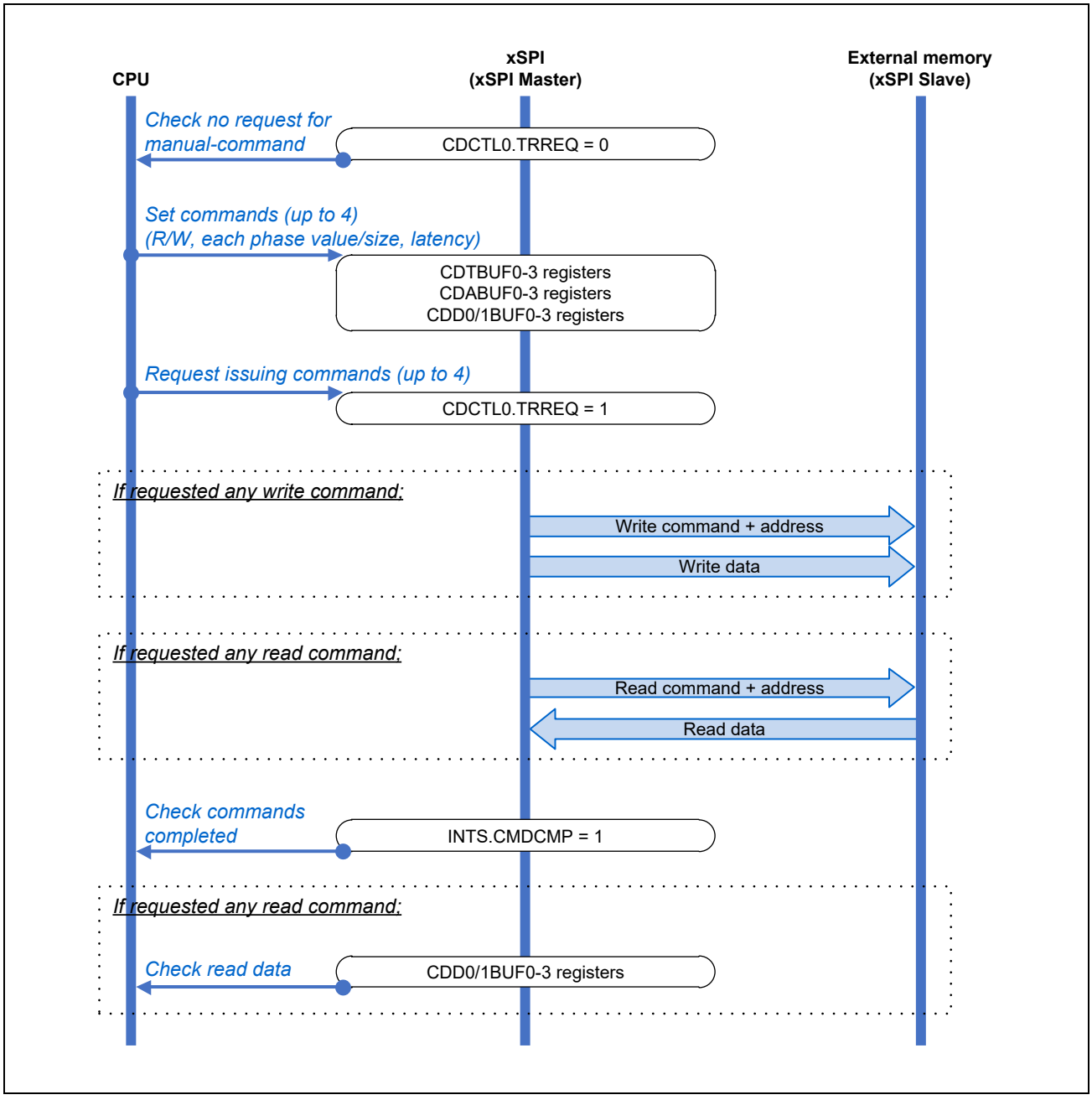


Figure 28.20 Flow of Manual-Command Procedure for Direct Mode

The figure below shows manual-command procedure for periodic mode.

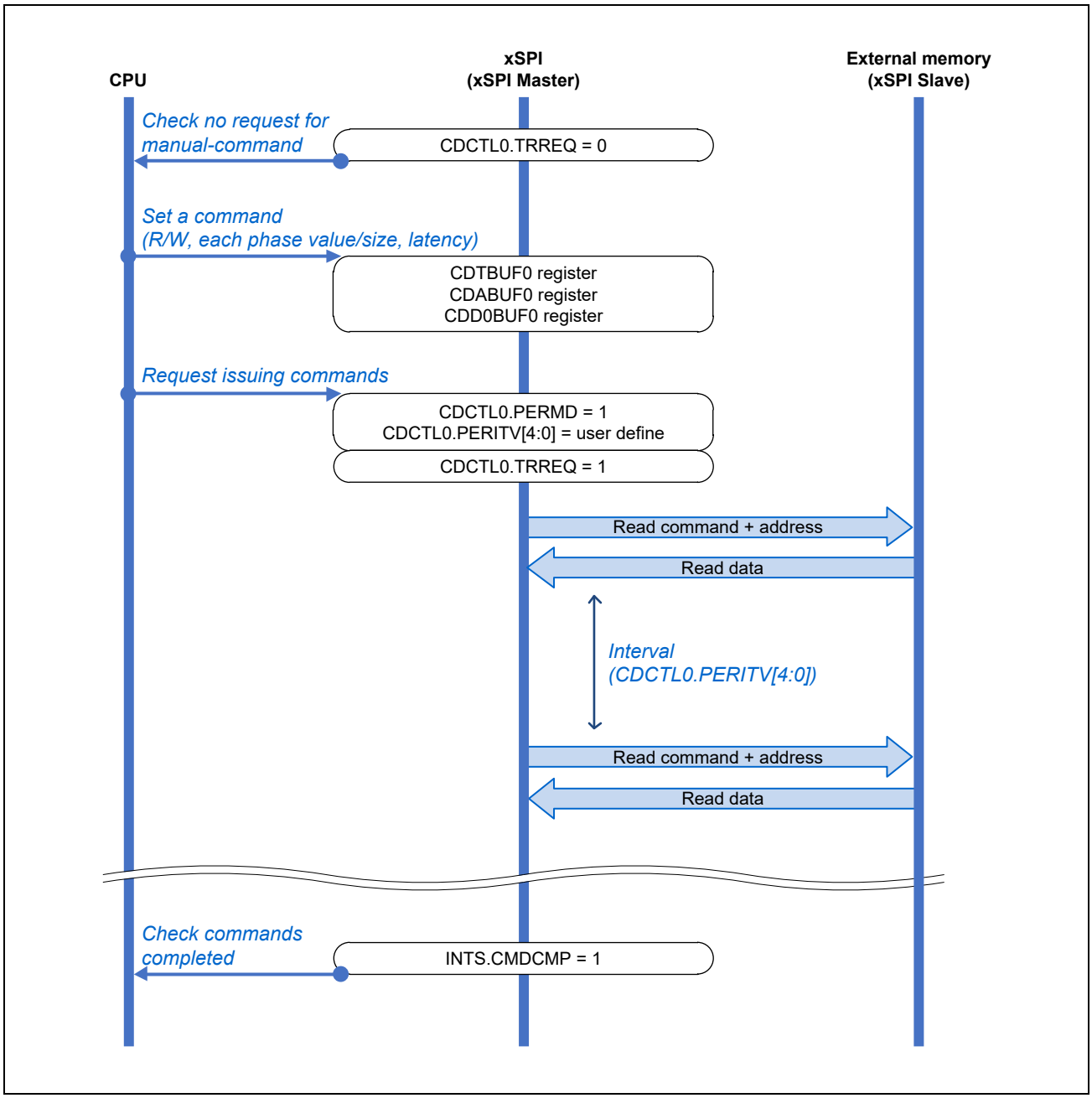


Figure 28.21 Flow of Manual-Command Procedure for Periodic Mode

28.4.7.5 Flow of Memory-Mapping

The figure below shows flow of memory-mapping.

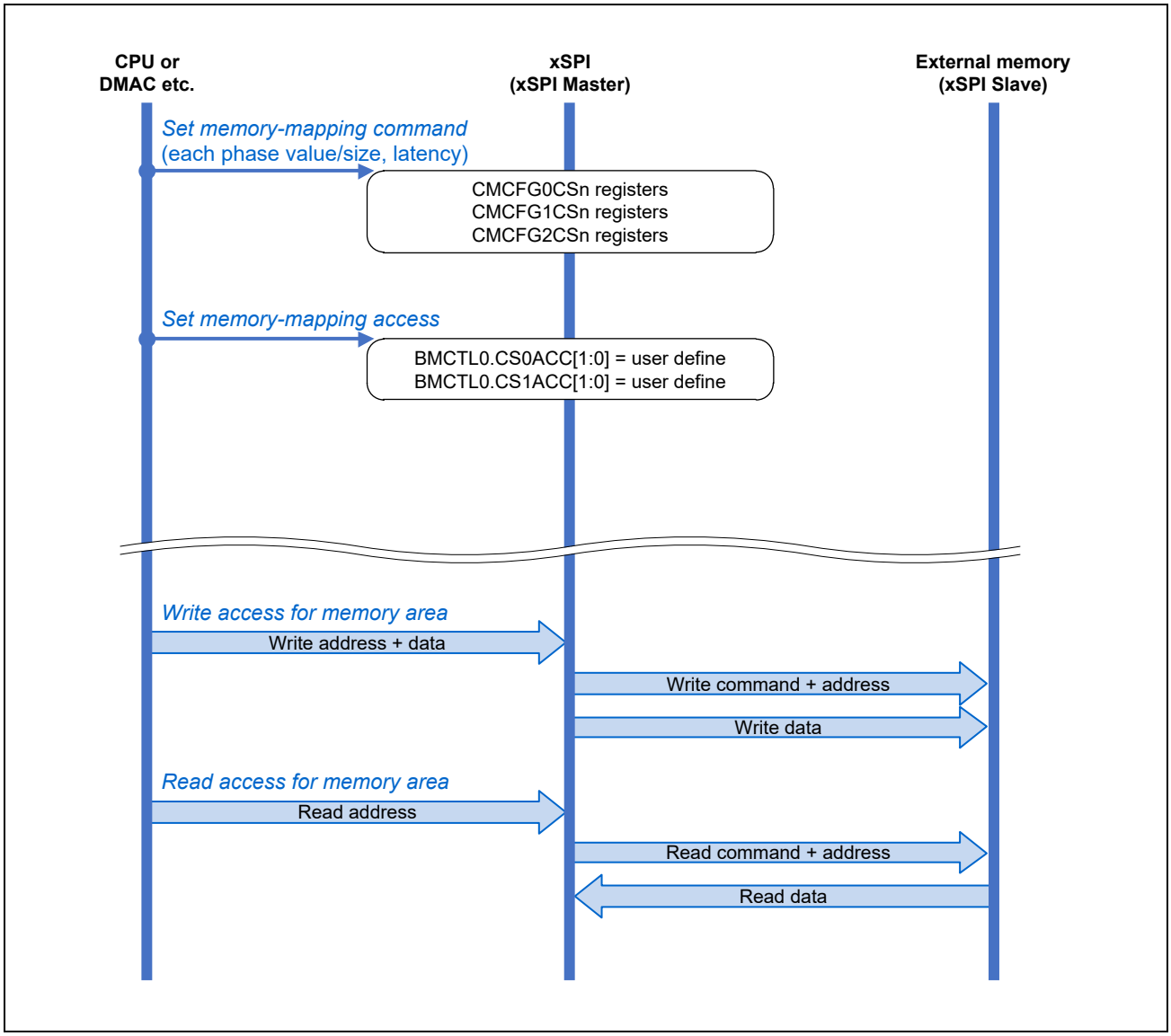


Figure 28.22 Flow of Memory-Mapping

28.4.7.6 Flow of Memory-mapping Stop

The figure below shows flow of memory-mapping stop.

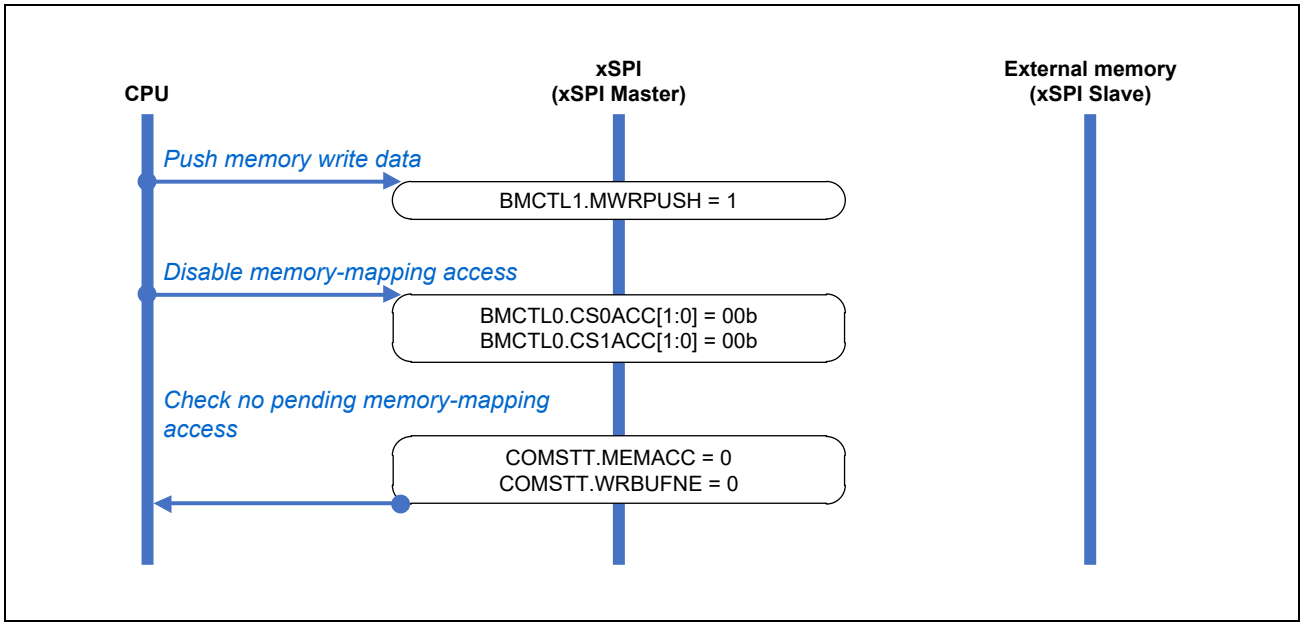


Figure 28.23 Flow of Memory-Mapping Stop

28.4.7.7 Flow of Pattern Request

The figure below shows flow of pattern request. Before requesting any pattern, any ongoing commands should be completed or canceled.

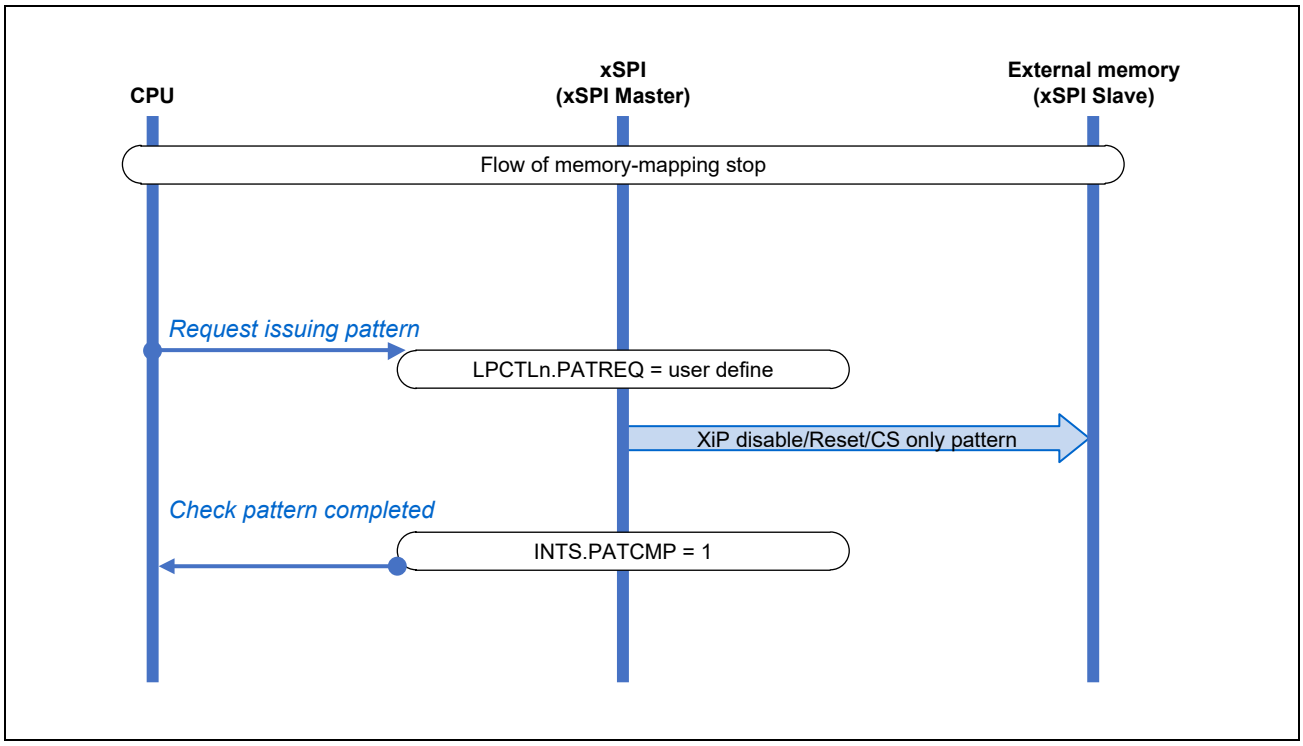


Figure 28.24 Flow of Pattern Request

28.4.7.8 Flow of XiP Mode

The figure below shows flow of XiP mode enable/disable.

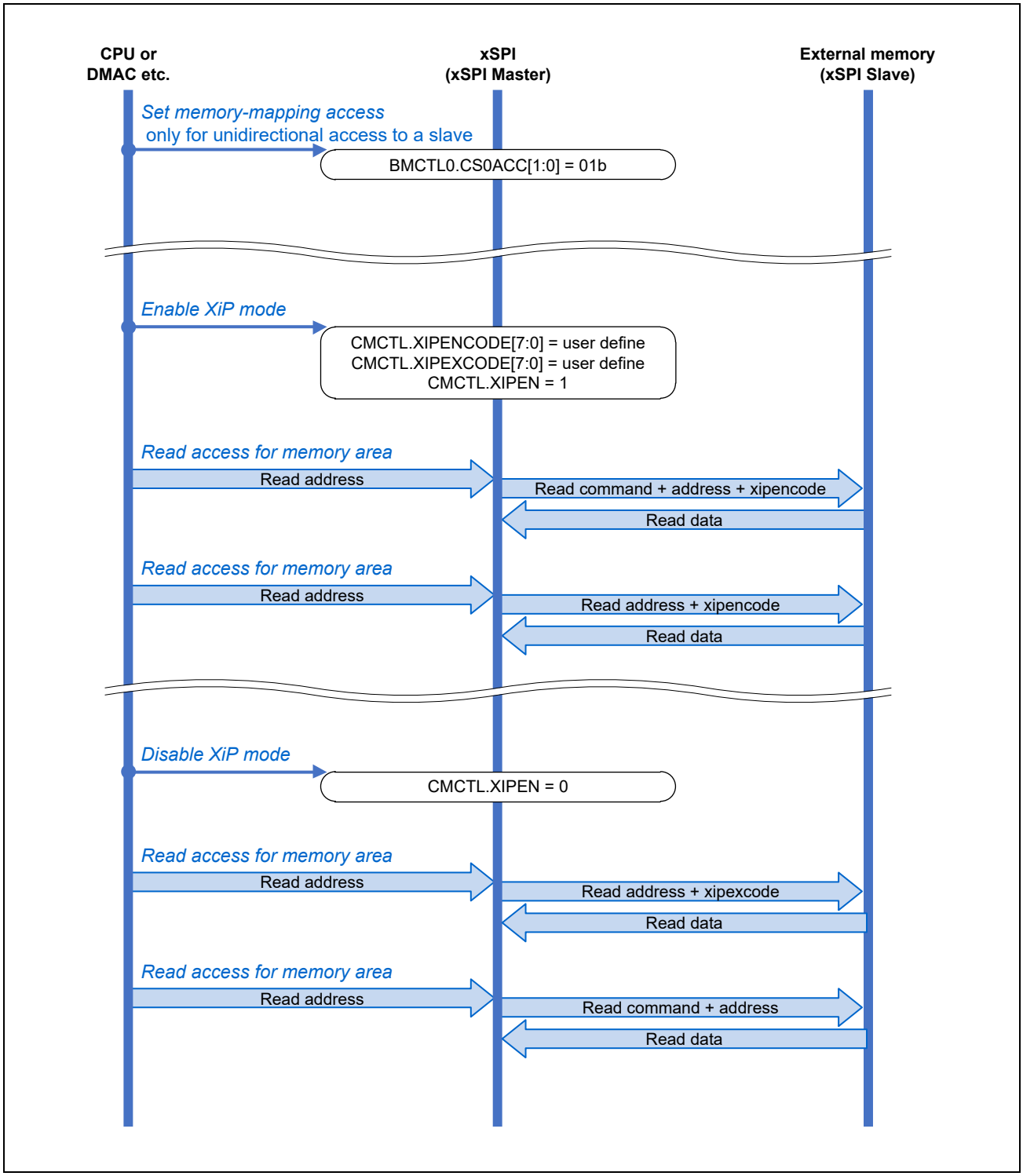


Figure 28.25 Flow of XiP Mode Enable/Disable

28.4.8 Usage Notes

DDR access without DS is not supported.

29. Octa Memory Controller

The Octa memory controller enables the connection of OctaFlash™ and OctaRAM™ to this LSI chip.

NOTE

OctaFlash™ and OctaRAM™ are trademarks of Macronix International Co., Ltd.

29.1 Features

- Macronix Serial Multi I/O (MXSMIO®) Octa Peripheral Interface (OPI) for high-end consumer applications is supported.
- One each of an OctaFlash device and an OctaRAM device compliant with the OPI specifications are connectable.
- A chip select signal is assigned to each memory device (XSPI_CS0#: OctaFlash; XSPI_CS1#: OctaRAM).

  Note: Only one of the memory devices can operate (be read or written to) at a time.

- Supported device interfaces

  SPI: Serial peripheral interface (OctaFlash, SPI mode)

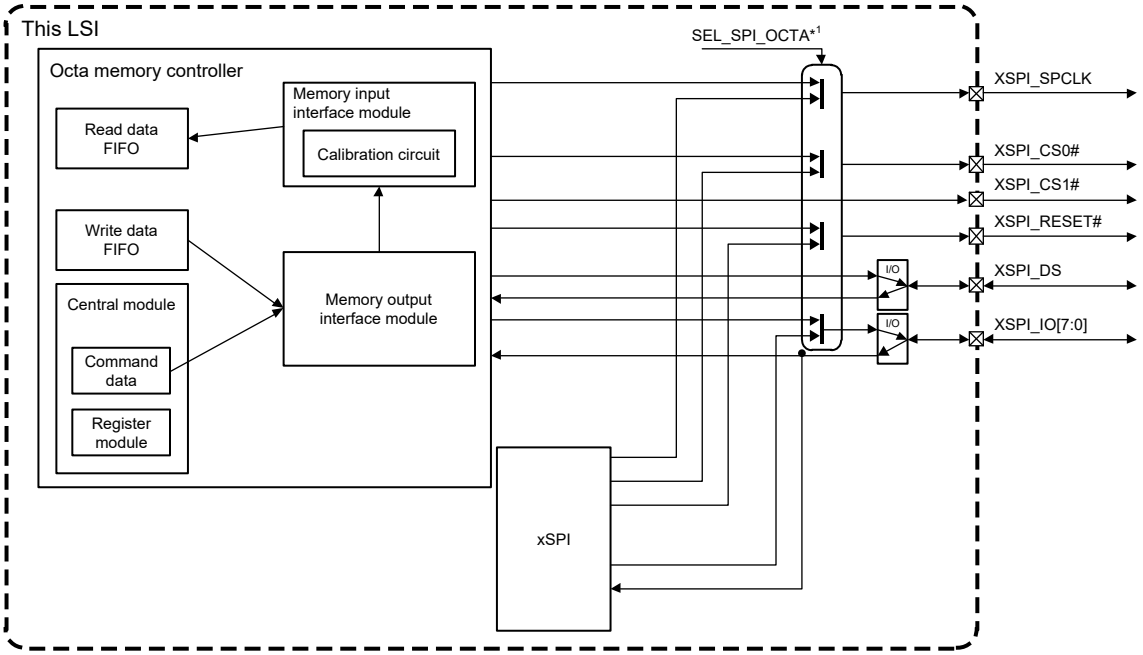
  SOPI: Single Octa I/O (8 bits) (OctaFlash, single data rate)

  DOPI: Double Octa I/O (8 bits) (OctaFlash and OctaRAM, double data rate)

- 3-byte and 4-byte OctaFlash address commands are supported.
- 4-byte OctaRAM address commands are supported.
- The read-while-write (RWW) operation is supported.
- 1LC and 2LC (latency count) OctaRAM devices are supported.
- The error corrected signal (ECS#) is available, and ECC errors can be detected (for OctaFlash only).
- Fast boot mode is not supported.
- Direct access (memory-mapped reading and writing) by the CPU to memory devices in separate flash and RAM address spaces is supported.
- XSPI_SPCLK: 100 MHz
- PVcc_HO: 1.8 V

29.2 Block Diagram

Figure 29.1 shows a block diagram of this module.



Note 1. Before access to the external memory, set up the SEL_SPI_OCTA bit in the xSPI / Octa Controller select register (SYS_IPCONT_SEL_SPI_OCTA). For the SEL_SPI_OCTA bit, refer to **Section 6, System Controller (SYSC)**.

Figure 29.1 Block Diagram

29.3 Input/Output Pins

Table 29.1 shows the pin configuration.

Table 29.1 Configuration of Octa Memory Controller Pins

Pin	I/O	Name	Function
XSPI_SPCLK	Output	Clock	Clock output
XSPI_CS0#	Output	Chip select 0	Chip select signal for an OctaFlash device
XSPI_CS1#	Output	Chip select 1	Chip select signal for an OctaRAM device
XSPI_DS	Input/Output	Read/write data strobe	Read data strobe/write data mask signal
XSPI_IO7 to XSPI_IO0	Input/Output	Data	Data input/output
XSPI_RESET#	Output	Reset output	Reset signal for both OctaFlash and OctaRAM devices

Note: XSPI_ECS# is an input pin that is multiplexed with alternative pin functions that are driven by the 3.3-V power supply. The other pins are solely used for functions that are driven by the 1.8-V power supply.

29.3.1 Device Interface

The device interface is compatible with OctaFlash/OctaRAM interface.

When reading data from the memory or programming the memory with data, the bit order changes with the I/O mode.

The bit order is shown below:

Table 29.2 Bit order in accordance with I/O mode

I/O Mode	Bit Order (MSB)
Read data from device	
1 I/O (SPI)	XSPI_IO1 input
8 I/O (SOPI)	XSPI_IO7 to XSPI_IO4, XSPI_IO3 to XSPI_IO0 input
8 I/O (DOPI)	XSPI_IO7 to XSPI_IO4, XSPI_IO3 to XSPI_IO0 input. Order of data: {D1, D0}, {D3, D2}, ...
Program/Write data to device	
1 I/O (SPI)	XSPI_IO0 output
8 I/O (SOPI)	XSPI_IO7 to XSPI_IO4, XSPI_IO3 to XSPI_IO0 output
8 I/O (DOPI)	XSPI_IO7 to XSPI_IO4, XSPI_IO3 to XSPI_IO0 output. Order of data: {D1, D0}, {D3, D2}, ...

29.4 Register Descriptions

Base address: H'1008_0000 (Overall Address Space)
Base address: H'4008_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
Base address: H'5008_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 29.3 shows the register configuration.

Access to the flash memory or RAM is allowed in either of the following modes.

Memory-map mode

Reading from and writing to the memory array areas of OctaFlash memory or RAM is possible.

Configuration mode

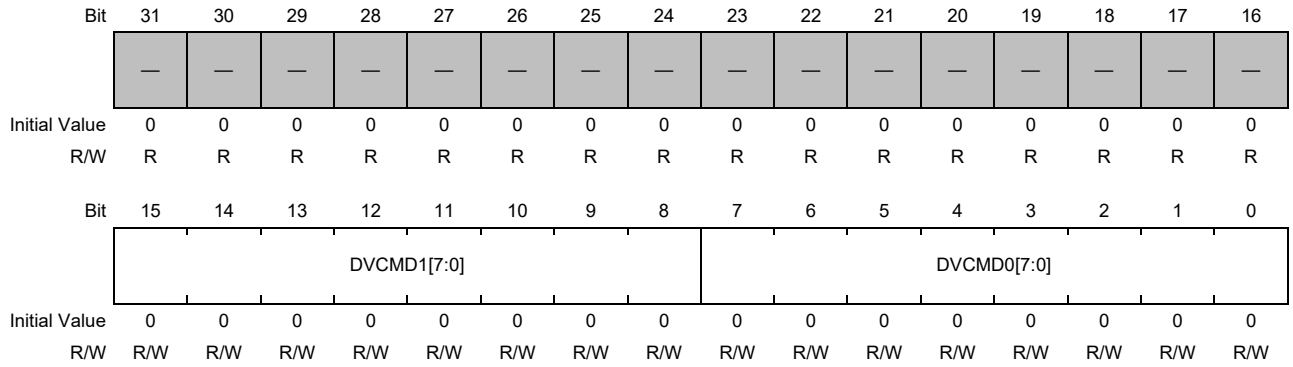
Transfer of any commands and data is possible.

Table 29.3 Octa Memory Controller Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Device command register	DCR	R/W	H'0000_0000	H'00	32
Device address register	DAR	R/W	H'0000_0000	H'04	32
Device command setting register	DCSR	R/W	H'0000_0000	H'08	32
Device size register 0	DSR0	R/W	H'0000_0000	H'0C	32
Device size register 1	DSR1	R/W	H'0000_0000	H'10	32
Memory delay trim register	MDTR	R/W	H'0600_9400	H'14	32
Auto-calibration timer register	ACTR	R/W	H'1000_0000	H'18	32
Auto-calibration address register 0	ACAR0	R/W	H'0000_0000	H'1C	32
Auto-calibration address register 1	ACAR1	R/W	H'0000_0000	H'20	32
Device memory map read chip select timing setting register	DRCSTR	R/W	H'0000_0000	H'34	32
Device memory map write chip select timing setting register	DWCSTR	R/W	H'0000_0000	H'38	32
Device chip select timing setting register	DCSTR	R/W	H'0000_0000	H'3C	32
Controller and device setting register	CDSR	R/W	H'0000_0000	H'40	32
Memory map dummy length register	MDLR	R/W	H'0000_0000	H'44	32
Memory map read/write command register 0	MRWCR0	R/W	H'0000_0000	H'48	32
Memory map read/write command register 1	MRWCR1	R/W	H'0000_0000	H'4C	32
Memory map read/write setting register	MRWCSR	R/W	H'0000_0000	H'50	32
Error status register	ESR	R	H'0000_0000	H'54	32
Configure write without data register	CWNDR	W	H'0000_0000	H'58	32
Configure write data register	CWDR	W	H'0000_0000	H'5C	32
Configure read register	CRR	R	H'0000_0000	H'60	32
Device reset register	RSTCN	R/W	H'0000_0000	H'80	32
Address extend register	AER	R/W	H'0000_0000	H'84	32

29.4.1 Device Command Register (DCR)

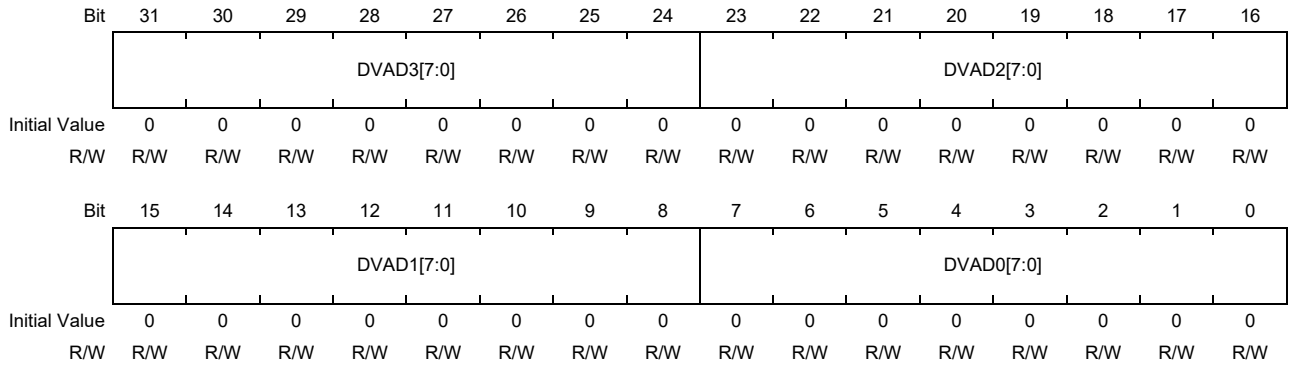
DCR sets the device commands for controller operation.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	DVCMD1[7:0]	All 0	R/W	Device Command data Sets the data to be transferred as a command to the device. If DCSR.CMDLEN[2:0] = 2, the controller will send DCR[15:8] as the 1st byte to memory.
7 to 0	DVCMD0[7:0]	All 0	R/W	Device Command data Sets the data to be transferred as a command to the device. If DCSR.CMDLEN[2:0] = 1, the controller will send DCR[7:0] as the 1st byte to memory. If DCSR.CMDLEN[2:0] = 2, the controller will send DCR[7:0] as the 2nd byte to memory.

29.4.2 Device Address Register (DAR)

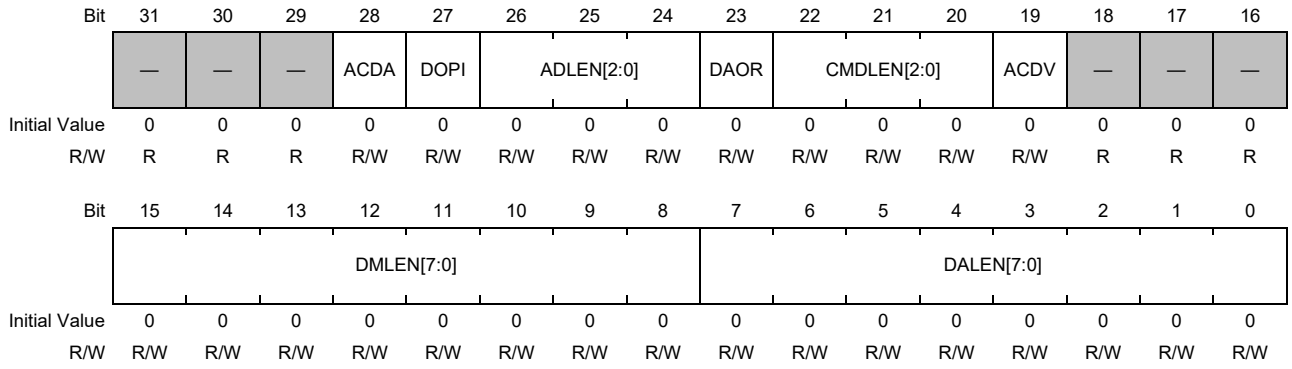
DAR sets the device address for controller operation.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	DVAD3[7:0]	All 0	R/W	Device Address data 3 Set the device address data. When DCSR.ADLEN[2:0] = 4 only, set as the first byte.
23 to 16	DVAD2[7:0]	All 0	R/W	Device Address data 2 Set the device address data. When DCSR.ADLEN[2:0] = 3, set as the first byte. When DCSR.ADLEN[2:0] = 4, set as the second byte.
15 to 8	DVAD1[7:0]	All 0	R/W	Device Address data 1 Set the device address data. When DCSR.ADLEN[2:0] = 3, set as the second byte. When DCSR.ADLEN[2:0] = 4, set as the third byte.
7 to 0	DVAD0[7:0]	All 0	R/W	Device Address data 0 Set the device address data. When DCSR.ADLEN[2:0] = 3, set as the third byte. When DCSR.ADLEN[2:0] = 4, set as the fourth byte.

29.4.3 Device Command Setting Register (DCSR)

DCSR sets device commands.

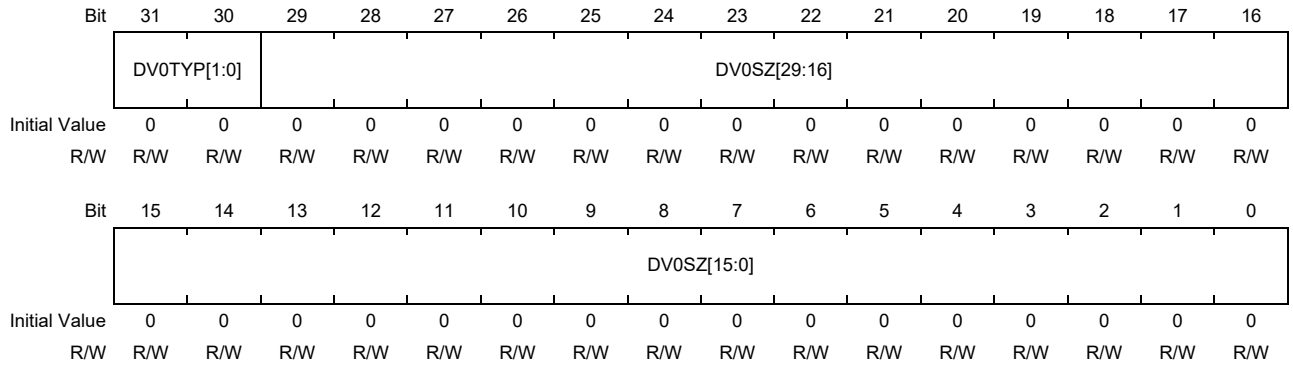


Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	ACDA	0	R/W	Data Access Control Set data access or register access 0: Register access Do not arrange the transfer data. 1: Data access (1) The address bit0 will be forced to 0 in DOPI mode. (2) If DAR[0] of the lead type command is 1, byte 0 from the device will be dropped in DOPI mode. (3) The AXI read data will be arranged according to the DAR. For example: When DAR[1:0] = 1, the first valid data is stored in RDATA[15:8]. (4) The AXI write data is used according to the DAR in DOPI mode. For example: If the target device is Flash Memory and DAR[1:0] = 1, the first byte sent to the flash memory is forced to H'FF. If the target device is RAM and DAR[1: 0] = 1, the first byte sent to RAM is masked by DQSM. (5) When the target device is RAM, the address sent to the memory changes according to RAM specification.
27	DOPI	0	R/W	DOPI single byte setting Even in DOPI mode, set this bit to 1 when the read data has only one byte in each cycle. For example: OctaFlash RDID, RDSR command 0: Each cycle has two bytes data. (normal DOPI mode) 1: Each cycle has one byte data. (The byte data changes at the rising edge of the clock and does not change at the falling edge of the clock.)
26 to 24	ADLEN[2:0]	000	R/W	Transfer address length setting Sets the length of the address to be transferred in bytes.
23	DAOR	0	R/W	Data order setting Sets the byte order of data during read and write operations. 0: byte0, byte1, byte2, byte3 1: byte1, byte0, byte3, byte2
22 to 20	CMDLEN[2:0]	000	R/W	Transfer command length setting Sets the length of the command to be transferred in bytes.
19	ACDV	0	R/W	Access Device setting Sets the device to be accessed. 0: Send commands to device 0. 1: Send commands to device 1.
18 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	DMLEN[7:0]	All 0	R/W	Dummy cycle (latency) setting Sets the length of dummy cycle (latency) in XSPI_SPCLK units.
7 to 0	DALEN[7:0]	All 0	R/W	Transfer data length setting Sets the length of data to be transferred in bytes. When [7:0] is 0, there are no transmission data to memory and reception data from memory.

29.4.4 Device Size Register 0 (DSR0)

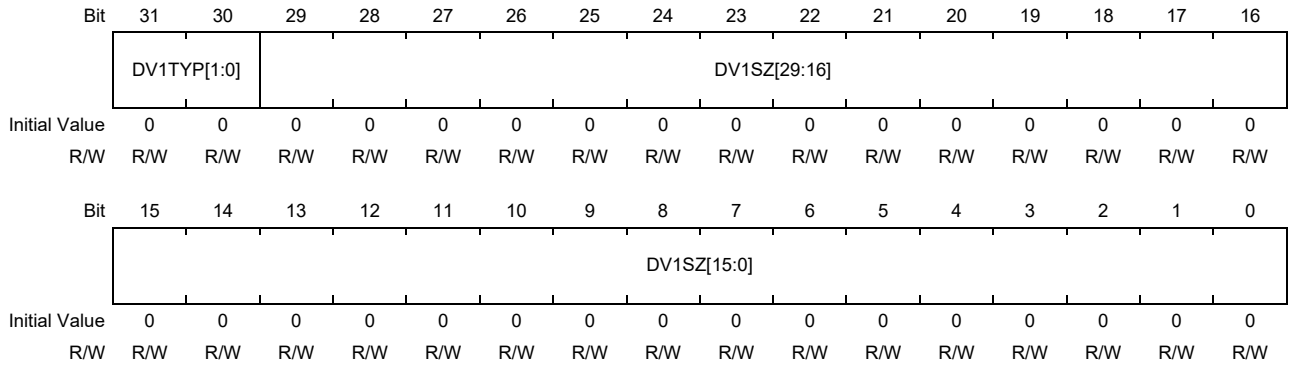
DSR0 specifies the type of memory to be accessed as device 0 and the size of the flash memory.



Bit	Bit Name	Initial Value	R/W	Description
31, 30	DV0TYP[1:0]	00	R/W	Device 0 type setting Set to 00.
29 to 0	DV0SZ[29:0]	All 0	R/W	Device 0 size setting Set a 30-bit value for the size of memory connected as device 0. [Examples] H'0080_0000: 8 Mbytes flash memory H'0100_0000: 16 Mbytes flash memory H'0200_0000: 32 Mbytes flash memory H'0400_0000: 64 Mbytes flash memory H'0800_0000: 128 Mbytes flash memory

29.4.5 Device Size Register 1 (DSR1)

DSR1 specifies the type of memory to be accessed as device 1 and the size of the RAM.



Bit	Bit Name	Initial Value	R/W	Description
31, 30	DV1TYP[1:0]	00	R/W	Device 1 type setting Set to 01.
29 to 0	DV1SZ[29:0]	All 0	R/W	Device 1 size setting Set a 30-bit value for the size of memory connected as device 1. [Examples] H'0080_0000: 8 Mbytes RAM H'0100_0000: 16 Mbytes RAM H'0200_0000: 32 Mbytes RAM H'0400_0000: 64 Mbytes RAM

29.4.6 Memory Delay Trim Register (MDTR)

MDTR sets the timing adjustment of memory access.

For the information about the auto-calibration, refer to **Section 29.5, Operation**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	DQSEDOPI[3:0]				DV1DEL[7:0]							
Initial Value	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DQSESOPI[3:0]				DQSERAM[3:0]				DV0DEL[7:0]							
Initial Value	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	DQSEDOPI [3:0]	H'6	R/W	XSPI_DS enable counter Setting for the flash memory in the DOPI mode
23 to 16	DV1DEL[7:0]	All 0	R/W	Device 1 delay setting These bits specify the number of cycles of delay to be inserted in the input strobe signal (XSPI_DS) to adjust the timing for latching data during read access to device 1 in the DOPI mode (the RAM).
15 to 12	DQSESOPI [3:0]	H'9	R/W	XSPI_DS enable counter Setting for the flash memory in the DOPI mode
11 to 8	DQSERAM [3:0]	H'4	R/W	XSPI_DS enable counter Setting for the RAM in the DOPI mode
7 to 0	DV0DEL[7:0]	All 0	R/W	Device 0 delay setting These bits specify the delay cycles to be inserted on the input strobe signal (XSPI_DS) to adjust the timing for latching data during read access to device 0 in the SOPI or DOPI mode for the flash memory.

During a read operation with the XSPI_DS clock input (in the SOPI or DOPI mode), the XSPI_DS clock transitions from the high-impedance state to the input state (for receiving a value of 0 from the external device) after the command and address phases are completed. To obtain valid data, adjust the DQS enable counter and delay cycles.

XSPI_DS enable counter setting (XSPI_SPCLK units)

0000: 1 clock cycle
0001: 2 clock cycles
0010: 3 clock cycles
0010: 4 clock cycles
…
1111: 16 clock cycles

XSPI_DS enable counter setting examples

Table 29.4 Examples of Setting the XSPI_DS Enable Counter for OctaRAM

Number of Dummy Cycles (Latency)	RAM	
	Pre-Cycle On	Pre-Cycle Off
3	4	4 to 5
4	4 to 5	4 to 6
5	4 to 6	4 to 7
6	4 to 7	4 to 8
7	4 to 8	4 to 9
8	4 to 9	4 to 10

Table 29.5 Examples of Setting the XSPI_DS Enable Counter for OctaFlash Memory

Number of Dummy Cycles (Latency)	SOPI	DOPI	
	Pre-Cycle On	Pre-Cycle On	Pre-Cycle Off
4	8 to 9	5 to 6	5 to 7
6	8 to 11	5 to 8	5 to 9
8	8 to 13	5 to 10	5 to 11
10	8 to 15	5 to 12	5 to 13
12	8 to 15	5 to 14	5 to 15
14	8 to 15	5 to 15	5 to 15
16	8 to 15	5 to 15	5 to 15
18	8 to 15	5 to 15	5 to 15
20	8 to 15	5 to 15	5 to 15

These two tables list cases of DQS delay.

If DQS delay cycle is not within this range, refer to **Section 29.7, XSPI_DS Enable Counter**, and recalculate the correct range.

29.4.7 Auto-Calibration Timer Register (ACTR)

ACTR sets the cycle at which automatic calibration is executed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CTP[31:16]															
Initial Value	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CTP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CTP[31:0]	H'1000_0000	R/W	Automatic calibration cycle time setting Sets performing cycle for the automatic calibration with 32 bits. Automatic calibration cycle time (ns) = CTP[31:0] × SCLK / 2 When automatic calibration is enabled (CDSR.ACMODE[1:0] = H'1) and the value of the internal timer is equal to that of this register, automatic calibration will start.

29.4.8 Auto-Calibration Address Register 0 (ACAR0)

ACAR0 sets the address in OctaFlash memory for writing to and reading from in auto-calibration of device 0.

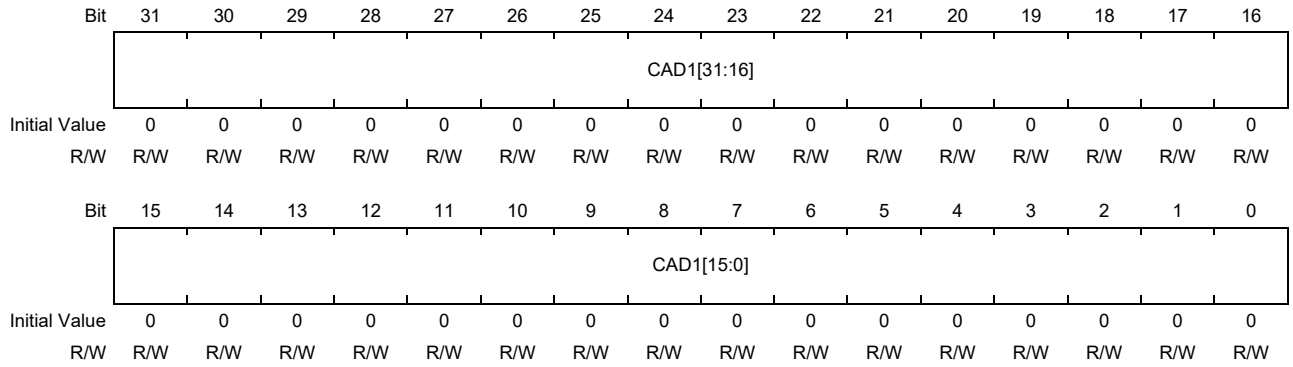
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAD0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CAD0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CAD0[31:0]	All 0	R/W	Automatic calibration address These bits set the address in OctaFlash memory for writing to and reading from in auto-calibration of device 0. *Note*: Set an address in terms of the memory space of the OctaFlash device, not in the OctaFlash space within this LSI chip.

29.4.9 Auto-Calibration Address Register 1 (ACAR1)

ACAR1 sets the address in OctaRAM for writing to and reading from in auto-calibration of device 1.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CAD1[31:0]	All 0	R/W	Automatic calibration address These bits set the address in OctaRAM for writing to and reading from in auto-calibration of device 1. *Note*: Set an address in terms of the memory space of the OctaRAM device, not in the OctaRAM space within this LSI chip.

29.4.10 Device Memory Map Read Chip Select Timing Setting Register (DRCSTR)

DRCSTR sets the timing of memory-mapped reading for each device.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DVRDLO1[1:0]		DVRDHI1[2:0]			DVRDCMD1[2:0]			—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DVRDLO0[1:0]		DVRDHI0[2:0]			DVRDCMD0[2:0]			CTR0	CTRW0[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	DVRDLO1 [1:0]	00	R/W	Device 1 select signal pull-down timing setting*1 Indicates the timing between the select signal pull-down to the read operation for the device 1. Timing definition from XSPI_CS1# low to the first XSPI_SPCLK high DOPI mode 00: 1.5 clock cycles 01: 2.5 clock cycles 10: 3.5 clock cycles 11: 4.5 clock cycles Other mode 00: 2 clock cycles 01: 3 clock cycles 10: 4 clock cycles 11: 5 clock cycles
29, 27	DVRDHI1[2:0]	000	R/W	Device 1 select signal High timing setting*1 Indicates the timing, from the end of read operation to the select signal is pulled up. Timing definition from the last XSPI_SPCLK low to XSPI_CS1# high DOPI or (SOPI) mode 000: Setting prohibited 001: Setting prohibited 010: Setting prohibited 011: Setting prohibited 100: Setting prohibited 101: 6.5 (7) clock cycles 110: 7.5 (8) clock cycles 111: 8.5 (9) clock cycles *Note*: The values in parentheses apply to SOPI mode. Other mode 000: 2 clock cycles 001: 3 clock cycles 010: 4 clock cycles 011: 5 clock cycles 100: 6 clock cycles 101: 7 clock cycles 110: 8 clock cycles 111: 9 clock cycles

Bit	Bit Name	Initial Value	R/W	Description
26 to 24	DVRDCMD1 [2:0]	000	R/W	Device 1 Command execution interval*1 Indicates the timing, between command and command for device 1. Timing, until the XSPI_CS1# signal goes from High to the next Low 000: 2 clock cycles 001: 5 clock cycles 010: 7 clock cycles 011: 9 clock cycles 100: 11 clock cycles 101: 13 clock cycles 110: 15 clock cycles 111: 17 clock cycles
23 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15, 14	DVRDLO0 [1:0]	00	R/W	Device 0 select signal pull-down timing setting*1 Indicates the timing between the select signal pull-down to the read operation for the device 0. Timing definition from XSPI_CS0# low to the first XSPI_SPCLK high DOPI mode 00: 1.5 clock cycles 01: 2.5 clock cycles 10: 3.5 clock cycles 11: 4.5 clock cycles Other mode 00: 2 clock cycles 01: 3 clock cycles 10: 4 clock cycles 11: 5 clock cycles
13 to 11	DVRDHI0[2:0]	000	R/W	Device 0 select signal pull-up timing setting*1 Indicates the timing, from the end of read operation to the select signal is pulled up. Timing definition from the last XSPI_SPCLK low to XSPI_CS0# high DOPI or (SOPI) mode 000: Setting prohibited 001: Setting prohibited 010: Setting prohibited 011: Setting prohibited 100: Setting prohibited 101: 6.5 (7) clock cycles 110: 7.5 (8) clock cycles 111: 8.5 (9) clock cycles *Note*: The values in parentheses apply to SOPI mode. Other mode 000: 2 clock cycles 001: 3 clock cycles 010: 4 clock cycles 011: 5 clock cycles 100: 6 clock cycles 101: 7 clock cycles 110: 8 clock cycles 111: 9 clock cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	DVRDCMD0 [2:0]	000	R/W	Device 0 Command execution interval setting*1 Indicates the timing between command to command for the device 0. Timing definition from XSPI_CS0# high to the next XSPI_CS0# low 000: 2 clock cycles 001: 5 clock cycles 010: 7 clock cycles 011: 9 clock cycles 100: 11 clock cycles 101: 13 clock cycles 110: 15 clock cycles 111: 17 clock cycles
7	CTR0	0	R/W	Device 0 Continuous read setting When the continuous read is enabled, wait for the continuous read operation to be performed using the value set in CTRW0[6:0]. Do not set this bit to 1 except for OctaFlash DOPI mode. 0: Continuous read disabled for device 0 1: Continuous read enabled for device 0
6 to 0	CTRW0[6:0]	All 0	R/W	Device 0 continuous read cycle setting When continuous read of Device 0 is enabled, the selection signal is fixed at Low for the set clock cycle, after the read operation is performed. If the number of clock cycles exceeds twice the value set in these bits before the next read operation is performed, the select signal is returned to High.

Note 1. These clock cycles are values in internal clock units. In XSPI_SPCLK units, the clock cycle is 1/2 of the above table.

29.4.11 Device Memory Map Write Chip Select Timing Setting Register (DWCSTR)

DWCSTR sets the timing of memory-mapped writing for each device.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DVWLO1[1:0]		DVWHI1[2:0]			DVWCMD1[2:0]			—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DVWLO0[1:0]		DVWHI0[2:0]			DVWCMD0[2:0]			—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	DVWLO1[1:0]	00	R/W	Device 1 select signal pull-down timing setting*1 Indicates the timing between the device 1 select signal pull-down to the write operation. Timing definition from XSPI_CS1# low to the first XSPI_SPCLK high DOPI mode 00: 1.5 clock cycles 01: 2.5 clock cycles 10: 3.5 clock cycles 11: 4.5 clock cycles Other mode 00: 2 clock cycles 01: 3 clock cycles 10: 4 clock cycles 11: 5 clock cycles
29 to 27	DVWHI1[2:0]	000	R/W	Device 1 select signal pull-up timing setting*1 Indicates the timing, from the end of write operation to the select signal is pulled up. Timing definition from the last XSPI_SPCLK low to XSPI_CS1# high DOPI mode 000: 1.5 clock cycles 001: 2.5 clock cycles 010: 3.5 clock cycles 011: 4.5 clock cycles 100: 5.5 clock cycles 101: 6.5 clock cycles 110: 7.5 clock cycles 111: 8.5 clock cycles Other mode 000: 2 clock cycles 001: 3 clock cycles 010: 4 clock cycles 011: 5 clock cycles 100: 6 clock cycles 101: 7 clock cycles 110: 8 clock cycles 111: 9 clock cycles

Bit	Bit Name	Initial Value	R/W	Description
26 to 24	DVWCMD1 [2:0]	000	R/W	Device 1 Command execution interval setting*1 Indicates the timing between command to command for the device 1. Timing definition from XSPI_CS1# high to the next XSPI_CS1# low 000: 2 clock cycles 001: 5 clock cycles 010: 7 clock cycles 011: 9 clock cycles 100: 11 clock cycles 101: 13 clock cycles 110: 15 clock cycles 111: 17 clock cycles
23 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15, 14	DVWLO0[1:0]	00	R/W	Device 0 select signal pull-down timing setting*1 Indicates the timing between the select signal pull-down to the write operation for the device 0. Timing definition from XSPI_CS0# low to the first XSPI_SPCLK high DOPI mode 00: 1.5 clock cycles 01: 2.5 clock cycles 10: 3.5 clock cycles 11: 4.5 clock cycles Other mode 00: 2 clock cycles 01: 3 clock cycles 10: 4 clock cycles 11: 5 clock cycles
13 to 11	DVWHI0[2:0]	000	R/W	Device 0 select signal pull-up timing setting*1 Indicates the timing, from the end of write operation to the select signal is pulled up. Timing definition from the last XSPI_SPCLK low to XSPI_CS0# high DOPI mode 000: 1.5 clock cycles 001: 2.5 clock cycles 010: 3.5 clock cycles 011: 4.5 clock cycles 100: 5.5 clock cycles 101: 6.5 clock cycles 110: 7.5 clock cycles 111: 8.5 clock cycles Other mode 000: 2 clock cycles 001: 3 clock cycles 010: 4 clock cycles 011: 5 clock cycles 100: 6 clock cycles 101: 7 clock cycles 110: 8 clock cycles 111: 9 clock cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	DVWCMD0 [2:0]	000	R/W	Device 0 Command execution interval setting*1 Indicates the timing between command to command for the device 0. Timing definition from XSPI_CS0# high to the next XSPI_CS0# low 000: 2 clock cycles 001: 5 clock cycles 010: 7 clock cycles 011: 9 clock cycles 100: 11 clock cycles 101: 13 clock cycles 110: 15 clock cycles 111: 17 clock cycles
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note 1. These clock cycles are values in internal clock units. In XSPI_SPCLK units, the clock cycle is 1/2 of the above table.

29.4.12 Device Chip Select Timing Setting Register (DCSTR)

DCSTR sets the timing of operations in configuration mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DVSELLO[1:0]		DVSELHI[2:0]			DVSELCMD[2:0]			—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15, 14	DVSELLO [1:0]	All 0	R/W	Device select signal pull-down timing setting*1 Indicates the timing, from the device selection signal is pulled-down to the command execution. Timing definition from XSPI_CS[1:0]# low to the first XSPI_SPCLK high DOPI mode 00: 1.5 clock cycles 01: 2.5 clock cycles 10: 3.5 clock cycles 11: 4.5 clock cycles Other mode 00: 2 clock cycles 01: 3 clock cycles 10: 4 clock cycles 11: 5 clock cycles

Bit	Bit Name	Initial Value	R/W	Description
13 to 11	DVSELHI[2:0]	000	R/W	Device select signal pull-up timing setting*1 Indicates the timing when the select signal is pulled up from the end of the command execution. Timing definition from the last XSPI_SPCLK low to XSPI_CS[1:0]# high DOPI or (SOPI) mode 000: Setting prohibited 001: Setting prohibited 010: Setting prohibited 011: Setting prohibited 100: Setting prohibited 101: 6.5 (7) clock cycles 110: 7.5 (8) clock cycles 111: 8.5 (9) clock cycles *Note*: The values in parentheses apply to SOPI mode. Other mode 000: 2 clock cycles 001: 3 clock cycles 010: 4 clock cycles 011: 5 clock cycles 100: 6 clock cycles 101: 7 clock cycles 110: 8 clock cycles 111: 9 clock cycles
10 to 8	DVSELCMD [2:0]	000	R/W	Device Command execution interval setting*1 Indicates the timing between command to command. Timing definition from XSPI_CS[1:0]# high to the next XSPI_CS[1:0]# low 000: 2 clock cycles 001: 5 clock cycles 010: 7 clock cycles 011: 9 clock cycles 100: 11 clock cycles 101: 13 clock cycles 110: 15 clock cycles 111: 17 clock cycles
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note 1. These clock cycles are values in internal clock units. In XSPI_SPCLK units, the clock cycle is 1/2 of the above table.

29.4.13 Controller and Device Setting Register (CDSR)

CDSR controls the automatic calibration, pre-cycle setting, and device transfer type of the controller and each of the devices.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DLFT	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	ACMODE[1:0]		ACMEME[1:0]		—	—	—	—	DV1PC	DV0PC	DV1TTYP[1:0]		DV0TTYP[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

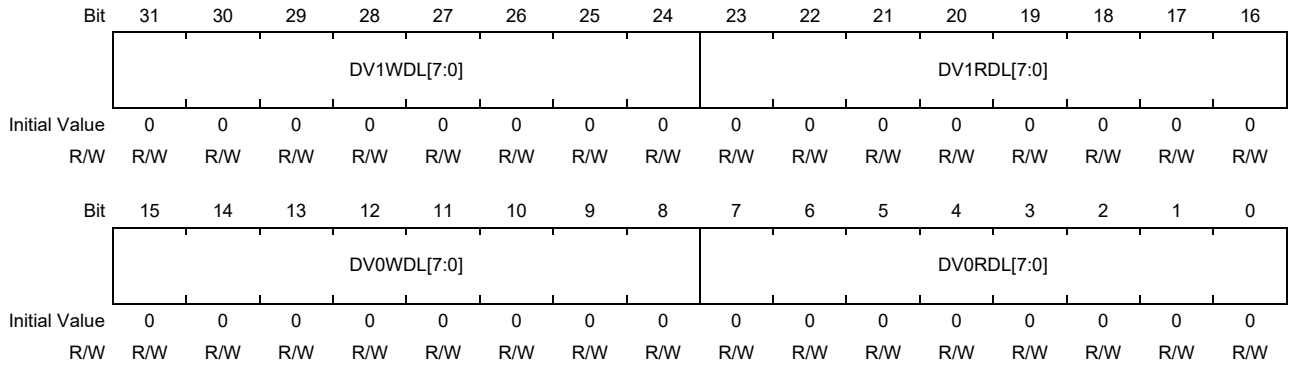
Bit	Bit Name	Initial Value	R/W	Description
31	DLFT	0	R/W	Deadlock Free Timer Enable Enable the timer to prevent the occurrence of controller deadlock. The timeout status is reset only by a system reset. 0: Enable timer 1: Disable timer *Note*: This function is available only when XSPI_SPCLK is 100 MHz. Otherwise, disable the timer.
30 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	ACMODE[1:0]	00	R/W	Automatic calibration mode 00: Automatic calibration is disabled.*1 01: Automatic calibration is enabled. 10: Setting prohibited. 11: Setting prohibited. The ACMODE bits should be set to disabled during read access to the OctaFlash device. Automatic calibration is only available in the DOPI mode.
11, 10	ACMEME[1:0]	00	R/W	Automatic calibration memory enable setting Automatic calibration supports only DOPI mode. To end DOPI mode, disable automatic calibration. [1]: Enable for device 1 1: Enable 0: Disable [0]: Enable for device 0 1: Enable 0: Disable
9 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	DV1PC	0	R/W	Device1_memory pre-cycle setting 0: Disable 1: Enable
4	DV0PC	0	R/W	Device0_memory pre-cycle setting 0: Disable 1: Enable

Bit	Bit Name	Initial Value	R/W	Description
3, 2	DV1TTYP [1:0]	00	R/W	Device1_transfer_type setting 00: SPI mode 01: SOPI mode 10: DOPI mode 11: Setting prohibited
1, 0	DV0TTYP [1:0]	00	R/W	Device0_transfer_type setting 00: SPI mode 01: SOPI mode 10: DOPI mode 11: Setting prohibited

Note 1. When automatic calibration is disabled, the device 0 delay (OctaFlash) and device 1 delay (OctaRAM) should be manually specified in the memory delay trim register (MDTR) (see **Section 29.4.6, Memory Delay Trim Register (MDTR)**). Set this register with reference to “initial setting MDTR (OctaFlash memory)” and “initial setting MDTR (OctaRAM)” in **Figure 29.16**.

29.4.14 Memory Map Dummy Length Register (MDLR)

MDLR sets the number of dummy cycles (latency) for writing and reading the memory map.

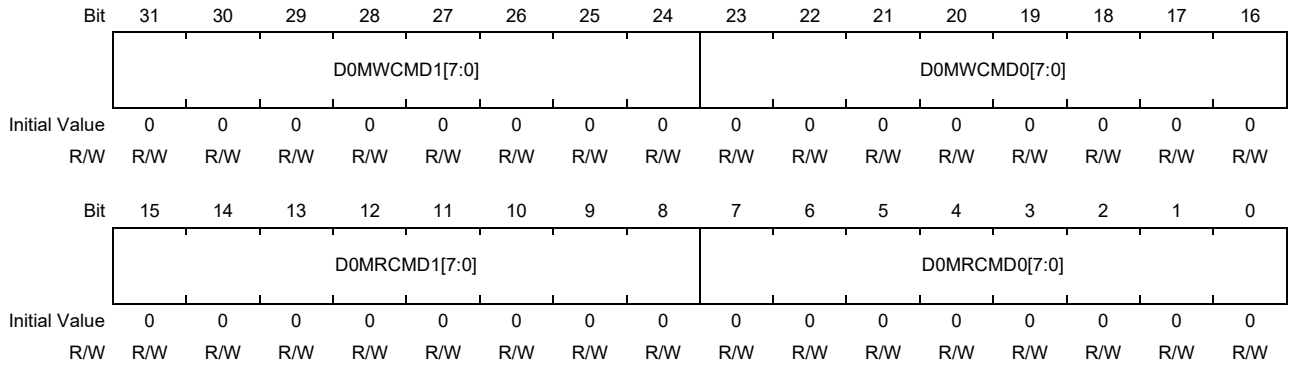


Bit	Bit Name	Initial Value	R/W	Description
31 to 24	DV1WDL[7:0]	All 0	R/W	Device 1 Write dummy cycles (latency) setting Set the dummy length when writing to device 1 in XSPI_SPCLK cycles unit.
23 to 16	DV1RDL[7:0]	All 0	R/W	Device 1 Read dummy cycles (latency) setting Set the dummy length when reading to device 1 in XSPI_SPCLK cycles unit.
15 to 8	DV0WDL[7:0]	All 0	R/W	Device 0 Write dummy cycles (latency) setting Set the dummy length when writing to device 0 in XSPI_SPCLK cycles unit.
7 to 0	DV0RDL[7:0]	All 0	R/W	Device 0 Read dummy cycles (latency) setting Set the dummy length when reading to device 0 in XSPI_SPCLK cycles unit.

Note: When changing the dummy cycles (latency) for the OctaFlash device, modify the setting on the memory and then modify the setting in the DMLEN[7:0] bits in the device command setting register (DCSR) so that the dummy length becomes the same between the memory and this Octa memory controller. After that, execute the status read command to confirm the memory status.

29.4.15 Memory Map Read/Write Command Register 0 (MRWCR0)

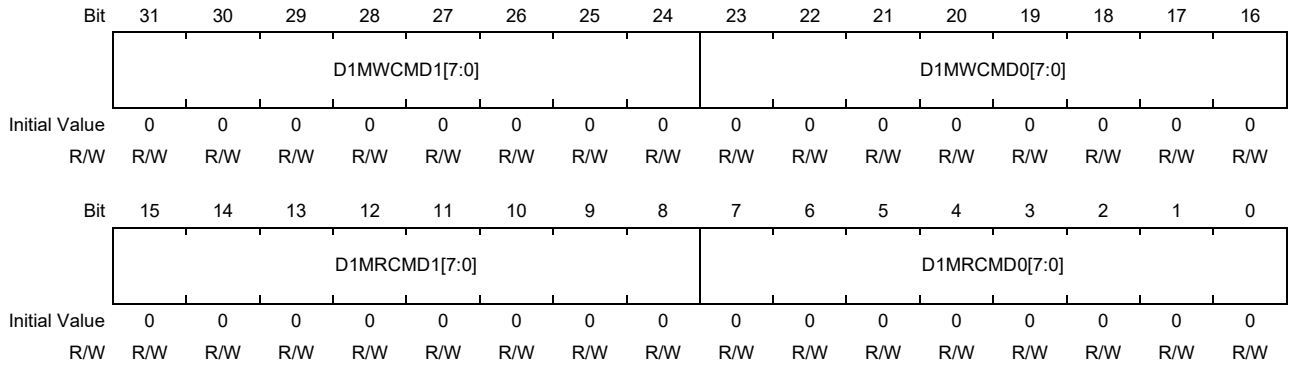
MRWCR0 sets the read/write commands for the device 0.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	D0MWCMD1 [7:0]	All 0	R/W	Memory map write command 1 setting, Set the memory map write command. According to the flash memory spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will not send command 1. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 0 is specified by CDSR.DV0TTYP[1:0].
23 to 16	D0MWCMD0 [7:0]	All 0	R/W	Memory map write command 0 setting, Set the memory map write command. According to the flash memory spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will send only command 0. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 0 is specified by CDSR.DV0TTYP[1:0].
15 to 8	D0MRCMD1 [7:0]	All 0	R/W	Memory map read command 1 setting Set the memory map read command. According to the flash memory spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will not send command 1. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 0 is specified by CDSR.DV0TTYP[1:0].
7 to 0	D0MRCMD0 [7:0]	All 0	R/W	Memory map read command 0 setting, Set the memory map read command. According to the flash memory spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will send only command 0. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 0 is specified by CDSR.DV0TTYP[1:0].

29.4.16 Memory Map Read/Write Command Register 1 (MRWCR1)

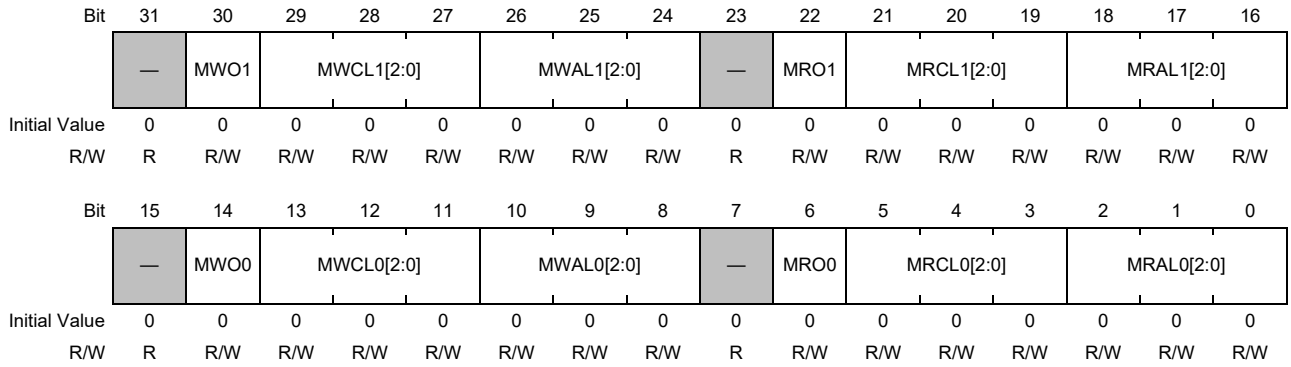
MRWCR1 sets the read/write commands for the device 1.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	D1MWCMD1 [7:0]	All 0	R/W	Memory map write command 1 setting Set the memory map write command. According to the RAM spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will not send command 1. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 1 is specified by CDSR.DV1TTYP[1:0].
23 to 16	D1MWCMD0 [7:0]	All 0	R/W	Memory map write command 0 setting Set the memory map write command. According to the RAM spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will send only command 0. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 1 is specified by CDSR.DV1TTYP[1:0].
15 to 8	D1MRCMD1 [7:0]	All 0	R/W	Memory map read command 1 setting Set the memory map read command. According to the RAM spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will not send command 1. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 1 is specified by CDSR.DV1TTYP[1:0].
7 to 0	D1MRCMD0 [7:0]	All 0	R/W	Memory map read command 0 setting Set the memory map read command. According to the RAM spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will send only command 0. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 1 is specified by CDSR.DV1TTYP[1:0].

29.4.17 Memory Map Read/Write Setting Register (MRWCSR)

MRWCSR sets the memory map read/write operations.



Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	MWO1	MWCL1[2:0]			MWAL1[2:0]			—	MRO1	MRCL1[2:0]			MRAL1[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	MWO0	MWCL0[2:0]			MWAL0[2:0]			—	MRO0	MRCL0[2:0]			MRAL0[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	MWO1	0	R/W	Device 1 write order setting 0: Write order is byte0, byte1, byte2, byte3. 1: Write order is byte1, byte0, byte3, byte2.
29 to 27	MWCL1[2:0]	000	R/W	Device 1 write command length setting Set the number of bytes in the write command for device 1.
26 to 24	MWAL1[2:0]	000	R/W	Device 1 write address length setting Set the memory map write address byte length for device 1.
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	MRO1	0	R/W	Device 1 read order setting 0: Read order is byte0, byte1, byte2, byte3. 1: Read order is byte1, byte0, byte3, byte2. *Note*: This setting is only valid in the DOPI mode. It has no effect in the SOPI or SPI mode.
21 to 19	MRCL1[2:0]	000	R/W	Device 1 read command length setting Set the read command byte length for device 1.
18 to 16	MRAL1[2:0]	000	R/W	Device 1 read address length setting Set the memory map read address byte length for device 1.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	MWO0	0	R/W	Device 0 write order setting 0: Write order is byte0, byte1, byte2, byte3. 1: Write order is byte1, byte0, byte3, byte2.
13 to 11	MWCL0[2:0]	000	R/W	Device 0 write command length setting Set the write command byte length for device 0.
10 to 8	MWAL0[2:0]	000	R/W	Device 0 write address length setting Set the memory map write address byte length for device 0.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	MRO0	0	R/W	Device 0 read order setting 0: Read order is byte0, byte1, byte2, byte3. 1: Read order is byte1, byte0, byte3, byte2. *Note*: This setting is only valid in the DOPI mode. It has no effect in the SOPI or SPI mode.
5 to 3	MRCL0[2:0]	000	R/W	Device 0 read command length setting Set the write command byte length for device 0.
2 to 0	MRAL0[2:0]	000	R/W	Device 0 read address length setting Set the memory map read address byte length for device 0.

29.4.18 Error Status Register (ESR)

ESR indicates the error status when performing memory map read and write commands.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MWESR[7:0]								MRESR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	0	R	Reserved These bits are always read as 0.
15 to 8	MWESR[7:0]	All 0	R	Memory map write error status H'80 : invalid command H'01 : write data length error
7 to 0	MRESR[7:0]	All 0	R	Memory map read error status H'80 : invalid command H'01 : ECC error H'02 : preamble error H'03 : wait XSPI_DS timeout

29.4.19 Configure Write without Data Register (CWNDR)

CWNDR is used to execute a write command without data in the configuration mode. Writing 0 to this register makes this module issue a write command without actually writing data to memory.
Specify an address in DAR, a command in DCSR, and command data in DCR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	—	All 0	W	The write value should always be 0.

29.4.20 Configure Write Data Register (CWDR)

CWDR is used to set the data for writing when a write command is to be executed in the configuration mode. Writing to this register makes the controller to issue a write command and transfer the data to be written. The byte order of data transferred by the controller to the memory is in accord with the setting of the DAOR bit in CDSR. Specify an address in DAR, a command in DCSR, and command data in DCR.

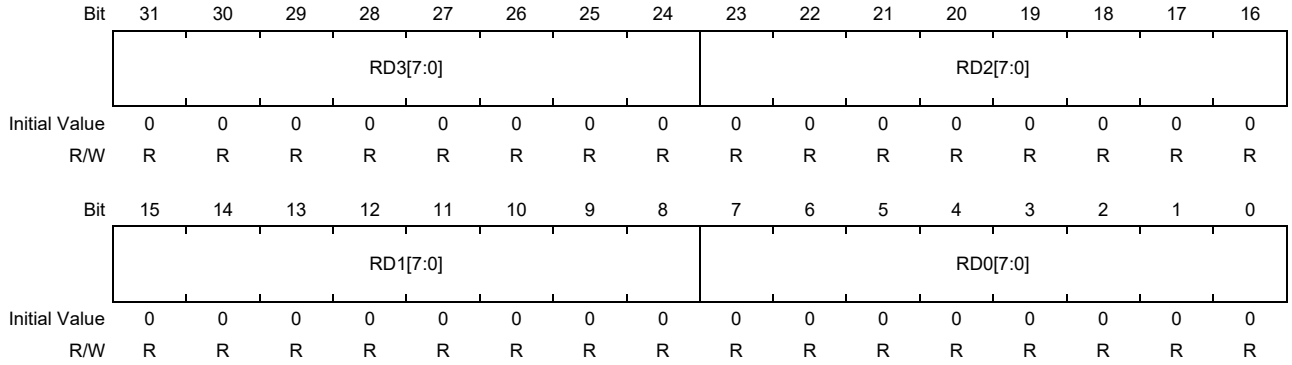
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WD3[7:0]								WD2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WD1[7:0]								WD0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	WD3[7:0]	All 0	W	Write data 3 Set the third byte of the write data.
23 to 16	WD2[7:0]	All 0	W	Write data 2 Set the second byte of the write data.
15 to 8	WD1[7:0]	All 0	W	Write data 1 Set the first byte of the write data.
7 to 0	WD0[7:0]	All 0	W	Write data 0 Set the zeroth byte of the write data.

29.4.21 Configure Read Register (CRR)

CRR is used to execute a read command in the configuration mode and holds the data that are read. Reading from this register makes the controller issue a read command and the data that are read are stored in this register. Specify an address in DAR, a command in DCSR, and command data in DCR.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	RD3[7:0]	All 0	R	Read data 3 Store the third byte of the read data.
23 to 16	RD2[7:0]	All 0	R	Read data 2 Store the second byte of the read data.
15 to 8	RD1[7:0]	All 0	R	Read data 1 Store the first byte of the read data.
7 to 0	RD0[7:0]	All 0	R	Read data 0 Store the zeroth byte of the read data.

29.4.22 Device Reset Register (RSTCNT)

RSTCNT is used to control O_MEM_RESET_N.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RSTVAL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
0	RSTVAL	0	R/W	Output value of O_MEM_RESET_N 0: O_MEM_RESET_N = L 1: O_MEM_RESET_N = H

29.4.23 Address Extend Register (AER)

AER is used to extend the accessible range of OctaRAM.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	AAS		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
2 to 0	AAS	000b	R/W	Accessible address space 000b: H'0_2800_0000 to H'0_287F_FFFF (8 Mbytes) 001b: H'0_2800_0000 to H'0_28FF_FFFF (16 Mbytes) 011b: H'0_2800_0000 to H'0_29FF_FFFF (32 Mbytes) 111b: H'0_2800_0000 to H'0_2BFF_FFFF (64 Mbytes) Others: Prohibited.

29.5 Operation

29.5.1 Octa Memory Controller System Configuration

Figure 29.2 shows an example of connections between the Octa memory controller in this LSI and OctaFlash and OctaRAM devices.

CS0# should be connected to the OctaFlash device and CS1# should be connected to the OctaRAM device.

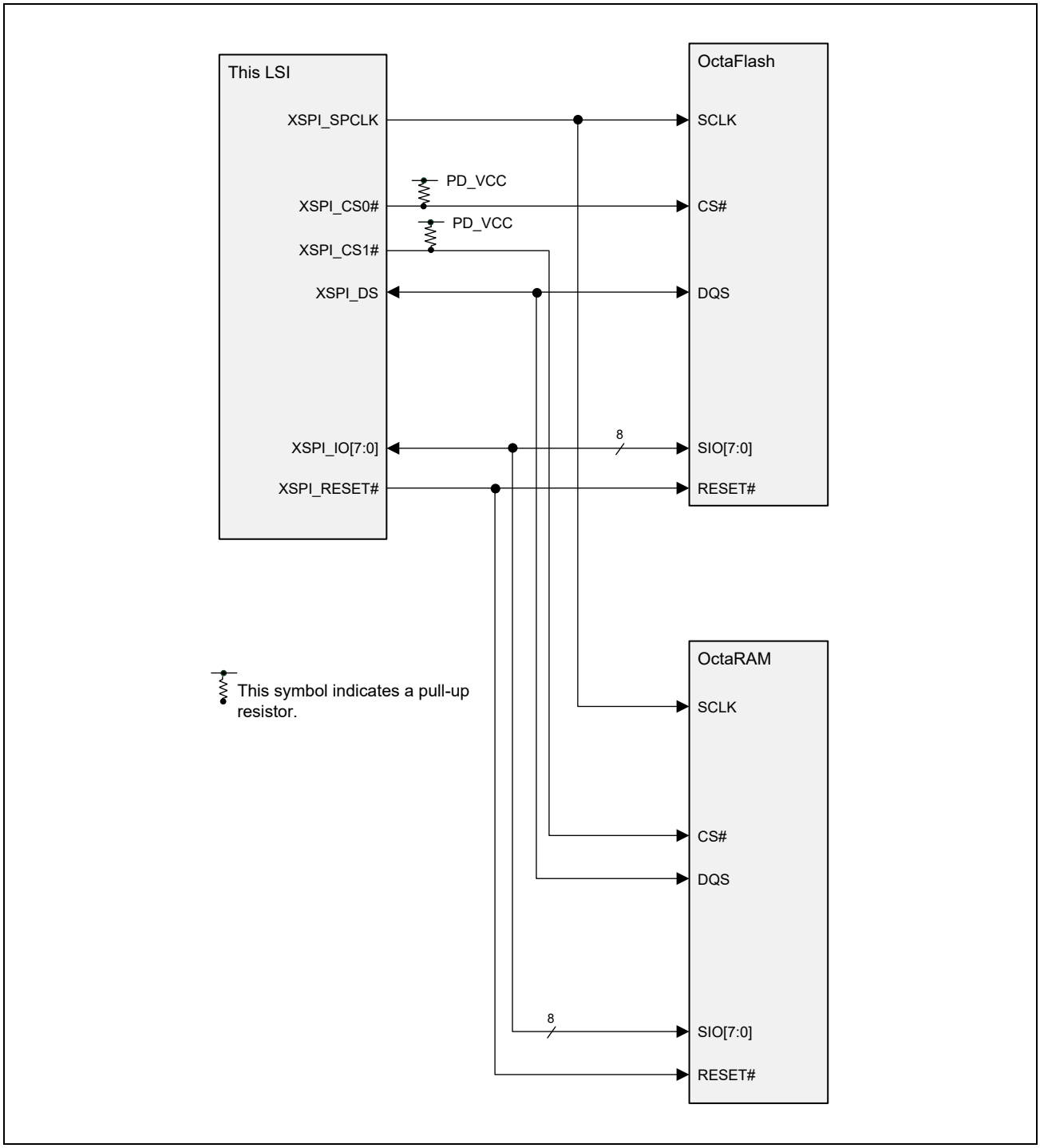


Figure 29.2 Example of Connections between this LSI and OctaFlash and OctaRAM Devices

29.5.2 Address Map

In the memory-map mode, OctaFlash is allocated to the OctaFlash space (H'2000_0000 to H'27FF_FFFF), and OctaRAM is allocated to the OctaRAM space (H'2800_0000 to H'2BFF_FFFF).

One OctaFlash device and one OctaRAM device can be connected to this LSI, and up to 128 Mbytes can be accessed in OctaFlash and up to 64 Mbytes can be accessed in OctaRAM.

	Internal Address	Maximum Accessible Area
OctaFlash	H'2000_0000 to H'27FF_FFFF	Up to 128 Mbytes
OctaRAM	H'2800_0000 to H'2BFF_FFFF	Up to 64 Mbytes

29.5.3 Octa Memory Interface

This section describes the Octa memory interface.

29.5.3.1 Write Operation

The Octa memory controller outputs a command and an address and writes one or more data bytes by using lines XSPI_CS0# (XSPI_CS1#), XSPI_SPCLK, and XSPI_IO7 to XSPI_IO0.

The write operation for OctaRAM has latency cycles before data is written to memory.

Figure 29.3 to **Figure 29.5** show waveforms of the write operation in the OctaFlash interface, and **Figure 29.6** shows a waveform of the write operation in the OctaRAM interface.

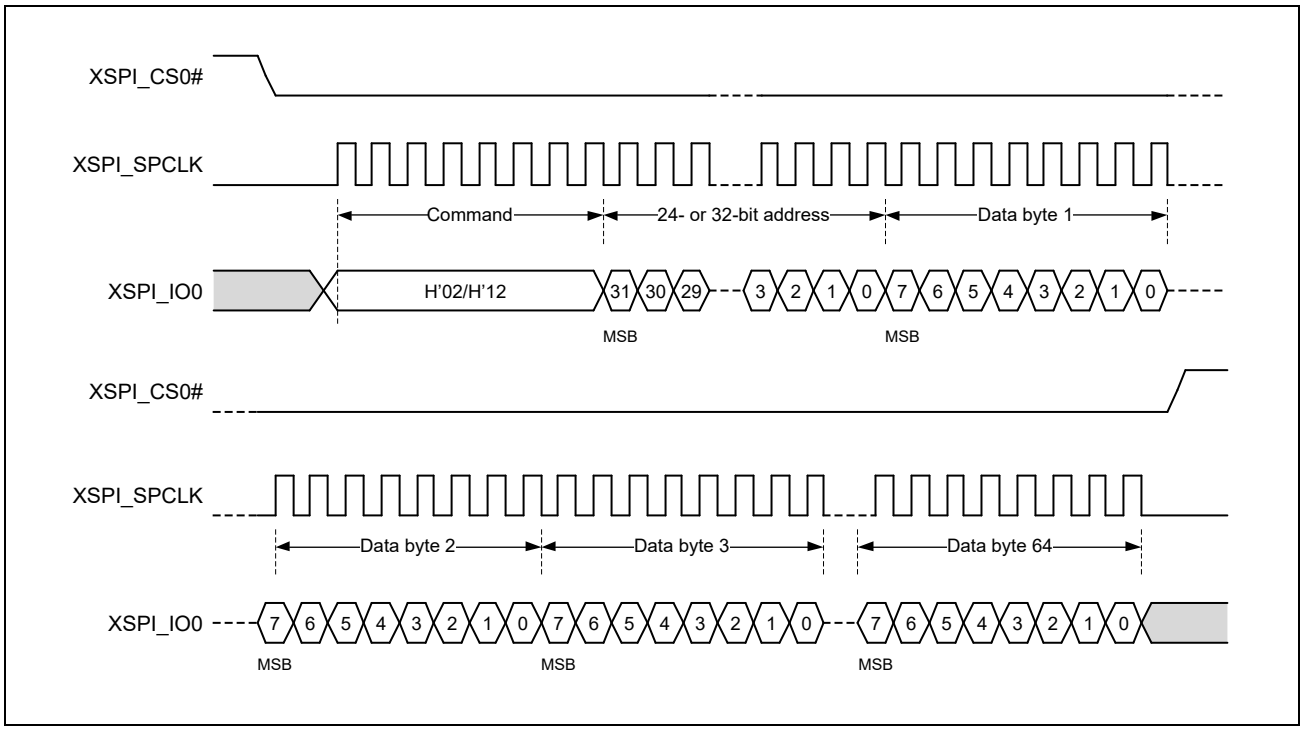


Figure 29.3 Waveform of Write Operation in OctaFlash Interface (SPI Mode)

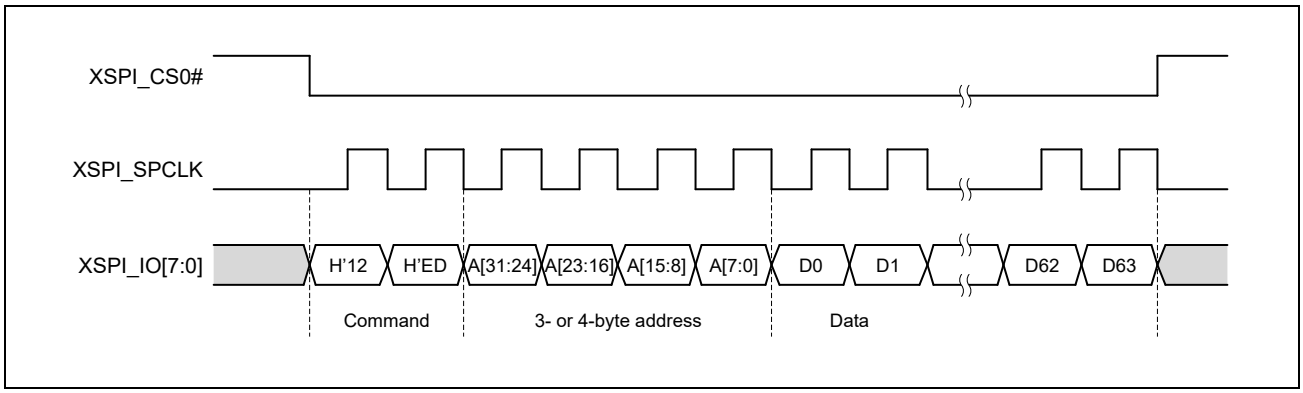


Figure 29.4 Waveform of Write Operation in OctaFlash Interface (SOPI Mode)

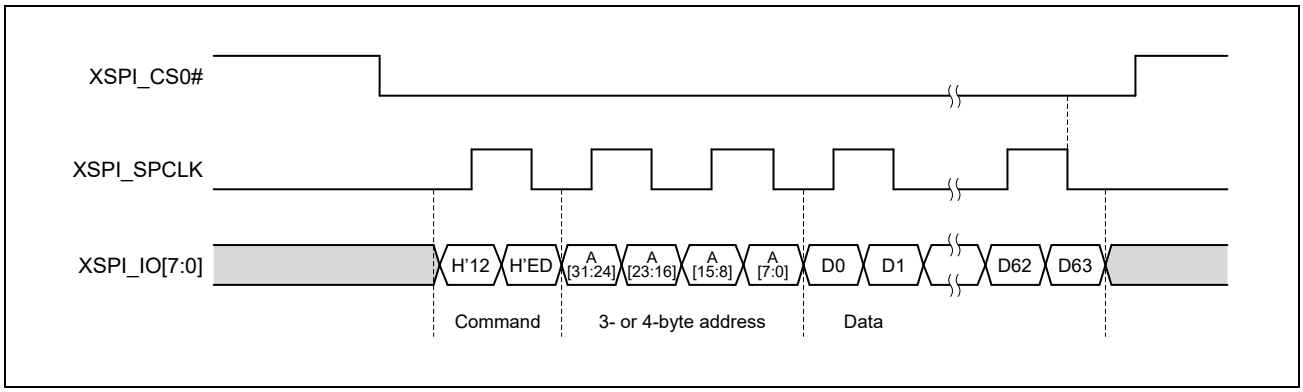


Figure 29.5 Waveform of Write Operation in OctaFlash Interface (DOPI Mode)

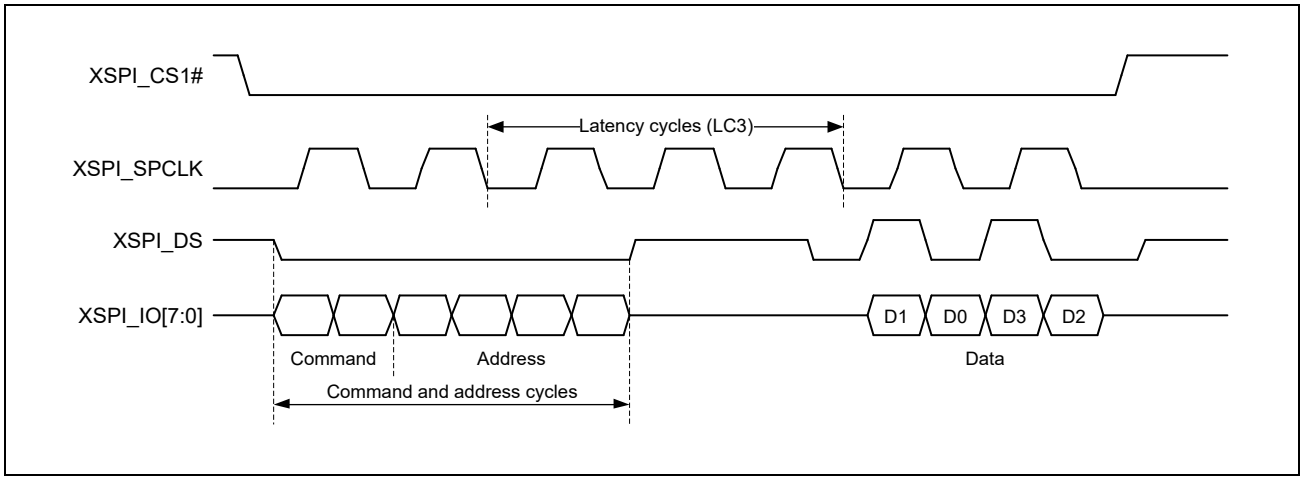


Figure 29.6 Waveform of Write Operation in OctaRAM Interface

29.5.3.2 Read Operation

The Octa memory controller outputs a command and an address and reads one or more data bytes by using lines XSPI_CS0# (XSPI_CS1#), XSPI_SPCLK, and XSPI_IO7 to XSPI_IO0.

The read operation for OctaFlash and OctaRAM has latency cycles before data is read from memory.

The number of latency cycles should be specified in the memory map dummy length setting register (MDLR) (see **Section 29.4.14**) in accordance with the setting in the memory device.

Figure 29.7 to **Figure 29.9** show waveforms of the read operation in the OctaFlash interface, and **Figure 29.10** shows a waveform of the read operation in the OctaRAM interface.

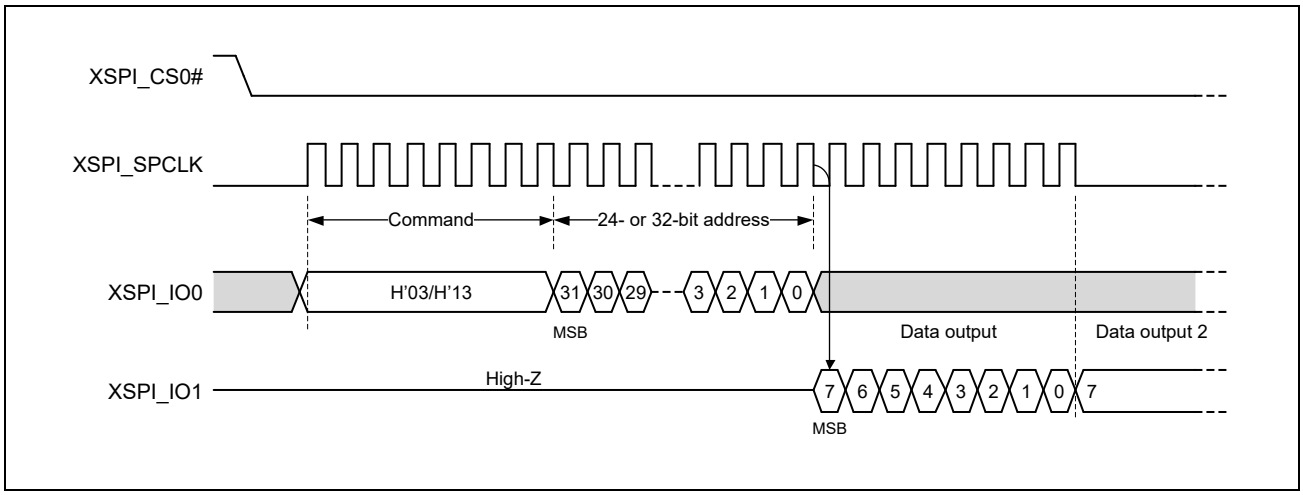


Figure 29.7 Waveform of Read Operation in OctaFlash Interface (SPI Mode)

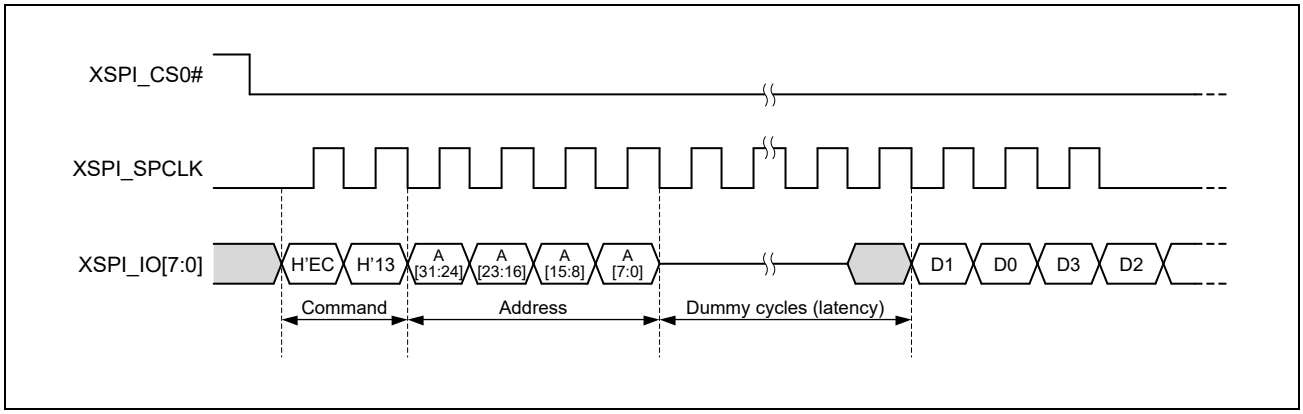


Figure 29.8 Waveform of Read Operation in OctaFlash Interface (SOPI Mode)

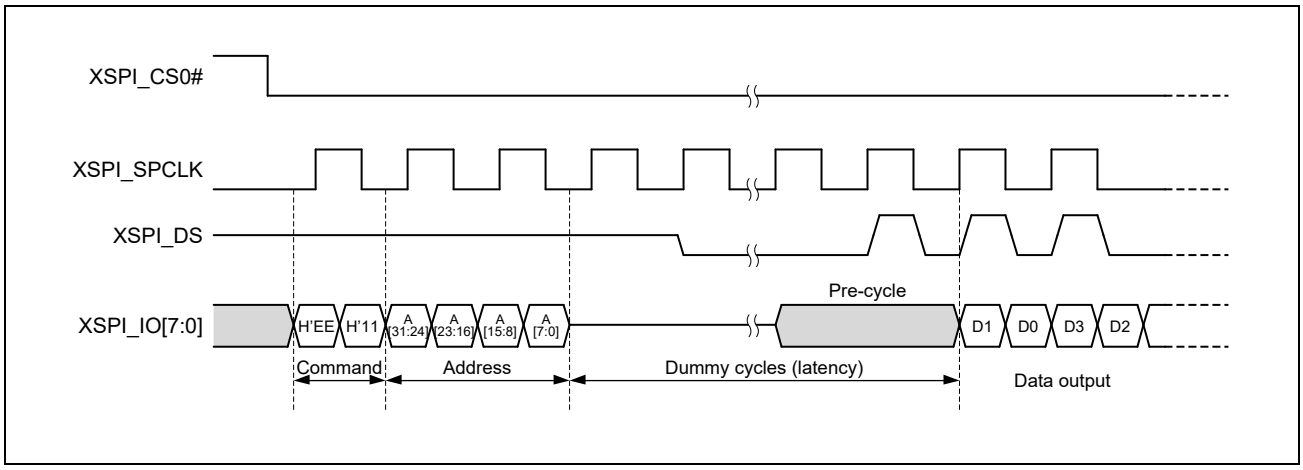


Figure 29.9 Waveform of Read Operation in OctaFlash Interface (DOPI Mode with Pre-cycle Enabled)

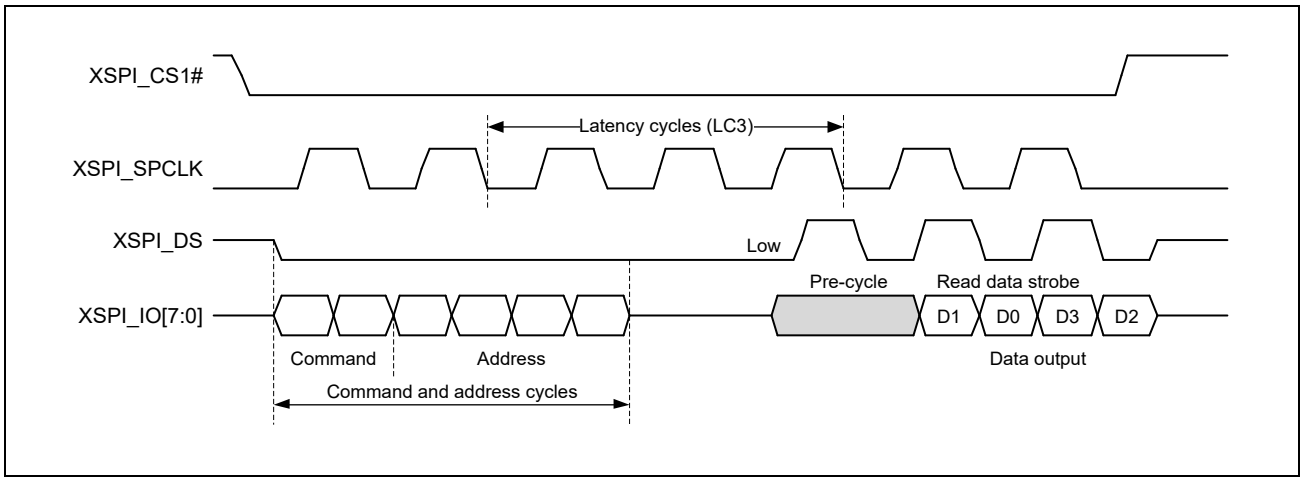


Figure 29.10 Waveform of Read Operation in OctaRAM Interface (with Pre-cycle Enabled)

29.5.4 Data Alignment in the OctaFlash and OctaRAM Spaces

Figure 29.11 shows the memory data alignment in the OctaFlash and OctaRAM spaces.

Figure 29.12 shows the data alignment when the configuration register in OctaFlash or OctaRAM is accessed.

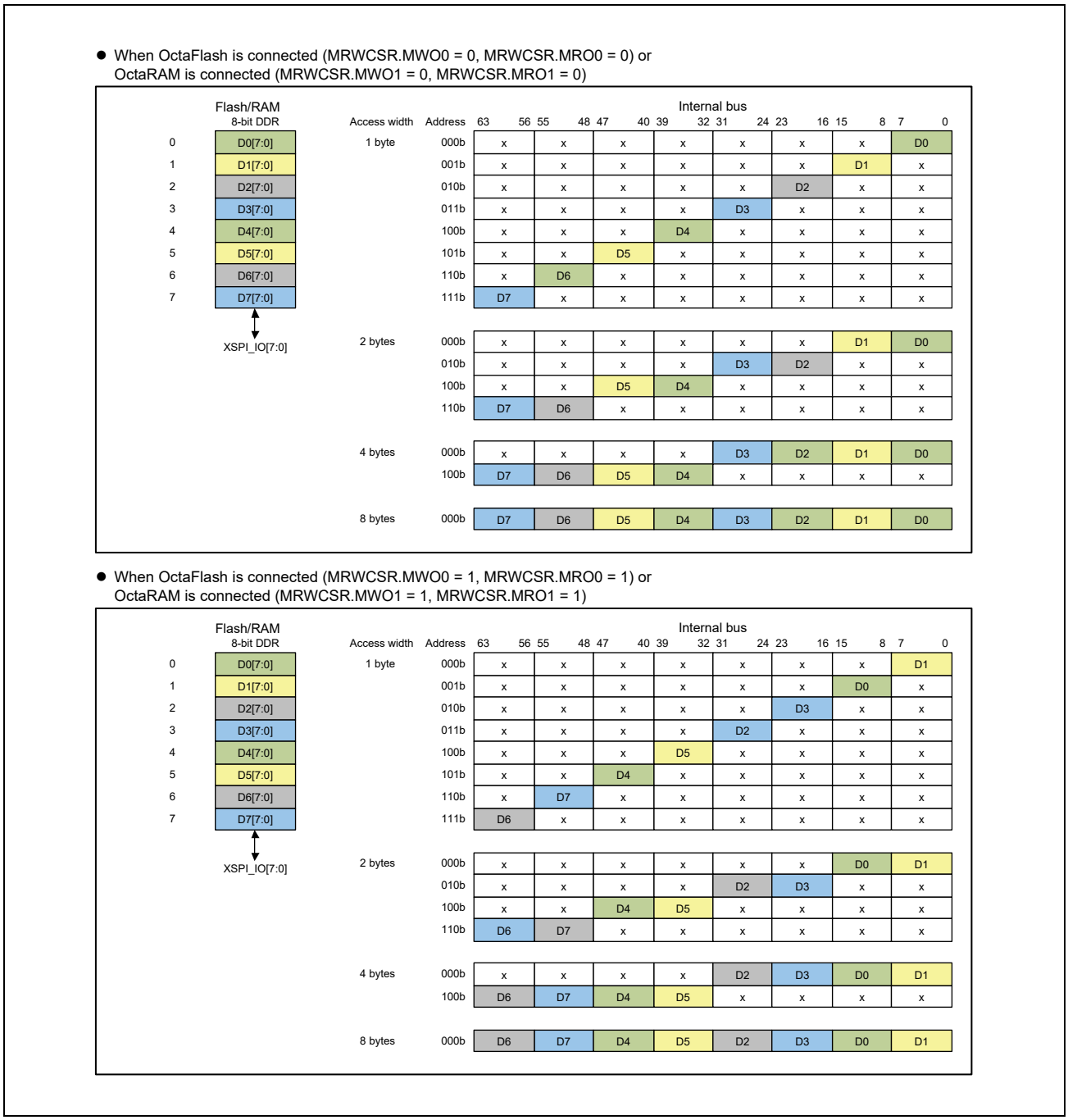


Figure 29.11 Memory Data Alignment in the OctaFlash and OctaRAM Spaces

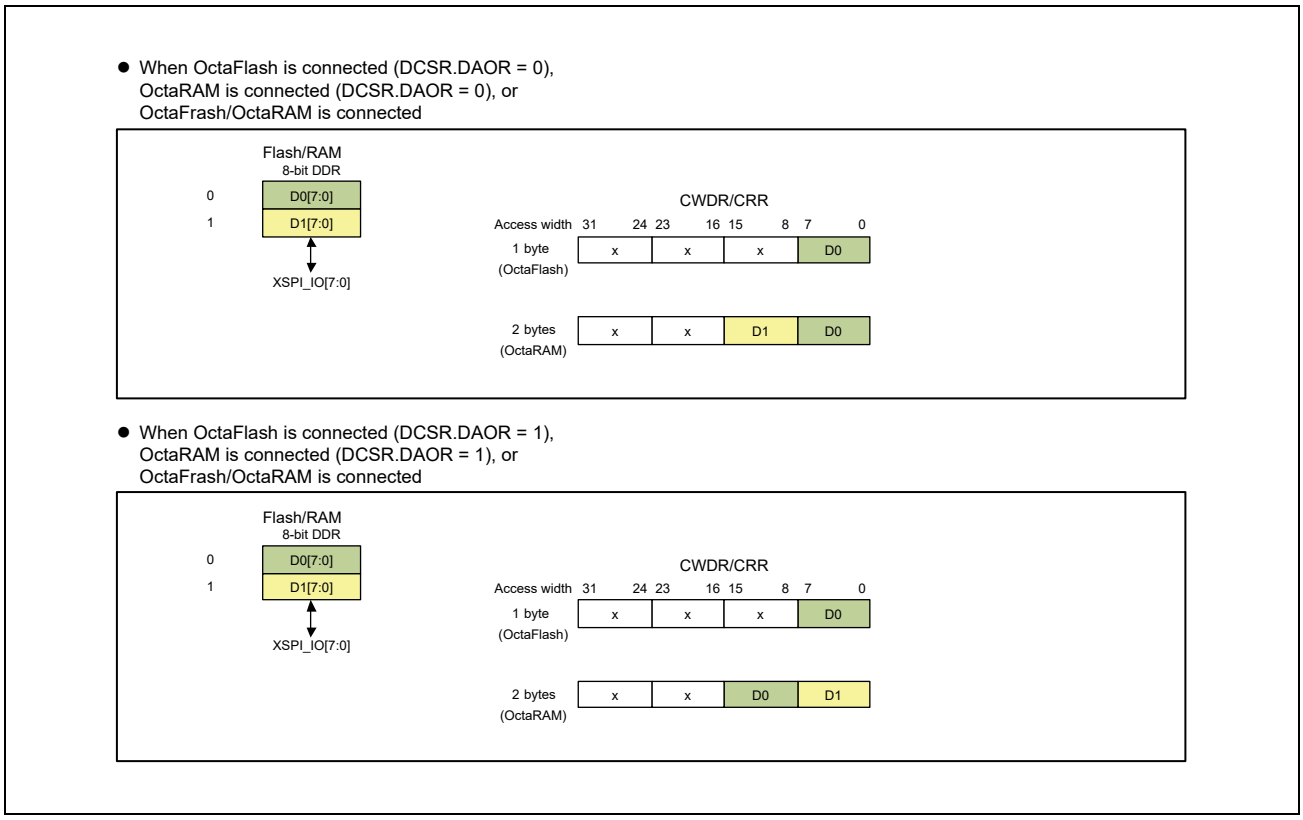


Figure 29.12 Configuration Register Data Alignment in the OctaFlash and OctaRAM Spaces

29.5.5 One-Byte Write Access to an Octa Memory Device in the DOPI Mode

This section describes one-byte write access in the DOPI mode.

Writing to the Octa memory device in the DOPI mode is handled in 2-byte units.

When the target device is OctaRAM, the output XSPI_DS signal of the controller can be used to mask bytes by driving it to the high level during the operation of writing. This allows the masking of invalid write data.

When the target device is OctaFlash, one-byte write access must not be attempted since OctaFlash only supports two byte access.

Figure 29.13 shows the waveforms in one-byte write access in the DOPI mode.

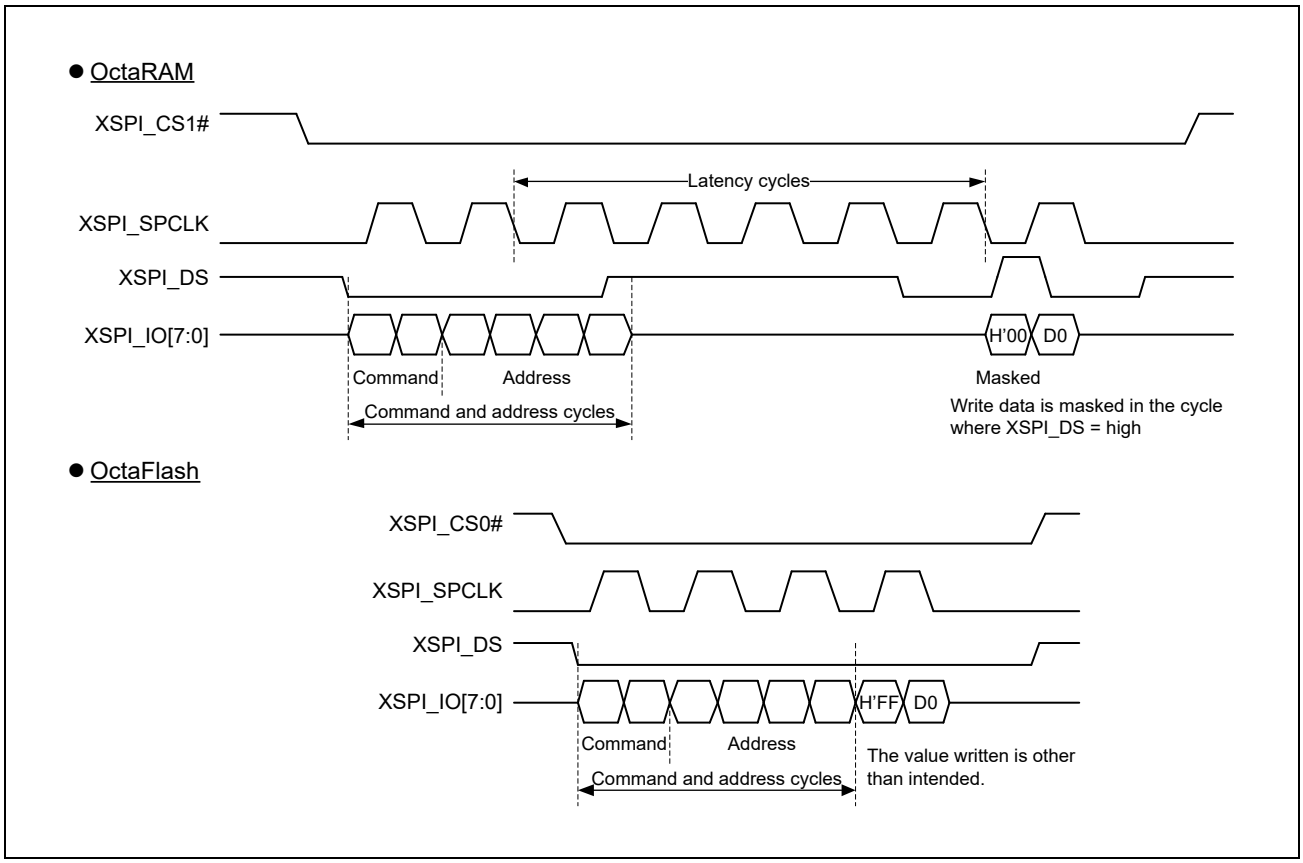


Figure 29.13 Waveforms in One-Byte Write Access in the DOPI Mode

29.5.6 Operation Flows

This section describes the flows of Octa memory controller operations for various purposes.

The following only shows sample settings, and appropriate commands and data should be specified in accordance with the specifications of the target memory device.

The sample settings assume the following environment.

(1) Memory connections: CS0 (device 0) to OctaFlash and CS1 (device 1) to OctaRAM

(2) XSPI_SPCLK (Octa memory clock) frequency: 100 MHz

29.5.6.1 Initial Settings

The following shows an example of initial settings of the Octa memory controller.
Before access to the external memory, set up the SEL_SPI_OCTA bit in the xSPI/Octa Controller select register (SYS_IPCONT_SEL_SPI_OCTA) in the initial setting steps.

Table 29.6 Example of Initial Settings of the Octa Memory Controller

Step	Read/Write	Address (Register)	Data	Data Length (Bytes)	Description
1	Write	DSR0	H'0800_0000	4	Specifies the type and size of device 0.
2	Write	DSR1	H'4080_0000	4	Specifies the type and size of device 1.
3	Write	MDTR	H'0740_A640	4	Adjusts the memory access timings.
4	Write	DRCSTR	H'2900_2900	4	Specifies the read timing for each device.
5	Write	DWCSTR	H'1A00_1A00	4	Specifies the write timing for each device.
6	Write	DCSTR	H'0000_2900	4	Specifies the timings of the chip select signals.
7	Write	CDSR	H'0000_000A	4	Sets up the controller functions. (1) Memory pre-cycle is enabled or disabled. (2) Transfer type is specified (SPI, SOPI, or DOPI).
8	Write	MDLR	H'0505_0E0E	4	Specifies number of dummy cycles (latency).
9	Write	MRWCR0	H'12ED_EE11	4	Specifies the read and write commands for device 0.
10	Write	MRWCR1	H'2000_A000	4	Specifies the read and write commands for device 1.
11	Write	MRWCSR	H'5454_5454	4	Sets up the memory-mapped reading and writing. (1) The length (bytes) of a command is specified. (2) The length (bytes) of an address is specified. (3) The order of bytes to be written is specified.

Note: Set up the SEL_SPI_OCTA bit in the xSPI/Octa Controller select register (SYS_IPCONT_SEL_SPI_OCTA).
For the SEL_SPI_OCTA bit, refer to **Section 45, General Purpose Input Output Port (GPIO)**.

29.5.6.2 Basic Operation Settings

This section describes the basic operations of OctaFlash and OctaRAM.

The following only shows sample settings, and appropriate commands and data should be specified in accordance with the specifications of the target memory device.

(1) OctaFlash Operation Settings

Table 29.7 Write Enable Settings

Step	Read/ Write	Register	SPI Mode		OPI Mode		Description
			Data	Data Length (Bytes)	Data	Data Length (Bytes)	
1	Write	DCR	H'0000_0006	4	H'0000_06F9	4	
2	Write	DCSR	H'0010_0000	4	H'0020_0000	4	
3	Write	CWNDR	Any value	4	Any value	4	

Table 29.8 Read Status Register

Step	Read/ Write	Register	SPI Mode		OPI Mode		Description
			Data	Data Length (Bytes)	Data	Data Length (Bytes)	
1	Write	DCR	H'0000_0005	4	H'0000_05FA	4	
2	Write	DAR	—	—	H'0000_0000	4	
3	Write	DCSR	H'0010_0001	4	H'0C20_0401	4	
4	Read	CRR	H'0000_00xx	1	H'0000_00xx	1	The high-order 24 bits are all set to 0s, and the read data is stored in the low order eight bits.

Table 29.9 Read Configuration Register

Step	Read/ Write	Register	SPI Mode		OPI Mode		Description
			Data	Data Length (Bytes)	Data	Data Length (Bytes)	
1	Write	DCR	H'0000_0015	4	H'0000_15EA	4	
2	Write	DAR	—	—	H'0000_0001	4	
3	Write	DCSR	H'0010_0001	4	H'0C20_0401	4	
4	Read	CRR	H'0000_00xx	1	H'0000_00xx	1	The high-order 24 bits are all set to 0s, and the read data is stored in the low order eight bits.

Table 29.10 Write Configuration Register

Step	Read/ Write	Register	SPI Mode		OPI Mode		Description
			Data	Data Length (Bytes)	Data	Data Length (Bytes)	
1	Write	DCR	—	—	H'0000_01FE	4	
2	Write	DAR	—	—	H'0000_0001	4	
3	Write	DCSR	—	—	H'0420_0001	4	
4	Write	CWDR	—	—	H'0000_00xx	1	Set all the high-order 24 bits to 0s, and specify the write data in the low-order eight bits.

Table 29.11 Read Configuration Register 2

Step	Read/ Write	Register	SPI Mode		OPI Mode		Description
			Data	Data Length (Bytes)	Data	Data Length (Bytes)	
1	Write	DCR	H'0000_0071	4	H'0000_718E	4	
2	Write	DAR	Address	4	Address	4	
3	Write	DCSR	H'0410_0001	4	H'0C20_0401	4	
4	Read	CWDR	H'0000_00xx	1	H'0000_00xx	1	The high-order 24 bits are all set to 0s, and the read data is stored in the low order eight bits.

Table 29.12 Write Configuration Register 2

Step	Read/ Write	Register	SPI Mode		OPI Mode		Description
			Data	Data Length (Bytes)	Data	Data Length (Bytes)	
1	Write	DCR	H'0000_0072	4	H'0000_728D	4	
2	Write	DAR	Address	4	Address	4	
3	Write	DCSR	H'0410_0001	4	H'0420_0001	4	
4	Write	CWDR	H'0000_00xx	1	H'0000_00xx	1	Set all the high-order 24 bits to 0s, and specify the write data in the low-order eight bits.

Table 29.13 Memory-Map Write and Read Command Settings

Step	Read/ Write	Register	SPI Mode		OPI Mode		Description
			Data	Data Length (Bytes)	Data	Data Length (Bytes)	
1	Write	MRWCR0	H'0012_0013	4	H'12ED_EE11	4	
2	Write	MRWCSR	H'0000_0C0C	4	H'0000_5454	4	

Table 29.14 Erase Sector

Step	Read/ Write	Register	SPI Mode		OPI Mode		Description
			Data	Data Length (Bytes)	Data	Data Length (Bytes)	
1	Write	DCR	H'0000_0021	4	H'0000_21DE	4	
2	Write	DAR	Address to be erased	4	Address to be erased	4	
3	Write	DCSR	H'0410_0000	4	H'0420_0000	4	
4	Write	CWNDR	Any value	4	Any value	4	

(2) OctaRAM Operation Settings

Table 29.15 Read Configuration Register

Step	Read/ Write	Register	Data	Data Length (Bytes)	Description
1	Write	DCR	H'0000_C000	4	
2	Write	DAR	H'0004_0000	4	
3	Write	DCSR	H'04A8_0502	4	
4	Read	CRR	H'0000_xxxx	2	The high-order 16 bits are all set to 0s, and the read data is stored in the low-order 16 bits.

Table 29.16 Write Configuration Register

Step	Read/ Write	Register	Data	Data Length (Bytes)	Description
1	Write	DCR	H'0000_4000	4	
2	Write	DAR	H'0004_0000	4	
3	Write	DCSR	H'04A8_0002	4	
4	Write	CWDR	H'0000_xxxx	2	Set all the high-order 16 bits to 0s, and specify the write data in the low-order 16 bits.

29.6 Delaying XSPI_DS and Auto-Calibration

29.6.1 Delaying XSPI_DS

Depending on the characteristics of the DOPI mode, the controller might not be able to latch data on the rising and falling edges of the XSPI_DS clock signal. XSPI_DS will then require a delay to allow the latching of data with sufficient margins for the setup and hold times. The DV1DEL[7:0] and DV0DEL[7:0] bits in the memory delay trimming register (MDTR) can be used to specify the amount of delay.

Figure 29.14 shows how a delayed XSPI_DS becomes asserted at a good time for latching of the data.

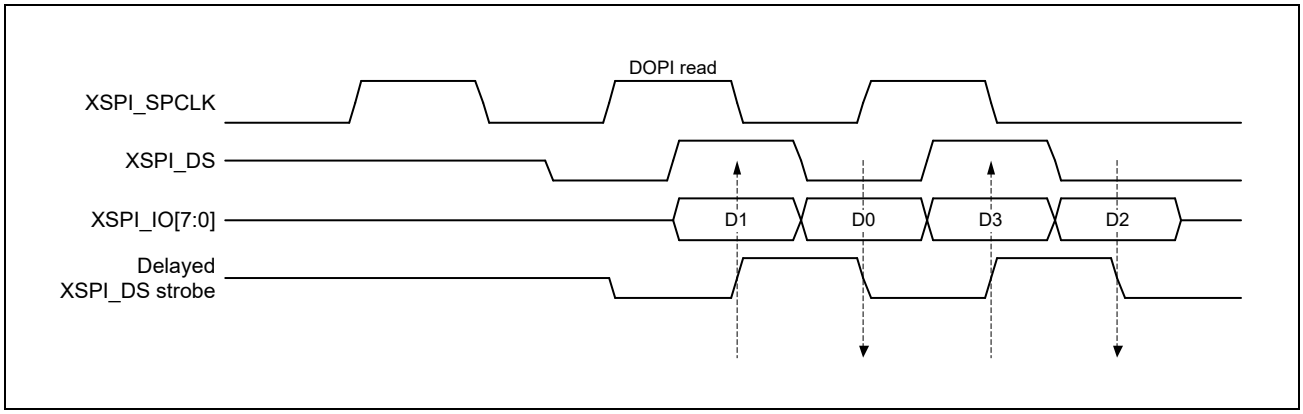


Figure 29.14 Example of Securing the Latching of Data by Delaying XSPI_DS

29.6.2 XSPI_DS Auto-Calibration

This LSI has an automatic calibration to prevent acquisition of wrong data due to the variation of voltage and temperature. When using the automatic calibration, set the following registers.

- Controller and device setting register (CDSR): Automatic calibration mode ACMODE[1:0] bits setting
- Auto-calibration timer register (ACTR): Automatic calibration cycle time setting
- Auto-calibration address registers 0 and 1 (ACAR0/ACAR1): Preamble pattern store address setting

Automatic calibration requires 16 bytes user area to store the preamble pattern. Store the preamble pattern to the setting address of ACAR0 and ACAR1.

Preamble pattern:

H’00F708F7_FF0000F7_0800FF00_FFFF0000
When MRWCSR.MRO1 or MRWCSR.MRO0 is 0,

H’00F708F7_FF0000F7_0800FF00_FFFF0000
When MRWCSR.MRO1 or MRWCSR.MRO0 is 1,

H’F700F708_00FFF700_000800FF_FFFF0000
store in the flash memory or RAM device.

The operation proceeds as follows when auto-calibration is enabled.

Once the time specified in ACTR has elapsed, the Octa memory controller varies the delay time for XSPI_DS and checks whether the preamble pattern can correctly be read from the address specified in ACAR0 or ACAR1.

The preamble pattern is read 48 times per auto-calibration.

The controller seeks the most appropriate timing of latching in this sequence of reading and updates the DV0DEL or DV1DEL bits in MDTR with the result. This re-starts the timer for auto-calibration.

Note that access to the OctaFlash memory or OctaRAM area by the CPU or other bus masters is held pending until reading of the preamble data for auto-calibration is completed.

Figure 29.15 describes the flow of automatic calibration settings for OctaFlash and OctaRAM devices.

Before starting settings, check that the Octa memory controller is selected in the SEL_SPI_OCTA bit in the xSPI/Octa Controller select register (SYS_IPCONT_SEL_SPI_OCTA).

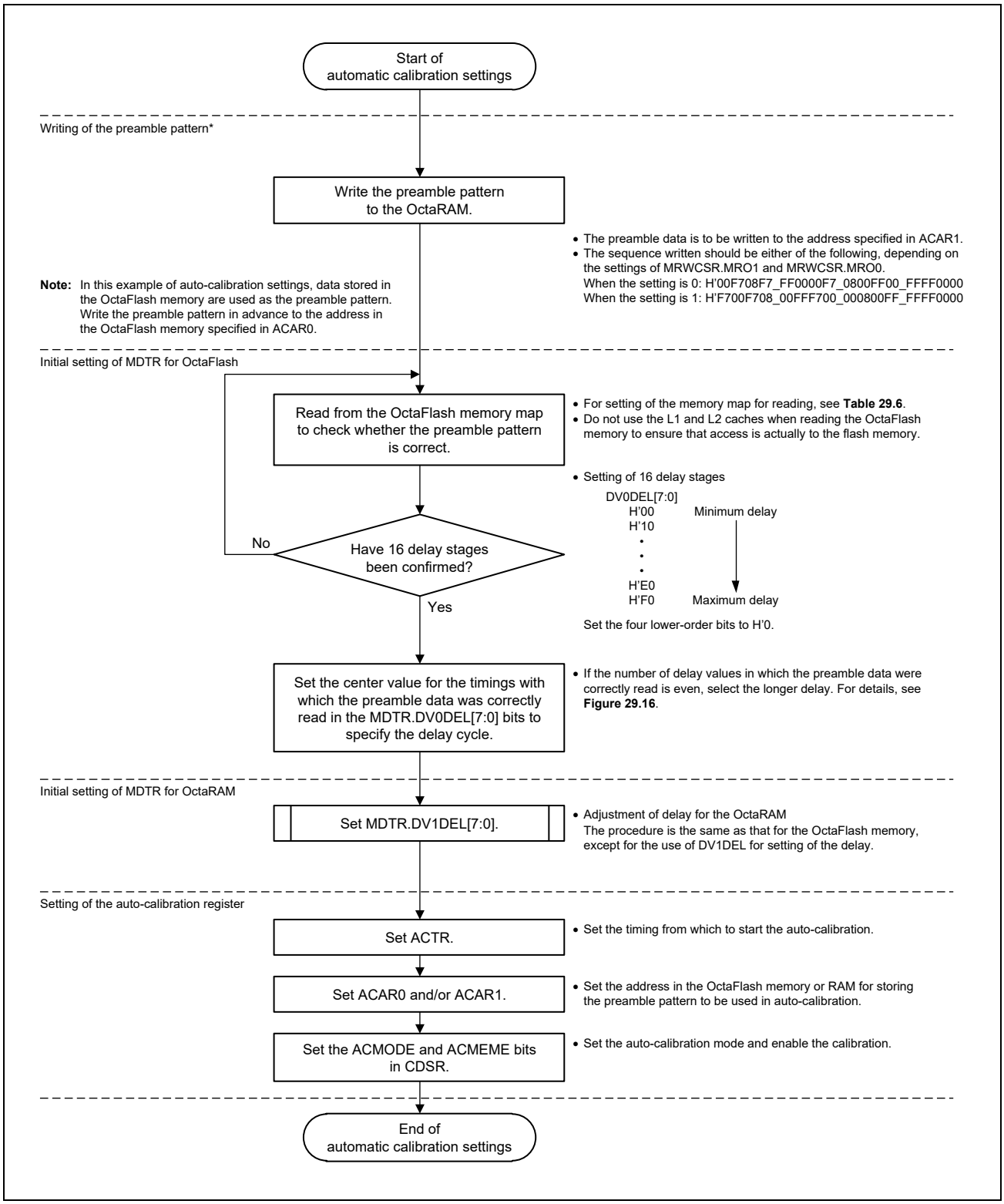


Figure 29.15 Flow of Settings for the Auto-Calibration Mode

Figure 29.16 shows an example of the initial setting of the DV0DEL or DV1DEL Bits in MDTR.

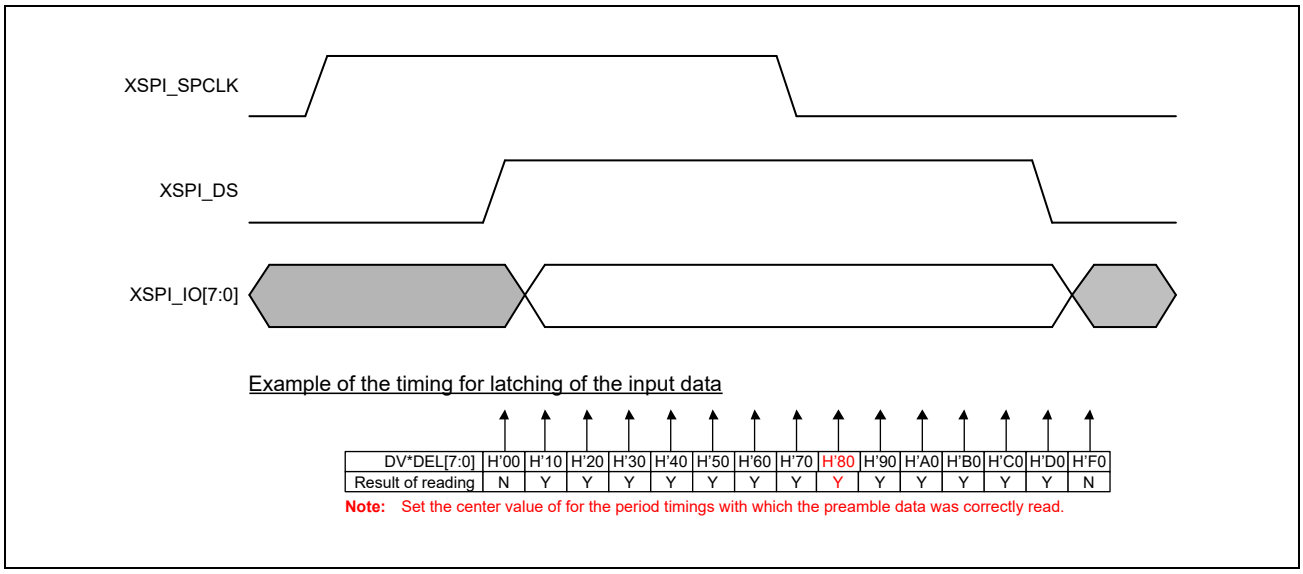


Figure 29.16 Example of the Initial Setting of the DV0DEL or DV1DEL Bits in MDTR

29.7 XSPI_DS Enable Counter

During read command operation, the XSPI_DS signal is used as the data latch clock by the controller. When the memory device is in the idle state, XSPI_DS maintains a high impedance (Hi-Z) state, and asserts low before data is output. This Hi-Z to low transition may latch wrong data by driving the XSPI_DS data latch logic of the controller.

Figure 29.15 shows the waveform when XSPI_DS maintains a high-impedance (Hi-Z) state and asserts low before data is output.

Avoiding such Hi-Z to low transitions, this controller can bypass wrong data with memory delay trim register (MDTR). (MDTR.DQSEDOPI[3:0] for Flash DOPI mode, MDTR.DQSESOPI[3:0] for Flash SOPI mode, and MDTR.DQSERAM[3:0] for RAM mode)

Each of the bit fields can be set within the range from 0 to 15.

XSPI_DS enable counter (MDTR) minimum value ≥ Number of the command cycles + number of the address cycles*1 + number of the stable XSPI_DS cycles*2 + number of the XSPI_DS delay cycles*3

XSPI_DS enable counter (MDTR) maximum value ≤ Number of the command cycles + number of the address cycles*1 + number of the memory dummy cycles (latency) + number of the XSPI_DS delay cycles*3 + pre-cycle setting*4

Note 1. Flash SOPI: Number of the command cycles + number of the address cycles = 2 + 4 = 6
Flash DOPI: Number of the command cycles + number of the address cycles = 1 + 2 = 3
RAM DOPI: number of command cycles + number of address cycles = 1 + 2 − 1 = 2 (start dummy (latency) calculation from the second cycle)

Note 2. Number of stable XSPI_DS cycles: Maximum value is 2 cycles of XSPI_SPCLK (Reference specification of OctaFlash/OctaRAM)

Note 3. Number of the XSPI_DS delay cycles: XSPI_DS clock delay due to external causes
When XSPI_DS delay ≤ 0.25 XSPI_SPCLK: XSPI_DS delay cycles = −1
When 0.25 XSPI_SPCLK < XSPI_DS delay ≤ 1.25 XSPI_SPCLK: XSPI_DS delay cycles = 0
When 1.25 XSPI_SPCLK < XSPI_DS delay ≤ 2.25 XSPI_SPCLK: XSPI_DS delay cycles = 1
In the flash SOPI mode, read 0.25 as 0.5, 1.25 as 1.5, and 2.25 as 2.5 in the above expressions.

Note 4. Pre-cycle setting: XSPI_DS pre-cycle on/off
XSPI_DS pre-cycle off: Pre-cycle setting = 0
XSPI_DS pre-cycle on: Pre-cycle setting = −1

Setting example: Flash DOPI mode, dummy cycles (latency) = 4, pre-cycle is off

Minimum value of XSPI_DS enable counter (MDTR) ≥ 1 + 2 + 2 + 0 = 5
Maximum value of XSPI_DS enable counter (MDTR) ≤ 1 + 2 + 4 + 0 = 7

From the above, setting MDTR .DQSEDOPI[3:0] from 5 to 7 can latch the correct data. If MDTR is too small, wrong data may be latched due to a transition from Hi-Z to Low. If MDTR is too large, no data is latched.

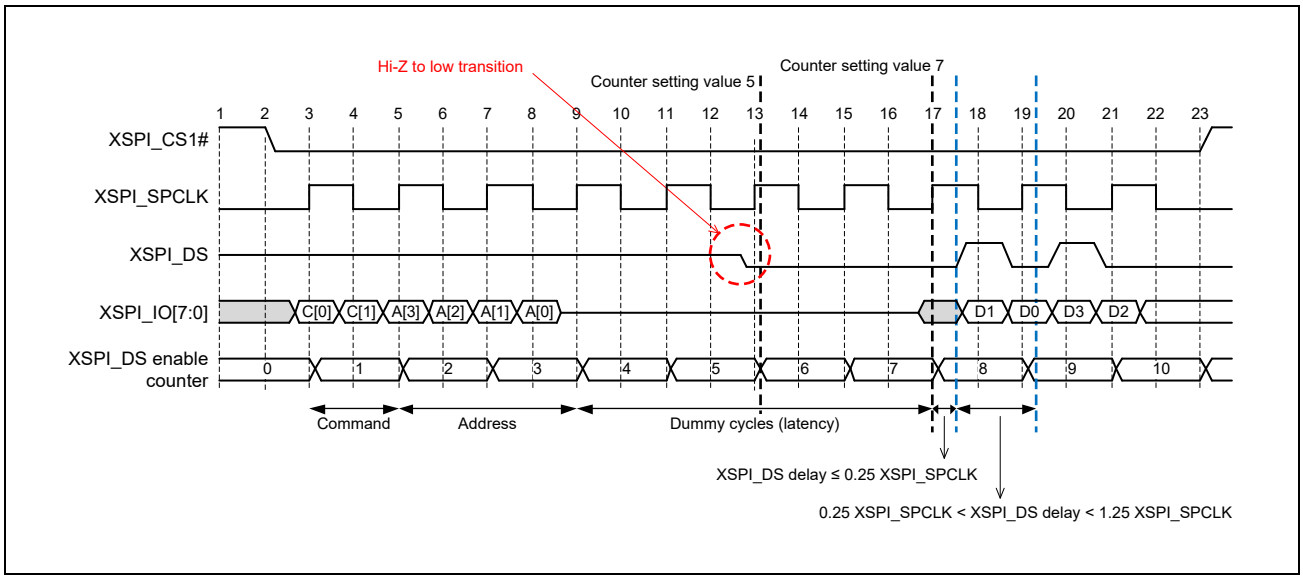


Figure 29.17 Timing of the XSPI_DS Enable Counter in DOPI Mode for the Flash Memory with 4 as the Number of Dummy Cycles (Latency), and XSPI_DS Pre-Cycle Off

30. CAN-FD Interface (CAN-FD)

30.1 Overview

The LSI has a 2-channel CAN-FD module (CAN-FD) that complies with ISO 11898-1 (2015) Standards. CAN-FD transmits and receives both formats of messages, namely the standard identifier (11 bits) (identifier is hereafter referred to as ID) and extended ID (29 bits). **Table 30.1** lists the specifications of the CAN-FD, **Figure 30.1** shows a block diagram of the CAN-FD, and **Table 30.2** lists the I/O pins.

Table 30.1 CAN-FD Specifications (1/2)

Parameter	Description
Number of channels	Two channels
Protocol	CAN-FD ISO 11898-1 (CD2015) compliant
Communication speed	• CAN-FD mode Nominal bit rate: Max. 1 Mbps Data bit rate: Max. 8 Mbps
Buffer	Total 192 buffers (when frame size is 76 bytes) • Individual buffers: 64 buffers (32 buffers × 2 channels) Transmit buffer: 32 buffers per channel Transmit queue: 4 queues per channel (shared with transmit buffer, up to 16 buffers allocatable) • Shared buffers: 128 buffers for all channels Receive buffers: Up to 32 RXMB Receive FIFO buffer: 8 FIFO Buffers Common FIFO buffer: Up to 6 FIFO Buffers individually configurable as RX FIFO/TX FIFO/ Gateway FIFO
Reception function	• Receives data frames and remote frames • Selects ID format (standard ID, extended ID, or both IDs) to be received • Sets interrupt enable/disable for each FIFO • Mirror function (reception of messages transmitted from the CAN node itself) • Timestamp function (to record message reception time as a 16-bit timer value)
Reception filter function	• Selects receive message according to 128 receive rules • Sets the number of receive rules (0 to 128) for each channel • Acceptance filter processing: Sets ID and mask for each receive rule • DLC filter processing: Enables DLC filter check for each acceptance rule
Receive message transfer function	• Routing function Transfers receive messages to arbitrary destinations (can be transferred to up to 8 buffers) Transfer destination: Receive buffer, receive FIFO buffer, and/or transmit/receive FIFO buffer • Label addition function Stores label information together with a message in a receive buffer and FIFO buffer
Transmission function	• Transmits data frames and remote frames • Selects ID format (standard ID, extended ID, or both IDs) to be transmitted • Sets interrupt enable/disable for each transmit buffer and transmit/receive FIFO buffer • Selects ID priority transmission or transmit buffer number priority transmission • Transmit request can be aborted (possible to confirm with a flag) • One-shot transmission function
Interval transmission function	Transmit message at configurable intervals (transmit mode or gateway mode of transmit/receive FIFO buffers)
Transmit queue function	Transmits all stored messages according to the ID priority
Transmit history function	Stores the history information of transmission-completed messages Adds timestamp (recording message transmission time as a 16-bit timer value) to the history information
Gateway function	Transmits a received message automatically

Table 30.1 CAN-FD Specifications (2/2)

Parameter	Description
Bus off recovery mode selection	Selects the method for returning from bus-off state. • ISO 11898-1 compliant • Automatic entry to channel halt mode at bus-off entry • Automatic entry to channel halt mode at bus-off end • Transition to channel halt mode by program request • Transition to the error-active state by program request (forcible return from the bus-off state)
Error status monitoring	• Monitors CAN protocol errors (stuff error, from error, ACK error, CRC error, bit error, ACK delimiter error, and bus dominant lock) • Defects error status transitions (error warning, error passive, bus-off entry, and bus-off recovery) • Reads the error counter • Monitors DLC errors
Interrupt source	8 interrupt sources • Global interrupts (2 sources) RX FIFO interrupt Global error interrupt • Channel interrupts (3 sources/channel) CANm transmit interrupt (m = 0, 1) – CANm transmit complete interrupt – CANm transmit abort interrupt – CANm transmit/receive FIFO transmit complete interrupt (in transmit mode, gateway mode) – CANm transmit history interrupt – CANm transmit queue interrupt CANm transmit/receive FIFO receive complete interrupt (in receive mode, gateway mode) CANm error interrupt 10 DMA requests • 8 for RX FIFO • 2 for the first common FIFO (1 source/channel)
CAN clock source	Selects the PCLKM/2 (40 MHz) or the PCLKCAN (Less than 80 MHz)
Test function	Test function for user evaluation • Listen-only mode • Self-test mode 0 (external loopback) • Self-test mode 1 (internal loopback) • Restricted operation mode • RAM test (read/write test) • Inter-channel communication test (CRC error test enabled)
Power down function	Module start stop function for each CAN node (Channel & Global Sleep Mode)
Bus traffic measurement	CAN bus traffic measurement of each Channel is possible

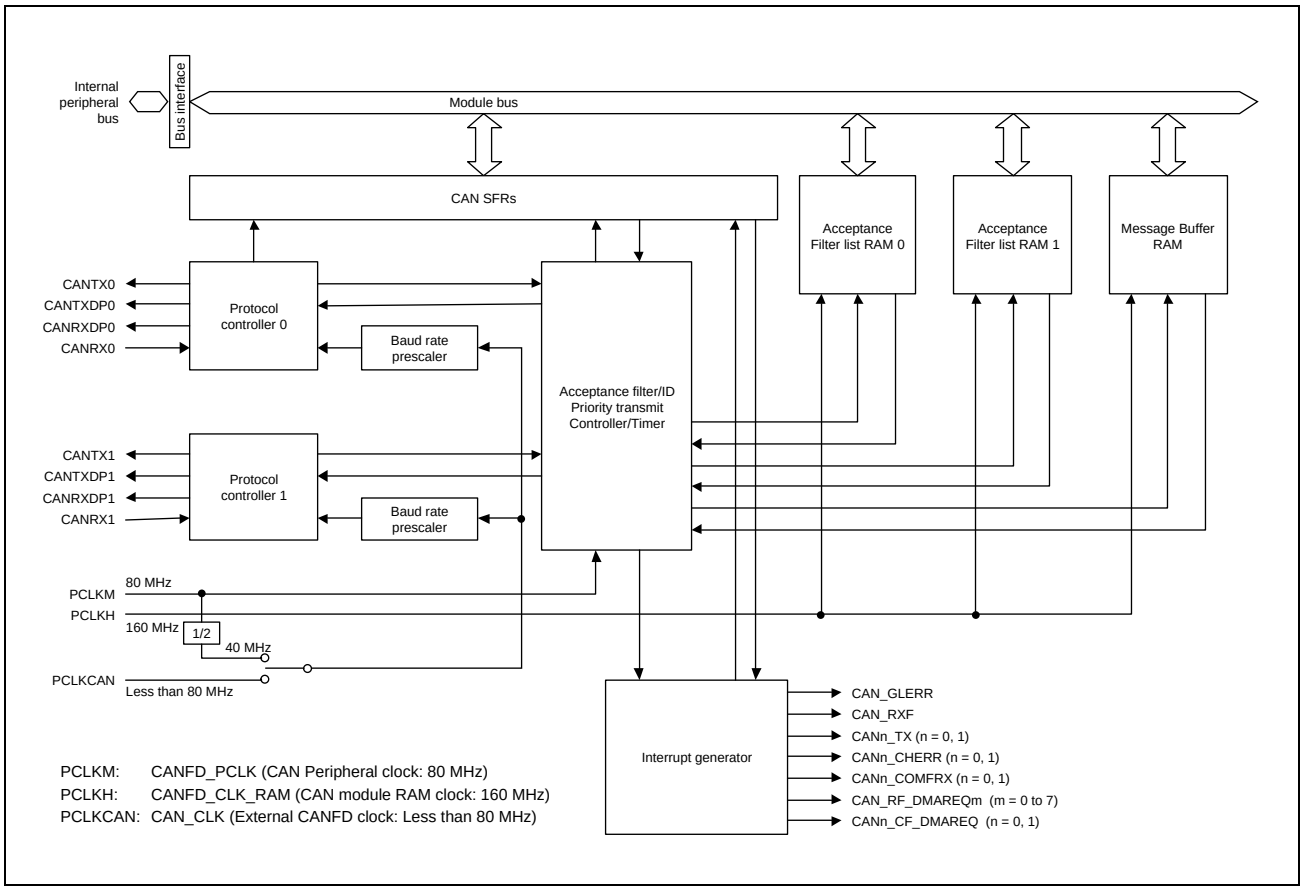


Figure 30.1 Block Diagram of CAN-FD

Table 30.2 lists the input/output pins of the CAN-FD module.

Table 30.2 Pin Configuration of CAN-FD

Channel	Pin Name	I/O	Function
CAN0	CAN0_RX	Input	CAN0 receive data input (CANRX0)
	CAN0_TX	Output	CAN0 transmit data output (CANTX0)
	CAN0_RX_DATARATE_EN	Output	CAN0 receive data phase output (CANRXDP0)
	CAN0_TX_DATARATE_EN	Output	CAN0 transmit data phase output (CANTXDP0)
CAN1	CAN1_RX	Input	CAN1 receive data input (CANRX1)
	CAN1_TX	Output	CAN1 transmit data output (CANTX1)
	CAN1_RX_DATARATE_EN	Output	CAN1 receive data phase output (CANRXDP1)
	CAN1_TX_DATARATE_EN	Output	CAN1 transmit data phase output (CANTXDP1)

Table 30.3 CAN-FD Interrupt Sources

Name	Interrupt Sources	Cortex-A55 Request	Cortex-M33/ Cortex-M33_ FPU Request	DMAC Activation
CAN_RXF	RX FIFO interrupt (INTRCANGRECC)	Possible	Possible	Not possible
CAN_GLERR	Global error interrupt (INTRCANGERR)	Possible	Possible	Not possible
CANn_TX	Channel n TX interrupt (INTRCANnTRX)	Possible	Possible	Not possible
CANn_CHERR	Channel n CAN error interrupt (INTRCANnREC)	Possible	Possible	Not possible
CANn_COMFRX	Channel n Common RX FIFO or TXQ interrupt (INTCANnERR)	Possible	Possible	Not possible
CAN_RF_DMAREQm	RX FIFO m DMA request (RXF_DMAm)	Not possible	Not possible	Possible
CANn_CF_DMAREQ	Channel n First common FIFO DMA request (COM_DMAn)	Not possible	Not possible	Possible

Note: m = 0 to 7, n = 0, 1

30.2 Register Descriptions

Table 30.4 shows the register configuration. The CAN-FD address space is offset from the base address. The base address of CAN-FD is as follows.

CAN-FD base address: H'0_100C_0000 (Overall Address Space)
CAN-FD base address: H'400C_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
CAN-FD base address: H'500C_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

For all repetitive registers and bits, a lowercase index is used to indicate which slice is being referenced. If an index is being used, it is defined and described below the Register Table it is being used in.

There is one global index used across all the registers and bits that need it.

Table 30.4 Register Configurations (1/4)

Register Name	Symbol	Value After Reset	Offset Address N/A (after)	Access Size
Channel n Nominal Bit Rate Configuration Register (n = 0, 1)	CFDCnNCFG	H'0000_0000	H'0000 + n × H'0010	8, 16, 32
Channel n Control Registers (n = 0, 1)	CFDCnCTR	H'0000_0005	H'0004 + n × H'0010	8, 16, 32
Channel n Status Registers (n = 0, 1)	CFDCnSTS	H'0000_0005	H'0008 + n × H'0010	8, 16, 32
Channel n Error Flag Registers (n = 0, 1)	CFDCnERFL	H'0000_0000	H'000C + n × H'0010	8, 16, 32
Global IP Version Register	CFDGIPV	H'3124_0143	H'0080	8, 16, 32
Global Configuration Register	CFDGCFG	H'0000_0000	H'0084	8, 16, 32
Global Control Register	CFDGCTR	H'0000_0005	H'0088	8, 16, 32
Global Status Register	CFDGSTS	H'0000_000D	H'008C	8, 16, 32
Global Error Flag Register	CFDGERFL	H'0000_0000	H'0090	8, 16, 32
Global Timestamp Counter Register	CFDGTSC	H'0000_0000	H'0094	16, 32
Global Acceptance Filter List Entry Control Register	CFDGAFLECTR	H'0000_0000	H'0098	8, 16, 32
Global Acceptance Filter List Configuration Register 0	CFDGAFLCFG0	H'0000_0000	H'009C	8, 16, 32
RX Message Buffer Number Register	CFDRMNB	H'0000_0000	H'00AC	8, 16, 32
RX Message Buffer New Data Register 0	CFDRMND0	H'0000_0000	H'00B0	8, 16, 32
RX FIFO Configuration / Control Registers n (n = 0 to 7)	CFDRFCCn	H'0000_0000	H'00C0 + n × H'0004	8, 16, 32
RX FIFO Status Registers n (n = 0 to 7)	CFDRFSTSn	H'0000_0001	H'00E0 + n × H'0004	8, 16, 32
RX FIFO Pointer Control Registers n (n = 0 to 7)	CFDRFPCTRn	H'0000_0000	H'0100 + n × H'0004	8, 16, 32
Common FIFO Configuration / Control Registers n (n = 0 to 5)	CFDCFCCn	H'0000_0000	H'0120 + n × H'0004	8, 16, 32
Common FIFO Configuration / Control Enhancement Registers n (n = 0 to 5)	CFDCFCCEn	H'0000_0000	H'0180 + n × H'0004	8, 16, 32
Common FIFO Status Registers n (n = 0 to 5)	CFDCFSTSn	H'0000_0001	H'01E0 + n × H'0004	8, 16, 32
Common FIFO Pointer Control Registers n (n = 0 to 5)	CFDCFPCTRn	H'0000_0000	H'0240 + n × H'0004	8, 16, 32
FIFO Empty Status Register	CFDFESTS	H'FFFF_FFFF	H'02A0	8, 16, 32
FIFO Full Status Register	CFDFFSTS	H'0000_0000	H'02A4	8, 16, 32
FIFO Message Lost Status Register	CFDFMSTS	H'0000_0000	H'02A8	8, 16, 32

Table 30.4 Register Configurations (2/4)

Register Name	Symbol	Value After Reset	Offset Address N/A (after)	Access Size
RX FIFO Interrupt Flag Status Register	CFDRFISTS	H'0000_0000	H'02AC	8, 16, 32
Common FIFO RX Interrupt Flag Status Register	CFDCFRISTS	H'0000_0000	H'02B0	8, 16, 32
Common FIFO TX Interrupt Flag Status Register	CFDCFTISTS	H'0000_0000	H'02B4	8, 16, 32
Common FIFO One Frame RX Interrupt Flag Status Register	CFDCFOFRISTS	H'0000_0000	H'02B8	8, 16, 32
Common FIFO One Frame TX Interrupt Flag Status Register	CFDCFOFTISTS	H'0000_0000	H'02BC	8, 16, 32
Common FIFO Message Over Write Status Register	CFDCFMOWSTS	H'0000_0000	H'02C0	8, 16, 32
FIFO FDC Full Status Register	CFDFFFSTS	H'0000_0000	H'02C4	8, 16, 32
TX Message Buffer Control Registers n (n = 0 to 127)	CFDTMCn	H'00	H'02D0 + n × H'0001	8
TX Message Buffer Status Registers n (n = 0 to 127)	CFDTMSTSn	H'00	H'07D0 + n × H'0001	8
TX Message Buffer Transmission Request Status Register n (n = 0 to 3)	CFDTMTRSTSn	H'0000_0000	H'0CD0 + n × H'0004	8, 16, 32
TX Message Buffer Transmission Abort Request Status Register n (n = 0 to 3)	CFDTMTARSTSn	H'0000_0000	H'0D70 + n × H'0004	8, 16, 32
TX Message Buffer Transmission Completion Status Register n (n = 0 to 3)	CFDTMTCSTSn	H'0000_0000	H'0E10 + n × H'0004	8, 16, 32
TX Message Buffer Transmission Abort Status Register n (n = 0 to 3)	CFDTMTASTSn	H'0000_0000	H'0EB0 + n × H'0004	8, 16, 32
TX Message Buffer Interrupt Enable Configuration Register n (n = 0 to 3)	CFDTMIECn	H'0000_0000	H'0F50 + n × H'0004	8, 16, 32
TX Queue Configuration / Control Registers 0 n (n = 0, 1)	CFDTXQCC0n	H'0000_0000	H'1000 + n × H'0004	8, 16, 32
TX Queue Status Registers 0 n (n = 0, 1)	CFDTXQSTS0n	H'0000_0001	H'1020 + n × H'0004	8, 16, 32
TX Queue Pointer Control Registers 0 n (n = 0, 1)	CFDTXQPCTR0n	H'0000_0000	H'1040 + n × H'0004	8, 16, 32
TX Queue Configuration / Control Registers 1 n (n = 0, 1)	CFDTXQCC1n	H'0000_0000	H'1060 + n × H'0004	8, 16, 32
TX Queue Status Registers 1 n (n = 0, 1)	CFDTXQSTS1n	H'0000_0001	H'1080 + n × H'0004	8, 16, 32
TX Queue Pointer Control Registers 1 n (n = 0, 1)	CFDTXQPCTR1n	H'0000_0000	H'10A0 + n × H'0004	8, 16, 32
TX Queue Configuration / Control Registers 2 n (n = 0, 1)	CFDTXQCC2n	H'0000_0000	H'10C0 + n × H'0004	8, 16, 32
TX Queue Status Registers 2 n (n = 0, 1)	CFDTXQSTS2n	H'0000_0001	H'10E0 + n × H'0004	8, 16, 32
TX Queue Pointer Control Registers 2 n (n = 0, 1)	CFDTXQPCTR2n	H'0000_0000	H'1100 + n × H'0004	8, 16, 32
TX Queue Configuration / Control Registers 3 n (n = 0, 1)	CFDTXQCC3n	H'0000_0000	H'1120 + n × H'0004	8, 16, 32
TX Queue Status Registers 3 n (n = 0, 1)	CFDTXQSTS3n	H'0000_0001	H'1140 + n × H'0004	8, 16, 32
TX Queue Pointer Control Registers 3 n (n = 0, 1)	CFDTXQPCTR3n	H'0000_0000	H'1160 + n × H'0004	8, 16, 32
TX Queue Empty Status Register	CFDTXQESTS	H'FFFF_FFFF	H'1180	8, 16, 32
TX Queue Full Interrupt Status Register	CFDTXQFISTS	H'0000_0000	H'1184	8, 16, 32
TX Queue Message Lost Status Register	CFDTXQMSTS	H'0000_0000	H'1188	8, 16, 32
TX Queue Message Overwrite Status Register	CFDTXQOWSTS	H'0000_0000	H'118C	8, 16, 32
TX Queue Interrupt Status Register	CFDTXQISTS	H'0000_0000	H'1190	8, 16, 32
TX Queue One Frame TX Interrupt Status Register	CFDTXQOFTISTS	H'0000_0000	H'1194	8, 16, 32
TX Queue One Frame RX Interrupt Status Register	CFDTXQOFRISTS	H'0000_0000	H'1198	8, 16, 32
TX Queue Full Status Register	CFDTXQFSTS	H'0000_0000	H'119C	8, 16, 32
TX History List Configuration / Control Register n (n = 0, 1)	CFDTHLCCn	H'0000_0000	H'1200 + n × H'0004	8, 16, 32
TX History List Status Register n (n = 0, 1)	CFDTHLSTSn	H'0000_0001	H'1220 + n × H'0004	8, 16, 32

Table 30.4 Register Configurations (3/4)

Register Name	Symbol	Value After Reset	Offset Address N/A (after)	Access Size
TX History List Pointer Control Registers n (n = 0, 1)	CFDTHLPCTRn	H'0000_0000	H'1240 + n × H'0004	8, 16, 32
Global TX Interrupt Status Register 0	CFDGTINTSTS0	H'0000_0000	H'1300	8, 16, 32
Global Test Configuration Register	CFDGTSTCFG	H'0000_0000	H'1308	8, 16, 32
Global Test Control Register	CFDGTSTCTR	H'0000_0000	H'130C	8, 16, 32
Global FD Configuration register	CFDGFDCFG	H'0000_0000	H'1314	8, 16, 32
Global Lock Key Register	CFDGLOCKK	H'0000_0000	H'131C	16, 32
DMA Transfer Control Register	CFDCDTCT	H'0000_0000	H'1330	8, 16, 32
DMA Transfer Status Register	CFDCDTSTS	H'0000_0000	H'1334	8, 16, 32
DMA TX Transfer Control Register	CFDCDTTCT	H'0000_0000	H'1340	8, 16, 32
DMA TX Transfer Status Register	CFDCDTTSTS	H'0000_0000	H'1344	8, 16, 32
Global RX Interrupt Status Register n (n = 0, 1)	CFDGRINTSTSn	H'0000_0000	H'1350 + n × H'0004	8, 16, 32
Global Reset Control Register	CFDGRSTC	H'0000_0000	H'1380	16, 32
Global Flexible CAN mode Configuration Register	CFDGFCMC	H'0000_0000	H'1384	8, 16, 32
Global Flexible Transmission Buffer Assignment Configuration Register	CFDGFTBAC	H'0000_0000	H'138C	8, 16, 32
Channel n Data Bitrate Configuration Register	CFDCnDCFG	H'0000_0000	H'1400 + n × H'0020	8, 16, 32
Channel n CAN-FD Configuration Register	CFDCnFDCFG	H'0000_0000	H'1404 + n × H'0020	8, 16, 32
Channel n CAN-FD Control Register	CFDCnFDCTR	H'0000_0000	H'1408 + n × H'0020	8, 16, 32
Channel n CAN-FD Status Register	CFDCnFDSTS	H'0000_0000	H'140C + n × H'0020	8, 16, 32
Channel n CAN-FD CRC Register	CFDCnFDCRC	H'0000_0000	H'1410 + n × H'0020	8, 16, 32
Channel n Bus load Control Register	CFDCnBLCT	H'0000_0000	H'1418 + n × H'0020	8, 16, 32
Channel n Bus load Status Register	CFDCnBLSTS	H'0000_0000	H'141C + n × H'0020	8, 16, 32
Global Acceptance Filter List ID Registers n (n = 0 to 15)	CFDGAFLIDn	H'0000_0000*1	H'1800 + n × H'0010	8, 16, 32
Global Acceptance Filter List Mask Registers n (n = 0 to 15)	CFDGAFLMn	H'0000_0000*1	H'1804 + n × H'0010	8, 16, 32
Global Acceptance Filter List Pointer 0 Registers n (n = 0 to 15)	CFDGAFLP0n	H'0000_0000*1	H'1808 + n × H'0010	8, 16, 32
Global Acceptance Filter List Pointer 1 Registers n (n = 0 to 15)	CFDGAFLP1n	H'0000_0000*1	H'180C + n × H'0010	8, 16, 32
Channel n TX History List Access Registers 0 (n = 0, 1)	CFDTHLACC0n	H'0000_0000*1	H'8000 + n × H'0008	8, 16, 32
Channel n TX History List Access Registers 1 (n = 0, 1)	CFDTHLACC1n	H'0000_0000*1	H'8004 + n × H'0008	8, 16, 32
RAM Test Page Access Registers n (n =0 to 63)	CFDRPGACCn	H'0000_0000*1	H'8400 + n × H'0004	8, 16, 32
RX Message Buffer Component b	CFDRMBCPb[i]	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
RX Message Buffer ID Registers n (n = 0 to 31)	CFDRMIDn	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
RX Message Buffer Pointer Registers n (n = 0 to 31)	CFDRMPTRn	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
RX Message Buffer CAN-FD Status Register n (n = 0 to 31)	CFDRMFDSTSn	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
RX Message Buffer Data Field p Registers n (p = 0 to 15, n = 0 to 31)	CFDRMDFp_n	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
RX FIFO Access Message Buffer Component b	CFDRFMBCPb[i]	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
RX FIFO Access ID Registers n (n = 0 to 7)	CFDRFIDn	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
RX FIFO Access Pointer Register n (n = 0 to 7)	CFDRFPTRn	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
RX FIFO Access CAN-FD Status Register n (n = 0 to 7)	CFDRFFDSTSn	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32

Table 30.4 Register Configurations (4/4)

Register Name	Symbol	Value After Reset	Offset Address N/A (after)	Access Size
RX FIFO Access Data Field p Registers n (p = 0 to 15, n = 0 to 7)	CFDRFDFp_n	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
Common FIFO Access Message Buffer Component b	CFDCFMBCPb[i]	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
Common FIFO Access ID Registers n (n = 0 to 5)	CFDCFIDn	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
Common FIFO Access Pointer Registers n (n = 0 to 5)	CFDCFPTRn	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
Common FIFO Access CAN-FD Control/Status Register n (n= 0 to 5)	CFDCFFDCSTSn	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
Common FIFO Access Data Field p Registers n (p= 0 to 15, n = 0 to 5)	CFDCFDFp_n	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
TX Message Buffer Component b	CFDTMBCPb[i]	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
TX Message Buffer ID Registers n (n = 0 to 127)	CFDTMIDn	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
TX Message Buffer Pointer Registers n (n = 0 to 127)	CFDTMPTRn	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
TX Message Buffer CAN-FD Control Register n (n = 0 to 127)	CFDTMFDCTRn	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32
TX Message Buffer Data Field p Registers n (p = 0 to 15, n = 0 to 127)	CFDTMDFp_n	H'0000_0000*1	Refer to **Table 30.16**	8, 16, 32

Note 1. The RAM area is initialized after HW reset, refer **Section 30.4.2, CAN Module Configuration after Hardware Reset**.

30.2.1 Channel n Nominal Bit Rate Configuration Register (CFDCnNCFG) (n = 0, 1)

The CFDCnNCFG register is used to configure the transmission/reception nominal bit rate parameters of the channels. Do not write this register in CH_OPERATION or CH_SLEEP mode. Set this register in CH_RESET or CH_HALT mode.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0000 + H'10 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NTSEG2[6:0]							NTSEG1[7:0]								NSJW[6]
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NSJW[5:0]						NBRP[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	NTSEG2[6:0]	All 0	R/W	Nominal Bit Rate Time Segment 2 Control H'00: Setting prohibited H'01: 2 Tq ⋮ H'7E: 127 Tq H'7F: 128 Tq
24 to 17	NTSEG1[7:0]	H'00	R/W	Nominal Bit Rate Time Segment 1 Control H'00: Setting prohibited H'01: 2 Tq H'02: 3 Tq ⋮ H'FE: 255 Tq H'FF: 256 Tq
16 to 10	NSJW[6:0]	All 0	R/W	Nominal Bit Rate Resynchronization Jump Width Control H'00: Setting prohibited H'01: 2 Tq ⋮ H'7E: 127 Tq H'7F: 128 Tq
9 to 0	NBRP[9:0]	All 0	R/W	Nominal Bit Rate Prescaler Nominal bit rate prescaler division ratio setting

NBRP[9:0] bits (Nominal Bit Rate Prescaler)

The NBRP[9:0] bits are used to define the peripheral bus clock periods contained in a time quantum.

NSJW[6:0] bits (Nominal Bit Rate Resynchronization Jump Width Control)

The NSJW[6:0] bits set the synchronization jump width. A value from 1 to 128 time quanta can be set.

NTSEG1[7:0] bits (Nominal Bit Rate Time Segment 1 Control)

The NTSEG1[7:0] bits set the segment TSEG1 to compensate for edges on the CAN bus with a positive phase error. These bits contain the propagation segment. Configure a Tq value only between 2 and 256, inclusive. See **Section 30.4.1.2, CAN Bit Timing** for more details.

NTSEG2[6:0] bits (Nominal Bit Rate Time Segment 2 Control)

The NTSEG2[6:0] bits set the segment TSEG2 to compensate for edges on the CAN bus with a negative phase error. Configure a Tq value only between 2 and 128, inclusive.

30.2.2 Channel n Control Register (CFDCnCTR) (n = 0, 1)

The CFDCnCTR register controls the modes of the related channel. It is used to enable generation of interrupts if errors are detected on the CAN bus connected to this channel. It is also used to configure the channel in test mode.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0004 + H'10 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ROM	CRCT	—	—	—	CTMS[1:0]		CTME	ERRD	BOM[1:0]		—	TDCVFIE	SOCOIE	EOCOIE	TAIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALIE	BLIE	OLIE	BORIE	BOEIE	EPIE	EWIE	BEIE	—	—	—	—	RTBO	CSLPR	CHMDC[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	ROM	0	R/W	Restricted Operation Mode Enable 0: Restricted operation mode disabled 1: Restricted operation mode enabled
30	CRCT	0	R/W	CRC Error Test Enable 0: The first bit of reception ID field is not inverted 1: The first bit of reception ID field is inverted
29 to 27	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.
26, 25	CTMS[1:0]	00	R/W	Communication Test Mode Select 00: Standard test mode 01: Listen-only mode 10: Self-test mode 0 (External loopback mode) 11: Self-test mode 1 (Internal loopback mode)
24	CTME	0	R/W	Communication Test Mode Enable 0: Communication test mode is disabled 1: Communication test mode is enabled
23	ERRD	0	R/W	Error Display Mode Select 0: Error flags are displayed only for the first error information after bits [14:8] in the CFDCnERFL register are all cleared 1: Error flag for all error information are displayed
22, 21	BOM[1:0]	00	R/W	Bus-Off Recovery Mode Select 00: ISO11898-1 compliant 01: Entry to Channel Halt mode automatically at bus-off entry 10: Entry to Channel Halt mode automatically at bus-off end 11: Entry to Channel Halt mode (in bus-off state) by program request
20	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
19	TDCVFIE*1	0	R/W	Transceiver Delay Compensation Violation Interrupt Enable 0: Transceiver delay compensation violation interrupt disabled 1: Transceiver delay compensation violation interrupt enabled

Bit	Bit Name	Initial Value	R/W	Description
18	SOCOIE	0	R/W	Successful Occurrence Counter Overflow Interrupt Enable 0: Successful occurrence counter overflow interrupt disabled 1: Successful occurrence counter overflow interrupt enabled
17	EOCOIE	0	R/W	Error Occurrence Counter Overflow Interrupt Enable 0: Error occurrence counter overflow interrupt disabled 1: Error occurrence counter overflow interrupt enabled
16	TAIE	0	R/W	Transmission Abort Interrupt Enable 0: Transmission abort interrupt disabled 1: Transmission abort interrupt enabled
15	ALIE	0	R/W	Arbitration Lost Interrupt Enable 0: Arbitration lost interrupt disabled 1: Arbitration lost interrupt enabled
14	BLIE	0	R/W	Bus Lock Interrupt Enable 0: Bus lock interrupt disabled 1: Bus lock interrupt enabled
13	OLIE	0	R/W	Overload Interrupt Enable 0: Overload interrupt disabled 1: Overload interrupt enabled
12	BORIE	0	R/W	Bus-Off Recovery Interrupt Enable 0: Bus-off recovery interrupt disabled 1: Bus-off recovery interrupt enabled
11	BOEIE	0	R/W	Bus-Off Entry Interrupt Enable 0: Bus-off entry interrupt disabled 1: Bus-off entry interrupt enabled
10	EPIE	0	R/W	Error Passive Interrupt Enable 0: Error passive interrupt disabled 1: Error passive interrupt enabled
9	EWIE	0	R/W	Error Warning Interrupt Enable 0: Error warning interrupt disabled 1: Error warning interrupt enabled
8	BEIE	0	R/W	Bus Error Interrupt Enable 0: Bus error interrupt disabled 1: Bus error interrupt enabled
7 to 4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
3	RTBO	0	R/W	Forcible Return from Bus-Off When this bit is set to 1, forcible return from the bus-off state is made. This bit is always read as 0.
2	CSLPR	1	R/W	Channel Stop Mode 0: Other than Channel Stop mode 1: Channel Stop mode
1, 0	CHMDC[1:0]	01	R/W	Mode Select 00: Channel Communication mode 01: Channel Reset mode 10: Channel Halt mode 11: Setting prohibited

CHMDC[1:0] bits (Mode Select)

The CHMDC[1:0] bits can be used to configure modes of the CAN channel.

CAN mode transitions are described in more details in **Section 30.3.3, Channel Modes**.

Setting CHMDC[1:0] bits to 11b has no effect. When the CAN-FD module is in GL_HALT mode, these bits can only be set to 10b or 01b. These bits cannot be set in CH_SLEEP mode.

These bits can change automatically when transitioning to Halt mode by the CFDCnCTR.BOM settings.

If CPU write access to CFDCnCTR.CHMDC occurs at the same time when the CAN channel enters Halt mode (at the start of bus-off when CFDCnCTR.BOM[1:0] = 01b, or at the end of bus-off when CFDCnCTR.BOM[1:0] = 10b), then the CPU write access has the highest priority.

The CAN channel changes the value of CFDCnCTR.CHMDC within the Channel Control Registers for the specified cases only if the CFDCnCTR.CHMDC[1:0] value is 00b (Operation mode).

CSLPR bit (Channel Stop Mode)

When the CSLPR bit is 1, a Sleep mode request is generated for the corresponding CAN channel.

When this bit is 0, a request to exit Sleep mode is generated for the related CAN-FD channel.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_SLEEP mode.

RTBO bit (Forcible Return from Bus-Off)

When the protocol controller of the CAN channel enters bus-off state, you can force a recovery from bus-off state by setting the RTBO bit in the Channel Control Register to 1.

The error state changes from bus-off state to integrating with a maximum delay of 1 CAN bit time.

When the RTBO bit is set to 1, the REC and TEC registers are initialized and the Bus-Off Status bit (Channel Bus-off Status, CFDCnSTS.BOSTS) is set to 0.

Registers other than the REC and TEC registers are not initialized by this command. Even if CFDCnCTR.BORIE is set, a bus-off recovery interrupt is not generated by this recovery from the bus-off state.

The RTBO bit cannot be set in CH_SLEEP mode. Setting this bit in any state other than bus-off state has no effect and the bit is cleared immediately. The read value is always 0.

Return from the Bus-Off command should be used only when CFDCnCTR.BOM[1:0] is set to 00b.

Only write to this bit when the related CAN-FD channel is in CH_OPERATION mode. This bit is automatically cleared when set by software.

BEIE bit (Bus Error Interrupt Enable)

When the BEIE and the CFDCnERFL.BEF bits are both 1, an error interrupt request is generated.

This bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

EWIE bit (Error Warning Interrupt Enable)

When the EWIE and the CFDCnERFL.EWF bits are both 1, an error interrupt request is generated.

The EWIE bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

EPIE bit (Error Passive Interrupt Enable)

An error interrupt request is generated when the EPIE bit and the CFDCnERFL.EPF are both 1.

The EPIE bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

BOEIE bit (Bus-Off Entry Interrupt Enable)

When the BOEIE and the CFDCnERFL.BOEF bits are both 1, an error interrupt request is generated.

The BOEIE bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

BORIE bit (Bus-Off Recovery Interrupt Enable)

When the BORIE and the CFDCnERFL.BORF bits are both 1, an error interrupt request is generated.

The BORIE bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

OLIE bit (Overload Interrupt Enable)

When the OLIE and the CFDCnERFL.OVLF bits are both 1, an error interrupt request is generated.

Do not write to this bit in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

BLIE bit (Bus Lock Interrupt Enable)

When the BLIE and the CFDCnERFL.BLF bits are both 1, an error interrupt request is generated.

Do not write to this bit in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

ALIE bit (Arbitration Lost Interrupt Enable)

When the ALIE and the CFDCnERFL.ALF bits are both 1, an error interrupt request is generated.

Do not write to this bit in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

TAIE bit (Transmission Abort Interrupt Enable)

When the TAIE bit is 1 and a transmission is successfully aborted from the transmit buffer, an interrupt request is generated.

Do not write to this bit in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

EOCOIE bit (Error Occurrence Counter Overflow Interrupt Enable)

When the EOCOIE bit is 1 and the CFDCnFDSTS.EOCO bit belonging to the corresponding CAN channel is 1, an error interrupt request is generated.

The EOCOIE bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

SOCOIE bit (Successful Occurrence Counter Overflow Interrupt Enable)

When the SOCOIE bit is 1 and the CFDCnFDSTS.SOCO bit belonging to the corresponding CAN channel is 1, an error interrupt request is generated.

The SOCOIE bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

TDCVFIE bit (Transceiver Delay Compensation Violation Interrupt Enable)

When the TDCVFIE bit is 1 and the CFDCnFDSTS.TDCVF bit belonging to the corresponding CAN channel is 1, an error interrupt request is generated.

The TDCVFIE bit cannot be set in CH_SLEEP mode.

Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

BOM[1:0] bits (Bus-Off Recovery Mode Select)

The BOM[1:0] bits control the timing of the recovery from Bus-Off mode of the CAN-FD Channel.

Do not write to these bits in CH_SLEEP mode. Only write to these bits when the related CAN-FD channel is in CH_RESET mode.

ERRD bit (Error Display Mode Select)

The ERRD bit controls the display mode of the error flag bits [14:8] in the Channel Error Flag Register (CFDCnERFL).

If the ERRD bit is 0 and more than one errors occur at the same time, the error flag bits are set for all the errors that occurred at the same time. No further errors are flagged until the error flag bits [14:8] are cleared.

Do not write to the ERRD bit in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

CTME bit (Communication Test Mode Enable)

The CTME bit enables the channel test modes.

Do not write to this bit in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_HALT mode.

CTMS[1:0] bits (Communication Test Mode Select)

The CTMS[1:0] bits are used to select the required test mode.

Do not write to these bits in CH_SLEEP or CH_RESET mode. Only write to these bits when the related CAN-FD channel is in CH_HALT mode.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

CRCT bit (CRC Error Test Enable)

The CRCT bit checks the internal CRC generator logic of the protocol controller.

This bit inverts the first bit (ID bit) of the CAN message data stream that is being received, so that the internal generated CRC result does not match the received CRC value of the frame. Refer to the bit stuffing rule when using this feature for the possibility of receiving a stuff error (due to the inversion) rather than a CRC error.

The internal generated CRC value is always observed in the following registers:

- CFDCnFDCRC.CRCREG (CAN-FD frames)

Some limitations exist when using this bit:

- It is not possible to use this feature with CAN nodes connected to the device externally, only with nodes connected to the internal CAN bus communication can be used
- One CAN node can send a reference message and the receiver node can invert one bit of the incoming bit stream

Note: The transmitter and receiver modes share the same CRC generator, therefore it is not necessary to consider the modes separately when testing this limitation.

The CRC Error Test mode is enabled if the CRCT (new control signal that inverts the first bit of the bit stream) and CTME bits are both 1.

Do not write to the CRCT bit in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_HALT mode.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

ROM bit (Restricted Operation Mode Enable)

When the ROM and CTME bits are both 1, the restricted operation mode is enabled. This mode should only be used in basic test mode (CFDCnCTR.CTMS[1:0] = 00b).

The ROM bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_HALT mode.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

30.2.3 Channel n Status Register (CFDCnSTS) (n = 0, 1)

The CFDCnSTS register shows the mode, error, and transmission/reception status of the related channel together with its reception and transmission error count values.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0008 + H'10 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEC[7:0]								REC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ESIF	COMSTS	RECSTS	TRMSTS	BOSTS	EPSTS	CSLPSTS	CHLTSTS	CRSTSTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TEC[7:0]	H'00	R	Transmission Error Count The transmit error counter (TEC) can be read.
23 to 16	REC[7:0]	H'00	R	Reception Error Count The receive error counter (REC) can be read.
15 to 9	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
8	ESIF	0	R/W	Error State Indication Flag 0: No CAN-FD message whose ESI bit is recessive has been received 1: At least one CAN-FD message whose ESI bit is recessive has been received
7	COMSTS	0	R	Communication Status Flag 0: Communication is not ready 1: Communication is ready
6	RECSTS	0	R	Receive Status Flag 0: Bus idle, in transmission or bus off state 1: In reception
5	TRMSTS	0	R	Transmit Status Flag 0: Bus idle or in reception 1: In transmission or bus off state
4	BOSTS	0	R	Bus-Off Status Flag 0: Not in bus-off state 1: In bus-off state
3	EPSTS	0	R	Error Passive Status Flag 0: Not in error passive state 1: In error passive state
2	CSLPSTS	1	R	Channel Stop Status Flag 0: Not in Channel Sleep mode 1: In Channel Sleep mode
1	CHLTSTS	0	R	Channel Halt Status Flag 0: Not in Channel Halt mode 1: In Channel Halt mode

Bit	Bit Name	Initial Value	R/W	Description
0	CRSTSTS	1	R	Channel Reset Status Flag 0: Not in Channel Reset mode 1: In Channel Reset mode

CRSTSTS bit (Channel Reset Status Flag)

The CRSTSTS bit indicates whether the related CAN channel is in Reset mode.

This bit is set automatically when the related CAN channel enters Channel Reset mode. When the mode is changed from Reset mode to Sleep mode, the CRSTSTS bit remains 1.

This bit is cleared automatically when the related CAN channel exits the Channel Reset mode, except when changing to Sleep mode.

CHLTSTS bit (Channel Halt Status Flag)

The CHLTSTS bit indicates whether the related CAN channel is in Halt mode.

This bit is set automatically when the related CAN module enters Halt mode, and is cleared automatically when the related CAN module exits Halt mode.

CSLPSTS bit (Channel Stop Status Flag)

The CSLPSTS bit indicates whether the related CAN channel is in Sleep mode.

This bit is set automatically when the related CAN-FD channel enters Sleep mode, and is cleared automatically when the related CAN-FD channel exits Sleep mode.

EPSTS bit (Error Passive Status Flag)

The EPSTS bit indicates whether the related CAN-FD channel has entered the error passive state.

This bit is set automatically when the value of the CAN Transmission or Reception Counter Register exceeds the value of H’7F.

This bit is cleared automatically when the related CAN-FD channel exits the error passive state or enters Reset mode.

BOSTS bit (Bus-Off Status Flag)

The BOSTS bit indicates whether the related CAN-FD channel has entered the error bus-off state.

This bit is set automatically when the value of the related CAN Transmission Error Count Register exceeds H’FF and the related CAN-FD channel is in the bus-off state (CAN Transmission Error Count Register > H’FF).

This bit is cleared automatically when the related CAN-FD channel exits bus-off state.

TRMSTS bit (Transmit Status Flag)

The TRMSTS bit indicates whether the related CAN-FD channel is transmitting a message.

This bit is set automatically when the related CAN-FD channel is operating as a transmitter node or is in the bus-off state.

This bit is cleared automatically when the related CAN-FD channel is in the bus-idle state or starts operating as a receiver node.

RECSTS bit (Receive Status Flag)

The RECSTS bit indicates whether the related CAN-FD channel is receiving a message.

This bit is set automatically when the related CAN-FD channel is operating as a receiver node.

This bit is cleared automatically when the related CAN-FD channel is in the bus-idle state or starts operating as a transmitter node.

COMSTS bit (Communication Status Flag)

The COMSTS bit indicates whether the related CAN-FD channel is ready for communication.

This bit is set automatically when the related CAN-FD channel is ready to perform communication following the detection of 11 consecutive recessive bits after exiting the Reset or Halt mode.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

Note: This bit is 1 during bus-off state.

ESIF bit (Error State Indication Flag)

The ESIF bit is set when the ESI bit is sampled recessive for a reception CAN message without any error. When in Loopback or Mirror mode, the self-transmitted messages are considered reception messages.

If a set from the CAN-FD channel occurs simultaneously with a clear by a write access, then the bit is set.

This bit is cleared by writing 0 to it. This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

Only write to this bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

REC[7:0] bits (Reception Error Count)

The REC[7:0] bits increment or decrement the counter value according to error status of the CAN-FD channel during reception, and display the value of the REC error counter.

The value in bus-off state is indeterminate.

These bits are cleared automatically when the CAN-FD module enters GL_RESET or the CAN-FD channel is in CH_RESET mode.

TEC[7:0] bits (Transmission Error Count)

The TEC[7:0] bits increment or decrement the counter value according to error status of the CAN-FD channel during transmission, and display the value of the TEC error counter.

Only write to these bits when in test mode and CAN-FD channel is in CH_HALT mode.

These bits are cleared automatically when CAN-FD module is in GL_RESET or CAN-FD channel is in CH_RESET mode.

30.2.4 Channel n Error Flag Register (CFDCnERFL) (n = 0, 1)

The CFDCnERFL register shows the status of various error conditions detectable regardless of the setting of the related CAN Channel Error Interrupt Enable Register. It also shows the status of the various bus errors detectable by the CAN channel. Refer to the CAN specification (ISO 11898-1) for a description of error occurrence conditions.

Only write to this register when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'000C + H'10 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CRCREG[14:0]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ADERR	B0ERR	B1ERR	CERR	AERR	FERR	SERR	ALF	BLF	OVLF	BORF	BOEF	EPF	EWF	BEF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
30 to 16	CRCREG [14:0]	All 0	R	CRC Calculation Data (CRC length: 15 bits) These bits show the CRC value calculated based on the transmit message or receive message.
15	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
14	ADERR	0	R/W*1	Acknowledge Delimiter Error Flag 0: Channel acknowledge delimiter error not detected 1: Channel acknowledge delimiter error detected
13	B0ERR	0	R/W*1	Dominant Bit Error Flag 0: Channel dominant bit error not detected 1: Channel dominant bit error detected
12	B1ERR	0	R/W*1	Recessive Bit Error Flag 0: Channel recessive bit error not detected 1: Channel recessive bit error detected
11	CERR	0	R/W*1	CRC Error Flag 0: Channel CRC error not detected 1: Channel CRC error detected
10	AERR	0	R/W*1	Acknowledge Error Flag 0: Channel acknowledge error not detected 1: Channel acknowledge error detected
9	FERR	0	R/W*1	Form Error Flag 0: Channel form error not detected 1: Channel form error detected
8	SERR	0	R/W*1	Stuff Error Flag 0: Channel stuff error not detected 1: Channel stuff error detected

Bit	Bit Name	Initial Value	R/W	Description
7	ALF	0	R/W*1	Arbitration Lost Flag 0: Channel arbitration-lost not detected 1: Channel arbitration-lost detected
6	BLF	0	R/W*1	Bus Lock Flag 0: Channel bus lock not detected 1: Channel bus lock detected
5	OVLF	0	R/W*1	Overload Flag 0: Channel overload not detected 1: Channel overload detected
4	BORF	0	R/W*1	Bus-Off Recovery Flag 0: Channel bus-off recovery not detected 1: Channel bus-off recovery detected
3	BOEF	0	R/W*1	Bus-Off Entry Flag 0: Channel bus-off entry not detected 1: Channel bus-off entry detected
2	EPF	0	R/W*1	Error Passive Flag 0: Channel error passive not detected 1: Channel error passive detected
1	EWF	0	R/W*1	Error Warning Flag 0: Channel error warning not detected 1: Channel error warning detected
0	BEF	0	R/W*1	Bus Error Flag 0: Channel bus error not detected 1: Channel bus error detected

Note 1. To clear each flag of this register, software must write 0 to the corresponding bit. These flags cannot be set to 1 by software.

BEF bit (Bus Error Flag)

The BEF bit indicates a detection of a CAN channel bus error state, flagged by bits [14:8] in this register.

This bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

This bit is set automatically when a bus error is detected, and is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

If a set from the CAN channel occurs simultaneously with a clear by a write access, then the bit is set.

EWF bit (Error Warning Flag)

The EWF bit indicates whether an error warning condition has been detected for the CAN channel.

This bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

This bit is set automatically when either TEC or REC exceeds H’5F.

The setting of this bit only occurs when the TEC or REC initially exceeds H’5F. Therefore, if the TEC or REC remains > H’5F and the EWF bit is cleared by software, it is not set again until both the TEC and REC go below H’60 and either TEC or REC crosses over again from a value H’5F to a value > H’5F.

If a set condition occurs simultaneously with a clear condition, then the bit is set. It is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

EPF bit (Error Passive Flag)

The EPF bit indicates a detection of a CAN channel error passive state.

This bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

This bit is set automatically when the CAN error state becomes error passive state.

The setting of this bit only occurs when the TEC or REC initially exceeds H’7F. Therefore, if the TEC or REC remains > H’7F and the bit is cleared by software, it is not set again until both the TEC and REC go below H’80 and either TEC or REC crosses over again from a value ≤ H’7F to a value > H’7F.

If a set condition occurs simultaneously with a clear condition, then the bit is set. It is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

BOEF bit (Bus-Off Entry Flag)

The BOEF bit indicates a detection of a CAN channel bus-off entry state.

This bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

This bit is set automatically when the CAN error state enters the bus-off state.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If a set condition occurs simultaneously with a clear condition, then the bit is set.

BORF bit (Bus-Off Recovery Flag)

The BORF bit indicates a detection of a CAN channel bus-off recovery state.

This bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

This bit is set automatically if CAN channel recovers from bus-off state in the following conditions:

- When CFDCnCTR.BOM[1:0] is 00b and normal recovery (11 consecutive recessive bits × 128 times detected) occurs
- When CFDCnCTR.BOM[1:0] is 10b and normal recovery (11 consecutive recessive bits × 128 times detected) occurs
- When CFDCnCTR.BOM[1:0] is 11b and normal recovery (11 consecutive recessive bits × 128 times detected) occurs.

The bit is not set if CAN channel recovers from bus-off state in the following conditions:

- When CAN Reset mode is requested
- When CFDCnCTR.RTBO is set to 1 (the CAN channel returns to error active)
- When CFDCnCTR.BOM[1:0] is 01b
- When CFDCnCTR.BOM[1:0] is 11b and a halt request is asserted before the CAN channel reaches the end of the bus- off state.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If a set condition occurs simultaneously with a clear condition, the flag is set.

OVLF bit (Overload Flag)

The OVLF flag indicates a detection of a CAN channel overload state.

The OVLF bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

This bit is set automatically when an overload condition is detected. If a set condition occurs simultaneously with a clear condition, then the bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

BLF bit (Bus Lock Flag)

The BLF bit indicates a detection of a CAN channel bus lock condition.

This bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

This bit is set automatically when 32 consecutive dominant bits are detected on the CAN bus while the CAN channel is in Operation mode.

If a set condition occurs simultaneously with a clear condition, then the bit is set. It is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

It is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

ALF bit (Arbitration Lost Flag)

The ALF bit indicates a detection of a CAN channel bus arbitration-lost condition.

This bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

The bit is set automatically when an arbitration-lost condition is detected on the CAN bus while the CAN channel is in Operation mode.

If a set condition occurs simultaneously with a clear condition, then the bit is set. It is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

SERR bit (Stuff Error Flag)

The SERR bit indicates a detection of a CAN stuff error.

This bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

To clear this bit, use the following sequence:

1. Clear the corresponding flag bit.
2. Read if the flag bit is cleared.
3. If yes, continue, else go back to step 1.

This bit is set automatically when a stuff error is detected. If CFDCnCTR.ERRD bit is 1 and if the set and clear conditions occur at the same time for this bit, then this bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If CFDCnCTR.ERRD bit is 0 and the set and clear conditions occur at the same time for this bit, then it is cleared if a bit at the error flag bits [14:8] in the CFDCnERFL register is already set. Otherwise, this bit is set if the error flag bits [14:8] in the CFDCnERFL register is 0000000b.

FERR bit (Form Error Flag)

The FERR bit indicates a detection of a CAN form error.

This bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

To clear this bit, use the following sequence:

1. Clear the corresponding flag bit.
2. Read if the flag bit is cleared.
3. If yes, continue, else go back to step 1.

This bit is set automatically when a form error is detected. If CFDCnCTR.ERRD bit is 1 and if the set and clear conditions occur at the same time for this bit, then this bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If CFDCnCTR.ERRD bit is 0 and the set and clear conditions occur at the same time for this bit, then it is cleared if a bit at the error flag bits [14:8] in the CFDCnERFL register is already set. Otherwise, this bit is set if the error flag bits [14:8] in the CFDCnERFL register is 0000000b.

AERR bit (Acknowledge Error Flag)

The AERR bit indicates a detection of a CAN acknowledge error.

This bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

To clear this bit, use the following sequence:

1. Clear the corresponding flag bit.
2. Read if the flag bit is cleared.
3. If yes, continue, else go back to step 1.

This bit is set automatically when an acknowledge error is detected. If CFDCnCTR.ERRD bit is 1 and if the set and clear conditions occur at the same time for this bit, then this bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If CFDCnCTR.ERRD bit is 0 and the set and clear conditions occur at the same time for this bit, then it is cleared if a bit at the error flag bits [14:8] in the CFDCnERFL register is already set. Otherwise, this bit is set if the error flag bits [14:8] in the CFDCnERFL register is 0000000b.

CERR bit (CRC Error Flag)

The CERR bit indicates a detection of a CAN CRC error.

This bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

To clear this bit, use the following sequence:

1. Clear the corresponding flag bit.
2. Read if the flag bit is cleared.
3. If yes, continue, else go back to step 1.

This bit is set automatically when a CRC error is detected. If CFDCnCTR.ERRD bit is 1 and if the set and clear conditions occur at the same time for this bit, then this bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If CFDCnCTR.ERRD bit is 0 and the set and clear conditions occur at the same time for this bit, then it is cleared if a bit at the error flag bits [14:8] in the CFDCnERFL register is already set. Otherwise, this bit is set if the error flag bits [14:8] in the CFDCnERFL register is 0000000b.

B1ERR bit (Recessive Bit Error Flag)

The B1ERR bit indicates a detection of a recessive bit error.

This bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

To clear this bit, use the following sequence:

1. Clear the corresponding flag bit.
2. Read if the flag bit is cleared.
3. If yes, continue, else go back to step 1.

This bit is set automatically when a recessive bit error (expected recessive bit, sampled as dominant bit) is detected. If CFDCnCTR.ERRD bit is 1 and if the set and clear conditions occur at the same time for this bit, then this bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If CFDCnCTR.ERRD bit is 0 and the set and clear conditions occur at the same time for this bit, then it is cleared if a bit at the error flag bits [14:8] in the CFDCnERFL register is already set. Otherwise, this bit is set if the error flag bits [14:8] in the CFDCnERFL register is 0000001b.

B0ERR bit (Dominant Bit Error Flag)

The B0ERR bit indicates a detection of a dominant bit error.

This bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

To clear this bit, use the following sequence:

1. Clear the corresponding flag bit.
2. Read if the flag bit is cleared.
3. If yes, continue, else go back to step 1.

This bit is set automatically when a dominant bit error (expected dominant bit, sampled as recessive bit) is detected. If CFDCnCTR.ERRD bit is 1 and if the set and clear conditions occur at the same time for this bit, then this bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If CFDCnCTR.ERRD bit is 0 and the set and clear conditions occur at the same time for this bit, then it is cleared if a bit at the error flag bits [14:8] in the CFDCnERFL register is already set. Otherwise, this bit is set if the error flag bits [14:8] in the CFDCnERFL register is 0000000b.

ADERR bit (Acknowledge Delimiter Error Flag)

The ADERR bit indicates a detection of an acknowledge delimiter bit error.

This bit is cleared by writing 0 to it, and can only be set by CAN-FD module logic. Writing 1 has no effect.

To clear this bit, use the following sequence:

1. Clear the corresponding flag bit.
2. Read if the flag bit is cleared.
3. If yes, continue, else go back to step 1.

This bit is set automatically when a form error is detected during the acknowledge delimiter state of frame transmission. If CFDCnCTR.ERRD bit is 1 and if the set and clear conditions occur at the same time for this bit, then this bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If CFDCnCTR.ERRD bit is 0 and the set and clear conditions occur at the same time for this bit, then it is cleared if a bit at the error flag bits [14:8] in the CFDCnERFL register is already set. Otherwise, this bit is set if the error flag bits [14:8] in the CFDCnERFL register is 0000000b.

CRCREG[14:0] bits (CRC Calculation Data (CRC length: 15 bits))

The CRCREG[14:0] bits read the calculated CRC value when CFDCnCTR.CTME bit is 1 for the channel.

If CFDCnCTR.CTME bit is 0, then these bits are always read as 0.

These bits show the CAN2.0 CRC value calculated by the CAN-FD channel logic when the CTME bit is enabled.

The CFDCnERFL.CRCREG value is updated in the first bit of the CRC field of the CAN frame (reception and transmission).

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

30.2.5 Channel n Data Bit Rate Configuration Register (CFDCnDCFG) (n = 0, 1)

The CFDCnDCFG register configures the transmission/reception data bit rate parameters of the channels.

Do not write to this register in CH_OPERATION or CH_SLEEP mode. Only write to this register in CH_RESET or CH_HALT mode.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1400 + H'20 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	DSJW[3:0]				—	—	—	—	DTSEG2[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	DTSEG1[4:0]					DBRP[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
27 to 24	DSJW[3:0]	H'0	R/W	Data Bit Rate Resynchronization Jump Width Control H'0: 1 Tq H'1: 2 Tq ⋮ H'E: 15 Tq H'F: 16 Tq
23 to 20	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
19 to 16	DTSEG2[3:0]	H'0	R/W	Data Bit Rate Time Segment 2 Control H'0: Setting prohibited H'1: 2 Tq H'2: 3 Tq H'3: 4 Tq ⋮ H'E: 15 Tq H'F: 16 Tq
15 to 13	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.
12 to 8	DTSEG1[4:0]	All 0	R/W	Data Bit Rate Time Segment 1 Control H'00: Setting prohibited H'01: 2 Tq H'02: 3 Tq H'03: 4 Tq ⋮ H'1E: 31 Tq H'1F: 32 Tq
7 to 0	DBRP[7:0]	H'00	R/W	Data Bit Rate Prescaler Division Ratio Setting When the set value = P (0 to 255), the data bit rate prescaler divides fCAN by (P + 1).

DBRP[7:0] bits (Data Bit Rate Prescaler Division Ratio Setting)

The DBRP[7:0] bits define the peripheral bus clock periods contained in a time quantum.

DTSEG1[4:0] bits (Data Bit Rate Time Segment 1 Control)

The DTSEG1[4:0] bits set the segment TSEG1 to compensate for edges on the CAN bus with a positive phase error. A value from 2 to 32 time quanta can be set.

The DTSEG1[4:0] bits are also used to set the propagation segment.

Do not write any other value to these bits. See **Section 30.4.1.2, CAN Bit Timing** for more details.

DTSEG2[3:0] bits (Data Bit Rate Time Segment 2 Control)

The DTSEG2[3:0] bits set the segment TSEG2 to compensate for edges on the CAN bus with a negative phase error. A value from 2 to 16 time quanta can be set.

Do not write any other value to these bits.

DSJW[3:0] bits (Data Bit Rate Resynchronization Jump Width Control)

The DSJW[3:0] bits set the synchronization jump width. A value from 1 to 16 time quanta can be set.

30.2.6 Channel n CAN-FD Configuration Register (CFDCnFDCFG) (n = 0, 1)

The CFDCnFDCFG register configures which communication direction (transmitter/receiver) errors are counted.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1404 + H'20 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	REFE	—	—	GWBRS	GWFDF	GWEN	TDCO[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ESIC	TDCE	TDCOC	—	—	—	—	—	EOCCFG[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
29	REFE	0	R/W	RX Edge Filter Enable 0: RX edge filter disabled 1: RX edge filter enabled
28	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
27	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
26	GWBRS	0	R/W	Gateway BRS Configuration Bit 0: Gateway frame is transmitted with BRS = 0 1: Gateway frame is transmitted with BRS = 1
25	GWFDF	0	R/W	Gateway FDF Configuration Bit 0: Gateway frame is transmitted as Classical CAN frame 1: Gateway frame is transmitted as CAN-FD frame
24	GWEN	0	R/W	CAN2.0, CAN-FD Multi-Gateway Enable 0: Multi gateway disabled 1: Multi gateway enabled
23 to 16	TDCO[7:0]	H'00	R/W	Transceiver Delay Compensation Offset These bits are set to the transmitter delay compensation offset value.
15, 14	—	00	R/W	Reserved These bits are read as 0. The write value should be always 0.
13 to 11	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.
10	ESIC	0	R/W	Error State Indication Configuration 0: The ESI bit in the frame represents the error state of the node itself 1: The ESI bit in the frame represents the error state of the message buffer if the node itself is not in error passive. If the node is in error passive, then the ESI bit is driven by the node itself.
9	TDCE	0	R/W	Transceiver Delay Compensation Enable 0: Transceiver delay compensation disabled 1: Transceiver delay compensation enabled

Bit	Bit Name	Initial Value	R/W	Description
8	TDCOC	0	R/W	Transmitter Delay Compensation Offset Configuration 0: Measurement and offset 1: Offset-only
7 to 3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
2 to 0	EOCCFG[2:0]	000	R/W	Error Occurrence Counter Configuration 000: All transmit messages and receive messages 001: All transmit messages 010: All receive messages 011: Setting prohibited 100: Only data phase of transmitted or received CAN-FD message 101: Only data phase of transmitted CAN-FD message 110: Only data phase of received CAN-FD message 111: Setting prohibited

EOCCFG[2:0] bits (Error Occurrence Counter Configuration)

The EOCCFG[2:0] bits select which type of CAN frame configuration and direction, including protocol errors are counted.

Do not write to these bits in CH_OPERATION or CH_SLEEP mode.

Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

TDCOC bit (Transmitter Delay Compensation Offset Configuration)

The TDCOC bit selects which offset is used when defining the position of the secondary sample point (SSP) for the CAN- FD channel. If the bit is set to 0, the position of the SSP is the measured transceiver delay plus the fixed offset. If the bit is 1, the position of the SSP is defined only by the offset.

Do not write to this bit in CH_OPERATION or CH_SLEEP mode.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

TDCE bit (Transceiver Delay Compensation Enable)

The TDCE bit enables the transceiver delay compensation for the CAN-FD channel.

Do not write to this bit in CH_OPERATION or CH_SLEEP mode.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

ESIC bit (Error State Indication Configuration)

Bus controllers that are used as CAN-to-CAN gateway support that in every forwarded CAN-FD message. The ESI flag does not change to reflect the status of the gateway, bridge, or router but instead the flag is sent as it was in the original message.

The ESIC bit controls the transmission of either the ESI flag information or the message of ESI flag information (CFDCFFDCSTSn.CFESI or CFDTMFDCTRn.TMESI).

Do not write to this bit in CH_OPERATION or CH_SLEEP mode.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

TDCO[7:0] bits (Transceiver Delay Compensation Offset)

The TDCO[7:0] bits set the secondary sample point offset. How this value is used, depends on the CFDCnFDCFG.TDCOC setting.

If CFDCnFDCFG.TDCOC = 0, the transceiver delay compensation result is equal to the Trv_Delay (measured delay) + the value in CFDCnFDCFG.TDCO, rounded down to the nearest integer number of time quanta. Otherwise, the result is equal to the value in CFDCnFDCFG.TDCO. See **Section 30.4.1.5, Transmitter Delay Compensation** for details on how CFDCnFDCFG.TDCO is used.

The actual offset value is interpreted as TDCO + 1. For example, if 4 is set in TDCO, the offset is 5 clock cycles. Clock cycle is 1 cycle of CAN channel DLL clock.

Do not write to the TDCO[7:0] bits in CH_OPERATION or CH_SLEEP mode.

Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GWEN bit (CAN2.0, CAN-FD Multi-Gateway Enable)

When the GWEN bit is enabled, a multi gateway is enabled. Message received on one node can be routed to another node using the COM FIFO when they are configured as gateway FIFO (CFDCFCCn.CFM[1:0] = 10b). Furthermore, when TX Queue is set as Gateway mode, the message received on one node can be stored in TX Queue and can be sent to another node.

The FDF and BRS bits of the routed message can be changed by the configuration value of the CFDCnFDCFG.GWFDF and CFDCnFDCFG.GWBRS bits. By this, the transmitted value of these bits can be replaced.

[Example]

CFDCnFDCFG.GWEN = 1 on channel y
CFDCnFDCFG.GWFDF = 1

If a Classical CAN frame is received on channel x and routed to a gateway FIFO or TX Queue of channel y, then this CAN frame is sent on channel y as a CAN-FD frame because of the CFDCnFDCFG.GWFDF bit.

Table 30.5 shows how the message information is changed depending on the received and configured data.

Table 30.5 Modified Message Information by Received and Configured Data

Routed CAN Frame	Routed Received DLC	CAN BRS Bit	Configured CFDCnFDCFG.GWFDF Bit	Gateway Message DLC	Gateway Message BRS Bit	Gateway Message Frame Type
CAN2.0	≤ 8	N/A	1	≤ 8	Based on configuration of CFDCnFDCFG.GWBRS	CAN-FD
CAN2.0	> 8	N/A	1	= 8	Based on configuration of CFDCnFDCFG.GWBRS	CAN-FD
CAN-FD	≤ 8	None	1	≤ 8	Based on configuration of CFDCnFDCFG.GWBRS	CAN-FD
CAN-FD	> 8	None	1	> 8	Based on configuration of CFDCnFDCFG.GWBRS	CAN-FD
CAN2.0	≤ 8	N/A	0	≤ 8	N/A	CAN2.0
CAN2.0	> 8	N/A	0	> 8	N/A	CAN2.0
CAN-FD	≤ 8	None	0	≤ 8	N/A	CAN2.0
CAN-FD	> 8	None	0	= 8	N/A	CAN2.0

Note: This gateway is limited to an 8-byte data payload for different frame type. If routing and target frame type is the same, the data length code (DLC) value remains the same. If the source frame is a CAN-FD with more than 8 data bytes. Only the first 8 bytes of data perform gateway transmission.

Note: Transmission buffers other than the gateway FIFO are not affected by this feature.

Do not route remote frames through the gateway when CFDCnFDCFG.GWEN is set.

Do not write to this bit in CH_OPERATION or CH_SLEEP mode.

Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

GWFDF bit (Gateway FDF Configuration Bit)

When the GWEN bit is set to 1, the FDF bit of the transmitting gateway frame is replaced by the value of the GWFDF bit.

Do not write to this bit in CH_OPERATION or CH_SLEEP mode.

Only write to these bits when the CAN-FD module is in CH_RESET mode.

GWBRS bit (Gateway BRS Configuration Bit)

When the GWEN bit is set to 1, the BRS bit of the transmitting gateway frame is replaced by the value of CFDCnFDCFG.GWBRS.

Do not write to this bit in CH_OPERATION or CH_SLEEP mode.

Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

REFE bit (RX Edge Filter Enable)

The REFE bit enables the RX edge filter during the IDLE detection (bus integration). When the bit is enabled, two consecutive dominant time quanta are required to detect a synchronization edge.

The REFE bit cannot be written in CH_OPERATION, CH_HALT and CH_SLEEP mode.

30.2.7 Channel n CAN-FD Control Register (CFDCnFDCTR) (n = 0, 1)

The CFDCnFDCTR register (n = 0, 1) controls the error and successful occurrence counters.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1408 + H'20 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOCCLR	EOCCLR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	H'00	W	Reserved These bits are read as 0. The write value should be always 0.
23 to 2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
1	SOCCLR	0	R/W	Successful Occurrence Counter Clear 0: No successful occurrence counter clear 1: Clear successful occurrence counter
0	EOCCLR	0	R/W	Error Occurrence Counter Clear 0: No error occurrence counter clear 1: Clear error occurrence counter

EOCCLR bit (Error Occurrence Counter Clear)

The EOCCLR bit is used to clear the error occurrence counter.

Do not write to this bit in CH_SLEEP or CH_RESET mode. The read value is always 0.

This bit is cleared automatically by the CAN-FD logic and when the related CAN-FD channel is in CH_RESET mode.

SOCCLR bit (Successful Occurrence Counter Clear)

The SOCCLR bit is used to clear the successful occurrence counter.

Do not write to this bit in CH_SLEEP or CH_RESET mode. The read value is always 0.

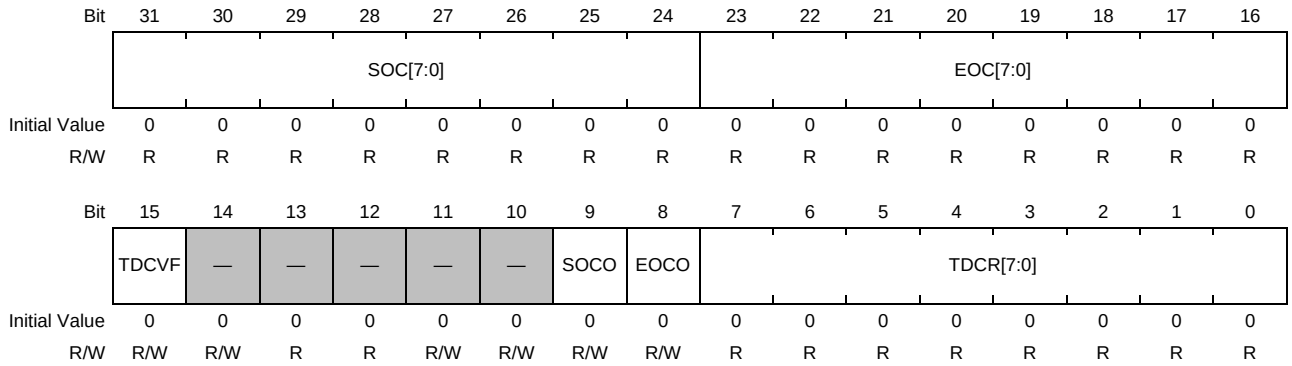
This bit is cleared automatically by the CAN-FD logic and when the related CAN-FD channel is in CH_RESET mode.

30.2.8 Channel n CAN-FD Status Register (CFDCnFDSTS) (n = 0, 1)

The CFDCnFDSTS register (n = 0, 1) indicates the transceiver compensation delay result and its related FIFO message lost status.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'140C + H'20 × n



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	SOC[7:0]	H'00	R	Successful Occurrence Counter These bits show the successful occurrence counter value.
23 to 16	EOC[7:0]	H'00	R	Error Occurrence Counter These bits show the error occurrence counter value.
15	TDCVF	0	R/W	Transceiver Delay Compensation Violation Flag 0: Transceiver delay compensation violation has not occurred 1: Transceiver delay compensation violation has occurred
14	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
13, 12	—	00	R	Reserved These bits are read as 0.
11, 10	—	00	R/W	Reserved These bits are read as 0. The write value should be always 0.
9	SOCO	0	R/W	Successful Occurrence Counter Overflow Flag 0: Successful occurrence counter has not overflowed 1: Successful occurrence counter has overflowed
8	EOCO	0	R/W	Error Occurrence Counter Overflow Flag 0: Error occurrence counter has not overflowed 1: Error occurrence counter has overflowed
7 to 0	TDCR[7:0]	H'00	R	Transceiver Delay Compensation Result

TDCR[7:0] bits (Transceiver Delay Compensation Result)

The TDCR[7:0] bits are set when the transceiver delay has been measured.

The measured delay is a multiple of the CAN channel DLL clock. The result depends on the CFDCnFDCFG.TDCOC configuration and the offset value in CFDCnFDCFG.TDCO. See **Section 30.4.1.5, Transmitter Delay Compensation** for details on how this value is derived.

The TDCR[7:0] bits are updated at the falling edge between the FDF bit and the RES bit when CFDCnFDCFG.TDCOC = 0 and the transceiver delay compensation is enabled (CFDCnFDCFG.TDCE = 1).

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

EOCO bit (Error Occurrence Counter Overflow Flag)

The EOCO bit indicates whether the related CAN channel error occurrence counter has overflowed. This bit is cleared by writing 0 to it. Writing 1 has no effect.

This bit is set automatically when CFDCnFDSTS.EOC is H'FF and a CAN bus error is detected based on the configuration defined in CFDCnFDCFG.EOCCFG.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

Only write to this bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Do not use the bit clear instruction to clear this bit.

SOCO bit (Successful Occurrence Counter Overflow Flag)

The SOCO bit indicates whether the related CAN channel successful occurrence counter has overflowed. This bit is cleared by writing 0 to it. Writing 1 has no effect.

This bit is set automatically when CFDCnFDSTS.SOC is H'FF and a successful message reception or successful message transmission occurs.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

Only write to this bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Do not use the bit clear instruction to clear this bit.

TDCVF bit (Transceiver Delay Compensation Violation Flag)

The CAN-FD module captures internally the transmitted data bit-by-bit. This data is then compared against the received CAN bus level which is delayed by the transceiver loop delay.

The transceiver delay has some variations depending on the physical parameters such as temperature. The result bit CFDCnFDSTS.TDCR is updated by each message. However, temporary maximum delay violation could be missed. Therefore, the TDCVF bit captures this violation.

This bit is cleared by writing 0 to it. Writing 1 has no effect.

This bit is set automatically when the transceiver delay compensation is greater than the maximum delay compensation (6 data bit times - 2 clk_dlc) and the internal bit is overrun.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

Only write to this bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Do not use the bit clear instruction to clear this bit.

EOC[7:0] bits (Error Occurrence Counter)

The EOC[7:0] bits are used together with the SOC[7:0] bits to support an option for host-controlled fall-back to payload bit rate identical to arbitration bit rate when messages utilizing the reduced payload bit length have significant higher error rates compared to other messages.

This higher error rate can be detected depending on the configuration of the CFDCnFDCFG.EOCCFG bits.

The EOC[7:0] bits are set only by CAN-FD module logic. These bits are cleared by writing 1 to CFDCnFDCTR.EOCCLR. Writing any other value has no effect.

These bits are updated when an error occurs, according to the configuration of the CFDCnFDCFG.EOCCFG bits. When the counter reaches the value of H'FF, the update stops.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

SOC[7:0] bits (Successful Occurrence Counter)

The SOC[7:0] bits are used together with the EOC[7:0] bits to support an option for host-controlled fall-back to payload bit rate identical to arbitration bit rate when messages utilizing the reduced payload bit length have significant higher error rates compared to other messages.

The SOC[7:0] bits are set only by CAN-FD module logic. Writing any other value has no effect.

These bits are updated when the occurrence of any error-free messages on the bus is detected through reception or transmission. When the counter reaches the value of H’FF, the update stops. In Loopback mode, the counter is incremented twice.

These bits are cleared by writing 1 to CFDCnFDCTR.SOCCLR.

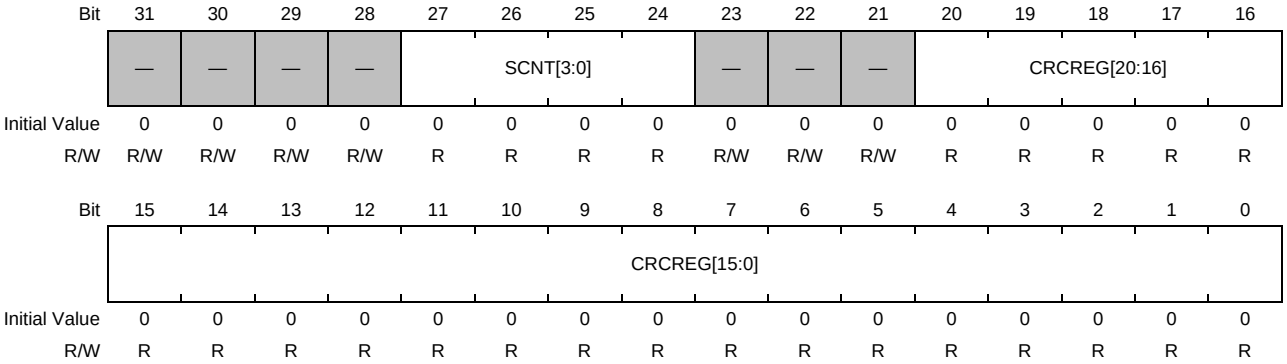
These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

30.2.9 Channel n CAN-FD CRC Register (CFDCnFDCRC) (n = 0, 1)

The CFDCnFDCRC register (n = 0, 1) holds the CRC value calculated for the CAN-FD frame.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1410 + H'20 × n



Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
27 to 24	SCNT[3:0]	H'0	R	Stuff Bit Count These bits shows the stuff bit count (mod 8) for the CAN-FD frame.
23 to 21	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.
20 to 0	CRCREG [20:0]	All 0	R	CRC Register Value These bits show the CRC value calculated for the CAN-FD frame.

CRCREG[20:0] bits (CRC Register Value)

The CRCREG[20:0] bits contain the CRC value calculated by the CAN-FD channel logic when the CFDCnCTR.CTME bit is enabled.

The CFDCnFDCRC.CRCREG value is updated in the first bit of the CRC field of the CAN-FD frame (reception and transmission).

When the CFDCnCTR.CTME bit is 0, the CRCREG[20:0] bits are always read as 0.

When bit 17th of the CRC field is used, CRCREG[20:17] are always read as 0.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

SCNT[3:0] bits (Stuff Bit Count)

The SCNT[3:0] bits contain the stuff count value of the CAN-FD frame. These bits indicate the number of inserted stuff bits (modulo 8, Gray-coded) for a CAN-FD frame when the CFDCnCTR.CTME bit is enabled in CFDCnFDCRC.SCNT[3:1]. SCNT[0] is the parity bit.

When the CFDCnCTR.CTME bit is 0, the SCNT[3:0] bits are always read as 0.

The SCNT value is updated in the first bit of CRC field of the CAN-FD frame (reception and transmission).

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

30.2.10 Global IP Version Register (CFDGIPV)

The CFDGIPV register shows the release version of the CAN-FD module.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0080

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	PSI[13:0]													
Initial Value	0	0	1	1	0	0	0	1	0	0	1	0	0	1	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	IPT[1:0]		IPV[7:0]							
Initial Value	0	0	0	0	0	0	0	*1	0	1	0	0	0	0	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00	R/W	Reserved These bits are read as 0. The write value should be always 0.
29 to 16	PSI[13:0]	11 0001 0010 0100	R	Parameter Status Information These bits show the IP configuration.
15 to 10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
9, 8	IPT[1:0]	0*1	R	IP Type These bits show the IP type used in the product. 00: CAN IP 01: CAN-FD IP 1x: Reserved
7 to 0	IPV[7:0]	H'43	R	IP Version These bits show the main release number.

Bits	Description
[29:27]	Number of channel (110b: 2 channels)
[26:24]	Number of TXMB (001b: 32/channel)
[23:21]	Number of AFL entry (001b: 128/channel)
[20:17]	Number of pool buffer (0010b: 64/channel)
[16]	Same ID overwrite function of TXQ (0b: Support)

Note 1. The value depends on product package type.

30.2.11 Global Configuration Register (CFDGCFG)

The CFDGCFG register is used to select the transmission priority to be used for all the TX message buffers and the clock source for the CAN protocol engine of all CAN channels. The CFDGCFG register is also used to select the source for the timestamp clock and to configure the frequency for the timestamp clock and interval timer reference clock.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0084

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ITRCP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSBTCS[2:0]			TSSS	TSP[3:0]				—	—	CMPOC	DCS	MME	DRE	DCE	TPRI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	ITRCP[15:0]	H'0000	R/W	Interval Timer Reference Clock Prescaler FIFO interval timer prescaler value
15 to 13	TSBTCS[2:0]	000	R/W	Timestamp Bit Time Channel Select 000: Select clock from channel 0 001: Select clock from channel 1 Others: Setting prohibited
12	TSSS	0	R/W	Timestamp Source Select 0: Source clock for timestamp counter is peripheral clock 1: Source clock for timestamp counter is bit time clock
11 to 8	TSP[3:0]	H'0	R/W	Timestamp Prescaler H'0: Timestamp prescaler = 1 H'1: Timestamp prescaler = 2 H'2: Timestamp prescaler = 4 H'3: Timestamp prescaler = 8 ⋮ H'D: Timestamp prescaler = 8192 H'E: Timestamp prescaler = 16384 H'F: Timestamp prescaler = 32768
7, 6	—	00	R/W	Reserved These bits are read as 0. The write value should be always 0.
5	CMPOC	0	R/W	CAN-FD Message Payload Overflow Configuration 0: Message is rejected 1: Message payload is cut to fit to configured message size
4	DCS	0	R/W	Data Link Controller Clock Select 0: PCLKM 1: PCLKCAN
3	MME	0	R/W	Mirror Mode Enable 0: Mirror mode disabled 1: Mirror mode enabled

Bit	Bit Name	Initial Value	R/W	Description
2	DRE	0	R/W	DLC Replacement Enable 0: DLC replacement disabled 1: DLC replacement enabled
1	DCE	0	R/W	DLC Check Enable 0: DLC check disabled 1: DLC check enabled
0	TPRI	0	R/W	Transmission Priority 0: ID priority 1: Message buffer number priority

TPRI bit (Transmission Priority)

The TPRI bit selects the transmission priority for all CAN channels.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

Message buffer number priority should not be used together with TX queue transmission.

DCE bit (DLC Check Enable)

The DCE bit enables data length code (DLC) check for all CAN channels.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

DRE bit (DLC Replacement Enable)

When the DRE bit is 1 and the DCE is 1, the CAN-FD stores the configured value (CFDGAFLP0n.GAFLDLC) of the DLC in the destination RX message buffer or FIFO buffer if the DLC check passes. Otherwise, the DLC value in the destination RX message buffer or FIFO buffer is unchanged.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

MME bit (Mirror Mode Enable)

The MME bit enables the Mirror mode for all CAN channels.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

DCS bit (Data Link Controller Clock Select)

The DCS bit selects the clock source for CAN communication.

Do not write to this bit in GL_SLEEP or GL_OPERATION mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

CMPOC bit (CAN-FD Message Payload Overflow Configuration)

The CMPOC bit controls the message payload acceptance mechanism when the received payload is higher than the message buffer payload size CFDRMNB.RMPLS, CFDRFCCn.RFPLS, and CFDCFCCn.CFPLS. The received message payload is always compared with the available message payload size in the message buffer.

Do not write to this bit in GL_SLEEP or GL_OPERATION mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

When this bit is set and payload overflow occurs, the DLC value is stored in the RX message buffer or FIFO buffer unchanged.

TSP[3:0] bits (Timestamp Prescaler)

The value configured in the TSP[3:0] bits defines the period of the clock source used for the timestamp counter.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

TSSS bit (Timestamp Source Select)

The TSSS bit allows the selection of the clock source for the timestamp counter.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode. Additionally, do not set this bit to 1 when CAN-FD communication is used.

Note: The bit time clock varies depending on the nominal and data rate bit configuration.

TSBTCS[2:0] bits (Timestamp Bit Time Channel Select)

The TSBTCS[2:0] bits allow the selection of the bit time clock of a particular channel for the timestamp counter.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

ITRCP[15:0] bits (Interval Timer Reference Clock Prescaler)

The ITRCP[15:0] bits allow the definition of a reference clock for the FIFO interval timer source clock.

When these bits are H’0000, the timer is disabled.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

30.2.12 Global Control Register (CFDGCTR)

The CFDGCTR register controls the global mode of the CAN-FD module and the timestamp function. The register also enables and disables the global error interrupts.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0088

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TSRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MOWEIE	QMEIE	—	QOWEIE	CMPOFIE	THLEIE	MEIE	DEIE	—	—	—	—	—	GSLPR	GMDC[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
16	TSRST	0	R/W	Timestamp Reset 0: Timestamp not reset 1: Timestamp reset
15	MOWEIE	0	R/W	GW FIFO Message overwrite Error Interrupt Enable 0: GW FIFO message overwrite error interrupt disabled 1: GW FIFO message overwrite error interrupt enabled
14	QMEIE	0	R/W	TXQ Message Lost Error Interrupt Enable 0: TXQ message lost error interrupt disabled 1: TXQ message lost error interrupt enabled
13	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
12	QOWEIE	0	R/W	TXQ Message Overwrite Error Interrupt Enable 0: TXQ message overwrite error interrupt disabled 1: TXQ message overwrite error interrupt enabled
11	CMPOFIE	0	R/W	CAN-FD Message Payload Overflow Flag Interrupt Enable 0: CAN-FD message payload overflow flag interrupt disabled 1: CAN-FD message payload overflow flag interrupt enabled
10	THLEIE	0	R/W	TX History List Entry Lost Interrupt Enable 0: TX history list entry lost interrupt disabled 1: TX history list entry lost interrupt enabled
9	MEIE	0	R/W	Message Lost Error Interrupt Enable 0: Message lost error interrupt disabled 1: Message lost error interrupt enabled
8	DEIE	0	R/W	DLC Check Interrupt Enable 0: DLC check interrupt disabled 1: DLC check interrupt enabled
7 to 3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.

Bit	Bit Name	Initial Value	R/W	Description
2	GSLPR	1	R/W	Global Sleep Request 0: Global sleep request disabled 1: Global sleep request enabled
1, 0	GMDC[1:0]	01	R/W	Global Mode Control 00: Global operation mode request 01: Global reset mode request 10: Global halt mode request 11: Keep current value

GMDC[1:0] bits (Global Mode Control)

The GMDC[1:0] bits can be used to configure the modes for the CAN-FD module. Additionally, if CFDGCTR.GSLPR bit is 1 when the CAN-FD module is in Reset mode, the CAN-FD module transits to Global Sleep mode. Additionally, if CFDGCTR.GSLPR is 1 when the CAN-FD module is in Reset Mode, then the CAN-FD module transits to Global Sleep Mode.

Setting the GMDC[1:0] bits to 11b has no effect. Mode transition is described in detail in **Section 30.3.2, Global Modes**.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

GSLPR bit (Global Sleep Request)

The GSLPR bit globally selects the sleep request for CAN-FD module including all CAN channels. Channel sleep request is set automatically for all channels.

Only write to this bit when the CAN-FD module is in GL_RESET or GL_SLEEP mode.

DEIE bit (DLC Check Interrupt Enable)

When the DEIE bit is 1, an interrupt is generated if a DLC error is detected in the received frames.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

MEIE bit (Message Lost Error Interrupt Enable)

When the MEIE bit is 1, an interrupt is generated if a message lost condition occurs.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

THLEIE bit (TX History List Entry Lost Interrupt Enable)

When the THLEIE bit is 1, an interrupt is generated if a TX history list entry lost condition occurs.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

CMPOFIE bit (CAN-FD Message Payload Overflow Flag Interrupt Enable)

When the CMPOFIE bit is 1, an interrupt is generated when a CAN-FD message payload overflow condition occurs.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

QOWEIE bit (TXQ Message Overwrite Error Interrupt Enable)

When the QOWEIE bit is 1, an interrupt is generated when a TXQ message overwrite condition occurs.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

QMEIE bit (TXQ Message Lost Error Interrupt Enable)

When the QMEIE bit is 1, an interrupt is generated when a TXQ message lost condition occurs.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

MOWEIE bit (GW FIFO Message Overwrite Error Interrupt Enable)

When the MOWEIE bit is 1, an interrupt is generated in GW mode and a GW FIFO message overwrite condition occurs.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

TSRST bit (Timestamp Reset)

When the TSRST bit is 1, the Global Timestamp Register is reset to H’0000.

Do not write to this bit when the CAN-FD module is in GL_SLEEP or GL_RESET mode.

Read value is always 0.

This bit is cleared automatically by the CAN-FD logic.

30.2.13 Global Status Register (CFDGSTS)

The CFDGSTS register indicates the global status of the CAN-FD module.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'008C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	GRAMINIT	GSLPSTS	GHLTSTS	GRSTSTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
3	GRAMINIT	1	R	Global RAM Initialization 0: RAM initialization is complete 1: RAM initialization is ongoing
2	GSLPSTS	1	R	Global Sleep Status 0: Not in Sleep mode 1: In Sleep mode
1	GHLTSTS	0	R	Global Halt Status 0: Not in Halt mode 1: In Halt mode
0	GRSTSTS	1	R	Global Reset Status 0: Not in Reset mode 1: In Reset mode

GRSTSTS bit (Global Reset Status)

The GRSTSTS bit indicates the status of Global CAN-FD module Reset mode.

This bit is set automatically when the CAN-FD module enters GL_RESET mode. When the mode changes from GL_RESET mode to GL_SLEEP mode, this bit remains set.

This bit is cleared automatically when the CAN-FD module exits the GL_RESET mode.

GHLTSTS bit (Global Halt Status)

The GHLTSTS bit indicates the status of Global CAN-FD module Halt mode.

This bit is set automatically when the CAN-FD module enters GL_HALT mode.

This bit is cleared automatically when the CAN-FD module exits the GL_HALT mode.

GSLPSTS bit (Global Sleep Status)

The GSLPSTS bit indicates the status of Global CAN-FD module Sleep mode.

This bit is set automatically when the CAN-FD module enters GL_SLEEP mode.

This bit is cleared automatically when the CAN-FD module exits the GL_SLEEP mode.

GRAMINIT bit (Global RAM Initialization)

The GRAMINIT bit indicates the status of Global CAN-FD module RAM initialization.

This bit is set automatically when the CAN-FD module enters GL_SLEEP mode after a hardware reset.

This bit is cleared automatically when the CAN-FD module completed RAM initialization.

30.2.14 Global Error Flag Register (CFDGERFL)

The CFDGERFL register indicates the detection of global errors.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0090

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EEF1	EEF0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	MOWES	QMES	—	QOWES	CMPOF	THLES	MES	DEF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
17	EEF1	0	R/W	ECC Error Flag for Channel 1 0: ECC error not detected during TX-SCAN 1: ECC error detected during TX-SCAN
16	EEF0	0	R/W	ECC Error Flag for Channel 0 0: ECC error not detected during TX-SCAN 1: ECC error detected during TX-SCAN
15 to 8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
7	MOWES	0	R	Message Overwrite Error Status 0: Message overwrite error not detected 1: Message overwrite error detected
6	QMES	0	R	TXQ Message Lost Error Status 0: TXQ message lost error not detected 1: TXQ message lost error detected
5	—	0	R	Reserved This bit is read as 0.
4	QOWES	0	R	TXQ Message Overwrite Error Status 0: TXQ message overwrite error not detected 1: TXQ message overwrite error detected
3	CMPOF	0	R/W	CAN-FD Message Payload Overflow Flag 0: CAN-FD message payload overflow not detected 1: CAN-FD message payload overflow detected
2	THLES	0	R	TX History List Entry Lost Error Status 0: TX history list entry lost error not detected 1: TX history list entry lost error detected
1	MES	0	R	Message Lost Error Status 0: Message lost error not detected 1: Message lost error detected
0	DEF	0	R/W	DLC Error Flag 0: DLC error not detected 1: DLC error detected

DEF bit (DLC Error Flag)

The DEF bit indicates the error status of the DLC.

Do not write to this bit when the CAN-FD module is in GL_SLEEP or GL_RESET mode. Writing 1 has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

This bit is set automatically when a DLC error is detected in a received frame.

The bit is cleared by writing 0 to it.

This bit is cleared automatically in GL_RESET mode.

MES bit (Message Lost Error Status)

The MES bit indicates status of the message lost error.

This bit is set automatically when a FIFO message lost error is detected.

This bit is cleared automatically when:

- All FIFO message lost flags are cleared
- The CAN-FD module is in GL_RESET mode.

THLES bit (TX History List Entry Lost Error Status)

The THLES bit indicates status of the TX history list entry lost error.

This bit is set automatically when a TX history list entry lost error is detected.

This bit is cleared automatically when:

- All TX history list entry lost flags are cleared
- The CAN-FD module is in GL_RESET mode.

CMPOF bit (CAN-FD Message Payload Overflow Flag)

The CMPOF bit is set automatically when a CAN-FD message payload overflow is detected on at least one channel.

Do not write to this bit when the CAN-FD module is in GL_SLEEP or GL_RESET mode.

This bit is cleared by writing 0 to it. Writing 1 to this bit has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

This bit is cleared automatically in GL_RESET mode.

QOWES bit (TXQ Message Overwrite Error Status)

The QOWES bit is set automatically when a TXQ message overwrite error is detected.

This bit is cleared automatically when all TXQ message overwrite flags are cleared.

This bit is cleared automatically in GL_RESET mode.

QMES bit (TXQ Message Lost Error Status)

The QMES bit is set automatically when a TXQ message lost error is detected.

This bit is cleared automatically when all TXQ message lost flags are cleared.

This bit is cleared automatically in GL_RESET mode.

MOWES bit (Message Overwrite Error Status)

The MOWES bit is set automatically when GW mode and COMFIFO message overwrite error is detected.

This bit is cleared automatically when all COMFIFO Message Overwrite flags are cleared.

This bit is cleared automatically in GL_RESET mode.

EEF0 bit (ECC Error Flag for Channel 0)

The EEF0 bit specifies whether an ECC error has occurred on Channel 0.

Do not write to this bit when the CAN-FD module is in GL_SLEEP or GL_RESET mode. Writing 1 to this bit has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

The bit is cleared by writing 0 to it. This bit is cleared automatically in GL_RESET mode.

EEF1 bit (ECC Error Flag for Channel 1)

The EEF1 bit specifies whether an ECC error has occurred on Channel 1.

Do not write to this bit when the CAN-FD module is in GL_SLEEP or GL_RESET mode. Writing 1 to this bit has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

The bit is cleared by writing 0 to it. This bit is cleared automatically in GL_RESET mode.

30.2.15 Global TX Interrupt Status Register 0 (CFDGTINTSTS0)

The CFDGTINTSTS0 indicates the detection of transmit specific interrupts.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1300

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	CFOTIF1	TQOFIF1	THIF1	CFTIF1	TQIF1	TAIF1	TSIF1	—	CFOTIF0	TQOFIF0	THIF0	CFTIF0	TQIF0	TAIF0	TSIF0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
14	CFOTIF1	0	R	COM FIFO One Frame Transmission Interrupt Flag Channel 1 0: Channel 1 COM FIFO One Frame Transmission Interrupt flag not set 1: Channel 1 COM FIFO One Frame Transmission Interrupt flag set
13	TQOFIF1	0	R	TX Queue One Frame Transmission Interrupt Flag Channel 1 0: Channel 1 TX Queue One Frame Transmission Interrupt flag not set 1: Channel 1 TX Queue One Frame Transmission Interrupt flag set
12	THIF1	0	R	TX History List Interrupt Channel 1 0: Channel 1 TX History List Interrupt flag not set 1: Channel 1 TX History List Interrupt flag set
11	CFTIF1	0	R	COM FIFO TX/GW Mode Interrupt Flag Channel 1 0: Channel 1 COM FIFO TX/GW Mode Interrupt flag not set 1: Channel 1 COM FIFO TX/GW Mode Interrupt flag set
10	TQIF1	0	R	TX Queue Interrupt Flag Channel 1 0: Channel 1 TX Queue Interrupt flag not set 1: Channel 1 TX Queue Interrupt flag set
9	TAIF1	0	R	TX Abort Interrupt Flag Channel 1 0: Channel 1 TX Abort Interrupt flag not set 1: Channel 1 TX Abort Interrupt flag set
8	TSIF1	0	R	TX Successful Interrupt Flag Channel 1 0: Channel 1 TX Successful Interrupt flag not set 1: Channel 1 TX Successful Interrupt flag set
7	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
6	CFOTIF0	0	R	COM FIFO One Frame Transmission Interrupt Flag Channel 0 0: Channel 0 COM FIFO One Frame Transmission Interrupt flag not set 1: Channel 0 COM FIFO One Frame Transmission Interrupt flag set
5	TQOFIF0	0	R	TX Queue One Frame Transmission Interrupt Flag Channel 0 0: Channel 0 TX Queue One Frame Transmission Interrupt flag not set 1: Channel 0 TX Queue One Frame Transmission Interrupt flag set

Bit	Bit Name	Initial Value	R/W	Description
4	THIF0	0	R	TX History List Interrupt Channel 0 0: Channel 0 TX History List Interrupt flag not set 1: Channel 0 TX History List Interrupt flag set
3	CFTIF0	0	R	COM FIFO TX/GW Mode Interrupt Flag Channel 0 0: Channel 0 COM FIFO TX/GW Mode Interrupt flag not set 1: Channel 0 COM FIFO TX/GW Mode Interrupt flag set
2	TQIF0	0	R	TX Queue Interrupt Flag Channel 0 0: Channel 0 TX Queue Interrupt flag not set 1: Channel 0 TX Queue Interrupt flag set
1	TAIF0	0	R	TX Abort Interrupt Flag Channel 0 0: Channel 0 TX Abort Interrupt flag not set 1: Channel 0 TX Abort Interrupt flag set
0	TSIF0	0	R	TX Successful Interrupt Flag Channel 0 0: Channel 0 TX Successful Interrupt flag not set 1: Channel 0 TX Successful Interrupt flag set

TSIFn bit (TX Successful Interrupt Flag Channel n (n = 0, 1))

The CFDGTINTSTS0.TSIFn bit is set to 1 when the TX Successful Interrupt flag of the related channel is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX MB Result Status bits are cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

TAIFn bit (TX Abort Interrupt Flag Channel n (n = 0, 1))

The CFDGTINTSTS0.TAIFn bit is set to 1 when the TX Abort Interrupt flag of the related channel is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX MB Result Status bits are cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

TQIFn bit (TX Queue Interrupt Flag Channel n (n = 0, 1))

The CFDGTINTSTS0.TQIFn bit is set to 1 when the TX Queue Interrupt flag of the related channel is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX Queue Interrupt flag is cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

CFTIFn bit (COM FIFO in TX/GW Mode Interrupt Flag Channel n (n = 0, 1))

The CFDGTINTSTS0.CFTIFn bit is set to 1 when the related COM TX/GW FIFO Mode Interrupt flag (CFDCFSTSn.CFTXIF) is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related COM TX/GW FIFO Mode Interrupt flag (CFDCFSTSn.CFTXIF) is cleared (when the interrupt enable is disabled)

- When in GL_RESET or CH_RESET mode.

THIFn bit (TX History List Interrupt Flag Channel n (n = 0, 1))

The CFDGTINTSTS0.THIFn bit is set to 1 when the related TX History List Interrupt flag (CFDTHLSTSn.THLIF) is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX History List Interrupt flag (CFDTHLSTSn.THLIF) is cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

TQOFIFn bit (TX Queue One Frame Transmission Interrupt Flag Channel n (n = 0, 1))

The CFDGTINTSTS0.TQOFIFn bit is set to 1 when the TX Queue One Frame Transmission Interrupt flag of the related channel is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX Queue One Frame Transmission Interrupt flag is cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

CFOTIFn bit (COM FIFO One Frame Transmission Interrupt Flag Channel n (n = 0, 1))

The CFDGTINTSTS0.CFOTIFn bit is set to 1 when the related COM FIFO One Frame Transmission Interrupt flag (CFDCFSTSn.CFOFTXIF) is set (when the Interrupt is enabled).

This bit is cleared automatically:

- When the related COM FIFO One Frame Transmission Interrupt flag (CFDCFSTSn.CFOFTXIF) is cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

30.2.16 Global Timestamp Counter Register (CFDGTSC)

The CFDGTSC register stores the timestamp based on the selected configuration.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0094

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
15 to 0	TS[15:0]	All 0	R	Timestamp value

TS[15:0] bits (Timestamp value)

The proper incrementing of the Time Stamp counter cannot be guaranteed when moving to Halt state.

The Timestamp value is stored in the Global Timestamp Counter register based on the configuration of TSSS, TSBTCS, and TSP.

Do not write to these bits when the CAN-FD module is in GL_RESET or GL_SLEEP mode.

The TS[15:0] bits are cleared automatically in GL_RESET mode.

30.2.17 Global Acceptance Filter List Entry Control Register (CFDGAFLECTR)

The CFDGAFLECTR register is used to select the Global Acceptance Filter List page for reading or writing entries into the Global Acceptance Filter List.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0098

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AFLDAE	—	AFLPN[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
8	AFLDAE	0	R/W	Acceptance Filter List Data Access Enable 0: Acceptance Filter List data access disabled 1: Acceptance Filter List data access enabled
7	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
6 to 0	AFLPN[6:0]	All 0	R/W	Acceptance Filter List Page Number Select an Acceptance Filter List page

AFLPN[6:0] bits (Acceptance Filter List Page Number)

The AFLPN[6:0] bits select the page number to access the desired RAM area of the Acceptance Filter List. One Acceptance Filter List page consists of 16 Acceptance Filter List entries.

Read/write accesses to the Acceptance Filter List can only be performed through a fixed window.

Do not write to these bits when the CAN-FD module is in GL_SLEEP mode. Enter only the values between H'00 and H'0F, inclusive.

AFLDAE bit (Acceptance Filter List Data Access Enable)

The AFLDAE bit prevents write access to the Acceptance Filter List when cleared after configuration of the Acceptance Filter List.

Data can be read from the Acceptance Filter List independent of the status of this bit.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode. Set this bit to enable write access for the Acceptance Filter List.

30.2.18 Global Acceptance Filter List Configuration Register 0 (CFDGAFLCFG0)

The CFDGAFLCFG0 register is used to define the number of rules for entries in the Acceptance Filter List, applicable for channels 0 to 1.

The total number of available entries in the Acceptance Filter List is 256 for 2 CAN channels. However, the filters can be allocated flexibly to the different channels depending on requirements as long as both of the following conditions are satisfied:

- The maximum number of acceptance filter per channel is 256
- The total number of rules defined for all channels is not exceeding the number of available entries in the Acceptance Filter List.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'009C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	RNC0[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	RNC1[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
24 to 16	RNC0[8:0]	All 0	R/W	Rule Number for Channel 0 Number of rules dedicated to channel 0
15 to 9	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
8 to 0	RNC1[8:0]	All 0	R/W	Rule Number for Channel 1 Number of rules dedicated to channel 1

RNCn[8:0] bits (Rule Number for Channel n (n = 0, 1))

The RNCn[8:0] bits define the number of rules in the Acceptance Filter List for channel n.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

30.2.19 Global Acceptance Filter List ID Register n (CFDGAFLIDn) (n = 0 to 15)

The CFDGAFLIDn register (n = 0 to 15) is used to configure the ID field for the rules of entries in the Global Acceptance Filter List.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1800 + H'10 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDE	GAFLRTR	GAFLLB	GAFLID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	GAFLIDE	0	R/W	Global Acceptance Filter List Entry IDE Field 0: Standard identifier of rule entry ID is valid for acceptance filtering 1: Extended identifier of rule entry ID is valid for acceptance filtering
30	GAFLRTR	0	R/W	Global Acceptance Filter List Entry RTR Field 0: Data frame 1: Remote frame
29	GAFLLB	0	R/W	Global Acceptance Filter List Entry Loopback Configuration 0: Global Acceptance Filter List entry ID for acceptance filtering with attribute RX 1: Global Acceptance Filter List entry ID for acceptance filtering with attribute TX
28 to 0	GAFLID[28:0]	All 0	R/W	Global Acceptance Filter List Entry ID Field ID part of the Global Acceptance Filter List entry

GAFLID[28:0] bits (Global Acceptance Filter List Entry ID Field)

The GAFLID[28:0] bits represent the CAN identifier (ID) field of each entry in the Global Acceptance Filter List.

The acceptance filter process compares this field against the ID of a received CAN message. For alignment of these bits in standard and extended frame formats, see **Section 30.2.89 Identifier Bits Alignment.**

Do not write to these bits when CFDGAFLECTR.AFLDAE bit is 0.

Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLLB bit (Global Acceptance Filter List Entry Loopback Configuration)

The GAFLLB bit selects whether entry in the Global Acceptance Filter List gets the attribute RX or TX.

This attribute determines the validity of the entry in Mirror mode, Loopback test mode, and during standard (non-loopback) reception. See **Section 30.5.5, Loopback Modes** for detailed description of the validity of the Global Acceptance Filter List entry depending on transmitter/receiver case, the type of loopback mode, and RX/TX attribute.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLRTR bit (Global Acceptance Filter List Entry RTR Field)

The GAFLRTR bit allows the configuration of the specified frame format (data frame or remote frame) for each entry of the Global Acceptance Filter List. For each rule entry in a CAN channel, the acceptance filter process compares this bit against the RTR bit of the received CAN message.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLIDE bit (Global Acceptance Filter List Entry IDE Field)

The GAFLIDE bit allows the configuration of the ID format (standard ID or extended ID) for each entry in the Global Acceptance Filter List. For each rule entry in a CAN channel, the acceptance filter process compares this bit against the IDE bit of the received CAN message.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

30.2.20 Global Acceptance Filter List Mask Register n (CFDGAFLMn) (n = 0 to 15)

The CFDGAFLMn register (n = 0 to 15) is used to configure the Mask field of each rule for entries in the Global Acceptance Filter List.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1804 + H'10 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDEM	GAFLRTRM	GAFLIFL1	GAFLIDM[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLIDM[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	GAFLIDEM	0	R/W	Global Acceptance Filter List IDE Mask 0: IDE bit is not used for ID matching 1: IDE bit is used for ID matching
30	GAFLRTRM	0	R/W	Global Acceptance Filter List Entry RTR Mask 0: RTR bit is not used for ID matching 1: RTR bit is used for ID matching
29	GAFLIFL1	0	R/W	Global Acceptance Filter List Information Label 1 Global Acceptance Filter List information label bit 1
28 to 0	GAFLIDM [28:0]	All 0	R/W	Global Acceptance Filter List ID Mask Field Global Acceptance Filter List Mask field bits for ID field

GAFLIDM[28:0] bits (Global Acceptance Filter List ID Mask Field)

GAFLIDM[28:0] bits are the filter mask bits for the related bits in the CAN Identifier field of each Global Acceptance Filter List entry.

0: Corresponding STD-ID/EXT-ID bit is not used for ID matching
1: Corresponding STD-ID/EXT-ID bit is used for ID matching

Do not write to these bits when CFDGAFLECTR.AFLDAE bit is 0.
Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLIFL1 bit (Global Acceptance Filter List Information Label 1)

The GAFLIFL1 bit allows the configuration of a 2-bit information label to be attached to a received message accepted by the associated entry in the Global Acceptance Filter List. This bit is a MSB bit of an information label.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

This bit is stored in the Information Label Field [1] (CFDRMFDSTS.RMIFL[1], CFDRFFDSTS.RFIFL[1], CFDCFFDCSTSn.CFIFL[1]) of the storage location of an incoming message.

Note: This bit is stored in CFDTHLACC1n.TIFL[1] when CFDTHLCCn.THLDGE = 1 is set up using Gateway function.

GAFLRTRM bit (Global Acceptance Filter List Entry RTR Mask)

The GAFLRTRM bit allows the configuration of the RTR mask bit for each entry in the Global Acceptance Filter List.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLIDEM bit (Global Acceptance Filter List IDE Mask)

The GAFLIDEM bit allows the configuration of the IDE mask bit for each entry in the Global Acceptance Filter List.

When the IDE mask bit is 0, the ID comparison depends on the received IDE bit.

If the received IDE bit is 0, the STD-ID comparison takes place.

If the received IDE bit is 1, the EXT-ID comparison takes place.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

30.2.21 Global Acceptance Filter List Pointer 0 Register n (CFDGAFLP0n) (n = 0 to 15)

The CFDGAFLP0n register (n = 0 to 15) is used to configure the data length code (DLC), software pointer, single message buffer select, and message buffer direction pointer for each rule entry in the Global Acceptance Filter List.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1808 + H'10 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLPTR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLRMV	GAFLRMDP[6:0]							GAFLIFL0	GAFLSRD2	GAFLSRD1	GAFLSRD0	GAFLDLC[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	GAFLPTR [15:0]	All 0	R/W	Global Acceptance Filter List Pointer
15	GAFLRMV	0	R/W	Global Acceptance Filter List RX Message Buffer Valid 0: Single message buffer direction pointer is invalid 1: Single message buffer direction pointer is valid
14 to 8	GAFLRMDP [6:0]	All 0	R/W	Global Acceptance Filter List RX Message Buffer Direction Pointer RX message buffer number for storage of received messages
7	GAFLIFL0	0	R/W	Global Acceptance Filter List Information Label 0
6	GAFLSRD2	0	R/W	Global Acceptance Filter List Select Routing Destination 2 0: Routing target is CFIFO2 1: Routing target is TX Queue 2 instead of CFIFO2
5	GAFLSRD1	0	R/W	Global Acceptance Filter List Select Routing Destination 1 0: Routing target is CFIFO1 1: Routing target is TX Queue 1 instead of CFIFO1
4	GAFLSRD0	0	R/W	Global Acceptance Filter List Select Routing Destination 0 0: Routing target is CFIFO0 1: Routing target is TX Queue 0 instead of CFIFO0
3 to 0	GAFLDLC [3:0]	H'0	R/W	Global Acceptance Filter List DLC Field Minimum number of data bytes in a data frame required for acceptance

GAFLDLC[3:0] bits (Global Acceptance Filter List DLC Field)

The GAFLDLC[3:0] bits allow the configuration of a minimum data length code (DLC) value for a message to be accepted by the associated entry in the Global Acceptance Filter List (automatic DLC filter function).

DLC filter process is only passed if the DLC value of the message accepted by an entry in the Global Acceptance Filter List is equal to or higher than the DLC value configured for this associated Global Acceptance Filter List entry. Automatic DLC filter function is disabled for the corresponding rule entry when this field is set to 0.

Table 30.6 shows DLC value that can be configured.

Table 30.6 Configuration of DLC Value

DLC[3:0]	Description
0000b	DLC of received message = 0 or more (DLC filter check is disabled)
0001b	DLC of received message = 1 or more
0010b	DLC of received message = 2 or more
0011b	DLC of received message = 3 or more
0100b	DLC of received message = 4 or more
0101b	DLC of received message = 5 or more
0110b	DLC of received message = 6 or more
0111b	DLC of received message = 7 or more
1000b	DLC of received message = 8 or more
1001b	DLC of received message = 12 or more
1010b	DLC of received message = 16 or more
1011b	DLC of received message = 20 or more
1100b	DLC of received message = 24 or more
1101b	DLC of received message = 32 or more
1110b	DLC of received message = 48 or more
1111b	DLC of received message = 64

Do not write to these bits when CFDGAFLECTR.AFLDAE bit is 0.

Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLSRD0 bit (Global Acceptance Filter List Select Routing Destination 0)

The GAFLSRD0 bit changes a copy destination to CFIFO0 or TXQ0 by routing.

If this bit is set as 1, the preset value of CFDGAFLP1n.GAFLFDP selects TXQ0.

If this bit is set to 0, the preset value of CFDGAFLP1n.GAFLFDP selects CFIFO0.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0.

Only write to the bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLSRD1 bit (Global Acceptance Filter List Select Routing Destination 1)

The GAFLSRD1 bit changes a copy destination to CFIFO1 or TXQ1 by routing.

If this bit is set to 1, the preset value of CFDGAFLP1n.GAFLFDP selects TXQ1.

If this bit is set to 0, the preset value of CFDGAFLP1n.GAFLFDP selects CFIFO1.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0.

Only write to the bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLSRD2 bit (Global Acceptance Filter List Select Routing Destination 2)

The GAFLSRD2 bit changes a copy destination to CFIFO2 or TXQ2 by routing.

If this bit is set to 1, the preset value of CFDGAFLP1n.GAFLFDP selects TXQ2.

If this bit is set to 0, the preset value of CFDGAFLP1n.GAFLFDP selects CFIFO2.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0.

Only write to the bit when the CAN-FD module is in CH_RESET or CH_HALT mode.

GAFLIFL0 bit (Global Acceptance Filter List Information Label 0)

The GAFLIFL0 bit allows the configuration of a 2-bit information label that can be attached to a received message accepted by the related Global Acceptance Filter List entry. This bit is a LSB bit of an information label.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0.

Only write to the bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

This bit is stored in Information Label Field[0] (CFDRMFDSTS.RMIFL[0], CFDRFFDSTS.RFIFL[0], CFDCFFDCSTSn.CFIFL[0]) of the storage location of an incoming message.

This bit is stored in CFDTHLACC1n.TIFL[0] when CFDTHLCCn.THLDGE = 1 is set up using the gateway function.

GAFLRMDP[6:0] bits (Global Acceptance Filter List RX Message Buffer Direction Pointer)

The GAFLRMDP[6:0] bits allow the configuration of a single reception message buffer as the destination target for a received message that passes the acceptance check of the related Global Acceptance Filter List entry. The value entered is the single destination message buffer number.

Do not write to these bits when CFDGAFLECTR.AFLDAE bit is 0.

Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

CFDRMNB.NRXMB[7:0] is the value entered in the RX Message Buffer Number Register to configure the number of RX message buffers. The value to be entered in CFDGAFLP0n.GAFLRMDP[6:0] bits should only be between H’00 and CFDRMNB.NRXMB[7:0] to 1 less.

If CFDRMNB.NRXMB[7:0] = H’00, the GAFLRMV bit should be configured as 0.

GAFLRMV bit (Global Acceptance Filter List RX Message Buffer Valid)

The GAFLRMV bit allows the enabling or disabling of a single reception message buffer as the target for a received message that passes the acceptance check of the related Global Acceptance Filter List entry.

Do not write to these bits when CFDGAFLECTR.AFLDAE bit is 0.

Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLPTR[15:0] bits (Global Acceptance Filter List Pointer)

The GAFLPTR[15:0] bits allow the configuration of a 16-bit pointer to be attached to a received message accepted by the related Global Acceptance Filter List entry. The pointer is added during message storage in the Message Buffer area and can be used by the application as a support function. The pointer information can be used for example, to support PDU Identifier allocation for the received message in AUTOSAR systems.

Do not write to these bits when CFDGAFLECTR.AFLDAE bit is 0.

Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

30.2.22 Global Acceptance Filter List Pointer 1 Register n (CFDGAFLP1n) (n = 0 to 15)

The CFDGAFLP1n register (n = 0 to 15) is used to configure the FIFO direction pointer fields in each Rule Entry of the Global Acceptance Filter List.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'180C + H'10 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	GAFLFDP[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
13 to 0	GAFLFDP [13:0]	All 0	R/W	Global Acceptance Filter List FIFO Direction Pointer FIFO direction pointer bits for received message storage

GAFLFDP[13:0] bits (Global Acceptance Filter List FIFO Direction Pointer)

The GAFLFDP[13:0] bits allow the configuration of FIFO buffers as the target for a received message that passes the acceptance check of the related Global Acceptance Filter List entry. Each bit of the CFDGAFLP1n.GAFLFDP[13:0] is configured as dedicated FIFO.

Table 30.7 Global Acceptance Filter List entry (1/2)

Bit	Value (binary)	Function
0	0	Disable RX FIFO 0 as target for reception
	1	Enable RX FIFO 0 as target for reception
1	0	Disable RX FIFO 1 as target for reception
	1	Enable RX FIFO 1 as target for reception
2	0	Disable RX FIFO 2 as target for reception
	1	Enable RX FIFO 2 as target for reception
3	0	Disable RX FIFO 3 as target for reception
	1	Enable RX FIFO 3 as target for reception
4	0	Disable RX FIFO 4 as target for reception
	1	Enable RX FIFO 4 as target for reception
5	0	Disable RX FIFO 5 as target for reception
	1	Enable RX FIFO 5 as target for reception
6	0	Disable RX FIFO 6 as target for reception
	1	Enable RX FIFO 6 as target for reception

Table 30.7 Global Acceptance Filter List entry (2/2)

Bit	Value (binary)	Function
7	0	Disable RX FIFO 7 as target for reception
	1	Enable RX FIFO 7 as target for reception
8	0	Disable Common FIFO 0 and Channel 0 TX Queue 0 as target for reception
	1	GAFLSRD0 = 0: Enable Common FIFO 0 as target for reception GAFLSRD0 = 1: Enable Channel 0 TX Queue 0 as target for reception
9	0	Disable Common FIFO 1 and Channel 0 TX Queue 1 as target for reception
	1	GAFLSRD1 = 0: Enable Common FIFO 1 as target for reception GAFLSRD1 = 1: Enable Channel 0 TX Queue 1 as target for reception
10	0	Disable Common FIFO 2 and Channel 0 TX Queue 2 as target for reception
	1	GAFLSRD2 = 0: Enable Common FIFO 2 as target for reception GAFLSRD2 = 1: Enable Channel 0 TX Queue 2 as target for reception
11	0	Disable Common FIFO 3 and Channel 1 TX Queue 0 as target for reception
	1	GAFLSRD0 = 0: Enable Common FIFO 3 as target for reception GAFLSRD0 = 1: Enable Channel 1 TX Queue 0 as target for reception
12	0	Disable Common FIFO 4 and Channel 1 TX Queue 1 as target for reception
	1	GAFLSRD1 = 0: Enable Common FIFO 4 as target for reception GAFLSRD1 = 1: Enable Channel 1 TX Queue 1 as target for reception
13	0	Disable Common FIFO 5 and Channel 1 TX Queue 2 as target for reception
	1	GAFLSRD2 = 0: Enable Common FIFO 5 as target for reception GAFLSRD2 = 1: Enable Channel 1 TX Queue 2 as target for reception

Do not write to these bits when CFDGAFLECTR.AFLDAE bit is 0. Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

For storage in TX queue, TX queue buffers of a target that is in GW mode is possible.

Only one of the following configurations is valid:

- Up to 8 destination FIFO buffers
- 7 destination FIFO buffers plus one RX message buffer
- 8 destination TX queue buffers
- 7 destination TX queue buffers plus one RX message buffer
- A maximum of 8 destinations in all at FIFO buffer and TX queue buffer.

30.2.23 RX Message Buffer Number Register (CFDRMNB)

The CFDRMNB register is used to configure the total number of RX message buffers allocated to all channels.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'00AC

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RMPLS[2:0]			NRXMB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
10 to 8	RMPLS[2:0]	000	R/W	Reception Message Buffer Payload Data Size 000: 8 bytes 001: 12 bytes 010: 16 bytes 011: 20 bytes 100: 24 bytes 101: 32 bytes 110: 48 bytes 111: 64 bytes
7 to 0	NRXMB[7:0]	H'00	R/W	Number of RX Message Buffers

NRXMB[7:0] bits (Number of RX Message Buffers)

The NRXMB[7:0] bits are used to configure the number of RX message buffers.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

Enter only values between 0 and 32 inclusive, with H'00 indicating that no RX message buffer is allocated.

RMPLS[2:0] bits (Reception Message Buffer Payload Data Size)

The RMPLS[2:0] bits are used to configure the message buffer payload data size.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

30.2.24 RX Message Buffer New Data Register 0 (CFDRMND0)

The CFDRMND0 specifies the new data storage status of the RX message buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'00B0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMNS[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMNS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RMNS[31:0]	All 0	R/W	RX Message Buffer New Data Status 0: New data not stored in corresponding RX message buffer 1: New data stored in corresponding RX message buffer

RMNS[31:0] bits (RX Message Buffer New Data Status)

The RMNSu[31:0] bits indicate the status of new data for the corresponding RX message buffer. RMNS bit [0] corresponds to RX message buffer [0] and so on.

no_of_channels = 2
no_of_CFDRMBCPs_per_channel = (number of RX message buffer components per channel) = 16
no_of_CFDRMBCPs = no_of_channels × no_of_CFDRMBCPs_per_channel (2 × 16 = 32)
no_of_bits_per_register = 32
no_of_CFDRMNDs (number of CFDRMND Registers) = no_of_CFDRMBCPs/no_of_bits_per_register (32/32 = 1)
t = [0 ... no_of_CFDRMNDs - 1]
u = [t × 32 ... (no_of_CFDRMBCPs − ((no_of_CFDRMNDs − 1 − t) × 32) − 1)]

The parameter t can be calculated from the target of the New Data Status flag (u) using the formula t = floor (u/32)

Bit position can be calculated using the formula (u − (t × 32))

Do not write to these bits when the CAN-FD module is in GL_RESET or GL_SLEEP mode. Writing 1 has no effect.

These bits cannot be cleared when message storage in the corresponding RX message buffer is in progress.

Do not use the bit clear instruction to clear these bits. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

These bits are set automatically when storage of new messages are in the corresponding RX message buffer. These bits are cleared by writing 0. These bits are cleared automatically when the CAN-FD module is in GL_RESET mode.

When CFDRMNB.RMPLS = 000b (maximum 8 bytes payload), the duration of message storage is 6 PCLKM cycles.

When CFDRMNB.RMPLS > 000b, the duration of message storage is 6 PCLKM cycles + 1 for each 4 bytes (maximum of 20 PCLKM cycles for 64 bytes).

30.2.25 RX FIFO Configuration/Control Register n (CFDRFCCn) (n = 0 to 7)

The CFDRFCCn registers (n = 0 to 7) are used to configure and control the eight RX FIFOs.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'00C0 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RFFIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFIGCV[2:0]			RFIM	—	RFDC[2:0]			—	RFPLS[2:0]			—	—	RFIE	RFE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
16	RFFIE	0	R/W	RX FIFO Full Interrupt Enable 0: FIFO interrupt generation disabled 1: FIFO interrupt generation enabled
15 to 13	RFIGCV[2:0]	000	R/W	RX FIFO Interrupt Generation Counter Value 000: Interrupt generated when FIFO is 1/8th full 001: Interrupt generated when FIFO is 1/4th full 010: Interrupt generated when FIFO is 3/8th full 011: Interrupt generated when FIFO is 1/2 full 100: Interrupt generated when FIFO is 5/8th full 101: Interrupt generated when FIFO is 3/4th full 110: Interrupt generated when FIFO is 7/8th full 111: Interrupt generated when FIFO is full
12	RFIM	0	R/W	RX FIFO Interrupt Mode 0: Interrupt generated when RX FIFO counter reaches RFIGCV value from values smaller than RFIGCV 1: Interrupt generated at the end of every received message storage
11	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
10 to 8	RFDC[2:0]	000	R/W	RX FIFO Depth Configuration 000: FIFO Depth = 0 messages 001: FIFO Depth = 4 messages 010: FIFO Depth = 8 messages 011: FIFO Depth = 16 messages 100: FIFO Depth = 32 messages 101: FIFO Depth = 48 messages 110: FIFO Depth = 64 messages 111: FIFO Depth = 128 messages
7	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.

Bit	Bit Name	Initial Value	R/W	Description
6 to 4	RFPLS[2:0]	000	R/W	Rx FIFO Payload Data Size Configuration 000: 8 bytes 001: 12 bytes 010: 16 bytes 011: 20 bytes 100: 24 bytes 101: 32 bytes 11 0: 48 bytes 111: 64 bytes
3, 2	—	00	R/W	Reserved These bits are read as 0. The write value should be always 0.
1	RFIE	0	R/W	RX FIFO Interrupt Enable 0: FIFO interrupt generation disabled 1: FIFO interrupt generation enabled
0	RFE	0	R/W	RX FIFO Enable 0: FIFO disabled 1: FIFO enabled

RFE bit (RX FIFO Enable)

The RFE bit enables the FIFO. When this bit is set to 0, the RX FIFO is cleared to empty.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode.

This bit can only be set if the configured FIFO depth is greater than 000b (CFDRFCCn.RFDC[2:0] > 000b).

Set the RFE bit with a separate write access to the CFDRFCCn register, after all the other bits in the CFDRFCCn register are set.

This bit is cleared automatically when the CAN-FD module is in GL_RESET mode.

RFIE bit (RX FIFO Interrupt Enable)

The RFIE bit enables generation of the FIFO interrupt.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

RFPLS[2:0] bits (Rx FIFO Payload Data Size Configuration)

The RFPLS[2:0] bits define the message data payload allocation in the RAM.

This is the maximum number of bytes which can be received by this FIFO.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

RFDC[2:0] bits (RX FIFO Depth Configuration)

The RFDC[2:0] bits select the depth of the FIFO in terms of the number of messages. If the FIFO depth is configured to 0 messages, the FIFO cannot be used.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

RFIM bit (RX FIFO Interrupt Mode)

The RFIM bit selects the interrupt generation condition for the FIFO.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

Only write to this bit when the CAN-FD module is in GL_RESET mode.

RFIGCV[2:0] bits (RX FIFO Interrupt Generation Counter Value)

The RFIGCV[2:0] bits select the counter value of the FIFO for generation of FIFO interrupts. These values represent fractions of the FIFO depth for which an interrupt is generated.

Do not write to these bits when the CAN-FD module is in GL_SLEEP mode.

The setting of the RFIGCV[2:0] bits should be synchronized with the RFDC[2:0] bits.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

RFFIE bit (RX FIFO Full Interrupt Enable)

The RFFIE bit enables generation of the RX FIFO full interrupt. Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

The following content shows examples of interruptions:

- Interruption output in number of arbitrary stages (CFDRFCCn.RFIGCV)
- Interruption output in FIFO full state.

Note: Management of the receiving data of FIFO can be performed by these notices of interruption.

30.2.26 RX FIFO Status Register n (CFDRFSTSn) (n = 0 to 7)

The CFDRFSTSn register (n = 0 to 7) shows the status of messages stored in the corresponding FIFO buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'00E0 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RFFIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFMC[7:0]								—	—	—	—	RFIF	RFMLT	RFFLL	RFEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.
28	—	0	R	Reserved This bit is read as 0.
27 to 17	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
16	RFFIF	0	R/W	RX FIFO Full Interrupt Flag 0: FIFO full interrupt condition not satisfied 1: FIFO full interrupt condition satisfied
15 to 8	RFMC[7:0]	H'00	R	RX FIFO Message Count Number of messages stored in FIFO
7 to 4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
3	RFIF	0	R/W	RX FIFO Interrupt Flag 0: FIFO interrupt condition not satisfied 1: FIFO interrupt condition satisfied
2	RFMLT	0	R/W	RX FIFO Message Lost 0: No message lost in FIFO 1: FIFO message lost
1	RFFLL	0	R	RX FIFO Full 0: FIFO not full 1: FIFO full
0	RFEMP	1	R	RX FIFO Empty 0: FIFO not empty 1: FIFO empty

RFEMP bit (RX FIFO Empty)

The RFEMP bit is set automatically when:

- The RFMC bit is 0
- RX FIFO is disabled by setting the CFDRFCCn.RFE bit to 0
- The CAN-FD module is in GL_RESET mode.

The RFEMP bit is cleared automatically when the first message is stored in the RX FIFO buffer.

RFFLL bit (RX FIFO Full)

The RFFLL bit is set automatically when the number of CAN messages stored in the FIFO buffer matches the configured FIFO depth.

The RFFLL is cleared automatically when:

- The number of CAN messages stored in the FIFO buffer is less than the configured FIFO depth
- RX FIFO is disabled by setting the CFDRFCCn.RFE bit to 0
- The CAN-FD module is in GL_RESET mode.

RFMLT bit (RX FIFO Message Lost)

Only write to the RFMLT bit when CAN-FD module is in GL_HALT or GL_OPERATION mode. Writing 1 has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

This bit is set automatically whenever a message is lost due to attempted storage when the FIFO buffer is already full. If a set from the CAN channel occurs simultaneously with a clear by a write access, then the bit is set.

The bit is cleared:

- By writing 0 to it
- When the CAN-FD module is in GL_RESET mode.

RFIF bit (RX FIFO Interrupt Flag)

The RFIF bit is set automatically when the configured interrupt condition is satisfied. This bit is not automatically cleared when the RX FIFO buffer is disabled.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode. Writing 1 has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

If a set from the CAN channel occurs simultaneously with a clear by a write access, then this bit is set.

The bit is cleared by writing 0 to it. The bit is also cleared when CAN-FD module is in GL_RESET mode.

RFMC[7:0] bits (RX FIFO Message Count)

The RFMC[7:0] bits indicate the number of CAN messages stored in the RX FIFO buffer that can be read by the CPU.

These bits are cleared automatically when the FIFO is disabled and when the CAN-FD module is in GL_RESET mode.

RFFIF bit (RX FIFO Full Interrupt Flag)

The RFFIF bit is not cleared automatically when the RX FIFO buffer is disabled.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode. Writing 1 has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

This bit is set automatically when the FIFO full interrupt condition is satisfied. If a set from the CAN channel occurs simultaneously with a clear by a write access, then the bit is set.

The bit is cleared by writing 0 to it.

The bit is also cleared when the CAN-FD module is in GL_RESET mode.

30.2.27 RX FIFO Pointer Control Register n (CFDRFPCTRn) (n = 0 to 7)

The CFDRFPCTRn register (n = 0 to 7) can be used to increment the read pointer of the corresponding RX FIFO buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0100 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RFPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
7 to 0	RFPC[7:0]	H'00	W	RX FIFO Pointer Control Increments read pointer of the corresponding RX FIFO buffers

RFPC[7:0] bits (RX FIFO Pointer Control)

When the value H'FF is written to the RFPC bits, the pointer of the corresponding RX FIFO buffer is moved to the next FIFO entry. Only write H'FF to these registers when the corresponding RX FIFO buffer is enabled and not empty.

The read value from these bits is always H'00.

Only write to these bits when the CAN-FD module is in GL_HALT or GL_OPERATION mode.

Do not write to the RX FIFO Pointer Control registers when DMA is enabled.

30.2.28 Common FIFO Configuration/Control Register n (CFDCFCCn) (n = 0 to 5)

The CFDCFCCn register (n = 0 to 5) is used to configure the Common FIFOs.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0120 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFITT[7:0]								CFDC[2:0]			CFTML[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFIGCV[2:0]			CFIM	CFITR	CFITSS	CFM[1:0]		—	CFPLS[2:0]			—	CFTXIE	CFRXIE	CFE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CFITT[7:0]	H'00	R/W	Common FIFO Interval Transmission Time Delay the start of transmission from the FIFO if configured in TX or GW mode, delay is a multiple of basic Interval Timer Clock Source unit
23 to 21	CFDC[2:0]	000	R/W	Common FIFO Depth Configuration 000: FIFO Depth = 0 messages 001: FIFO Depth = 4 messages 010: FIFO Depth = 8 messages 011: FIFO Depth = 16 messages 100: FIFO Depth = 32 messages 101: FIFO Depth = 48 messages 110: FIFO Depth = 64 messages 111: FIFO Depth = 128 messages
20 to 16	CFTML[4:0]	All 0	R/W	Common FIFO TX Message Buffer Link Transmission scan link position of the corresponding channel
15 to 13	CFIGCV[2:0]	000	R/W	Common FIFO Interrupt Generation Counter Value 000: Interrupt generated when FIFO is 1/8th full 001: Interrupt generated when FIFO is 1/4th full 010: Interrupt generated when FIFO is 3/8th full 011: Interrupt generated when FIFO is 1/2 full 100: Interrupt generated when FIFO is 5/8th full 101: Interrupt generated when FIFO is 3/4th full 110: Interrupt generated when FIFO is 7/8th full 111: Interrupt generated when FIFO is full

Bit	Bit Name	Initial Value	R/W	Description
12	CFIM	0	R/W	Common FIFO Interrupt Mode 0: RX FIFO mode: RX interrupt generated when Common FIFO counter reaches CFIGCV value from a lower value TX FIFO mode: TX interrupt generated when Common FIFO transmits the last message successfully GW FIFO mode: For RX interrupt flag: Interrupt generated when FIFO counter increments and reaches the value configured in CFIGCV For TX interrupt flag: Interrupt generated when FIFO transmits the last message successfully 1: RX FIFO mode: RX interrupt generated at the end of every received message storage TX FIFO mode: TX interrupt generated for every successfully transmitted message GW FIFO mode: For RX interrupt flag: Interrupt generated when a message is stored in the FIFO For TX interrupt flag: Interrupt generated when a message is successfully transmitted from the FIFO
11	CFITR	0	R/W	Common FIFO Interval Timer Resolution 0: Reference clock period × 1 1: Reference clock period × 10
10	CFITSS	0	R/W	Common FIFO Interval Timer Source Select 0: Reference clock (× 1 / × 10 period) 1: Bit time clock of related channel (FIFO is linked to fixed channel)
9, 8	CFM[1:0]	00	R/W	Common FIFO Mode 00: RX FIFO mode 01: TX FIFO mode 10: CAN – CAN GW FIFO mode 11: Reserved
7	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
6 to 4	CFPLS[2:0]	000	R/W	Common FIFO Payload Data Size Configuration 000: 8 bytes 001: 12 bytes 010: 16 bytes 011: 20 bytes 100: 24 bytes 101: 32 bytes 110: 48 bytes 111: 64 bytes
3	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
2	CFTXIE	0	R/W	Common FIFO TX Interrupt Enable 0: FIFO interrupt generation disabled for Frame TX 1: FIFO interrupt generation enabled for Frame TX
1	CFRXIE	0	R/W	Common FIFO RX Interrupt Enable 0: FIFO interrupt generation disabled for Frame RX 1: FIFO interrupt generation enabled for Frame RX
0	CFE	0	R/W	Common FIFO Enable 0: FIFO disabled 1: FIFO enabled

CFE bit (Common FIFO Enable)

The CFE bit enables the FIFO when set. FIFO is disabled when this bit is cleared.

This bit can also be used, by clearing it, to abort transmission from Common FIFO when configured in TX mode or GW mode, or to stop reception into the Common FIFO in RX mode.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for FIFOs configured as TX or GW FIFO.

This bit can only be set if the configured FIFO depth is greater than 0 (CFDC bit > 0).

Set the CFE bit with a separate write access to the CFDCFCCn register, after all the other bits in this register are set.

This bit is cleared automatically when the CAN-FD module is in GL_RESET mode.

This bit is also cleared automatically when the related channel is in CH_RESET mode if the FIFO is configured in TX or GW mode.

CFRXIE bit (Common FIFO RX Interrupt Enable)

The CFRXIE bit enables generation of FIFO interrupts when the interrupt flag is set after reception of a frame in the corresponding FIFO buffer.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

CFTXIE bit (Common FIFO TX Interrupt Enable)

The CFTXIE bit enables generation of common FIFO interrupts when the interrupt flag is set after transmission of a frame from the corresponding FIFO buffer.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

CFPLS[2:0] bits (Common FIFO Payload Data Size Configuration)

The CFPLS[2:0] bits define the message data payload allocation in the RAM. This is the maximum number of bytes which can be received or transmitted by the FIFO buffer.

For details, see **Section 30.6, FIFO Buffers and Normal Message Buffer Configuration**.

Only write to this bit when the CAN-FD module is in GL_RESET mode.

CFM[1:0] bits (Common FIFO Mode)

The CFM[1:0] bits select the mode of the FIFO. When a hardware reset is applied, all the Common FIFO buffers are configured in RX FIFO mode. Do not configure these bits to 11b.

Do not write to these bits in GL_OPERATION or GL_SLEEP mode.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

CFITSS bit (Common FIFO Interval Timer Source Select)

The CFITSS bit selects the basic clock source for the Interval Transmission Timer.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode. In addition, do not write to this bit when the CFE bit is set to 1.

Do not write 1 to this bit when CAN-FD communication is used.*[1]

Note: The bit time clock can vary depending on the nominal and data rate bit configuration.

CFITR bit (Common FIFO Interval Timer Resolution)

The CFITR bit selects the resolution of the reference clock for the Interval Transmission Timer (peripheral clock is the source for the reference clock).

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode. Also, do not write to this bit when the CFE bit is set to 1.

CFIM bit (Common FIFO Interrupt Mode)

The CFIM bit selects the interrupt generation condition for the FIFO buffer.

Do not write to this bit in GL_SLEEP mode.

Only write to this bit when the CAN-FD module is in GL_RESET mode.

CFIGCV[2:0] bits (Common FIFO Interrupt Generation Counter Value)

The CFIGCV[2:0] bits select the message counter value for the generation of FIFO interrupts. These values represent fractions of the FIFO depth at which the interrupt is to be generated.

Do not write to these bits when the CAN-FD module is in GL_SLEEP mode.

The setting of these bits should be synchronized with the CFDC[2:0] bits.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

CFTML[4:0] bits (Common FIFO TX Message Buffer Link)

The CFTML[4:0] bits select the normal transmit message buffer position where the TX or GW FIFO is linked to, for transmission scanning.

Do not write to these bits in GL_OPERATION or GL_SLEEP mode.

Only write to this bit when the CAN-FD module is in GL_RESET mode.

CFDC[2:0] bits (Common FIFO Depth Configuration)

The CFDC[2:0] bits select the depth of the common FIFO in terms of the number of messages. If the FIFO depth is configured to 0 message, the FIFO cannot be used.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

CFITT[7:0] bits (Common FIFO Interval Transmission Time)

The CFITT[7:0] bits select the delay in the start of transmission for all messages transmitted from this FIFO buffer when configured in TX or GW mode. The delay is a multiple of the basic interval timer clock source period (reference clock × 1, reference clock × 10, or bit time clock of the related CAN channel).

Do not write to these bits when the CAN-FD module is in GL_SLEEP mode.

Do not write to these bits when the CFE bit is set to 1.

When CFDGCFG.ITRCP[15:0] = H’0000, set the CFITT[7:0] bits to H’00.

30.2.29 Common FIFO Configuration/Control Enhancement Register n (CFDCFCCEn) (n = 0 to 5)

The CFDCFCCEn register (n = 0 to 5) is used to configure the Common FIFOs.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0180 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CFBME
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CFMOWM	—	—	—	—	—	CFOFTXIE	CFOFRXIE	CFFIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
16	CFBME	0	R/W	Common FIFO Buffering Mode Enable 0: Transmission from Common FIFO 1: Transmission halt from Common FIFO
15 to 9	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
8	CFMOWM	0	R/W	Common FIFO Message Overwrite Mode 0: Message discarded mode 1: Message overwrite mode
7 to 3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
2	CFOFTXIE	0	R/W	Common FIFO One Frame Transmission Interrupt Enable 0: One Frame TX interrupt generation disabled 1: One Frame TX interrupt generation enabled
1	CFOFRXIE	0	R/W	Common FIFO One Frame Reception Interrupt Enable 0: One Frame RX interrupt generation disabled 1: One Frame RX interrupt generation enabled
0	CFFIE	0	R/W	Common FIFO Full Interrupt Enable 0: FIFO Interrupt generation disabled 1: FIFO Interrupt generation enabled

CFFIE bit (Common FIFO Full Interrupt Enable)

The CFFIE bit enables generation of the FIFO full interrupt when the interrupt flag is set after reception of a frame in the corresponding FIFO buffer.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

The following content shows examples of interruptions:

1. Interruption output in number of arbitrary stages (CFDCFCCn.CFIGCV).
2. Interruption output in FIFO full state.

Management of the receiving data of FIFO can be performed by these notices of interruption.

CFOFRXIE bit (Common FIFO One Frame Reception Interrupt Enable)

The CFOFRXIE bit enables generation of the one frame reception interrupt when the interrupt flag is set after reception of a frame in the corresponding FIFO buffer.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

CFOFTXIE bit (Common FIFO One Frame Transmission Interrupt Enable)

The CFOFTXIE bit enables generation of the one frame transmission interrupt when the interrupt flag is set after transmission of a frame in the corresponding FIFO buffer.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

CFMOWM bit (Common FIFO Message Overwrite Mode)

When the CFMOWM bit is 0, a receiving message is discarded and FIFO is full. When the CFMOWM bit is 1, a receiving message is overwritten and FIFO is full.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

Only write 1 to this bit when the common FIFO is in GW mode.

Do not write 1 to this bit when the CFE bit is 1.

CFBME bit (Common FIFO Buffering Mode Enable)

When the CFBME bit is 0, messages are transmitted from FIFO. When the CFBME bit is 1, messages are not transmitted from FIFO.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode. Additionally, do not write 1 to this bit when the CFE bit is 1.

30.2.30 Common FIFO Status Register n (CFDCFSTSn) (n = 0 to 5)

The CFDCFSTSn register (n = 0 to 5) shows the status of messages stored in the corresponding FIFO buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'01E0 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	CFMOW	—	—	—	—	—	CFOFTXIF	CFOFRXIF	CFFIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFMC[7:0]								—	—	—	CFTXIF	CFRXIF	CFMLT	CFFLL	CFEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.
28	—	0	R	Reserved This bit is read as 0.
27 to 25	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.
24	CFMOW	0	R/W	Common FIFO Message Overwrite 0: No message overwrite occurred in FIFO 1: Message overwrite occurred in FIFO
23 to 19	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
18	CFOFTXIF	0	R/W	Common FIFO One Frame Transmission Interrupt Flag For each FIFO that transmits a frame, a corresponding interrupt is set.
17	CFOFRXIF	0	R/W	Common FIFO One Frame Reception Interrupt Flag For each FIFO that receives a frame, a corresponding interrupt is set.
16	CFFIF	0	R/W	Common FIFO Full Interrupt Flag 0: Interrupt condition not satisfied for FIFO full interrupt 1: Interrupt condition satisfied for FIFO full interrupt
15 to 8	CFMC[7:0]	H'00	R	Common FIFO Message Count Number of messages stored in FIFO
7 to 5	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.
4	CFTXIF	0	R/W	Common TX FIFO Interrupt Flag 0: FIFO interrupt condition not satisfied after frame transmission 1: FIFO Interrupt condition satisfied after frame transmission
3	CFRXIF	0	R/W	Common RX FIFO Interrupt Flag 0: FIFO interrupt condition not satisfied after frame reception 1: FIFO interrupt condition satisfied after frame reception
2	CFMLT	0	R/W	Common FIFO Message Lost 0: FIFO message not lost 1: FIFO message lost

Bit	Bit Name	Initial Value	R/W	Description
1	CFFLL	0	R	Common FIFO Full 0: FIFO not full 1: FIFO full
0	CFEMP	1	R	Common FIFO Empty 0: FIFO not empty 1: FIFO empty

CFEMP bit (Common FIFO Empty)

The CFEMP bit is set automatically when:

- The CPU has read all messages from the FIFO configured in RX mode
- All messages have been transmitted from the FIFO configured in TX or GW mode
- The FIFO is disabled by setting the CFE bit to 0
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET when FIFO configured in TX or GW mode.

The CFEMP bit is cleared automatically when:

- The first reception message is stored in the FIFO buffer when configured in RX mode
- The first message to be transmitted is stored in the FIFO buffer when configured in TX or GW mode.

CFFLL bit (Common FIFO Full)

The CFFLL bit is set automatically when the number of CAN messages stored in the FIFO matches the configured FIFO depth.

The CFFLL bit is cleared automatically when:

- The number of CAN messages stored in the FIFO is less than the configured FIFO depth
- The FIFO is disabled by setting the CFE bit to 0
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode when FIFO buffer is configured in TX or GW mode.

CFMLT bit (Common FIFO Message Lost)

The CFMLT bit is set automatically whenever a message is lost due to attempted storage of a new message when FIFO is already full in RX or GW mode.

If a set from the CAN channel occurs simultaneously with a clear by a write access, then this bit is set.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for FIFO configured as TX or GW FIFO. Writing 1 has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

The CFMLT bit is cleared:

- By writing 0 to it
- When the CAN-FD module is in GL_RESET mode
- When the related CAN-FD channel is in CH_RESET mode if the FIFO buffer is configured in TX or GW mode.

CFRXIF bit (Common RX FIFO Interrupt Flag)

The CFRXIF bit is not cleared automatically if the Common FIFO buffer is disabled.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for FIFO configured as TX or GW FIFO. Writing 1 has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

This bit is set automatically when the configured interrupt condition is satisfied for Common FIFO buffers when configured in GW mode or RX mode.

The CFRXIF bit is cleared:

- By writing 0 to it
- When the CAN-FD module is in GL_RESET mode
- When the related CAN-FD channel is in CH_RESET mode if the FIFO buffer is configured in GW mode.

CFTXIF bit (Common TX FIFO Interrupt Flag)

The CFTXIF bit is not cleared automatically if the Common FIFO buffer is disabled.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for FIFO buffer configured as TX or GW FIFO. Writing 1 has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

This bit is set automatically when the configured interrupt condition is satisfied for Common FIFO buffers configured in GW or TX mode.

The CFTXIF bit is cleared:

- By writing 0 to it
- When the CAN-FD module is in GL_RESET mode
- When the related CAN-FD channel is in CH_RESET mode if the FIFO buffer is configured in TX or GW mode.

CFMC[7:0] bits (Common FIFO Message Count)

The CFMC[7:0] bits indicate the following:

- Number of CAN messages stored by the CPU in the FIFO buffer configured in TX mode pending for transmission
- Number of CAN messages stored in the FIFO buffer configured in RX mode by CAN-FD to be read by the CPU
- Number of CAN messages stored by the CAN-FD in the GW FIFO pending for transmission.

The CFMC[7:0] bits are cleared automatically when:

- The FIFO is disabled
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode if the FIFO buffer is configured in TX or GW mode.

CFFIF bit (Common FIFO Full Interrupt Flag)

The CFFIF bit is not cleared automatically if the Common FIFO buffer is disabled.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for FIFO configured as TX or GW FIFO. Writing 1 has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

This bit is set automatically when the FIFO Full Interrupt condition is satisfied for Common FIFO buffers when configured in GW or RX mode.

If a set from the CAN channel occurs simultaneously with a clear by a write access, then the bit is set.

The CFFIF bit is cleared:

- By writing 0 to it
- When the CAN-FD module is in GL_RESET mode
- When the related CAN-FD channel is in CH_RESET mode if the FIFO buffer is configured in TX or GW mode.

CFOFRXIF bit (Common FIFO One Frame Reception Interrupt Flag)

The CFOFRXIF bit is not cleared automatically if the Common FIFO buffer is disabled.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for the FIFO buffer configured as TX or GW FIFO. Writing 1 has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

This bit is set automatically when the One Frame Reception Interrupt condition is satisfied for the Common FIFO buffers when configured in GW or RX mode.

If a set from the CAN channel occurs simultaneously with a clear by a write access, then the bit is set.

The CFOFRXIF bit is cleared:

- By writing 0 to it
- When the CAN-FD module is in GL_RESET mode
- When the related CAN-FD channel is in CH_RESET mode if the FIFO buffer is configured in TX or GW mode.
- This bit is not influenced by the value of **CFDCFCCn.CFIM**.

CFOFTXIF bit (Common FIFO One Frame Transmission Interrupt Flag)

The CFOFTXIF bit is not cleared automatically if the Common FIFO buffer is disabled.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for FIFOs configured as TX or GW FIFO. Writing 1 has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

This bit is set automatically when the One Frame Transmission Interrupt condition is satisfied for Common FIFO buffers configured in GW mode or TX mode.

If a set from the CAN channel occurs simultaneously with a clear by a write access, then the bit is set.

The bit is cleared:

- By writing 0 to it
- When the CAN-FD module is in GL_RESET mode
- When the related CAN-FD channel is in CH_RESET mode if the FIFO is configured in TX or GW mode.
- This bit is not influenced by the value of **CFDCFCCn.CFIM**

CFMOW bit (Common FIFO Message Overwrite)

The CFMOW bit is set automatically whenever a message is an overwrite storage of a new message when CFDCFCCEn.CFMOWM = 1 and FIFO is already full in GW mode.

If the set from the CAN channel occurs simultaneously with the clear by a write access, the bit is set.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for FIFO buffer configured as TX or GW FIFO. Writing 1 has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

The CFMOW bit is cleared:

- By writing 0 to it
- When the CAN-FD module is in GL_RESET mode
- When the related CAN-FD channel is in CH_RESET mode if the FIFO buffer is configured in TX or GW mode.

30.2.31 Common FIFO Pointer Control Register n (CFDCFPCTRn) (n = 0 to 5)

The CFDCFPCTRn register (n = 0 to 5) can be used to increment the read or write pointer of the corresponding Common FIFO buffer.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0240 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CFPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
7 to 0	CFPC[7:0]	H'00	W	Common FIFO Pointer Control Increments read or write pointer of the corresponding Common FIFO buffers depending on the mode configuration.

CFPC[7:0] bits (Common FIFO Pointer Control)

When the value H'FF is written into the CFPC[7:0] bits, the read pointer of the corresponding Common FIFO buffer (when configured in RX mode), or the write pointer of the corresponding Common FIFO buffer (when configured in TX mode) moves to the next FIFO entry.

The read value from these bits is always H'00.

Only write to these bits when the CAN-FD module is in GL_HALT or GL_OPERATION mode.

Only write H'FF to this register when:

- The Common FIFO buffer is enabled and is not empty if configured in RX mode
- The Common FIFO buffer is enabled and is not full if configured in TX mode
- The Common FIFO buffer is enabled and is not configured in GW mode.

Do not write to the Common FIFO Pointer Control registers when DMA is enabled.

30.2.32 FIFO Empty Status Register (CFDFESTS)

The CFDFESTS register shows status of the empty bits of the FIFO buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'02A0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CFXEMP[5:0]						RFXEMP[7:0]							
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 1	R	Reserved These bits are read as 1.
13 to 8	CFXEMP[5:0]	All 1	R	Common FIFO Empty Status 0: Corresponding FIFO not empty 1: Corresponding FIFO empty
7 to 0	RFXEMP[7:0]	All 1	R	RX FIFO Empty Status 0: Corresponding FIFO not empty 1: Corresponding FIFO empty

RFXEMP[7:0] bits (RX FIFO Empty Status)

Bit [7] (RFXEMP[7]) is associated with FIFO index 7 and bit [0] (RFXEMP[0]) is associated with FIFO index 0.

The RFXEMP[7:0] bits are set when the CAN-FD module is in GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the RX FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the RX FIFO Status Registers.

CFXEMP[5:0] bits (Common FIFO Empty Status)

Bit [13] (CFXEMP[5]) is associated with common FIFO index 5 and bit [8] (CFXEMP[0]) is associated with common FIFO index 0.

The CFXEMP[5:0] bits are set when the CAN-FD module is in GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

30.2.33 FIFO Full Status Register (CFDFFSTS)

The CFDFFSTS register shows status of the full bits of the FIFO buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'02A4

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CFXFLL[5:0]						RFXFLL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31:14	—	All 0	R	Reserved These bits are read as 0.
13:8	CFXFLL[5:0]	All 0	R	Common FIFO Full Status 0: Corresponding FIFO not full 1: Corresponding FIFO full
7:0	RFXFLL[7:0]	All 0	R	RX FIFO Full Status 0: Corresponding FIFO not full 1: Corresponding FIFO full

RFXFLL[7:0] bits (RX FIFO Full Status)

Bit [7] (RFXFLL[7]) is associated with FIFO index 7 and bit [0] (RFXFLL[0]) is associated with FIFO index 0.

The RFXFLL[7:0] bits are cleared when CAN-FD module is in GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the RX FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the RX FIFO Status Registers.

CFXFLL[5:0] bits (Common FIFO Full Status)

Bit [13] (CFXFLL[5]) is associated with common FIFO index 5 and bit [8] (CFXFLL[0]) is associated with common FIFO index 0.

The CFXFLL[5:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

30.2.34 FIFO Message Lost Status Register (CFDFMSTS)

The CFDFMSTS register shows status of the message lost bits of the FIFO buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'02A8

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CFXMLT[5:0]						RFXMLT[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are read as 0.
13 to 8	CFXMLT[5:0]	All 0	R	Common FIFO Message Lost Status 0: Corresponding FIFO Message Lost flag not set 1: Corresponding FIFO Message Lost flag set
7 to 0	RFXMLT[7:0]	All 0	R	RX FIFO Message Lost Status 0: Corresponding FIFO Message Lost flag not set 1: Corresponding FIFO Message Lost flag set

RFXMLT[7:0] bits (RX FIFO Message Lost Status)

Bit [7] (RFXML[7]) is associated with FIFO index 7 and bit [0] (RFXML[0]) is associated with FIFO index 0.

The RFXMLT[7:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the RX FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the RX FIFO Status Registers.

CFXMLT[5:0] bits (Common FIFO Message Lost Status)

Bit [13] (CFXMLT[5]) is associated with common FIFO index 5 and bit [8] (CFXMLT[0]) is associated with common FIFO index 0.

The CFXMLT[5:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

30.2.35 RX FIFO Interrupt Flag Status Register (CFDRFISTS)

The CFDRFISTS register shows status of the interrupt flag bits of the RX FIFO buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'02AC

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	RFXFFLL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RFXIF[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
23 to 16	RFXFFLL[7:0]	All 0	R	RX FIFO[x] Interrupt Full Flag Status 0: Corresponding RX FIFO Interrupt Full flag not set 1: Corresponding RX FIFO Interrupt Full flag set
15 to 8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
7 to 0	RFXIF[7:0]	All 0	R	RX FIFO[x] Interrupt Flag Status 0: Corresponding RX FIFO Interrupt flag not set 1: Corresponding RX FIFO Interrupt flag set

RFXIF[7:0] bits (RX FIFO Interrupt Flag Status)

Bit [7] (RFXIF[7]) is associated with FIFO index 7 and bit [0] (RFXIF[0]) is associated with FIFO index 0.

Each bit is set automatically when the corresponding interrupt flag bit is set in the RX FIFO Status Registers.

The RFXIF[7:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is cleared automatically when the corresponding interrupt flag bit is cleared in the RX FIFO Status Registers.

RFXFFLL[7:0] bits (RX FIFO Interrupt Full Flag Status)

Bit [23] (RFXFFLL[7]) is associated with FIFO index 7 and bit [16] (RFXFFLL[0]) is associated with FIFO index 0.

Each bit is set automatically when the corresponding interrupt full flag bit is set in the RX FIFO Status Registers.

The RFXFFLL[7:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is cleared automatically when the corresponding interrupt full flag bit is cleared in the RX FIFO Status Registers.

30.2.36 Common FIFO RX Interrupt Flag Status Register (CFDCFRISTS)

The CFDCFRISTS register shows status of the interrupt flag bits of the Common FIFO buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'02B0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CFXRXIF[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
5 to 0	CFXRXIF[5:0]	All 0	R	Common FIFO RX Interrupt Flag Status 0: Corresponding Common FIFO RX Interrupt flag not set 1: Corresponding Common FIFO RX Interrupt flag set

CFXRXIF[5:0] bits (Common FIFO RX Interrupt Flag Status)

Bit [5] (CFXRXIF[5]) is associated with FIFO index 5 and bit [0] (CFXRXIF[0]) is associated with FIFO index 0.

Each bit is set automatically when the corresponding RX interrupt flag bit is set in the Common FIFO Status Registers.

The CFXRXIF[5:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is cleared automatically when the corresponding RX interrupt flag bit is cleared in the Common FIFO Status Registers.

30.2.37 Common FIFO TX Interrupt Flag Status Register (CFDCFTISTS)

The CFDCFTISTS register shows status of the interrupt flag bits of the Common FIFO buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'02B4

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CFXTXIF[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
5 to 0	CFXTXIF[5:0]	All 0	R	Common FIFO TX Interrupt Flag Status 0: Corresponding Common FIFO TX Interrupt flag not set 1: Corresponding Common FIFO TX Interrupt flag set

CFXTXIF[5:0] bits (Common FIFO TX Interrupt Flag Status)

Bit [5] (CFXTXIF[5]) is associated with FIFO index 5 and bit [0] (CFXTXIF[0]) is associated with FIFO index 0.

Each bit is set automatically when the corresponding TX interrupt flag bit is set in the Common FIFO Status Registers.

The CFXTXIF[5:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is cleared automatically when the corresponding TX interrupt flag bit is cleared in the Common FIFO Status Registers.

30.2.38 FIFO FDC Full Status Register (CFDFFFSTS)

The CFDFFFSTS register shows status of the full interrupt flag bits of the FIFO buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'02C4

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CFXFFLL[5:0]						RFXFFLL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are read as 0.
13 to 8	CFXFFLL[5:0]	All 0	R	COMMON FIFO FDC Level Full Status 0: Corresponding FIFO full interrupt not set 1: Corresponding FIFO full interrupt set
7 to 0	RFXFFLL[7:0]	All 0	R	RX FIFO FDC Level Full Status 0: Corresponding FIFO full interrupt not set 1: Corresponding FIFO full interrupt set

RFXFFLL[7:0] bits (RX FIFO FDC Level Full Status)

Bit [7] (RFXFFLL[7]) is associated with FIFO index 7 and bit [0] (RFXFFLL[0]) is associated with FIFO index 0.

Each bit is set automatically when the corresponding bit is set in the RX FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the RX FIFO Status Registers.

The RFXFFLL[7:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

CFXFFLL[5:0] bits (COMMON FIFO FDC Level Full Status)

Bit [13] (CFXFFLL[5]) is associated with common FIFO index 5 and bit [8] (CFXFFLL[0]) is associated with common FIFO index 0.

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

The CFXFFLL[5:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

30.2.39 Common FIFO Message Overwrite Status Register (CFDCFMOWSTS)

The CFDCFMOWSTS register shows the status of the Interrupt Flag bits of the Common FIFO Buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'02C0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CFXMOW[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
5 to 0	CFXMOW [5:0]	All 0	R	Common FIFO Massage Overwrite Status 0: Corresponding FIFO Overwrite flag not set 1: Corresponding FIFO Overwrite flag set

CFXMOW[5:0] bits (Common FIFO Massage Overwrite Status)

Bit [5] (CFXMOW[5]) is associated with common FIFO index 5 and bit [0] (CFXFMOW[0]) is associated with common FIFO index 0.

The CFXMOW[5:0] bits are cleared when the CAN-FD module is in GL_RESET mode. This register is only valid in GW mode.

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

30.2.40 Common FIFO One Frame RX Interrupt Flag Status Register (CFDCFOFRISTS)

The CFDCFOFRISTS register shows status of the interrupt flag bits of the Common FIFO buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'02B8

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CFXOFRXIF[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are read as 0.
5 to 0	CFXOFRXIF [5:0]	All 0	R	Common FIFO One Frame RX Interrupt Flag Status 0: Corresponding Common FIFO One Frame RX Interrupt flag not set 1: Corresponding Common FIFO One Frame RX Interrupt flag set

CFXOFRXIF[5:0] bits (Common FIFO One Frame RX Interrupt Flag Status)

Bit [5] (CFXFRXIF[5]) is associated with common FIFO index 5 and bit [0] (CFXFRXIF[0]) is associated with common FIFO index 0.

Each bit is set automatically when the corresponding One Frame RX Interrupt flag bit is set in the Common FIFO Status Registers.

The CFXOFRXIF[5:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is cleared automatically when the corresponding One Frame RX Interrupt flag bit is cleared in the Common FIFO Status Registers.

30.2.41 Common FIFO One Frame TX Interrupt Flag Status Register (CFDCFOFTISTS)

The CFDCFOFTISTS register shows status of the interrupt flag bits of the Common FIFO buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'02BC

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CFXOFTXIF[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are read as 0.
5 to 0	CFXOFTXIF [5:0]	All 0	R	Common FIFO One Frame TX Interrupt Flag Status 0: Corresponding Common FIFO One Frame TX Interrupt flag not set 1: Corresponding Common FIFO One Frame TX Interrupt flag set

CFXOFTXIF[5:0] bits (Common FIFO One Frame TX Interrupt Flag Status)

Bit [5] (CFXOFTXIF [5]) is associated with common FIFO index 5 and bit [0] (CFXOFTXIF[0]) is associated with common FIFO index 0.

Each bit is set automatically when the corresponding One Frame TX Interrupt flag bit is set in the Common FIFO Status Registers.

The CFXOFTXIF[5:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is cleared automatically when the corresponding One Frame TX Interrupt flag bit is cleared in the Common FIFO Status Registers.

30.2.42 DMA Transfer Control Register (CFDCDTCT)

The CFDCDTCT register controls the start and stop of DMA transfer operation.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1330

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CFDMAE1	CFDMAE0	RFDMAE7	RFDMAE6	RFDMAE5	RFDMAE4	RFDMAE3	RFDMAE2	RFDMAE1	RFDMAE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
9	CFDMAE1	0	R/W	DMA Transfer Enable for Common FIFO 0 of Channel 1 0: DMA transfer request disabled 1: DMA transfer request enabled
8	CFDMAE0	0	R/W	DMA Transfer Enable for Common FIFO 0 of Channel 0 0: DMA transfer request disabled 1: DMA transfer request enabled
7	RFDMAE7	0	R/W	DMA Transfer Enable for RX FIFO 7 0: DMA transfer request disabled 1: DMA transfer request enabled
6	RFDMAE6	0	R/W	DMA Transfer Enable for RX FIFO 6 0: DMA transfer request disabled 1: DMA transfer request enabled
5	RFDMAE5	0	R/W	DMA Transfer Enable for RX FIFO 5 0: DMA transfer request disabled 1: DMA transfer request enabled
4	RFDMAE4	0	R/W	DMA Transfer Enable for RX FIFO 4 0: DMA transfer request disabled 1: DMA transfer request enabled
3	RFDMAE3	0	R/W	DMA Transfer Enable for RX FIFO 3 0: DMA transfer request disabled 1: DMA transfer request enabled
2	RFDMAE2	0	R/W	DMA Transfer Enable for RX FIFO 2 0: DMA transfer request disabled 1: DMA transfer request enabled
1	RFDMAE1	0	R/W	DMA Transfer Enable for RX FIFO 1 0: DMA transfer request disabled 1: DMA transfer request enabled
0	RFDMAE0	0	R/W	DMA Transfer Enable for RX FIFO 0 0: DMA transfer request disabled 1: DMA transfer request enabled

RFDMAEn (n = 0 to 7) bit (DMA Transfer Enable for RX FIFOn)

Number of RX FIFOs = 8

The RFDMAEn bit cannot be set in GL_SLEEP or GL_RESET mode.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

CFDMAEn (n = 0, 1) bit (DMA Transfer Enable for Common FIFO 0 of Channel 0, 1)

The CFDMAEn bit enables or disables DMA transfer request for common FIFO 0 of channel 0 or 1. Only Common FIFO 0 can be linked to a DMA channel with this bit. Common FIFO 1 cannot be linked to a DMA channel.

To link Common FIFO 2, see bit CFDCDTTCT.CFDMAEn in **Section 30.2.44, DMA TX Transfer Control Register (CFDCDTTCT)**.

The CFDMAEn bit cannot be set in GL_SLEEP or GL_RESET mode.

Do not enable a DMA transfer for a Common FIFO that is configured as TX or GW FIFO.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

30.2.43 DMA Transfer Status Register (CFDCDTSTS)

The CFDCDTSTS register shows the status of the DMA transfer.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1334

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CFDMASTS1	CFDMASTS0	RFDMASTS7	RFDMASTS6	RFDMASTS5	RFDMASTS4	RFDMASTS3	RFDMASTS2	RFDMASTS1	RFDMASTS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
9	CFDMASTS1	0	R	DMA Transfer Status only for Common FIFO 0 of Channel 1 0: DMA transfer stopped 1: DMA transfer on going
8	CFDMASTS0	0	R	DMA Transfer Status only for Common FIFO 0 of Channel 0 0: DMA transfer stopped 1: DMA transfer on going
7	RFDMASTS7	0	R	DMA Transfer Status for RX FIFO 7 0: DMA transfer stopped 1: DMA transfer on going
6	RFDMASTS6	0	R	DMA Transfer Status for RX FIFO 6 0: DMA transfer stopped 1: DMA transfer on going
5	RFDMASTS5	0	R	DMA Transfer Status for RX FIFO 5 0: DMA transfer stopped 1: DMA transfer on going
4	RFDMASTS4	0	R	DMA Transfer Status for RX FIFO 4 0: DMA transfer stopped 1: DMA transfer on going
3	RFDMASTS3	0	R	DMA Transfer Status for RX FIFO 3 0: DMA transfer stopped 1: DMA transfer on going
2	RFDMASTS2	0	R	DMA Transfer Status for RX FIFO 2 0: DMA transfer stopped 1: DMA transfer on going
1	RFDMASTS1	0	R	DMA Transfer Status for RX FIFO 1 0: DMA transfer stopped 1: DMA transfer on going
0	RFDMASTS0	0	R	DMA Transfer Status for RX FIFO 0 0: DMA transfer stopped 1: DMA transfer on going

RFDMASTSn (n = 0 to 7) bit (DMA Transfer Status for RX FIFO n)

Number of RX FIFOs = 8

Each bit is set automatically when the corresponding DMA enable bit is set and the corresponding DMA FIFO is not empty.

Each bit is cleared automatically when the DMA transfer stops either because the DMA is disabled or the DMA FIFO is empty.

When CFDCDTCT.RFDMAEn (see CFDCDTCT.RFDMAEn bit in **Section 30.2.42, DMA Transfer Control Register (CFDCDTCT)**) is set to 0 while DMA transfer for the corresponding FIFO is ongoing, the RFDMASTSn bit becomes 0 when the DMA transfer is complete.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

CFDMASTSn (n = 0, 1) bit (DMA Transfer Status only for Common FIFO 0 of Channel 0, 1)

Each bit is set automatically when the corresponding DMA enable bit is set and the corresponding DMA FIFO is not empty.

Each bit is cleared automatically when the DMA transfer stops either because the DMA is disabled or the DMA FIFO is empty.

When CFDCDTCT.CFDMAEn (see CFDCDTCT.CFDMAEn bit in **Section 30.2.42, DMA Transfer Control Register (CFDCDTCT)**) is set to 0 while DMA transfer for the corresponding FIFO is ongoing, the CFDMASTSn bit becomes 0 when the DMA transfer is complete.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

30.2.44 DMA TX Transfer Control Register (CFDCDTTCT)

The CFDCDTTCT register controls the start and stop of DMA transfer operation.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1340

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CFDMAE1	CFDMAE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TQ3DMAE1	TQ3DMAE0	—	—	—	—	—	—	TQ0DMAE1	TQ0DMAE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
17	CFDMAE1	0	R/W	DMA TX Transfer Enable for Common FIFO 2 of Channel 1 0: DMA TX transfer request disabled 1: DMA TX transfer request enabled
16	CFDMAE0	0	R/W	DMA TX Transfer Enable for Common FIFO 2 of Channel 0 0: DMA TX transfer request disabled 1: DMA TX transfer request enabled
15 to 10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
9	TQ3DMAE1	0	R/W	DMA TX Transfer Enable for TXQ 3 of Channel 1 0: DMA TX transfer request disabled 1: DMA TX transfer request enabled
8	TQ3DMAE0	0	R/W	DMA TX Transfer Enable for TXQ 3 of Channel 0 0: DMA TX transfer request disabled 1: DMA TX transfer request enabled
7 to 2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
1	TQ0DMAE1	0	R/W	DMA TX Transfer Enable for TXQ 0 of Channel 1 0: DMA TX transfer request disabled 1: DMA TX transfer request enabled
0	TQ0DMAE0	0	R/W	DMA TX Transfer Enable for TXQ 0 of Channel 0 0: DMA TX transfer request disabled 1: DMA TX transfer request enabled

TQ0DMAEn (n = 0, 1) bit (DMA TX Transfer Enable for TXQ 0 of Channel n)

The TQ0DMAEn bit cannot be set in GL_SLEEP or GL_RESET mode.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

TQ3DMAEn (n = 0, 1) bit (DMA TX Transfer Enable for TXQ 3 of Channel n)

The TQ3DMAEn bit cannot be set in GL_SLEEP or GL_RESET mode.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

CFDMAEn (n = 0, 1) bit (DMA TX Transfer Enable for Common FIFO 2 of Channel n)

The CFDMAEn bit cannot be set in GL_SLEEP or GL_RESET mode.

Only common FIFO 2 can be linked to a DMA channel with this bit. Common FIFO 1 cannot be linked to a DMA channel. To link Common FIFO 0, see CFDCDTCT.CFDMAEn bit in **Section 30.2.42, DMA Transfer Control Register (CFDCDTCT)**.

Do not enable a DMA transfer for a Common FIFO that is configured as RX or GW FIFO.

The CFDMAEn bit is cleared when the CAN-FD module is in GL_RESET mode.

30.2.45 DMA TX Transfer Status Register (CFDCDTTSTS)

The CFDCDTTSTS register shows the status of the DMA TX transfer.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1344

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CFDMASTS1	CFDMASTS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TQ3DMASTS1	TQ3DMASTS0	—	—	—	—	—	—	TQ0DMASTS1	TQ0DMASTS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are read as 0.
17	CFDMASTS1	0	R	DMA TX Transfer Status for Common FIFO 2 of Channel 1 0: DMA TX transfer stopped 1: DMA TX transfer enabled
16	CFDMASTS0	0	R	DMA TX Transfer Status for Common FIFO 2 of Channel 0 0: DMA TX transfer stopped 1: DMA TX transfer enabled
15 to 10	—	All 0	R	Reserved These bits are read as 0.
9	TQ3DMASTS1	0	R	DMA TX Transfer Status for TXQ3 of Channel 1 0: DMA TX transfer stopped 1: DMA TX transfer enabled
8	TQ3DMASTS0	0	R	DMA TX Transfer Status for TXQ3 of Channel 0 0: DMA TX transfer stopped 1: DMA TX transfer enabled
7 to 2	—	All 0	R	Reserved These bits are read as 0.
1	TQ0DMASTS1	0	R	DMA TX Transfer Status for TXQ0 of Channel 1 0: DMA TX transfer stopped 1: DMA TX transfer enabled
0	TQ0DMASTS0	0	R	DMA TX Transfer Status for TXQ0 of Channel 0 0: DMA TX transfer stopped 1: DMA TX transfer enabled

TQ0DMASTSn (n = 0, 1) bit (DMA TX Transfer Status for TXQ 0 of Channel n)

The TQ0DMASTSn bit is set when the CFDCDTTCT.TQ0DMAEn bit in the corresponding CFDCDTTCT register is set (see **Section 30.2.44, DMA TX Transfer Control Register (CFDCDTTCT)**).

This bit is cleared when:

- The CFDCDTTCT.TQ0DMAEn bit in the corresponding CFDCDTTCT register is cleared
- The CAN-FD module is in GL_RESET mode.

TQ3DMASTSn (n = 0, 1) bit (DMA TX Transfer Status for TXQ 3 of Channel n)

The TQ3DMASTSn bit is set when the CFDCDTTCT.TQ3DMAEn bit in the corresponding CFDCDTTCT register is set (see **Section 30.2.44, DMA TX Transfer Control Register (CFDCDTTCT)**).

This bit is cleared when:

- The CFDCDTTCT.TQ3DMAEn bit in the corresponding CFDCDTTCT register is cleared
- The CAN-FD module is in GL_RESET mode.

CFDMASTSn (n = 0 ,1) bit (DMA TX Transfer Status only for Common FIFO 2 of Channel n)

The CFDMASTSn bit is set when the CFDCDTTCT.CFDMAEn bit in the corresponding CFDCDTTCT register is set (see **Section 30.2.44, DMA TX Transfer Control Register (CFDCDTTCT)**).

This bit is cleared when:

- The CFDCDTTCT.CFDMAEn bit in the corresponding CFDCDTTCT register is cleared
- The CAN-FD module is in GL_RESET mode.

30.2.46 Global RX Interrupt Status Register n (CFDGRINTSTSn) (n = 0, 1)

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1350 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CFOFRIF[2:0]			—	CFRFIF[2:0]			—	—	—	—	—	CFRIF[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	BQOFRIF[1:0]		—	QOFRIF[2:0]			—	—	BQFIF[1:0]		—	QFIF[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R/W	R	R	R	R/W	R/W	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
30 to 28	CFOFRIF[2:0]	000	R	Common FIFO One Frame RX Interrupt Flag Channel n (n = 0, 1) 0: Corresponding Common FIFO One Frame RX Interrupt flag is not set 1: Corresponding Common FIFO One Frame RX Interrupt flag is set
27	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
26 to 24	CFRFIF[2:0]	000	R	Common FIFO FDC Level Full Interrupt Flag Channel n (n = 0, 1) 0: Corresponding Common FIFO Full Interrupt flag is not set 1: Corresponding Common FIFO Full Interrupt flag is set
23 to 19	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
18 to 16	CFRIF[2:0]	000	R	Common FIFO RX Interrupt Flag Channel n (n = 0, 1) 0: Corresponding Common FIFO RX Interrupt flag is not set 1: Corresponding Common FIFO RX Interrupt flag is set
15, 14	—	00	R/W	Reserved These bits are read as 0. The write value should be always 0.
13, 12	BQOFRIF [1:0]	00	R	Borrowed TXQ One Frame RX Interrupt Flag Channel n (n = 0, 1) 0: Corresponding TXQ One Frame RX Interrupt flag is not set 1: Corresponding TXQ One Frame RX Interrupt flag is set
11	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
10 to 8	QOFRIF[2:0]	000	R	TXQ One Frame RX Interrupt Flag Channel n (n = 0, 1) 0: Corresponding TXQ One Frame RX Interrupt flag is not set 1: Corresponding TXQ One Frame RX Interrupt flag is set
7, 6	—	00	R/W	Reserved These bits are read as 0. The write value should be always 0.
5, 4	BQFIF[1:0]	00	R	Borrowed TXQ Full Interrupt Flag Channel n (n = 0, 1) 0: Corresponding TXQ Full Interrupt flag is not set 1: Corresponding TXQ Full Interrupt flag is set
3	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	QFIF[2:0]	000	R	TXQ Full Interrupt Flag Channel n (n = 0, 1) 0: Corresponding TXQ Full Interrupt flag is not set 1: Corresponding TXQ Full Interrupt flag is set

QFIF[2:0] bits (TXQ Full Interrupt Flag Channel n (n = 0, 1))

The QFIF[2:0] bits are set automatically when the TXQ Full Interrupt flag of the related channel is set when the interrupt is enabled.

The QFIF[2:0] bits are cleared automatically when:

- The related TXQ result status bits are cleared or the interrupt enable is disabled
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

BQFIF[1:0] bits (Borrowed TXQ Full Interrupt Flag Channel n (n = 0, 1))

The BQFIF[1:0] bits are set when a flexible transmission buffer assignment function is used and the borrowed TXQ is in full status. Operation is the same as CFDGRINTSTSn.QFIF.

The bit of the channel that lends TXMB is a reserved bit.

QOFRIF[2:0] bits (TXQ One Frame RX Interrupt Flag Channel n (n = 0, 1))

The QOFRIF[2:0] bits are set automatically when the TXQ One Frame RX Interrupt flag of the related channel is set when the interrupt is enabled.

The QOFRIF[2:0] bits are cleared automatically when:

- The related TXQ result status bits are cleared or the interrupt enable is disabled
- The CAN-FD module is in GL_RESET or CH_RESET mode.

BQOFRIF[1:0] bits (Borrowed TXQ One Frame RX Interrupt Flag Channel n (n = 0, 1))

The BQOFRIF[1:0] bits are set when a flexible transmission buffer assignment function is used and the borrowed TXQ receives one frame. Operation is the same as CFDGRINTSTSn.QOFRIF.

The bit of the channel that lends TXMB is a reserved bit.

CFRIF[2:0] bits (Common FIFO RX Interrupt Flag Channel n (n = 0, 1))

The CFRIF[2:0] bits are set automatically when the Common FIFO RX Interrupt flag of the related channel is set when the interrupt is enabled.

The CFRIF[2:0] bits are cleared automatically when:

- The related Common FIFO RX result status bits are cleared or the interrupt enable is disabled
- The CAN-FD module is in GL_RESET or CH_RESET mode.

CFRFIF[2:0] bits (Common FIFO FDC Level Full Interrupt Flag Channel n (n = 0, 1))

The CFRFIF[2:0] bits are set automatically when the Common FIFO Full Interrupt flag of the related channel is set when the interrupt is enabled.

The CFRFIF[2:0] bits are cleared automatically when:

- The related Common FIFO RX result status bits are cleared or the interrupt enable is disabled

- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

CFOFRIF[2:0] bits (Common FIFO One Frame RX Interrupt Flag Channel n (n = 0, 1))

The CFOFRIF[2:0] bits are set automatically when the Common FIFO One Frame RX Interrupt flag of the related channel is set when the interrupt is enabled.

The CFOFRIF[2:0] bits are cleared automatically when:

- The related Common FIFO RX result status bits are cleared or the interrupt enable is disabled
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

30.2.47 TX Message Buffer Control Register n (CFDTMCn) (n = 0 to 127)

The CFDTMCn register (n = 0 to 127) configures the TX message buffer functions.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'02D0 + H'01 × n

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	TMOM	TMTAR	TMTR
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
2	TMOM	0	R/W	TX Message Buffer One-shot Mode 0: TX message buffer not configured in one-shot mode 1: TX message buffer configured in one-shot mode
1	TMTAR	0	R/W	TX Message Buffer Transmission Abort Request 0: TX message buffer transmission request abort not requested 1: TX message buffer transmission request abort requested
0	TMTR	0	R/W	TX Message Buffer Transmission Request 0: TX Message buffer transmission not requested 1: TX message buffer transmission requested

TMTR bit (TX Message Buffer Transmission Request)

When the TMTR bit is set, the CAN-FD module logic tries to transmit the message stored in the corresponding message buffer.

Only write to this bit when the related CAN-FD module is in CH_HALT or CH_OPERATION mode.

Do not set this bit if the corresponding TX message buffer is linked to a COM FIFO in TX or GW mode or is a part of TX Queue.

This bit cannot be directly cleared by a CPU write access.

This bit can only be set when the Transmission Result flag bits (CFDTMSTSn.TMTRF[1:0]) in the CFDTMSTSn register corresponding to the message buffer are cleared to 00b.

The TMTR bit is automatically cleared by the:

- CAN-FD module logic at the end of a successful transmission
- CAN-FD module logic at the end of a transmission abort, requested by the corresponding CFDTMCn.TMTAR bit
- CAN-FD module logic when there is a detection of a CAN bus error or arbitration loss if CFDTMCn.TMOM bit is set for the message buffer
- CAN-FD module logic when the CAN-FD module is in GL_RESET mode or the related channel is in CH_RESET mode.

TMTAR bit (TX Message Buffer Transmission Abort Request)

When the TMTAR bit is set, the CAN-FD module logic tries to abort the transmission of the frame stored in the corresponding message buffer.

In most cases, transmission cannot be aborted if the internal scan for transmission is complete and the message buffer has already been selected for transmission. In this case, frame may be transmitted successfully from the message buffer. The message buffer selection is released by entering CH_HALT mode.

However, message buffer selected for transmission can be aborted by an abort request when the CAN node detects a new message on the bus (RX pin) before it starts transmission from the selected message buffer.

Only write to the TMTAR bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode. This bit can only be set when the related transmit request TMTR bit is set.

The TMTAR bit cannot be cleared by a CPU write access. Clearing of this bit by CAN-FD has priority over setting by a CPU write access.

The TMTAR bit is automatically cleared by:

- The CAN-FD module logic at the end of a successful transmission
- The CAN-FD module logic at the end of a transmission abort
- The CAN-FD module logic when there is detection of a CAN bus error or arbitration loss
- The CAN-FD module logic when the CAN-FD module is in GL_RESET mode or the related channel enters CH_RESET mode.

TMOM bit (TX Message Buffer One-shot Mode)

When the TMOM bit is set, the CAN-FD module logic tries to transmit the message only once.

If the transmission is successful, the CFDTMSTSn.TMTRF[1:0] bits are set to 10b or 11b. Otherwise, the transmission is automatically aborted and CFDTMSTSn.TMTRF[1:0] bits are set to 01b due to a bus error or a bus arbitration lost.

The TMOM bit remains set if the transmission has completed successfully or aborted due to an error or a loss of arbitration.

Only write to this bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Set this bit at the same time as the TMTR bit. Clear this bit with a write access.

If a message has already been requested for transmission, do not write to this bit until the message has been successfully transmitted or transmission has been aborted.

The TMOM bit is automatically cleared by the CAN-FD module logic when the CAN-FD module is in GL_RESET mode or the related channel is in CH_RESET mode.

30.2.48 TX Message Buffer Status Register n (CFDTMSTSn) (n = 0 to 127)

The CFDTMSTSn register (n = 0 to 127) shows status of the transmission and transmission abort for the corresponding message buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'07D0 + H'01 × n

Bit	7	6	5	4	3	2	1	0
	—	—	—	TMTARM	TMTRM	TMTRF[1:0]		TMTSTS
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.
4	TMTARM	0	R	TX Message Buffer Transmission Abort Request Mirrored 0: TX message buffer transmission request abort not requested 1: TX message buffer transmission request abort requested
3	TMTRM	0	R	TX Message Buffer Transmission Request Mirrored 0: TX message buffer transmission not requested 1: TX message buffer transmission requested
2, 1	TMTRF[1:0]	00	R/W	TX Message Buffer Transmission Result Flag 00: No result 01: Transmission aborted from the TX message buffer 10: Transmission successful from the TX message buffer and transmission abort was not requested 11: Transmission successful from the TX message buffer and transmission abort was requested
0	TMTSTS	0	R	TX Message Buffer Transmission Status 0: No on-going transmission 1: On-going transmission

TMTSTS bit (TX Message Buffer Transmission Status)

The TMTSTS bit is set automatically at the start of the transmission from the corresponding TX message buffer.

This bit is cleared automatically when:

- Transmission stops
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

TMTRF[1:0] bits (TX Message Buffer Transmission Result Flag)

The TMTRF[1:0] bits show the result for the corresponding TX message buffer. The status is as follows:

- 00b: Transmission in progress or has not been requested
- 01b: Transmission has been aborted from the corresponding TX message buffer
- 10b: Transmission was successful from the corresponding TX message buffer and the CFDTMCn.TMTAR bit was not set for this TX message buffer

- 11b: Transmission was successful from the corresponding TX message buffer, but the CFDTMCn.TMTAR bit was set for this TX message buffer.

Only write to these bits when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

The TMTRF[1:0] bits are cleared automatically when the CAN-FD module is in GL_RESET mode or the related channel is in CH_RESET mode.

TMTRM bit (TX Message Buffer Transmission Request Mirrored)

The TMTRM bit is set when the CFDTMCn.TMTR bit in the corresponding CFDTMCn register is set.

This bit is cleared when the CFDTMCn.TMTR bit in the corresponding CFDTMCn register is cleared.

TMTARM bit (TX Message Buffer Transmission Abort Request Mirrored)

The TMTARM bit is set when the CFDTMCn.TMTAR bit in the corresponding CFDTMCn register is set.

This bit is cleared when the CFDTMCn.TMTAR bit in the corresponding CFDTMCn register is cleared.

30.2.49 TX Message Buffer Transmission Request Status Register n (CFDTMTRSTSn) (n = 0 to 3)

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0CD0 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTRSTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0.
15 to 0	TMTRSTS [15:0]	All 0	R	TX Message Buffer Transmission Request Status 0: Transmission not requested for corresponding TX message buffer 1: Transmission requested for corresponding TX message buffer

TMTRSTS[15:0] bits (TX Message Buffer Transmission Request Status)

The TMTRSTS[15:0] bits show status of the CFDTMCn.TMTR bits of the TX Message Buffer Control Registers.

Alignment of the TMTRSTS[15:0] bits is shown in **Table 30.8**.

Table 30.8 Alignment of TMTRSTS[15:0] Mirror Bits

Bit Position	TX Message Buffer Number
n × 64 − fmin × 32	n × 64 + 0
n × 64 + 1 − fmin × 32	n × 64 + 1
⋮	⋮
n × 64 + 31 − fmin × 32	n × 64 + 31
n × 64 + 32 − fmax × 32	n × 64 + 32
n × 64 + 33 − fmax × 32	n × 64 + 33
⋮	⋮
n ×64 + 62 − fmax × 32	n × 64 + 62
n × 64 + 63 − fmax × 32	n × 64 + 63

Note: When n = 0, fmin = 0, fmax = 1
When n = 1, fmin = 2, fmax = 3

Each bit is set automatically when the corresponding bit is set in the TX Message Buffer Control Registers (CFDTMCn), and only when the message buffer does not belong to a TX Queue.

Each bit is cleared automatically when:

- The corresponding bit is cleared in the TX Message Buffer Control Registers

- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

30.2.50 TX Message Buffer Transmission Abort Request Status Register n (CFDTMTARSTSn) (n = 0 to 3)

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0D70 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTARSTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0.
15 to 0	TMTARSTS [15:0]	All 0	R	TX Message Buffer Transmission Abort Request Status 0: Transmission abort not requested for corresponding TX message buffer 1: Transmission abort requested for corresponding TX message buffer

TMTARSTS[15:0] bits (TX Message Buffer Transmission Abort Request Status)

The TMTARSTS[15:0] bits show status of the CFDTMCn.TMTAR bits of the TX Message Buffer Control Registers. Alignment of the TMTARSTS[15:0] bits is shown in **Table 30.9**.

Table 30.9 Alignment of TMTARSTS[15:0] Mirror Bits

Bit Position	TX Message Buffer Number
n × 64 − fmin × 32	n × 64 + 0
n × 64 + 1 − fmin × 32	n × 64 + 1
⋮	⋮
n × 64 + 31 − fmin × 32	n × 64 + 31
n × 64 + 32 − fmax × 32	n × 64 + 32
n × 64 + 33 − fmax × 32	n × 64 + 33
⋮	⋮
n × 64 + 62 − fmax × 32	n × 64 + 62
n × 64 + 63 − fmax × 32	n × 64 + 63

Note: When n = 0, fmin = 0, fmax = 1
When n = 1, fmin = 2, fmax = 3

Each bit is set automatically when the corresponding bit is set in the TX Message Buffer Control Registers, and when the message buffer belongs to a TX Queue.

Each bit is cleared automatically when:

- The corresponding bit is cleared in the TX Message Buffer Control Registers

- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

30.2.51 TX Message Buffer Transmission Completion Status Register n (CFDTMTCSTSn) (n = 0 to 3)

Base address: H'0_100C_0000

Offset address: H'0E10 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTCSTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0.
15 to 0	TMTCSTS [15:0]	All 0	R	TX Message Buffer Transmission Completion Status 0: Transmission not completed for corresponding TX message buffer 1: Transmission completed for corresponding TX message buffer

TMTCSTS[15:0] bits (TX Message Buffer Transmission Completion Status)

The TMTCSTS[15:0] bits show status of successful completion of the TX Message Buffer Status Registers.

Alignment of the TMTCSTS[15:0] bits is shown in **Table 30.10**.

Table 30.10 Alignment of TMTCSTS[15:0] Mirror Bits

Bit Position	TX Message Buffer Number
n × 64 − fmin × 32	n × 64 + 0
n × 64 + 1 − fmin × 32	n × 64 + 1
⋮	⋮
n × 64 + 31 − fmin × 32	n × 64 + 31
n × 64 + 32 − fmax × 32	n × 64 + 32
n × 64 + 33 − fmax × 32	n × 64 + 33
⋮	⋮
n × 64 + 62 − fmax × 32	n × 64 + 62
n × 64 + 63 − fmax × 32	n × 64 + 63

Note: When n = 0, fmin = 0, fmax = 1
When n = 1,fmin = 2, fmax = 3

Each bit is set automatically when the corresponding bit is set in the TX Message Buffer Status Registers.

Each bit is cleared automatically when:

- The corresponding bit is cleared in the TX Message Buffer Status Registers
- The CAN-FD module is in GL_RESET mode

- The related CAN-FD channel is in CH_RESET mode.

If a CAN channel enters CH_RESET mode, then the bits related to that channel are cleared.

30.2.52 TX Message Buffer Transmission Abort Status Register n (CFDTMTASTSn) (n = 0 to 3)

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0EB0 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTASTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0.
15 to 0	TMTASTS [15:0]	All 0	R	TX Message Buffer Transmission Abort Status 0: Transmission not aborted for corresponding TX message buffer 1: Transmission aborted for corresponding TX message buffer

TMTASTS[15:0] bits (TX Message Buffer Transmission Abort Status)

The TMTASTS[15:0] bits show status of the successful transmission abort of the corresponding TX message buffer.

Alignment of the TMTASTS[15:0] bits is shown in **Table 30.11**.

Table 30.11 Alignment of TMTASTS[15:0] Mirror Bits

Bit Position	TX Message Buffer Number
n × 64 − fmin × 32	n × 64 + 0
n × 64 + 1 − fmin × 32	n × 64 + 1
⋮	⋮
n × 64 + 31 − fmin × 32	n × 64 + 31
n × 64 + 32 − fmax × 32	n × 64 + 32
n × 64 + 33 − fmax × 32	n × 64 + 33
⋮	⋮
n × 64 + 62 − fmax × 32	n × 64 + 62
n × 64 + 63 − fmax × 32	n × 64 + 63

Note: When n = 0, fmin = 0, fmax = 1
When n = 1, fmin = 2, fmax = 3

Each bit is set automatically when the CFDTMSTSn.TMTRF[1:0] bits are set to 01b in the corresponding TX Message Buffer Status Register.

Each bit is cleared automatically when:

- The CFDTMSTSn.TMTRF[1:0] bits are cleared in the corresponding TX Message Buffer Status Register

- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

30.2.53 TX Message Buffer Transmission Interrupt Enable Register n (CFDTMIECn) (n = 0 to 3)

Base address: CAN-FD = H'0_100C_0000

Offset address: H'0F50 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMIE[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
15 to 0	TMIE[15:0]	All 0	R/W	TX Message Buffer Interrupt Enable 0: TX message buffer interrupt disabled for corresponding TX message buffer 1: TX message buffer interrupt enabled for corresponding TX message buffer

TMIE[15:0] bits (TX Message Buffer Interrupt Enable)

If the TMIE[15:0] bits are set, an interrupt is generated at the end of a successful transmission from the corresponding message buffer.

See **Section 30.7, Interrupt and DMA** for TX Message Buffer Interrupt specification.

Alignment of the TMIE[15:0] bits is shown in **Table 30.12**.

Table 30.12 Alignment of TMIE[15:0] bits

Bit Position	TX Message Buffer Number
n × 64 − fmin × 32	n × 64 + 0
n × 64 + 1 − fmin × 32	n × 64 + 1
⋮	⋮
n × 64 + 31 − fmin × 32	n × 64 + 31
n × 64 + 32 − fmax × 32	n × 64 + 32
n × 64 + 33 − fmax × 32	n × 64 + 33
⋮	⋮
n × 64 + 62 − fmax × 32	n × 64 + 62
n × 64 + 63 − fmax × 32	n × 64 + 63

Note: When n = 0, fmin = 0, fmax = 1
When n = 1, fmin = 2, fmax = 3

Do not write to the TMIE[15:0] bits when:

- The CAN-FD module is in GL_SLEEP mode

- The related CAN-FD channel is in CH_SLEEP mode
- The corresponding TX message buffer is part of a TX Queue
- The corresponding TX message buffer is linked to a Common FIFO with the CFDCFCCn.CFTML[4:0] bits.

30.2.54 TX Queue Configuration/Control Register 0n (CFDTXQCC0n) (n = 0, 1)

The CFDTXQCC0n register (n = 0, 1) is used to configure the TX Queue transmission.

TXQ0 is composed of TXMB0 to TXMB31 (at the maximum) when TXQE is enabled.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1000 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	TXQOFTXIE	TXQOFRXIE	TXQFIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	TXQDC[4:0]					TXQIM	—	TXQTXIE	—	—	TXQOWE	TXQGWE	TXQE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
18	TXQOFTXIE	0	R/W	TXQ One Frame Transmission Interrupt Enable 0: One Frame TX interrupt generation disabled 1: One Frame TX interrupt generation enabled
17	TXQOFRXIE	0	R/W	TXQ One Frame Reception Interrupt Enable 0: One Frame RX interrupt generation disabled 1: One Frame RX interrupt generation enabled
16	TXQFIE	0	R/W	TXQ Full Interrupt Enable 0: TX Queue full interrupt generation disabled 1: TX Queue full interrupt generation enabled
15 to 13	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.
12 to 8	TXQDC[4:0]	All 0	R/W	TX Queue Depth Configuration H'00: 0 messages H'01: Reserved H'02: 3 messages H'03: 4 messages ⋮ H'0F: 16 messages Others: Setting prohibited
7	TXQIM	0	R/W	TX Queue Interrupt Mode 0: When the last message is successfully transmitted 1: At every successful transmission
6	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
5	TXQTXIE	0	R/W	TX Queue TX Interrupt Enable 0: TX Queue TX interrupt disabled 1: TX Queue TX interrupt enabled
4, 3	—	00	R/W	Reserved These bits are read as 0. The write value should be always 0.

Bit	Bit Name	Initial Value	R/W	Description
2	TXQOWE	0	R/W	TX Queue Overwrite Mode Enable 0: TX Queue OW mode disabled 1: TX Queue OW mode enabled
1	TXQGWE	0	R/W	TX Queue Gateway Mode Enable 0: TX Queue GW mode disabled 1: TX Queue GW mode enabled
0	TXQE	0	R/W	TX Queue Enable 0: TX Queue disabled 1: TX Queue enabled

TXQE bit (TX Queue Enable)

The TXQE bit cannot be set if the configured TX Queue depth is H'00 (CFDTXQCC0n.TXQDC[4:0] = H'00).

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_RESET or CH_SLEEP mode.

The TXQE bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

TXQGWE bit (TX Queue Gateway Mode Enable)

When the TXQGWE bit is set, the TX Queue is in TX Queue GW mode.

When this bit is set, CPU must not access the TX Queue.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQOWE bit (TX Queue Overwrite Mode Enable)

When the TXQOWE bit is set, the TX Queue is in TX Queue Overwrite mode.

An overwrite function is valid when the same ID such as ID of the data written in from the gateway or CPU is in TX Queue. For example, when a frame is received and is stored into the TX Queue, if a message with the same ID is stored in the TX Queue, the old message is overwritten by the new message.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

When using the function in GW mode, the depth of TXQ (CFDTXQCC0n.TXQDC) should be configured to the value which is the various number of ID that is used in the TX Queue plus 3.

The function is valid for the standard ID frame but is invalid for the extended ID frame.

Do not modify this bit when the CFDTXQCC0n.TXQE bit is 1.

TXQTXIE bit (TX Queue TX Interrupt Enable)

When the TXQTXIE bit is set, an interrupt is generated based on the setting of the TXQIM bit.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

TXQIM bit (TX Queue Interrupt Mode)

The TXQIM bit selects the interrupt generation condition for the TX Queue.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQDC[4:0] bits (TX Queue Depth Configuration)

The TXQDC[4:0] bits select the depth of the transmission queue. The message buffer selection starts from MB[0] up to MB[31] depending on the configured depth.

When using TXQ1 and TXQ0 simultaneously, the total depth of TXQ1 and TXQ0 should be 32 or less.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQFIE bit (TXQ Full Interrupt Enable)

When the TXQFIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS0n.TXQFIF bit.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

Only write 1 to this bit in Gateway mode (CFDTXQCC0n.TXQGWE = 1).

TXQOFRXIE bit (TXQ One Frame Reception Interrupt Enable)

When the TXQOFRXIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS0n.TXQOFRXIF bit.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

Only write 1 to this bit in GW mode (CFDTXQCC0n.TXQGWE = 1).

TXQOFTXIE bit (TXQ One Frame Transmission Interrupt Enable)

When the TXQOFTXIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS0n.TXQOFTXIF bit.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

30.2.55 TX Queue Configuration/Control Register 1n (CFDTXQCC1n) (n = 0, 1)

The CFDTXQCC1n register (n = 0, 1) is used to configure the TX Queue transmission.

TXQ1 is composed of TXMB31 to TXMB0 (at the maximum) when TXQE is enabled.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1060 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	TXQOFTXIE	TXQOFRXIE	TXQFIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	TXQDC[4:0]					TXQIM	—	TXQTXIE	—	—	TXQOWE	TXQGWE	TXQE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
18	TXQOFTXIE	0	R/W	TXQ One Frame Transmission Interrupt Enable 0: One Frame TX interrupt generation disabled 1: One Frame TX interrupt generation enabled
17	TXQOFRXIE	0	R/W	TXQ One Frame Reception Interrupt Enable 0: One Frame RX interrupt generation disabled 1: One Frame RX interrupt generation enabled
16	TXQFIE	0	R/W	TXQ Full Interrupt Enable 0: TX Queue full interrupt generation disabled 1: TX Queue full interrupt generation enabled
15 to 13	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.
12 to 8	TXQDC[4:0]	All 0	R/W	TX Queue Depth Configuration H'00: 0 messages H'01: Reserved H'02: 3 messages H'03: 4 messages ⋮ H'0F: 16 messages Others: Setting prohibited
7	TXQIM	0	R/W	TX Queue Interrupt Mode 0: When the last message is successfully transmitted 1: At every successful transmission
6	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
5	TXQTXIE	0	R/W	TX Queue TX Interrupt Enable 0: TX Queue TX interrupt disabled 1: TX Queue TX interrupt enabled
4, 3	—	00	R/W	Reserved These bits are read as 0. The write value should be always 0.

Bit	Bit Name	Initial Value	R/W	Description
2	TXQOWE	0	R/W	TX Queue Overwrite Mode Enable 0: TX Queue OW mode disabled 1: TX Queue OW mode enabled
1	TXQGWE	0	R/W	TX Queue Gateway Mode Enable 0: TX Queue GW mode disabled 1: TX Queue GW mode enabled
0	TXQE	0	R/W	TX Queue Enable 0: TX Queue disabled 1: TX Queue enabled

TXQE bit (TX Queue Enable)

The TXQE bit cannot be set if the configured TX Queue depth is H'00 (CFDTXQCC1n.TXQDC[4:0] = H'00).

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do write to this bit when the related CAN-FD channel is in CH_RESET or CH_SLEEP mode.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

TXQGWE bit (TX Queue Gateway Mode Enable)

When the TXQGWE bit is set, the TX Queue is in TX Queue Gateway mode.

When this bit is set, CPU must not access the TX Queue.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQOWE bit (TX Queue Overwrite Mode Enable)

When the TXQOWE bit is set, the TX Queue is in TX Queue Overwrite mode.

An overwrite function is valid when the same ID such as ID of the data written in from the gateway or CPU is in TX Queue. For example, when a frame is received and is stored into the TX Queue, if a message with the same ID is stored in the TX Queue, the old message is overwritten by the new message.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

When using the function in GW mode, the depth of TXQ (CFDTXQCC1n.TXQDC) should be configured to the value which is the various number of ID that is used in the TX Queue plus 3.

The function is valid for the standard ID frame but is invalid for the extended ID frame.

Do not modify this bit when the CFDTXQCC1n.TXQE bit is 1.

TXQTXIE bit (TX Queue TX Interrupt Enable)

When the TXQTXIE bit is set, an interrupt is generated based on the setting of the TXQIM bit.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

TXQIM bit (TX Queue Interrupt Mode)

The TXQIM bit selects the interrupt generation condition for the TX Queue.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQDC[4:0] bits (TX Queue Depth Configuration)

The TXQDC[4:0] bits select the depth of the transmission queue. The message buffer selection starts from MB[31] down to MB[0] depending on the configured depth.

When using TXQ1 and TXQ0 simultaneously, the total depth of TXQ1 and TXQ0 should be 32 or less.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQFIE bit (TXQ Full Interrupt Enable)

When the TXQFIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS1n.TXQFIF bit.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

Only write 1 to this bit when in Gateway mode (CFDTXQCC1n.TXQGWE = 1).

TXQOFRXIE bit (TXQ One Frame Reception Interrupt Enable)

When the TXQOFRXIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS1n.TXQOFRXIF bit.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

Only write 1 to this bit when in Gateway mode (CFDTXQCC1n.TXQGWE = 1).

TXQOFTXIE bit (TXQ One Frame Transmission Interrupt Enable)

When the TXQOFTXIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS1n.TXQOFTXIF bit.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

30.2.56 TX Queue Configuration/Control Register 2n (CFDTXQCC2n) (n = 0, 1)

The CFDTXQCC2n register (n = 0, 1) is used to configure the TX Queue transmission.

TXQ2 is composed of TXMB32 to TXMB63 (at the maximum) when TXQE is enabled.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'10C0 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	TXQOFTXIE	TXQOFRXIE	TXQFIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	TXQDC[4:0]					TXQIM	—	TXQTXIE	—	—	TXQOWE	TXQGWE	TXQE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
18	TXQOFTXIE	0	R/W	TXQ One Frame Transmission Interrupt Enable 0: One Frame TX interrupt generation disabled 1: One Frame TX interrupt generation enabled
17	TXQOFRXIE	0	R/W	TXQ One Frame Reception Interrupt Enable 0: One Frame RX interrupt generation disabled 1: One Frame RX interrupt generation enabled
16	TXQFIE	0	R/W	TXQ Full Interrupt Enable 0: TX Queue full interrupt generation disabled 1: TX Queue full interrupt generation enabled
15 to 13	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.
12 to 8	TXQDC[4:0]	All 0	R/W	TX Queue Depth Configuration H'00: 0 messages H'01: Reserved H'02: 3 messages H'03: 4 messages ⋮ H'0F: 16 messages Others: Setting prohibited
7	TXQIM	0	R/W	TX Queue Interrupt Mode 0: When the last message is successfully transmitted 1: At every successful transmission
6	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
5	TXQTXIE	0	R/W	TX Queue TX Interrupt Enable 0: TX Queue TX interrupt disabled 1: TX Queue TX interrupt enabled
4, 3	—	00	R/W	Reserved These bits are read as 0. The write value should be always 0.

Bit	Bit Name	Initial Value	R/W	Description
2	TXQOWE	0	R/W	TX Queue Overwrite Mode Enable 0: TX Queue OW mode disabled 1: TX Queue OW mode enabled
1	TXQGWE	0	R/W	TX Queue Gateway Mode Enable 0: TX Queue GW mode disabled 1: TX Queue GW mode enabled
0	TXQE	0	R/W	TX Queue Enable 0: TX Queue disabled 1: TX Queue enabled

TXQE bit (TX Queue Enable)

The TXQE bit cannot be set if the configured TX Queue depth is H'00 (CFDTXQCC2n.TXQDC[4:0] = H'00).

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_RESET or CH_SLEEP mode.

TXQGWE bit (TX Queue Gateway Mode Enable)

When the TXQGWE bit is set, the TX Queue is in TX Queue Gateway mode. When this bit is set, CPU must not access the TX Queue.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQOWE bit (TX Queue Overwrite Mode Enable)

When the TXQOWE bit is set, the TX Queue is in TX Queue Overwrite mode.

An overwrite function is valid when the same ID such as ID of the data written in from the gateway or CPU is in TX Queue. For example, when a frame is received and is stored into the TX Queue, if a message with the same ID is stored in the TX Queue, the old message is overwritten by the new message.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

When using the function in GW mode, the depth of TXQ (CFDTXQCC2n.TXQDC) should be configured to the value which is the various number of ID that is used in the TX Queue plus 3.

The function is valid for the standard ID frame but is invalid for the extended ID frame.

Do not modify this bit when the CFDTXQCC2n.TXQE bit is 1.

TXQTXIE bit (TX Queue TX Interrupt Enable)

When the TXQTXIE bit is set, an interrupt is generated based on the setting of the TXQIM bit.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

TXQIM bit (TX Queue Interrupt Mode)

The TXQIM bit selects the interrupt generation condition for the TX Queue.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQDC[4:0] bits (TX Queue Depth Configuration)

The TXQDC[4:0] bits select the depth of the transmission queue. The message buffer selection starts from MB[32] up to MB[63] depending on the configured depth.

When using TXQ3 and TXQ2 simultaneously, the total depth of TXQ3 and TXQ2 should be 32 or less.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION

TXQFIE bit (TXQ Full Interrupt Enable)

When the TXQFIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS2n.TXQFIF bit.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

Only write 1 to this bit when in GW mode (CFDTXQCC2n.TXQGWE = 1).

TXQOFRXIE bit (TXQ One Frame Reception Interrupt Enable)

When the TXQOFRXIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS2n.TXQOFRXIF bit.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

Only write 1 to this bit when in GW mode (CFDTXQCC2n.TXQGWE = 1).

TXQOFTXIE bit (TXQ One Frame Transmission Interrupt Enable)

When the TXQOFTXIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS2n.TXQOFTXIF bit.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

30.2.57 TX Queue Configuration/Control Register 3n (CFDTXQCC3n) (n = 0, 1)

The CFDTXQCC3n register (n = 0, 1) is used to configure the TX Queue transmission.

TXQ3 is composed of TXMB63 to TXMB32 (at the maximum) when TXQE is enabled.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1120 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	TXQOFTXIE	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	TXQDC[4:0]					TXQIM	—	TXQTXIE	—	—	TXQOWE	—	TXQE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
18	TXQOFTXIE	0	R/W	TXQ One Frame Transmission Interrupt Enable 0: One Frame TX interrupt generation disabled 1: One Frame TX interrupt generation enabled
17 to 13	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
12 to 8	TXQDC[4:0]	All 0	R/W	TX Queue Depth Configuration H'00: 0 messages H'01: Reserved H'02: 3 messages H'03: 4 messages ⋮ H'0F: 16 messages Others: Setting prohibited
7	TXQIM	0	R/W	TX Queue Interrupt Mode 0: When the last message is successfully transmitted 1: At every successful transmission
6	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
5	TXQTXIE	0	R/W	TX Queue TX Interrupt Enable 0: TX Queue TX interrupt disabled 1: TX Queue TX interrupt enabled
4, 3	—	00	R/W	Reserved These bits are read as 0. The write value should be always 0.
2	TXQOWE	0	R/W	TX Queue Overwrite Mode Enable 0: TX Queue OW mode disabled 1: TX Queue OW mode enabled
1	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.

Bit	Bit Name	Initial Value	R/W	Description
0	TXQE	0	R/W	TX Queue Enable 0: TX Queue disabled 1: TX Queue enabled

TXQE bit (TX Queue Enable)

The TXQE bit cannot be set if the configured TX Queue depth is H'00 (CFDTXQCC3n.TXQDC[4:0] = H'00).

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_RESET or CH_SLEEP mode.

TXQOWE bit (TX Queue Overwrite Mode Enable)

When the TXQOWE bit is set, the TX Queue is in TX Queue Overwrite mode.

An overwrite function is valid when the same ID such as ID of the data written in from the gateway or CPU is in TX Queue. For example, when a frame is received and is stored into the TX Queue, if a message with the same ID is stored in the TX Queue, the old message is overwritten by the new message.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

When using the function in GW mode, the depth of TXQ (CFDTXQCC3n.TXQDC) should be configured to the value which is the various number of ID that is used in the TX Queue plus 3.

The function is valid for the standard ID frame but is invalid for the extended ID frame.

Do not modify this bit when the CFDTXQCC3n.TXQE bit is 1.

TXQTXIE bit (TX Queue TX Interrupt Enable)

When the TXQTXIE bit is set, an interrupt is generated based on the setting of the TXQIM bit.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

TXQIM bit (TX Queue Interrupt Mode)

The TXQIM bit selects the interrupt generation condition for the TX Queue.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQDC[4:0] bits (TX Queue Depth Configuration)

The TXQDC[4:0] bits select the depth of the transmission queue. The message buffer selection starts from MB[63] down to MB[32] depending on the configured depth.

When using TXQ3 and TXQ2 simultaneously, the total depth of TXQ3 and TXQ2 should be 32 or less.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQOFTXIE bit (TXQ One Frame Transmission Interrupt Enable)

When the TXQOFTXIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS3n.TXQOFTXIF bit.

Cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

30.2.58 TX Queue Status Register 0n (CFDTXQSTS0n) (n = 0, 1)

The CFDTXQSTS0n register (n = 0, 1) shows the status of the TX Queue of corresponding CAN Channel.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1020 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	TXQMOW	TXQMLT	TXQOFTXIF	TXQOFRXIF	TXQFIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXQMC[5:0]						—	—	—	—	—	TXQTXIF	TXQFLL	TXQEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.
28	—	0	R	Reserved These bits are read as 0.
27 to 21	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
20	TXQMOW	0	R/W	TXQ Message Overwrite 0: No message overwrite in TXQ 1: Message overwrite in TXQ
19	TXQMLT	0	R/W	TXQ Message Lost 0: No message lost in TXQ 1: TXQ message lost
18	TXQOFTXIF	0	R/W	TXQ One Frame Transmission Interrupt Flag When one frame transmits from TXQ, an interrupt is set.
17	TXQOFRXIF	0	R/W	TXQ One Frame Reception Interrupt Flag When TXQ receives one frame, an interrupt is set.
16	TXQFIF	0	R/W	TXQ Full Interrupt Flag When TXQ is in full status, an interrupt is set.
15, 14	—	00	R/W	Reserved These bits are read as 0. The write value should be always 0.
13 to 8	TXQMC[5:0]	All 0	R	TX Queue Message Count Number of messages in the TX Queue.
7 to 3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
2	TXQTXIF	0	R/W	TX Queue TX Interrupt Flag 0: TX Queue interrupt condition not satisfied after a frame TX 1: TX Queue interrupt condition satisfied after a frame TX
1	TXQFLL	0	R	TX Queue Full 0: TX Queue not full 1: TX Queue full

Bit	Bit Name	Initial Value	R/W	Description
0	TXQEMP	1	R	TX Queue Empty 0: TX Queue not empty 1: TX Queue empty

TXQEMP bit (TX Queue Empty)

The TXQEMP bit is set automatically when the TX Queue is disabled or no messages are stored in the TX Queue.

This bit is set automatically when:

- The last message is transmitted from the TX Queue
- The related CAN-FD channel is in CH_RESET mode.

This bit is cleared automatically when the first message to be transmitted is stored in the TX Queue.

TXQFLL bit (TX Queue Full)

The TXQFLL bit is set automatically when the number of CAN messages stored in the TX Queue matches the configured TX Queue depth.

This bit is cleared automatically when:

- The number of CAN messages stored in the TX Queue is less than the configured TX Queue depth
- The related CAN-FD channel is in CH_RESET mode.

TXQTXIF bit (TX Queue TX Interrupt Flag)

The TXQTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of the TX Queue.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is set automatically when the configured interrupt condition is satisfied for the TX Queue.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

This bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMC[5:0] bits (TX Queue Message Count)

The TXQMC[5:0] bits show the number of CAN messages in the TX Queue.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

TXQFIF bit (TXQ Full Interrupt Flag)

The TXQFIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of the TX Queue.

Only in Gateway mode (CFDTXQCC0n.TXQGWE = 1) that this bit is set automatically when the TX Queue transits to a buffer full status.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQOFRXIF bit (TXQ One Frame Reception Interrupt Flag)

The TXQOFRXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When receiving data is stored in the TX Queue in Gateway mode, this bit is set automatically.

This function can only be used in Gateway mode of the TX Queue.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQOFTXIF bit (TXQ One Frame Transmission Interrupt Flag)

The TXQOFTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When transmission is successful in the TX Queue, this bit is set automatically.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

This bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMLT bit (TXQ Message Lost)

The TXQMLT bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When a message lost occurs in Gateway mode of the TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMOW bit (TXQ Message Overwrite)

The TXQMOW bit is not cleared automatically if the TX Queue is disabled.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When CFDTXQCC0n.TXQOWE = 1 and message overwrite occurs in TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

30.2.59 TX Queue Status Register 1n (CFDTXQSTS1n) (n = 0, 1)

The CFDTXQSTS1n register (n = 0, 1) shows the status of the TX Queue of corresponding CAN Channel.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1080 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	TXQMOW	TXQMLT	TXQOFTXIF	TXQOFRXIF	TXQFIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXQMC[5:0]						—	—	—	—	—	TXQTXIF	TXQFLL	TXQEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
20	TXQMOW	0	R/W	TXQ Message Overwrite 0: No message overwrite in TXQ 1: Message overwrite in TXQ
19	TXQMLT	0	R/W	TXQ Message Lost 0: No message lost in TXQ 1: TXQ message lost
18	TXQOFTXIF	0	R/W	TXQ One Frame Transmission Interrupt Flag When one frame transmits from TXQ, an interrupt is set.
17	TXQOFRXIF	0	R/W	TXQ One Frame Reception Interrupt Flag When TXQ receives one frame, an interrupt is set.
16	TXQFIF	0	R/W	TXQ Full Interrupt Flag When TXQ is in full status, an interrupt is set.
15, 14	—	00	R/W	Reserved These bits are read as 0. The write value should be always 0.
13 to 8	TXQMC[5:0]	All 0	R	TX Queue Message Count Number of messages in the TX Queue
7 to 3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
2	TXQTXIF	0	R/W	TX Queue TX Interrupt Flag 0: TX Queue interrupt condition not satisfied after a frame TX 1: TX Queue interrupt condition satisfied after a frame TX
1	TXQFLL	0	R	TX Queue Full 0: TX Queue not full 1: TX Queue full
0	TXQEMP	1	R	TX Queue Empty 0: TX Queue not empty 1: TX Queue empty

TXQEMP bit (TX Queue Empty)

The TXQEMP bit is set automatically when the TX Queue is disabled or no messages are stored in the TX Queue.

This bit is set automatically when:

- The last message is transmitted from the TX Queue
- The related CAN-FD channel is in CH_RESET mode.

This bit is cleared automatically when the first message to be transmitted is stored in the TX Queue.

TXQFLL bit (TX Queue Full)

The TXQFLL bit is set automatically when the number of CAN messages stored in the TX Queue matches the configured TX Queue depth.

This bit is cleared automatically when:

- The number of CAN messages stored in the TX Queue is less than the configured TX Queue depth
- The related CAN-FD channel is in CH_RESET mode.

TXQTXIF bit (TX Queue TX Interrupt Flag)

The TXQTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of the TX Queue.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is set automatically when the configured interrupt condition is satisfied for the TX Queue.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

This bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMC[5:0] bits (TX Queue Message Count)

The TXQMC[5:0] bits show the number of CAN messages in the TX Queue.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

TXQFIF bit (TXQ Full Interrupt Flag)

The TXQFIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of the TX Queue.

Only in Gateway mode (CFDTXQCC1n.TXQGWE = 1) that this bit is set automatically when the TX Queue transits to a buffer full status.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQOFRXIF bit (TXQ One Frame Reception Interrupt Flag)

The TXQOFRXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When receiving data is stored in the TX Queue in Gateway mode, this bit is set automatically.

This function can only be used in Gateway mode of the TX Queue.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQOFTXIF bit (TXQ One Frame Transmission Interrupt Flag)

The TXQOFTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When transmission is successful in the TX Queue, this bit is set automatically.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

The bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMLT bit (TXQ Message Lost)

The TXQMLT bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When a message lost occurs in Gateway mode of the TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode

TXQMOW bit (TXQ Message Overwrite)

The TXQMOW bit is not cleared automatically if the TX Queue is disabled.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When CFDTXQCC1n.TXQOWE = 1 and message overwrite occurs in TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

30.2.60 TX Queue Status Register 2n (CFDTXQSTS2n) (n = 0, 1)

The CFDTXQSTS2n register (n = 0, 1) shows the status of the TX Queue of corresponding CAN Channel.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'10E0 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	TXQMOW	TXQMLT	TXQOFTXIF	TXQOFRXIF	TXQFIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXQMC[5:0]						—	—	—	—	—	TXQTXIF	TXQFLL	TXQEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
20	TXQMOW	0	R/W	TXQ Message Overwrite 0: No message overwrite in TXQ 1: Message overwrite in TXQ
19	TXQMLT	0	R/W	TXQ Message Lost 0: No message lost in TXQ 1: TXQ message lost
18	TXQOFTXIF	0	R/W	TXQ One Frame Transmission Interrupt Flag When one frame transmits from TXQ, an interrupt is set.
17	TXQOFRXIF	0	R/W	TXQ One Frame Reception Interrupt Flag When TXQ receives one frame, an interrupt is set.
16	TXQFIF	0	R/W	TXQ Full Interrupt Flag When TXQ is in full status, an interrupt is set.
15, 14	—	00	R/W	Reserved These bits are read as 0. The write value should be always 0.
13 to 8	TXQMC[5:0]	All 0	R	TX Queue Message Count Number of messages in the TX Queue.
7 to 3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
2	TXQTXIF	0	R/W	TX Queue TX Interrupt Flag 0: TX Queue interrupt condition not satisfied after a frame TX 1: TX Queue interrupt condition satisfied after a frame TX
1	TXQFLL	0	R	TX Queue Full 0: TX Queue not full 1: TX Queue full
0	TXQEMP	1	R	TX Queue Empty 0: TX Queue not empty 1: TX Queue empty

TXQEMP bit (TX Queue Empty)

The TXQEMP bit is set automatically when the TX Queue is disabled or no messages are stored in the TX Queue.

This bit is set automatically when:

- The last message is transmitted from the TX Queue
- The related CAN-FD channel is in CH_RESET mode.

This bit is cleared automatically when the first message to be transmitted is stored in the TX Queue.

TXQFLL bit (TX Queue Full)

The TXQFLL bit is set automatically when the number of CAN messages stored in the TX Queue matches the configured TX Queue depth.

This bit is cleared automatically when:

- The number of CAN messages stored in the TX Queue is less than the configured TX Queue depth
- The related CAN-FD channel is in CH_RESET mode.

TXQTXIF bit (TX Queue TX Interrupt Flag)

The TXQTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of the TX Queue.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is set automatically when the configured interrupt condition is satisfied for the TX Queue.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

The bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMC[5:0] bits (TX Queue Message Count)

The TXQMC[5:0] bits show the number of CAN messages in the TX Queue.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

TXQFIF bit (TXQ Full Interrupt Flag)

The TXQFIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of the TX Queue.

Only in Gateway mode (CFDTXQCC2n.TXQGWE = 1) that this bit is set automatically when the TX Queue transits to a buffer full status.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQOFRXIF bit (TXQ One Frame Reception Interrupt Flag)

The TXQOFRXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When receiving data is stored in the TX Queue in Gateway mode, this bit is set automatically.

This function can only be used in Gateway mode of the TX Queue.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQOFTXIF bit (TXQ One Frame Transmission Interrupt Flag)

The TXQOFTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When transmission is successful in the TX Queue, this bit is set automatically.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

The bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMLT bit (TXQ Message Lost)

The TXQMLT bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When a message lost occurs in Gateway mode of the TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMOW bit (TXQ Message Overwrite)

The TXQMOW bit is not cleared automatically if the TX Queue is disabled.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When CFDTXQCC2n.TXQOWE = 1 and message overwrite occurs in TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

30.2.61 TX Queue Status Register 3n (CFDTXQSTS3n) (n = 0, 1)

The CFDTXQSTS3n register (n = 0, 1) shows the status of the TX Queue of corresponding CAN Channel.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1140 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	TXQMOW	—	TXQOFTXIF	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXQMC[5:0]						—	—	—	—	—	TXQTXIF	TXQFLL	TXQEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.
28	—	0	R	Reserved These bits are read as 0.
27 to 21	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
20	TXQMOW	0	R/W	TXQ Message Overwrite 0: No message overwrite in TXQ 1: Message overwrite in TXQ
19	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
18	TXQOFTXIF	0	R/W	TXQ One Frame Transmission Interrupt Flag When one frame transmits from TXQ, an interrupt is set.
17 to 14	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
13 to 8	TXQMC[5:0]	All 0	R	TX Queue Message Count Number of messages in the TX Queue.
7 to 3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
2	TXQTXIF	0	R/W	TX Queue TX Interrupt Flag 0: TX Queue interrupt condition not satisfied after a frame TX 1: TX Queue interrupt condition satisfied after a frame TX
1	TXQFLL	0	R	TX Queue Full 0: TX Queue not full 1: TX Queue full
0	TXQEMP	1	R	TX Queue Empty 0: TX Queue not empty 1: TX Queue empty

TXQEMP bit (TX Queue Empty)

The TXQEMP bit is set automatically when the TX Queue is disabled or no messages are stored in the TX Queue.

This bit is set automatically when:

- The last message is transmitted from the TX Queue
- The related CAN-FD channel is in CH_RESET mode.

This bit is cleared automatically when the first message to be transmitted is stored in the TX Queue.

TXQFLL bit (TX Queue Full)

The TXQFLL bit is set automatically when the number of CAN messages stored in the TX Queue matches the configured TX Queue depth.

This bit is cleared automatically when:

- The number of CAN messages stored in the TX Queue is less than the configured TX Queue depth
- The related CAN-FD channel is in CH_RESET mode.

TXQTXIF bit (TX Queue TX Interrupt Flag)

The TXQTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of the TX Queue.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is set automatically when the configured interrupt condition is satisfied for the TX Queue.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

This bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMC[5:0] bits (TX Queue Message Count)

The TXQMC[5:0] bits show the number of CAN messages in the TX Queue.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

TXQOFTXIF bit (TXQ One Frame Transmission Interrupt Flag)

The TXQOFTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When transmission is successful in the TX Queue, this bit is set automatically.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

This bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMOW bit (TXQ Message Overwrite)

The TXQMOW bit is not cleared automatically if the TX Queue is disabled.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When CFDTXQCC3n.TXQOWE = 1 and message overwrite occurs in TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1. Writing 1 has no effect.

Cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

30.2.62 TX Queue Pointer Control Register 0n (CFDTXQPCTR0n) (n = 0, 1)

The CFDTXQPCTR0n register (n = 0, 1) is used to confirm storage of a full message in the corresponding TX Queue buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1040 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
7 to 0	TXQPC[7:0]	H'00	W	TX Queue Pointer Control Increments the write pointer to the TX Queue buffer in the corresponding channel

TXQPC[7:0] bits (TX Queue Pointer Control)

When the value H'FF is written to the TXQPC[7:0] bits, the write pointer of the corresponding TX Queue buffer is updated and a transmit request is initiated for this message.

The read value from these bits is always H'00. Do not write to the FIFO control registers when DMA is enabled.

Cannot write to these bits when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Only write H'FF to this register when:

- The corresponding TX Queue is enabled and not full
- The Common FIFO is enabled and is not configured in GW mode.

30.2.63 TX Queue Pointer Control Register 1n (CFDTXQPCTR1n) (n = 0, 1)

The CFDTXQPCTR1n register (n = 0, 1) is used to confirm storage of a full message in the corresponding TX Queue buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'10A0 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
7 to 0	TXQPC[7:0]	H'00	W	TX Queue Pointer Control Increments the write pointer to the TX Queue buffer in the corresponding channel

TXQPC[7:0] bits (TX Queue Pointer Control)

When the value H'FF is written to the TXQPC[7:0] bits, the write pointer of the corresponding TX Queue buffer is updated and a transmit request is initiated for this message.

The read value from these bits is always H'00. Cannot write to the FIFO control registers when DMA is enabled.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Only write H'FF to this register when:

- The corresponding TX Queue is enabled and not full
- The Common FIFO is enabled and is not configured in GW mode.

30.2.64 TX Queue Pointer Control Register 2n (CFDTXQPCTR2n) (n = 0, 1)

The CFDTXQPCTR2n register (n = 0, 1) is used to confirm storage of a full message in the corresponding TX Queue buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1100 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
7 to 0	TXQPC[7:0]	H'00	W	TX Queue Pointer Control Increments the write pointer to the TX Queue buffer in the corresponding channel

TXQPC[7:0] bits (TX Queue Pointer Control)

When the value H'FF is written to the TXQPC[7:0] bits, the write pointer of the corresponding TX Queue buffer is updated and a transmit request is initiated for this message.

The read value from these bits is always H'00. Cannot write to the FIFO control registers when DMA is enabled.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Only write H'FF to this register when:

- The corresponding TX Queue is enabled and not full
- The Common FIFO is enabled and is not configured in GW mode.

30.2.65 TX Queue Pointer Control Register 3n (CFDTXQPCTR3n) (n = 0, 1)

The CFDTXQPCTR3n register (n = 0, 1) is used to confirm storage of a full message in the corresponding TX Queue buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1160 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
7 to 0	TXQPC[7:0]	H'00	W	TX Queue Pointer Control Increments the write pointer to the TX Queue buffer in the corresponding channel

TXQPC[7:0] bits (TX Queue Pointer Control)

When the value H'FF is written to the TXQPC[7:0] bits, the write pointer of the corresponding TX Queue buffer is updated and a transmit request is initiated for this message.

The read value from these bits is always H'00. Do not write to the FIFO control registers when DMA is enabled.

Cannot write to these bits when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Only write H'FF to this register when:

- The corresponding TX Queue is enabled and not full
- The Common FIFO is enabled and is not configured in Gateway mode.

30.2.66 TX Queue Empty Status Register (CFDTXQESTS)

The CFDTXQESTS register shows the status of the empty bits of the TXQ buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1180

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQxEMP[7:0]							
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 1	R	Reserved These bits are read as 1.
7 to 0	TXQxEMP [7:0]	H'FF	R	TXQ Empty Status 0: TXQ not empty 1: TXQ empty

TXQxEMP[7:0] bits (TXQ Empty Status)

Each bit is set automatically when the corresponding bit is set in the TX Queue Empty Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue Empty Status Register.

This bit is set when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Channel 0 TX Queue 3
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
7	Channel 1 TX Queue 3
31:8	Reserved

30.2.67 TX Queue Full Interrupt Status Register (CFDTXQFISTS)

The CFDTXQFISTS register shows the status of the full interrupt bits of the TXQ buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1184

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TXQ1FULL[2:0]			—	TXQ0FULL[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
6 to 4	TXQ1FULL [2:0]	000	R	TXQ Full Interrupt Status Flag for Channel 1 0: TXQ full interrupt not set 1: TXQ full interrupt set
3	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
2 to 0	TXQ0FULL [2:0]	000	R	TXQ Full Interrupt Status Flag for Channel 0 0: TXQ full interrupt not set 1: TXQ full interrupt set

TXQnFULL[2:0] (n = 0 ,1) bits (TXQ Full Interrupt Status Flag)

Each bit is set automatically when the corresponding bit is set in the TX Queue Full Interrupt Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue Full Interrupt Status Register.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Reserved
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
31:7	Reserved

30.2.68 TX Queue Message Lost Status Register (CFDTXQMSTS)

The CFDTXQMSTS register shows the status of the message lost bits of the TXQ buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1188

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TXQ1ML[2:0]			—	TXQ0ML[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
6 to 4	TXQ1ML[2:0]	000	R	TXQ Message Lost Status Flag for Channel 1 0: TXQ message lost flag not set 1: TXQ message lost flag set
3	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
2 to 0	TXQ0ML[2:0]	000	R	TXQ Message Lost Status Flag for Channel 0 0: TXQ message lost flag not set 1: TXQ message lost flag set

TXQnML[2:0] (n = 0, 1) bits (TXQ Message Lost Status Flag)

Each bit is set automatically when the corresponding bit is set in the TX Queue Message Lost Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue Message Lost Status Register.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Reserved
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
31:7	Reserved

30.2.69 TX Queue Message Overwrite Status Register (CFDTXQOWSTS)

The CFDTXQMSTS register shows the status of the message lost bits of the TXQ buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'118C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQ1OW[3:0]				TXQ0OW[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0.
7 to 4	TXQ1OW[3:0]	H'0	R	TXQ Message Overwrite Status Flag for Channel 1 0: TXQ message overwrite flag not set 1: TXQ message overwrite flag set
3 to 0	TXQ0OW[3:0]	H'0	R	TXQ Message Overwrite Status Flag for Channel 0 0: TXQ message overwrite flag not set 1: TXQ message overwrite flag set

TXQnOW[3:0] (n = 0 ,1) bits (TXQ Message Overwrite Status Flag)

Each bit is set automatically when the corresponding bit is set in the TX Queue Message Overwrite Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue Message Overwrite Status Register.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Channel 0 TX Queue 3
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
7	Channel 1 TX Queue 3
31:8	Reserved

30.2.70 TX Queue Interrupt Status Register (CFDTXQISTS)

The CFDTXQISTS register shows the status of the interrupt flag of the TXQ buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1190

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQ1ISF[3:0]				TXQ0ISF[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0.
7 to 4	TXQ1ISF[3:0]	H'0	R	TXQ Interrupt Status Flag for Channel 1 0: TXQ Interrupt flag not set 1: TXQ Interrupt flag set
3 to 0	TXQ0ISF[3:0]	H'0	R	TXQ Interrupt Status Flag for Channel 0 0: TXQ Interrupt flag not set 1: TXQ Interrupt flag set

TXQnISF[3:0] (n = 0, 1) bits (TXQ Interrupt Status Flag)

Each bit is set automatically when the corresponding bit is set in the TX Queue Interrupt Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue Interrupt Status Register.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Channel 0 TX Queue 3
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
7	Channel 1 TX Queue 3

30.2.71 TX Queue One Frame TX Interrupt Status Register (CFDTXQOFTISTS)

The CFDTXQOFTISTS register shows the status of the One Frame TX Interrupt flag of the TXQ buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1194

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQ1OFTISF[3:0]				TXQ0OFTISF[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0.
7 to 4	TXQ1OFTISF [3:0]	H'0	R	TXQ One Frame TX Interrupt Status Flag for Channel 1 0: TXQ One Frame TX Interrupt flag not set 1: TXQ One Frame TX Interrupt flag set
3 to 0	TXQ0OFTISF [3:0]	H'0	R	TXQ One Frame TX Interrupt Status Flag for Channel 0 0: TXQ One Frame TX Interrupt flag not set 1: TXQ One Frame TX Interrupt flag set

TXQnOFTISF[3:0] (n = 0, 1) bits (TXQ One Frame TX Interrupt Status Flag)

Each bit is set automatically when the corresponding bit is set in the TX Queue One Frame TX Interrupt Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue One Frame TX Interrupt Status Register.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Channel 0 TX Queue 3
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
7	Channel 1 TX Queue 3

30.2.72 TX Queue One Frame RX Interrupt Status Register (CFDTXQOFRISTS)

The CFDTXQOFRISTS register shows the status of the One Frame RX Interrupt flag of the TXQ buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1198

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TXQ1OFRISF[2:0]			—	TXQ0OFRISF[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are read as 0.
6 to 4	TXQ1OFRISF[2:0]	000	R	TXQ One Frame RX Interrupt Status Flag for Channel 1 0: TXQ One Frame RX Interrupt flag not set 1: TXQ One Frame RX Interrupt flag set
3	—	0	R	Reserved This bit is read as 0.
2 to 0	TXQ0OFRISF[2:0]	000	R	TXQ One Frame RX Interrupt Status Flag for Channel 0 0: TXQ One Frame RX Interrupt flag not set 1: TXQ One Frame RX Interrupt flag set

TXQnOFRISF[2:0] (n = 0, 1) bits (TXQ One Frame RX Interrupt Status Flag)

Each bit is set automatically when the corresponding bit is set in the TX Queue One Frame RX Interrupt Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue One Frame RX Interrupt Status Register.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Reserved
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
7	Reserved

30.2.73 TX Queue Full Status Register (CFDTXQFSTS)

The CFDTXQFSTS register shows the status of the Full Status flag bits of the TXQ buffers.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'119C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQ1FSF[3:0]				TXQ0FSF[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0.
7 to 4	TXQ1FSF [3:0]	H'0	R	TXQ Full Status Flag for Channel 1 0: TXQ Full flag not set 1: TXQ Full flag set
3 to 0	TXQ0FSF [3:0]	H'0	R	TXQ Full Status Flag for Channel 0 0: TXQ Full flag not set 1: TXQ Full flag set

TXQnFSF[3:0] (n = 0, 1) bits (TXQ Full Status Flag)

Each bit is set automatically when the corresponding bit is set in the TX Queue Full Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue Full Status Register.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Channel 0 TX Queue 3
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
7	Channel 1 TX Queue 3

30.2.74 TX History List Configuration/Control Register n (CFDTHLCCn) (n = 0, 1)

The CFDTHLCCn register (n = 0, 1) configures the TX History List functions.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1200 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	THLDGE	THLDTE	THLIM	THLIE	—	—	—	—	—	—	—	THLE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
11	THLDGE	0	R/W	TX History List Dedicated Gateway Enable 0: Not dedicated Gateway FIFO + Gateway TX Queue 1: Dedicated Gateway FIFO + Gateway TX Queue
10	THLDTE	0	R/W	TX History List Dedicated TX Enable 0: TX FIFO + TX Queue 1: Flat TX MB + TX FIFO + TX Queue
9	THLIM	0	R/W	TX History List Interrupt Mode 0: Interrupt generated if TX History List level reaches ¾ of the TX History List depth 1: Interrupt generated for every successfully stored entry
8	THLIE	0	R/W	TX History List Interrupt Enable 0: TX History List Interrupt disabled 1: TX History List Interrupt enabled
7 to 1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
0	THLE	0	R/W	TX History List Enable 0: TX History List disabled 1: TX History List enabled

THLE bit (TX History List Enable)

The THLE bit enables the TX History List buffer when it is set.

Do not write to this bit when the related CAN-FD channel is in CH_RESET or CH_SLEEP mode.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

THLIE bit (TX History List Interrupt Enable)

The THLIE bit enables the generation of the TX History List interrupt when it is set.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

THLIM bit (TX History List Interrupt Mode)

The THLIM bit selects the interrupt generation condition for the FIFO.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode.

THLDTE bit (TX History List Dedicated TX Enable)

The THLDTE bit selects the condition for storing an entry in the TX History List after successful transmission.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode.

THLDGE bit (TX History List Dedicated Gateway Enable)

The THLDGE bit selects the condition for storing an entry in the TX History List after successful transmission.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode.

30.2.75 TX History List Status Register n (CFDTHLSTSn) (n = 0, 1)

The CFDTHLSTSn register (n = 0, 1) shows the status of data stored in the TX History List buffer.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1220 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	THLMC[5:0]						—	—	—	—	THLIF	THLELT	THLFLL	THLEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
13 to 8	THLMC[5:0]	All 0	R	TX History List Message Count Number of messages stored in TX History List
7 to 4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
3	THLIF	0	R/W	TX History List Interrupt Flag 0: TX History List interrupt condition not satisfied 1: TX History List interrupt condition satisfied
2	THLELT	0	R/W	TX History List Entry Lost 0: No entry lost in TX History List 1: TX History List entry Lost
1	THLFLL	0	R	TX History List Full 0: TX History List not full 1: TX History List full
0	THLEMP	1	R	TX History List Empty 0: TX History List not empty 1: TX History List empty

THLEMP bit (TX History List Empty)

The THLEMP bit is set automatically when the CPU has read all the entries from the TX History List buffer.

This bit is cleared automatically when the first entry is stored to the TX History List.

This bit is set automatically when:

- TX History List is disabled
- The related CAN-FD channel is in CH_RESET mode.

THLFLL bit (TX History List Full)

The THLFLL bit is set automatically when the number of entries in the TX History List buffer matches the TX History List depth.

Each TX History List can store up to 32 entries (each channel has a dedicated TX History List).

This bit is cleared automatically when:

- The number of entries in the TX History List buffer is less than the TX History List depth
- The TX History List is disabled
- The related CAN-FD channel is in CH_RESET mode.

THLELT bit (TX History List Entry Lost)

The THLELT bit is set when a new entry cannot be stored because the related TX History List buffer is already full.

Only write to this bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode. Writing 1 has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

This bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

THLIF bit (TX History List Interrupt Flag)

The THLIF bit is set when the configured interrupt condition is satisfied.

Only write to this bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode. Writing 1 has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1.

This bit is cleared:

- By writing 0 to it
- When the related CAN-FD channel is in CH_RESET mode.

THLMC[5:0] bits (TX History List Message Count)

The THLMC[5:0] bits show the number of transmitted messages stored in the TX History List.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

30.2.76 Channel n TX History List Access Register 0 (CFDTHLACC0n) (n = 0, 1)

The CFDTHLACC0n register (n = 0, 1) provides access to the entry in the TX History List based on the read timestamp value.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'8000 + H'08 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TGW	—	—	—	—	—	BN[6:0]							BT[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMTS[15:0]	All 0	R	Transmit Timestamp Transmit timestamp value for software drivers.
15	TGW	0	R	Transmit Gateway Buffer Indication 0: No transmission from gateway 1: Transmission from gateway
14 to 10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
9 to 3	BN[6:0]	All 0	R	Buffer Number Number of the message buffer
2 to 0	BT[2:0]	000	R	Buffer Type 001: Flat TX message buffer 010: TX FIFO message buffer number and gateway FIFO message number 100: TX Queue message buffer number Others: Setting prohibited

BT[2:0] bits (Buffer Type)

The BT[2:0] bits indicate whether data has been stored following a transmission from a FIFO buffer, a TX Queue or a TX message buffer.

BN[6:0] bits (Buffer Number)

The BN[6:0] bits show the message buffer from which transmission was successfully completed. If a message from a Common FIFO is transmitted, then these bits show the message buffer that is linked to the Common FIFO for transmission.

TGW bit (Transmit Gateway Buffer Indication)

The TGW bit is automatically set to 1 when transmission is completed in GW mode.

TMTS[15:0] bits (Transmit Timestamp)

The TMTS[15:0] bits indicate the timestamp for use by software drivers.

In the case of can_race_ts_en=1, the information as which CFDTHLACC0n.TMTS is inputted from can_race_ts is reflected.

See **Section 30.8.1.3 Timestamp** for the timestamp in detail.

30.2.77 Channel n TX History List Access Register 1 (CFDTHLACC1n) (n = 0, 1)

The CFDTHLACC1n register (n = 0, 1) provides access to entry in the TX History List based on the read pointer value.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'8004 + H'08 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TIFL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R/W	Reserved These bits are read as 0. The write value should be aways 0.
17, 16	TIFL[1:0]	00	R	Transmit Information Label These bits indicate that message buffer information label, TX FIFO information label, or AFL information label is stored for software drivers.
15 to 0	TID[15:0]	All 0	R	Transmit ID These bits indicate that message buffer reference ID, TX FIFO reference ID, or AFL pointer field is stored for software drivers.

TID[15:0] bits (Transmit ID)

The TID[15:0] bits indicate whether the message buffer reference ID (CFDTMFDCTRn.TMPTR) or the TX FIFO reference ID (CFDCFFDCSTSn.CFPTR) is for use by software drivers.

When transmission in Gateway mode, these bits indicate the AFL pointer field (CFDGAFLP0n.GAFLPTR) instead of the message buffer reference ID (CFDTMFDCTRn.TMPTR).

TIFL[1:0] bits (Transmit Information Label)

The TIFL[1:0] bits indicate whether the message buffer information label (CFDTMFDCTRn.TMIFL) or the TX FIFO information label (CFDCFFDCSTSn.CFIFL) is for use by software drivers.

When transmission in Gateway mode, these bits indicate the AFL pointer field (CFDGAFLMn.GAFLIFL1 and CFDGAFLP0n.GAFLIFL0) instead of the MB information label (CFDTMFDCTRn.TMIFL).

30.2.78 TX History List Pointer Control Register n (CFDTHLPCTRn) (n = 0, 1)

The CFDTHLPCTRn register (n = 0, 1) is used to increment the read pointer of the TX History List.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1240 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	THLPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
7 to 0	THLPC[7:0]	H'00	W	TX History List Pointer Control Increments the write pointer to the TX History List in the corresponding channel

THLPC[7:0] bits (TX History List Pointer Control)

When H'FF is written to the THLPC[7:0] bits, the read pointer of the TX History List is moved to the next TX History List entry address.

The read value from these bits is always H'00. Only write to these bits when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Only write H'FF to these registers when the corresponding TX History List is enabled and not empty.

30.2.79 Global Reset Control Register (CFDGRSTC)

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1380

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	KEY[7:0]								—	—	—	—	—	—	—	SRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
15 to 8	KEY[7:0]	H'00	W	Key Code These bits control the validity of rewriting of the SRST bit.
7 to 1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
0	SRST	0	R/W	Software Reset 0: Normal state 1: Software reset state

SRST bit (Software Reset)

When the SRST bit is set, the CAN-FD module is in the same state as hardware reset. When a reset is required, write 1 then write 0 to this bit.

This bit is cleared when the CAN-FD module is in GL_SLEEP mode.

When this bit is cleared, the RAM initialization sequence does not operate. The configuration of RAM is performed by software.

The RAM is not initialized when software reset is performed during the initialization of RAM. Software must perform the initialization of RAM.

KEY[7:0] bits (Key Code)

When H'C4 is written in the KEY[7:0] bits, a write to the SRST bit is valid.

The read value from these bits is always H'00.

Write a SRST bit and the KEY bits simultaneously.

30.2.80 Global Flexible CAN Mode Configuration Register (CFDGFCMC)

Flexible CAN mode configured in the CFDGFCMC register and Flexible transmission buffer assignment configured in the CFDGFTBAC register should not be used simultaneously.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1384

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FLXC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
0	FLXC0	0	R/W	Flexible CAN Mode between Channel 0 and Channel 1 0: Normal mode 1: Flexible CAN mode

FLXC0 bit (Flexible CAN Mode between Channel 0 and Channel 1)

When the FLXC0 bit is set, Channel 0 and Channel 1 of a CAN-FD module are in Flexible CAN mode.

Channel 1 uses TX/RX terminal of Channel 0. The TX/RX terminal of Channel 1 cannot be used.

Only write to this bit when the CAN-FD module is in GL_RESET mode.

30.2.81 Global Flexible Transmission Buffer Assignment Configuration Register (CFDGFTBAC)

Flexible transmission buffer assignment configured in the CFDGFTBAC register and Flexible CAN mode configured in the CFDGFCMC register should not be used simultaneously.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'138C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	FLXMB0[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
3 to 0	FLXMB0[3:0]	H'0	R/W	Flexible Transmission Buffer Assignment between Channel 0 and Channel 1 By setting these bits, the even channel can use the configured number of TX mailboxes of the odd channel H'0: 0 H'1: 4 H'2: 8 H'3: 12 H'4: 16 H'5: 20 H'6: 24 H'7: 28 H'8: 32 Others: Setting prohibited

FLXMB0[3:0] bits (Flexible Transmission Buffer Assignment between Channel 0 and Channel 1)

Channel 0 can use the number TXMB of channel 1 from 0 to 32 by the configuration of these bits.

Only write to this bit when the CAN-FD module is in GL_RESET mode.

30.2.82 Global Test Configuration Register (CFDGTSTCFG)

The CFDGTSTCFG register is used to configure the CAN channels joining the internal CAN bus communication test mode and the RAM test mode page.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1308

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	RTMPS[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	C1ICBCE	C0ICBCE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
25 to 16	RTMPS[9:0]	All 0	R/W	RAM Test Mode Page Select Select a RAM test mode page
15 to 2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
1	C1ICBCE	0	R/W	Channel 1 Internal CAN Bus Communication Test Mode Enable 0: Channel 1 internal CAN bus communication disabled 1: Channel 1 internal CAN bus communication enabled
0	C0ICBCE	0	R/W	Channel 0 Internal CAN Bus Communication Test Mode Enable 0: Channel 0 internal CAN bus communication disabled 1: Channel 0 internal CAN bus communication enabled

CnICBCE (n = 0, 1) bits (Channel n Internal CAN Bus Communication Test Mode Enable)

When the C0ICBCE and C1ICBCE bits are set and CAN-FD module is configured in the internal CAN bus communication test mode, then CAN channel n joins the internal CAN bus communication test mode operation.

Do not write to these bits when the CAN-FD module is in GL_RESET or GL_SLEEP mode.

Only write to these bits when the CAN-FD module is in GL_HALT mode.

These bits are cleared automatically when the CAN-FD module is in GL_RESET mode.

RTMPS[9:0] bits (RAM Test Mode Page Select)

The RTMPS[9:0] bits select the RAM page mode for CPU read/write access when the CAN-FD module is configured in RAM test mode.

See **Section 30.9.2.1, RAM Test Mode** for the RAM test mode specification.

Do not write to these bits when the CAN-FD module is in GL_RESET or GL_SLEEP mode.

Only enter values from 0 to 15 (H'00F) for the AFL RAM and 16 to 76 (H'04C) for the message buffer RAM.

The setting range of these bits depends on the combination of parameters.

Only write to these bits when the CAN-FD module is in GL_HALT mode.

These bits are cleared automatically when the related CAN-FD channel is in GL_RESET mode.

30.2.83 Global Test Control Register (CFDGTSTCTR)

The CFDGTSTCTR register is used to control the global test modes of the CAN-FD module.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'130C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RTME	—	ICBCTME
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
2	RTME	0	R/W	RAM Test Mode Enable 0: RAM test mode disabled 1: RAM test mode enabled
1	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
0	ICBCTME	0	R/W	Internal CAN Bus Communication Test Mode Enable 0: Internal CAN Bus Communication test mode disabled 1: Internal CAN Bus Communication test mode enabled

ICBCTME bit (Internal CAN Bus Communication Test Mode Enable)

When the ICBCTME bit is set, internal CAN bus communication is enabled for the CAN channels that are configured for internal CAN bus communication participation. See **Section 30.9.2.2, Internal CAN Bus Communication Mode** for the specification of internal CAN bus communication test mode.

Only write to this bit when the CAN-FD module is in GL_HALT mode.

Clear this bit when the CAN-FD module is in GL_HALT mode.

This bit is cleared automatically when the CAN-FD module is in GL_RESET mode.

RTME bit (RAM Test Mode Enable)

When the RTME bit is set, the CAN-FD module is configured in RAM test mode. See **Section 30.9.2.1, RAM Test Mode** for RAM test mode specification.

Only write to this bit when the CAN-FD module is in GL_HALT mode.

Clear this bit when the CAN-FD module is in GL_HALT mode.

This bit is cleared automatically when the CAN-FD module is in GL_RESET mode.

30.2.84 Global FD Configuration Register (CFDGFDCFG)

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1314

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSCCFG[1:0]		—	—	—	—	—	—	—	RPED
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
9, 8	TSCCFG[1:0]	00	R/W	Timestamp Capture Configuration 00: Timestamp capture at the sample point of SOF (start of frame) 01: Timestamp capture at frame valid indication 10: Timestamp capture at the sample point of RES bit 11: Reserved
7 to 1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
0	RPED	0	R/W	RES Bit Protocol Exception Disable 0: Protocol exception event detection enabled 1: Protocol exception event detection disabled

RPED bit (RES Bit Protocol Exception Disable)

The RPED bit configures the protocol exception event handling according to ISO 11898-1.

When this bit is enabled, the protocol exception event detection is disabled, and the protocol controller transmits an error frame when the protocol exception event is detected (RES bit is sampled recessive).

Only write to this bit when the CAN-FD module is in GL_RESET mode.

TSCCFG[1:0] bits (Timestamp Capture Configuration)

The TSCCFG[1:0] bits configure the different capture points of the timestamp for transmission and reception.

When CFDGFDCFG.TSCCFG[1:0] = 10b, the timestamp capture is performed for CAN-FD frames at RES bit and for Classical frames at the start of frame.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

Set 01b to these bits when can_race_ts_en is 1.

See **Section 30.8.1.3 Timestamp** for the timestamp in detail.

30.2.85 Global Lock Key Register (CFDGLOCKK)

The CFDGLOCKK register is a write-only register that is used to unlock the protection for special test bits. See **Section 30.9.2, Global Test Modes** for Lock key specification.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'131C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCK[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
15 to 0	LOCK[15:0]	H'0000	W	Lock Key Key bits for unlocking the protection of test modes

LOCK[15:0] bits (Lock Key)

The unlock key sequence must be written in the LOCK[15:0] bits to configure the CAN-FD module in RAM test modes.

The read value from these bits is always H'0000.

Cannot write to these bits when the CAN-FD module is in GL_SLEEP or GL_RESET mode.

Do not write to these bits when the CAN-FD module is in GL_OPERATION mode.

30.2.86 RAM Test Page Access Register n (CFDRPGACCn) (n = 0 to 63)

Base address: CAN-FD = H'0_100C_0000

Offset address: H'8400 + H'04 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDTA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDTA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDTA[31:0]	All 0	R/W	RAM Data Test Access RAM data bytes

RDTA[31:0] bits (RAM Data Test Access)

Data can be read from or written into the RDTA[31:0] bits when the CAN-FD module is configured in RAM test mode.

Only write to this bit when the CAN-FD module is in GL_HALT mode and RAM test mode is enabled.

If data is read when RAM test mode is not enabled, then it is always read as H'0000_0000.

Software data should be read/written in the RAM Test Page Access registers during RAM test mode.

30.2.87 Channel n Bus Load Control Register (CFDCnBLCT) (n = 0, 1)

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1418 + H'20 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	BLCLD	—	—	—	—	—	—	—	BLCE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
8	BLCLD	0	W	Bus Load Counter Load When CFDCnBLCT.BLCLD is set, it is reset after a bus load counter value is loaded.
7 to 1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
0	BLCE	0	R/W	Bus Load Counter Enable 0: Bus load counter disabled 1: Bus load counter enabled

BLCE bit (Bus Load Counter Enable)

The BLCE bit indicates the enabling signal of a bus load counter. Write to this bit after setting up the CFDCnNCFG register.

When this bit is 0, the bus load counter stops but the counter is not clear.

The bit is cleared when the related CAN-FD channel is in CH_RESET mode.

Do not write to this bit in CH_SLEEP mode.

BLCLD bit (Bus Load Counter Load)

When the BLCLD bit is set, the bus load counter value is loaded into CFDCnBLSTS.BLC and the bus load counter is reset.

The read value is always 0.

30.2.88 Channel n Bus Load Status Register (CFDCnBLSTS) (n = 0, 1)

Base address: CAN-FD = H'0_100C_0000

Offset address: H'141C + H'20 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BLC[28:13]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BLC[12:0]													—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	BLC[28:0]	All 0	R	Bus Load Counter Status These bits indicate the bus load counter value.
2 to 0	—	000	R/W	Reserved These bits are read as 0. The write value should be always 0.

BLC[28:0] bits (Bus Load Counter Status)

The bus load counter increases by PCLKM/2 period while CAN bus is in an idle state.

When the CFDCnBLCT.BLCLD bit is set, the bus load counter value is loaded to BLC[28:0] bits and the bus load counter is reset.

These bits are set only by the CAN-FD module. Writing any value has no effect.

30.2.89 Identifier Bits Alignment

Standard Identifier (11 bits) format: ID28 to ID18 is aligned to b10 to b0, b11 to b28 should be 0.

Extended Identifier (29 bits) format: ID28 to ID0 is aligned to b28 to b0

For Standard Identifier format bits 11 to 28 (b11 to b28) should be all 0.

Table 30.13 Standard Identifier (11-Bit Format)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
IDE = 0	RTR	—	0	0	0	0	0	0	0	0	0	0	0	0	0

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0	0	ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21	ID20	ID19	ID18

Table 30.14 Extended Identifier (29-Bit Format)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
IDE = 1	RTR	—	ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21	ID20	ID19	ID18	ID17	ID16

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0

30.2.90 Message Buffer Component Structure

30.2.90.1 Start Addresses

The start address for each of the Message Buffer component is calculated using the number of related Message Buffer components and the number of channels.

Table 30.15 Channel Components Features

number_of_channels	i	2
number_of_RMBCPs_per_channel	b	16
number_of_RFMBCPs	b	8
number_of_CFMBCPs_per_channel	b	3
number_of_TMBCPs_per_channel	b	64

The start addresses for each register in the Message Buffer component are depicted in **Table 30.16**.

Table 30.16 Message Buffer Component Register Start Addresses

b = Message Buffer Component Index	MBCP	Register	p	Start Address n = 0, 1
0 to number_of_RMBCPs_per_channel - 1	RMBCPb[i]	RMID	—	H’2000 + b × H’0080 + n × H’800
		RMPTR	—	H’2004 + b × H’0080 + n × H’800
		RMFDSTS	—	H’2008 + b × H’0080 + n × H’800
		RMDFp	0 to 15	H’200C + p ×H’0004 + b ×H’0080 + n ×H’800
0 to number_of_RFMBCPs - 1	RFMBCPb[i]	RFID	—	H’6000 + b × H’0080
		RFPTR	—	H’6004 + b × H’0080
		RFFDSTS	—	H’6008 + b × H’0080
		RFDFp	0 to 15	H’600C + p × H’0004 + b × H’0080
0 to number_of_CFMBCPs_per_channel - 1	CFMBCPb[i]	CFID	—	H’6400 + b × H’0080 + n × H’180
		CFPTR	—	H’6404 + b × H’0080 + n × H’180
		CFFDCSTS	—	H’6408 + b × H’0080 + n × H’180
		CFDFp	0 to 15	H’640C + p × H’0004 + b × H’0080 + n × H’180
0 to number_of_TMBCPs_per_channel - 1	TMBCPb[i]	TMID	—	H’1_0000 + b × H’0080 + n × H’2000
		TMPTR	—	H’1_0004 + b × H’0080 + n × H’2000
		TMFDCTR	—	H’1_0008 + b × H’0080 + n × H’2000
		TMDFp	0 to 15	H’1_000C + p × H’0004 + b × H’0080 + n × H’2000

The message buffer configuration consists of four types of Message Buffer components:

- RX Message Buffer Component (CFDRMBCPb[i])
- RX FIFO Access Message Buffer Component (CFDRFMBCPb[i])
- Common FIFO Access Message Buffer Component (CFDCFMBCPb[i])
- TX Message Buffer Component (CFDTMBCPb[i]).

Where b = the Message Buffer component index having a range that varies based on the type of Message Buffer component and i = channel index that has a range from 0 to n.

For a summary of this configuration, see **Figure 30.28, Message Buffer Configuration**. For a detailed description of the number of and the different types of message buffers, see **Section 30.6, FIFO Buffers and Normal Message Buffer Configuration**.

As described in **Section 30.2, Register Descriptions**, each Message Buffer component consists of the following registers:

- Identifier (ID)
- Pointer (PTR)
- Data Field (DFp).

p is the Data Field register index having a range that varies based on the type of Message Buffer component.

Rc is the Message Buffer Component register where c = Message Buffer Component register index having a range that varies based on the type of Message Buffer component.

A description of the registers, their associated bits and their accessibility are shown below the summary and detailed figures of each component.

In each of the figures, a cell that contains '-' means reserved and has the same behavior as reserved bits for registers in **Section 30.2 Register Descriptions**.

30.2.90.2 RX Message Buffer Component b (CFDRMBCPb[i])

Base address: CAN-FD = H'0_100C_0000

Offset address: RMID: H'2000 + H'0080 × b + H'800 × n
RMPTR: H'2004 + H'0080 × b + H'800 × n
RMFDSTS: H'2008 + H'0080 × b + H'800 × n
RMDFp: H'200C + H'0004 × p + H'080 × b + H'800 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Rc[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Rc[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Rc	All 0	R	RX Message Buffer Component c Refer to **Table 30.17**, **Table 30.18** and the descriptions that follow for a detailed description of each register and its related bits, contained within this message buffer component.

(no_of_channels = 2)

(i = 0 .. no_of_channels – 1)

(b = RX Message Buffer Component index = [0…no_of_CFDRMBCPs_per_channel-1])

(no_of_CFDRMBCPs_per_channel = No. of RX Message Buffer Components per Channel = 16)

Where the total number of CFDRMBCPs = no_of_CFDRMBCPs_per_channel × no_of_channels = 16 × 2 = 32 as shown in **Figure 30.28 Message Buffer Configuration**.

(c = RX Message Buffer Component Register index = [0…no_of_REGs_per_CFDRMBCP-1])

no_of_REGs_per_CFDRMBCP = No. of Registers per RX Message Buffer Component = 19

Rc bit (RX Message Buffer Component c)

The RX Message Buffer Component is made up of the following registers: **CFDRMID**, **CFDRMPTR**, **CFDRMFDSTS**, and **CFDRMDFp**. Refer to **Table 30.18, RX Message Buffer Component Detailed** for details of how to interpret the structure of this buffer component and how to access the respective registers.

Table 30.17 RX Message Buffer Component Summary

RX Message Buffer Component (RMBCP)	
Rc	Description
R0	RX Message Buffer ID Registers CHn
R1	RX Message Buffer Pointer Register CHn
R2	RX Message Buffer CAN-FD Status Registers CHn
R3	RX Message Buffer Data Filed 0 Registers CHn
R4	RX Message Buffer Data Filed 1 Registers CHn
R5	RX Message Buffer Data Filed 2 Registers CHn
R6	RX Message Buffer Data Filed 3 Registers CHn
R7	RX Message Buffer Data Filed 4 Registers CHn
R8	RX Message Buffer Data Filed 5 Registers CHn
R9	RX Message Buffer Data Filed 6 Registers CHn
R10	RX Message Buffer Data Filed 7 Registers CHn
R11	RX Message Buffer Data Filed 8 Registers CHn
R12	RX Message Buffer Data Filed 9 Registers CHn
R13	RX Message Buffer Data Filed 10 Registers CHn
R14	RX Message Buffer Data Filed 11 Registers CHn
R15	RX Message Buffer Data Filed 12 Registers CHn
R16	RX Message Buffer Data Filed 13 Registers CHn
R17	RX Message Buffer Data Filed 14 Registers CHn
R18	RX Message Buffer Data Filed 15 Registers CHn

Table 30.18 RX Message Buffer Component Detailed

RX Message Buffer Component (RMBCP) [31:16]																	
Rc	p	Symbol	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R0	X	CFDRMID	RMIDE	RMRTR	—	RMID[28:16]												
R1	x	CFDRMPTR	RMDLC				—	—	—	—	—	—	—	—	—	—	—	—
R2	x	CFDRMFDSTS	RMPTR[15:0]															
R3	0	CFDRMDFp	RMDB_HH[7:0]								RMDB_HL[7:0]							
R4 to R18	1 to 15	CFDRMDFp	RMDB_HH[7:0]								RMDB_HL[7:0]							

RX Message Buffer Component (RMBCP) [15:0]																		
Rc	p	Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R0	x	CFDRMID	RMID[15:0]															
R1	x	CFDRMPTR	RMTS[15:0]															
R2	x	CFDRMFDSTS	—	—	—	—	—	—	RMIFL		—	—	—	—	—	RMFDF	RMBRS	RMESI
R3	0	CFDRMDFp	RMDB_LH[7:0]								RMDB_LL[7:0]							
R4 to R18	1 to 15	CFDRMDFp	RMDB_LH[7:0]								RMDB_LL[7:0]							

30.2.90.3 RX Message Buffer ID Register n (CFDRMIDn) (n = 0 to 31)

The CFDRMIDn register (n = 0 to 31) stores the ID field, IDE bit, and RTR bit of the received message.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'2000 + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMIDE	RMRTR	—	RMID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RMIDE	0	R	RX Message Buffer IDE Bit 0: STD-ID is stored 1: EXT-ID is stored
30	RMRTR	0	R	RX Message Buffer RTR Bit 0: Data frame 1: Remote frame
29	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
28 to 0	RMID[28:0]	All 0	R	RX Message Buffer ID Field STD-ID/EXT-ID fields

RMID[28:0] bits (RX Message Buffer ID Field)

The RMID[28:0] are the bits of the STD-ID/EXT-ID fields of the message stored in the RX message buffer.

For alignment of these bits in standard and extended frame format, see **Section 30.2.89, Identifier Bits Alignment**.

See **Section 30.2.90.1, Start Addresses** for details on how to interpret the structure of this buffer component.

RMRTR bit (RX Message Buffer RTR Bit)

The RMRTR bit shows whether a data frame or a remote frame was stored in the RX message buffer.

Note: There are no remote frames in CAN-FD format. When a CAN-FD frame is received, the register reflects the state of the received value (the RRS bit in FD frame format).

RMIDE bit (RX Message Buffer IDE Bit)

The RMIDE bit shows whether message with Standard Identifier or Extended Identifier was stored in the RX message buffer.

30.2.90.4 RX Message Buffer Pointer Register n (CFDRMPTRn) (n = 0 to 31)

The CFDRMPTRn register (n = 0 to 31) stores the DLC and Timestamp fields for the received message.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'2004 + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDLC[3:0]				—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	RMDLC[3:0]	H'0	R	RX Message Buffer DLC Field Number of data bytes received in a CAN frame.
27 to 16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
15 to 0	RMTS[15:0]	H'0000	R	RX Message Buffer Timestamp Field Timestamp value stored for the message in the RX message buffer

RMTS[15:0] bits (RX Message Buffer Timestamp Field)

The RMTS[15:0] bits store the timestamp value taken at the capture point as configured by CFDGFDCFG.TSCCFG of the received message.

In the case of can_race_ts_en=1, the information as which CFDRMPTR.RMTS is inputted from can_race_ts is reflected. See **Section 30.8.1.3 Timestamp** for the timestamp in detail.

RMDLC[3:0] bits (RX Message Buffer DLC Field)

The RMDLC[3:0] bits store the number of data bytes that were received in the RX message buffer.

See *Table 5 in ISO 11898-1 (2015) Specification* for details in defining the number of data bytes that were received.

Note: The maximum capacity of the buffer belongs to CFDRMNB.RMPLS.

30.2.90.5 RX Message Buffer CAN-FD Status Register n (CFDRMFDSTSn) (n = 0 to 31)

The CFDRMFDSTSn register (n = 0 to 31) shows the status of the FDF, BRS, and ESI bits, and pointer of the received CAN-FD frame.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'2008 + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMPTR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RMIFL[1:0]		—	—	—	—	—	RMFDF	RMBRS	RMESI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	RMPTR[15:0]	H'0000	R	RX Message Buffer Pointer Field
15 to 10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
9, 8	RMIFL[1:0]	00	R	RX Message Buffer Information Label Field
7 to 3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
2	RMFDF	0	R	CAN-FD Format bit 0: Non CAN-FD frame received 1: CAN-FD frame received
1	RMBRS	0	R	Bit Rate Switch bit 0: CAN-FD frame received with no bit rate switch 1: CAN-FD frame received with bit rate switch
0	RMESI	0	R	Error State Indicator bit 0: CAN-FD frame received from error active node 1: CAN-FD frame received from error passive node

RMESI bit (Error State Indicator bit)

The RMESI bit has the same value as the ESI bit of the received CAN-FD frame.

When the received FDF bit is 0, this means a CAN2.0 frame is received and 0 is stored to this bit.

RMBRS bit (Bit Rate Switch bit)

The RMBRS bit has the same value as the BRS bit of the received CAN-FD frame.

When the received FDF bit is 0, this means a CAN2.0 frame is received and 0 is stored to this bit.

RMFDF bit (CAN-FD Format bit)

The RMFDF bit has the same value as the FDF bit of the received CAN-FD frame.

RMIFL[1:0] bits (RX Message Buffer Information Label Field)

The RMIFL[1:0] bits store the information label value from the related Global Acceptance Filter List entry.

RMPTR[15:0] bits (RX Message Buffer Pointer Field)

The RMPTR[15:0] bits store the pointer value from the related Global Acceptance Filter List entry.

30.2.90.6 RX Message Buffer Data Field p Register n (CFDRMDFp_n) (p = 0 to 15, n = 0 to 31)

The CFDRMDFp_n register (p = 0 to 15, n = 0 to 31) stores the data bytes (4 × p) to data bytes (4 × p + 3) of the received message.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'200C + H'004 × p + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB_HH[7:0]								RMDB_HL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB_LH[7:0]								RMDB_LL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	RMDB_HH [7:0]	H'00	R	RX Message Buffer Data Byte (4 × p + 3)
23 to 16	RMDB_HL [7:0]	H'00	R	RX Message Buffer Data Byte (4 × p + 2)
15 to 8	RMDB_LH [7:0]	H'00	R	RX Message Buffer Data Byte (4 × p + 1)
7 to 0	RMDB_LL [7:0]	H'00	R	RX Message Buffer Data Byte (4 × p)

RMDB_LL[7:0] bits (RX Message Buffer Data Byte (4 × p))

The RMDB_LL[7:0] bits store data bytes (4 × p) of the message in the RX message buffer. Unused data bytes are filled with H'00.

RMDB_LH[7:0] bits (RX Message Buffer Data Byte (4 × p + 1))

The RMDB_LH[7:0] bits store data bytes (4 × p + 1) of the message in the RX message buffer. Unused Data Bytes are filled with H'00.

RMDB_HL[7:0] bits (RX Message Buffer Data Byte (4 × p + 2))

The RMDB_HL[7:0] bits store data bytes (4 × p + 2) of the message in the RX message buffer. Unused data bytes are filled with H'00.

RMDB_HH[7:0] bits (RX Message Buffer Data Byte (4 × p + 3))

The RMDB_HH[7:0] bits store data bytes (4 × p + 3) of the message in the RX message buffer. Unused data bytes are filled with H'00.

30.2.90.7 RX FIFO Access Message Buffer Component b (CFDRFMBCPb[i])

Base address: CAN-FD = H'0_100C_0000

Offset address: RFID: H'6000 + H'0080 × b
RFPTR: H'6004 + H'0080 × b
RFFDSTS: H'6008 + H'0080 × b
RFDFp: H'600C + H'0004 × p + H'080 × b

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Rc[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Rc[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Rc	All 0	R	RX FIFO Access Message Buffer Component c Refer to **Table 30.19**, **Table 30.20** and the descriptions that follow for a detailed description of each register and its related bits, contained within this message buffer component.

(no_of_channels = 2)

(i = 0 .. no_of_channels – 1)

(b = RX FIFO Access Message Buffer Component index = [0…no_of_CFDRFMBCPs-1])

(no_of_CFDRFMBCPs = No. of RX FIFO Access Message Buffer Components = 8)

as shown in **Figure 30.28 Message Buffer Configuration**.

(c = RX FIFO Access Message Buffer Component Register index = [0…no_of_REGs_per_CFDRFMBCP-1])

no_of_REGs_per_CFDRFMBCP = No. of Registers per RX FIFO Access Message Buffer Component = 19

Rc bit (RX Message Buffer Component c)

The RX Message Buffer Component is made up of the following registers: **CFDRFID**, **CFDRFPTR**, **CFDRFFDSTS**, and **CFDRFDFp.** Refer to **Table 30.20, RX FIFO Access Message Buffer Component Detailed** for details of how to interpret the structure of this buffer component and how to access the respective registers.

Table 30.19 RX FIFO Access Message Buffer Component Summary

RX FIFO Access Message Buffer Component (RFMBCP)	
Rc	Description
R0	RX FIFO Access ID Registers
R1	RX FIFO Access Pointer Register
R2	RX FIFO Access CAN-FD Status Registers
R3	RX FIFO Access Data Filed 0 Registers
R4	RX FIFO Access Data Filed 1 Registers
R5	RX FIFO Access Data Filed 2 Registers
R6	RX FIFO Access Data Filed 3 Registers
R7	RX FIFO Access Data Filed 4 Registers
R8	RX FIFO Access Data Filed 5 Registers
R9	RX FIFO Access Data Filed 6 Registers
R10	RX FIFO Access Data Filed 7 Registers
R11	RX FIFO Access Data Filed 8 Registers
R12	RX FIFO Access Data Filed 9 Registers
R13	RX FIFO Access Data Filed 10 Registers
R14	RX FIFO Access Data Filed 11 Registers
R15	RX FIFO Access Data Filed 12 Registers
R16	RX FIFO Access Data Filed 13 Registers
R17	RX FIFO Access Data Filed 14 Registers
R18	RX FIFO Access Data Filed 15 Registers
R19 to R31	—

Table 30.20 RX FIFO Access Message Buffer Component Detailed

RX FIFO Access Message Buffer Component (RFMBCP) [31:16]																		
Rc	p	Symbol	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R0	X	CFDRFID	RFIDE	RFRTR	—	RFID[28:16]												
R1	x	CFDRFPTR	RFDLC				—	—	—	—	—	—	—	—	—	—	—	—
R2	x	CFDRFFDSTS	CFDRFPTR[15:0]															
R3	0	CFDRFDFp	RFDB_HH[7:0]								RFDB_HL[7:0]							
R4 to R18	1 to 15	CFDRFDFp	RFDB_HH[7:0]								RFDB_HL[7:0]							

RX FIFO Access Message Buffer Component (RFMBCP) [15:0]																		
Rc	p	Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R0	x	CFDRFID	RFID[15:0]															
R1	x	CFDRFPTR	RFTS[15:0]															
R2	x	CFDRFFDSTS	—	—	—	—	—	—	RFIFL		—	—	—	—	—	RFFDF	RFBRS	RFESI
R3	0	CFDRFDFp	RFDB_LH[7:0]								RFDB_LL[7:0]							
R4 to R18	1 to 15	CFDRFDFp	RFDB_LH[7:0]								RFDB_LL[7:0]							

30.2.90.8 RX FIFO Access ID Register n (CFDRFIDn) (n = 0 to 7)

The CFDRFIDn register (n = 0 to 7) stores the ID field, IDE bit, and RTR bit of the message.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'6000 + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFIDE	RFRTR	—	RFID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RFIDE	0	R	RX FIFO Buffer IDE bit 0: STD-ID has been received 1: EXT-ID has been received
30	RFRTR	0	R	RX FIFO Buffer RTR bit 0: Data frame 1: Remote frame
29	—	0	R/W	Reserved This bit is read as 0. The write value should be always 0.
28 to 0	RFID[28:0]	All 0	R	RX FIFO Buffer ID Field STD-ID/EXT-ID fields

RFID[28:0] bits (RX FIFO Buffer ID Field)

The RFID[28:0] bits are the bits of the STD-ID/EXT-ID fields of the message in the FIFO buffer.

For alignment of these bits in standard and extended frame format, see **Section 30.2.89, Identifier Bits Alignment**.

RFRTR bit (RX FIFO Buffer RTR bit)

The RFRTR bit shows whether a data frame or a remote frame was stored in the FIFO buffer.

Note: There are no remote frames in CAN-FD format. When a CAN-FD frame was received, the register reflects the state of the received value (RRS bit in FD frame format).

RFIDE bit (RX FIFO Buffer IDE bit)

The RFIDE bit shows whether message with the Standard Identifier or Extended Identifier was received in the FIFO buffer.

30.2.90.9 RX FIFO Access Pointer Register n (CFDRFPTRn) (n = 0 to 7)

The CFDRFPTRn register (n = 0 to 7) stores the DLC and Timestamp fields for the received message.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'6004 + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDLC[3:0]				—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	RFDLC[3:0]	H'0	R	RX FIFO Buffer DLC Field Number of data bytes received in a CAN frame
27 to 16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
15 to 0	RFTS[15:0]	H'0000	R	RX FIFO Timestamp Value Timestamp value of the received CAN frame

RFTS[15:0] bits (RX FIFO Timestamp Value)

The RFTS[15:0] bits store the timestamp value taken at the capture point as configured by the CFDGFDCFG.TSCCFG bit of the received message.

In the case of can_race_ts_en=1, the information as which CFDRFPTR.RFTS is inputted from can_race_ts is reflected.

See **Section 30.8.1.3 Timestamp** for the timestamp in detail.

RFDLC[3:0] bits (RX FIFO Buffer DLC Field)

The RFDLC[3:0] bits store the number of data bytes that were received in the RX FIFO buffer.

See *Table 5 in ISO 11898-1 (2015) Specification* for details in defining the number of data bytes that were received.

30.2.90.10 RX FIFO Access CAN-FD Status Register n (CFDRFFDSTSn) (n = 0 to 7)

The CFDRFFDSTSn register (n = 0 to 7) shows the status of the FDF, BRS, and ESI bits, including the pointer of the received CAN-FD frame.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'6008 + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDRFPTR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RFIFL[1:0]		—	—	—	—	—	RFFDF	RFBRS	RFESI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	CFDRFPTR [15:0]	H'0000	R	RX FIFO Buffer Pointer Field
15 to 10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
9, 8	RFIFL[1:0]	00	R	RX FIFO Buffer Information Label Field
7 to 3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
2	RFFDF	0	R	CAN-FD Format bit 0: Non CAN-FD frame received 1: CAN-FD frame received
1	RFBRS	0	R	Bit Rate Switch bit 0: CAN-FD frame received with no bit rate switch 1: CAN-FD frame received with bit rate switch
0	RFESI	0	R	Error State Indicator bit 0: CAN-FD frame received from error active node 1: CAN-FD frame received from error passive node

RFESI bit (Error State Indicator bit)

The RFESI bit has the same value as the ESI bit of the received CAN-FD frame.

When the received FDF bit is 0, this means a CAN2.0 frame is received and 0 is stored to this bit.

RFBRS bit (Bit Rate Switch bit)

The RFBRS bit has the same value as the BRS bit of the received CAN-FD frame.

When the received FDF bit is 0, this means a CAN2.0 frame is received and 0 is stored to this bit.

RFFDF bit (CAN-FD Format bit)

The RFFDF bit has the same value as the FDF bit of the received CAN-FD frame.

RFIFL[1:0] bits (RX FIFO Buffer Information Label Field)

The RFIFL[1:0] bits store the information label value from the related Global Acceptance Filter List entry.

CFDRFPTR[15:0] bits (RX FIFO Buffer Pointer Field)

The CFDRFPTR[15:0] bits store the pointer value from the related Global Acceptance Filter List entry.

30.2.90.11 RX FIFO Access Data Field p Register n (CFDRFDFp_n) (p = 0 to 15, n = 0 to 7)

The CFDRFDFpn register (p = 0 to 15, n = 0 to 7) stores data bytes (4 × p) to data bytes (4 × p + 3) of the received message.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'600C + H'004 × p + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB_HH[7:0]								RFDB_HL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB_LH[7:0]								RFDB_LL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	RFDB_HH [7:0]	H'00	R	RX FIFO Buffer Data Byte (4 × p + 3)
23 to 16	RFDB_HL [7:0]	H'00	R	RX FIFO Buffer Data Byte (4 × p + 2)
15 to 8	RFDB_LH [7:0]	H'00	R	RX FIFO Buffer Data Byte (4 × p + 1)
7 to 0	RFDB_LL[7:0]	H'00	R	RX FIFO Buffer Data Byte (4 × p)

RFDB_LL[7:0] bits (RX FIFO Buffer Data Byte (4 × p))

The RFDB_LL[7:0] bits store data bytes (4 × p) of the message present in the FIFO buffer. Unused data bytes are filled with H'00 according to the configured data payload size CFDRFCCn.RFPLS.

RFDB_LH[7:0] bits (RX FIFO Buffer Data Byte (4 × p + 1))

The RFDB_LH[7:0] bits store data bytes (4 × p + 1) of the message present in the FIFO buffer. Unused data bytes are filled with H'00.

RFDB_HL[7:0] bits (RX FIFO Buffer Data Byte (4 × p + 2))

The RFDB_HL[7:0] bits store data bytes (4 × p + 2) of the message present in the FIFO buffer. Unused data bytes are filled with H'00.

RFDB_HH[7:0] bits (RX FIFO Buffer Data Byte (4 × p + 3))

The RFDB_HH[7:0] bits store data bytes (4 × p + 3) of the message present in the FIFO buffer. Unused data bytes are filled with H'00.

30.2.90.12 Common FIFO Access Message Buffer Component b (CFDCFMBCPb[i])

Base address: CAN-FD = H'0_100C_0000

Offset address: CFID: H'6400 + H'0080 × b + H'180 × n
CFPTR: H'6404 + H'0080 × b + H'180 × n
CFFDSTS: H'6408 + H'0080 × b + H'180 × n
CFDFp: H'640C + H'0004 × p + H'080 × b + H'180 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Rc[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Rc[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Rc	All 0	R	RX Message Buffer Component c Refer to **Table 30.21**, **Table 30.22** and the descriptions that follow for a detailed description of each register and its related bits, contained within this message buffer component.

(no_of_channels = 2)

(i = 0 .. no_of_channels – 1)

(b = Common FIFO Access Message Buffer Component index = [0…no_of_CFDCFMBCPs_per_channel-1])

(no_of_CFDCFMBCPs_per_channel = No. of Common FIFO Access Message Buffer Components per Channel = 3)
Where the total number of CFDCFMBCPs = no_of_CFDCFMBCPs_per_chennel × no_of_channels = 3 × 2 = 6 as shown in **Figure 30.28 Message Buffer Configuration**.

(c = Common FIFO Access Message Buffer Component Register index = [0…no_of_REGs_per_CFDCFMBCP-1])

no_of_REGs_per_CFDCFMBCP = No. of Registers per Common FIFO Access Message Buffer Component = 19

Rc bit (Common FIFO Access Message Buffer Component c)

The Common FIFO Access Message Buffer Component is made up of the following registers: **CFDCFID**, **CFDCFPTR, CFDCFFDCSTS**, and **CFDCFDFp**. Refer to **Table 30.22, Common FIFO Access Message Buffer Component Detailed** for details of how to interpret the structure of this buffer component and how to access the respective registers.

Table 30.21 Common FIFO Access Message Buffer Component Summary

Common FIFO Access Message Buffer Component (RFMBCP)	
Rc	Description
R0	Common FIFO Access ID Registers
R1	Common FIFO Access Pointer Register
R2	Common FIFO Access CAN-FD Status Registers
R3	Common FIFO Access Data Filed 0 Registers
R4	Common FIFO Access Data Filed 1 Registers
R5	Common FIFO Access Data Filed 2 Registers
R6	Common FIFO Access Data Filed 3 Registers
R7	Common FIFO Access Data Filed 4 Registers
R8	Common FIFO Access Data Filed 5 Registers
R9	Common FIFO Access Data Filed 6 Registers
R10	Common FIFO Access Data Filed 7 Registers
R11	Common FIFO Access Data Filed 8 Registers
R12	Common FIFO Access Data Filed 9 Registers
R13	Common FIFO Access Data Filed 10 Registers
R14	Common FIFO Access Data Filed 11 Registers
R15	Common FIFO Access Data Filed 12 Registers
R16	Common FIFO Access Data Filed 13 Registers
R17	Common FIFO Access Data Filed 14 Registers
R18	Common FIFO Access Data Filed 15 Registers
R19 to R31	—

Table 30.22 Common FIFO Access Message Buffer Component Detailed

Common FIFO Access Message Buffer Component (CFMBCP) [31:16]																		
Rc	p	Symbol	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R0	X	CFDCFID	CFIDE	CFRTR	THLEN	CFID[28:16]												
R1	x	CFDCFPTR	CFDLC				—	—	—	—	—	—	—	—	—	—	—	—
R2	x	CFDCFFDCSTS	CFPTR[15:0]															
R3	0	CFDCFDFp	CFDB_HH[7:0]								CFDB_HL[7:0]							
R4 to R18	1 to 15	CFDCFDFp	CFDB_HH[7:0]								CFDB_HL[7:0]							

Common FIFO Access Message Buffer Component (CFMBCP) [15:0]																		
Rc	p	Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R0	x	CFDCFID	CFID[15:0]															
R1	x	CFDCFPTR	CFTS[15:0]															
R2	x	CFDCFFDCSTS	—	—	—	—	—	—	CFIFL		—	—	—	—	—	CFFDF	CFBRS	CFESI
R3	0	CFDCFDFp	CFDB_LH[7:0]								CFDB_LL[7:0]							
R4 to R18	1 to 15	CFDCFDFp	CFDB_LH[7:0]								CFDB_LL[7:0]							

30.2.90.13 Common FIFO Access ID Register n (CFDCFIDn) (n = 0 to 5)

The CFDCFIDn register (n = 0 to 5) stores the ID field, IDE bit, and RTR bit of the message.

In TX mode, you can read data from the FIFO, only for the current entry based on the write pointer value, and not for the other entries.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'6400 + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFIDE	CFRTR	THLEN	CFID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	CFIDE	0	R/W	Common FIFO Buffer IDE bit 0: STD-ID is to be transmitted or has been received 1: EXT-ID is to be transmitted or has been received
30	CFRTR	0	R/W	Common FIFO Buffer RTR bit 0: Data frame 1: Remote frame
29	THLEN	0	R/W	THL Entry Enable TX FIFO mode: 0: Entry is not to be stored in THL after successful TX 1: Entry is to be stored in THL after successful TX RX FIFO mode: Reserved, this bit is read as 0.
28 to 0	CFID[28:0]	All 0	R/W	Common FIFO Buffer ID Field STD-ID/EXT-ID fields

CFID[28:0] bits (Common FIFO Buffer ID Field)

The CFID[28:0] bits are the bits of the STD-ID/EXT-ID fields of the message in the FIFO buffer.

For alignment of these bits in standard and extended frame format, see **Section 30.2.89, Identifier Bits Alignment**.

In TX mode, you can write and read data from FIFO buffers.

In RX mode, you can only read data from FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

THLEN bit (THL Entry Enable)

The THLEN bit controls the storage of an entry corresponding to the transmitted message in the TX History list at the end of a successful transmission.

In TX mode, you can write and read data from FIFO buffers.

In RX mode, you can only read data from FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

CFRTR bit (Common FIFO Buffer RTR bit)

The CFRTR bit selects whether a data frame or a remote frame is to be transmitted from or was received in the FIFO buffer.

Note: There are no remote frames in CAN-FD format. When a CAN-FD frame is received (RX mode), the register reflects the state of the received value (RRS bit in FD frame format). When CAN-FD transmission (TX or GW mode CFDCFID.CFFDF = 1), the bit is always transmitted dominant (data frame).

In TX mode, you can write and read data from FIFO buffers.

In RX mode, you can only read data from FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

CFIDE bit (Common FIFO Buffer IDE bit)

The CFIDE bit selects whether a message with EXT-ID or STD-ID is to be transmitted from or was received in the FIFO buffer.

In TX mode, you can write and read data from FIFO buffers.

In RX mode, you can only read data from FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

30.2.90.14 Common FIFO Access Pointer Register n (CFDCFPTRn) (n = 0 to 5)

The CFDCFPTRn register (n = 0 to 5) stores the DLC and Timestamp fields.

In TX mode, you can read data from the FIFO buffer, only for the current entry based on the write pointer value, and not for the other entries.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'6404 + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDLC[3:0]				—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	CFDLC[3:0]	H'0	R/W	Common FIFO Buffer DLC Field Number of data bytes received in a CAN frame, or transmitted in a CAN frame.
27 to 16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
15 to 0	CFTS[15:0]	H'0000	R/W	Common FIFO Timestamp Value Timestamp value of the received CAN frame (FIFO in RX mode).

CFTS[15:0] bits (Common FIFO Timestamp Value)

The CFTS[15:0] bits store the timestamp value taken at the capture point as configured by the CFDGFDCFG.TSCCFG bit of the received message (if FIFO is configured in RX mode).

In TX mode, you can read and write data from FIFO buffers.

In RX mode, you can only read data from FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

In the case of can_race_ts_en=1, the information as which CFDCFPTR.CFTS is inputted from can_race_ts

is reflected. See **Section 30.8.1.3 Timestamp** for the timestamp in detail.

CFDLC[3:0] bits (Common FIFO Buffer DLC Field)

The CFDLC[3:0] bits store the number of data bytes that were received in the FIFO buffer or are to be transmitted.

See *Table 5 in ISO 11898-1 (2015) Specification* for details in defining the number of data bytes.

In TX mode, you can read and write data from the FIFO buffers. Do not read data for the other entries in the FIFO when configured in TX mode.

In RX mode, you can only read data from the FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

30.2.90.15 Common FIFO Access CAN-FD Control/Status Register n (CFDCFFDCSTSn) (n = 0 to 5)

The CFDCFFDCSTSn register (n = 0 to 5) shows the status of the FDF, BRS, and ESI bits, including the pointer of the received CAN-FD frame or the CAN-FD frame to transmit.

In TX mode, you can read data from the FIFO, only for the current entry based on the write pointer value, and not for the other entries.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'6408 + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFPTR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CFIFL[1:0]		—	—	—	—	—	CFFDF	CFBRS	CFESI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	CFPTR[15:0]	H'0000	R/W	Common FIFO Buffer Pointer Field
15 to 10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
9, 8	CFIFL[1:0]	00	R/W	COMMON FIFO Buffer Information Label Field
7 to 3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
2	CFFDF	0	R/W	CAN-FD Format bit 0: Non CAN-FD frame received or to transmit 1: CAN-FD frame received or to transmit
1	CFBRS	0	R/W	Bit Rate Switch bit 0: CAN-FD frame received or to transmit with no bit rate switch 1: CAN-FD frame received or to transmit with bit rate switch
0	CFESI	0	R/W	Error State Indicator bit 0: CAN-FD frame received or to transmit by error active node 1: CAN-FD frame received or to transmit by error passive node

CFESI bit (Error State Indicator bit)

In TX mode, you can read and write data from FIFO buffers. In this mode, when the CAN-FD module is not in error passive, the CFESI bit equals the write value. Otherwise, it is a don't care and the bit is transmitted as 1 on the CAN bus, indicating this is an error passive node.

In RX mode, you can only read data from FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

In RX or GW mode, the CFESI bit is updated with the ESI bit value of the CAN-FD frame when it has been received, indicating the error state of the transmitting node. In RX or GW mode, 0 is stored to this bit when the received FDF bit is 0, this means a CAN 2.0 frame is received.

CFBRS bit (Bit Rate Switch bit)

In TX mode, you can read and write data from FIFO buffers. In this mode, the CAN-FD module either transmits a 0 to indicate no bit rate switch in the frame to be transmitted or a 1 to indicate a bit rate switch in the frame to be transmitted.

In RX mode, you can only read data from FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

In RX or GW mode, the CFBRS bit is updated with the BRS bit value of the CAN-FD frame when it has been received, indicating whether there is a bit rate switch (1) or (0) on the CAN-FD frame.

In RX or GW mode, 0 is stored to the CFBRS bit when the received FDF bit is 0, this means a CAN 2.0 frame is received.

CFFDF bit (CAN-FD Format bit)

In TX mode, you can read and write data from FIFO buffers. In this mode, the CAN-FD module either transmits a 0 to indicate a CAN 2.0 frame is to be transmitted or a 1 to indicate a CAN-FD frame is to be transmitted.

In RX mode, you can only read data from FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

In RX or GW mode, the CFFDF bit is updated with the FDF bit value of the CAN frame when it has been received, indicating whether it is a CAN 2.0 frame (0) or a CAN-FD frame (1).

CFIFL[1:0] bits (COMMON FIFO Buffer Information Label Field)

If the Common FIFO is configured in TX mode, the value programmed in CFDCFFDCSTSn.CFIFL[1:0] is stored together with additional message information, to the TX History List after successful transmission of the message.

The information label value from the related Global Acceptance Filter List entry is stored in these bits (if FIFO is configured in either RX or GW mode).

In TX mode, you can read and write data from FIFO buffers.

In RX mode, you can only read data from FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

CFPTR[15:0] bits (Common FIFO Buffer Pointer Field)

If the Common FIFO is configured in TX mode, the value programmed in CFDCFFDCSTSn.CFPTR[15:0] is stored together with additional message information, to the TX History List after successful transmission of the message.

The pointer value from the related Global Acceptance Filter List entry is stored in these bits (if FIFO is configured in either RX or GW mode).

In TX mode, you can read and write data from FIFO buffers.

In RX mode, you can only read data from FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

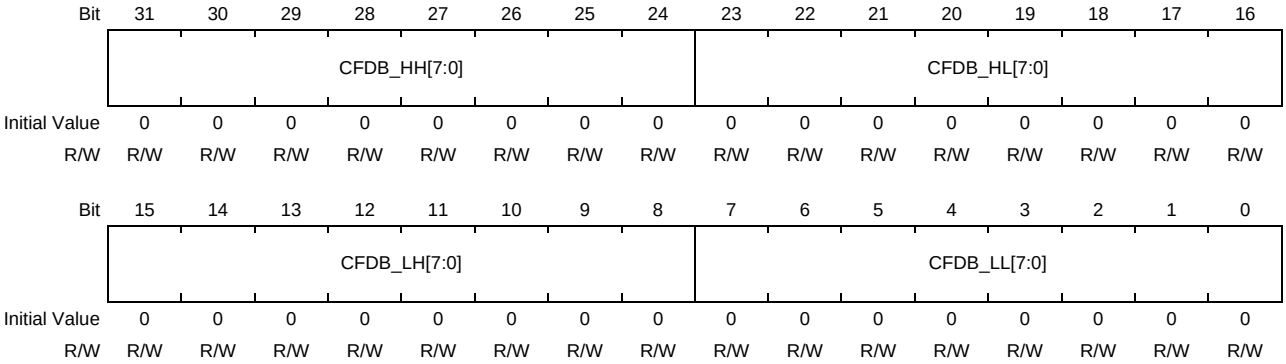
30.2.90.16 Common FIFO Access Data Field p Register n (CFDCFDFp_n) (p = 0 to 15, n = 0 to 5)

The CFDCFDFpn register (p = 0 to 15, n = 0 to 5) stores data bytes (4 × p) to data bytes (4 × p + 3) of the message.

In TX mode, you can read data from the FIFO, only for the current entry based on the write pointer value, and not for the other entries.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'640C + H'004 × p + H'080 × n



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CFDB_HH [7:0]	H'00	R/W	Common FIFO Buffer Data Byte (4 × p + 3)
23 to 16	CFDB_HL [7:0]	H'00	R/W	Common FIFO Buffer Data Byte (4 × p + 2)
15 to 8	CFDB_LH [7:0]	H'00	R/W	Common FIFO Buffer Data Byte (4 × p + 1)
7 to 0	CFDB_LL[7:0]	H'00	R/W	Common FIFO Buffer Data Byte (4 × p)

CFDB_LL[7:0] bits (Common FIFO Buffer Data Byte (4 × p))

The CFDB_LL[7:0] bits store data bytes (4 × p) of the message present in the FIFO buffer.

In TX mode, you can read and write data from the FIFO buffers.

In RX mode, you can only read data from the FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

In RX or GW mode, unused data bytes are filled with H'00, according to their configured data payload size CFDCFCCn.CFPLS.

CFDB_LH[7:0] bits (Common FIFO Buffer Data Byte (4 × p + 1))

The CFDB_LH[7:0] bits store data bytes (4 × p + 1) of the message present in the FIFO buffer.

In TX mode, you can read and write data from the FIFO buffers.

In RX mode, you can only read data from the FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

In RX or GW mode, unused data bytes are filled with H'00, according to their configured data payload size CFDCFCCn.CFPLS.

CFDB_HL[7:0] bits (Common FIFO Buffer Data Byte (4 × p + 2))

The CFDB_HL[7:0] bits store data bytes (4 × p + 2) of the message present in the FIFO buffer.

In TX mode, you can read and write data from the FIFO buffers.

In RX mode, you can only read data from the FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

In RX or GW mode, unused data bytes are filled with H’00, according to their configured data payload size CFDCFCCn.CFPLS.

CFDB_HH[7:0] bits (Common FIFO Buffer Data Byte (4 × p + 3))

The CFDB_HH[7:0] bits store data bytes (4 × p + 3) of the message present in the FIFO buffer.

In TX mode, you can read and write data from the FIFO buffers.

In RX mode, you can only read data from the FIFO buffers.

In GW mode, Cannot write data to the FIFO buffers.

In RX or GW mode, unused data bytes are filled with H’00, according to their configured data payload size CFDCFCCn.CFPLS.

30.2.90.17 TX Message Buffer Component b (CFDTMBCPb[i])

Base address: CAN-FD = H'0_100C_0000

Offset address: TMID: H'1_0000 + H'0080 × b + H'2000 × n
TMPTR: H'1_0004 + H'0080 × b + H'2000 × n
TMFDSTS: H'1_0008 + H'0080 × b + H'2000 × n
TMDFp: H'1_000C + H'0004 × p + H'0080 × b + H'2000 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Rc[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Rc[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31:0	Rc	All 0	R	TX Message Buffer Component c Refer to **Table 30.23**, **Table 30.24** and the descriptions that follow for a detailed description of each register and its related bits, contained within this message buffer component.

(no_of_channels = 2)

(i = 0 .. no_of_channels – 1)

(b = TX Message Buffer Component index = [0…no_of_CFDTMBCPs_per_channel-1])

(no_of_CFDTMBCPs_per_channel = No. of TX Message Buffer Components per Channel = 64)

Where the total number of CFDTMBCPs = no_of_CFDTMBCPs_per_channel × no_of_channels = 64 × 2 = 128 as shown in **Figure 30.28 Message Buffer Configuration**.

(c = TX Message Buffer Component Register index = [0…no_of_REGs_per_CFDTMBCP-1])

no_of_REGs_per_CFDTMBCP = No. of Registers per TX Message Buffer Component = 19

Rc bit (TX Message Buffer Component c)

The TX Message Buffer Component is made up of the following registers: **CFDTMID**, **CFDTMPTR**, **CFDTMFDCTR**, and **CFDTMDFp**. Refer to **Table 30.24, TX Message Buffer Component Detailed** for details of how to interpret the structure of this buffer component and how to access the respective registers.

Table 30.23 TX Message Buffer Component Summary

TX Message Buffer Component (RMBCP)	
Rc	Description
R0	TX Message Buffer ID Registers CHn
R1	TX Message Buffer Pointer Register CHn
R2	TX Message Buffer CAN-FD Status Registers CHn
R3	TX Message Buffer Data Filed 0 Registers CHn
R4	TX Message Buffer Data Filed 1 Registers CHn
R5	TX Message Buffer Data Filed 2 Registers CHn
R6	TX Message Buffer Data Filed 3 Registers CHn
R7	TX Message Buffer Data Filed 4 Registers CHn
R8	TX Message Buffer Data Filed 5 Registers CHn
R9	TX Message Buffer Data Filed 6 Registers CHn
R10	TX Message Buffer Data Filed 7 Registers CHn
R11	TX Message Buffer Data Filed 8 Registers CHn
R12	TX Message Buffer Data Filed 9 Registers CHn
R13	TX Message Buffer Data Filed 10 Registers CHn
R14	TX Message Buffer Data Filed 11 Registers CHn
R15	TX Message Buffer Data Filed 12 Registers CHn
R16	TX Message Buffer Data Filed 13 Registers CHn
R17	TX Message Buffer Data Filed 14 Registers CHn
R18	TX Message Buffer Data Filed 15 Registers CHn

Table 30.24 TX Message Buffer Component Detailed

TX Message Buffer Component (TMBCP) [31:16]																		
Rc	p	Symbol	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R0	X	CFDTMID	TMIDE	TMRTR	THLEN	TMID[28:16]												
R1	x	CFDTMPTR	TMDLC				—	—	—	—	—	—	—	—	—	—	—	—
R2	x	CFDTMFDCTR	TMPTR[15:0]															
R3	0	CFDTMDFp	TMDB_HH[7:0]								TMDB_HL[7:0]							
R4 to R18	1 to 15	CFDTMDFp	TMDB_HH[7:0]								TMDB_HL[7:0]							

TX Message Buffer Component (TMBCP) [15:0]																		
Rc	p	Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R0	x	CFDTMID	TMID[15:0]															
R1	x	CFDTMPTR	MTS[15:0]															
R2	x	CFDTMFDCTR	—	—	—	—	—	—	RMIFL		—	—	—	—	—	TMFDF	TMBRS	TMESI
R3	0	CFDTMDFp	TMDB_LH[7:0]								TMDB_LL[7:0]							
R4 to R18	1 to 15	CFDTMDFp	TMDB_LH[7:0]								TMDB_LL[7:0]							

30.2.90.18 TX Message Buffer ID Register n (CFDTMIDn) (n = 0 to 127)

The CFDTMIDn register (n = 0 to 127) stores the ID, IDE, RTR fields and history configuration of the message to be transmitted from the associated buffer.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1_0000 + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIDE	TMRTR	THLEN	TMID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	TMIDE	0	R/W	TX Message Buffer IDE bit 0: STD-ID is transmitted 1: EXT-ID is transmitted
30	TMRTR	0	R/W	TX Message Buffer RTR bit 0: Data frame 1: Remote frame
29	THLEN	0	R/W	Tx History List Entry 0: Entry not stored in THL after successful TX 1: Entry stored in THL after successful TX
28:0	TMID[28:0]	All 0	R/W	TX Message Buffer ID Field STD-ID/EXT-ID fields

TMID[28:0] bits (TX Message Buffer ID Field)

The TMID[28:0] bits are bits of the STD-ID/EXT-ID fields of the message stored in this TX message buffer.

For alignment of these bits in standard and extended frame format, see **Section 30.2.89, Identifier Bits Alignment**.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

THLEN bit (Tx History List Entry)

The THLEN bit controls the storage of an entry corresponding to the transmitted message in the TX History list at the end of a successful transmission.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

TMRTR bit (TX Message Buffer RTR bit)

The TMRTR bit selects whether a data frame or remote frame is to be transmitted from this TX message buffer.

Note: There are no remote frames in CAN-FD format. For a CAN-FD transmission (CFDTMFDCTR.CFFDF = 1), this bit is always transmitted dominant (data frame).

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

TMIDE bit (TX Message Buffer IDE bit)

The TMIDE bit selects whether a message with EXT-ID or STD-ID is to be transmitted from this TX message buffer.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

30.2.90.19 TX Message Buffer Pointer Register n (CFDTMPTRn) (n = 0 to 127)

The CFDTMPTRn register (n = 0 to 127) stores the DLC fields of the message to transmit from the associated buffer.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1_0004 + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDLC[3:0]				—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	TMDLC[3:0]	H'0	R/W	TX Message Buffer DLC Field Number of data bytes to be transmitted in a CAN frame.
27 to 0	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.

TMDLC[3:0] bits (TX Message Buffer DLC Field)

The TMDLC[3:0] bits select the number of data bytes to be transmitted from this TX message buffer when the corresponding TMRTR bit is configured as 0.

See *Table 5 in ISO 11898-1 (2015) Specification* for details in defining the number of data bytes to be transmitted.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

30.2.90.20 TX Message Buffer CAN-FD Control Register n (CFDTMFDCTRn) (n = 0 to 127)

The CFDTMFDCTRn register (n = 0 to 127) shows the status of the FDF, BRS, and ESI bits, including the pointer fields of the CAN-FD frame to be transmitted.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1_0008 + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMPTR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TMIFL[1:0]		—	—	—	—	—	TMFDF	TMBRS	TMESI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMPTR[15:0]	H'0000	R/W	TX Message Buffer Pointer Field
15 to 10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
9, 8	TMIFL[1:0]	00	R/W	TX Message Buffer Information Label Field
7 to 3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be always 0.
2	TMFDF	0	R/W	CAN-FD Format bit 0: Non CAN-FD frame to transmit 1: CAN-FD frame to transmit
1	TMBRS	0	R/W	Bit Rate Switch bit 0: CAN-FD frame to transmit with no bit rate switch 1: CAN-FD frame to transmit with bit rate switch
0	TMESI	0	R/W	Error State Indicator bit 0: CAN-FD frame to transmit by error active node 1: CAN-FD frame to transmit by error passive node

TMESI bit (Error State Indicator bit)

If the channel is not in error passive, then the TMESI bit equals the write value, otherwise it is a don't care and the bit is transmitted as 1 on the CAN bus, indicating this is an error passive node.

Do not write to the TMESI bit when the related CAN-FD channel is in CH_SLEEP mode.

TMBRS bit (Bit Rate Switch bit)

Do not write to the TMBRS bit when the related CAN-FD channel is in CH_SLEEP mode.

TMFDF bit (CAN-FD Format bit)

Do not write to the TMFDF bit when the related CAN-FD channel is in CH_SLEEP mode.

TMIFL[1:0] bits (TX Message Buffer Information Label Field)

The TMIFL[1:0] bits store the information label value to be copied, together with additional message information, in the TX History List after successful transmission of the message.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

TMPTR[15:0] bits (TX Message Buffer Pointer Field)

The TMPTR[15:0] bits store the pointer value to be copied, together with additional message information in the TX History List after successful transmission of the message.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

30.2.90.21 TX Message Buffer Data Field p Register n (CFDTMDFp_n) (p = 0 to 15, n = 0 to 127)

The CFDTMDFp_n register (p = 0 to 15, n = 0 to 127) stores data bytes (4 × p) to data bytes (4 × p + 3) of the message to transmit from the associated buffer.

Base address: CAN-FD = H'0_100C_0000

Offset address: H'1_000C + H'004 × p + H'080 × n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB_HH[7:0]								TMDB_HL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB_LH[7:0]								TMDB_LL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TMDB_HH [7:0]	H'00	R/W	TX Message Buffer Data Byte (4 × p + 3)
23 to 16	TMDB_HL [7:0]	H'00	R/W	TX Message Buffer Data Byte (4 × p + 2)
15 to 8	TMDB_LH [7:0]	H'00	R/W	TX Message Buffer Data Byte (4 × p + 1)
7 to 0	TMDB_LL [7:0]	H'00	R/W	TX Message Buffer Data Byte (4 × p)

TMDB_LL[7:0] bits (TX Message Buffer Data Byte (4 × p))

Data bytes (4 × p)[7:0] of the message stored in the TX message buffer.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

TMDB_LH[7:0] bits (TX Message Buffer Data Byte (4 × p + 1))

Data bytes (4 × p + 1)[7:0] of the message stored in the TX message buffer.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

TMDB_HL[7:0] bits (TX Message Buffer Data Byte (4 × p + 2))

Data bytes (4 × p + 2)[7:0] of the message stored in the TX message buffer.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

TMDB_HH[7:0] bits (TX Message Buffer Data Byte (4 × p + 3))

Data bytes (4 × p + 3)[7:0] of the message stored in the TX message buffer.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

30.3 Operation

30.3.1 Overview

The modes of the CAN-FD module can be classified into 2 groups:

- Global modes
- Channel modes

30.3.2 Global Modes

Global modes are applicable for the complete CAN-FD module. The global modes of the CAN-FD module are:

- Global Sleep
- Global Reset
- Global Halt
- Global Operation

Figure 30.2 shows the possible transitions between the Global modes.

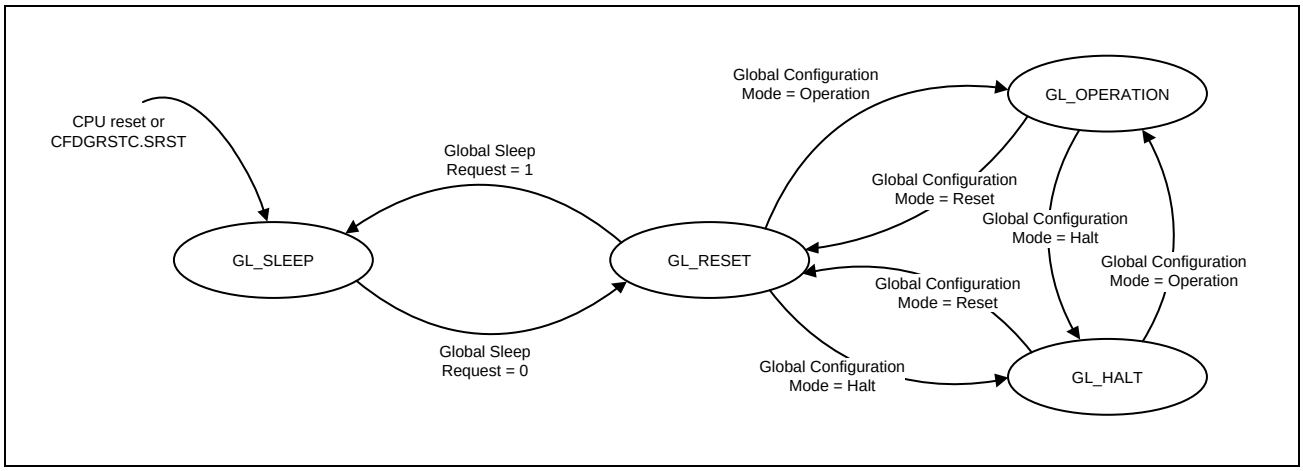


Figure 30.2 Transition between CAN-FD Global Modes

Changes in the Global mode can affect the Channel mode. **Table 30.25** shows the effect of a Global mode transition on a Channel mode.

Table 30.25 Possible CAN-FD Channel Modes and Global Modes

	Target Global Mode			
Current Global Mode	Sleep	Reset	Halt	Operation
Sleep	—	Ch-Sleep: Keep Ch-Reset: N/A Ch-Halt: N/A Ch-Oper: N/A	—	—
Reset	Ch-Sleep: Keep Ch-Reset: → Ch-Sleep Ch-Halt: N/A Ch-Oper: N/A	—	Ch-Sleep: Keep Ch-Reset: Keep Ch-Halt: N/A Ch-Oper: N/A	Ch-Sleep: Keep Ch-Reset: Keep Ch-Halt: N/A Ch-Oper: N/A
Halt	—	Ch-Sleep: Keep Ch-Reset: Keep Ch-Halt: → Ch-Reset Ch-Oper: N/A	—	Ch-Sleep: Keep Ch-Reset: Keep Ch-Halt: Keep Ch-Oper: N/A
Operation	—	Ch-Sleep: Keep Ch-Reset: Keep Ch-Halt: → Ch-Reset Ch-Oper: → Ch-Reset	Ch-Sleep: Keep Ch-Reset: Keep Ch-Halt: Keep Ch-Oper: → Ch-Halt	—

30.3.2.1 Global Sleep Mode

After the release of a hardware reset or after setting and clearing the CFDGRSTC.SRST bit, the CAN-FD module automatically enters Global Sleep mode.

The CAN-FD module also enters the Global Sleep mode when the Global Sleep Request bit is set while it is in Global Reset mode. This control bit cannot be set in Global Halt mode or Global Operation mode.

Setting the Global Sleep Request bit sets all Channel Sleep Request bits and forces all channels into the Channel Sleep mode. Sleep mode is used for power saving purpose. When CAN-FD module is in Global Sleep mode, only the clock for CPU write access to the Global Sleep Mode Request bit is active. All other clocks are stopped and all other functions of the CAN-FD module are suspended.

Read access from all registers is still possible and all register values are preserved.

In Global Sleep mode, RAM access is prohibited because the logic which generates a RAM address does not operate.

After setting the Global Sleep Request bit, it is necessary to confirm that the Global Sleep status has been updated, indicating successful transition to Global Sleep mode before the Global Sleep Request bit can be cleared again.

Figure 30.3 shows the procedure for entering Global Sleep mode and **Figure 30.4** shows the procedure for exiting Global Sleep mode.

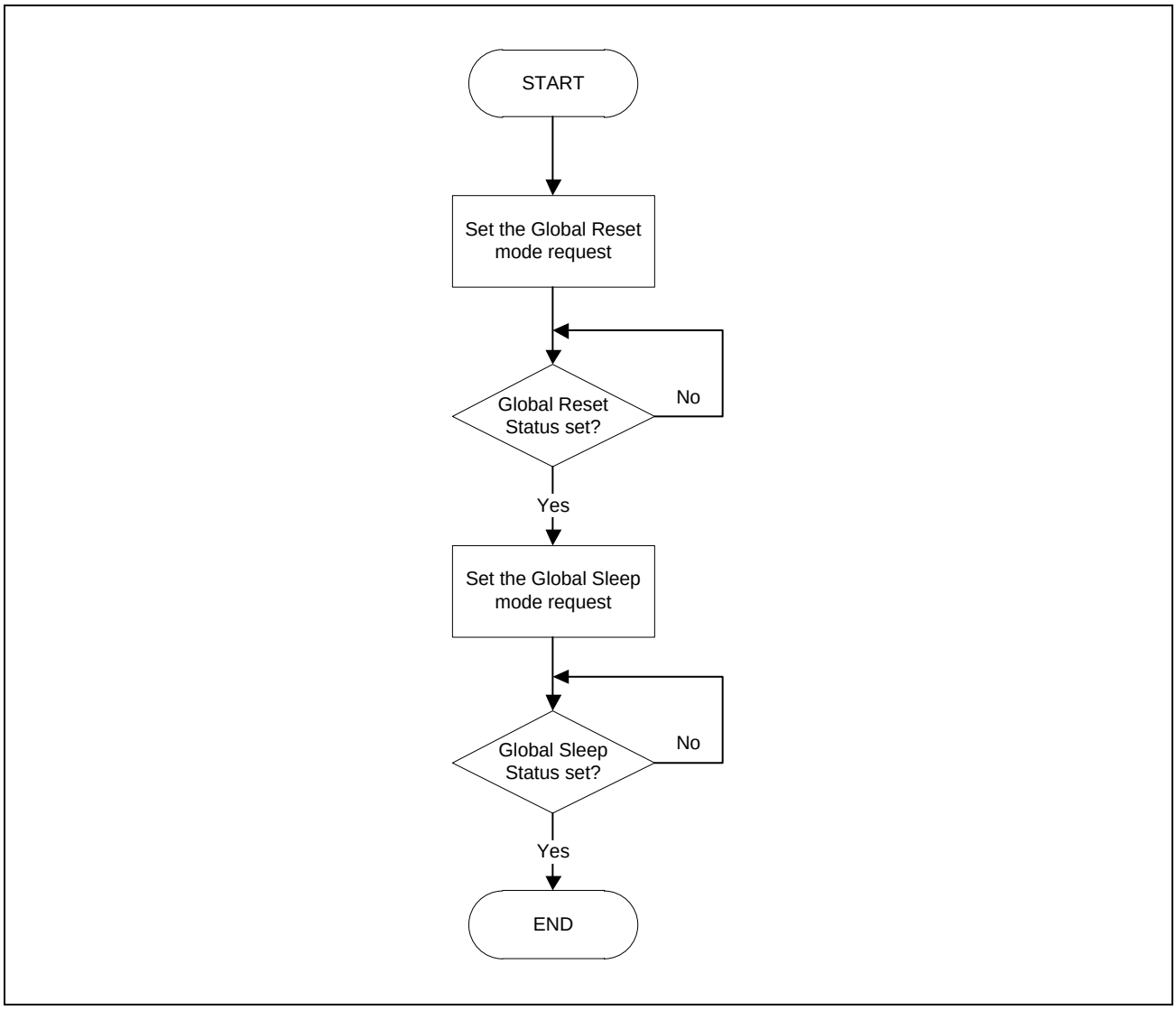


Figure 30.3 Procedure for Entering Global Sleep Mode

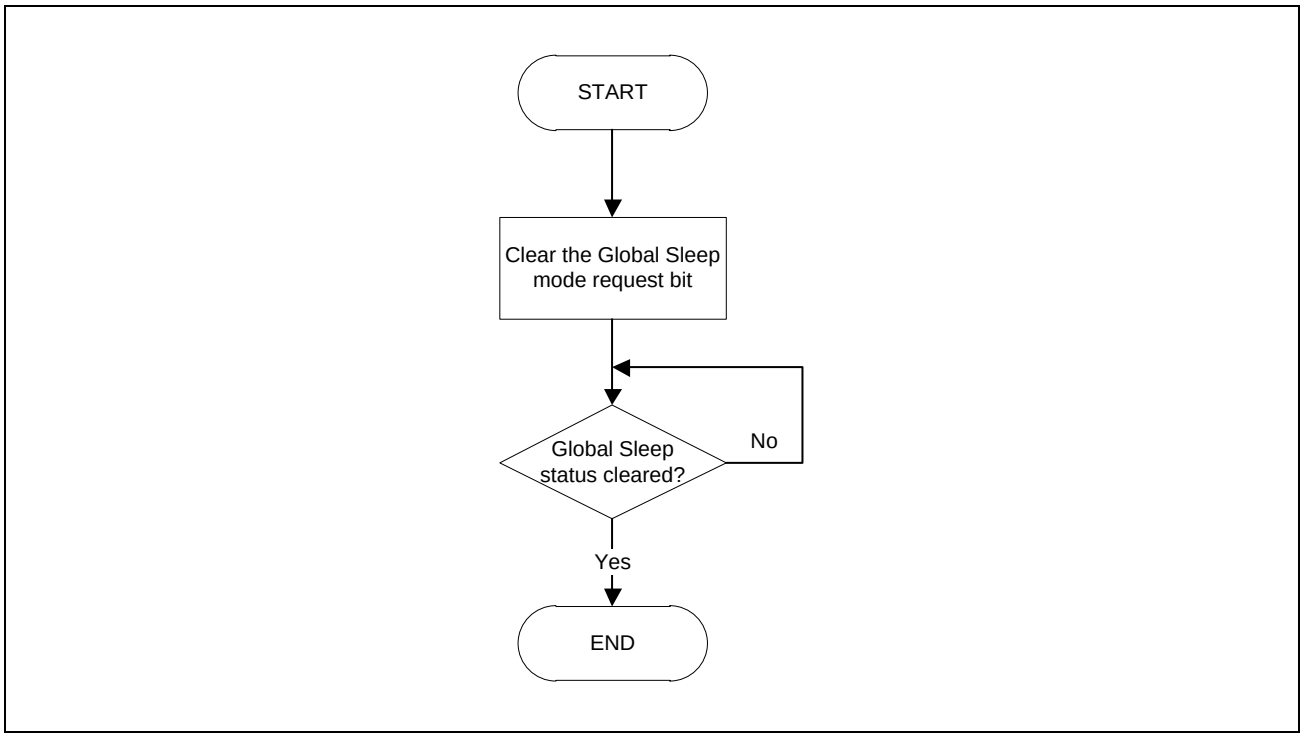


Figure 30.4 Procedure for Exiting Global Sleep Mode

30.3.2.2 Global Reset Mode

The CAN-FD module enters the Global Reset mode in the following ways:

- Global Mode Control bits CFDGCTR.GMDC[1:0] in the Global Control Register is configured for Global Reset mode while the CAN-FD module is in Global Halt or Global Operation mode.
- Global Sleep Mode Request bit is cleared while CAN-FD module is in Global Sleep mode.

In Global Reset mode, all CAN-FD module functions are suspended and all status and flag registers are initialized.

Additionally all FIFOs and all channel TX Queues are disabled and transmission control bits are cleared.

Configuration registers (except the test mode registers) are not initialized in this mode to their device reset values and the CAN-FD module can be configured.

See **Section 30.3.4, Global Mode and Channel Mode Transition Interactions** for a detailed description of the behavior of all registers when transition to Global Reset mode.

Setting the Global mode to Reset by setting the Global Mode Control bits CFDGCTR.GMDC[1:0] in the Global Control Register to 01b sets all Channel Mode Control bits CFDCnCTR.CHMDC[1:0] in the Channel Control Registers to 01b and forces all channels into the Channel Reset mode.

For channels that are already in Channel Reset mode or Channel Sleep mode, this automatic transition is not performed (CFDCnCTR.CHMDC[1:0] of related channel already set to 01b).

After setting Global Mode Control bits CFDGCTR.GMDC[1:0] to Reset mode, it is necessary to confirm that the Reset Mode Status bit CFDGSTS.GRSTSTS in the Global Status Register has been updated, indicating successful transition to Global Reset mode before CFDGCTR.GMDC can be changed again.

Figure 30.5 shows the procedure for entering Global Reset mode and **Figure 30.6** shows the procedure for exiting Global Reset mode.

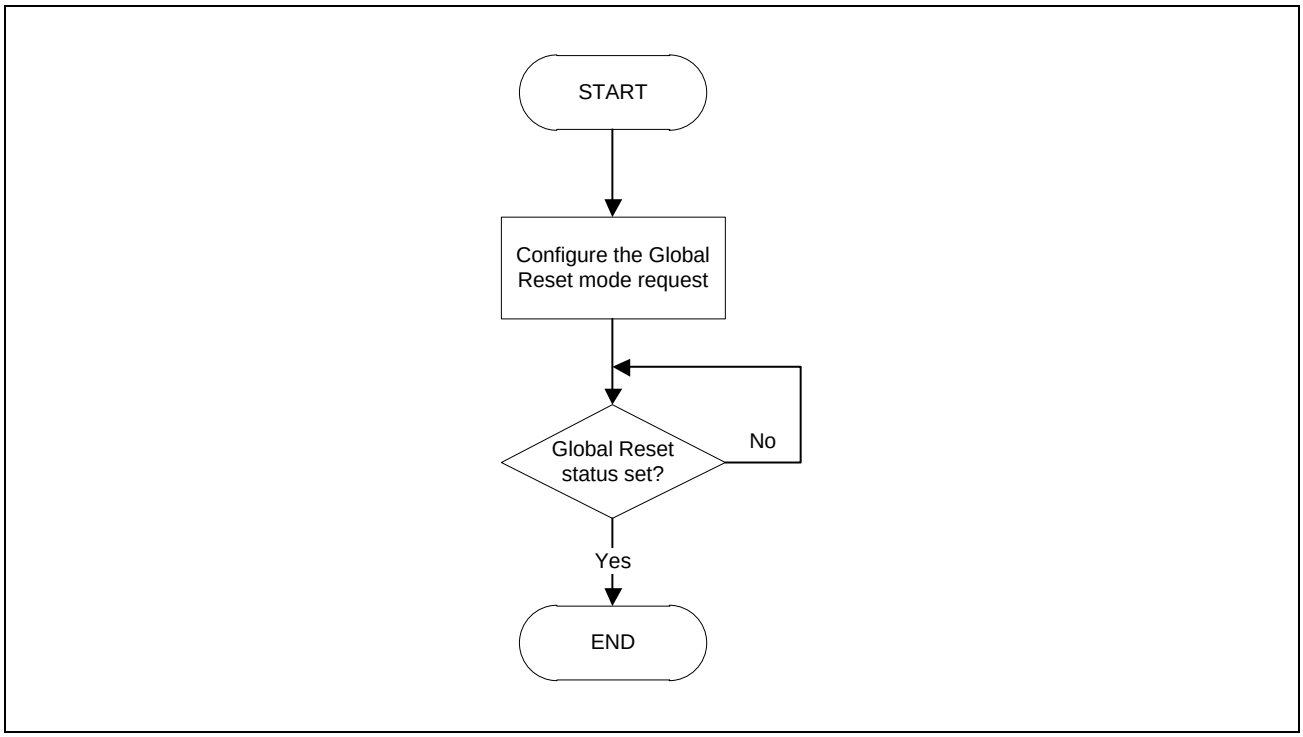


Figure 30.5 Procedure for Entering Global Reset Mode

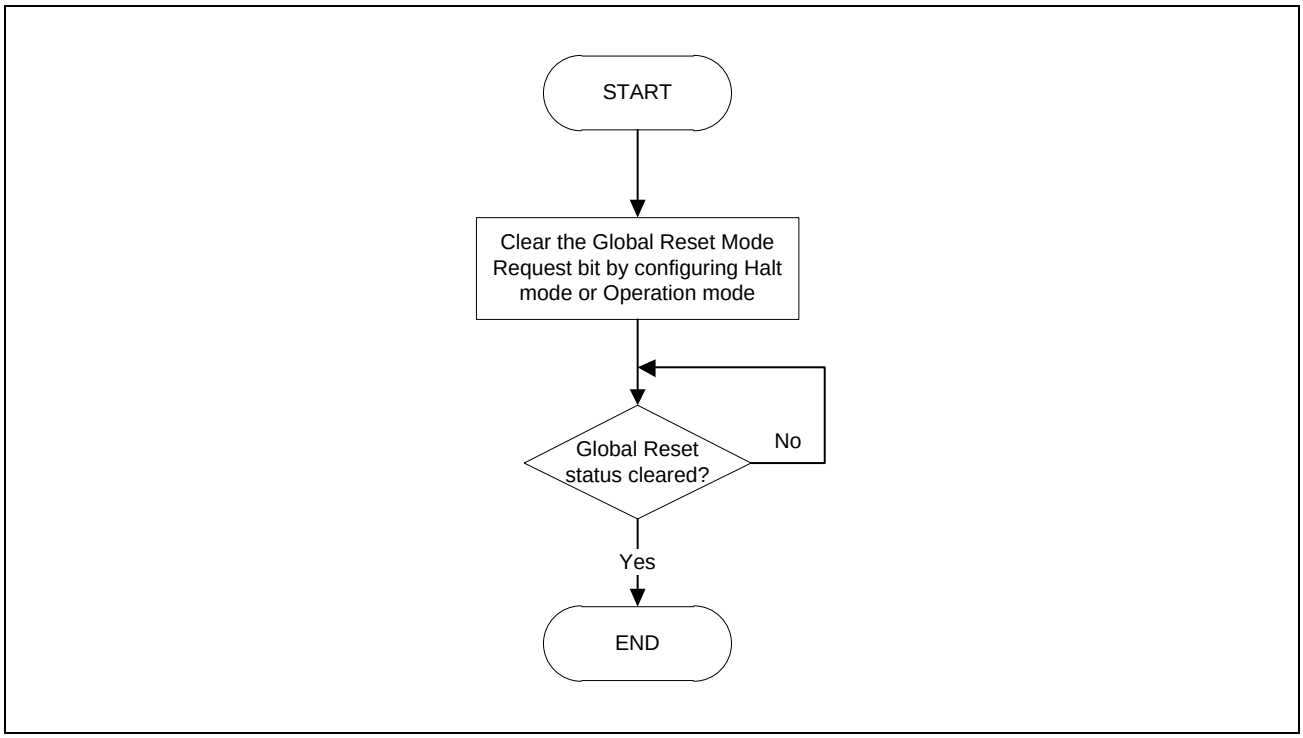


Figure 30.6 Procedure for Exiting Global Reset Mode

30.3.2.3 Global Halt Mode

The CAN-FD module enters the Global Halt mode in the following ways:

- Global Mode Control bits CFDGCTR.GMDC[1:0] in the Global Control Register is configured for Global Halt mode while the CAN-FD module is in Global Reset mode:
  - The channel is in either Channel Reset or Channel Sleep mode and remains in this mode.
- Global Mode Control bits CFDGCTR.GMDC[1:0] in the Global Control Register is configured for Global Halt mode while the CAN-FD module is in Global Operation mode:
  - All channels in Channel Reset, Channel Halt, or Channel Sleep mode remain in this mode
  - All channels in Channel Operation mode transit to Channel Halt mode
  - Global Halt Mode Status bit is set when all channels have left Channel Operation mode.

If a transmission or reception is ongoing for a channel, the transition to Channel Halt mode is delayed until completion of the communication.

Similarly, if a channel is in bus-off state, the full bus-off recovery sequence may be delayed depending on the channel configuration.

In Global Halt mode, all communications are suspended and CAN-FD logic does not cause any change to the Status and Flag registers (only when a channel is in the bus-off state that its REC and TEC values are cleared). Additionally, the test mode configuration and control registers are not initialized in this mode.

The Global Halt mode should be used to configure global module test modes.

See **Section 30.3.4, Global Mode and Channel Mode Transition Interactions** for a detailed description of the behavior of all registers when transition to Global Halt mode is performed.

Setting the Global mode to Halt mode by setting the Global Mode Control bits CFDGCTR.GMDC[1:0] in the Global Control Register to 10b sets all Channel Mode Control bits CFDCnCTR.CHMDC[1:0] in the Channel Control Registers to 10b for the channels that are in Channel Operation mode and forces these channels into the Channel Halt mode.

For channels that are already in Channel Reset, Channel Halt, or Channel Sleep mode, this automatic transition is not performed.

Therefore, the Global Halt mode request can be used to shut down all CAN-FD channel communications without loss of messages and disruption on the related CAN bus (no interruption of reception/transmission processes on the channels).

After setting the Global Mode Control bits CFDGCTR.GMDC[1:0] to Halt mode, it is necessary to confirm that the Halt Mode Status bit CFDGSTS.GHLTSTS in the Global Status Register has been updated to indicate a successful transition to Global Halt mode. Do not specify any other SFR setting until confirming CFDGSTS.GHLTSTS is set.

Figure 30.7 shows the procedure for entering Global Halt mode and **Figure 30.8** shows the procedure for exiting Global Halt mode.

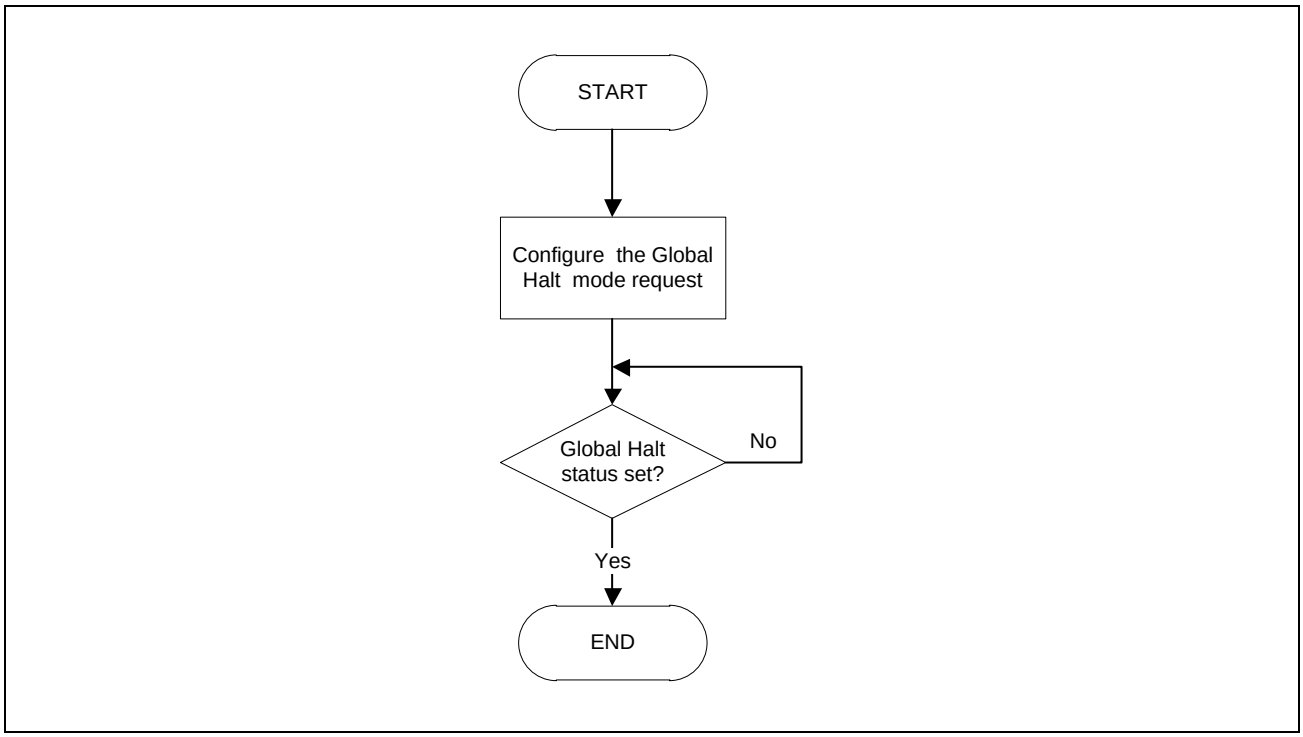


Figure 30.7 Procedure for Entering Global Halt Mode

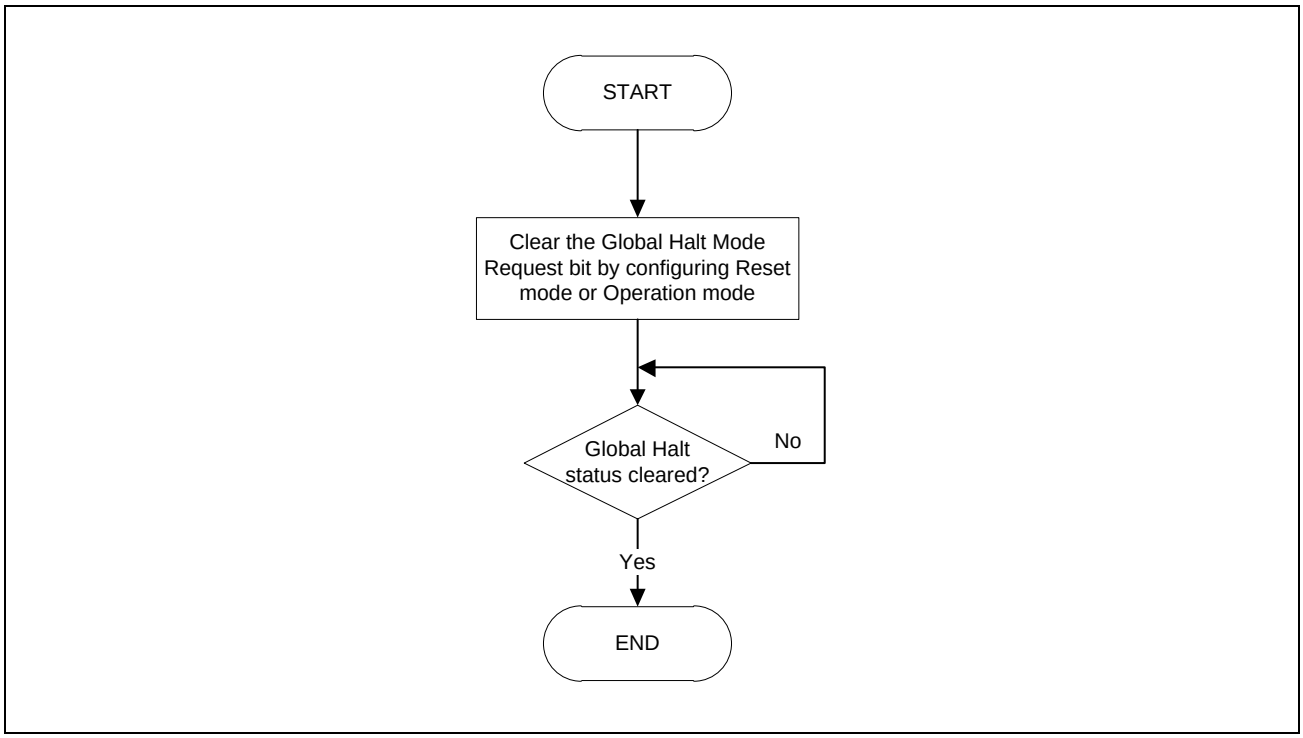


Figure 30.8 Procedure for Exiting Global Halt Mode

30.3.2.4 Global Operation Mode

The CAN-FD module enters the Global Operation mode when the Global Mode Configuration bits are set to Global Operation mode.

The CAN-FD channels can only be set to Channel Operation mode and start CAN communication when CAN-FD is in Global Operation mode.

After setting the Global Mode Control bits CFDGCTR.GMDC[1:0] to Global Operation mode, it is necessary to confirm that the Global Reset Mode Status bit CFDGSTS.GRSTSTS and the Global Halt Mode Status bit CFDGSTS.GHLTSTS in the Global Status Register have been cleared to indicate a successful transition to Global Operation mode before CFDGCTR.GMDC can be modified again.

Figure 30.9 shows the procedure for entering Global Operation mode and **Figure 30.10** shows the procedure for exiting Global Operation mode.

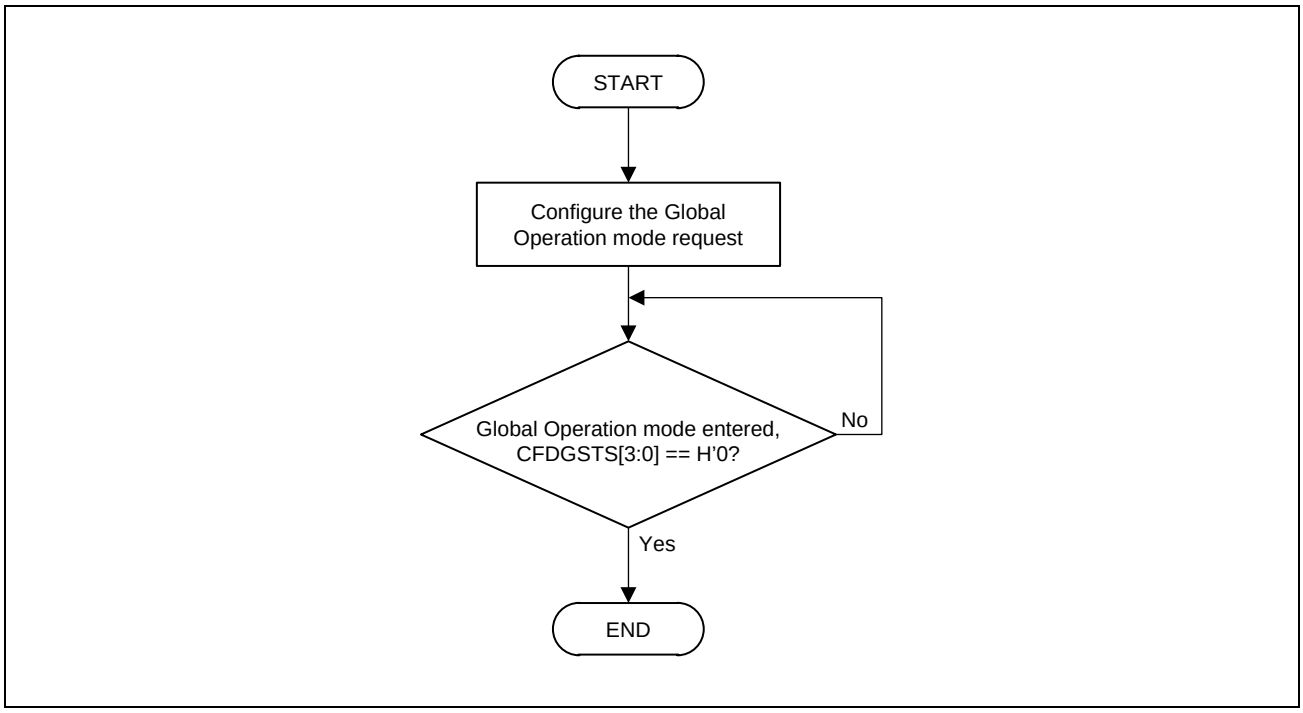


Figure 30.9 Procedure for Entering Global Operation Mode

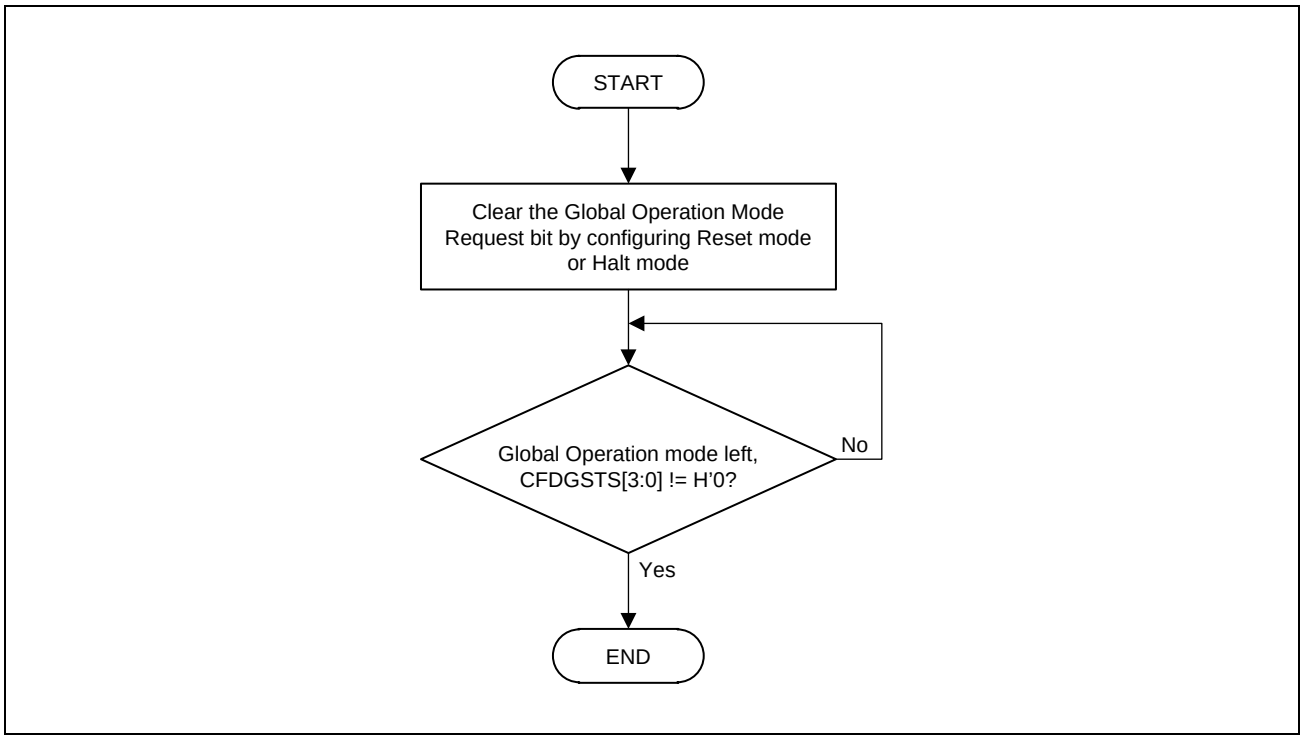


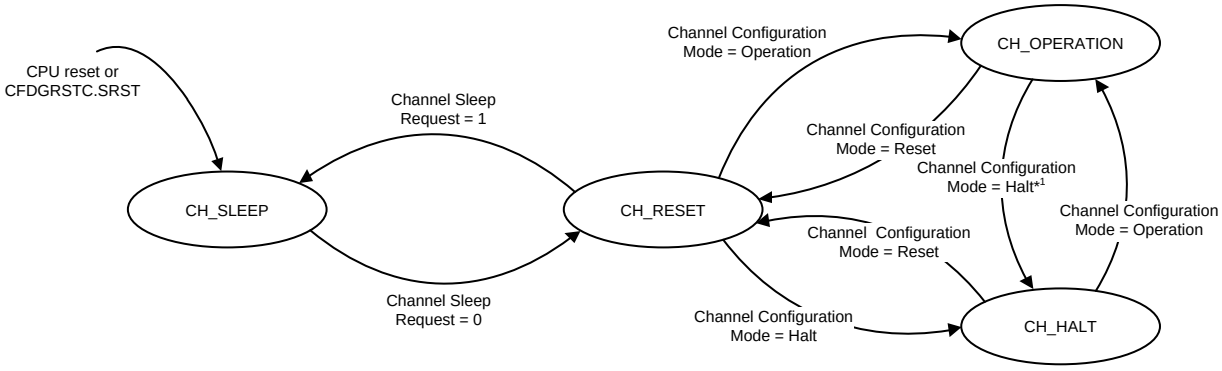
Figure 30.10 Procedure for Exiting Global Operation Mode

30.3.3 Channel Modes

Each CAN channel can be in one of the following four channel modes:

- Reset
- Halt
- Operation
- Sleep

Figure 30.11 shows possible transitions between the channel modes.



Note 1. The transition timing depends on BOM[1:0] setting in the Channel Control Register (CFDCnCTR):
When BOM[1:0] = 01b, the state transition timing is immediate after entering bus-off state
When BOM[1:0] = 10b, the state transition timing is at the end of bus-off state
When BOM[1:0] = 11b, the state transition timing is matching with the setting of CHMDC to HALT mode

Figure 30.11 Transition between CAN Channel Modes

30.3.3.1 CAN Channel Sleep Mode

After the release of a hardware reset or after setting and clearing the CFDGRSTC.SRST bit, each CAN channel of the CAN- FD module automatically enters Channel Sleep mode.

Each CAN channel also enters Channel Sleep mode when the related Channel Sleep Mode Request bit is set while the CAN channel is in Channel Reset mode. Do not set this control bit in Channel Halt mode or Channel Operation mode.

Entering the CAN Channel Sleep mode instantly stops the clock supplied to the CAN channel unit and therefore reduces power consumption.

After setting the Channel Sleep Mode Request bit, it is necessary to confirm that the Channel Sleep mode status has been updated to indicate a successful transition to Channel Sleep mode before the Channel Sleep Mode Request bit can be cleared again.

During Channel Sleep mode, do not write to channel related registers. Read operation is still possible.

30.3.3.2 CAN Channel Reset Mode

A CAN-FD CAN channel enters the CAN Channel Reset mode in the following ways:

- Channel Mode Control bits CFDCnCTR.CHMDC[1:0] in the Channel Control Registers is configured for Channel Reset mode while the related CAN channel is in Channel Halt mode or Channel Operation mode
- Channel Sleep Mode Request bit is cleared while the related CAN channel is in Channel Sleep mode
- Global Mode Control bits CFDGCTR.GMDC[1:0] is set to Global Reset mode and CAN channel is not in Channel Sleep mode or Channel Reset mode.

In Channel Reset mode, all CAN channel status and flag registers are initialized.

Additionally all channel-related transmission control bits are cleared and the channel-related TX Queue is disabled.

Configuration registers (except the Channel Test Mode registers) are not initialized in this mode and the CAN channel can be configured for communication.

See **Section 30.3.4, Global Mode and Channel Mode Transition Interactions** for a detailed description of the behavior of all registers when transition to Channel Reset mode is performed.

After setting the Channel Mode Control bits CFDCnCTR.CHMDC[1:0] to Channel Reset mode, it is necessary to confirm that the Reset Mode Status bit CFDCnSTS.CRSTSTS in the related Channel Status Registers has been updated to indicate a successful transition to Channel Reset mode before the related CFDCnCTR.CHMDC[1:0] bits can be modified again.

See **Table 30.26** for the behavior of transitioning to Channel Reset mode while CAN communication is ongoing.

Table 30.26 Transition Behavior in CAN Reset Mode and Halt Mode

	State		
Mode	Receiver	Transmitter	Bus-Off
CAN Channel Reset mode (CFDCnCTR.CHMDC[1:0] = 01b)	The CAN channel transits to Channel Reset mode without waiting for the completion of the ongoing reception.*1	The CAN channel transits to Channel Reset mode without waiting for the completion of the ongoing transmission.*1	The CAN channel transits to Channel Reset mode without waiting for the completion of the bus-off recovery.
CAN Channel Halt Mode (CFDCnCTR.CHMDC[1:0] = 10b)	The CAN channel transits to Channel Halt mode at the end of the ongoing reception or error.*2	The CAN channel transits to Channel Halt mode after completion of the ongoing transmission.	When CFDCnCTR.BOM[1:0] is set to 00b, a Channel Halt mode request is accepted only after the completion of the full bus-off recovery sequence. When CFDCnCTR.BOM[1:0] is set to 10b, the CAN channel transits automatically to Channel Halt mode after waiting for the completion of the bus-off recovery. When CFDCnCTR.BOM[1:0] is set to 01b, the CAN channel transits automatically to Channel Halt mode without waiting for the completion of the bus-off recovery. When CFDCnCTR.BOM[1:0] is set to 11b, the CAN channel transits to Channel Halt mode as soon as Channel Halt mode is requested (without waiting for the completion of the bus-off recovery).

Note 1. If the entry to Channel Reset is required only at the end of an ongoing communication, then Channel Halt mode can be requested first to prevent interruption of CAN communication by direct transition to Channel Reset mode. After the CAN channel enters Channel Halt mode, the Channel Reset mode can be requested.

Note 2. If CAN communication is locked at a dominant level after an error flag, software can detect this situation by monitoring the channel related Bus Lock flag and resolve the lock condition by setting the CAN channel to Channel Reset mode.

30.3.3.3 CAN Channel Halt Mode

A CAN-FD CAN channel enters the CAN Channel Halt mode in the following ways:

- Channel Mode Control bits CFDCnCTR.CHMDC[1:0] in the Channel Control Registers are configured for Channel Halt mode while the related CAN channel is in Channel Reset mode or Channel Operation mode
- Global Mode Control bits CFDGCTR.GMDC[1:0] are set to Global Halt mode and CAN channel is in Channel Operation mode.

In Channel Halt mode, all channel CAN communication is suspended but all status and flag registers remain unchanged during Channel Halt mode entry (except for the bus-off case where REC and TEC values are cleared for the channel).

In addition, the Channel Test Mode Configuration and Control registers are not initialized in this mode.

The Channel Halt mode should be used to configure Channel Test modes.

See **Section 30.3.4, Global Mode and Channel Mode Transition Interactions** for a detailed description of the behavior of all registers when transition to Channel Halt mode is performed.

After setting the Channel Mode Control bits CFDCnCTR.CHMDC[1:0] to Channel Halt mode, it is necessary to confirm that the Halt Mode Status bit CFDCnSTS.CHLTSTS in the related Channel Status Register is updated to indicate a successful transition to Channel Halt mode before the related CFDCnCTR.CHMDC[1:0] bits can be modified again.

See **Table 30.26** for the behavior of transitioning to Channel Halt mode while CAN communication is ongoing.

30.3.3.4 CAN Channel Operation Mode

The Channel Operation mode is activated by setting the CFDCnCTR.CHMDC[1:0] bits to 00b. If 11 consecutive recessive bits are detected after entering CAN Channel Operation mode, the CFDCnSTS.COMSTS bit is set and the CAN channel:

- Enables the functions of the channel communication by allowing the channel to become an active node on the CAN network
- Releases the internal fault confinement logic including receive and transmit error counters

At this point, the CAN channel can start transmission and reception of CAN messages.

Within the CAN Channel Operation mode, the channel may be in four different sub-modes, depending on which type of communication functions are performed (see **Figure 30.12**):

- Channel idle: The CAN channel is neither receiving nor transmitting
- Channel receives: The channel is receiving a CAN message sent by another CAN node
- Channel transmits: The channel is transmitting a CAN message

  Note: The channel may receive its own message simultaneously when Self-Test mode is enabled.
- Channel is in bus-off state: The CAN channel is cut-off from CAN bus communication.

After setting the Channel Mode Control bits CFDCnCTR.CHMDC[1:0] to Channel Operation mode, it is necessary to confirm that the Channel Reset Mode Status bit CFDCnSTS.CRSTSTS and the Channel Halt Mode Status bit CFDCnSTS.CHLTSTS in the Channel Status Register have been updated to indicate a successful transition to Channel Operation mode before the related CFDCnCTR.CHMDC[1:0] bits can be changed again.

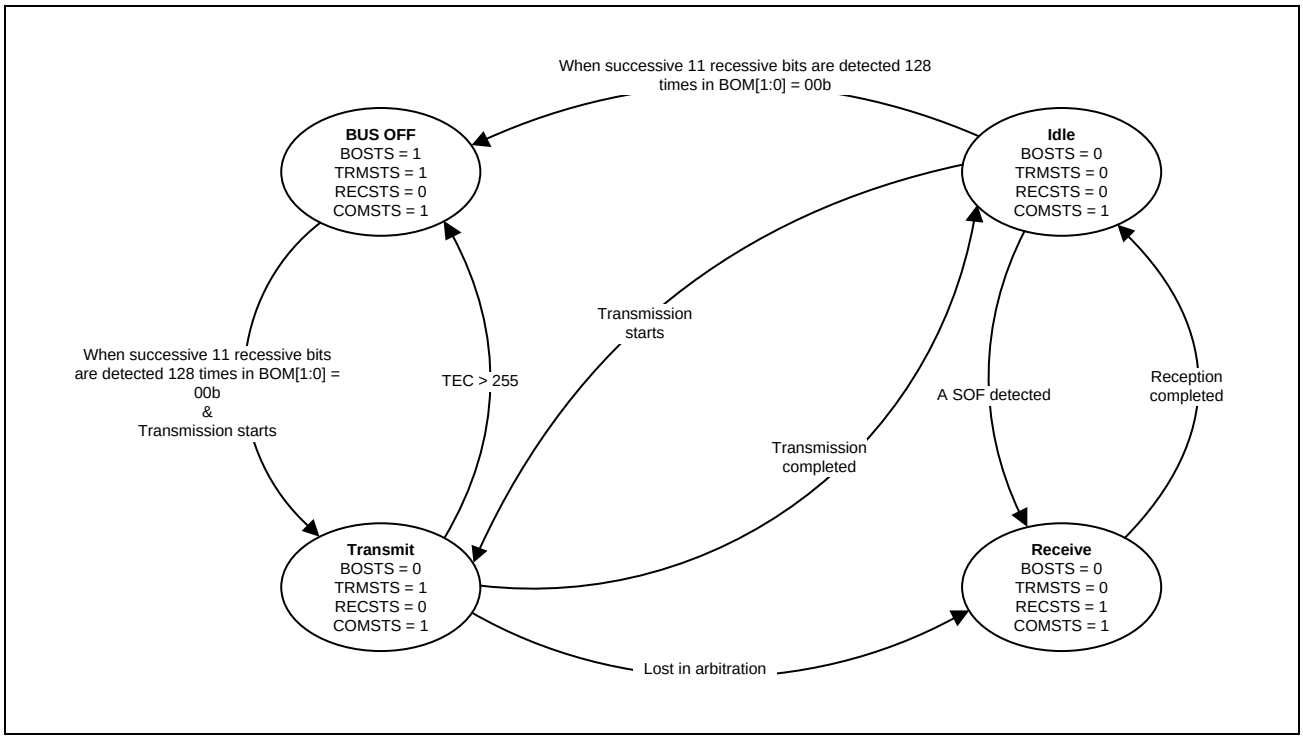


Figure 30.12 Sub-Modes of CAN Channel Operation Mode (Only when BOM[1:0] = 00b)

30.3.3.5 CAN Channel Bus-Off State

The CAN channel bus-off state is entered according to the fault confinement rules of the CAN specification. The following modes can be configured for returning to the CAN Channel Operation mode from the bus-off state:

- CFDCnCTR.BOM[1:0] = 00b:
  Bus-Off recovery is compliant to ISO 11898-1, namely the CAN channel re-enters CAN communication (error active state) after 11 consecutive recessive bits are detected 128 times. TEC and REC counters are initialized to 0. The Bus-Off Recovery Flag CFDCnERFL.BORF is set in this case.
- CFDCnCTR.BOM[1:0] = 01b:
  The CAN channel changes the value of the CFDCnCTR.CHMDC[1:0] bits within the CAN Channel Control Register to 10b and switches immediately to Channel Halt mode automatically after entering bus-off state. TEC and REC counters are initialized to 0 and the Bus-Off Recovery Flag CFDCnERFL.BORF is not set in this case.
- CFDCnCTR.BOM[1:0] = 10b:
  The CAN channel changes the value of the CFDCnCTR.CHMDC[1:0] bits within the CAN Channel Control Register to 10b as soon as it reaches bus-off state and enters Channel Halt mode automatically after the CAN channel has completed the bus-off recovery sequence (after 11 consecutive recessive bits are detected 128 times). TEC and REC counters are initialized to 0 and the Bus-Off Recovery Flag CFDCnERFL.BORF is set in this case.
- CFDCnCTR.BOM[1:0] = 11b:
  Bus-off recovery is initiated but CAN channel can immediately enter Channel Halt mode when still in bus-off state if a request is made to enter Channel Halt mode.
  TEC and REC counters are initialized to 0 and the Bus-Off Recovery Flag CFDCnERFL.BORF is not set. Without setting CFDCnCTR.CHMDC[1:0] = 10b and when 11 recessive bits is detected 128 times continuously, transition conditions become the same as CFDCnCTR.BOM[1:0] = 00b.

Note: If the recovery from bus-off occurs normally in Channel Halt mode (after waiting for 128 sequences of 11 consecutive recessive bits), and no halt request has been generated during this period, then the Bus-Off Recovery flag CFDCnERFL.BORF is set.

When software writes to the CFDCnCTR.CHMDC[1:0] bit at the same time as the CAN channel enters Halt mode (at the start of bus-off when CFDCnCTR.BOM[1:0] = 01b, or at the end of bus-off when CFDCnCTR.BOM[1:0] = 10b), the software request has the highest priority.

Note: In the above case, the automatic setting of the CFDCnCTR.CHMDC[1:0] bits to Channel Halt mode request is performed when the CFDCnCTR.CHMDC[1:0] bits value is previously 00b (Channel Operation mode).

Additionally, it is possible to force the CAN channel to recover from the bus-off state by setting CFDCnCTR.RTBO to 1. The error state changes from bus-off state to integrating state with a maximum delay of 1 CAN bit time, and the CAN communication becomes possible again after 11 consecutive recessive bits are detected. The Bus-Off Recovery Flag is not set in this case, and the TEC and REC counters are initialized to 0.

Before setting CFDCnCTR.RTBO to 1, all pending transmissions from the TX message buffers, TX Queues and/or Common FIFO in TX or GW mode should be disabled.

The disable of the pending transmission message buffer, TX Queue or FIFO must be confirmed by the corresponding acknowledge flags.

For the TX message buffer, the acknowledge flags are the Transmission Result Flags (CFDTMSTSn.TMTRF[1:0]). For the TX Queue, it is the TX Queue Empty flag (CFDTXQSTSn.TXQEMP). For the FIFO, it is the FIFO Empty flag (CFDCFSTSn.CFEMP).

The CFDCnCTR.RTBO bit should be used for bus-off recovery only when CFDCnCTR.BOM[1:0] is set to 00b.

Setting this bit in any state other than bus-off has no effect and the bit is cleared immediately.

Table 30.27 shows the settings for the Bus-Off Entry flag CFDCnERFL.BOEF and the Bus-Off Recovery flag CFDCnERFL.BORF for the different configurations of CFDCnCTR.BOM[1:0].

Table 30.27 Behavior of Bus-off Entry and Recovery Flags

BOM[1:0]	BOEF Bit Setting	BORF Bit Setting
00b	Always (on entry to bus-off)	Always (on exit from bus-off)
00b CFDCnCTR.RTBO set to 1	Always (on entry to bus-off)	Only if normal bus-off recovery occurs before software sets CFDCnCTR.RTBO to 1
01b	Always (on entry to bus-off)	Never
10b	Always (on entry to bus-off)	Always (on exit from bus-off)
11b	Always (on entry to bus-off)	Only if normal bus-off recovery occurs before software issues a Halt request

For an efficient software procedure, it is not necessary to wait for the bus-off recovery sequence to end.

It is possible to perform a transmission re-initialization during bus-off recovery. To do this, follow the recommended software flow in **Figure 30.13**.

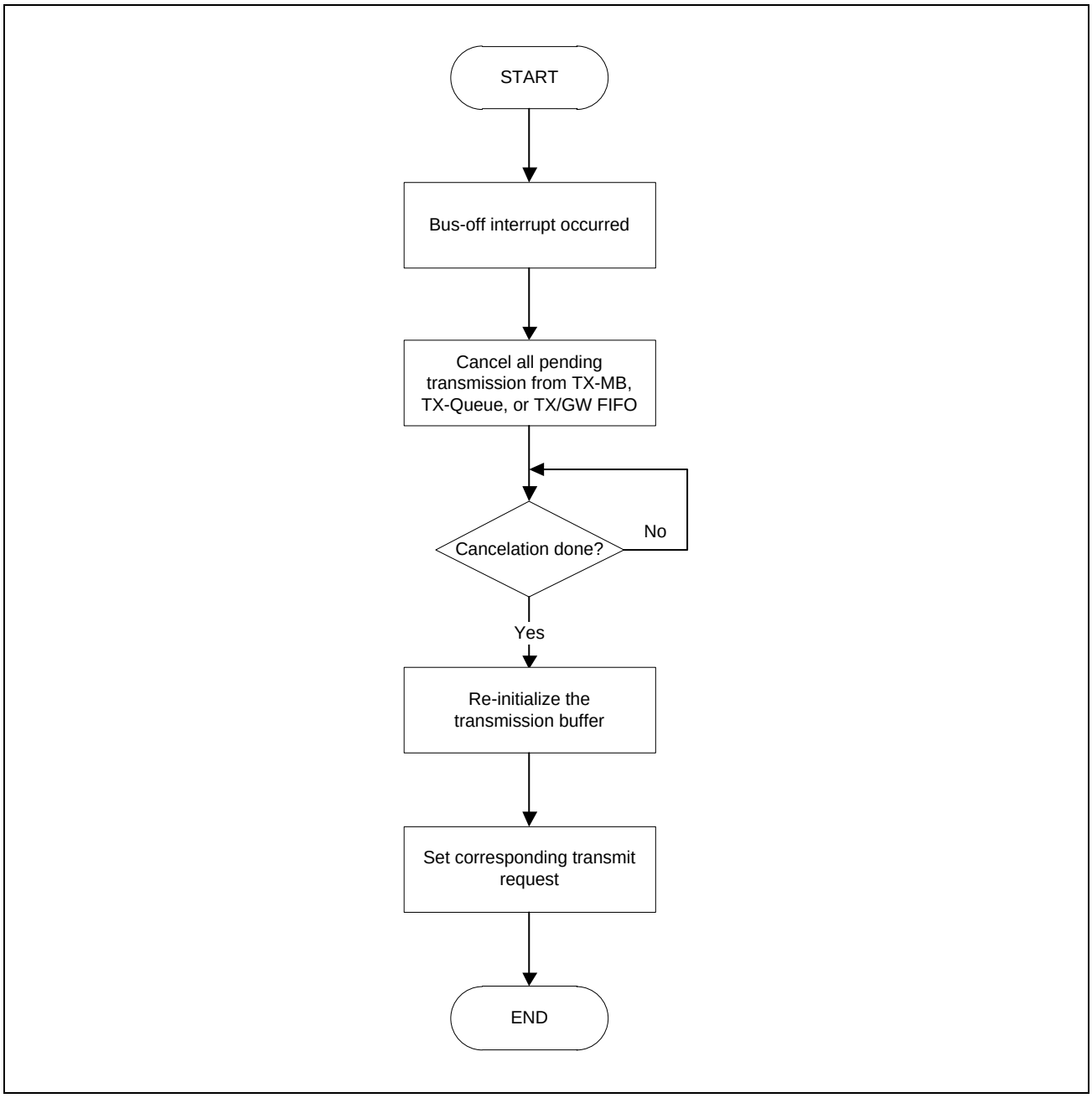


Figure 30.13 Transmission Re-Initialization during Bus-Off State

30.3.4 Global Mode and Channel Mode Transition Interactions

The interaction between Global mode setting and Channel mode setting is as follows:

- Changing the Channel Mode Control bits CFDCnCTR.CHMDC[1:0] in the Channel Control Registers does not affect the Global Mode Control bits CFDGCTR.GMDC[1:0].
- Changing the Global Mode Control bits CFDGCTR.GMDC[1:0] affects the channel mode control as described in **Table 30.28**.

Table 30.28 Instruction between Global and Channel Mode Transition

Global Mode Change	Channel Mode	Channel Mode Transition Action
Sleep → Reset	Sleep	Channel remains in Sleep mode
Sleep → Halt	— (Global mode change not possible)	
Sleep → Operation	— (Global mode change not possible)	
Reset → Sleep	Sleep	Channel remains in Sleep mode
	Reset	Channel Sleep request bit is set automatically, channel transits to Sleep mode
Reset → Halt	Sleep	Channel remains in Sleep mode
	Reset	Channel remains in Reset mode
Reset → Operation	Sleep	Channel remains in Sleep mode
	Reset	Channel remains in Reset mode
Halt → Sleep	— (Global mode change not possible)	
Halt → Reset	Sleep	Channel remains in Sleep mode
	Reset	Channel remains in Reset mode
	Halt	Channel mode control is set to Reset mode, channel transits to Reset mode
Halt → Operation	Sleep	Channel remains in Sleep mode
	Reset	Channel remains in Reset mode
	Halt	Channel remains in Halt mode
Operation → Sleep	— (Global mode change not possible)	
Operation → Reset	Sleep	Channel remains in Sleep mode
	Reset	Channel remains in Reset mode
	Halt	Channel mode control is set to Reset mode, channel transits to Reset mode
	Operation	Channel mode control is set to Reset mode, channel transits to Reset mode
Operation → Halt	Sleep	Channel remains in Sleep mode
	Reset	Channel remains in Reset mode
	Halt	Channel remains in Halt mode
	Operation	Channel mode control is set to Halt mode, channel transits to Halt mode after communication finished

30.3.4.1 Timing of Global Mode Change

The transition time for the Global mode changes are shown in the following table.

From	To	Maximum Transition Time
GL_SLEEP	GL_RESET	3 peripheral clock cycles*2
GL_RESET	GL_SLEEP	3 peripheral clock cycles
GL_RESET	GL_HALT	10 peripheral clock cycles
GL_RESET	GL_OPERATION	10 peripheral clock cycles
GL_HALT	GL_RESET	2 CAN bit times
GL_HALT	GL_OPERATION	3 peripheral clock cycles
GL_OPERATION	GL_RESET	2 CAN bit times
GL_OPERATION	GL_HALT	3 CAN frames*1,*3

Note 1. The given transition time is the time without any errors on the bus. In case of an error condition, the transition time can lengthen to an uncalculated result. The transition time can also come to a stuck condition for locked RX lines or continued error conditions.

Note 2. Exit GL_SLEEP mode only when CFDGSTS.GRAMINIT is cleared.

Note 3. TQ, CAN frame, and CAN bits are related to the individual channels. For maximum transition time, the channel with the lowest baud rate must be used.

30.3.4.2 Timing of Channel Mode Change

The transition time for the Channel mode changes are shown in the following table.

From	To	Maximum transition time
CH_SLEEP	CH_RESET	3 peripheral clock cycles
CH_RESET	CH_SLEEP	3 peripheral clock cycles
CH_RESET	CH_HALT	3 CAN bit times
CH_RESET	CH_OPERATION	4 CAN bit times
CH_HALT	CH_RESET	2 CAN bit times
CH_HALT	CH_OPERATION	4 CAN bit times*2
CH_OPERATION	CH_RESET	2 CAN bit times
CH_OPERATION	CH_HALT	2 CAN frames*1,*3

Note 1. The time specified for this transition does not include the case where channel enters bus-off state. For bus-off, the timing depends on the configuration of the CFDCnCTR.BOM[1:0] bits.

Note 2. The given transition time is the time without any errors on the bus. In case of an error condition, the transition time can lengthen to an uncalculated result. The transition time can also come to a stuck condition for locked RX lines or continued error conditions.

Note 3. In general, if the baud rate prescaler value CFDCnNCFG.NBRP is changed in CH_HALT mode, the transition time can deviate. The internal prescaler is a free running down counter that creates the TQ clock, and new BRP value is captured when the counter reaches the value 0.

30.4 Initialization

Before joining CAN communications, configure the following settings:

- Clock setting
- Bit timing setting (nominal and data rate)
- Baud rate setting (nominal and data rate)
- CAN-FD setting
- Acceptance filter setting (configuration of Global Acceptance Filter List)
- Reception, Transmission and GW-FIFO setting
- CAN Operation mode setting

30.4.1 Initialization of CAN Clock, Bit Timing and Baud Rate

30.4.1.1 Bit Timing Conditions

The following lines describe the composition of each segment and the limitations that apply to the segment setting.

1. Each segment setting SS = Fixed to 1 TQ
   TSEG1 = Refer to (CFDCnNCFG) and (CFDCnDCFG)
   TSEG2 = Refer to (CFDCnNCFG) and (CFDCnDCFG)
   SJW = Refer to (CFDCnNCFG) and (CFDCnDCFG)
   SS + TSEG1 + TSEG2 = 5 to 49 TQs for data bit rate and 8 to 385 for nominal bit rate

2. Limitations on TSEG1, TSEG2 and SJW
   TSEG1 (N) > TSEG2 (N) ≥ SJW (N)
   TSEG1 (D) ≥ TSEG2 (D) ≥ SJW (D)
   Table 30.29 shows an example of how to set the bit timing to achieve the required Sample Point settings.

Table 30.29 Transition Behavior in CAN Reset Mode and Halt Mode

	Set Value (TQ)				
1 Bit	SS	TSEG1	TSEG2	SJW	Sample Point*1 (%)
5 TQ	1	2	2	1	60.00
8 TQ	1	4	3	1	62.50
	1	5	2	1	75.00
10 TQ	1	6	3	1	70.00
	1	7	2	1	80.00
12 TQ	1	8	3	1	75.00
	1	9	2	1	83.33
15 TQ	1	10	4	1	73.33
	1	11	3	1	80.00
16 TQ	1	10	5	1	68.75
	1	11	4	1	75.00
20 TQ	1	12	7	1	65.00
	1	13	6	1	70.00
24 TQ	1	15	8	1	66.66
	1	16	7	1	70.83
50 TQ	1	39	10	4	80.00

Note 1. Sample Point (in Case of 75%)

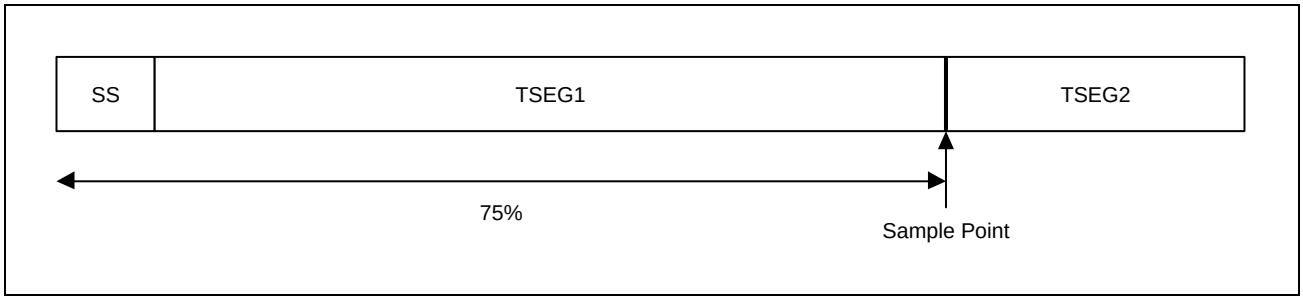


Figure 30.14 Sample Point (in Case of 75%)

30.4.1.2 CAN Bit Timing

In the CAN protocol, each bit in a communication frame is composed of three segments that can be configured individually for each channel using the related CFDCnNCFG and CFDCnDCFG registers.

Figure 30.15 shows the segment composition of a bit and the sample point in it.

Of these segments, the Time Segment 1 (TSEG1) and Time Segment 2 (TSEG2) are used to specify the position of the sample point, so that the timing at which each bit on the CAN bus is sampled can be altered by changing the values of these segments.

The minimum resolution for this timing is referred to as Time Quantum (TQ), which is determined by the clock frequency supplied to the CAN channel and the divide-by-N value of the baud rate prescaler (nominal and data rate).

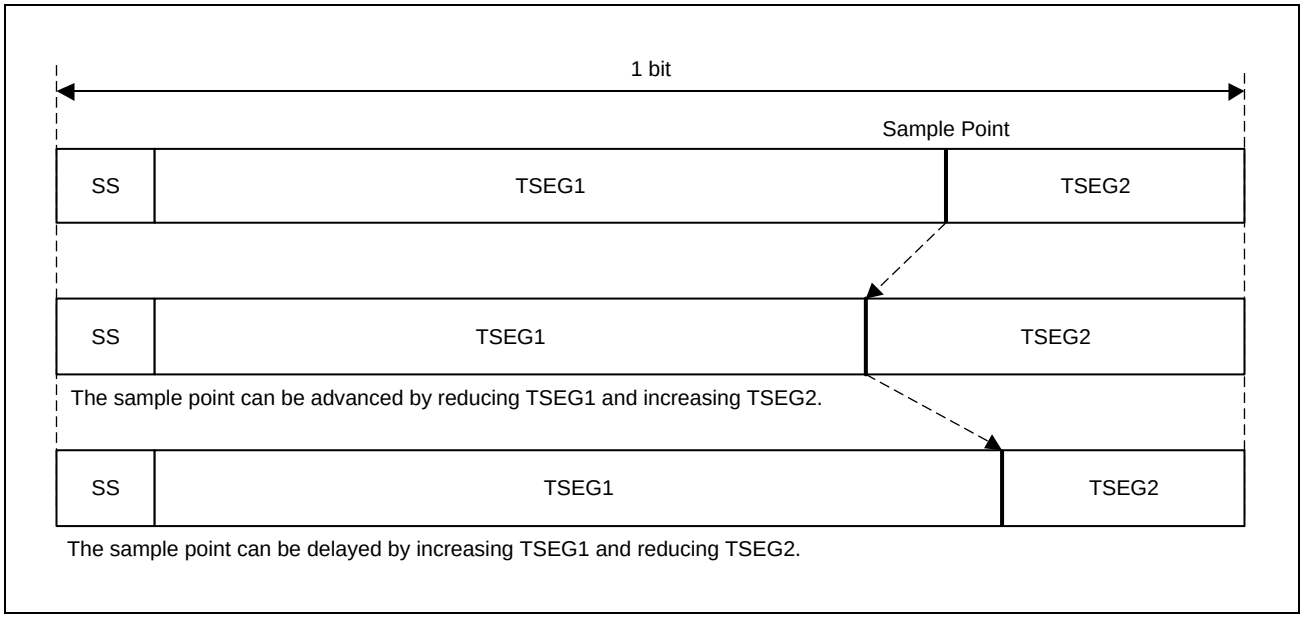


Figure 30.15 Segment Composition of a Bit and the Sample Point

1. SS: Synchronization Segment
   This segment is used to synchronize bits by monitoring a recessive-to-dominant edge during the interframe space. This comprises of intermission, suspend transmission, bus idle, during bus idle, and all nodes that can start transmission.
2. TSEG1: Time Segment 1
   This segment absorbs physical delays on the CAN network. A physical delay on the network is two times the total sum of a bus delay, input comparator delay, and output driver delay. It can be lengthened by SJW.
3. TSEG2: Time Segment 2
   This segment is used to correct a phase error by performing resynchronization. It can be shortened by SJW. While sending or receiving a message, communication frames between some nodes may get out of sync due to a drift in the oscillator frequency or a delay in the transmission path. This is referred to as a phase error.
4. SJW: Resynchronization Jump Width
   This is the maximum width by which bits that have become out of sync due to a phase error may be corrected.

Figure 30.15 shows only one symbolic sample point.

30.4.1.3 Baud Rate

Either PCLKM or PCLKCAN can be selected globally for all CAN channels as CAN communication clock.

The transfer speed is determined by the DLL clock, the divide-by-N value of the baud rate prescaler, and the number of TQs in one bit.

$$\text{Baud rate} = \frac{\text{DLL\_Clock}}{(\text{number\_of\_time\_quanta\_per\_bit}) \times (\text{BRP} + 1)}$$

Figure 30.16 shows a block diagram of the circuit that generates the CAN channel system clock, **Table 30.30** and **Table 30.31** show baud rate examples.

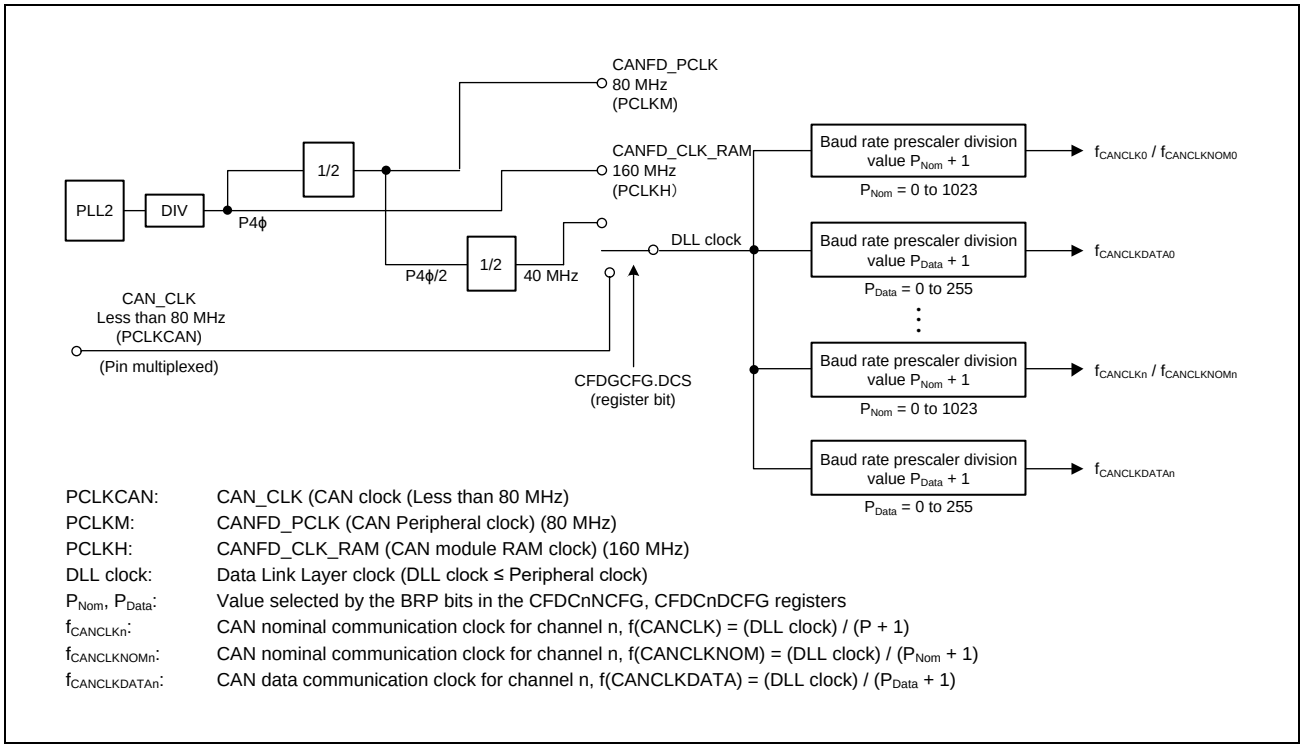


Figure 30.16 Block Diagram of the Circuit that Generates the CAN Channel Communication Clock

Table 30.30 Nominal Baud Rate Calculation Formula and Example CAN Communication Configurations

Baud rate calculation Formula	(DLL Clock) (Baud Rate Prescaler Divide-By-N Value*1) × (Number of TQs in One Bit)								
	80 MHz	40 MHz	32 MHz	30 MHz	24 MHz	20 MHz	16 MHz	10 MHz	8 MHz*2
1 Mbps	8 TQ (10) 20 TQ (4)	8 TQ (5) 20 TQ (2)	8 TQ (4) 16 TQ (2)	10 TQ (3) 15 TQ (2)	8 TQ (3) 12 TQ (2) 24 TQ (1)	10 TQ (2) 20 TQ (1)	8 TQ (2) 16 TQ (1)	10 TQ (1)	8 TQ (1)
500 Kbps	8 TQ (20) 20 TQ (8)	8 TQ (10) 20 TQ (4)	8 TQ (8) 16 TQ (4)	10 TQ (6) 15 TQ (4) 20 TQ (3)	8 TQ (6) 12 TQ (4) 24 TQ (2)	10 TQ (4) 20 TQ (2)	8 TQ (4) 16 TQ (2)	10 TQ (2) 20 TQ (1)	8 TQ (2) 16 TQ (1)
250 Kbps	8 TQ (40) 20 TQ (16)	8 TQ (20) 20 TQ (8)	8 TQ (16) 16 TQ (8)	10 TQ (12) 15 TQ (8) 20 TQ (6)	8 TQ (12) 12 TQ (8) 24 TQ (4)	10 TQ (8) 20 TQ (4)	8 TQ (8) 16 TQ (4)	10 TQ (4) 20 TQ (2)	8 TQ (4) 16 TQ (2)
125 Kbps	8 TQ (80) 20 TQ (32)	8 TQ (40) 20 TQ (16)	8 TQ (32) 16 TQ (16)	10 TQ (24) 15 TQ (16) 20 TQ (12)	8 TQ (24) 12 TQ (16) 24 TQ (8)	10 TQ (16) 20 TQ (8)	8 TQ (16) 16 TQ (8)	10 TQ (8) 20 TQ (4)	8 TQ (8) 16 TQ (4)
83.3 Kbps	8 TQ (120) 12 TQ (80) 16 TQ (60) 24 TQ (40)	8 TQ (60) 12 TQ (40) 16 TQ (30) 24 TQ (20)	8 TQ (48) 12 TQ (32) 16 TQ (24) 24 TQ (16)	8 TQ (45) 10 TQ (36) 12 TQ (30) 15 TQ (24) 20 TQ (18) 24 TQ (15)	8 TQ (36) 12 TQ (24) 16 TQ (18) 24 TQ (12)	8 TQ (30) 10 TQ (24) 12 TQ (20) 15 TQ (16) 16 TQ (15) 20 TQ (12) 24 TQ (10)	8 TQ (24) 12 TQ (16) 16 TQ (12) 24 TQ (8)	8 TQ (15) 10 TQ (12) 12 TQ (10) 15 TQ (8) 20 TQ (6) 24 TQ (5)	8 TQ (12)
33.3 Kbps	8 TQ (300) 12 TQ (200) 16 TQ (150) 20 TQ (120) 24 TQ (100)	8 TQ (150) 12 TQ (100) 16 TQ (75) 20 TQ (60) 24 TQ (50)	8 TQ (120) 10 TQ (96) 12 TQ (80) 15 TQ (64) 16 TQ (60) 20 TQ (48) 24 TQ (40)	10 TQ (90) 12 TQ (75) 15 TQ (60) 20 TQ (45)	8 TQ (90) 10 TQ (72) 12 TQ (60) 15 TQ (48) 16 TQ (45) 20 TQ (36) 24 TQ (30)	8 TQ (75) 10 TQ (60) 12 TQ (50) 15 TQ (40) 20 TQ (30) 24 TQ (25)	8 TQ (60) 10 TQ (48) 12 TQ (40) 15 TQ (32) 16 TQ (30) 20 TQ (24) 24 TQ (20)	10 TQ (30) 12 TQ (25) 15 TQ (20) 20 TQ (15)	8 TQ (30)

Note: Shown in ( ) are the baud rate prescaler divided-by-N values.

Note 1. Baud rate prescaler divide-by-N value = P + 1 (P = 0 to 1023), P: value selected by the BRP bits in the Channel Configuration Registers.

Note 2. Minimum Frequency to achieve maximum. Nominal Baud Rate of 1Mbps.

Table 30.31 Baud Rate Calculation Example for Nominal and Data Bit Rate CAN Communication Configurations

Baud rate calculation formula	(DLL Clock) (Baud Rate Prescaler Divide-By-N Value*1) × (Number of TQs in One Bit)		
	80 MHz	40 MHz	20 MHz
Normal 1 Mbps Data 8 Mbps	80 TQ (1)	40 TQ (1)	20 TQ (1)
	10 TQ (1)	5 TQ (1)	Not possible
Normal 1 Mbps Data 5 Mbps	80 TQ (1)	40 TQ (1)	20 TQ (1)
	16 TQ (1)	8 TQ (1)	Not possible
Normal 500 Kbps Data 2 Mbps	160 TQ (1)	80 TQ (1)	40 TQ (1)
	40 TQ (1)	20 TQ (1)	10 TQ (1)

Note: Shown in ( ) are the baud rate prescaler divided-by-N values.

Note 1. Baud rate prescaler divide-by-N value = P + 1 (P = 0 to 1023), P: value selected by the BRP bits in the Channel Configuration Registers.

For optimum clock tolerance in networks using the FD frame format, the length of the time quantum should be the same in nominal bit time and in data bit time. This means CFDCnNCFG.NBRP = CFDCnDCFG.DBRP.

Additionally, if transceiver delay compensation is used, do not program the CFDCnDCFG.DBRP[7:0] bits to be greater than 1, as 1 means divide by 2.

30.4.1.4 Setting of CAN Clock, Bit Timing and Baud Rate

Figure 30.17 shows the procedure for setting the CAN clock and the baud rate for each channel.

These settings should be performed during Channel Reset mode (Configuration mode) for the CAN channels.

Before going to channel communication state, the baud rate must be configured, otherwise the mode does not switch correctly.

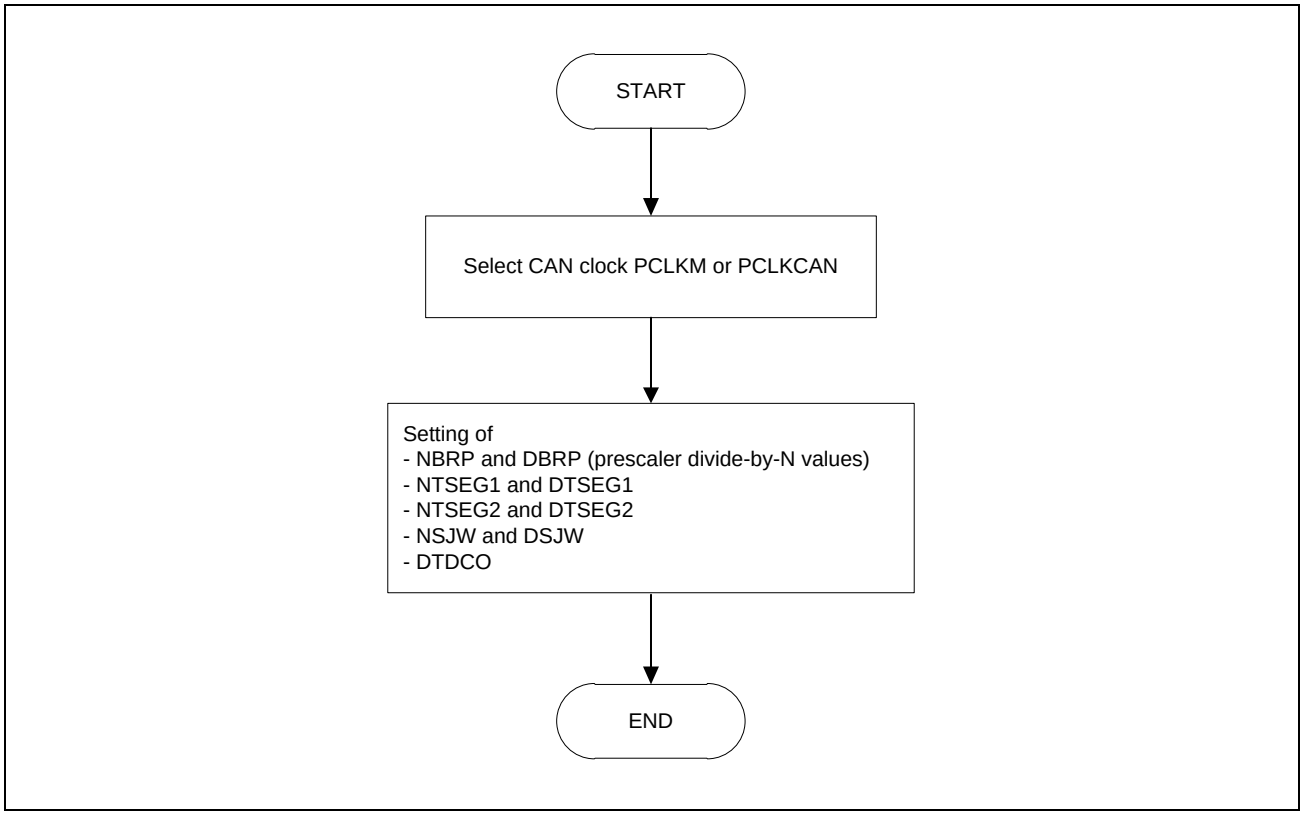


Figure 30.17 Procedure for Setting the CAN Bit Timing and Baud Rate

30.4.1.5 Transmitter Delay Compensation

When a high baud rate is used such as 5 to 8 Mbps for the data phase, the transmitter delay can become greater than TSEG1. In this case, the transmitter always detects a bit-error in the data phase of the CAN-FD frame. The TDC compensates for the inability of the transmitter to receive its own transmitted bit at the sample point of that bit.

There is another symbolic sample point known as the Secondary Sample Point (SSP) that is used only during the data phase of CAN-FD frames. This is derived from the Transceiver Delay Compensation Result bit (CFDCnFDSTS.TDCR) as shown in **Figure 30.18**.

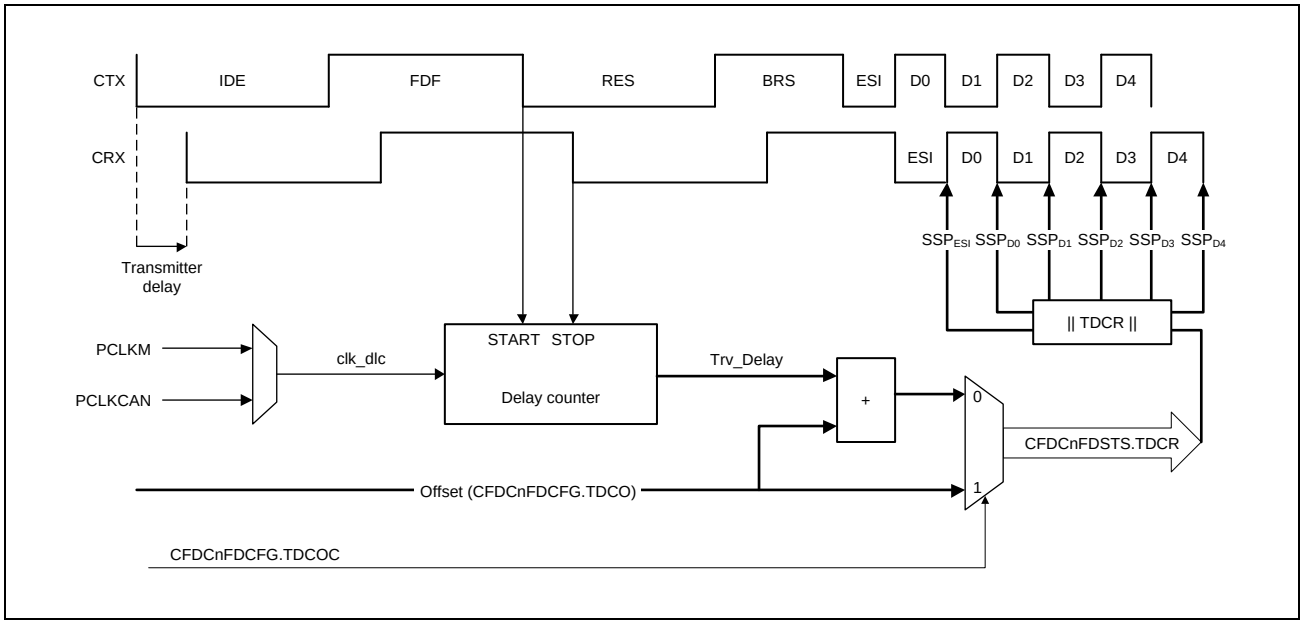


Figure 30.18 Transmitter Delay Compensation

The measured Trv_Delay is based on the number of clk_dlc clock cycles. The delay is counted up by 1 for each started clock until the dominant value is seen on CAN_RX. **Figure 30.19** shows the measured result. Trv_Delay counted to maximum 127 with a clk_dlc clock.

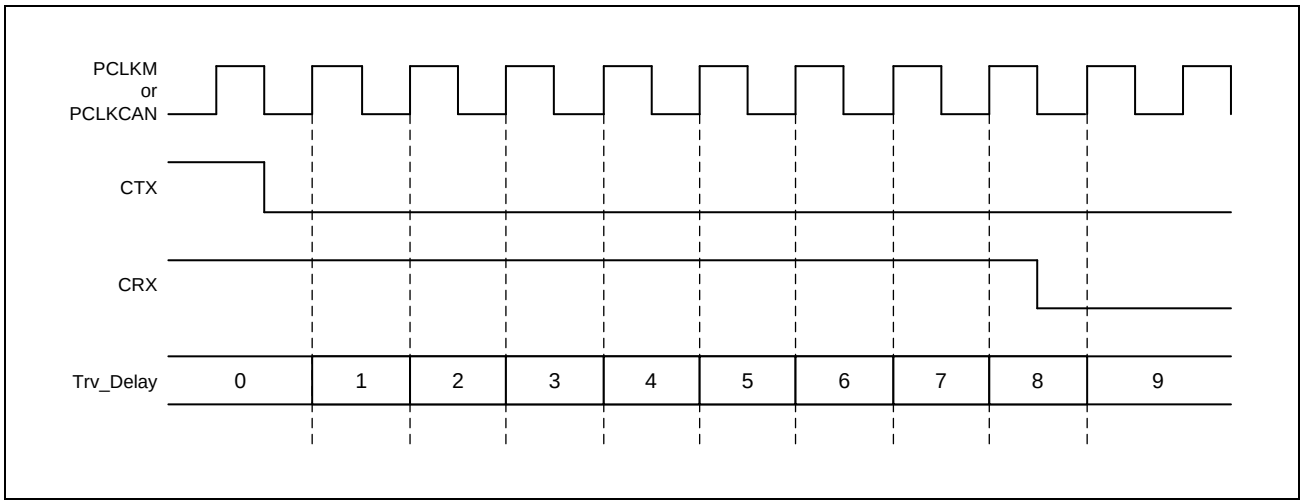


Figure 30.19 Trv_Delay Measurement Example

The SSP is calculated by taking the result from CFDCnFDSTS.TDCR and rounding the value down to the nearest integer number of data time quanta.

Figure 30.20 shows the positioning of the secondary sample point. When CFDCnFDCFG.TDCOC is equal to 0, the SSP is equal to the Trv_Delay (measured delay) + CFDCnFDCFG.TDCO, rounded down to the nearest integer number of time quanta. Usually, the TDCO value should have the size of (Sync Segment data + TSEG1 data) to position the SSP to a theoretical location of the sample point.

If the CFDCnFDCFG.TDCOC is equal to 1, the SSP is defined by CFDCnFDCFG.TDCO. If CFDCnDCFG.DBRP[7:0] is greater than 0, the value is also rounded down to the nearest integer number of time quanta.

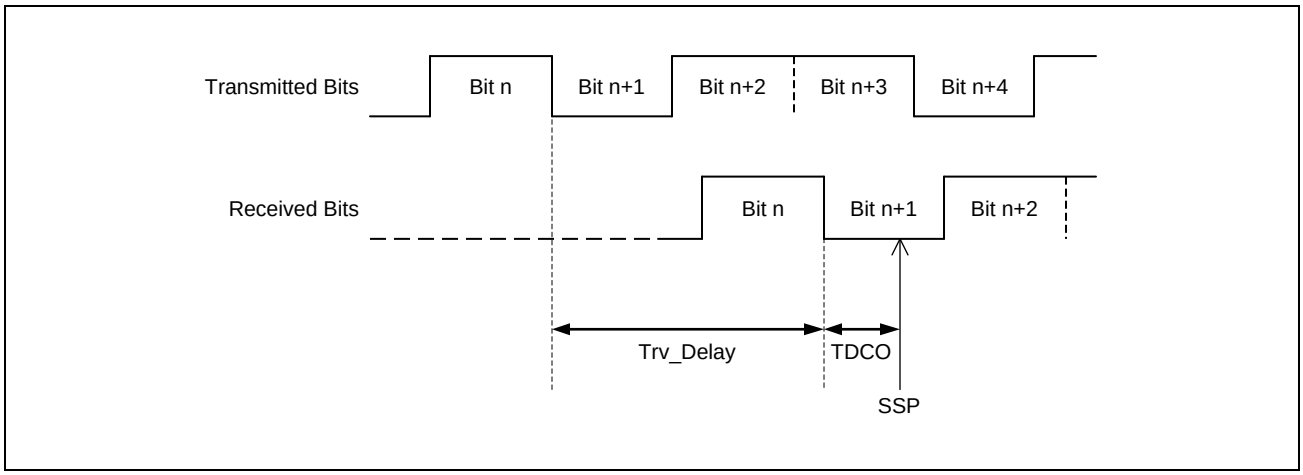


Figure 30.20 Position of the Secondary Sample Point

The maximum delay (Trv_Delay + TDCO) which can be compensated by the CAN-FD module is (6 data bits - 2clk_dlc).

The ISO 11898-1 allows you to set different values for BRP_data and BRP_nom.

If different values are used for CFDCnNCFG.NBRP and CFDCnDCFG.DBRP, then two CAN nodes may be out of synchronization at the point when the bit rate changes from nominal bit rate to data bit rate after sample point of the BRS bit. This condition is shown in **Figure 30.21**.

The length of the time quantum should be the same in the nominal bit time and in the data bit time. This means CFDCnNCFG.NBRP = CFDCnDCFG.DBRP.

Different bit rates can be achieved by selecting different configuration values for the Time Segments. The nominal bit rate can be configured from 8 to 385 TQs and the data bit rate from 5 to 49 TQs.

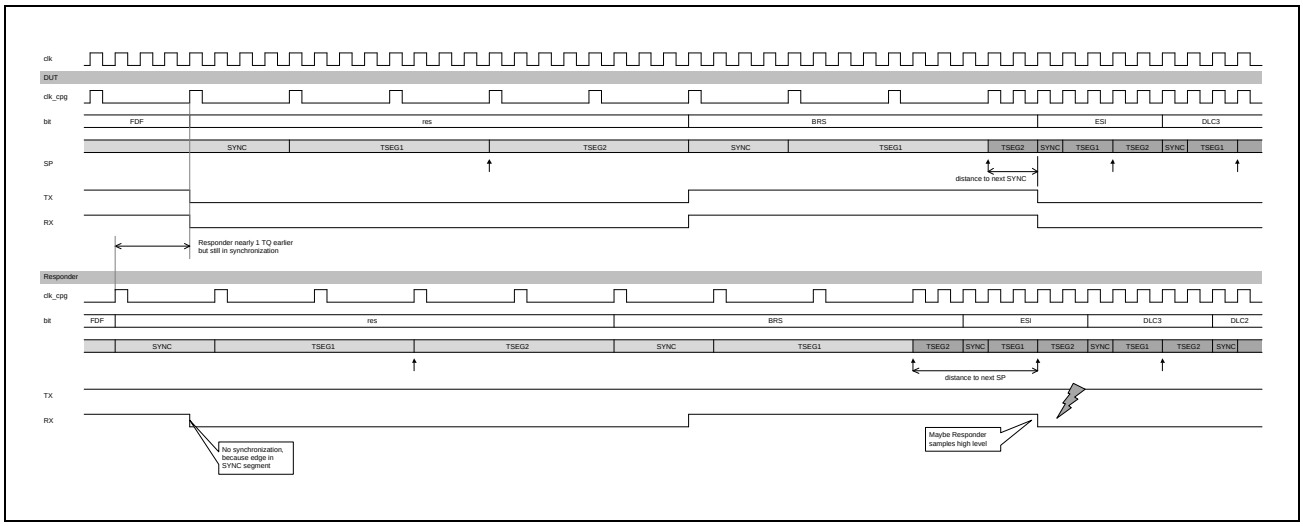


Figure 30.21 Loss of Synchronization between 2 CAN Nodes

The transmitter delay compensation measurement result is updated at the falling edge from FDF bit to RES bit when configured accordingly (CFDCnFDCFG.TDCE = 1, CFDCnFDCFG.TDCOC = 0).

Figure 30.22 shows the read flow to get the measured transmitter delay compensation result.

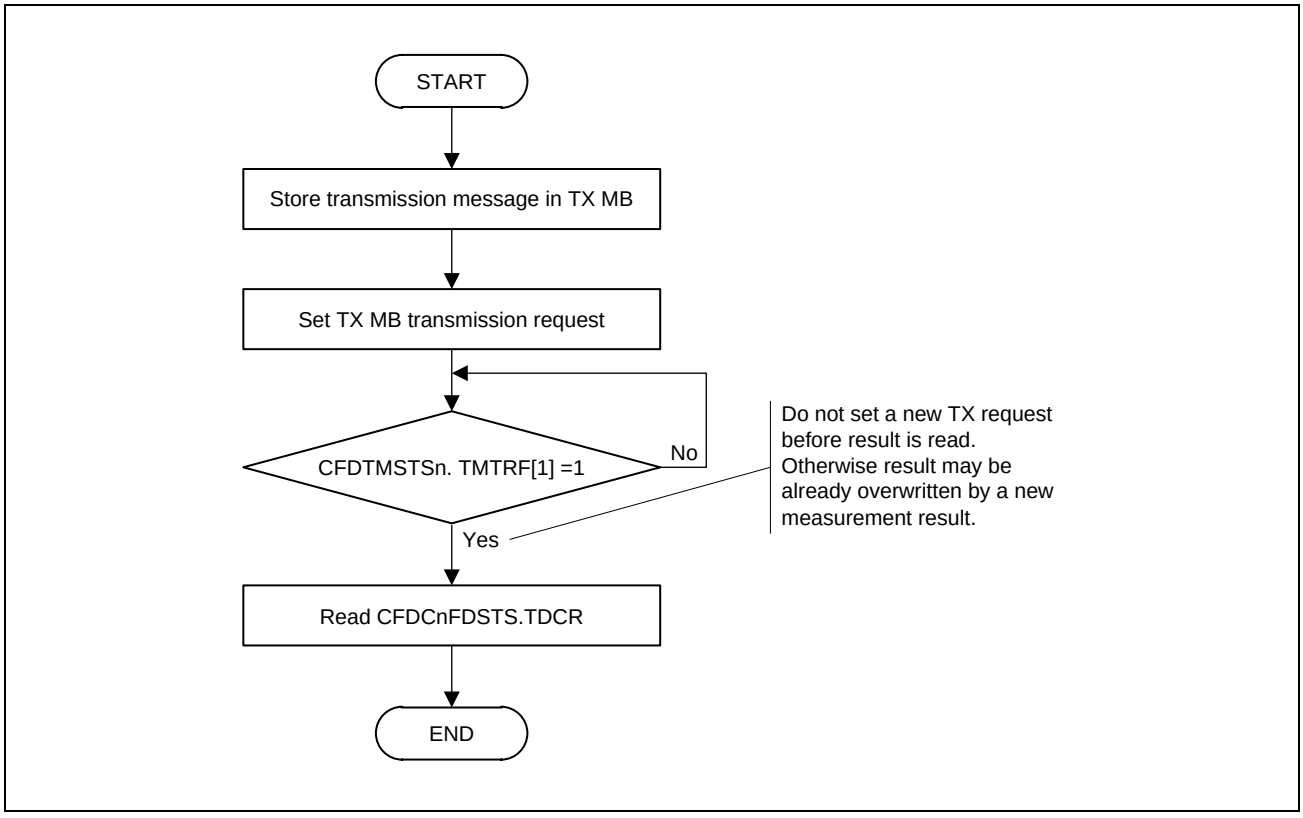


Figure 30.22 TDC Result Read Flow

30.4.2 CAN Module Configuration after Hardware Reset

After a hardware reset (power on reset) or after setting and clearing the CFDGRSTC.SRST bit, the CAN-FD module enters Global Sleep mode automatically.

To enable configuration of the CAN-FD module, you must exit Sleep mode by clearing the Global Sleep Request bit CFDGCTR.GSLPR to 0.

After a hardware reset, the module starts RAM initialization, the CFDGSTS.GRAMINIT bit in the Global Status Register is set automatically to indicate that the CAN-FD logic is initializing the RAM.

After RAM initialization is complete, this bit is cleared automatically.

RAM initialization is necessary to avoid setting of false ECC error flag after hardware resets the random data presented in the RAM.

Do not access registers of the CAN-FD in either read or write until RAM initialization is complete and the CFDGSTS.GRAMINIT bit is cleared.

Before going to communication mode, the Global Acceptance Filter List and message FIFO buffers must be configured. In addition, each required CAN channel must be configured such as CAN bit timing. For this configuration, all required CAN channels must be released from Channel Sleep mode and must be configured for communication in Channel Reset mode (Configuration mode).

For this, all required CAN channels must be released from Channel Sleep mode and must be configured for communication in Channel Reset mode (Configuration mode).

Figure 30.23 shows the configuration procedure. For details about each step, see **Section 30.5, Acceptance Filtering Function using Global Acceptance Filter List (AFL)**, **Section 30.6, FIFO Buffers and Normal Message Buffer Configuration**, **Section 30.7, Interrupt and DMA**, and **Section 30.4.1.3, Baud Rate**.

The CAN-FD module does not perform RAM initialization sequence after executing a software reset by setting CFDGRSTC.SRST.

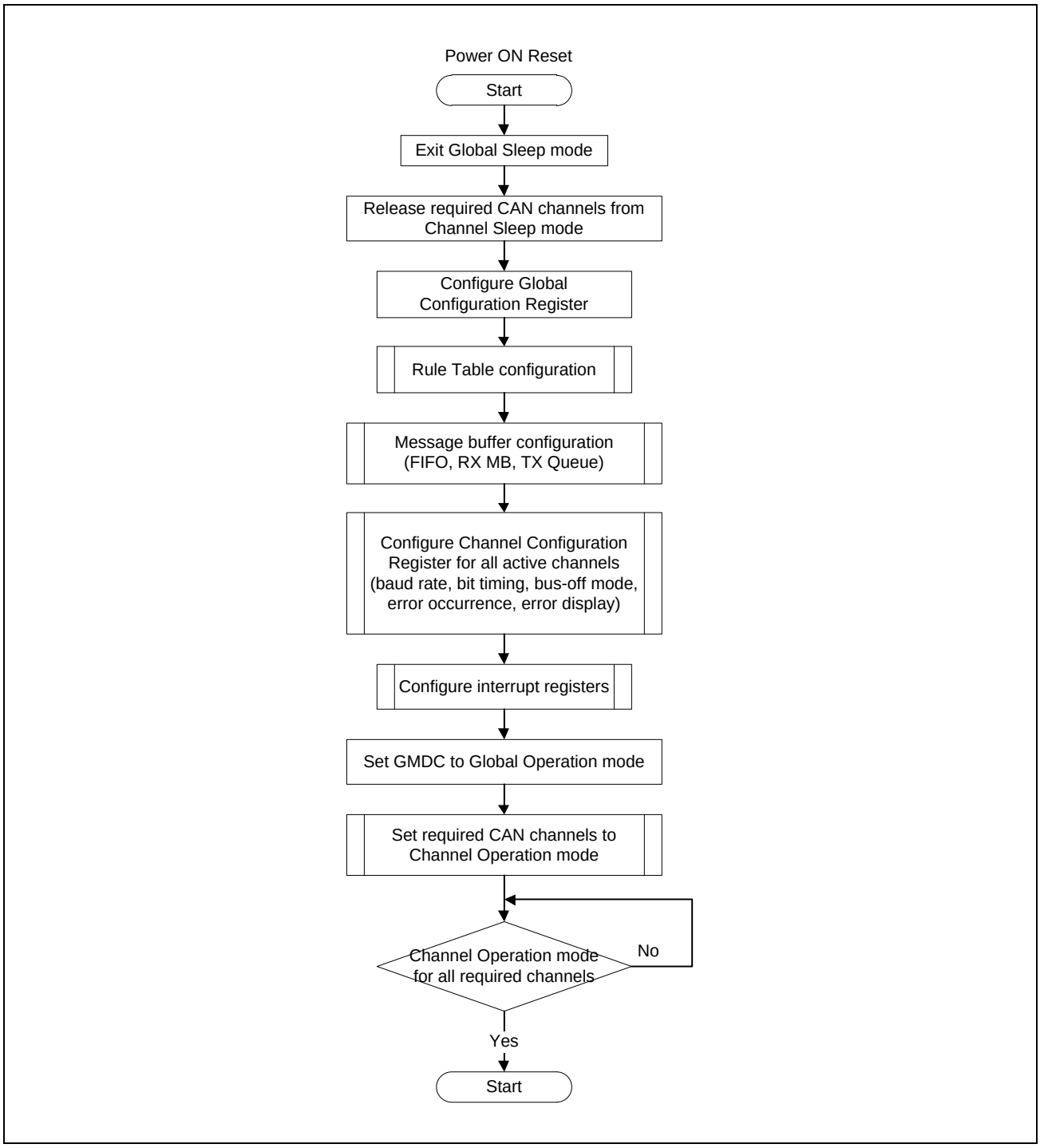


Figure 30.23 Configuration Procedure after a Hardware Reset

30.5 Acceptance Filtering Function using Global Acceptance Filter List (AFL)

30.5.1 Overview

The CAN-FD module can handle message acceptance filtering for all channels with a global Acceptance Filter List (AFL). Each element of the AFL defines a filter rule for messages received on a specific channel.

The following actions are performed based on the AFL entries:

- Acceptance filtering based on received CAN identifier and masking
- DLC filtering based on received DLC value
- Message data payload according to the CFDGCFG.CMPOC bit
- Storage of accepted messages in the message buffer objects defined in the related AFL entry
- Attaching a 16-bit pointer to the stored messages defined in the related AFL entry, for example to support AUTOSAR applications
- Attaching a 2-bit information label to the stored messages defined in the related AFL entry

The 2-channel CAN-FD module allows a maximum of 256 AFL entries across all channels with a maximum of 256 AFL entries per channel.

During the acceptance filtering process, each AFL entry in a channel is checked against the received message by the acceptance filter unit. The check starts from the lowest AFL entry number for this channel.

The AFL search stops when a match of the received identifier with a configured identifier/mask combination occurs or when the received identifier has been compared against all AFL entries defined for the related channel. If no match occurs, then the received message is rejected. No notification is given to the application in this case.

Additionally, an automatic DLC filtering is performed for each accepted message if DLC check is globally enabled. If the DLC value of the received message is equal to or higher than the configured DLC value in the matching AFL entry, the DLC check is passed.

If DLC replacement (CFDGCFG.DRE bit) is enabled, DLC value configured in the matching AFL entry is greater than H’0 and DLC check passes, then the configured value of DLC in the matching AFL entry is stored in the destination RXMB or FIFO Buffer.

If the received value of DLC is greater than the configured DLC value in the matching AFL entry, then the additional data bytes received on the CAN bus are not stored in the destination RXMB or FIFO Buffer. These additional data bytes are stored as H’00 in the destination RXMB or FIFO Buffer.

If DLC replacement is enabled and DLC value of matching AFL entry is H’0, then the received value of DLC is stored in the destination RX MB or FIFO Buffer.

If DLC replacement (the CFDGCFG.DRE bit) is disabled and DLC check passes, then the received value of DLC on the CAN bus is stored in the destination RXMB or FIFO buffer.

If the received value of DLC is greater than the configured DLC value in the matching AFL entry, then the additional data bytes received from the CAN bus are also stored in the destination RXMB or FIFO buffer.

If DLC value of the received message is less than the configured DLC value in the matching AFL entry, then DLC check fails. In this case, the received message is rejected and is not stored in any RXMB or FIFO buffer.

Additionally, DLC check failure is flagged by the DLC Error Flag in the Global Error Flag Register. If configured, an error interrupt is also generated. The DLC replacement configuration has no impact if the DLC check fails.

If a message has passed both acceptance filtering and DLC filtering, it is stored in a single reception message buffer and/or in FIFO buffers configured for reception or gateway function.

This message storage target information is also defined in the same AFL entry. Do not set a target at the AFL entry which is not configured.

Each accepted received message can be stored into a maximum of 8 different target destinations (single reception message buffer and/or FIFO buffers).

The programming of more than eight target destinations is not allowed. When more destinations are programmed then internal timing, a race condition can occur and received message may not be stored to the message RAM. Correct configuration of the numbers of target destination is the responsibility of the application.

Additional protection mechanism is made for the case when a received message contains more data payload bytes than possible to store in the target destination (CFDRMNB.RMPLS[2:0], CFDRFCCn.RFPLS[2:0] or CFDCFCCn.CFPLS[2:0]).

If CFDGCFG.CMPOC = 0, the message is completely rejected and is stored in the target destination. When CFDGCFG.CMPOC = 0 and RX or Common FIFO full including the received message contains more data payload bytes than possible to store in the target destination (CFDRMNB.RMPLS[2:0], CFDRFCCn.RFPLS[2:0] or CFDCFCCn.CFPLS[2:0]), the corresponding CFDFMSTS.RFXMLT[7:0] or CFDFMSTS.CFXMLT[5:0] bits are not set to 1, respectively.

When CFDGCFG.CMPOC = 1, the received data bytes greater than CFDRMNB.RMPLS is rejected. When CFDGCFG.CMPOC = 1 and RX or Common FIFO full including the received message contains more data payload bytes than possible to store in the target destination (CFDRMNB.RMPLS[2:0], CFDRFCCn.RFPLS[2:0] or CFDCFCCn.CFPLS[2:0]), the corresponding CFDFMSTS.RFXMLT[7:0] or CFDFMSTS.CFXMLT[5:0] bits are set to 1, respectively.

Depending on the CFDGCFG.DRE bit, the original received DLC or the DLC value configured at the AFL entry is stored.

Regardless of the CFDGCFG.CMPOC bit setting, CFDGERFL.CMPOF is set to 1 if a payload overflow condition is detected.

The DLC filtering is performed before the payload overflow function. So for one reception frame, only one flag can be set at the same time with CFDGERFL.DEF or CFDGERFL.CMPOF.

30.5.2 Allocation of AFL Entries to Each CAN Channel

The number of AFL entries per channel can be configured using the dedicated field in the related Global Acceptance Filter Configuration Registers (see **Figure 30.24**).

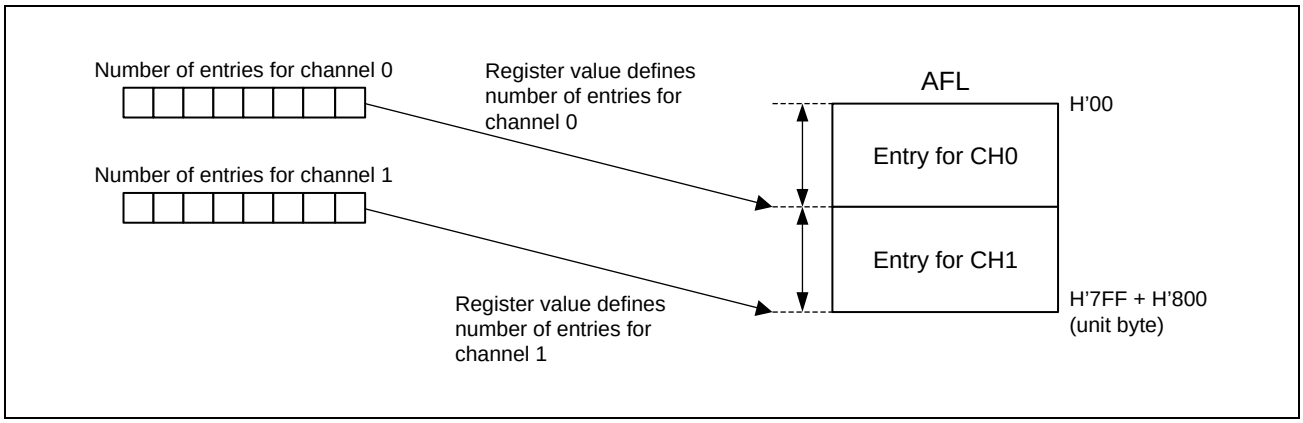


Figure 30.24 Configuration of AFL for Each Channel

The minimum number of entries for one channel is 0 (no entries defined for the channel) and the maximum number of entries for one channel is 256. The total number of entries for all channels should not exceed the maximum limit of (n + 1) × 128.

All entries are unique for a channel and overlapping or sharing of entries is not supported. Correct configuration of the AFL is the responsibility of the application.

The CAN-FD module does not flag errors related to the configuration of the AFL.

30.5.3 AFL Entry Description

Each AFL entry consists of 16 bytes. The fields in all entries are identical.

Each entry contains the following information for acceptance filtering and DLC filtering:

- Identifier (11 bits for Standard Frame format, 29 bits for Extended Frame format):
  Acceptance filter unit checks the identifier field of the received message against the identifier field of each AFL entry (full 29 bits masking of identifier bits is possible, see information that follows).
- IDE bit:
  Acceptance filter unit checks the IDE bit of the received message against this bit and selects the relevant part of the identifier field for acceptance filtering (masking of IDE bit is possible, see the information that follows).
- RTR bit:
  Acceptance filter unit only accepts data frames (RTR = 0) or remote frames (RTR = 1) according to the setting of this bit (masking of RTR bit is possible, see the information that follows).
- Loopback Configuration bit:
  This bit can enable or disable the AFL entry depending on the Loopback Configuration or Mirror mode condition.
- Mask for Identifier bits (29 bits):
  Each bit in the identifier mask can mask the corresponding identifier bit in the AFL entry during acceptance filtering, see **Figure 30.25**.
- Mask for IDE bit:
  If this Mask bit masks the IDE bit of the AFL entry in both Standard Identifier and Extended Identifier format, messages can be accepted by this AFL entry. The identifier of the received message is compared against the Standard Identifier part of the AFL entry for Standard Identifier format messages and against the Extended Identifier part of the AFL entry for Extended Identifier format messages.
- Mask for RTR bit:
  If this Mask bit masks the RTR bit of the AFL entry in both frame formats, data frame and remote frame formats are accepted by this AFL entry.
- Pointer information (16 bits):
  This 16-bit pointer is attached to a received message accepted by the related AFL entry. The pointer is added during message storage in the message buffer area and can be used by application as support function. The pointer information can be used for example to support PDU identifier allocation for the received message in AUTOSAR systems.
- Information label (2 bits):
  This 2-bit label is attached to a received message accepted by the related AFL entry. The label is added during message storage in the message buffer area and can be used by application as support function.
- DLC value for automatic DLC filtering:
  If the DLC value of the received message is equal or higher than the configured DLC value, the DLC check is passed. If the DLC value in this AFL entry is configured to 0, DLC filtering is effectively disabled for this entry (all accepted messages pass DLC filtering).

Each AFL entry contains the following information for the handling of received messages:

- Message buffer number of one single reception message buffer as target for received message storage
- Single reception message buffer enable bit to configure the single reception message buffer number to be valid or invalid, as target for received message storage
- FIFO direction pointer - each bit of the FIFO direction pointer configures a dedicated FIFO as possible target for a received message

Note: A message received on channel A can be routed to Common FIFO buffer of another channel. If this Common FIFO buffer is configured in Gateway mode, then the message stored in this Common FIFO Buffer is transmitted on that channel because Common FIFO buffer is associated with channel.

There is no hardware protection against such storage of message. Therefore, the FIFO direction pointer must be configured carefully.

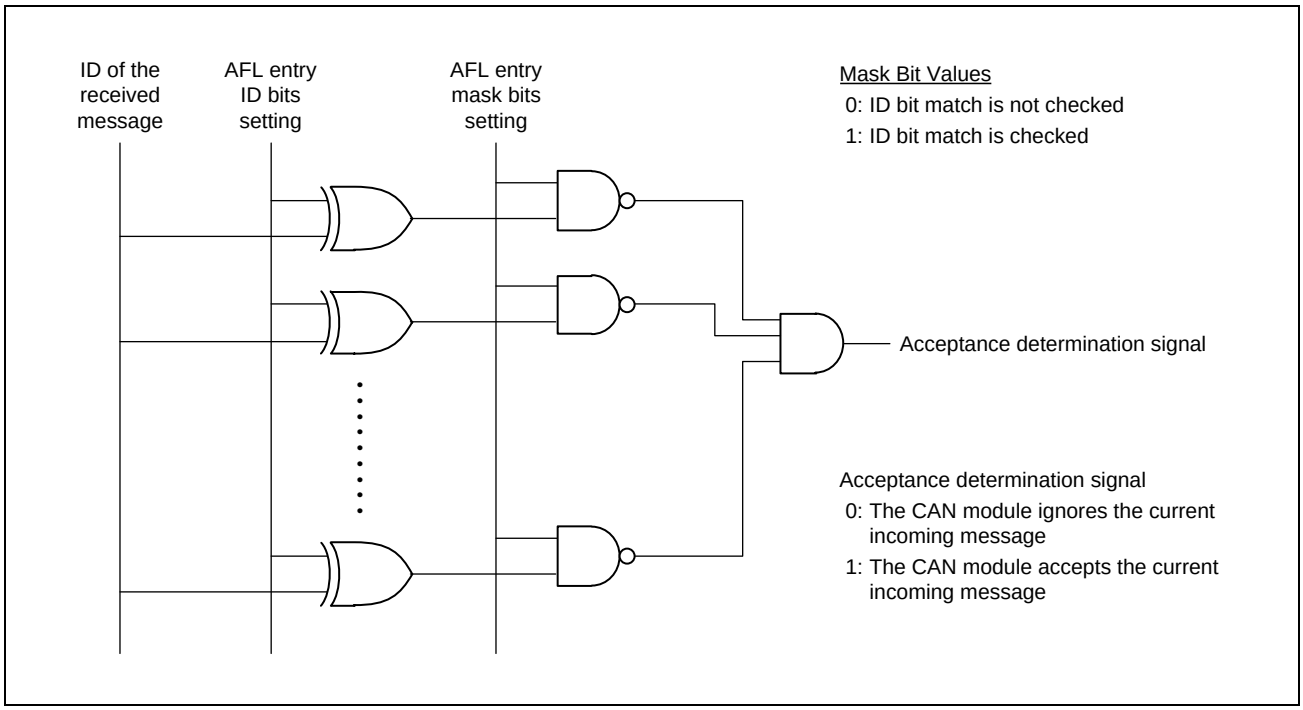


Figure 30.25 Acceptance Function

30.5.4 Entering Entries in the AFL

Application software can enter one full entry into the AFL using the following registers:

- Global AFL ID Entry Register: Part 1 of the AFL entry
- Global AFL Mask Entry Register: Part 2 of the AFL entry
- Global AFL Pointer 0 Entry Register: Part 3 of the AFL entry
- Global AFL Pointer 1 Entry Register: Part 4 of the AFL entry

16 sets of these registers form a group of AFL entries. Each group can be accessed through a page mechanism. For the CAN-FD module, 2-channel version, 16 of these pages exist to allow access to the whole AFL range. The AFL should only be configured in CH_RESET or CH_HALT mode. Pages are linked to the AFL entries in the following way:

Table 30.32 Mapping between Pages and AFL Entries

Page 0	Entry 0 - 15
Page 1	Entry 16 - 31
Page 2	Entry 32 - 47
Page 3	Entry 48 - 63
Page 4	Entry 64 - 79
Page 5	Entry 78 - 95
Page 6	Entry 96 - 111
Page 7	Entry 112 - 127
Page 8	Entry 128 - 143
Page 9	Entry 144 - 159
Page 10	Entry 160 - 175
Page 11	Entry 176 - 191
Page 12	Entry 192 - 207
Page 13	Entry 208 - 223
Page 14	Entry 224 - 239
Page 15	Entry 240 - 255

The selection of the AFL access page is done using the Global Acceptance Filter List Entry Control Register (CFDGAFLECTR) (see **Figure 30.26**). This register has the following fields:

- 4 bits to select the AFL page number
- 1 bit to enable or disable the AFL data access to prevent unwanted write access to the AFL

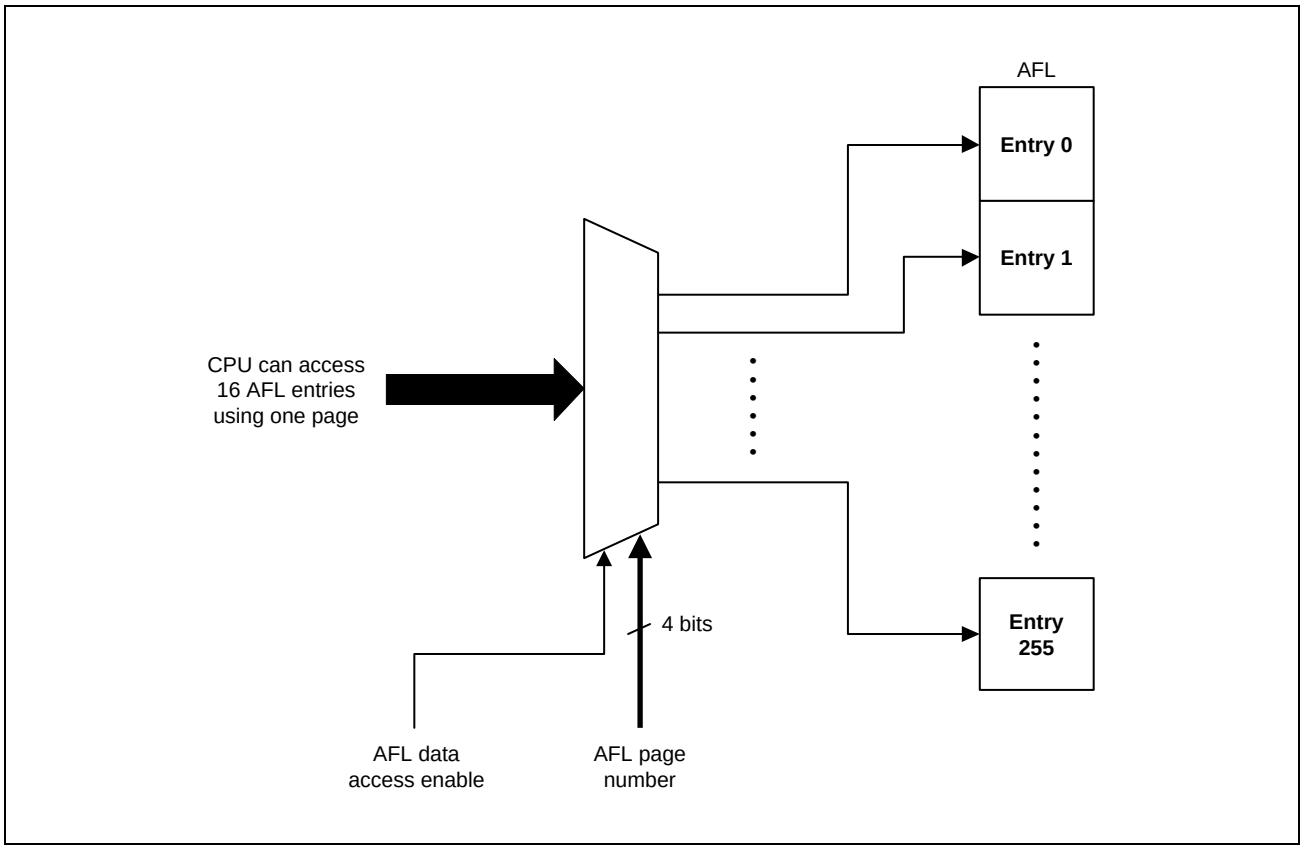


Figure 30.26 AFL Page Access

Application software should not write numbers higher than H’F for the AFL page number.

Follow the configuration shown in **Figure 30.27** to program the AFL.

After entering all entries in Configuration mode, locking of the AFL access should be performed to protect unwanted write access to the AFL.

Write protection is active during all Global modes (GL_RESET, GL_HALT, and GL_OPERATION) if the lock bit is set.

Read access to AFL is still possible during all Global modes even when AFL data access is disabled (consistency check of AFL contents is possible during run time).

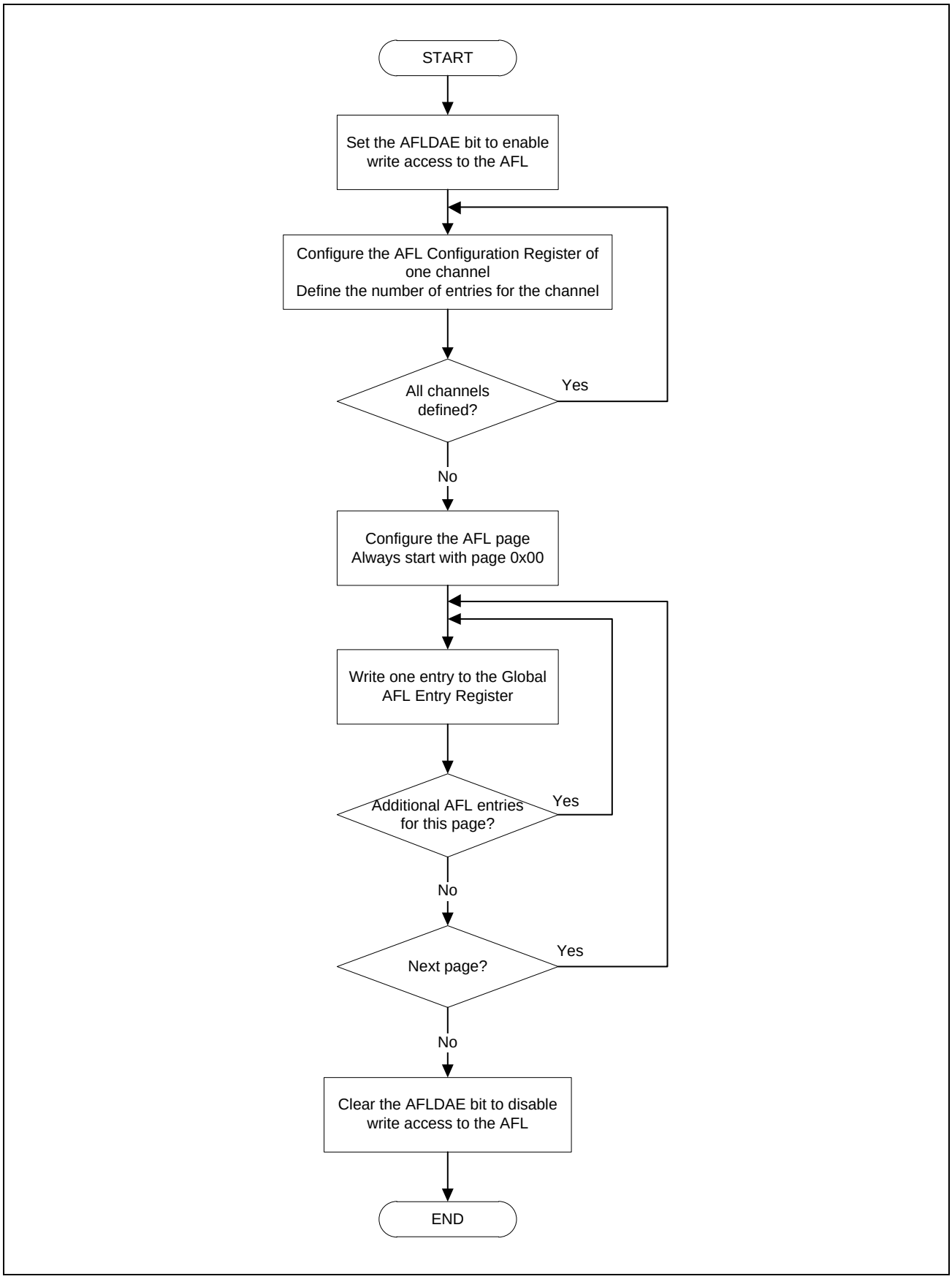


Figure 30.27 AFL Configuration Flow

30.5.5 Loopback Modes

If the Loopback Configuration bit is set, the AFL entry is only valid in loopback test mode (Self-test mode 0 or Self-test mode 1) or in mirror mode when receiving messages that were transmitted by the respective CAN channel itself.

The AFL entry is not valid for received messages in loopback mode transmitted by other CAN nodes on the bus. The expression valid or invalid for the related entry means that this AFL entry is or is not compared against the received message ID respectively.

If the Loopback Configuration bit is 0, the AFL entry is only valid for:

- Received messages transmitted by other CAN nodes on the bus in normal (non-loopback mode) and mirror modes
- Received messages transmitted by other CAN nodes or the CAN channel itself in loopback test mode.

The mirror mode can be enabled with the CFDGCFG.MME bit in the Global Configuration Register. If CFDGCFG.MME bit is set, then a successfully transmitted message can be stored back in an RX message buffer or FIFO buffer if a matching entry is configured in the AFL for that channel.

The Loopback Configuration bit in the matching AFL entry must be set to store this frame.

If mirror mode and loopback test mode are configured at the same time, the loopback test mode behavior applies.

Table 30.33 shows the behavior of the acceptance filter unit depending on the setting of the related input signals.

Table 30.33 Behavior of Acceptance Filter Based on the Loopback Configuration Setting in AFL Entry

Mirror Mode Enable (MME Configuration Bit)	Loopback in Test Mode (Self-test mode 0 or Self-test mode 1)	Channel Mode	Loopback Configuration Bit in AFL Entry	AFL Entry
0	0	Receiver	0	Valid
			1	Invalid
		Transmitter	0	Invalid
			1	Invalid
	1	Receiver	0	Valid
			1	Invalid
		Transmitter	0	Valid
			1	Valid
1	0	Receiver	0	Valid
			1	Invalid
		Transmitter	0	Invalid
			1	Valid
	1	Receiver	0	Valid
			1	Invalid
		Transmitter	0	Valid
			1	Valid

Note: The expression valid or invalid for the related entry means that this AFL entry is or is not compared against the received message ID, respectively.

30.5.6 IDE Masking

When the GAFLIDEM bit is 0 in an AFL entry, the IDE bit configured in the AFL entry is not used for ID matching. In this case, the use of ID[10:0] or ID[28:0] matching is based on the received IDE bit.

Consider the following example:

- The ID and Mask fields of an AFL entry x is configured as follows:
  - CFDGAFLID [x] = H’C055_3A20 → IDE = 1, RTR = 1, LLB = 0, ID[10:0] = H’220/ID[28:0] = H’055_3A20
  - CFDGAFLMn = H’0000_FFFF → IDEM = 0, RTRM = 0, IDM[10:0] = H’7FF/IDM[28:0] = H’0000_FFFF
- The comparison result for the four different received IDs with AFL entry x is described as follows:
  - If a frame with IDE = 0 and ID = H’220 is received, this is considered as a match
  - If a frame with IDE = 0 and ID = H’320 is received, this is not a match
  - If a frame with IDE = 1 and ID = H’1FFF_3A20 is received, this is considered as a match
  - If a frame with IDE = 1 and ID = H’0880_3220 is received, this is not a match.

30.6 FIFO Buffers and Normal Message Buffer Configuration

This section describes the process for configuring the number of RX message buffers, the FIFO buffers, and the flat TX message buffers in the CAN-FD module. The message buffers are mapped as shown in **Figure 30.28**.

The RX message buffers can be accessed with the RX Message Buffer Registers.

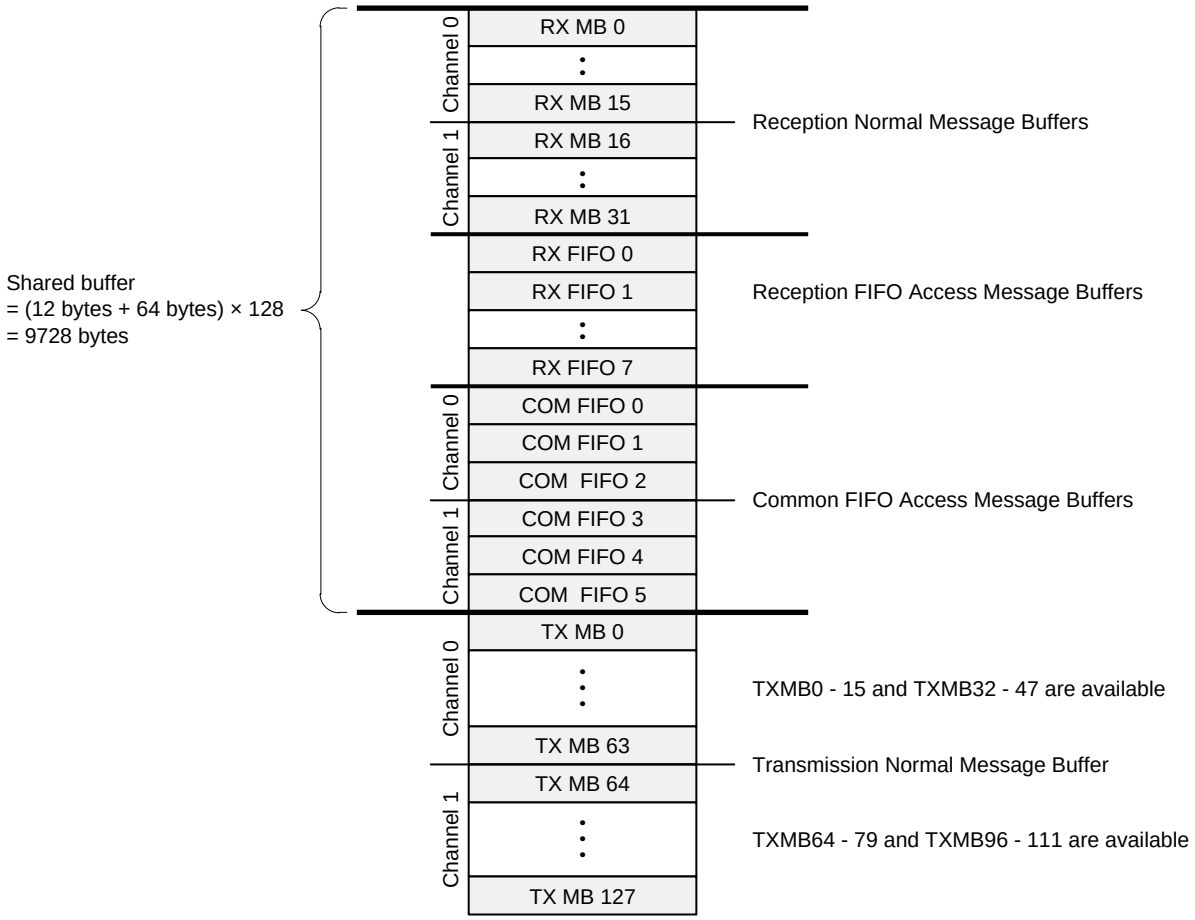
The RX FIFO buffers and the common FIFO buffers configured in RX mode, TX mode, or GW mode can only be accessed with the FIFO Access Registers.

If the common FIFO is configured in TX mode, you can only write data into the FIFO buffer using the FIFO Access registers.

If the common FIFO is configured in GW mode or RX mode, you can only read data from the FIFO Access Registers.

The TX message buffers can be accessed with the TX Message Buffer Registers.

If unused message buffer locations are read, the message buffer locations are read as unknown values.



Note: The total number of bytes in RXMB_cfg, RXFIFO_cfg, and CFIFO_cfg are less than or equal to 9728 bytes.
RXMB_cfg = (12 bytes + CFDRMND.RMPLS) × CFDRMNB.NRXMB
RXFIFO_cfg = SUM ((12 bytes + CFDRFCCn.RFPLS) × CFDRFCCn.RFDC)
CFIFO_cfg = SUM ((12 bytes + CFDCFCCn.CFPLS) × CFDCFCCn.CFDC)

Figure 30.28 Message Buffer Configuration

30.6.1 Normal RX Message Buffers

In CAN-FD module, the frames received by various channels can be stored in normal RX message buffers based on the configuration of the AFL entries.

Additionally, the number of normal RX message buffers required in the system can be chosen up to a fixed maximum limit.

30.6.1.1 Normal RX Message Buffer Configuration

In CAN-FD module, the number of normal RX message buffers can be configured by writing to the RX Message Buffer Number Register.

The limiting values for the configuration of number of message buffers are:

- Minimum value = H’00 (no normal RX MB)
- Maximum value = (16 × number of CAN channels) = 16 × 2 = 32 = H’20 (32 flat RX MBs for 2 channels)

Do not use values outside these limits.

The AFL entries for routing the received messages to normal RX message buffers must be configured to match the requirements of the system.

The AFL entries must also be configured properly, and an AFL entry for normal RX message buffers should not exceed the number of message buffers configured in the RX Message Buffer Number Register.

Note: There is no internal check procedure provided in CAN-FD module against incorrect configuration of the AFL.

The data field size of the RX message buffer can be configured with the CFDRMNB.RMPLS[2:0] bits. The default size is 8 bytes and the maximum data payload size is 64 bytes.

When the receiving frame exceeds the data field size, then the acceptance depends on the configuration of CFDGCFG.CMPOC (message rejecting or data payload cut).

30.6.2 FIFO Buffer

The CAN-FD module provides a fixed number of FIFO buffers to support storage of frames for reception, transmission and gateway functions for various CAN channels.

The number of reception-only FIFO buffers is fixed to 8. However, 3 common FIFO buffers per channel can be configured to store messages for transmission, reception, or gateway function.

These FIFO buffers can be enabled or disabled, and the following parameters can be configured to match the system requirements:

- Size
- Interrupt structure
- Message lost mechanism
- Message overwrite mechanism of the FIFO buffers
- Location of the TX FIFO or GW FIFO

When the receiving frame exceeds the data field size, the acceptance depends on the configuration of the CFDGCFG.CMPOC bit (message rejecting or data payload cut).

30.6.2.1 FIFO Buffers Configuration

In CAN-FD module, the FIFO buffers can be configured to match the system requirements.

The total number of FIFO buffers = 8 RX FIFO buffers + 6 common FIFO buffers = 14 FIFO buffers for 2 channels and message overwrite mechanism.

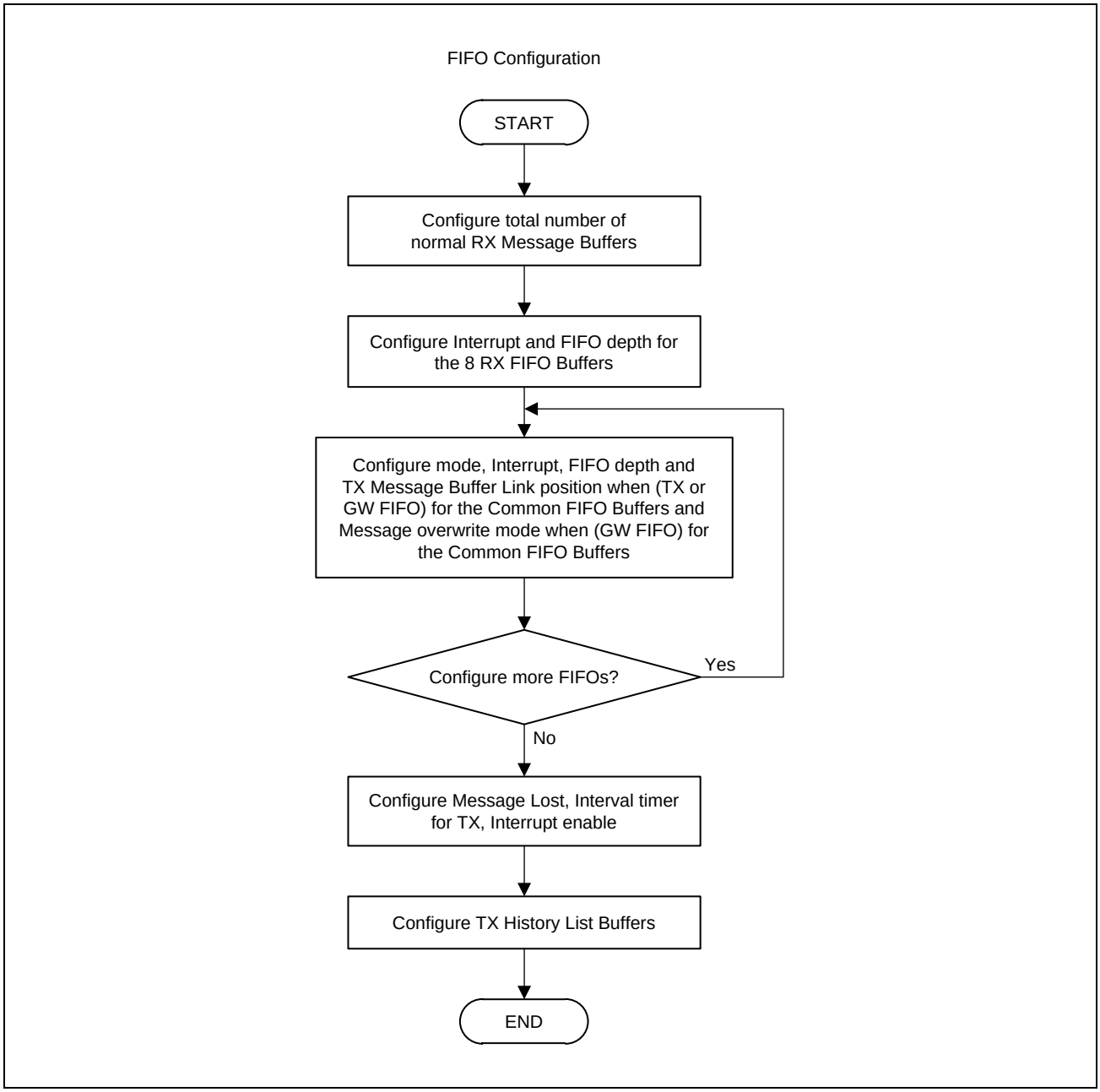


Figure 30.29 FIFO Buffer Configuration Flow in CAN-FD Module

As shown in **Figure 30.29**, the various FIFO buffers can be configured by writing to the RX FIFO Configuration/Control Registers and the Common FIFO Configuration/Control Registers.

For the 8 RX FIFO buffers, the following parameters can be configured:

- Interrupts
- FIFO depth

- FIFO payload data size

For the common FIFO buffers, the following parameters can be configured:

- Mode
- Interrupts FIFO depth
- FIFO payload data size
- FIFO TX link position

(1) FIFO Mode Configuration of Common FIFO Buffers

The mode of the common FIFO buffers can be configured by writing to the CFDCFCCn.CFM[1:0] bits in the Common FIFO Configuration/Control Registers. The possible modes of configuration for Common FIFO buffers are:

- 00b: RX mode (default mode after hardware reset)
- 01b: TX mode
- 10b: GW mode
- 11b: Reserved (do not write this value to the register bits)

Messages can only be read from the RX FIFO buffers and the Common FIFO buffers configured in RX mode. Messages are stored by the CAN module in these FIFO buffers based on the AFL entries.

Messages can be read and written into the Common FIFO buffers configured in TX mode. These messages are transmitted on the appropriate CAN channel.

Messages can only be read from the Common FIFO buffers configured in GW mode. However, the CPU read access has no impact on the read or write pointers. The pointers can only be incremented when a new message is stored in the FIFO buffer and decremented when a message is transmitted on the corresponding CAN channel by the CAN-FD module.

After a hardware reset, all the Common FIFO buffers are configured in RX mode by default. Only enable the FIFO buffers after configuring the Common FIFO buffers in the required modes.

In GW mode, when a transmit/receive FIFO buffer is trying to receive a new message while the transmit/receive FIFO buffer is already full of data, the oldest data of the buffer is overwritten with the message received or the message is discarded. The behavior is determined by setting the CFDCFCCEn.CFMOWM bit.

- When CFDCFCCEn.CFMOWM = 0:
  When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the received message is discarded, and CFDCFSTSn.CFMLT bit is set to 1.
- When CFDCFCCEn.CFMOWM = 1:
  When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the oldest data in the buffer is overwritten with the received message. The read pointer of the transmit/receive FIFO buffer simultaneously moves to the next oldest message. The CFDCFSTSn.CFMOW bit is then set to 1, which notifies that the oldest message has been overwritten with the received message.
  In addition, when a CAN bus error or arbitration-lost for the transmitting message occurs in the transmit/receive FIFO buffer full, the transmitting message is lost and retransmission for the message is not performed. The read point is then moves to the next message automatically. Do not write to this bit when the CFDCFCCn.CFE bit is 1.

(2) FIFO TX Message Buffer Link Configuration

When the common FIFO is configured as TX or GW FIFO, the FIFO buffer must be linked to a normal TX message buffer to participate in the transmission scan of a CAN channel.

The link to a normal TX message buffer must be unique, for example the same TX message buffer cannot be shared between 2 or more common FIFO buffers.

Do not write data into a TX message buffer that is linked to a Common FIFO buffer. Also, the TX message buffer linked to a Common FIFO buffer should not be a part of the TX Queue.

The TX message buffer link of each Common FIFO buffer can be configured by writing to the CFDCFCCn.CFTML[4:0] bits in the Common FIFO Configuration/Control Registers. Available options for TX message buffer link configuration are:

- H’00: TX Message Buffer 32
- H’01: TX Message Buffer 33
  ⋮
- H’1F: TX Message Buffer 63

(3) FIFO Depth Configuration

The depth of each FIFO buffer can be configured by writing to the CFDRFCCn.RFDC[2:0] bits and CFDCFCCn.CFDC[2:0] bits in the RX FIFO Configuration/Control Registers and the Common FIFO Configuration/ Control Registers. The 8 available options for depth configuration are:

- 000b: 0 Messages (FIFO buffer cannot be enabled)
- 001b: 4 Messages
- 010b: 8 Messages
- 011b: 16 Messages
- 100b: 32 Messages
- 101b: 48 Messages
- 110b: 64 Messages
- 111b: 128 Messages

The RAM allocation for RX message buffers along with FIFO buffers is limited to (n + 1) × 256 messages. Configuration of the RX message buffers, along with FIFO buffers, that exceeds this maximum limit should not be done.

CAN-FD module logic does not check the validity of the configuration.

Note: If the FIFO depth of a common FIFO is 4 messages or more (CFDCFCCn.CFDC[2:0] > 000b), then the Common FIFO TX message buffer link is valid when the FIFO is disabled or enabled.

Note: If FIFO depth is 0 messages, then the Common FIFO TX message buffer link is not valid when the FIFO is disabled or enabled.

(4) FIFO Payload Size Configuration

The data size of each FIFO buffer can be configured by writing to the CFDRFCCn.RFPLS[2:0] bits and CFDCFCCn.CFPLS[2:0] bits in the RX FIFO Configuration/Control Registers and the Common FIFO Configuration/ Control Registers. The eight available options for depth configuration are:

- 000b: 8 bytes

- 001b: 12 bytes
- 010b: 16 bytes
- 011b: 20 bytes
- 100b: 24 bytes
- 101b: 32 bytes
- 110b: 48 bytes
- 111b: 64 bytes

(5) FIFO Interrupt Configuration

The interrupt generation conditions for the FIFO buffers can be configured by writing to the CFDRFCCn.RFIM bit and the CFDCFCCn.CFIM bit in the RX FIFO Configuration/Control Registers and the Common FIFO Configuration/Control Registers. The two available options are:

- 0:
  - RX FIFO mode: Interrupt generated when the Common FIFO counter reaches CFDRFCCn.RFIGCV/ CFDCFCCn.CFIGCV value
  - TX FIFO mode: Interrupt generated when the Common FIFO transmits the last message successfully
  - GW FIFO mode
    Frame RX: Interrupt generated when message counter increments and reaches the interrupt threshold value
    Frame TX: Interrupt generated when the last message is transmitted successfully from FIFO
- 1:
  - RX FIFO mode: Interrupt generated at the end of storage of every received message
  - TX FIFO mode: Interrupt generated for every successfully transmitted message
  - GW FIFO mode
    Frame RX: Interrupt generated when message is stored in the FIFO
    Frame TX: Interrupt generated when message is successfully transmitted from the FIFO

If the Interrupt Mode bit is 0 for a RX FIFO, then interrupt is generated based on the configuration of the CFDRFCCn.RFIGCV[2:0] bits.

Similarly, if the Interrupt Mode bit is 0 for a Common FIFO configured in RX mode, then interrupt is generated based on the configuration of CFDCFCCn.CFIGCV[2:0] bits.

The eight available options for configuring the FIFO counter value for generation of an interrupt are:

- 000b: Interrupt generated when FIFO is 1/8th Full
- 001b: Interrupt generated when FIFO is 1/4th Full
- 010b: Interrupt generated when FIFO is 3/8th Full
- 011b: Interrupt generated when FIFO is 1/2 Full
- 100b: Interrupt generated when FIFO is 5/8th Full
- 101b: Interrupt generated when FIFO is 3/4th Full
- 110b: Interrupt generated when FIFO is 7/8th Full
- 111b: Interrupt generated when FIFO is Full

In this case, an interrupt is generated when the message count matches the configured value.

However, there are some limitations on the configuration of the CFDRFCCn.RFIGCV[2:0] and CFDCFCCn.CFIGCV[2:0] bits depending on the FDC[2:0] bits (FIFO Depth Configuration), see **Table 30.34**.

Table 30.34 FIFO Interrupt Generation Counter and FIFO Depth Configuration

RFDC[2:0] (FDC[2:0])	RFIGCV[2:0] (CFGCV[2:0])							
	111b	110b	101b	100b	011b	010b	001b	000b
000b	Don't care (FIFO cannot be enabled)							
001b	Allowed	Not allowed	Allowed	Not allowed	Allowed	Not allowed	Allowed	Not allowed
010b	Allowed							
011b	Allowed							
100b	Allowed							
101b	Allowed							
110b	Allowed							
111b	Allowed							

Common FIFO buffer can set an interrupt output at the completion time of transmitting one frame, or the completion of reception. In addition, Common FIFO and RX FIFO can set an interrupt output, when stored to the setup number (CFDC/ RFDC) of FIFO stages.

30.6.2.2 FIFO Buffers Control

The FIFO interrupt must be enabled by setting any one of the following bits in the RX FIFO Configuration/Control Registers:

- CFDRFCCn.RFIE
- CFDRFCCn.RFFIE

In addition, the FIFO interrupt must be enabled by setting any one of the following bits in the Common FIFO Configuration/Control Registers:

- CFDCFCCn.CFRXIE
- CFDCFCCn.CFTXIE
- CFDCFCCEn.CFFIE
- CFDCFCCEn.CFOFRXIE
- CFDCFCCEn.CFOFTXIE

After configuration is complete, each FIFO can be enabled by setting the CFDRFCCn.RFE and CFDCFCCn.CFE bits in the RX FIFO Configuration/Control Registers and the Common FIFO Configuration/Control Registers to allow transmission and reception of messages.

When CFDCFCCEn.CFBME = 1, it becomes FIFO buffering mode, send data is stored in the Common FIFO, and transmission is stopped. Transmission starts when CFDCFCCEn.CFBME = 0.

Do not write 1 from 0 for this bit when the CFDCFCCn.CFE bit is 1.

30.7 Interrupt and DMA

30.7.1 Interrupts

The CAN-FD module generates several interrupts. The interrupt output, which is connected to the Interrupt Controller Unit (ICU), can be controlled by the corresponding interrupt enable bit.

The status flag is set independent from this enable bit.

The channel transmission interrupt has an additional Status Flag register. These status bits are only set when the corresponding interrupt enables are set.

The Status Flag register supports the identification of the interrupt source for the channel transmission, as this interrupt is driven by several trigger sources.

The interrupts in the CAN-FD module can be classified into 2 groups, global interrupts and channel interrupts:

- Global interrupts

The CAN-FD module can generate 2 global interrupts:

1. One Global interrupt for successful reception into the 8 RX FIFO buffers
2. One Global error interrupt

- Channel interrupts:

Each channel of the CAN-FD module can generate 3 channel interrupts:

1. Channel transmission
   Transmission completed from channel
   Transmission abort from channel
   Transmission from TX Queue for a channel
   Channel THL Interrupt
   Successful transmission from a Common FIFO in TX or GW mode for a channel
2. Channel error interrupt
3. Successful reception in a Common FIFO in RX or GW mode for a channel or successful routing in a TXQ

The interrupts are cleared when the corresponding flag bits are cleared or interrupt enable bits are cleared.

Table 30.35 provides an overview of interrupt.

Table 30.35 Overview of Interrupt Source

Parameter	Interrupt	Interrupt Source	Interrupt Clearing
Global Interrupts	Successful reception into at least one RX FIFO	Interrupt flag of corresponding RX FIFO for which interrupt is enabled	Clear the interrupt flag of corresponding RX FIFO buffer for which interrupt is enabled
	FIFO full into at least one RX FIFO	FIFO Full Interrupt flag of corresponding RX FIFO for which interrupt is enabled	Clear the FIFO Full Interrupt flags of corresponding RX FIFO buffer for which interrupt is enabled
	Global error	Any of the following: • DLC Error flag • Message Lost Status bit • Message Overwrite Status bit • TXQ Message Lost Status bit • TXQ Message Overwrite Status bit • TX History Entry Lost Status bit • CAN-FD Message Payload overflow flag	Clear all of: • DLC Error flag • Message Lost flags in all of the FIFO Status Registers • Message Overwrite flags in all of the Common FIFO Status Registers • Message Lost flags in all of the TXQ Status Registers • Message Overwrite Flags in all of the TXQ Status Registers • TX History List Entry Lost flag • CAN-FD Message Payload Overflow flag
Channel Transmission Interrupts	Channel n successful transmission	Any channel related TX MB Successful flag when interrupt is enabled Separate interrupts are provided for common FIFO buffers and TX Queue*1	Clear all channel related TX MB Result Status bits for which the interrupt is enabled
	Channel n abort	Any channel related TX MB Abort flag when interrupt is enabled Separate interrupts are provided for common FIFO buffers and TX Queue*1	Clear all channel related TX MB Result Status bits for which the interrupt is enabled globally
	Channel n transmission from TX Queue	Related channel TX Queue Interrupt flag	Clear related channel TX Queue Interrupt flag
	Channel n THL Interrupt	Channel n THL Interrupt Status flag	Clear the relevant THL Interrupt Status flag
	Channel n COM FIFO TX Interrupt	Interrupt flag for Common FIFOs in TX or GW mode belonging to the related channel	Clear the interrupt flags of Common FIFOs in TX or GW mode belonging to the related channel
	Channel n COM FIFO One Frame TX Interrupt	One Frame Transmission Interrupt flag for Common FIFOs belonging to the related channel	Clear the One Frame Transmission Interrupt flags of Common FIFOs belonging to the related channel
	Channel n TXQ One Frame TX Interrupt	One Frame Transmission Interrupt flag for TXQs belonging to the related channel	Clear the One Frame Transmission Interrupt flags of TXQs belonging to the related channel
Channel COM RX FIFO Interrupt	Channel n COM FIFO RX Interrupt	Interrupt flag for Common FIFOs in RX or GW mode belonging to the related channel	Clear the interrupt flags of Common FIFOs in RX or GW mode belonging to the related channel
	Channel n COM FIFO One Frame RX Interrupt	One Frame Reception Interrupt flag for Common FIFOs belonging to the related channel	Clear the One Frame Reception Interrupt flags of Common FIFOs belonging to the related channel
	Channel n COM FIFO Full Interrupt	FIFO Full Interrupt flag for Common FIFOs in RX or GW mode belonging to the related channel	Clear the FIFO Full Interrupt flags of Common FIFOs in RX or GW mode belonging to the related channel
	Channel n TXQ One Frame Routing Interrupt	One Frame Routing Interrupt flag for TXQs in GW mode belonging to the related channel	Clear the One Frame Routing Interrupt flags of TXQs in GW mode belonging to the related channel
	Channel n TXQ Full Interrupt	TXQ Full Interrupt flag for TXQs in GW mode belonging to the related channel	Clear the FIFO Full Interrupt flags of TXQs in GW mode belonging to the related channel
Channel Error Interrupt	Channel n Error	Any channel related error flag in the Channel Error Flag Register for which interrupt is enabled in the Channel Error Interrupt Enable Register	Clear all channel related error flags in the Channel Error Flag Register for which interrupt is enabled in the Channel Error Interrupt Enable Register

Note 1. These interrupts are only set for TX Message Buffers that do not belong to an enabled TX Queue and are not pointing to a common FIFO.

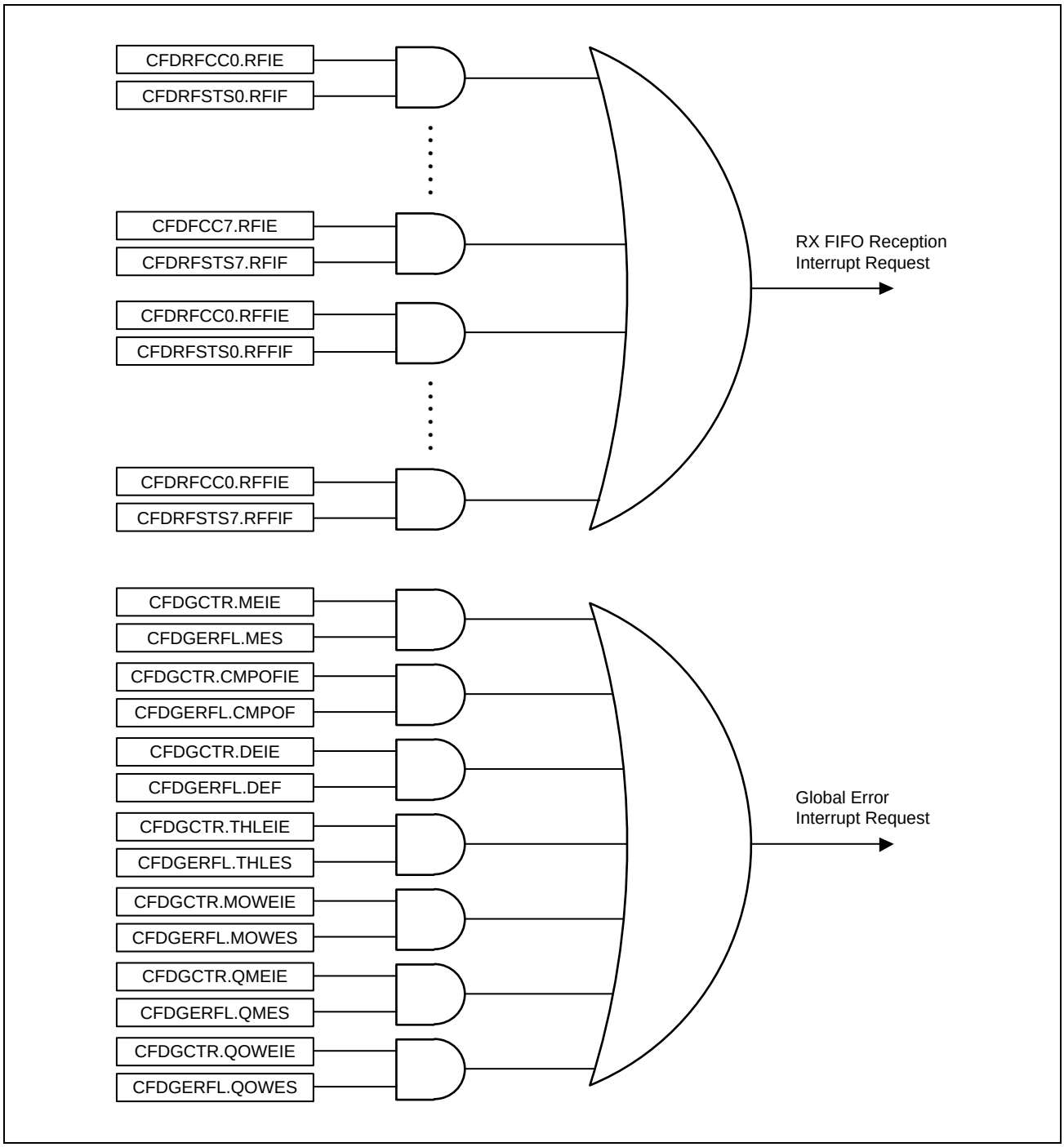


Figure 30.30 Block Diagram of Global Interrupt

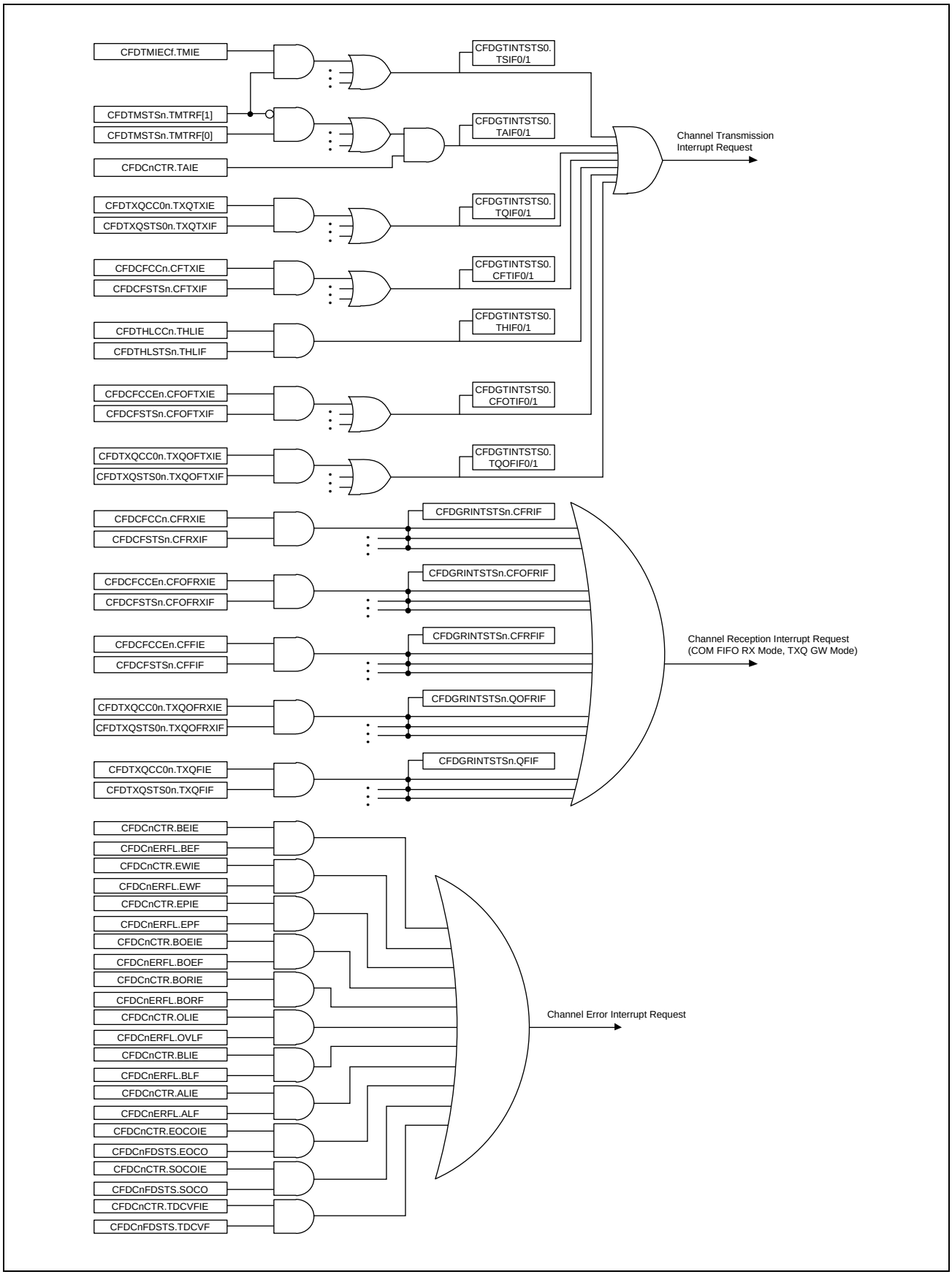


Figure 30.31 Block Diagram of Channel Interrupt

30.7.2 DMA Transfer

The CAN-FD module has some message buffer which can be associated with a DMA channel:

- Reception DMA
  - 8 RX FIFO Message Buffers
  - 2 Common FIFO Message Buffers
- Transmission DMA
  - 16 TXQ Message Buffers (TXQ0, TXQ3)
  - 2 Common FIFO Message Buffer

Figure 30.32 shows the potential DMA channels.

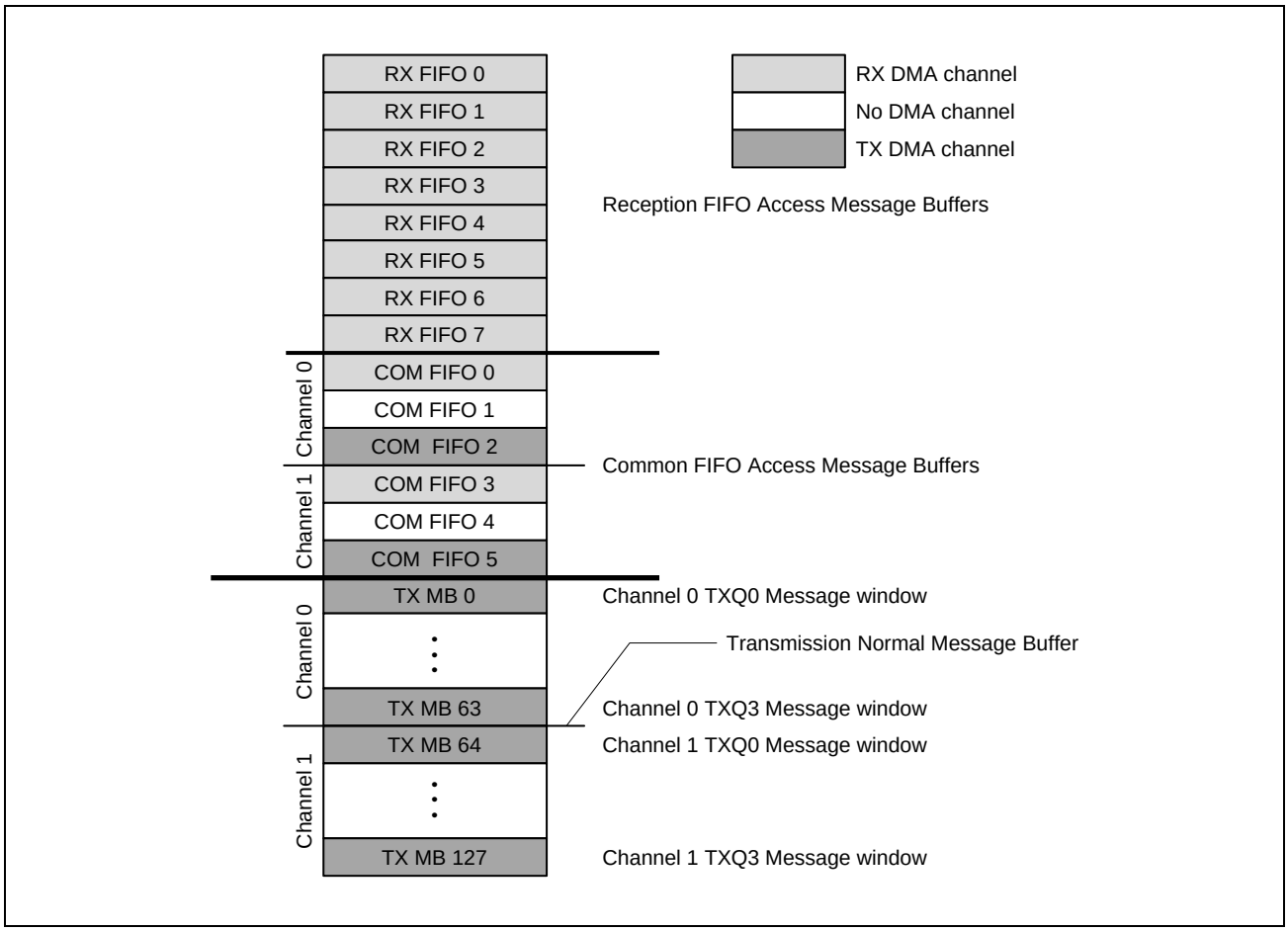


Figure 30.32 Message Buffer Connectable to DMA Channel

A DMA channel transfer request is generated for each FIFO entry to the DMAC when the related CFDCDTCT.RFDMAE or CFDCDTCT.CFDMAE is set to 1 and the belonging FIFO is not empty.

Reception FIFO interrupt should be disabled for this particular FIFO (CFDRFCCn.RFIE or CFDCFCCn.CFRXIE).

Use the regular start address for the DMA access window address and add H’8000 to the regular start address for the debugger access window. See **Table 30.36**.

Table 30.36 DMA Channel Access Window Address

b = Message Buffer Component Index	MBCP	Register	p	Start Address n = 0, 1
0 to number_of_RFMBCPs - 1	RFMBCPb	RFID	x	H'6000 + b × H'0080
		RFPTR	x	H'6004 + b × H'0080
		RFFDSTS	x	H'6008 + b × H'0080
		RFDFp	0 to 15	H'600C + p × H'0004 + b × H'0080
0 to number_of_CFMBCPs_per_channel - 1	CFMBCPb	CFID	x	H'6400 + b × H'0080 + n × H'0180
		CFPTR	x	H'6404 + b × H'0080 + n × H'0180
		CFFDCSTS	x	H'6408 + b × H'0080 + n × H'0180
		CFDFp	0 to 15	H'640C + p × H'0004 + b × H'0080 + n × H'0180

DMA FIFO pointer decrement is done automatically by reading the last configured data payload byte (CFDRFCCn.RFPLS or CFDCFCCn.CFPLS).

Note: The DMA must read the exact length of the configured data payload size (CFDRFCCn.RFPLS or CFDCFCCn.CFPLS).

The software debugger must access outside of the regular SFR address range from H'E000 to H'EFFF.

Do not write to the FIFO and TXQ control registers when DMA is enabled.

The DMA enable of the particular DMA FIFO (CFDCDTCT.RFDMAE or CFDCDTCT.CFDMAE) can be set at any time, the procedure in this section is a configuration flow for an initial setup.

When CFDCDTTCT.TQ0DMAE or CFDCDTTCT.TQ3DMAE or CFDCDTTCT.CFDMAE is set, the messages of the corresponding TXQ or Common FIFO can be handled by DMA controller.

Use the following procedure when the TXQ or the Common FIFO can be handled by DMA controller.

1. CPU checks the TXQ or the Common FIFO is not full.
2. When transmit data can be used, CPU sets this data to Common FIFO or TXQ.
   When using Common FIFO, transmit data is write in CFDCFIDn, CFDCFPTRn, CFDCFFDCSTSn, and CFDTMBCPb[i] registers.
   When using TXQ, transmit data is write in CFDTMIDn, CFDTMPTRn, CFDTMFDCTRn, and CFDTMDFp_n registers.
3. For Common FIFO, the common FIFO pointer is incremented automatically when DMA controller writes the last data payload byte configured by CFDCFCCn.CFPLS.
   For TXQ, if the data of 64 data payload is written, a TXQ pointer increases automatically. When payload data is less than 64 bytes, dummy data must be written in and 64 data payload size must be done.

Note: Only 32-bit write-access can be possible on the DMA message handling.

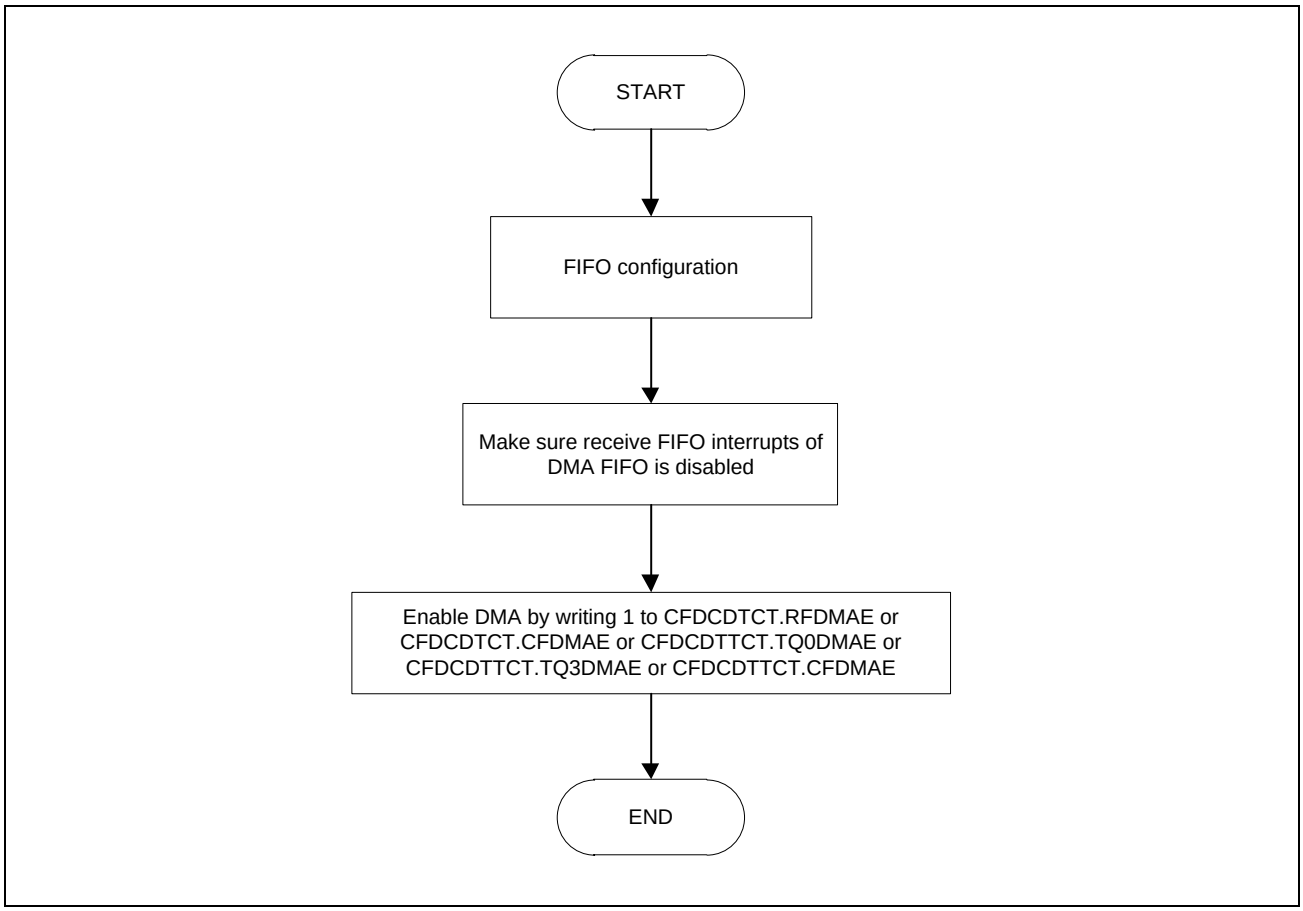


Figure 30.33 DMA Enable Flow

To disable a DMA transfer requested, disable the particular DMA enable bit (CFDCDTCT.RFDMAE or CFDCDTCT.CFDMAE). If the disable is made during an ongoing transfer, then this must be completed first before further action can be taken. The transfer status can be identified by the CFDCDTSTS.RFDMASTS or CFDCDTSTS.CFDMASTS. For reference, see the flow in **Figure 30.34**. When the DMA is disabled, then consider what to do with the remaining or new incoming messages to this particular FIFO reception.

When the FIFO is not disabled, reception to the FIFO continues.

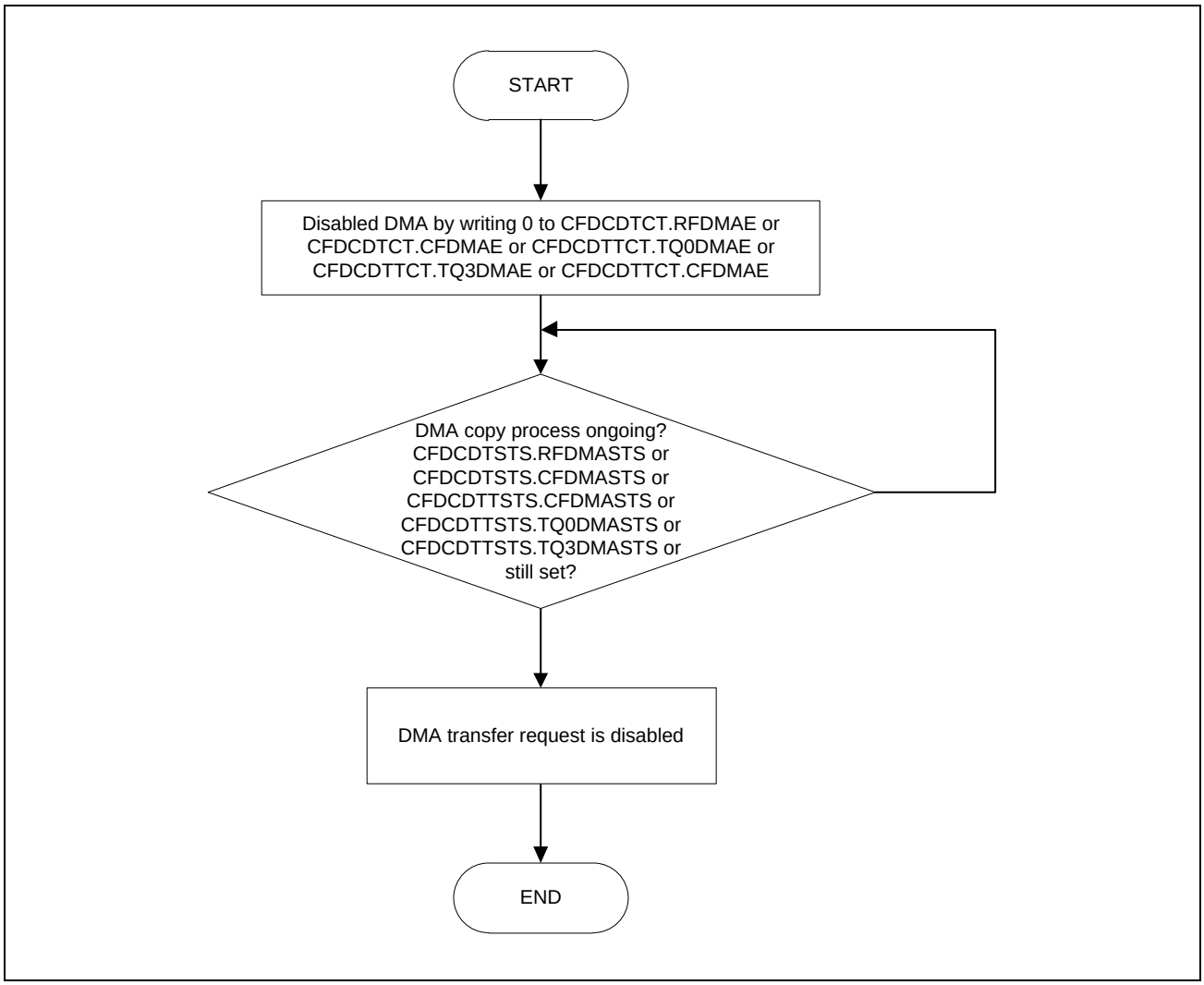


Figure 30.34 DMA Disable Flow

30.8 Reception and Transmission

30.8.1 Reception

FIFO buffers or Common FIFO buffers configured in RX mode or GW mode depending on the Acceptance Filter List entries are as follows:

- Up to 32 RX Message Buffers can be configured
- 8 RX FIFO buffers available
- Up to 6 Common FIFO buffers can be configured in RX mode or GW mode
- Up to 4 TX Queues can be configured in GW mode

30.8.1.1 Message Storage in RX Message Buffers

When a message is successfully received and stored in a RX Message Buffer, the corresponding New Data flag is set in the RX Message Buffer New Data Register.

The CAN message can be read from the corresponding RX Message Buffer.

If a new message is stored into a RX Message Buffer before the previous message in this message buffer can be read, then the original message is overwritten. There is no mechanism for preventing a new message from overwriting the current message in the RX Message Buffer. If such a loss of messages is not acceptable, then RX FIFO should be used to store related messages.

Note: Interrupts are not provided for the RX Message Buffers in the CAN-FD module and so the RX Message Buffer New Data Register should be accessed periodically to check if a new message has been stored in the RX Message Buffers.

Note: Unused data bytes are filled with H'00 depending on the DLC value.

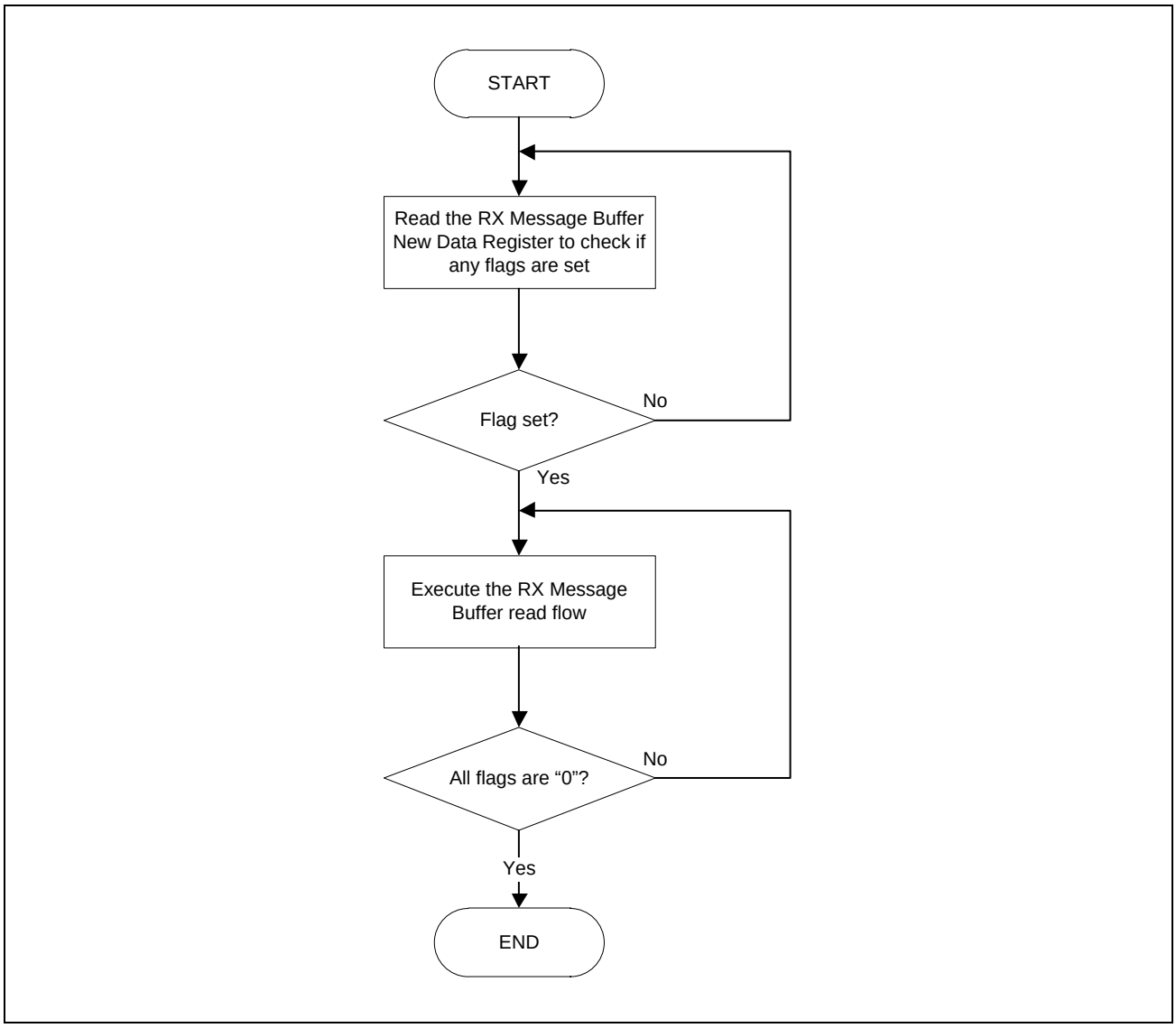


Figure 30.35 RX Message Buffer Message Access Flow

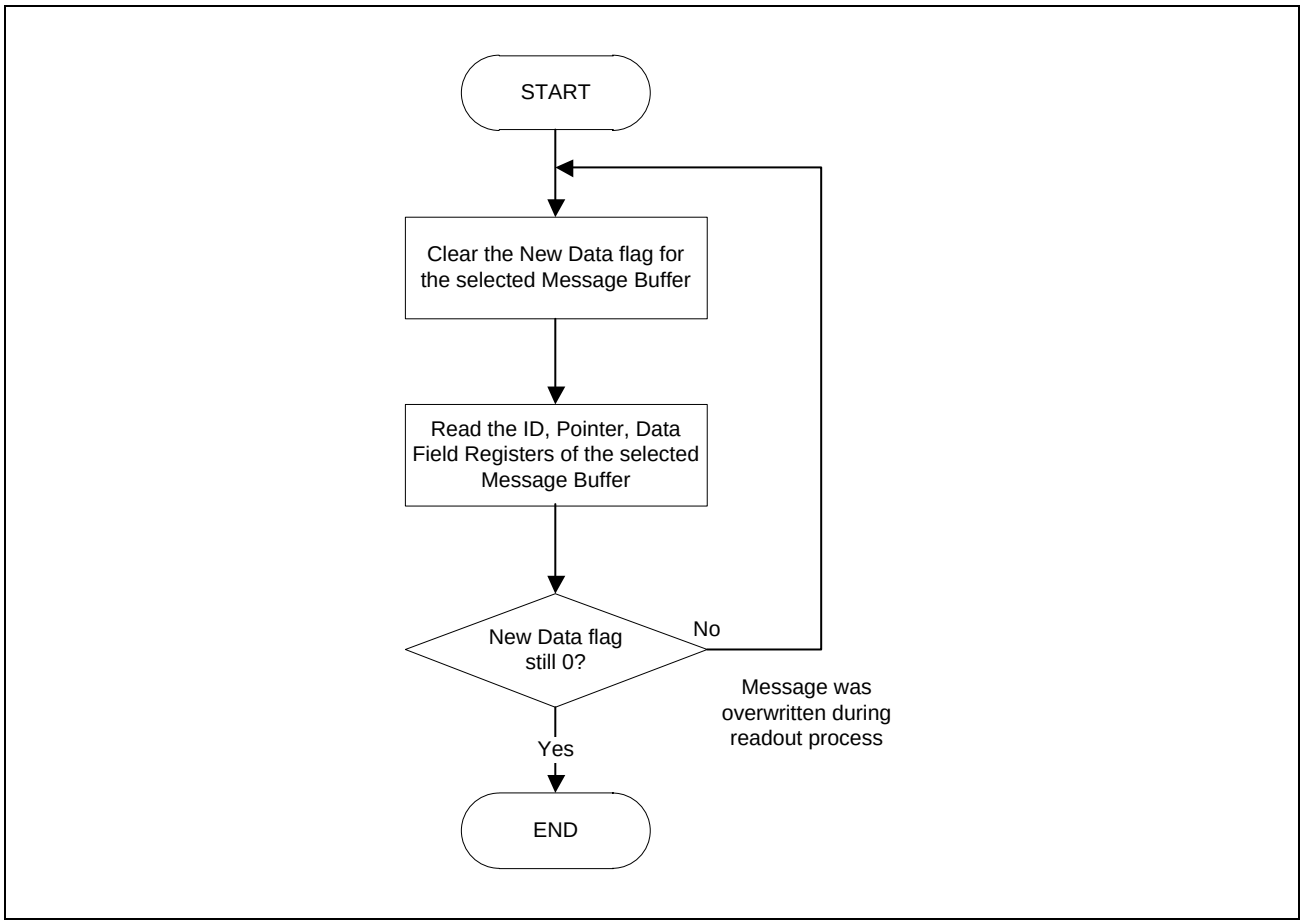


Figure 30.36 RX Message Buffer Read Flow

30.8.1.2 Message Storage in FIFO Buffers

The AFL entries for routing the received messages to RX FIFO buffers or Common FIFO buffers configured in RX or GW mode should be configured based on system requirements.

The CFDGAFLP1n.GAFLFDP[31:0] field in the matching AFL entry selects the FIFO buffers to which the related reception message is stored.

When the received message is stored in one or more RX FIFO buffers or Common FIFO buffers configured in RX mode or GW mode, the message counter value is incremented in the corresponding RX FIFO Status Registers or Common FIFO Status Registers.

Depending on the configuration of the FIFO buffers, an interrupt may also be generated.

The message can be read from the corresponding FIFO Access registers.

Note: Because many messages can be stored in the FIFO buffers, reading more than one message may be required to read the latest message stored in a FIFO buffer.

If the message count value matches the FIFO depth, the FIFO Full flag is set.

When the value H'FF is written to the corresponding FIFO Pointer Control Register, the message count is decremented by 1.

Only write H'FF to the FIFO Pointer Control register after reading the complete message from the FIFO Access registers of the corresponding FIFO.

When all the messages stored in the FIFO are read, the FIFO Empty flag is set.

If a new message is stored into the FIFO when the FIFO message count matches the FIFO depth (FIFO Full condition), the FIFO Message Lost flag is set and the new message is lost (no overwrite of already stored messages takes place).

An appropriate value can be configured as warning level to generate an interrupt before the FIFO full condition occurs to avoid loss of a message due to overrun condition.

In GW mode, when a transmit/receive FIFO buffer is trying to receive a new message while the transmit/receive FIFO buffer is already full of data, the oldest data of the buffer is overwritten with the message received or the message is discarded. The behavior is determined by setting CFDCFCCEn.CFMOWM bit.

- When CFDCFCCEn.CFMOWM = 0:
  When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the received message is discarded and CFDCFSTSn.CFMLT bit is set to 1.
- When CFDCFCCEn.CFMOWM = 1:
  When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the oldest data in the buffer is overwritten with the received message.
  The read pointer of the transmit/receive FIFO buffer simultaneously moves to the next oldest message.
  The CFDCFSTSn.CFMOW bit is then set to 1, which notifies that the oldest message has been overwritten with the received message.
  In addition, if a CAN bus error or arbitration-lost for the transmitting message occurs in the transmit/receive FIFO buffer full, the transmitting message is lost and retransmission of the message is not performed. The read point moves to the next message automatically.

Do not write change for this bit when the CFDCFCCn.CFE bit is 1. Common FIFO can set interrupt when:

- CAN frame reception is completed
- FIFO is in full status in RX mode or GW mode

Note: The message lost can be set only in RX or GW mode by CAN, the flag is not set when the CPU is overloading the FIFO buffers.

Note: When CFDGAFLP0n.GAFLSRDi (i = 0 to 2) is set and CFDTXQCCin.TXQGWE (i = 0 to 2, n = 0, 1) is also set, a receiving frame is stored in the target TXQ as send data by routing.

The RX FIFO Buffers and the Common FIFO Buffers configured in RX or GW mode can be disabled at any time by clearing the CFDRFCCn.RFE or CFDCFCCn.CFE bit in the RX FIFO Configuration/Control Registers and the Common FIFO Configuration/Control Registers.

When the CFDRFCCn.RFE or CFDCFCCn.CFE bit is cleared, the message read and write pointers of the FIFO are cleared and are no longer active. Therefore, all messages in the FIFO Buffers are lost and no further messages can be stored into the FIFO.

When the RX FIFO Buffers or Common FIFO Buffers configured in RX mode are assigned as DMA channel, software should not access the FIFO Access Register of this FIFO buffer or write H’FF to the FIFO Pointer Control Register (CFDCFPCTRn.CFPC or CFDRFPCTRn.RFPC), because this could lead to unintended FIFO message decrement. The DMA channel controls the FIFO decrement automatically.

Note: If the interrupt flag is set for a FIFO Buffer and then the FIFO is disabled, the interrupt flag is not automatically cleared. The interrupt flag should be cleared before disabling the FIFO.

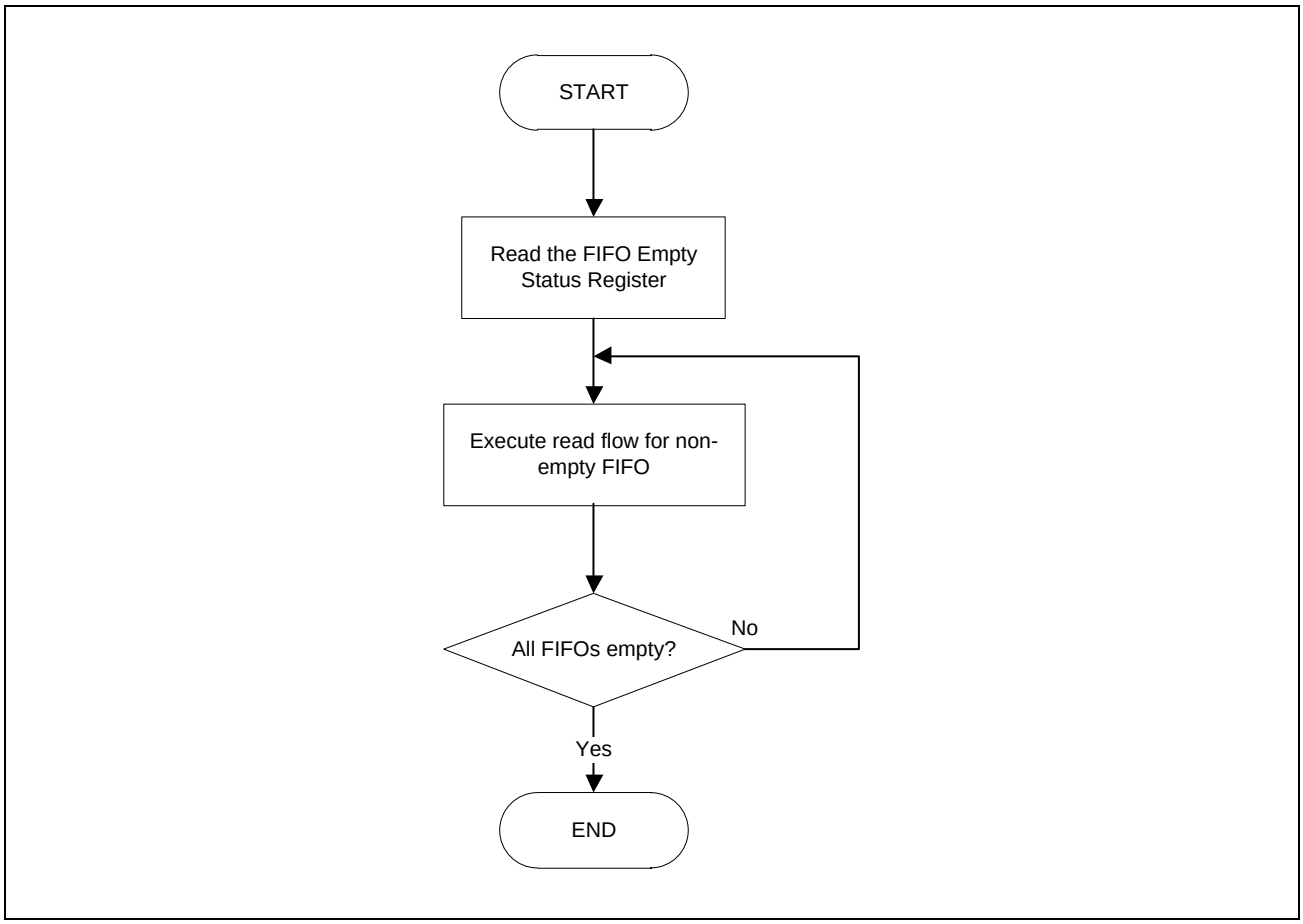


Figure 30.37 FIFO Buffer Message Access Flow (Example for Polling Case)

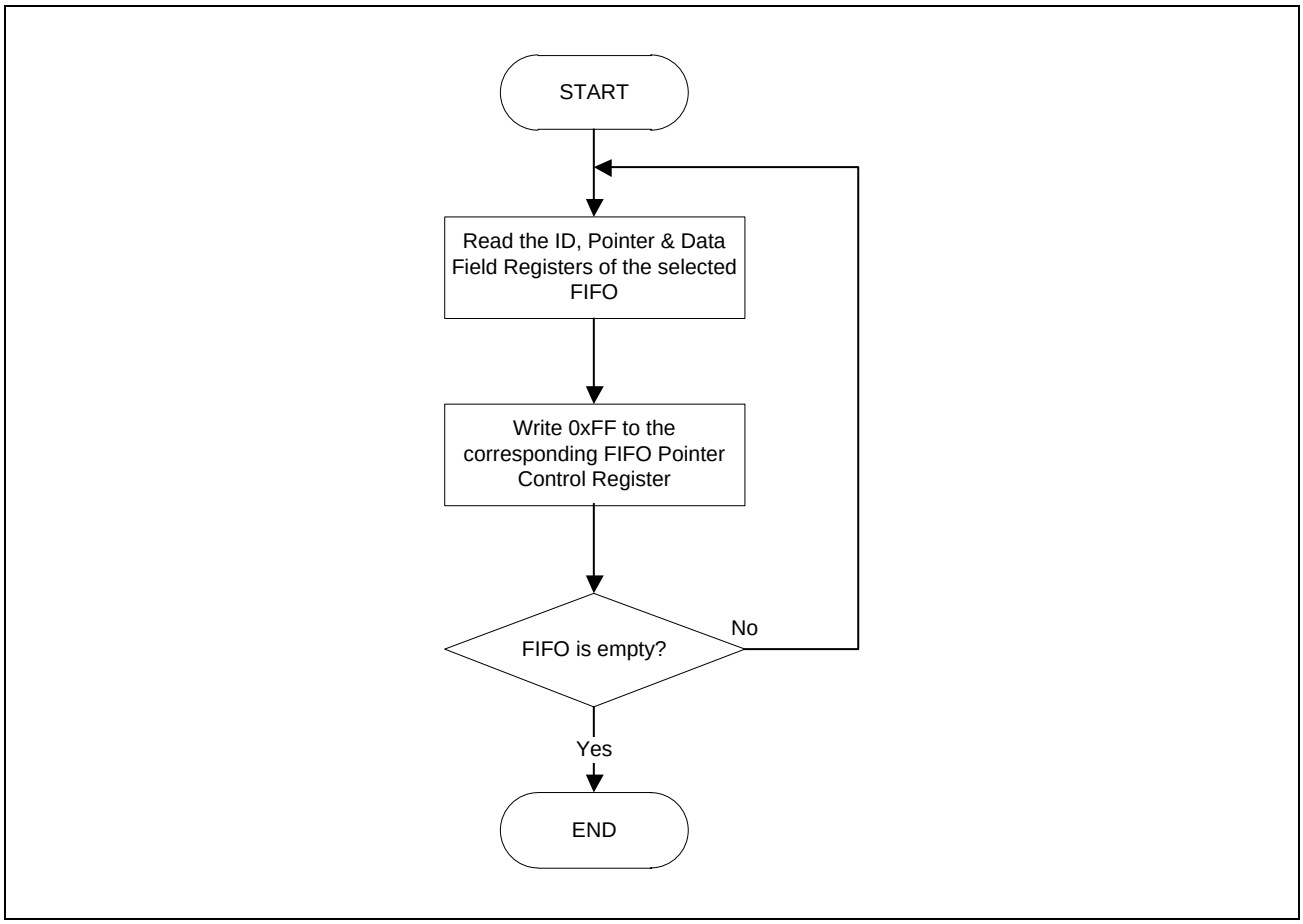


Figure 30.38 RX FIFO Buffer Read Flow (Example for Polling Case)

Note: When the next frame is received before clearing the completion interrupt flag of reception, the completion interrupt of reception is not set again.

Even if it clears an interruption flag after the completion processing of reception, the already received interrupt flag is not set.

It is necessary to perform the completion processing of reception even before the next completion of frame reception, and to clear an interruption flag.

When processing does not meet the deadline, after checking that receiving data is empty, interrupt flag is cleared and it checks that receiving data is empty again.

30.8.1.3 Timestamp

The Timestamp counter is a free-running counter that can be used to check reception time of an incoming message or transmission time of successful transmitted messages. The Timestamp counter value is captured based on the CFDGFDCFG.TSCCFG[1:0] configuration (at the sample point of start of frame, point in time when the frame is valid, or for CAN-FD frames also at the sample point of the RES bit). For reception, it is stored together with the message ID and Data into the target RX Message Buffer or RX/GW FIFO.

For transmit message, the Timestamp counter value is stored as part of the TX History List entry.

The counter can be clocked with the peripheral clock or with the CAN channel bit timing clock. The counter source clock can be configured with the CFDGCFG.TSSS bit of the Global Configuration Register. If it is 0, the peripheral clock is used. If it is 1, the selected CAN channel bit time clock is used.

The channel selection is done with the CFDGCFG.TSBTCS bit of the Global Configuration Register.

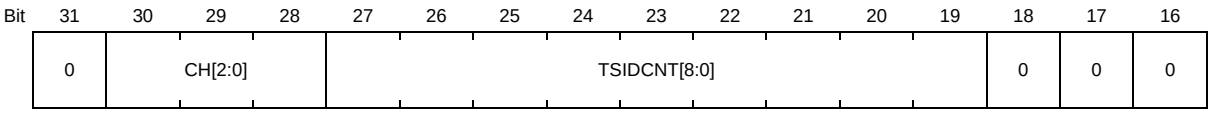
Care must be taken when using selected CAN channel bit time clock as the clock source. When entering Channel Halt mode or Channel Reset mode for this channel, the Timestamp counter is stopped. Therefore, for other CAN channels, the Timestamp counter value is also not updated.

If peripheral clock is selected as the Timestamp counter clock source, Channel modes do not influence the Timestamp counter function.

The source clock for the Timestamp counter can be divided by a factor defined by the CFDGCFG.TSP bits (Timestamp Prescaler) in the Global Configuration Register.

The Timestamp counter can be reset to H’0000 with the CFDGCTR.TSRST bit (Timestamp Reset).

The Timestamp counter filed format is described below.



30.8.2 Transmission

There are several possible transmission configurations for each channel:

- Normal transmission
- FIFO transmission
- Gateway transmission
- TX Queue transmission
  A fixed number of transmission message buffers (64 TX message buffers) are dedicated for each channel. These message buffers are only used for transmission and cannot be configured for reception.
  Additionally transmission from TX Queue and/or Common FIFO in TX or GW mode can be configured in the following way (see **Figure 30.39**):
  - TX Queue: Up to 32 transmission message buffers for one channel can be grouped to form a TX Queue with a common access window.
    Upper transmission message buffers are used to form the TXQ1 or TXQ3. Lower transmission message buffers are used to form the TXQ0 or TXQ2.
    Transmission control and status registers of these transmission message buffers should not be used. One channel has four TX Queues. Each TXQ has an access window:
    - TXQ0 is transmission Message Buffer 0 of each channel
    - TXQ1 is transmission Message Buffer 31 of each channel
    - TXQ2 is transmission Message Buffer 32 of each channel
    - TXQ3 is transmission Message Buffer 63 of each channel

    When using TXQ1 and TXQ0 simultaneously, the sum of the depths of TXQ1 and TXQ0 should not exceed 32.
    When using TXQ3 and TXQ2 simultaneously, the sum of the depths of TXQ3 and TXQ2 should not exceed 32.
- Common FIFO (TX/GW mode): Each Common FIFO in TX or GW mode is linked to a dedicated channel.
  Each channel has a fixed number of three Common FIFOs assigned to it. Within the channel, a Common FIFO configured in TX or GW mode, can be freely linked (assigned) between 32 and 39 transmission message buffers (only one FIFO to one transmission message buffer).
  The Common FIFO buffer then replaces the transmission message buffer linked to it.
  Transmission control and status registers of these transmission message buffers should not be used.

See **Figure 30.28** for information on Common FIFO buffer assignment to related channels.

Note: Common FIFO buffers should not be linked to TX message buffers that are already part of a TX Queue.

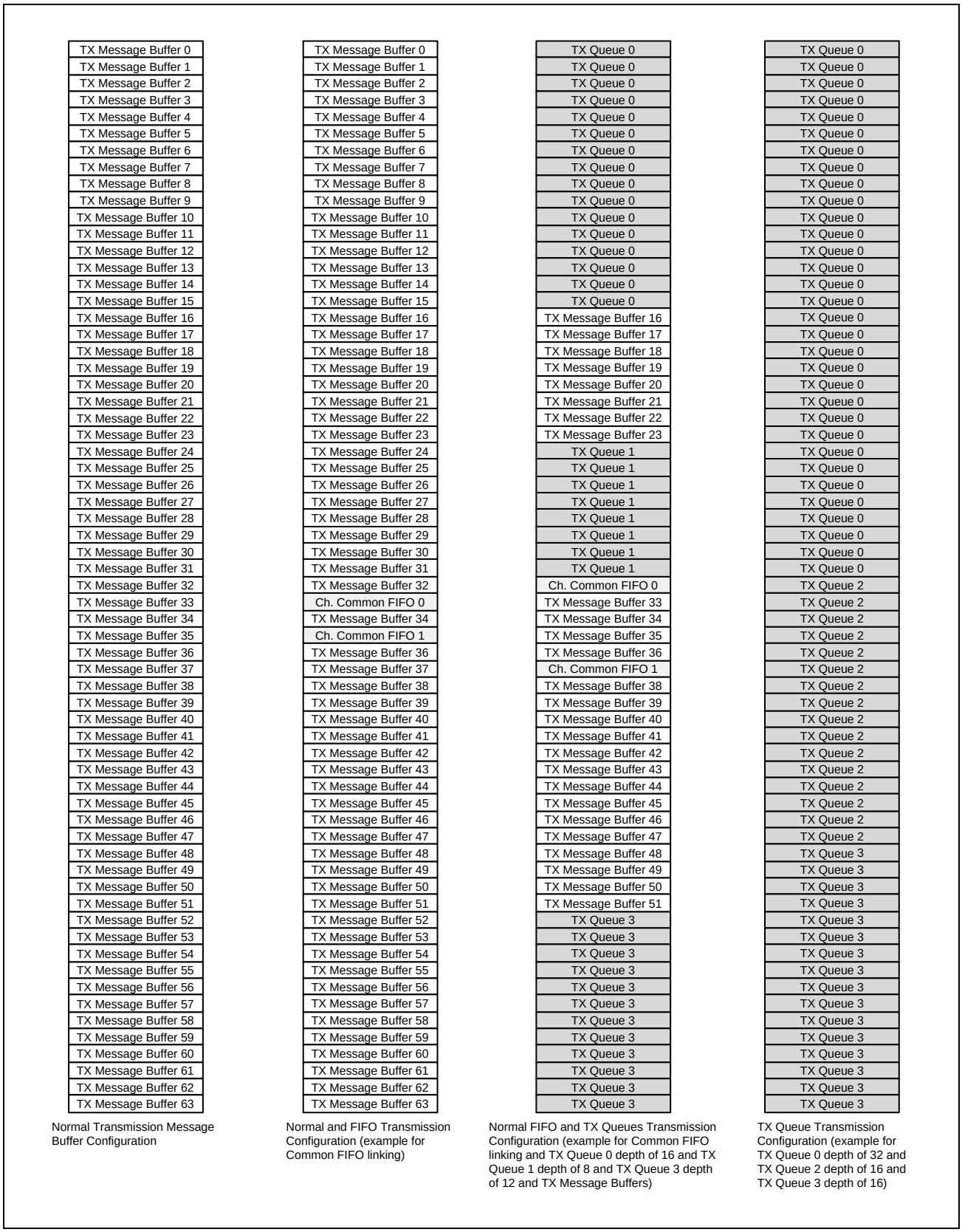


Figure 30.39 Channel Transmission Message Buffer Configuration

30.8.2.1 Transmission Priority

If two or more transmission message buffers of a channel are configured for transmission, the transmission priority in the CAN-FD module can be selected from the following two modes:

- CAN ID priority
- Message buffer number priority

The transmission priority mode is common for all message buffers and all CAN channels. It can be configured with the CFDGCFG.TPRI bit in the Global Configuration Register.

For message buffer number priority transmission, the smallest message buffer number with transmission request has the highest priority for transmission. This also includes the TX message buffers linked to the Common FIFO buffers configured in TX mode or GW mode.

However, message buffer number priority should not be used if TX Queue is enabled.

For CAN ID priority transmission, ID priority complies with the CAN bus arbitration rule (as specified in ISO 11898-1 specification). All TX message buffers can enter the ID priority comparison for message buffers configured for transmission. This also includes the TX message buffers linked to the Common FIFO buffers configured in TX mode or GW mode and includes the TX Queue message buffers.

If the ID of two or more message buffers is the same, the smaller message buffer number has higher priority for transmission.

Note: For Common FIFO buffers configured in TX mode or GW mode, only the message currently being pointed to by the FIFO read pointer can be included in the transmission arbitration.
If the message is being transmitted from the FIFO, the next pending message within the same FIFO will be considered in the transmission arbitration.
In contrast to this, all transmission message buffers of a TX Queue participate in internal transmission arbitration.

Figure 30.40 shows the transmission configuration flow.

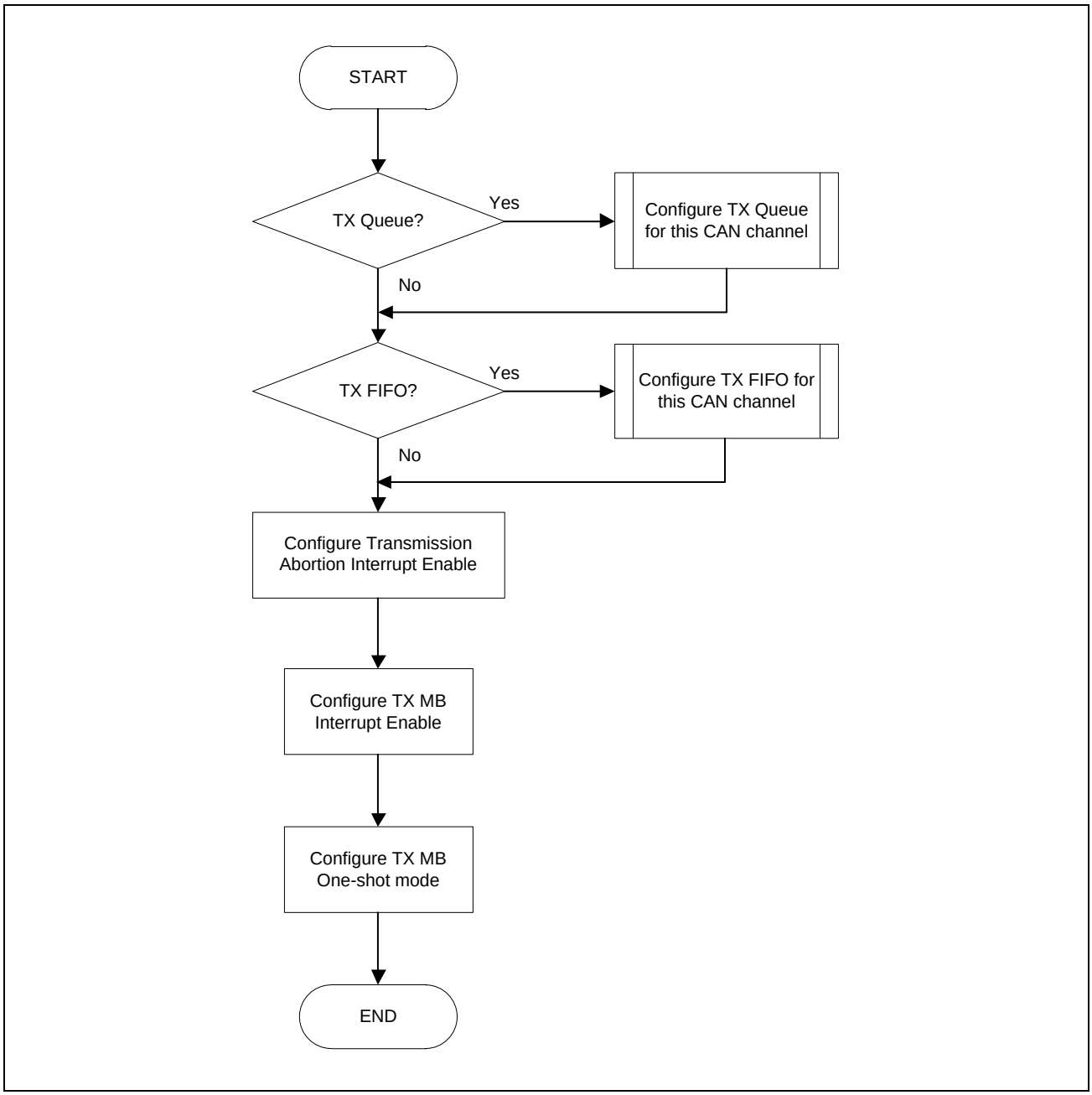


Figure 30.40 Transmission Configuration Flow

30.8.2.2 Normal Transmission

Each transmission message buffer has two modes of message transmission:

- Regular transmission mode
  If the message buffer is placed in regular transmission mode, the data frame or remote frame set in that message buffer can be transmitted.
  Completion of regular transmission can be checked through the related TX Message Buffer Transmission Result flag bits (CFDTMSTSn.TMTRF[1:0]) in the TX Message Buffer Status Registers. These bits are set to 10b or 11b when the regular transmission is successful.
  When arbitration is lost or an error occurs, message transmission will be attempted further if no transmission abort request is set for this transmission message buffer.
  New internal transmission arbitration for this channel is performed for all message buffers with transmission request.
- One-shot transmission mode
  When the CFDTMCn.TMOM bit of the TX Message Buffer Control Registers is set for a transmission message buffer, then the message buffer is placed in one-shot transmission mode and attempts to transmit a message only once.
  Completion of one-shot transmission can be checked through the related TX Message Buffer Transmission Result flag bits (CFDTMSTSn.TMTRF[1:0]) in the TX Message Buffer Status Registers. The CFDTMSTSn.TMTRF[1:0] bits are set to 10b or 11b when the one-shot transmission is successful.
  The CFDTMSTSn.TMTRF[1:0] bits are set to 01b when arbitration is lost or an error occurs during the transmission of the related message buffer.
  Additional message transmission will not be attempted in this case.

The regular transmission request procedure after a configuration is shown in **Figure 30.41**.

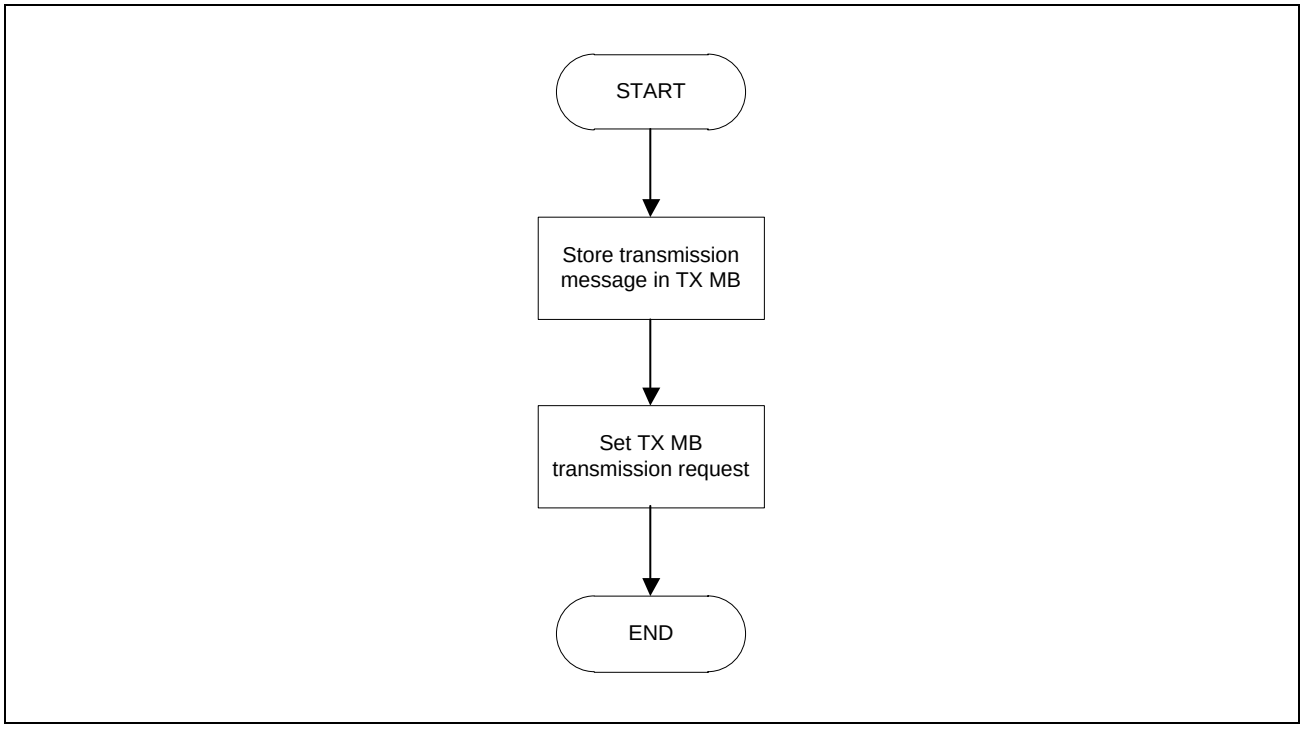


Figure 30.41 Transmission Request Procedure Using Normal TX Message Buffer Mode

(1) TX Message Buffer Control Register Setting

Table 30.37 shows configuration of the normal CAN transmission mode.

Table 30.37 Configuration of CAN Transmission Mode

Transmission Request CFDTMCn.TMTR	Transmission Abortion Request CFDTMCn.TMTAR	One Shot Enable CFDTMCn.TMOM	Communication Activity
0	0	0	Message buffer is disabled
0	0	1	Message buffer is disabled
1	0	0	Configured as a transmission message buffer for a data frame or a remote frame
1	0	1	Configured as a one shot transmission message buffer for a data frame or a remote frame
1	1	0	Transmission abortion is requested
1	1	1	One shot transmission abortion is requested

The configuration bits can be configured in the TX Message Buffer Control Registers.

Figure 30.42 shows timings for successful transmission for two message buffers of one channel.

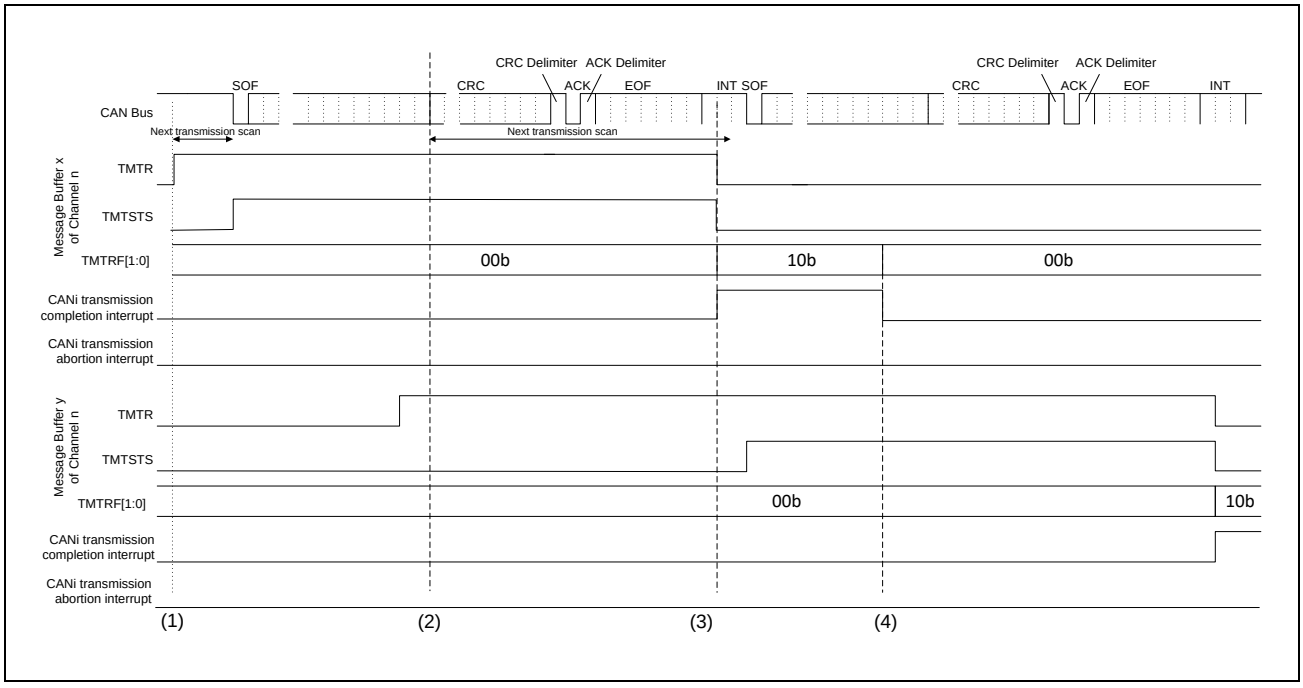


Figure 30.42 Timing of Request and Flag Bits for Successful Transmission

1. If the CFDTMCn.TMTR bit in the TX Message Buffer Control Registers is set in the bus idle state, message buffer scanning procedure starts to determine the highest priority message buffer for transmission.
   When the transmission message buffer is determined, the CFDTMSTSn.TMTSTS bit in the related TX Message Buffer Status Registers is set (Transmitting/Transmitter), and CAN channel starts the transmission*[1].
2. On the first bit of CRC, the transmission scanning procedure starts for the next possible transmission when pending transmission requests exist.

The scan time can be delayed due to other transmission scan on other channels, but it will be finished before Intermission 3 to be able to continue transmission without any gaps.

3. If the message is successfully transmitted, the CFDTMSTSn.TMTRF[1:0] bits in the corresponding TX Message Buffer Status Registers are set to 10b and CFDTMSTSn.TMTSTS and the CFDTMCn.TMTR bits are cleared. When the TMIE bits in the TX Message Buffer Interrupt Enable Configuration Registers is set (interrupt enabled), the CAN successful transmission interrupt request is generated.
   To clear the related interrupt line the CFDTMSTSn.TMTRF flag bits must be cleared.
4. Before starting the next transmission, clear the CFDTMSTSn.TMTRF[1:0] bits. Load the next message in the transmission message buffer and set the CFDTMCn.TMTR bit again. CFDTMCn.TMTR bit cannot be set again before CFDTMSTSn.TMTRF[1:0] bits are cleared.

Note 1. If arbitration is lost after the CAN channel starts the transmission, the CFDTMSTSn.TMTSTS bit is cleared.

The transmission scanning procedure is performed again to search for the highest priority transmission message buffer from the beginning of the first CRC bit.

If an error occurs either during the transmission or following the loss of arbitration, then during Error Frame, the transmission scanning procedure is performed again to search for the highest priority transmission message buffer.

Note: The setting point of CFDTMSTSn.TMTSTS is not always fixed at the start of the SOF. It may be delayed up to the start of the standard ID due to the synchronization logic implemented for the PLL bypass.

Figure 30.43 shows timings for transmission abort for two message buffers of one channel.

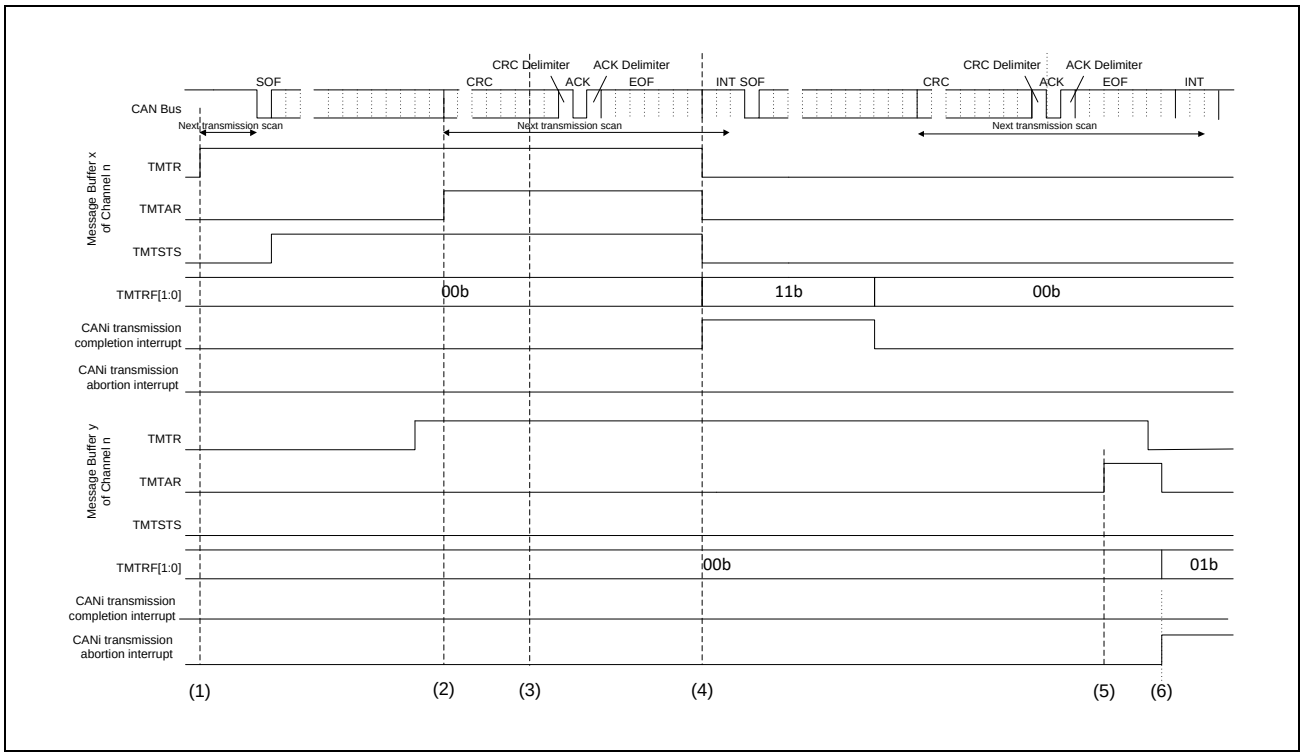


Figure 30.43 Timing of Request and Flag Bits for Transmission Abort

1. If the CFDTMCn.TMTR bit in the TX Message Buffer Control Registers is set in the bus idle state, message buffer scanning procedure starts to determine the highest priority message buffer for transmission.
   When the transmission message buffer is determined, the CFDTMSTSn.TMTSTS bit in the TX Message Buffer Status Registers is set (Transmitting/Transmitter), and CAN channel starts the transmission*[1].
2. If the CFDTMCn.TMTAR bit is set when the related message buffer is already selected for transmission or currently transmitting, the message is not aborted, if no error occurs or arbitration is lost.
3. On the first CRC bit, the transmission scanning procedure starts for the next transmission. In this example timing diagram, message buffer y is not selected as the next transmission message buffer.
   The scan time can be delayed due to other transmission scan on other channels, but it will be finished before Intermission 3 to be able to continue transmission without any gaps.
4. If the message is successfully transmitted, the CFDTMSTSn.TMTRF[1:0] bits in the corresponding TX Message Buffer Status Registers are set to 11b and CFDTMSTSn.TMTSTS and the CFDTMCn.TMTR bits are cleared.
   When the TMIE bits in the TX Message Buffer Interrupt Enable Configuration Registers is set (interrupt enabled), the CAN successful transmission interrupt request is generated.
   To clear the related interrupt line, clear the CFDTMSTSn.TMTRF[1:0] bits.
5. Another CAN node is transmitting on the CAN bus (CFDTMSTSn.TMTSTS not set). If the CFDTMCn.TMTAR bit is set when the related channel is under transmission scan, the transmission request cannot be cleared.
6. After internal processing time, the transmission is aborted and the CFDTMSTSn.TMTRF[1:0] bits are set to 01b.
   If the message buffer is not transmitting or selected as the next transmission message buffer or under transmit scan, then the abort is immediately accepted and the corresponding CFDTMSTSn.TMTRF[1:0] bits in the TX Message Buffer Status Registers are set to 01b.
   In addition, CFDTMCn.TMTR, and CFDTMCn.TMTAR bits are cleared automatically.
   When the transmission abort interrupt enable TAIE bit of the related Channel Control Register is set, an interrupt is generated for successful transmission abort.
   To clear the related interrupt line, clear the CFDTMSTSn.TMTRF[1:0] bits.

Note 1. If arbitration is lost after the CAN channel starts the transmission, the CFDTMSTSn.TMTSTS is cleared. The transmission scanning procedure is performed again to search for the highest priority transmission message buffer from the beginning of the first CRC bit. If an error occurs, either during the transmission, or following the loss of arbitration, then during Error Frame, the transmission scanning procedure is performed again to search for the highest priority transmission Message Buffer.

30.8.2.3 TX FIFO or GW FIFO Transmission

Three common FIFO buffers are assigned to each channel. The three FIFO buffers can be linked to any normal TX Message Buffer position for this channel by the CFDCFCCn.CFTML[4:0] bits in the Common FIFO Configuration/Control Register if configured in TX or GW mode.

When the transmission scan starts and the FIFO buffer corresponding to this TX Message Buffer is enabled, the relevant message in the FIFO buffer participates in the transmission scan.

Configuration of a TX Message Buffer linked to a FIFO buffer configured in TX or GW mode should not be done.

(1) TX FIFO Operation

CAN messages can be written into the TX FIFO by writing to the corresponding FIFO Access registers.

When the value H'FF is written into the corresponding FIFO Pointer Control Register, the message count of the related FIFO is incremented by 1.

Only write to the FIFO Pointer Control register after writing the complete message to the corresponding FIFO Access registers.

If the message count matches the FIFO Depth, the FIFO Full flag is set.

The oldest message in the TX FIFO is included in the scan for transmission by the corresponding CAN-FD module channel logic.

When a message is successfully transmitted from the TX FIFO, the message count value is decremented by 1. When all the messages from the FIFO are transmitted, the FIFO Empty flag is set.

The interrupt generation conditions for the TX FIFO buffers can be configured by configuring the CFDCFCCn.CFIM bit in the corresponding Common FIFO Configuration/Control Registers.

If CFDCFCCn.CFIM bit is 0, an interrupt is generated when the last message is successfully transmitted from the TX FIFO buffer.

If CFDCFCCn.CFIM bit is 1, an interrupt is generated for every successfully transmitted message from the TX FIFO buffer.

Common FIFO can set interrupt, when CAN frame transmitted is completed.

The Common FIFO buffers configured in TX mode can be disabled by clearing the CFDCFCCn.CFE bit in the Common FIFO Configuration/Control Registers. If this bit is cleared to 0, the FIFO Empty flag is set as follows:

- Immediately if the message from the TX FIFO is neither scheduled for the next transmission nor in transmission
- Following the transmission completion, the detection of an error on the CAN bus, loss of arbitration or transition to Channel or Global Halt mode if the transmission from the TX FIFO is already scheduled for transmission or already in transmission.

Note: The Common FIFO buffer is considered as disabled after clearing the CFDCFCCn.CFE bit only when the Empty flag is set for the corresponding Common FIFO Buffer.

Other possible messages pending from the TX FIFO are lost and their transmission must be requested again. Before CFDCFCCn.CFE is set again, ensure that CFDCFSTSn.CFEMP bit is set and that there is no pending abort from the TX FIFO.

When the CFDCFCCn.CFE bit is cleared, the message read and write pointers of the FIFO are cleared and are no longer active. Therefore, all messages in the FIFO buffers are lost and no further message can be stored into the FIFO.

The FIFO transmission request procedure after configuration is shown in **Figure 30.44**.

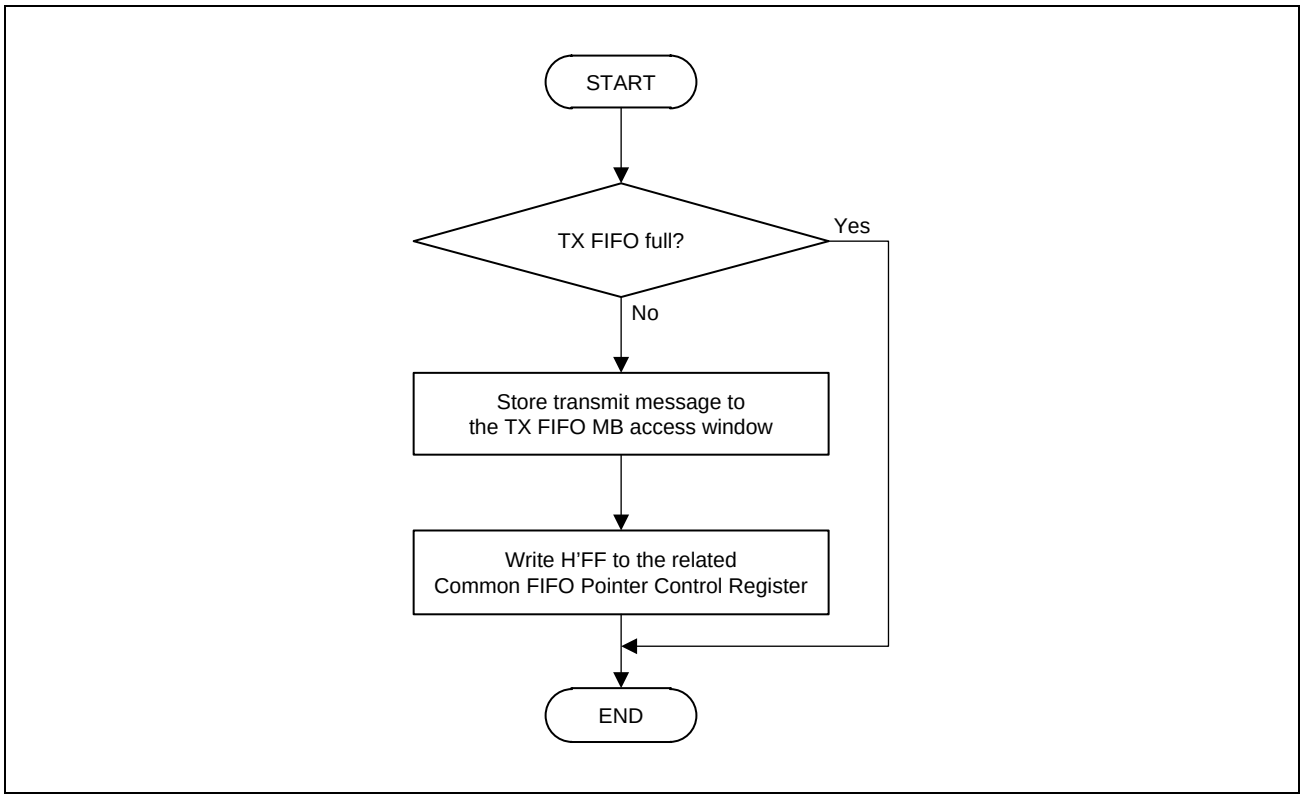


Figure 30.44 TX FIFO Transmission Request Procedure

(2) GW FIFO Operation

The AFL entries for routing the received messages to GW FIFO buffers should be configured based on the requirements of the system. The matching AFL entry selects the GW FIFO buffer for storage of a received message on any of the CAN channels.

When a message is successfully received and stored in a GW FIFO buffer, the FIFO message count in the corresponding FIFO Status Register is incremented by 1.

If the message count matches the FIFO depth, then the FIFO Full flag is set.

The oldest message in the GW FIFO is included in the scan for transmission by the corresponding CAN-FD module channel logic.

When a message is successfully transmitted from the GW FIFO, the message count value is decremented by 1. When all the messages from the GW FIFO are transmitted, the FIFO Empty flag is set.

If a new message is stored into the FIFO when the FIFO message count matches the FIFO depth (FIFO full condition), the FIFO Message Lost flag is set and the new message is lost (no overwrite of already stored messages takes place).

In GW mode, when a transmit/receive FIFO buffer is trying to receive a new message while the transmit/receive FIFO buffer is already full of data, the oldest data of the buffer is overwritten with the message received or the message is discarded. The behavior is determined by setting CFDCFCCEn.CFMOWM bit.

- When CFDCFCCEn.CFMOWM = 0:
  When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the received message is discarded and CFDCFSTSn.CFMLT bit is set to 1.
- When CFDCFCCEn.CFMOWM = 1:
  When writing of data is required due to reception of a new message while a transmit/receive. FIFO buffer is full of

data, the oldest data in the buffer is overwritten with the received message.
The read pointer of the transmit/receive FIFO buffer simultaneously moves to the next oldest message.
The CFDCFSTSn.CFMOW bit is then set to 1, which notifies that the oldest message is overwritten with the received message.
In addition, when a CAN bus error or arbitration-lost for the transmitting message occurs in transmit/receive FIFO buffer full, the transmitting message is lost and retransmission for the message is not performed. The read point moves to the next message automatically.

The interrupt generation conditions for the GW FIFO buffers can be configured by configuring the CFDCFCCn.CFIM bit in the corresponding Common FIFO Configuration/Control Registers.

If CFDCFCCn.CFIM bit is 0, then RX interrupt flag is set when FIFO counter increments and reaches value configured by CFDCFCCn.CFIGCV and the TX interrupt flag is set when FIFO transmits the last message successfully.

If CFDCFCCn.CFIM bit is 1, then RX interrupt flag is set at the end of storage of every received message and TX interrupt flag is set if a message is successfully transmitted from the FIFO.

Common FIFO can set interrupt when:

- CAN frame transmitted is completed
- CAN frame reception is completed
- FIFO is in full status in RX mode or GW mode

When CFDCFCCEn.CFBME = 1, it becomes FIFO buffering mode, send data is stored in Common FIFO, and transmission is stopped. Transmission will be started if it is set as CFDCFCCEn.CFBME = 0.

The Common FIFO buffers configured in GW mode can be disabled by clearing the CFDCFCCn.CFE bit in the Common FIFO Configuration/Control Register. If this bit is cleared, the GW FIFO becomes empty as follows:

- Immediately if the message from the GW FIFO is neither scheduled for the next transmission nor in transmission
- Following the transmission completion, the detection of an error on the CAN bus, loss of arbitration or transition to Channel or Global Halt mode if the transmission from the GW FIFO is already scheduled for transmission or already in transmission

Other possible messages pending from the GW FIFO are lost.

Before CFDCFCCn.CFE is set again, ensure that the CFDCFSTSn.CFEMP bit is set and that there is no pending abort from the GW FIFO.

When the CFDCFCCn.CFE bit is cleared and the CFDCFSTSn.CFEMP bit is set, the message read and write pointers of the GW FIFO are cleared and are no longer active. Therefore, all messages in the GW FIFO buffers are lost and no further message can be stored into the GW FIFO.

In applications intended to be used as CAN-to-CAN gateways, it is useful if the Error State Indication (ESI) information of the routing messages is not replaced by the sending node Error State Indication. For this, each channel has the control function register CFDCnFDCFG.ESIC to replace their own ESI information by the routing ESI information.

Note: If the sending node is error passive, the ESI bit is sent anyway as error passive (ESI = 1).

(3) Interval Timer for FIFO Transmission

For each Common FIFO in TX or GW mode, it is possible to specify a delay between two consecutive messages that are configured for transmission from the same FIFO buffer. This delay is called interval time. This interval time starts after the first message has been successfully transmitted from the FIFO buffer after the CFDCFCCn.CFE bit is set.

When the Common FIFO in TX or GW mode is enabled, the first message is transmitted without considering this interval time.

The interval timer stops counting when:

- FIFO is disabled by clearing the CFDCFCCn.CFE bit
- CAN channel is in CH_RESET mode

The interval time is specified by the CFDCFCCn.CFITT value in the Common FIFO Configuration/Control Register and can be specified from 0 to 255 timer units.

The timer unit can be defined based on two different source clocks for the interval timer. To disable the interval timer for FIFO transmission, a value of 0 should be selected.

The timer source can be selected by the configuration bit CFITSS in the Common FIFO Configuration/Control Register. For the timer source the CAN bit timing clock of the FIFO related channel or a global reference clock can be selected.

If CAN channel bit time clock is configured as clock source and the CAN channel enters CH_HALT or CH_RESET or CH_SLEEP mode, the interval timer is stopped for that channel.

If peripheral clock is selected as interval timer clock source, then the interval timer is stopped only when the CAN channel is in CH_RESET or CH_SLEEP mode.

The reference clock can be used to configure the interval time in fixed time units. It is based on the peripheral clock. The reference clock prescaler value CFDGCFG.ITRCP in the Global Configuration Register defines the relation between the peripheral clock frequency/period and the reference clock period.

See **Table 30.38** for CFDGCFG.ITRCP configuration values to achieve different reference clock periods based on the peripheral clock frequency/period.

Table 30.38 Configuration Example for the FIFO Interval Timer Reference Clock

	Reference Clock		
Peripheral Clock (PCLKM)	1 µs	100 µs	500 µs
80 MHz / 12.5 ns	80	8000	40000
100 MHz / 10 ns	100	10000	50000

Additionally, the reference clock resolution can be specified by the interval timer reference clock resolution value CFDCFCCn.CFITR in the Common FIFO Configuration/Control Register.

The interval time is based on the reference clock period multiplied by the configured value (×1 or ×10).

The reference clock based interval timer can be used to follow the requirements of the ISO 15765-2 Separation Time. The whole range for the separation time from 100 µs to 127 ms can be covered.

The specified interval time starts after successful transmission event (after EOF7 state of the CAN protocol).

When the interval time has elapsed, the next transmission request is raised by the related TX/GW FIFO. Therefore, the interval time defines the minimum time between two messages transmitted from one FIFO.

The next message is sent at earliest after this interval time.

Figure 30.45 shows an example timing of the internal processing.

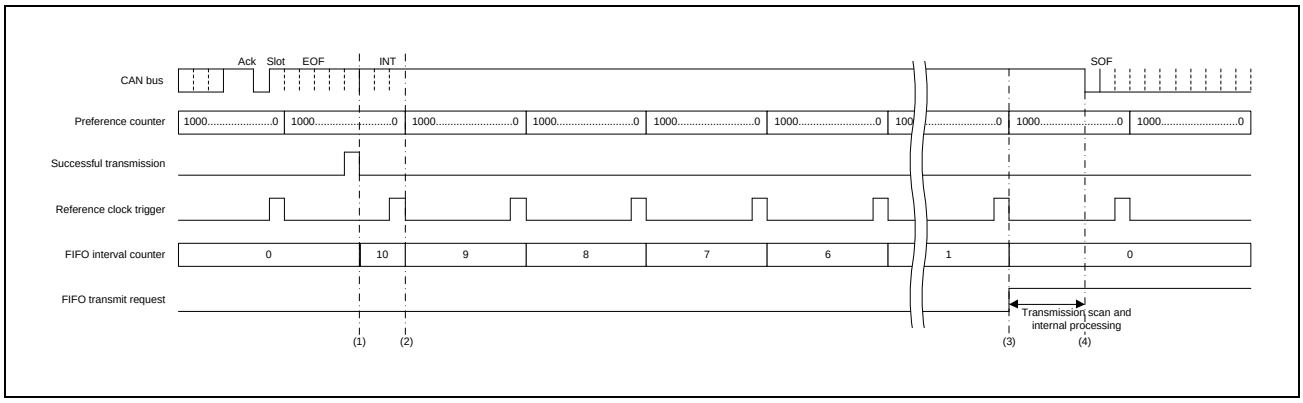


Figure 30.45 Example for Interval Processing Time

The configuration for the above timing is as follows:

- Peripheral clock frequency = 100 MHz
- Interval timer reference clock (CFDGCFG.ITRCP) = 1000 times
- Reference clock due to the above settings = 10 µs
- Common FIFO interval timer source selection (CFDCFCCn.CFITSS) = 0
- Common FIFO interval timer resolution (CFDCFCCn.CFITR) = 0
- Common FIFO interval transmission time (CFDCFCCn.CFITT) = 10 times
- Theoretical message separation interval = 100 µs

1. Internal FIFO interval timer is restarted with the occurrence of successful transmission result. This restart is not synchronized to the reference clock trigger. Therefore, the first interval is counting less or equal to one reference clock interval.
2. With the next reference clock trigger, the FIFO interval timer is decremented.
3. When the FIFO interval timer reached the value 0 the FIFO transmit request is set.
4. When the FIFO is selected for transmission, the transmission starts. Due to internal processing this usually takes less than 3 CAN bit time, between internal FIFO transmit request set 3 and actual transmission.

In the worst case when multi events such as reception scan, internal message routing, transmit scan on all channels occur, then it can take up to 432 peripheral clock cycles.

As shown in **Figure 30.45**, it is not guaranteed that the minimum interval is always equal to the configured value. If a minimum time must never be breached, configure CFDCFCCn.CFITT to the required minimum value + 1.

If additional TX message buffers or TX/GW FIFOs are configured for transmission for the same channel, the real delay between two messages transmitted from a TX FIFO can be much longer than specified by the interval time due to higher priority message transmission from these TX message buffers or TX/GW FIFOs.

Figure 30.46 shows a block diagram of the FIFO interval time generation circuit.

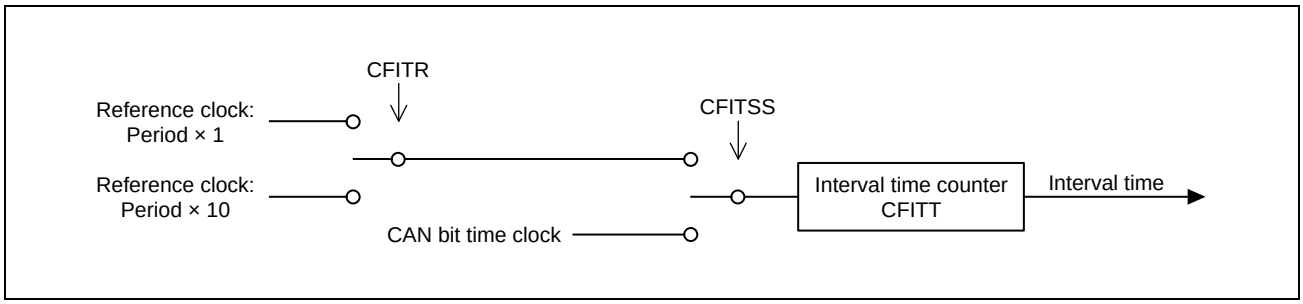


Figure 30.46 Block Diagram of FIFO Interval Timer

30.8.2.4 TX Queue

Each enabled TX Queue for a specific channel consists of 3 to 32 TX message buffers, which are accessed by one access window. One channel has 4 TX Queues. One TX Queue can be configured with a depth of 3 up to 32 buffers and it uses TX Message Buffer No. 0 as access window (referred to as TXQ0). The second TX Queue can be configured with a depth of 3 up to 32 buffers and it uses TX Message Buffer No. 31 as access window (referred to as TXQ1). The third TX Queue can be configured with a depth of 3 up to 32 buffers and it uses TX Message Buffer No. 32 as access window (referred to as TXQ2). The fourth TX Queue can be configured with a depth of 3 up to 32 buffers and it uses TX Message Buffer No. 63 as access window (referred to as TXQ3).

All TXQ0, TXQ1, TXQ2, and TXQ3 messages enter the priority comparison for the transmission, which should be only ID Priority (CFDGCFG.TPRI = 0).

The registers for TXQ0 are CFDTXQCC0n, CFDTXQSTS0n, and CFDTXQPCTR0n.
The registers for TXQ1 are CFDTXQCC1n, CFDTXQSTS1n, and CFDTXQPCTR1n.
The registers for TXQ2 are CFDTXQCC2n, CFDTXQSTS2n, and CFDTXQPCTR2n.
The registers for TXQ3 are CFDTXQCC3n, CFDTXQSTS3n, and CFDTXQPCTR3n.

When access window TX Message Buffer No.63 (TXQ3) or TX Message Buffer No.32 (TXQ2) or TX Message Buffer No.31 (TXQ1) or TX Message Buffer No.0 (TXQ0) is used, refer to the related access registers TX Message Buffer ID Registers (CFDTMIDn), TX Message Buffer Pointer Registers (CFDTMPTRn), TX Message Buffer Data Field 0 Registers (CFDTMDF0_n), and TX Message Buffer Data Field 1 Registers (CFDTMDF1_n).

The depth of each TXQ0 buffer can be configured by writing to the CFDTXQCC0n.TXQDC[4:0] bits of the TX Queue Configuration/Control Register.

TXQ0 can set from TXMB0 to TXMB31 as a queue buffer at the maximum.

The 31 available options for depth configuration are:

- H’00: TX Queue disabled
- H’01: Reserved
- H’02: 3 messages
  ⋮
- H’1D: 30 messages
- H’1E: 31 messages
- H’1F: 32 messages

The depth of each TXQ1 buffer can be configured by writing to the CFDTXQCC1n.TXQDC[4:0] bits of the TX Queue Configuration/Control Register.

TXQ1 can set from TXMB31 to TXMB0 as a queue buffer at the maximum.

The 31 available options for depth configuration are:

- H’00: TX Queue disabled
- H’01: Reserved
- H’02: 3 messages
  ⋮
- H’1D: 30 messages
- H’1E: 31 messages
- H’1F: 32 messages

The depth of each TXQ2 buffer can be configured by writing to the CFDTXQCC2n.TXQDC[4:0] bits of the TX Queue Configuration/Control Register.

TXQ2 can set from TXMB32 to TXMB63 as a queue buffer at the maximum.

The 31 available options for depth configuration are:

- H’00: TX Queue disabled
- H’01: Reserved
- H’02: 3 messages
  ⋮
- H’1D: 30 messages
- H’1E: 31 messages
- H’1F: 32 messages

The depth of each TXQ3 buffer can be configured by writing to the CFDTXQCC3n.TXQDC[4:0] bits of the TX Queue Configuration/Control Register.

TXQ3 can set from TXMB63 to TXMB32 as a queue buffer at the maximum.

The 31 available options for depth configuration are:

- H’00: TX Queue disabled
- H’01: Reserved
- H’02: 3 messages
  ⋮
- H’1D: 30 messages
- H’1E: 31 messages
- H’1F: 32 messages

When using TXQ1 and TXQ0 simultaneously, the depth of TXQ is 32 or less in total.

When using TXQ3 and TXQ2 simultaneously, the depth of TXQ is 32 or less in total.

Do not access all the TX message buffers forming the TX Queue directly (except TX Message Buffer No. 63, TX Message Buffer No. 32, TX Message Buffer No. 31 and TX Message Buffer No. 0, which acts as TX Queue access window).

When CFDGAFLP0n.GAFLSRD i (i = 0 to 2) is set and the CFDTXQCCin.TXQGWE (i = 0 to 2, n = 0, 1) is also set, a receiving frame is stored in the target TXQ as send data by routing.

When CFDTXQCCn.TXQOWE bit is 1, the TX Queue is in TX Queue overwrite mode. If the message of the same ID is stored in TX Queue when a frame is received and it is stored in TX Queue, an old message is overwritten by a new message. Therefore, an old message is not transmitted. When the old message of the same ID is transmitting and a CAN bus error and an arbitration-lost occur, the message of old ID is not resent.

When using the function in GW mode and TX Queue overwrite mode, the depth of TXQ (CFDTXQCC0n.TXQDC) should be configured to the value which is the various number of ID that is used in the TX Queue plus 3. If it accesses by routing in gateway mode when a TXQ buffer is full, CFDTXQSTS.TXQMLT is set and send data is thrown away.

The function is valid for the standard ID frame and is invalid for the extended ID frame.

Explanation of operation of the TX Queue with same ID over-writing function in GW mode is shown in **Figure 30.47**.

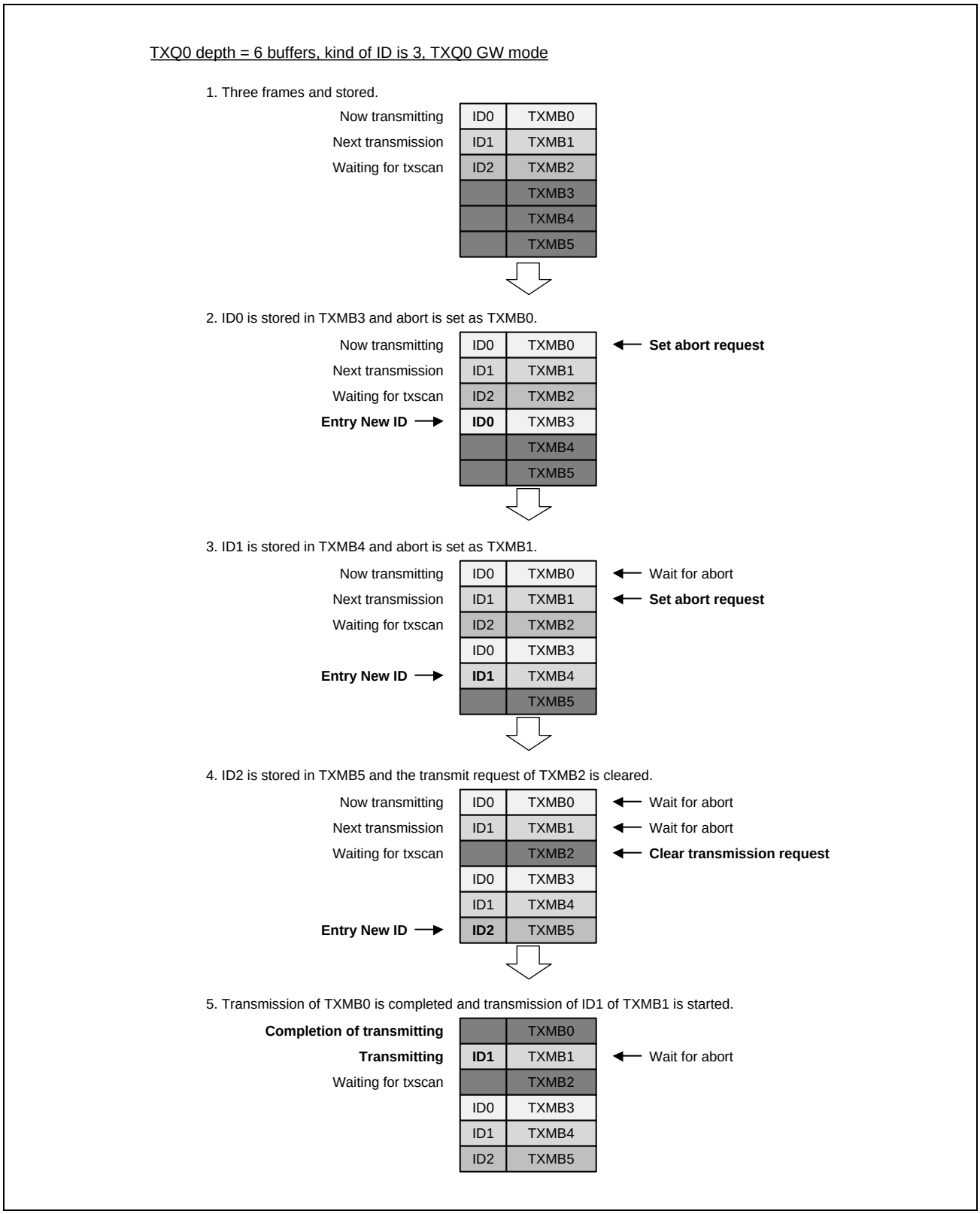


Figure 30.47 Operation of TXQ0 in Gateway Mode

When a system writes in TXQ, a system should write in send data, after checking the state of TXQ.

Do not access or configure the related TX Message Buffer Control Registers.

The messages stored to the TX Queue access window are internally stored to a free buffer of the TX Queue.

When the buffer is full then no further access should be done to the queue, until it is no longer full. If it accesses by software writing in when the buffer of TXQ is full, send data is overwritten.

The TX Queue can be disabled by clearing the TXQE bit in the TX Queue Configuration/Control Register. If this bit is cleared, the TX Queue Empty flag is set as follows:

- Immediately if the message from the TX Queue is neither scheduled for the next transmission nor in transmission
- Following the transmission completion, the detection of an error on the CAN bus, loss of arbitration or transition to Channel or Global Halt mode if the transmission from the TX Queue is already scheduled for transmission or already in transmission.

Note: The TX Queue is disabled only when the Empty flag is set after clearing the TXQE bit for the corresponding TX Queue.

Other possible messages pending from the TX Queue are lost and their transmission must be requested again.

Before TXQE is set again, ensure that the CFDTXQSTSn.TXQEMP bit is set and that there is no pending abort from the TX Queue.

When the TXQE bit is cleared, all messages in the TX Queue buffers are lost and no further message should be stored into the TX Queue.

When a message has been stored to the TX Queue, H’FF must be written into the TX Queue Pointer Control Register. This sets the transmit request automatically and changes the internal message buffer pointer to the next free message buffer location of the TX Queue.

Note: If two messages with the same identifier are stored in the TX Queue, then the order of transmission of these messages can be different from the order in which they were stored in the TX Queue.

To avoid this condition, it is important to confirm that the previous message with the same ID was successfully transmitted before a new message with the same identifier is stored in the TX Queue. Or if TX queue overwrite mode is used, the frame of the same ID is rewritten on a new frame.

For the TX Queue a dedicated interrupt can be enabled by setting the TXQIE bit of the TX Queue Configuration/Control Register.

The interrupt mode can be configured with the CFDTXQCCn.TXQIM bit of the same register either to generate an interrupt for every transmitted message or for the last transmitted message.

The TX Queue transmission request procedure after configuration is shown in **Figure 30.48**.

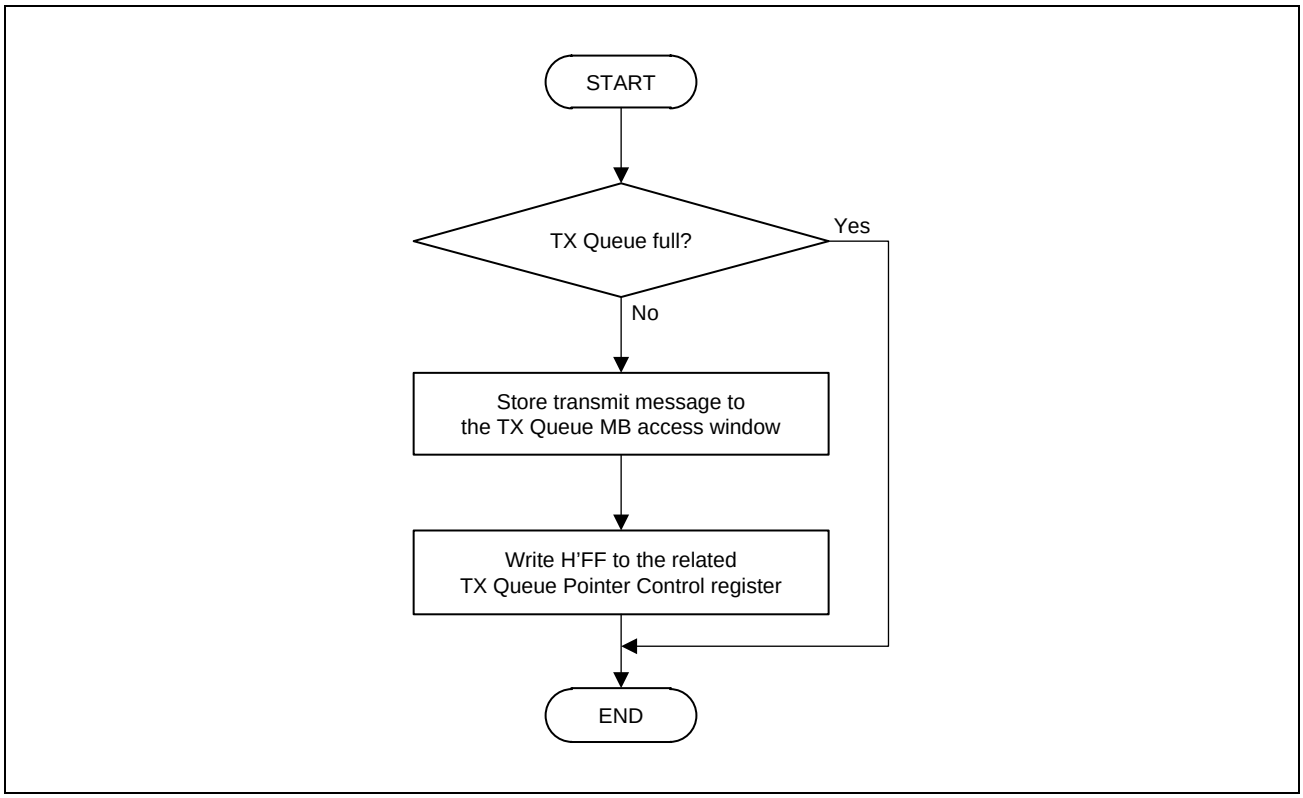


Figure 30.48 TX Queue Transmission Request

TXQ Name	Access Window	Range Width	Direction	Hardware Routing Access Point	CPU Access Point	DMA Access Point	Note
TXQ0	TXMB0	0, 3 - 32	TXMB0 → TXMB31	Yes	Yes	Yes	When using both TXQ0/1, the total number of stages is 32 or less
TXQ1	TXMB31	0, 3 - 32	TXMB31 → TXMB0	Yes	Yes	No	
TXQ2	TXMB32	0, 3 - 32	TXMB32 → TXMB63	Yes	Yes	No	When using both TXQ2/3, the total number of stages is 32 or less
TXQ3	TXMB63	0, 3 - 32	TXMB63 → TXMB32	No	Yes	Yes	

TXQ0 can use hardware routing, CPU access, and DMA access.

Hardware routing access, CPU access, and DMA access should not be used simultaneously. Choose one access method.

30.8.2.5 TX History List

The TX History List function records the information of the successfully transmitted message in the TX History List buffers for each CAN channel. Two TX History List buffers are provided for each CAN channel and each TX History List buffer can store up to 16 TX History List entries for a CAN channel.

The CFDTHLCCn.THLDTE bit of the TX History List Configuration/Control Register can be used to configure if only message information from TX FIFOs/TX Queue is stored or if all transmit message information from TX Queue, TX FIFO or normal TX message buffers should be stored in the TX History List for a CAN channel.

When a CFDTHLCCn.THLDGE bit is set up, the information on all the frames transmitted in GW mode is stored in TX History List.

Each transmit message can be individually configured for acceptance to the TX History List by the CFDCFID.THLEN bit in the Message Buffer Pointer Register.

The message information is stored to the TX History List Buffer of a CAN channel after the message is successfully transmitted on that CAN channel.

Storing to the List is not synchronized with the status of CFDTMSTSn.TMTRF[1:0] bits in the TX Message Buffer Status Register.

Due to internal processing, the storage to the List can happen with a delay after the successful transmission indication.

Storing the TX History List data can be recognized by the condition that the THLIF bit is set to 1 when the THLIE is configured to 1 or when the TX History List counter CFDTHLSTSn.THLMC[5:0] is increased.

The delay time is dependent on the number of channels due to internal processing.

- Maximum delay time from setting the CFDTMSTSn.TMTRF to storing the TX History List data is 224 peripheral bus clock cycles.

The History list records the following information of the transmitted message:

- Buffer type:
  - 001b: TX Message Buffer
  - 010b: TX FIFO
  - 100b: TX Queue
- Buffer number:
  TX Message Buffer, TX Queue Message Buffer or TX Message Buffer Link for the Common FIFO Buffer from which the transmission occurred. The number depends on the buffer type, see **Table 30.39**.
- Transmission ID:
  Transmission pointer stored in the transmission message
- Transmit timestamp:
  Message timestamp captured at the capture point as configured by CFDGFDCFG.TSCCFG.
- Transmission information label:
  Transmission information label stored in the transmission message.
- Transmit gateway buffer indication:
  For data transmitted from the gateway, CFDTHLACC0n.TGW bit is set to 1.

Table 30.39 TX History List Buffer Number Entry

CFDTHLACC0n.BT[2:0] Buffer Type		
001b	101b	100b
TX Message Buffer	TX FIFO	TX Queue
TXMB0	Number shown corresponds to the common FIFO. TX Message Buffer Link CFTML of the related Common FIFO Configuration	Number shown corresponds to the message buffer belonging to the TX Queue for which the frame was transmitted.
TXMB1		
TXMB2		
TXMB3		
⋮		
TXMB30		
TXMB31		
TXMB32		
TXMB33		
TXMB34		
⋮		
TXMB61		
TXMB62		
TXMB63		

The Transmission ID entry is used to identify which message of a TX FIFO or TX Queue has been successfully transmitted because the TX FIFO or TX Queue number alone is not sufficient.

Therefore, a unique number can be attached to each transmission message stored in a TX FIFO or TX Queue. This unique identification number should be written to the CFDCFFDCSTSn.CFPTR[15:0] part of the Common FIFO Access Pointer Register for a TX FIFO or to the CFDTMFDCTRn.TMPTR[15:0] part of the TX Message Buffer Pointer Register of the TX Queue access window message buffer.

When the message is successfully transmitted, the identification number is then stored together with the other message related information to the TX History List and can be read with the Transmission ID (TID) of the TX History List Access Register.

Also for normal TX message buffers, the CFDTMFDCTRn.TMPTR[15:0] part of the TX Message Buffer Pointer Register is stored in the Transmission History List. Information label is the same.

Figure 30.49 shows a transmission preparation flow when TX History List is used.

Read access to the TX History List Access Register is done for every single entry.

After reading one entry, H’FF must be written to the corresponding TX History List Pointer Control Register to be able to access the next entry until TX History List is empty.

Figure 30.50 shows an example flow for processing the TX History List information.

The TX History Lists have dedicated interrupts, which can be configured with the CFDTHLCCn.THLIM bit of the corresponding TX History List Configuration/Control Registers and enabled with the CFDTHLCCn.THLIE bit of the same registers, either to generate an interrupt when the history list reached a filling level of 75% or for every new TX History List entry.

An entry lost indication is flagged by the CFDTHLSTSn.THLELT bit in the TX History List Status Register.

Status of this bit is also shown by the THLES bit in the Global Error Flag Register.

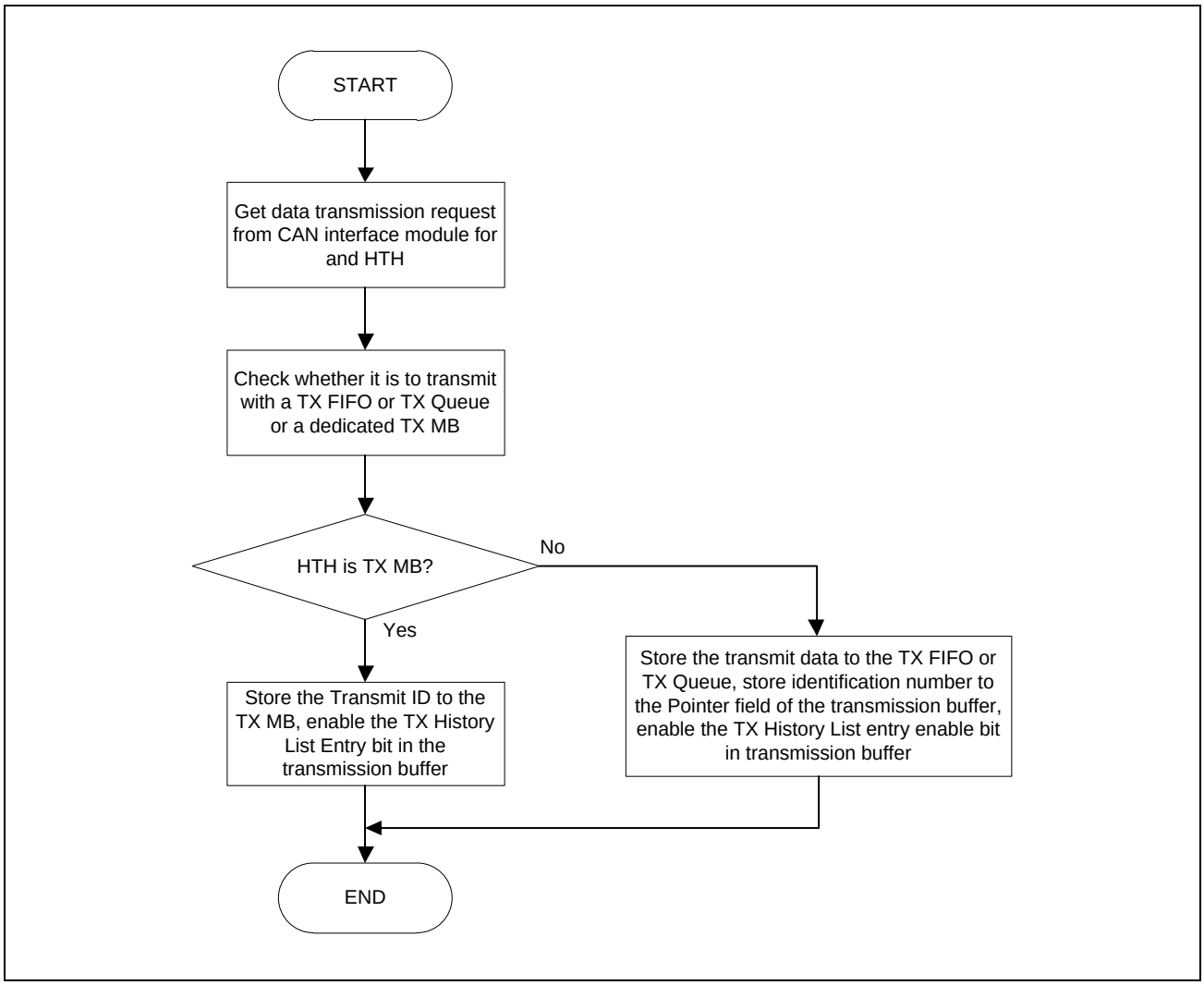


Figure 30.49 Transmission Preparation Flow when TX History List is Used

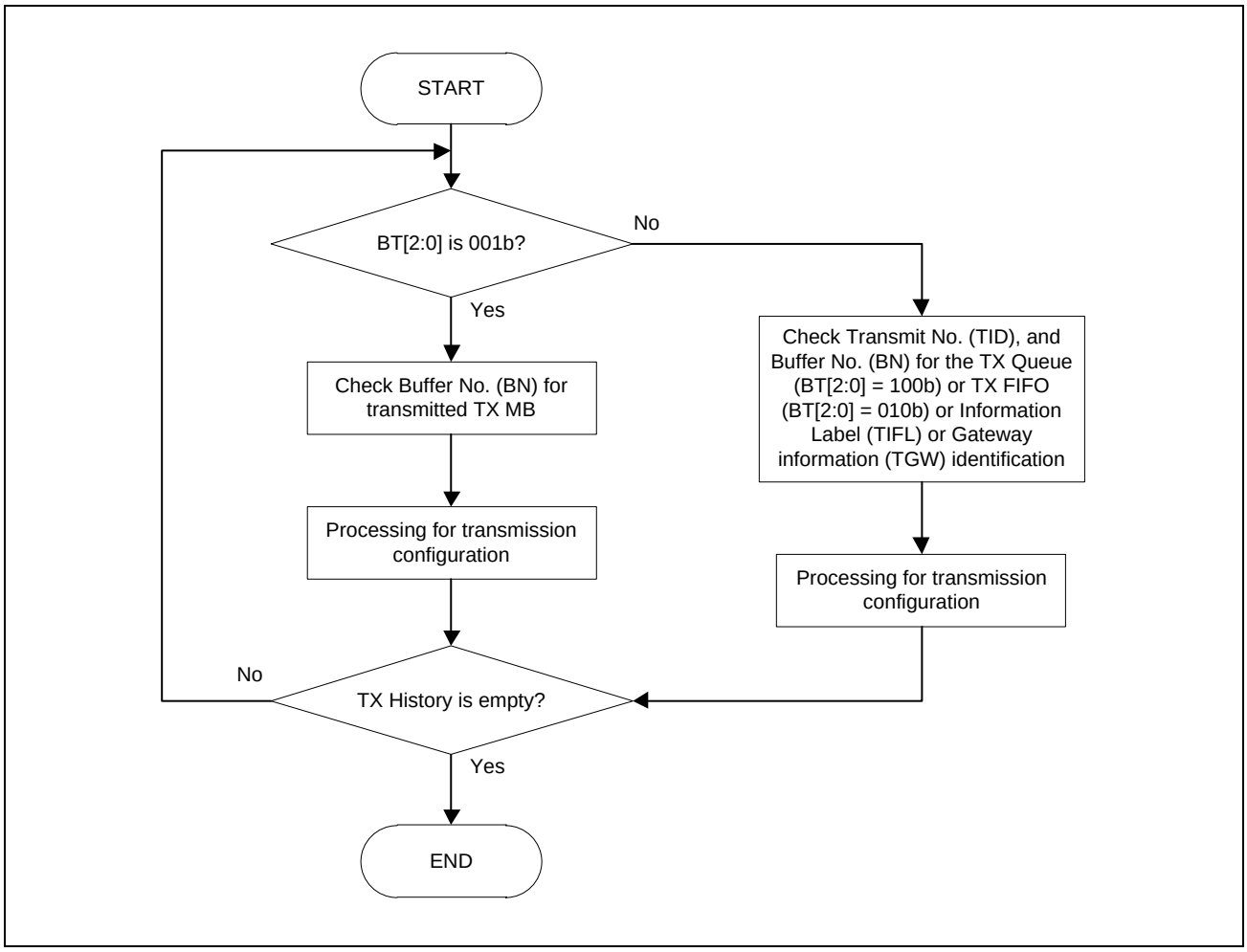


Figure 30.50 Example Flow for Processing TX History List Information

30.8.2.6 TX Data Padding

If the data length code (DLC) of the transmitting message has a higher number of data bytes than the buffer size, then the data bytes beyond the restricted range are replaced by bytes with the value of CC HEX.

This can happen for Common FIFOs configured as (TX or GW) when the transmit message DLC is higher than CFDCFCCn.CFPLS.

30.9 Test Mode

The CAN-FD module can be configured into test modes to allow testing of certain features. These features are provided only for special purposes and care must be taken when configuring the CAN-FD module in test modes.

All test modes are mutually exclusive unless it is explicitly stated that some functions can be enabled across other test modes.

Do not enable any combinations of the various test modes specified in this section.

The test modes can be broadly split into 2 groups:

- Channel specific test modes
- Global test modes

30.9.1 Channel Specific Test Modes

Each CAN channel can be configured into the following test modes:

- Basic test mode
- Listen-only mode
- Self-test mode 0 (External Loop Back mode)
- Self-test mode 1 (Internal Loop Back mode)
- Restricted operation mode

30.9.1.1 Basic Test Mode

The Basic test mode should be used when a particular test setting must be enabled other than when in Listen-Only and Self- test modes.

30.9.1.2 Listen-Only Mode

The ISO 11898-1 recommends an optional Bus-monitoring mode. In this mode, the CAN channel is able to receive valid data frames and valid remote frames. However, it sends only recessive bits on the CAN bus and is not allowed to transmit.

If the CAN engine is required to send a dominant bit (ACK bit, Overload flag, Active Error flag), the bit is routed internally so that the CAN engine monitors this as dominant. The external TX pin remains in recessive state.

This mode can be used for baud rate detection. In this mode, an error interrupt is generated if a bus error occurs and the interrupt is enabled.

In this mode, it is not permitted to request transmission from any normal TX Message Buffer or TX-/GW-FIFO of this channel.

Note: If a message is stored in GW FIFO or Routing TXQ, ensure that the transmitting channel is not in Listen-only mode so that transmission is not requested for this channel from the GW FIFO or Routing TXQ.

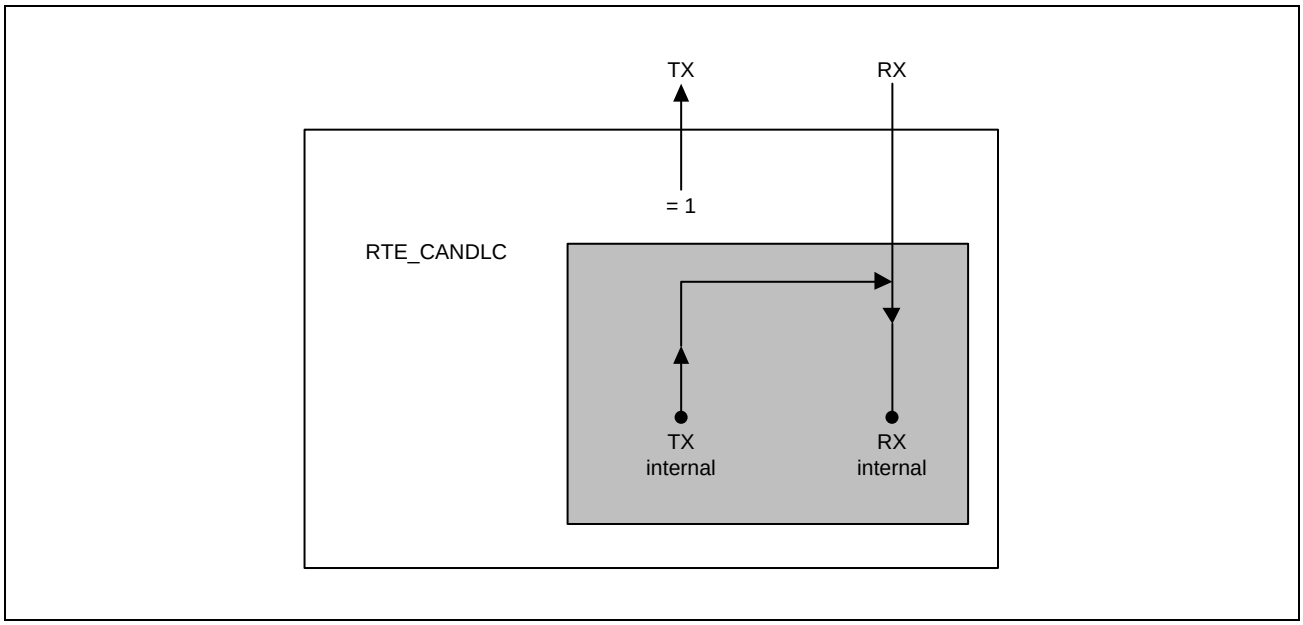


Figure 30.51 Listen-Only Mode

30.9.1.3 Self-Test Mode 0 (External Loop Back Mode)

In Self-test mode 0, the CAN engine treats its own transmitted messages as received messages through the CAN transceiver and can store them into its receive message buffers.

To be independent from external stimulation, the engine generates its own Acknowledge bit.

This test can be used for CAN transceiver tests.

The RX/TX pins should be connected to the transceiver.

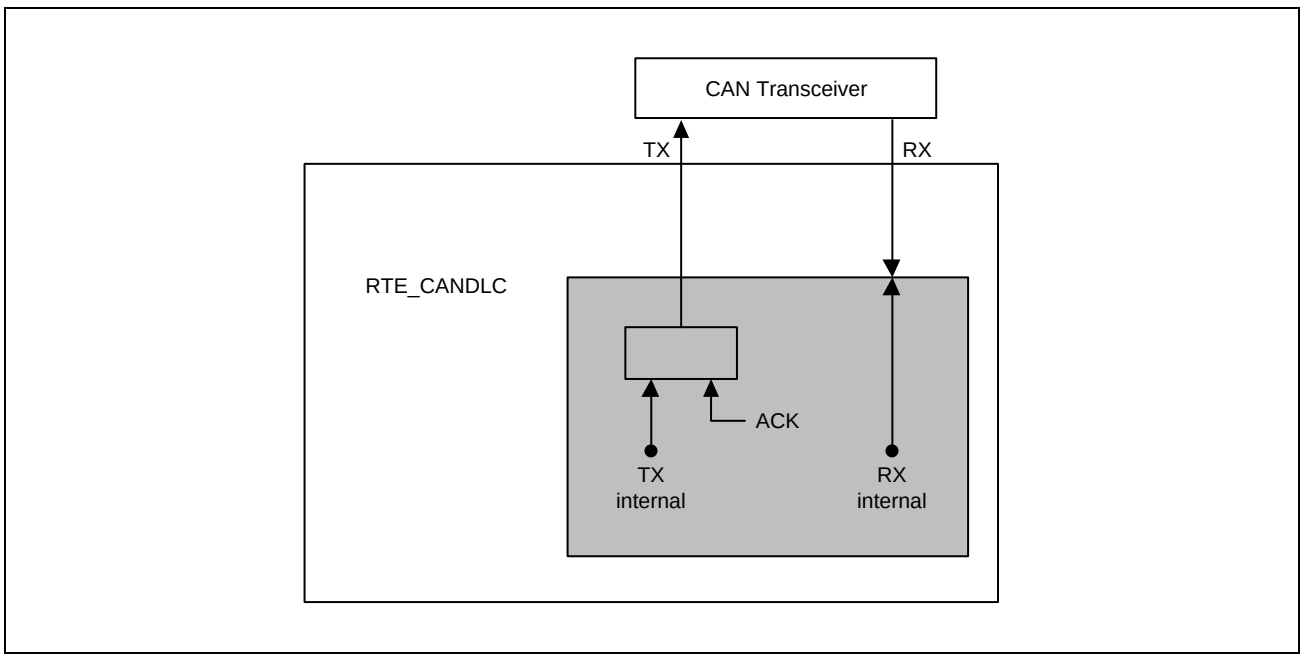


Figure 30.52 Self-Test Mode 0 (External Loop Back Mode)

30.9.1.4 Self-Test Mode 1 (Internal Loop Back Mode)

In Self-test mode 1, the CAN engine treats its own transmitted messages as received messages and stores them into the receive buffer. This mode is provided for self-test functions. To be independent from external stimulation, the CAN engine generates its own Acknowledge bit. In this mode the CAN engine performs an internal feedback from TX internal to RX internal. The actual value of the external RX input is disregarded by the CAN engine.

The external TX pin outputs only recessive bits.

The RX/TX pins do not need to be connected to the CAN bus or any external device.

Note: The channel pins are also disconnected from the internal CAN bus communication line.

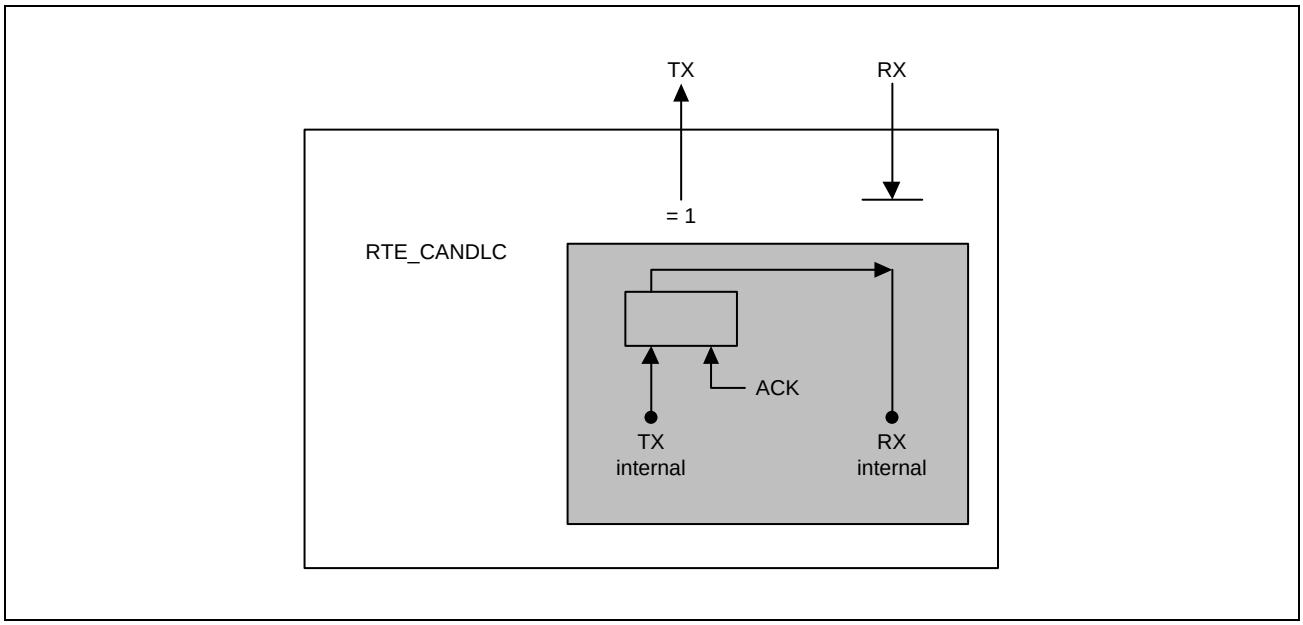


Figure 30.53 Self-Test Mode 1 (Internal Loop Back Mode)

30.9.1.5 Restricted Operation Mode

In Restricted operation mode, the CAN node is able to receive valid data and remote frames that generates the Acknowledge bit.

Active error and overload frames cannot be transmitted, instead it waits for the occurrence of bus idle condition to resynchronize itself to the CAN communication after an error or overload condition occurs.

Additionally, the Receive and Transmit Error Counter (REC and TEC) are frozen independently from the occurrence of errors.

The mode is specified as in ISO 11898-1. However, it is permitted to set any requested transmit.

30.9.2 Global Test Modes

The CAN-FD module can be configured into the following test modes:

- RAM test mode
- Internal CAN bus communication mode
- CRC error test

These test modes are protected by a special software procedure to enable the mode. This software procedure enables write access to the test mode by a specific unlock key, the related unlock key can be seen in the following table.

Test Mode	Unlock Key 1	Unlock Key 2
RAM test mode	H'7575	H'8A8A

If the software sequence of the two consecutive unlock key write accesses (half-word or word accesses) is interrupted by any other write access to the SFR or if incorrect data is written to the Global Unlock Key Register then the corresponding test mode cannot be set and the sequence must be restarted.

After the two unlock key write accesses, the next write access should be to set the corresponding Test Mode Enable bit. If this is not followed, the unlock mechanism resets and the Test Mode Enable bit cannot be set. At this time, the unlock sequence must be restarted.

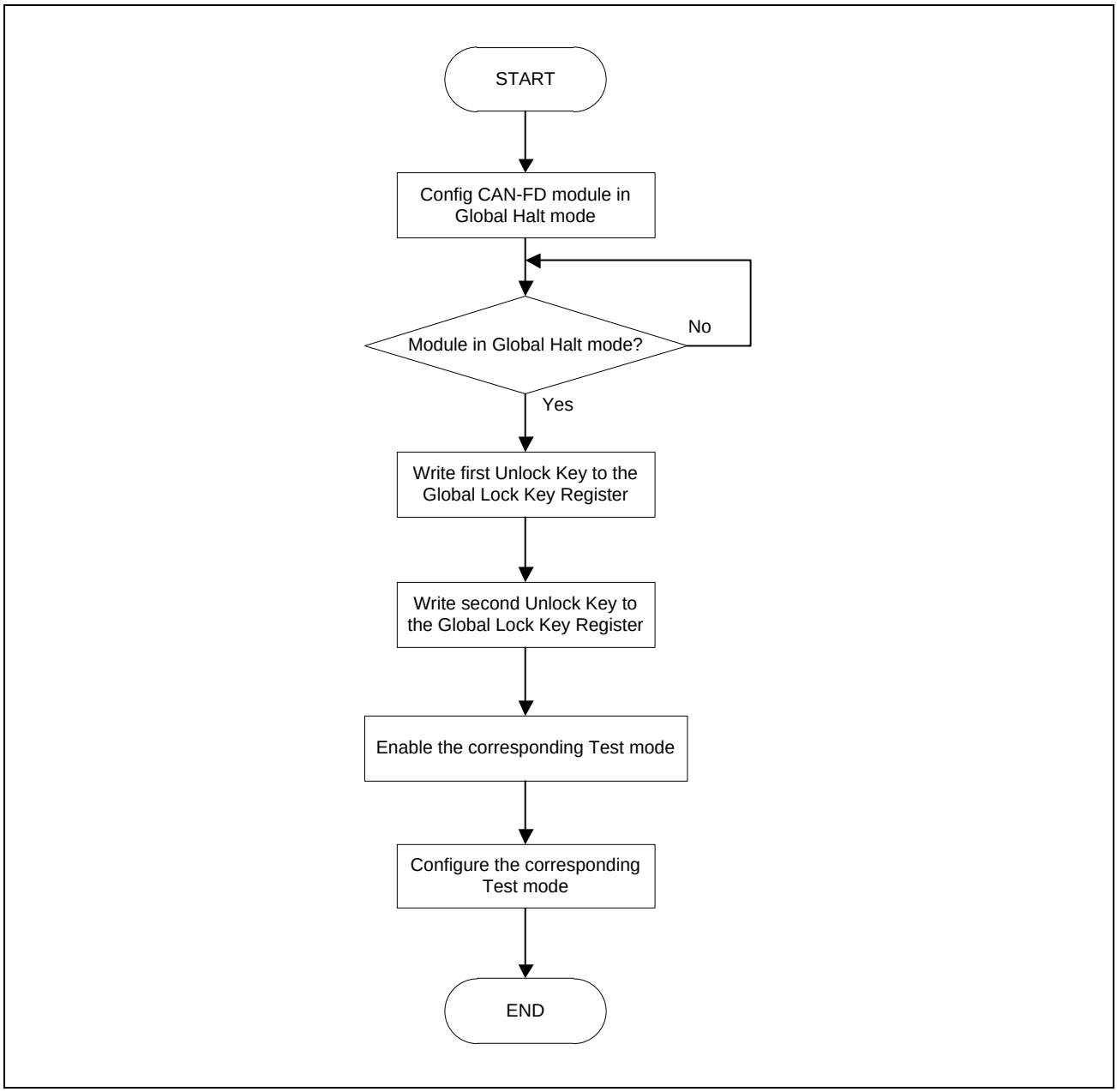


Figure 30.54 Unlock Software Protection Routine

30.9.2.1 RAM Test Mode

The CAN-FD module can be configured in RAM test mode by setting the CFDGTSTCTR.RTME bit in the Global Test Control Register when the corresponding lock key is previously written. This is a special test mode, in which, the complete RAM area can be accessed.

In this mode, the RAM area is split into number of pages (pn) of 256 bytes each. Which can be accessed with the CFDRPGACCn register.

The page should be selected for read/write access by writing to the CFDGTSTCFG.RTMPS[9:0] bits in the Global Test Control Register. Data can then be read from or written into the RAM Test Page Access Registers.

Figure 30.55 shows the structure of the pages in the RAM when performing a RAM test mode.

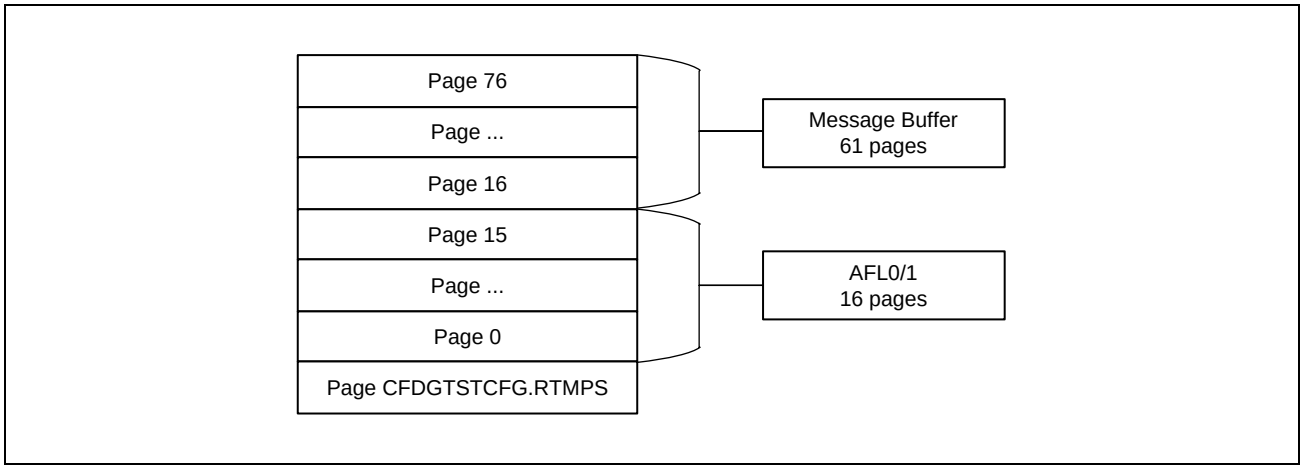


Figure 30.55 RAM Page Structure

The total available RAM size for a 2-CAN channel version is 4096 bytes for the AFL RAM and 15488 bytes for the Message Buffer RAM.

AFL RAM0/1 can treat RAM test mode as one RAM.

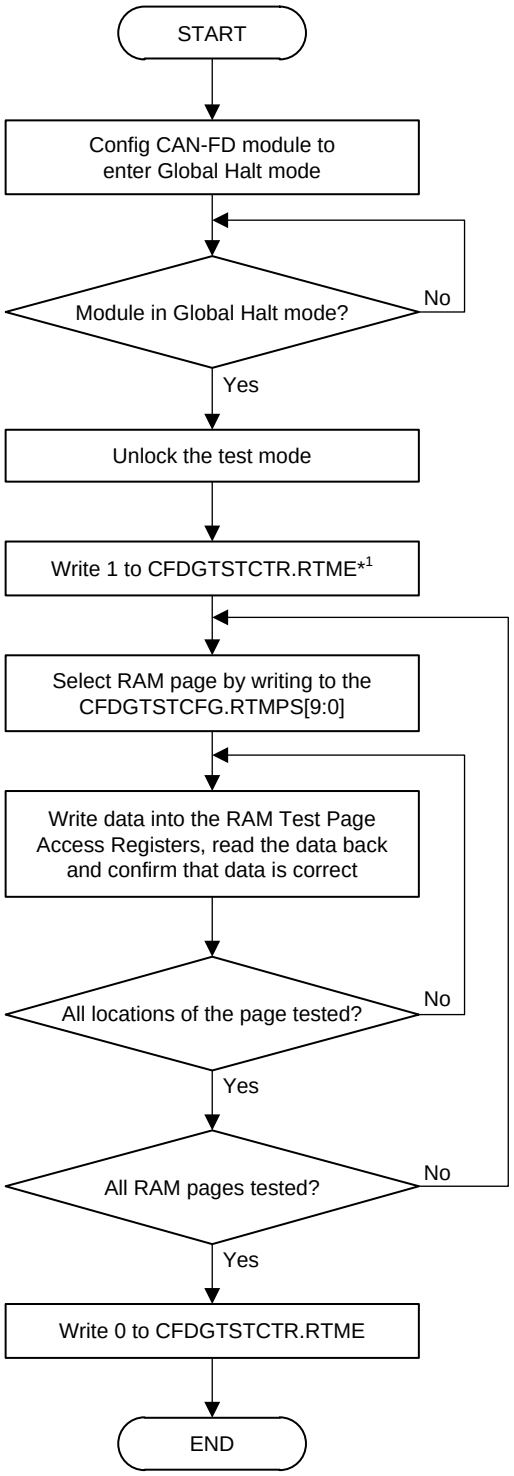
The number of pages (pn) and CFDGTSTCFG.RTMPS[9:0] values for the AFL and MB RAMs are calculated in the following way:

pn = ceil (total RAM size in bytes/number of bytes per page)

- AFL RAM:
  pn = ceil (4096/256) = 16 pages
  CFDGTSTCFG.RTMPS[9:0] = 0 to 15 (H’00F) inclusive
- MB RAM:
  pn = ceil (15488/256) = 61 pages
  CFDGTSTCFG.RTMPS[9:0] = 16 to 76 (H’04C) inclusive

Do not access more than 128 bytes of RAM on the last page (RTMPS[9:0] = H’04C).

Figure 30.56 shows the software flow for RAM test mode.



Note 1. Change into the following status before changing to RAMTEST.
- Cancel of a request of transmission
- Disable of all the FIFO and TXQ
- Clear of the receiving flag of a receiving buffer

Figure 30.56 Software Flow of RAM Test Mode

To exit this test mode, clear the CFDGTSTCTR.RTME bit. The CFDGTSTCTR.RTME bit is cleared by writing 0 to it.

The CFDGTSTCTR.RTME bit is cleared automatically when the CAN-FD module enters Global Reset mode from the test mode.

30.9.2.2 Internal CAN Bus Communication Mode

The CAN-FD module can be configured in Internal CAN bus communication test mode by setting the CFDGTSTCTR.ICBCTME bit in the Global Test Control Register. This is a special test mode, in which the CAN channels can be connected together internally to generate a CAN cluster within the CAN-FD module.

Only use the following sequence to enter Internal CAN bus communication test mode:

1. Configure all channels in Halt mode and check that all channels have entered Global Halt mode.
2. Write data into the Global Test Configuration Register to select the channels participating in the Internal CAN bus communication test.
3. Set the CFDGTSTCTR.ICBCTME bit of the Global Test Control Register.
4. Check that CFDGTSTCTR.ICBCTME bit is set in the Global Test Control Register.

In this mode, the TXD outputs of the channels participating (configured) in Internal CAN bus communication mode are connected together using AND gate. The output of the AND gate is connected to the RXD inputs of all participating channels to create a CAN cluster within the CAN-FD module. The channels are isolated from the external CAN bus while the CAN-FD module is in this test mode.

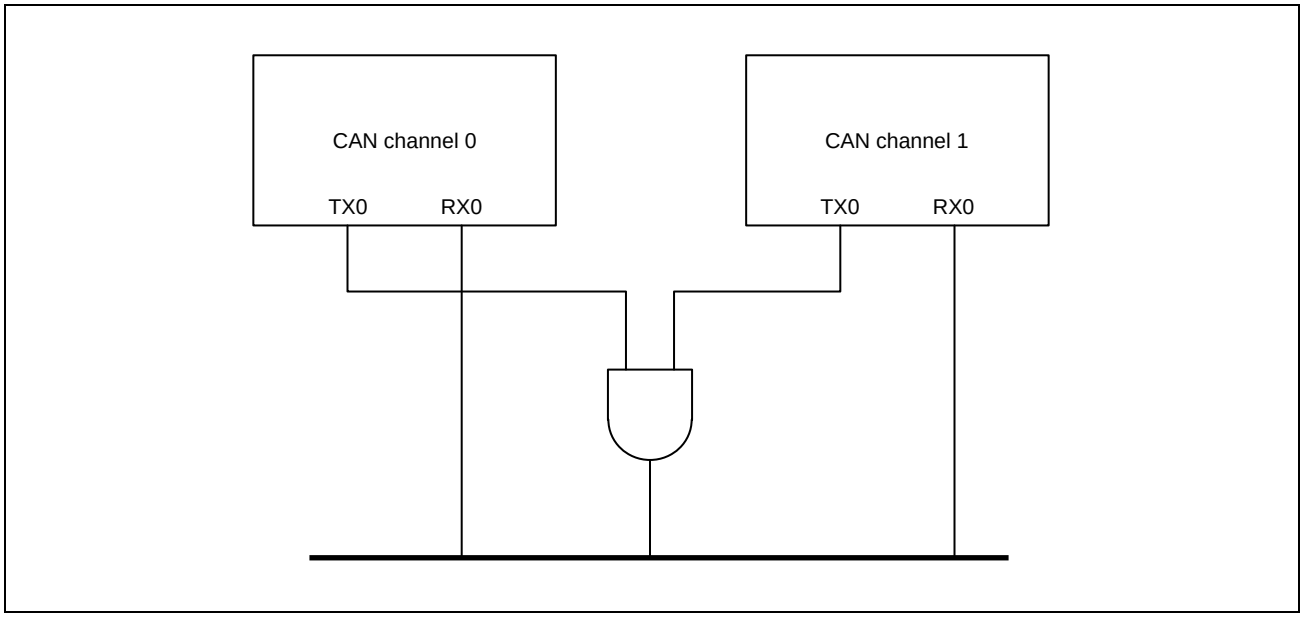


Figure 30.57 Internal CAN Bus Connections

The AFL, Flat RX message buffers, FIFO buffers, Flat TX message buffers, and various registers can now be configured as normal to start communication between channels.

The channels not participating in Internal CAN bus should only be configured in Halt mode.

30.9.2.3 CRC Error Test

After the CAN-FD module has been configured in Internal CAN bus communication test mode, use the following sequence to perform CRC error testing. In the following sequence, channel x is the reference transmitter CAN-FD module and channel y is the receiver CAN-FD module where (x, y = [0, …, n] and x ≠ y):

1. Configure channel x node to transmit one reference message.
2. Set the CFDCyCTR.CRCT bit to 1, in order to invert the first bit of the incoming bit stream from channel x.
3. Set the CFDTMCx.TMTR.
4. Read either CFDCyERFL.CRCREG or CFDCyFDCRC.CRCREG (depending on the received frame type: Classical or FD). The value should be different from the received CRC value of the reference message from channel x.
5. Check that CFDCyERFL.CERR is 1.

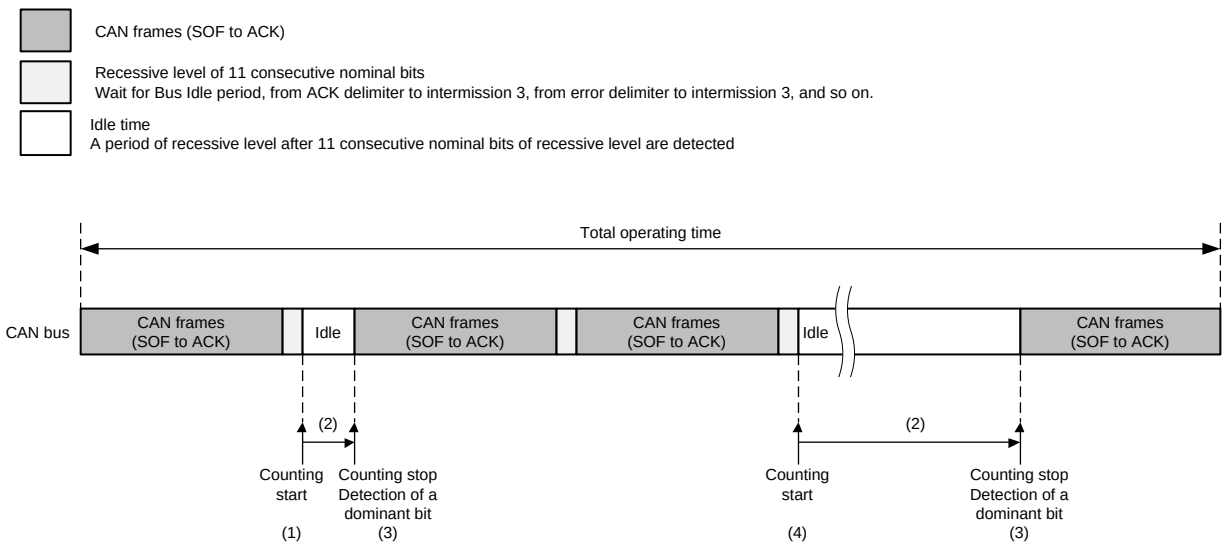
As the CRC generator logic is shared for RX and TX there is no need to create a separate TX CRC error test.

30.10 Bus Traffic Measurement

The idle time of the CAN bus can be measured using PCLKM or PCLKCAN. Bus traffic can be calculated based on the measurement results.

30.10.1 How to Count the CAN Bus Idle Time

Figure 30.58 shows the concept of measuring the idle time of the CAN bus.



(1) With the PCLKM clock, counting of the idle time starts after a recessive level input of 11 consecutive nominal bits are detected.
(2) The counting continues until a dominant level is detected on the CAN bus.
(3) The counting stops after a dominant level is detected on the CAN bus.
(4) After that, the counting restarts when 11 consecutive nominal bits are detected.

Figure 30.58 How to Measure Idle Time of the CAN Bus

30.10.2 Operations and Measurement Procedure

The following procedure is for measuring the idle time of the CAN bus.

1. The channel to be measured transits to operation mode.
2. Write 1 to CFDCnBLCT.BLCE bit to set the measuring counter to operating mode.
3. Write 1 to CFDCnBLCT.BLCLD bit to clear the counter register.
4. Detect a recessive level input of 11 consecutive nominal bits.
5. Start counting the bus idle time.
6. Detect a dominant level.
7. The counting stops.
8. Detect a recessive level input of 11 consecutive nominal bits.
9. The counting starts.

10. Write 1 to CFDCnBLCT.BLCLD bit to clear the counter register and simultaneously load the counter value to CFDCnBLSTS.
11. Read the value of CFDCnBLSTS.

To stop the measurement counter, write 0 to CFDCnBLCT.BLCE bit.

To initialize the counter, write 1 to CFDCnBLCT.BLCLD bit.

This measurement is enabled when the channel to be measured is in Operation mode.

When the relevant channels are in reset mode, the counter does not operate.

Also, accurate measurements are not available in test mode.

Write 1 to CFDCnBLCT.BLCLD bit to clear the counter register and simultaneously load the value of the counter to CFDCnBLSTS.

The lower 3 bits of CFDCnBLSTS are fixed to 0.

Based on the values of the counter, software can calculate the CAN bus traffic according to the following formulas:

$$\frac{(\text{Total operating time} - \text{Total idle time})}{\text{Total operating time}} = \frac{\text{Bus operating time}}{\text{Total operating time}} = \text{Bus usage ratio}$$

- Total idle time: a value read from CFDCnBLSTS × a clock cycle of PCLKM
- Total operating time: a setting interval of CFDCnBLCT.BLCLD bit

Example: Below is a calculation example under the following conditions:

- Conditions: nominal bit rate = 1 Mbps
- PCLKCAN clock = 40 MHz (= 25 ns)
- A setting interval of CFDCnBLCT.BLCLD bit = cycle of 1 ms
- A read value of CFDCnBLSTS register = H’4E20

$$\frac{(\text{Total operating time} - \text{Total idle time})}{\text{Total operating time}} = \frac{(1000000\ \text{ns} - 20000 \times 25\ \text{ns})}{1000000\ \text{ns}} = 50\%$$

31. Gigabit Ethernet Interface

Gigabit Ethernet Interface includes Ethernet controller (E-MAC) that conforms to the definition of the MAC (Media Access Control) layer for Ethernet in the IEEE 802.3 standard. When connected with a physical-layer LSI chip (PHY-LSI) that complies with the standard, the E-MAC is able to transmit and receive Ethernet (IEEE 802.3) frames. The E-MAC has a single MAC layer interface.

Internal TCP/IP Offload Engine (TOE) calculates Checksum by hardware and has functions filtering Ethernet Frames and automatically responding specific frames. Dedicated Direct memory access controller (DMAC) for transferring transmitted Ethernet frames to and received Ethernet frames from respective storage areas in the URAM at high speed.

31.1 Features

31.1.1 Specification

- 2 channels
- Supports transfer at 1000 Mbps and 100 Mbps, 10 Mbps
- Supports filtering of Ethernet frames
- Supports interface conforming to IEEE802.3 PHY RGMII (Reduced Gigabit Media Independent Interface)
- Supports interface conforming to IEEE802.3 PHYMII (Media Independent Interface)

Table 31.1 lists the features of the Gigabit Ethernet Interface block.

Table 31.1 Specifications (Functions) (1/2)

IP Name	1. Function
E-MAC	(1) Transmission and reception of Ethernet (IEEE 802.3) frames (2) Supports transfer at 10, 100 and 1000 Mbps in Full-duplex Mode, 10, 100 Mbps in Half-duplex Mode (3) Flow control conforming with the IEEE 802.3x standard (4) Supports MII (Media Independent Interface) and RGMII (Reduced Gigabit Media Independent Interface) (5) Transmission/Reception Low Power Idle Code
TOE	(1) Calculation of Checksum for following part of frames – IPv4 Header – IPv4 TCP/UDP/ICMP – IPv6 TCP/UDP/ICMP (2) Filtering of Ethernet Frames The Possible Configuration of Filter Condition are followings. – Ethernet Type – IP Protocol No. – UDP destination port No. – Destination MAC address (Unicast) – Destination MAC address (Broadcast) – Destination MAC address (Multicast) – ARP Request of Local Station – Neighbor Solicitation of Local Station – Except IPv6 Next Header (Analyzable) (3) Automatically Responding to ARP Request and Neighbor Solicitation

Table 31.1 Specifications (Functions) (2/2)

IP Name	1. Function
DMAC	(1) Supports AXI for DMA (128bit data width) (2) Supports APB for register access (32bit data width) (3) Direct Memory Access between URAM and TOE/E-MAC by Descriptor Method (4) Supported Maximum Frame Size are followings. – Transmission Frame is 1.5 Kbytes – Reception Frame is 8 Kbytes

31.1.2 Block Diagram

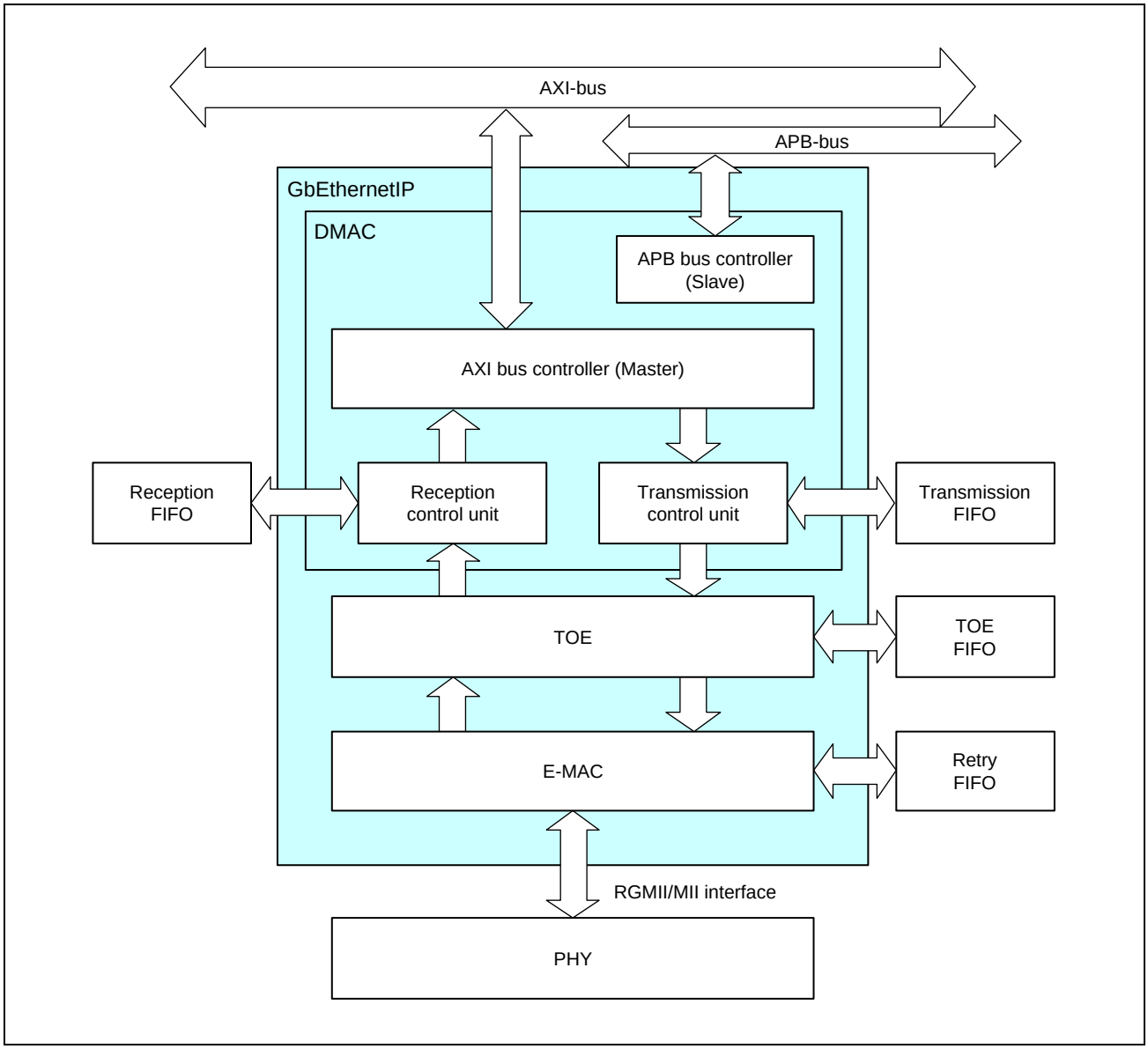


Figure 31.1 Block Diagram of Gigabit Ethernet Interface

31.2 External signal pins

Table 31.2 shows external signal pins for Gigabit Ethernet Interface.

Table 31.2 External Pin Interface

Name	I/O	Description	Active Level	Initial Value	Handling in Not Used
ET0_TXC/TX_CLK	I	Ch0 transmit clock signal	—	—	Pull-Down
ET0_TX_CTL/TX_EN	O	Ch0 Transmit control/enable signal	H	0b	Open
ET0_TXD[3-0]	O	Ch0 Transmit data signal	—	0b	Open
ET0_TX_ERR	O	Ch0 Transmit error signal	H	0b	Open
ET0_TX_COL	I	Ch0 Transmit collision signal	H	—	Pull-Down
ET0_TX_CRS	I	Ch0 Transmit carrier sense signal	H	—	Pull-Down
ET0_RXC/RX_CLK	I	Ch0 Receive clock signal	—	—	Pull-Down
ET0_RX_CTL/RX_DV	I	Ch0 Receive data Control/valid signal	H	—	Pull-Down
ET0_RXD[3-0]	I	Ch0 Receive data signal	—	—	Pull-Down
ET0_RX_ERR	I	Ch0 Receive error signal	H	—	Pull-Down
ET0_MDC	O	Ch0 Management information transfer clock signal	—	0b	Open
ET0_MDIO	I/O	Ch0 Management information transmit/receive data	—	Open	Pull-Down
ET0_LINKSTA	I	Ch0 PHY Link status signal	H	—	Pull-Down
ET1_TXC/TX_CLK	I	Ch1 transmit clock signal	—	—	Pull-Down
ET1_TX_CTL/TX_EN	O	Ch1 Transmit control/enable signal	H	0b	Open
ET1_TXD[3-0]	O	Ch1 Transmit data signal	—	0b	Open
ET1_TX_ERR	O	Ch1 Transmit error signal	H	0b	Open
ET1_TX_COL	I	Ch1 Transmit collision signal	H	—	Pull-Down
ET1_TX_CRS	I	Ch1 Transmit carrier sense signal	H	—	Pull-Down
ET1_RXC/RX_CLK	I	Ch1 Receive clock signal	—	—	Pull-Down
ET1_RX_CTL/RX_DV	I	Ch1 Receive data Control/valid signal	H	—	Pull-Down
ET1_RXD[3-0]	I	Ch1 Receive data signal	—	—	Pull-Down
ET1_RX_ERR	I	Ch1 Receive error signal	H	—	Pull-Down
ET1_MDC	O	Ch1 Management information transfer clock signal	—	0b	Open
ET1_MDIO	I/O	Ch1 Management information transmit/receive data	—	Open	Pull-Down
ET1_LINKSTA	I	Ch1 PHY Link status signal	H	—	Pull-Down

31.3 Register Configuration

31.3.1 Register Base Address

The base addresses of each channel are listed in the following table.

Table 31.3 Register Base Address

Register Name	Base Address Name	Base Address
ETH0	<ETH0_CA55_base>	H'0_11C3_0000
	<ETH0_CM33/CM33_FPU_S_base>	H'41C3_0000
	<ETH0_CM33/CM33_FPU_NS_base>	H'51C3_0000
ETH1	<ETH1_CA55_base>	H'0_11C4_0000
	<ETH1_CM33/CM33_FPU_S_base>	H'41C4_0000
	<ETH1_CM33/CM33_FPU_NS_base>	H'51C4_0000

Note: Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

31.3.2 IO Mode Registers

Refer to **Section 45, General Purpose Input Output Port (GPIO)** for Gigabit Ethernet Interface IO mode configuration.

31.3.3 DMAC Registers

Table 31.4 Configuration of DMAC-related Registers (From H'000 to H'4FF) (1/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
DMAC mode register	CCC	R/W	H'0000_0000	H'000	32
Descriptor base address table register	DBAT	R/W	H'0000_0000	H'004	32
Descriptor base address load request register	DLR	R/W	H'003F_FFFF	H'008	32
DMAC status register	CSR	R	H'0000_0001	H'00C	32
Current descriptor address register q (q = 0, 4)	CDARq	R	H'0000_0000	H'010 + q × 4	32
Error status register	ESR	R	H'0000_0000	H'088	32
Receive configuration register	RCR	R/W	H'6000_0000	H'090	32
Reception Truncation Configuration register	RTC	R/W	H'7FFC_7FFC	H'0B4	32
Transmit configuration register	TGC	R/W	H'0022_2200	H'300	32
Transmit configuration control register	TCCR	R/W	H'0000_0000	H'304	32
Transmit status register	TSR	R	H'0000_0000	H'308	32
MAC status FIFO Access register	MFA	R	H'0000_0000	H'30C	32
MAC status FIFO Access 2 register	MFA2	R	H'0000_0000	H'340	32
Descriptor interrupt control register	DIC	R/W	H'0000_0000	H'350	32
Descriptor interrupt status register	DIS	R/W	H'0000_0000	H'354	32
Error interrupt control register	EIC	R/W	H'0000_0000	H'358	32
Error interrupt status register	EIS	R/W	H'0000_0000	H'35C	32
Receive interrupt control register 0	RIC0	R/W	H'0000_0000	H'360	32
Receive interrupt status register 0	RIS0	R/W	H'0000_0000	H'364	32
Receive interrupt control register 1	RIC1	R/W	H'0000_0000	H'368	32
Receive interrupt status register 1	RIS1	R/W	H'0000_0000	H'36C	32
Receive interrupt control register 2	RIC2	R/W	H'0000_0000	H'370	32
Receive interrupt status register 2	RIS2	R/W	H'0000_0000	H'374	32
Transmit interrupt control register	TIC	R/W	H'0000_0000	H'378	32
Transmit interrupt status register	TIS	R/W	H'0000_0000	H'37C	32
Interrupt summary status register	ISS	R	H'0000_0000	H'380	32
Common Interrupt Enable register	CIE	R/W	H'0000_0000	H'384	32
Receive Interrupt Control register 3	RIC3	R/W	H'0000_0000	H'388	32
Receive Interrupt Status register 3	RIS3	R/W	H'0000_0000	H'38C	32
Descriptor Interrupt Enable register	DIE	R/W	H'0000_0000	H'450	32
Descriptor Interrupt Disable register	DID	R/W	H'0000_0000	H'454	32
Error Interrupt Enable register	EIE	R/W	H'0000_0000	H'458	32
Error Interrupt Disable register	EID	R/W	H'0000_0000	H'45C	32
Receive Interrupt Enable register 0	RIE0	R/W	H'0000_0000	H'460	32
Receive Interrupt Disable register 0	RID0	R/W	H'0000_0000	H'464	32
Receive Interrupt Enable register 1	RIE1	R/W	H'0000_0000	H'468	32
Receive Interrupt Disable register 1	RID1	R/W	H'0000_0000	H'46C	32
Receive Interrupt Enable register 2	RIE2	R/W	H'0000_0000	H'470	32
Receive Interrupt Disable register 2	RID2	R/W	H'0000_0000	H'474	32
Transmit Interrupt Enable register	TIE	R/W	H'0000_0000	H'478	32
Transmit Interrupt Disable register	TID	R/W	H'0000_0000	H'47C	32
Receive Interrupt Enable register 3	RIE3	R/W	H'0000_0000	H'488	32
Receive Interrupt Disable register 3	RID3	R/W	H'0000_0000	H'48C	32

31.3.4 E-MAC Registers

Table 31.5 Configuration of E-MAC-related Registers (From H'500 to H'7FF) (1/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
E-MAC operating mode register	CXR20	R/W	H'0000_0000	H'500	32
Maximum receive frame length register	CXR2A	R/W	H'0000_0000	H'508	32
Interrupt status register	CXR21	R/W	H'0000_000X	H'510	32
Enables/disables interrupts register	CXR22	R/W	H'0000_0000	H'518	32
PHY register access register	CXR23	R/W	H'0000_000X	H'520	32
Link status register	CXR2B	R	H'0000_000X	H'528	32
Polarity set register	CXR2C	R/W	H'0000_0000	H'52C	32
In-Band Status set register	CXR31	R/W	H'0000_0000	H'530	32
In-Band Status indicate register	CXR32	R	H'0000_000X	H'534	32
Mode indicate register	CXR33	R	H'0000_0000	H'538	32
PHY interface select register	CXR35	R/W	H'FFFF_0000	H'540	32
PHY interface indicate register	CXR36	R	H'0000_0000	H'544	32
PAUSE frame register 1	CXR71	R/W	H'0000_0000	H'554	32
PAUSE frame register 2	CXR72	R/W	H'0000_0000	H'558	32
PAUSE frame register 3	CXR8A	R/W	H'0000_0000	H'55C	32
PAUSE frame register 4	CXR80	R/W	H'0000_0000	H'560	32
PAUSE frame register 5	CXR81	R/W	H'0000_0000	H'564	32
PAUSE frame register 6	CXR82	R	H'0000_0000	H'568	32
E-MAC operating mode 2 register	CXR2D	R/W	H'0000_0000	H'5B0	32
Software LINK status register	CXR2G	R/W	H'0000_0000	H'5BC	32
Mac Address register 1	CXR24	R/W	H'0000_0000	H'5C0	32
Mac Address register 2	CXR25	R/W	H'0000_0000	H'5C8	32
TINT1 counter register	CXR40	R/W	H'0000_0000	H'700	32
TINT2 counter register	CXR41	R/W	H'0000_0000	H'708	32
TINT3 counter register	CXR42	R/W	H'0000_0000	H'710	32
RINT1 counter register	CXR50	R/W	H'0000_0000	H'740	32
RINT2 counter register	CXR51	R/W	H'0000_0000	H'748	32
RINT3 counter register	CXR52	R/W	H'0000_0000	H'750	32
RINT4 counter register	CXR53	R/W	H'0000_0000	H'758	32
RINT5 counter register	CXR54	R/W	H'0000_0000	H'760	32
RINT6 counter register	CXR55	R/W	H'0000_0000	H'768	32
RINT7 counter register	CXR56	R/W	H'0000_0000	H'770	32
RINT8 counter register	CXR57	R/W	H'0000_0000	H'778	32
MDIO status register	MDIOSTS	R	H'0000_0000	H'780	32
MDIO command register	MDIOCMD	R/W	H'0000_0000	H'784	32
MDIO address register	MDIOADR	R/W	H'0000_0000	H'788	32
MDIO data register	MDIODAT	R/W	H'0000_0000	H'78C	32
MDIO mode register	MDIOMOD	R/W	H'8000_00FF	H'790	32
Low Power Mode register 1	LPTXMOD1	R/W	H'0000_0000	H'7B0	32
Low Power Mode register 2	LPTXMOD2	R/W	H'0000_0000	H'7B4	32
RGMII Low Power parameter register 1	LPTXGTH1	R/W	H'0000_0000	H'7C0	32
RGMII Low Power parameter register 2	LPTXGTH2	R/W	H'0000_0000	H'7C4	32
RGMII Low Power parameter register 3	LPTXGTH3	R/W	H'0000_0000	H'7C8	32

Table 31.5 Configuration of E-MAC-related Registers (From H'500 to H'7FF) (2/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
RGMII Low Power parameter register 4	LPTXGTH4	R/W	H'0000_0000	H'7CC	32
MII Low Power parameter register 1	LPTXMTH1	R/W	H'0000_0000	H'7D0	32
MII Low Power parameter register 2	LPTXMTH2	R/W	H'0000_0000	H'7D4	32
MII Low Power parameter register 3	LPTXMTH3	R/W	H'0000_0000	H'7D8	32
MII Low Power parameter register 4	LPTXMTH4	R/W	H'0000_0000	H'7DC	32

31.3.5 TOE Registers

Table 31.6 Configuration of TOE-related Registers (From H'800 to H'9FF) (1/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
Checksum operating mode register	CSR0	R/W	H'0000_0000	H'800	32
Tx Frame Checksum Enable register	CSR1	R/W	H'0000_0000	H'804	32
Rx Frame Checksum Enable register	CSR2	R/W	H'0000_0000	H'808	32
Tx Extended Header Number register	CSR3	R/W	H'0000_0008	H'80C	32
Rx Extended Header Number register	CSR4	R/W	H'0000_0008	H'810	32
Unsupported Tx Frame Counter register	CSR20	R/W	H'0000_0000	H'820	32
Unsupported Rx Frame Counter register	CSR30	R/W	H'0000_0000	H'824	32
Rx IPv4 Header checksum Error Counter register	CSR31	R/W	H'0000_0000	H'828	32
Rx TCP/UDP/ICMP checksum Error Counter register	CSR32	R/W	H'0000_0000	H'82C	32
Filter Operation/Auto Response Enable register	CSFR00	R/W	H'1020_0000	H'840	32
Local IPv4 address register	CSFR01	R/W	H'0000_0000	H'844	32
Local IPv6 address register i (i = 0 to 3)	CSFR02_i	R/W	H'0000_0000	H'848 + 4 × i	32
Upper Local MAC address register	CSFR03_U	R/W	H'0000_0000	H'858	32
Lower Local MAC address register	CSFR03_L	R/W	H'0000_0000	H'85C	32
IPv6 no_next_header Protocol Number register	CSFR04	R/W	H'0000_003B	H'860	32
Ethernet Type Condition register i (i = 0 to 3)	CSFR10_i	R/W	H'0000_0000	H'870 + 4 × i	32
Ethernet Type Condition Enable register	CSFR10	R/W	H'0000_0000	H'880	32
Protocol Condition register i (i = 0 to 3)	CSFR11_i	R/W	H'0000_0000	H'884 + 4 × i	32
Protocol Condition Enable register	CSFR11	R/W	H'0000_0000	H'894	32
UDP Por Condition register i (i = 0 to 11)	CSFR12_i	R/W	H'0000_0000	H'898 + 4 × i	32
UDP Port Condition Enable register	CSFR12	R/W	H'0000_0000	H'8C8	32
Upper MAC DA unicast address Condition register	CSFR13_U	R/W	H'0000_0000	H'8CC	32
Lower MAC DA unicast address Condition register	CSFR13_L	R/W	H'0000_0000	H'8D0	32
Upper MAC DA broadcast address Condition register	CSFR14_U	R/W	H'0000_FFFF	H'8D4	32
Lower MAC DA broadcast address Condition register	CSFR14_L	R/W	H'FFFF_FFFF	H'8D8	32
Upper MAC DA multicast address Condition register i (i = 0 to 19)	CSFR15_U_i	R/W	H'0000_0100	H'8DC + 8 × i	32
Lower MAC DA multicast address Condition register i (i = 0 to 19)	CSFR15_L_i	R/W	H'5E00_0000	H'8E0 + 8 × i	32
MAC DA Condition Enable register	CSFR15	R/W	H'0000_0000	H'97C	32
IPv6 analysis Protocol Condition register 0	CSFR16_0	R/W	H'2911_0600	H'980	32
IPv6 analysis Protocol Condition register 1	CSFR16_1	R/W	H'3A33_2C2B	H'984	32
IPv6 analysis Protocol Condition register 2	CSFR16_2	R/W	H'0000_003C	H'988	32
IPv6 analysis Protocol Condition Enable register	CSFR16	R/W	H'0000_01FF	H'98C	32
Wake Up Interrupt Status register	CSFR20	R/W	H'0000_0000	H'9A0	32
Wake Up Interrupt Mask register	CSFR21	R/W	H'101F_FFFF	H'9A4	32
Auto Response Configuration register1	CSFR30	R/W	H'0001_7B35	H'9B0	32
Auto Response Configuration register2	CSFR31	R/W	H'0000_0000	H'9B4	32
ARPREQ/Neighbor Solicitations Receive Interrupt Status register	CSFR40	R/W	H'0000_0000	H'9C0	32

Table 31.6 Configuration of TOE-related Registers (From H’800 to H’9FF) (2/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
ARPREQ/Neighbor Solicitations Receive Interrupt Mask register	CSFR41	R/W	H’0000_0007	H’9C4	32
Rx ARPREQ/Neighbor Solicitation Upper MAC SA register	CSFR42_U	R	H’0000_0000	H’9C8	32
Rx ARPREQ/Neighbor Solicitation Lower MAC SA register	CSFR42_L	R	H’0000_0000	H’9CC	32
Rx ARPREQ/Neighbor Solicitation IP SA register i (i = 0 to 3)	CSFR43_i	R	H’0000_0000	H’9D0 + 4 × i	32

31.4 Register Descriptions

31.4.1 DMAC Registers

31.4.1.1 DMAC Mode Register (CCC)

The CCC register specifies the operating mode of the DMAC.

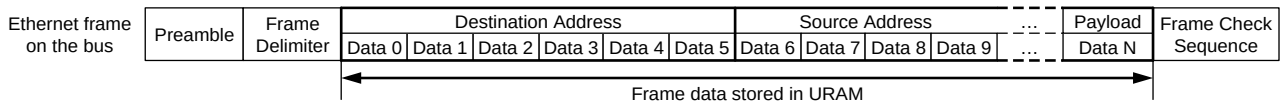
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	FCE	—	—	—	—	BOC	—	ERCS	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R	R	R	R	R/W	R	R/W	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RDFD	DTSR	—	—	—	—	—	—	OPC[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
25	FCE	0b	R/W	Flow Control Enable 0b Flow control disabled 1b Flow control enabled This bit enables the flow control support of E-MAC. When flow control is enabled, the E-MAC gets informed about the Reception FIFO level (Reception FIFO fill level reached RCR.RFCL).
24 to 21	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
20	BOC	0b	R/W	Byte Order Configuration 0b First Ethernet byte in URAM [7:0] This bit defines the byte ordering of frame data in URAM in relation to the byte order in the Ethernet frame. The CPU can only write 0b to this bit.*1
19	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	ERCS	0b	R/W	Enable Reference 250 MHz Clock Stop function 0b Disable reference 250 MHz clock(clk_miitx_gtx_refclk) stop function. 1b Enable reference 250 MHz clock(clk_miitx_gtx_refclk) stop function. This bit enables the stop function of reference clock of Ethernet. When this bit is '1', Gigabit Ethernet Interface asserts Output Signal “refclk_stop_en” during it does not need reference clock. *Note*: If PHY does not support stopping TXC of RGMII Interface, set “0” to this bit. Because TXC is generated from clk_miitx_gtx_refclk. [Changing condition] This bit is set to 0b when RESET mode is entered.
17 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
9	RDFD	0b	R/W	Reception Descriptor Fetch Disable 0b Reception Descriptor Fetch Enable 1b Reception Descriptor Fetch Disable This register is used to stop the fetch of reception descriptor. By fetch stops, the state of Reception Descriptor Controller of DMAC will stop at IDLE. CSR.RDFDM indicates the current fetch status of reception descriptor. The CPU should not change any configuration or mode until CSR.RDFDM is updated once the Reception Descriptor Fetch Disable request is given. The CPU should not cancel the Reception Descriptor Fetch Disable request (set this bit to 0b) before CSR.RDFDM is read as 1b.
8	DTSR	0b	R/W	Data Transfer Suspend Request 0b Normal operation 1b Request suspension of data transfer This bit requests to suspend further Gigabit Ethernet Interface accesses to URAM. A currently on-going access is completed. CSR.DTS indicates the current suspend status. The CPU should not change any configuration or mode until CSR.DTS is updated once data transfer suspend request is given. The CPU should not cancel the data transfer suspend request (set this bit to 0b) before CSR.DTS is read as 1b.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1 to 0	OPC	00b	R/W	OPeration Command 00b Enter RESET mode 01b Enter CONFIG mode 10b Enter OPERATION mode 11b Reserved These bits define the operating mode DMAC should enter. Refer to **Section 31.5.1, DMAC Operating Modes** for details about the operating modes. The CPU can write to these bits in any operating mode. [Changing condition] These bits are set to 00b when RESET mode is entered.

Note 1. Following illustrates a received Ethernet frame and the part of the frame stored in URAM.

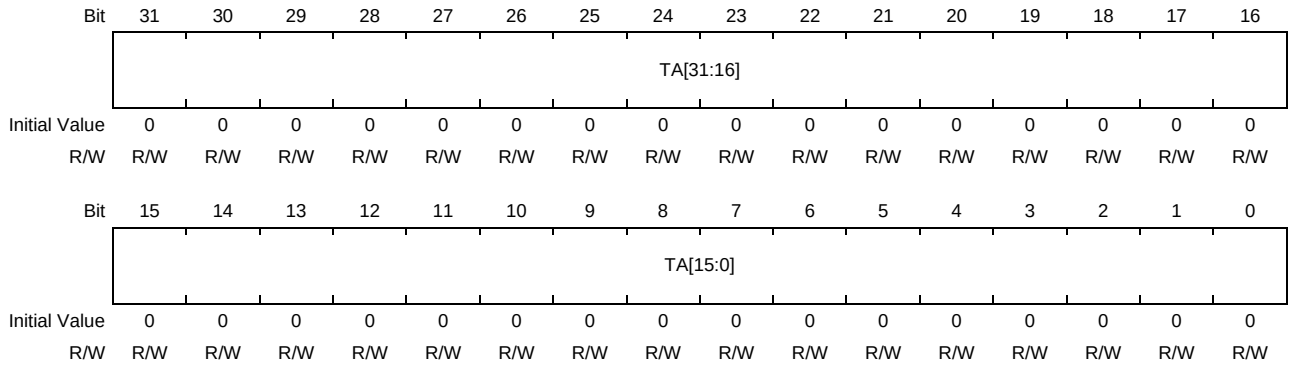


- When CCC.BOC is 0b, an Ethernet frame is represented in URAM as:

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPTR + 0	Data 3								Data 2								Data 1								Data 0							
DPTR + 4	Data 7								Data 6								Data 5								Data 4							
DPTR + 8	Data 11								Data 10								Data 9								Data 8							

31.4.1.2 Descriptor Base Address Table Register (DBAT)

The DBAT register specifies the base address of the descriptor table in the URAM.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TA[31:0]	H'0000_0000	R/W	Table Address Address of descriptor base address table in URAM These bits define the address of the first entry of the descriptor base address table. Refer to **Section 31.5.2.3 Descriptor Base Address Table** about function and structure of the descriptor base address table. The CPU can only write to these bits if CSR.OPS is CONFIG and OPERATION. The CPU should only write values which are a multiple of 4 to these bits.

31.4.1.3 Descriptor Base Address Load Request Register (DLR)

The DLR register is used to issue a request to load the values from the count descriptor address register q (CDARq) for each queue to the descriptor base address table register (DBAT).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	LBA4	—	—	—	LBA0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21 to 5	—	All 1	R	Reserved Whenever it is read, all bits 1 are read. The written value will be ignored.
4	LBA4	1b	R/W	Load Base Address 4(Rx0) 0b No pending load request 1b Pending load request This bit requests to set the current descriptor address of queue q (CDARq.CDA) to an entry of the descriptor base address table (DBAT.TA + 8 × 4). By setting this bit to 1b CPU requests a load of the current descriptor address of queue q. If there is an on-going transfer for queue q, the load is processed after the whole frame has been transferred. Note: In case of a transmission queue, the load function is not influencing the transmit start request (TCCR.TSRQ); so, fetching continues at base address. To prevent this behavior, it is recommended to request the load of transmission queues only when TCCR.TSRQ is 0b. The CPU can only write to this bit if CSR.OPS is OPERATION. The CPU can only write 1b to this bit. [Changing condition] This bit is set to 1b when leaving OPERATION mode. This bit is set to 0b when there is no on-going transfer (related to old chain) for queue q. Note for verification: In Rx, if the entirety or the final part of a frame is stored into a FEMPTY_ND descriptor, then there is no actual frame transfer to URAM but still LBA reload is processed only when the transfer from RxRAM to BMI is completed for that frame.
3 to 1	—	All 1	R	Reserved Whenever it is read, all bits 1 are read. The written value will be ignored.
0	LBA0	1b	R/W	Load Base Address 0(Tx0) 0b No pending load request 1b Pending load request This bit requests to set the current descriptor address of queue q (CDARq.CDA) to an entry of the descriptor base address table (DBAT.TA + 8 × 0). For the other description, refer to LBA4 bit of this register.

31.4.1.4 DMAC Status Register (CSR)

The CSR register is used to indicate the operating mode in which the DMAC is running and the individual communications states.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	TDUO	RPO	—	—	—	TPO
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RCSI	RDFDM	DTS	—	—	—	—	OPS[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	TDUO	0b	R	Transmission Descriptor Update On-going 0b No pending descriptor update 1b Pending descriptor update in URAM This bit indicates that there is pending descriptor update for a transmit queue. When this bit is 1b, there are descriptors of already or currently processed transmit storage elements not updated in URAM. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when all transmit queue related storage elements processed. This bit is set to 1b when descriptor related status update procedure starts (descriptor update and SFR flagging).
20	RPO	0b	R	Reception Process On-going 0b No pending reception 1b Pending received frames in Reception FIFO This bit indicates that there is pending receive data for a receive queues. When this bit is 1b, there are frames in Reception FIFO that are not completely stored in URAM. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when a frame is completely stored in URAM, the descriptor update has been completed, and there are no pending frames in Reception FIFO for storage to URAM. This bit is set to 1b when a frame received by E-MAC is completely stored in Reception FIFO.
19 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	TPO	0b	R	Transmission Process On-going 0b No pending transmission 1b Pending frames for transmission This bit indicates that there is pending or on-going transmission process for transmit queue. When this bit is 1b, a frame fetch from the queue in URAM has been started or completed but this frame has not been completely transmitted by E-MAC Descriptor write back is not included in this bit.
15 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
10	RCSI	0b	R	Reference 250 MHz Clock is being Stopped Internally 0b Reference 250 MHz Clock is being not stopped internally 1b Reference 250 MHz Clock is being stopped internally This bit indicates that Gigabit Ethernet Interface has stopped Reference 250 MHz Clock (clk_miitx_gtx_refclk) internally or not. When this bit is 1b, Internal Gated Clock Cell for Reference 250 MHz Clock is closed. During that time, this clock is not provided to internal logics. [Changing condition] This bit is set to 0b when the condition of input signal “refclk_stop” changes to 0b. This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when CCC.ERCS is 0b. This bit is set to 1b when the condition of output signal “refclk_stop_en” changes to 1b.
9	RDFDM	0b	R	Reception Descriptor Fetch Disable Monitor 0b Reception Descriptor Fetch will happen 1b Reception Descriptor Fetch is stopped This bit is for monitoring. It can be used in order to confirm a state of Reception Descriptor Controller of DMAC. If this bit is '1', it means the fetch of Reception Descriptor from URAM by DMAC is completely stopped. [Changing condition] This bit is set to 0b when CCC.RDFD is 0b. This bit is set to 0b when RESET mode is entered. This bit is set to 1b when CCC.RDFD is 1b and fetch of Reception Descriptor from URAM is completely stopped.
8	DTS	0b	R	Data Transfer Suspended 0b Access to URAM will happen 1b Access to URAM is suspended This bit indicates if DMAC is allowed to access URAM. [Changing condition] This bit is set to 0b when CCC.DTSR is 0b. This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when CCC.DTSR is 1b and there is no on-going URAM access.
7 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	OPS	0001b	R	OPerating mode Status 0001b RESET 0010b CONFIG 0100b OPERATION others reserved These bits represent the current operating mode of DMAC. [Changing condition] These bits are updated when an operating mode changes is processed.

31.4.1.5 Current Descriptor Address Register q (CDARq) (q = 0,4)

The CDARq register indicates the current descriptor address.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CDA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CDA	All 0	R	Current Descriptor Address Current descriptor address of transmission / receive queue q These bits indicate the address of the descriptor which is currently being processed by queue q or which will be processed when queue q gets active. For q = 0 the addresses refer to the transmit queues. From q = 4 the addresses refer to the receive queues. *Note*: The current descriptor address cannot be used for HW/SW synchronization. CDARq.CDA may one or more descriptors ahead from information available in URAM. [Changing condition] These bits are set to 0 when leaving OPERATION mode. These bits are set to DBAT.TA + 8 × q when entering OPERATION mode. These bits are set to DBAT.TA + 8 × q when the load base address request of queue (DLR.LBA) has been processed. These bits are set to DESCR.DPTR when a link descriptor (LINK, LINKFIX) has been processed. These bits are incremented by the size of the descriptor control structure (8 for normal) when a frame data descriptor has been processed.

31.4.1.6 Error Status Register (ESR)

Error Status Register

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

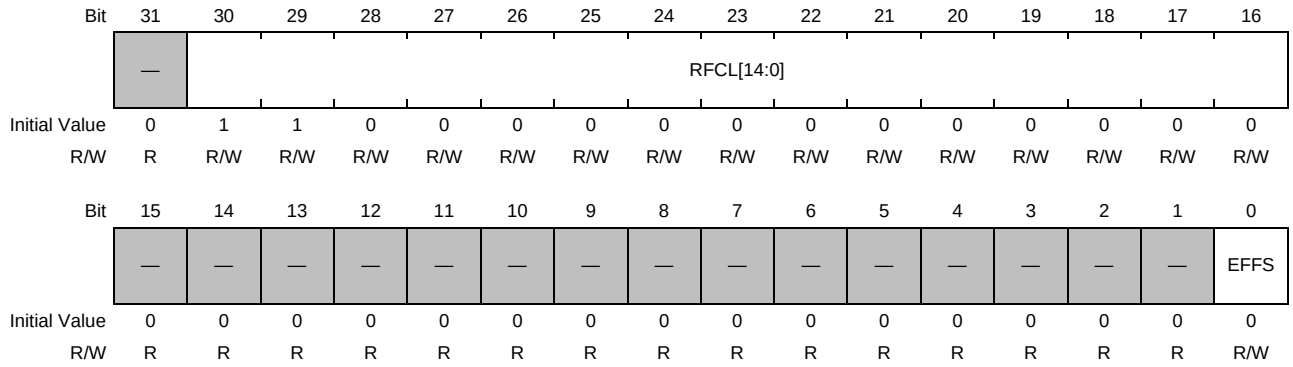
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	EIL	ET[3:0]				—	—	—	EQN[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	EIL	0b	R	Error Information Lost 0b No loss of error information 1b Lost of error information detected This bit indicates that error information detected by Gigabit Ethernet Interface is lost because the previous reported error has not been processed by CPU. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when CPU writes 0b to EIS.QEF. This bit is set to 1b when the set condition of EIS.QEF is fulfilled while EIS.QEF is 1b.
11 to 8	ET	0000b	R	Error Type 0000b (0) Read descriptor from URAM 0001b (1) Write descriptor to URAM 0010b (2) Interpret read descriptor 0011b (3) Transmission FIFO is corrupted 0100b (4) Read data from URAM 0101b (5) Write data to URAM 0110b (6) Reading data from Reception FIFO 0111b (7) Reception FIFO is corrupted 1000b (8) Frame size error during reception detected 1001b (9) Reserved 1010b (10) Transmission FIFO Data access error others reserved These bits indicate details about the transfer stage which was handled when Gigabit Ethernet Interface has detected an error. When the fault is related to the read descriptor (ESR.ET=0 or 2), CPU needs to correct the faulty descriptor before the related queue can continue processing. In this case the queue halts at the faulty descriptor and CDARq.CDA (with q=ESR.EQN) identifies faulty descriptor. All other errors are transient in nature and may be corrected by continuation of HW or SW operation; so there is no strong demand on CPU interaction. The CPU should only evaluate these bits when EIS.QEF is 1b. [Changing condition] These bits are updated when the set condition of EIS.QEF is fulfilled and EIS.QEF is 0b.
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
4 to 0	EQN	00000b	R	Number of faulty queue These bits indicate the queue number which was being handled when Gigabit Ethernet Interface has detected an error. A fault reported for ESR.EQN = 0 is related to transmit queue. From ESR.EQN = 4 the fault is related to receive queue. The CPU should only evaluate these bits when EIS.QEF is 1b. The CPU should not evaluate these bits when ESR.ET is 0011b, 0111b. [Changing condition] These bits are updated when the set condition of EIS.QEF is fulfilled and EIS.QEF is 0b.

31.4.1.7 Receive Configuration Register (RCR)

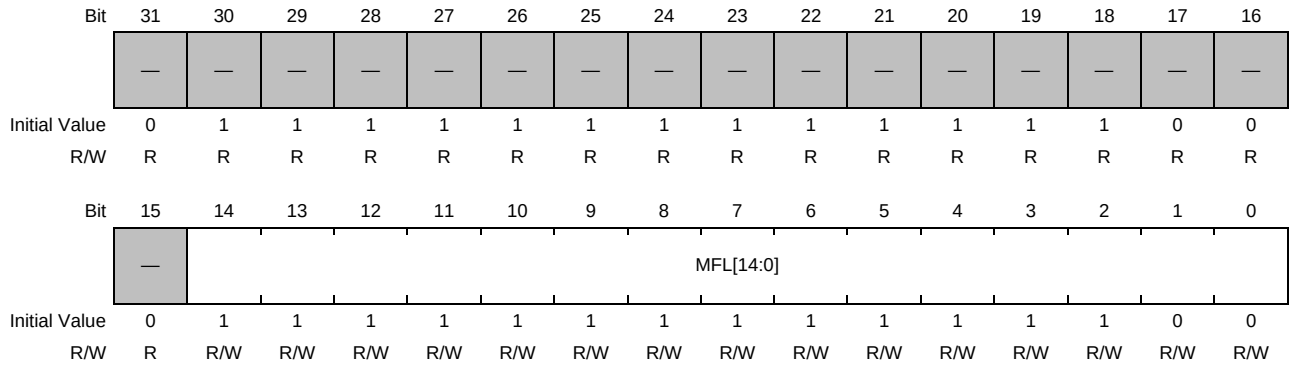
The RCR register is used to make settings related to reception for the DMAC.



Bit	Bit Name	Initial Value	R/W	Description
31	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 16	RFCL[14:0]	H'6000	R/W	Reception FIFO Critical Level Number of bytes stored in Reception FIFO before critical level is reached (RFCL[1:0] are fixed to 0). These bits define a fill level of the Reception FIFO in bytes to generate FIFO critical level reached warning notification (RIS1.RFWF). The CPU can only write to these bits if CSR.OPS is CONFIG. The CPU should only write values which are a multiple of 4 to these bits.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	EFFS	0b	R/W	Enable Faulty Frame Storage 0b Storage of faulty frames disabled 1b Storage of faulty frames enabled (for debug) This bit enables the storage of received frames classified as faulty by the TOE or E-MAC. Faulty received frames will always be stored in receive queue. *Note*: DESCR.MSC informs about the reception fault detected by the TOE or E-MAC. The CPU can only write to this bit if CSR.OPS is CONFIG. *Note*: This bit is used only for debug. Do not set RCR.EFFS=1 during normal operation.

31.4.1.8 Reception Truncation Configuration Register I (RTC)

Reception Truncation Configuration register



Bit	Bit Name	Initial Value	R/W	Description
31	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 18	—	All 1	R	Reserved Whenever it is read, all bits 1 are read. The written value will be ignored.
17 to 15	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14 to 0	MFL	H'7FFC	R/W	Maximum Frame Length These bits define the maximum frame length (64 to 8k bytes) stored to URAM. When a frame received by E-MAC is truncated, CPU gets informed by EIS.QEF and frame size error (ESR.ET is 1000b). Additionally, the DESCR.TI in the updated descriptor is set to 1b. The CPU can only write to these bits if CSR.OPS is CONFIG. The CPU should only write values which are a multiple of 4 to these bits. The CPU should not write values less than 64 to these bits.

31.4.1.9 Transmit Configuration Register (TGC)

The TGC register is used to make settings related to transmission for the DMAC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TBD[1:0]		—	—	—	—	—	—	—	—
Initial Value	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	—	1b	R	Reserved Whenever it is read, 1 is read. The written value will be ignored.
20 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	—	1b	R	Reserved Whenever it is read, 1 is read. The written value will be ignored.
16 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	—	1b	R	Reserved Whenever it is read, 1 is read. The written value will be ignored.
12 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9 to 8	TBD	10b	R/W	Transmission FIFO Depth This bit defines the depth of the Transmission FIFO used for transmit queue. Gigabit Ethernet Interface fetches up to TGC.TBD frames from URAM to Transmission FIFO for transmit queue. The CPU can only write “10b” to this bit. Frame transmission by E-MAC can start for transmission queue when at least one frame from transmit queue is available in Transmission FIFO. The CPU can only write to these bits if CSR.OPS is CONFIG.
7 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

31.4.1.10 Transmit Configuration Control Register (TCCR)

The TCCR register controls transmission by the DMAC and is used to make related settings.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	MFR2	MFEN2	MFR	MFEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

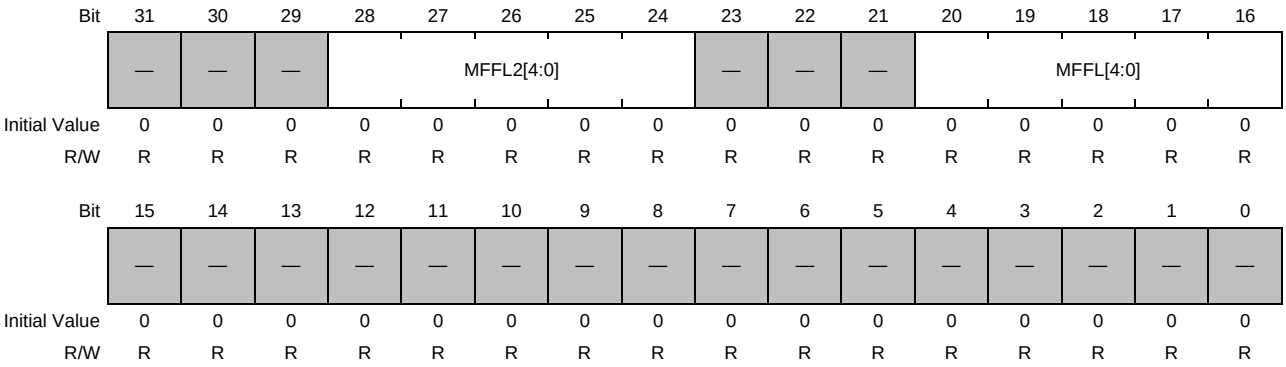
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TSRQ
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	MFR2	0b	R/W	MAC status FIFO2 Release 0b No requests to MAC status FIFO2 1b Release oldest entry of MAC status FIFO2 This bit releases the oldest entry of the MAC status FIFO2. When 1b is written to this bit, Gigabit Ethernet Interface gets informed that CPU has processed the oldest MAC status FIFO2 entry as visible in MFA2 register. The oldest entry is removed from MAC status FIFO2. This bit is always read as 0b. *Note*: Do not use the function of this bit in the Duplex Mode (CXR20.DPM=1).
18	MFEN2	0b	R/W	MAC status FIFO2 Enable 0b Disabled 1b Enabled This bit enables the storage of transmission status information (the E-MAC flags for frame transmission) in the MAC status FIFO2. When this bit is set to 0b all pending FIFO entries are removed. *Note*: Do not use the function of this bit in the Duplex Mode (CXR20.DPM=1).
17	MFR	0b	R/W	MAC status FIFO Release 0b No request to MAC status FIFO 1b Release oldest entry of MAC status FIFO This bit releases the oldest entry of the MAC status FIFO. When 1b is written to this bit, DMAC gets informed that CPU has processed the oldest MAC status FIFO entry as visible in MFA register. The oldest entry is removed from MAC status FIFO. This bit is always read as 0b.
16	MFEN	0b	R/W	MAC status FIFO Enable 0b Disabled 1b Enabled This bit enables the storage of transmission status information (the TOE flags for frame transmission) in the MAC status FIFO. Additionally, CPU can control the status storage for each frame provided for transmission using DESCR.MSR. When this bit is set to 0b all pending FIFO entries are removed.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	TSRQ	0b	R/W	Transmission Start Request Queue 0b Transmission queue empty or stopped 1b Pending frames in queue to fetch This bit defines that frames, provided by CPU in transmit queue in URAM should be transmitted. When this bit is 1b, there are pending frames in transmit queue which have not been fetched to the Transmission FIFO. *Note*: The transmission start request cannot be used for HW/SW synchronization. TCCR.TSRQ may be cleared before frame data is completely fetched from URAM. *Note*: The frame transmission by E-MAC is decoupled from the fetching to the Transmission FIFO. Transmission scheduling for a queue depends on the transmission priority. CSR.TPO gives an indication about pending transmissions. The CPU can only write to this bit if CSR.OPS is OPERATION. The CPU can only write 1b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when CPU writes 1b to it. This bit is set to 0b when DMAC has processed an EEMPTY, FEMPTY or LEMPTY descriptor (no data available). This bit is set to 0b when DMAC has processed an EOS descriptor (end of set). This bit is set to 0b when DMAC has processed a faulty descriptor. This bit is set to 0b when CCC.OPC is 01b and there is no on-going fetch transfer for queue.

31.4.1.11 Transmit Status Register (TSR)

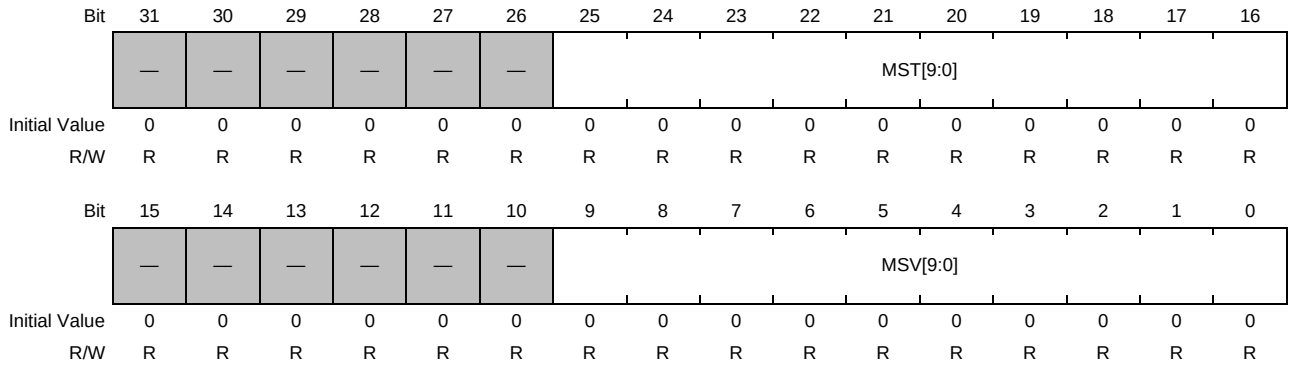
The TSR register indicates the state of transmission by the DMAC.



Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
28 to 24	MFFL2	00000b	R	MAC status FIFO2 Fill Level Number of entries stored in the MAC status FIFO2 These bits indicate the number of entries available in the MAC status FIFO2. If this value is 0, the FIFO2 is empty. If this value is 16, the FIFO2 is full. The values 17 to 31 are reserved. [Changing condition] These bits are set to 0 when leaving OPERATION mode. These bits are set to 0 when TCCR.MFEN2 is 0b. This value is incremented when TCCR.MFEN2 is 1b, TSR.MFFL2 is not 16, and E-MAC detects an error during transmission of frame. This value is decremented when 1b is written to TCCR.MFR2 and TSR.MFFL2 is not 0.
23 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20 to 16	MFFL	00000b	R	MAC status FIFO Fill Level Number of entries stored in the MAC status FIFO These bits indicate the number of entries available in the MAC status FIFO. If this value is 0, the FIFO is empty. If this value is 16, the FIFO is full. The values 17 to 31 are reserved. [Changing condition] These bits are set to 0 when leaving OPERATION mode. These bits are set to 0 when TCCR.MFEN is 0b. This value is incremented when frame with DESCR.MSR has been transmitted to TOE, TCCR.MFEN is 1b and TSR.MFFL is not 16. This value is decremented when 1b is written to TCCR.MFR and TSR.MFFL is not 0.
15 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

31.4.1.12 MAC status FIFO Access Register (MFA)

MAC status FIFO Access register



Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25 to 16	MST	H'000	R	MAC Status Tag Tag number from descriptor identifying the frame MAC status relation These bits represent the DESCR.TAG bits from the descriptor defining the data for frame transmission. The tag is used to get the relation between the frame inside a transmit queue and the TOE status available in the FIFO (MFA.MSV). The CPU should not read these bits when TSR.MFFL is 0. [Changing condition] These bits are updated when the first entry is stored in FIFO (TSR.MFFL changes from 0 to 1). These bits are updated when the oldest entry is released (writing 1b to TCCR.MFR). These bits are updated when the FIFO is overwritten (a frame with DESCR.MSR in descriptor has been transmitted to TOE, TCCR.MFEN is 1b and TSR.MFFL is 16). Note for verification: These bits are set to 0 when leaving OPERATION mode.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9 to 0	MSV	H'000	R	MAC status value These bits represent the TOE status bits as stored in the oldest MAC status FIFO entry. MSV [9] Unsupported transmission frame MSV [8:0] Reserved The CPU should not read these bits when TSR.MFFL is 0. [Changing condition] These bits are updated when the first entry is stored in FIFO (TSR.MFFL changes from 0 to 1). These bits are updated when the oldest entry is released (writing 1b to TCCR.MFR). These bits are updated when the FIFO overwritten (a frame with DESCR.MSR in descriptor has been transmitted to TOE, TCCR.MFEN is 1b and TSR.MFFL is 16). Note for verification: These bits are set to 0 when leaving OPERATION mode.

31.4.1.13 MAC status FIFO Access 2 Register (MFA2)

MAC status FIFO Access 2 register

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MSV2[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8 to 0	MSV2	H'000	R	MAC Status Value2 These bits represent the MAC status bits as stored in the oldest MAC status FIFO2 entry. MSV [8] abort information MSV [7:3] Reserved MSV [2] Carrier lost during frame transmission MSV [1] Delay collision MSV [0] Transmission time out The CPU should not read these bits when TSR.MFFL2 is 0. [Changing condition] These bits are updated when the first entry is stored in FIFO2 (TSR.MFFL2 changes from 0 to 1). These bits are updated when the oldest entry is released (writing 1b to TCCR.MFR2). These bits are updated when the FIFO2 is overwritten when TCCR.MFEN2 is 1b, TSR.MFFL2 is 16, and E-MAC detects an error during transmission of frame.

31.4.1.14 Descriptor Interrupt Status Register (DIS)

The DIS register indicates the state of descriptor interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DPF15	DPF14	DPF13	DPF12	DPF11	DPF10	DPF9	DPF8	DPF7	DPF6	DPF5	DPF4	DPF3	DPF2	DPF1	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 1	DPFi	All 0	R/W	Descriptor Processed Flag i (i = 1 to 15) 0b No interrupt pending 1b Descriptor interrupt pending This bit indicates that a descriptor in a reception or transmit queue has been processed where DESCR.DIE is i. *Note*: When DESCR.DIE is 0000b, no descriptor interrupt is generated. *Note*: When DESCR.DIE is 0001b, in addition a queue specific descriptor interrupt is generated. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when a descriptor has been processed where DESCR.DIE is i.
0	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

31.4.1.15 Descriptor Interrupt Control Register (DIC)

The DIC register is used to control descriptor interrupts 1 to 15.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DPE15	DPE14	DPE13	DPE12	DPE11	DPE10	DPE9	DPE8	DPE7	DPE6	DPE5	DPE4	DPE3	DPE2	DPE1	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 1	DPEi	All 0	R/W	Descriptor Processed interrupt Enable i (i = 1 to 15) 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when DIS.DPFi is 1b. [Changing condition] This bit is set to 0b when writing 1b to DID.DPDi. This bit is set to 1b when writing 1b to DIE.DPSi.
0	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

31.4.1.16 Descriptor Interrupt Enable Register (DIE)

The DIE register set to 1 each bits in descriptor interrupt control register (DIC).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DPS15	DPS14	DPS13	DPS12	DPS11	DPS10	DPS9	DPS8	DPS7	DPS6	DPS5	DPS4	DPS3	DPS2	DPS1	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 1	DPSi	All 0	R/W	Descriptor Processed interrupt Set i (i = 1 to 15) 0b No change of DIC.DPEi 1b Set DIC.DPEi to 1b This bit supports interrupt enable. It controls set of DIC.DPEi. This bit is always read as 0b.
0	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

31.4.1.17 Descriptor Interrupt Disable Register (DID)

The DID register set to 0 each bits in descriptor interrupt control register (DIC).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DPD15	DPD14	DPD13	DPD12	DPD11	DPD10	DPD9	DPD8	DPD7	DPD6	DPD5	DPD4	DPD3	DPD2	DPD1	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 1	DPDi	All 0	R/W	Descriptor Processed interrupt Disable i (i = 1 to 15) 0b No change of DIC.DPEi 1b Set DIC.DPEi to 0b This bit supports interrupt enable. It controls set of DIC.DPEi. This bit is always read as 0b.
0	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

31.4.1.18 Error Interrupt Status Register (EIS)

The EIS register indicates the states of DMAC-related error interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	QFS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MFFF2	—	MFFF	—	—	—	—	—	—	QEF	MTEF	MREF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R	R/W	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	QFS	0b	R	Queue Full Summary 0b No interrupt pending 1b Queue full interrupt pending Summary bit to indicate that at least one queue is full. [Changing condition] This bit is set when any matching pair of RIC2.QFE enable and RIS2.QFF flag are both 1b; or when both the RIS2.RFFF flag and RIC2.RFFE enable are 1b.
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	MFFF2	0b	R/W	MAC status FIFO2 Full Flag 0b No interrupt pending 1b MAC status FIFO2 full, oldest MAC status lost MAC status FIFO2 Full Flag This bit indicates that a transmission MAC status is overwritten because the MAC status FIFO2 is full (overwrite condition). The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when TCCR.MFEN2 is 1b, TSR.MFFL2 is 16, and E-MAC detects an error during transmission of frame.
10	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	MFFF	0b	R/W	MAC status FIFO Full Flag 0b No interrupt pending 1b MAC status FIFO full, oldest MAC status lost This bit indicates that a transmission MAC status is overwritten because the MAC status FIFO is full (overwrite condition). The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when frame with DESCR.MSR has been transmitted to TOE, TCCR.MFEN is 1b and TSR.MFFL is 16.
8 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
2	QEF	0b	R/W	Queue Error Flag 0b No interrupt pending 1b Interrupt pending This bit indicates that an error has been detected while processing receive or transmit queue. Detail about the detected error is indicated by ESR. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when an error condition is detected.
1	MTEF	0b	R/W	MAC Transmission Error Flag 0b No interrupt pending 1b MAC has reported an error during transmission This bit indicates that the E-MAC has detected a fault during transmission. For detail, the E-MAC registers have to be checked. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when E-MAC detects an error during frame transmission.
0	MREF	0b	R/W	MAC Reception Error Flag 0b No interrupt pending 1b MAC has reported an error during reception This bit indicates that the E-MAC has detected a fault during reception. For detail, the E-MAC registers have to be checked. *Note*: When the storage of faulty received frames (RCR.EFFS) is enabled, the E-MAC error code is stored in the descriptor (DESCR.MSC). By evaluating this information CPU can identify corrupted frames in URAM. The CPU can only write 0b to this bit.

31.4.1.19 Error Interrupt Control Register (EIC)

The EIC register controls the DMAC-related error interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MFFE2	—	MFFE	—	—	—	—	—	—	QEE	MTEE	MREE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R	R/W	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	MFFE2	0b	R/W	MAC status FIFO2 Full interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when EIS.MFFF2 is 1b. [Changing condition] This bit is set to 0b when writing 1b to EID.MFFD2. This bit is set to 1b when writing 1b to EIE.MFFS2.
10	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	MFFE	0b	R/W	MAC status FIFO Full interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when EIS.MFFF is 1b. [Changing condition] This bit is set to 0b when writing 1b to EID.MFFD. This bit is set to 1b when writing 1b to EIE.MFFS.
8 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	QEE	0b	R/W	Queue Error interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when EIS.QEF is 1b. [Changing condition] This bit is set to 0b when writing 1b to EID.QED. This bit is set to 1b when writing 1b to EIE.QES.
1	MTEE	0b	R/W	MAC Transmission Error interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when EIS.MTEF is 1b. [Changing condition] This bit is set to 0b when writing 1b to EID.MTED. This bit is set to 1b when writing 1b to EIE.MTES.

Bit	Bit Name	Initial Value	R/W	Description
0	MREE	0b	R/W	MAC Reception Error interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when EIS.MREF is 1b. [Changing condition] This bit is set to 0b when writing 1b to EID.MRED. This bit is set to 1b when writing 1b to EIE.MRES.

31.4.1.20 Error Interrupt Enable Register (EIE)

The EIE register set to 1 each bits in error interrupt control register (EIC).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MFFS2	—	MFFS	—	—	—	—	—	—	QES	MTES	MRES
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R	R/W	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	MFFS2	0b	R/W	MAC status FIFO2 Full interrupt Set 0b No change of EIC.MFFE2 1b Set EIC.MFFE2 to 1b This bit supports interrupt enable. It controls set of EIC.MFFE2. This bit is always read as 0b.
10	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	MFFS	0b	R/W	MAC status FIFO Full interrupt Set 0b No change of EIC.MFFE 1b Set EIC.MFFE to 1b This bit supports interrupt enable. It controls set of EIC.MFFE. This bit is always read as 0b.
8 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	QES	0b	R/W	Queue Error interrupt Set 0b No change of EIC.QEE 1b Set EIC.QEE to 1b This bit supports interrupt enable. It controls set of EIC.QEE. This bit is always read as 0b.
1	MTES	0b	R/W	MAC Transmission Error interrupt Set 0b No change of EIC.MTEE 1b Set EIC.MTEE to 1b This bit supports interrupt enable. It controls set of EIC.MTEE. This bit is always read as 0b.
0	MRES	0b	R/W	MAC Reception Error interrupt Set 0b No change of EIC.MREE 1b Set EIC.MREE to 1b This bit supports interrupt enable. It controls set of EIC.MREE. This bit is always read as 0b.

31.4.1.21 Error Interrupt Disable Register (EID)

The EID register set to 0 each bit in error interrupt control register (EIC).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MFFD2	—	MFFD	—	—	—	—	—	—	QED	MTED	MRED
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R	R/W	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	MFFD2	0b	R/W	MAC status FIFO2 Full interrupt Disable 0b No change of EIC.MFFE 1b Set EIC.MFFE to 0b This bit supports interrupt enable. It controls set of EIC.MFFE2. This bit is always read as 0b.
10	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	MFFD	0b	R/W	MAC status FIFO Full interrupt Disable 0b No change of EIC.MFFE 1b Set EIC.MFFE to 0b This bit supports interrupt enable. It controls set of EIC.MFFE. This bit is always read as 0b.
8 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	QED	0b	R/W	Queue Error interrupt Disable 0b No change of EIC.QEE 1b Set EIC.QEE to 0b This bit supports interrupt enable. It controls set of EIC.QEE. This bit is always read as 0b.
1	MTED	0b	R/W	MAC Transmission Error interrupt Disable 0b No change of EIC.MTEE 1b Set EIC.MTEE to 0b This bit supports interrupt enable. It controls set of EIC.MTEE. This bit is always read as 0b.
0	MRED	0b	R/W	MAC Reception Error interrupt Disable 0b No change of EIC.MREE 1b Set EIC.MREE to 0b This bit supports interrupt enable. It controls set of EIC.MREE. This bit is always read as 0b.

31.4.1.22 Receive Interrupt Status Register 0 (RIS0)

The RIS0 register indicates the states of the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FRF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	FRF	0b	R/W	Frame Received Flag 0b No interrupt pending 1b New frame data is available in URAM This bit indicates that in receive queue r a frame is stored, and data is available to be processed by CPU. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when a frame is stored in receive queue.

31.4.1.23 Receive Interrupt Control Register 0 (RIC0)

The RIC0 register controls the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FRE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	FRE	0b	R/W	Frame Received interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when RIS0.FRF is 1b. [Changing condition] This bit is set to 0b when writing 1b to RID0.FRD. This bit is set to 1b when writing 1b to RIE0.FRS.

31.4.1.24 Receive Interrupt Enable Register 0 (RIE0)

The RIE0 register set to 1 each bits in receive interrupt control register (RIC0).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FRS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	FRS	0b	R/W	Frame Received interrupt Set r 0b No change of RIC0.FRE 1b Set RIC0.FRE to 1b This bit supports interrupt enable. It controls set of RIC0.FRE. This bit is always read as 0b.

31.4.1.25 Receive Interrupt Disable Register 0 (RID0)

The RID0 register set to 0 each bits in receive interrupt control register (RIC0).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FRD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	FRD	0b	R/W	Frame Received interrupt Disable 0b No change of RIC0.FRE 1b Set RIC0.FRE to 0b This bit supports interrupt enable. It controls set of RIC0.FRE. This bit is always read as 0b.

31.4.1.26 Receive Interrupt Status Register 1 (RIS1)

The RIS1 register indicates the states of the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFWF	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RFWF	0b	R/W	Reception FIFO Warning Flag 0b No interrupt pending 1b Reception FIFO warning level reached This bit indicates that the Reception FIFO has reached the configured critical warning level. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when a complete frame is received in the Reception FIFO and the fill level goes above the configured critical warning level during reception of this frame (RCR.RFCL).
30 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

31.4.1.27 Receive Interrupt Control Register 1 (RIC1)

The RIC1 register controls the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFWE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RFWE	0b	R/W	Reception FIFO Warning interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when RIS1.RFWF is 1b. [Changing condition] This bit is set to 0b when writing 1b to RID1.RFWD. This bit is set to 1b when writing 1b to RIE1.RFWS.
30 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

31.4.1.28 Receive Interrupt Enable Register 1 (RIE1)

The RIE register set to 1 each bits in error receive control register (RIC1).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFWS	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RFWS	0b	R/W	Reception FIFO Warning interrupt Set 0b No change of RIC1.RFWE 1b Set RIC1.RFWE to 1b This bit supports interrupt enable. It controls set of RIC1.RFWE. This bit is always read as 0b.
30 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

31.4.1.29 Receive Interrupt Disable Register 1 (RID1)

The RID1 register set to 1 each bits in receive interrupt control register (RIC1).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFWD	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RFWD	0b	R/W	Reception FIFO Warning interrupt Disable 0b No change of RIC1.RFWE 1b Set RIC1.RFWE to 0b This bit supports interrupt enable. It controls set of RIC1.RFWE. This bit is always read as 0b.
30 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

31.4.1.30 Receive Interrupt Status Register 2 (RIS2)

The RIS2 register indicates the states of the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFFF	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	QFF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	RFFF	0b	R/W	Reception FIFO Full Flag 0b No interrupt pending 1b Reception FIFO full, received frame data lost This bit indicates that a frame was received by E-MAC which could not be completely stored in the Reception FIFO. The incomplete frame is lost. There is no information available which receive queue would handle this frame. Frames which are marked by E-MAC as faulty after complete reception can also trigger an overrun. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when frame data provided by E-MAC cannot be stored in the Reception FIFO.
30 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	QFF	0b	R/W	Queue Full Flag 0b No interrupt pending 1b Queue full, received frame data lost This bit indicates that there was no sufficient space in receive queue to store completely a received frame. A receive queue is treated as full when no empty descriptor (read descriptor with DESCR.DT = FEMPTY or FEMPTY_ND) is available. *Note*: If there is no empty descriptor left in queue during storage of a split frame, an incomplete frame will be stored in the queue. Such incomplete frame can be identified based on a descriptor sequence error (no FEND). *Note*: Reading descriptor may be one or more storage elements ahead from actual storage in URAM. CDARq.CDA provides information where queue has been stopped. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when there is pending receive data in Reception FIFO for receive queue and there is no space left for storage. This bit is set to 1b when a SW defined stop point (EOS descriptor) reached within a split frame.

31.4.1.31 Receive Interrupt Control Register 2 (RIC2)

The RIC2 register controls the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFFE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	QFE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	RFFE	0b	R/W	Reception FIFO Full interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when RIS2.RFFF is 1b. [Changing condition] This bit is set to 0b when writing 1b to RID2.RFFD. This bit is set to 1b when writing 1b to RIE2.RFFS.
30 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	QFE	0b	R/W	Queue Full interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when RIS2.QFF is 1b. [Changing condition] This bit is set to 0b when writing 1b to RID2.QFD. This bit is set to 1b when writing 1b to RIE2.QFS.

31.4.1.32 Receive Interrupt Enable Register 2 (RIE2)

The RIE2 register set to 1 each bits in receive interrupt control register (RIC2).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFFS	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	QFS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	RFFS	0b	R/W	Reception FIFO Full interrupt Set 0b No change of RIC2.RFFE 1b Set RIC2.RFFE to 1b This bit supports interrupt enable. It controls set of RIC2.RFFE. This bit is always read as 0b.
30 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	QFS	0b	R/W	Queue Full interrupt Set 0b No change of RIC2.QFE 1b Set RIC2.QFE to 1b This bit supports interrupt enable. It controls set of RIC2.QFE. This bit is always read as 0b.

31.4.1.33 Receive Interrupt Disable Register 2 (RID2)

The RID2 register set to 1 each bit in receive interrupt control register (RIC2).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFFD	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	QFD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	RFFD	0b	R/W	Reception FIFO Full interrupt Disable 0b No change of RIC2.RFFE 1b Set RIC2.RFFE to 0b This bit supports interrupt enable. It controls set of RIC2.RFFE. This bit is always read as 0b.
30 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	QFD	0b	R/W	Queue Full interrupt Disable 0b No change of RIC2.QFE 1b Set RIC2.QFE to 0b This bit supports interrupt enable. It controls set of RIC2.QFE. This bit is always read as 0b.

31.4.1.34 Receive Interrupt Status Register 3 (RIS3)

The RIS3 register indicates the states of the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RDPF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RDPF	0b	R/W	Receive Descriptor Processed Flag 0b No interrupt pending 1b Descriptor interrupt pending This bit indicates that a descriptor in reception queue has been processed where DESCR.DIE is 0001b. *Note*: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when a descriptor in reception queue has been processed where DESCR.DIE is 1.

31.4.1.35 Receive Interrupt Control Register 3 (RIC3)

The RIC3 register controls the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RDPE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RDPE	0b	R/W	Receive Descriptor Processed interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when RIS3.RDPF is 1b. [Changing condition] This bit is set to 0b when writing 1b to RID3.RDPD. This bit is set to 1b when writing 1b to RIE3.RDPS.

31.4.1.36 Receive Interrupt Enable Register 3 (RIE3)

The RIE3 register set to 1 each bits in receive interrupt control register (RIC3).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RDPS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RDPS	0b	R/W	Receive Descriptor Processed interrupt Set 0b No change of RIC3.RDPE 1b Set RIC3.RDPE to 1b This bit supports interrupt enable. It controls set of RIC3.RDPE. This bit is always read as 0b.

31.4.1.37 Receive Interrupt Disable Register 3 (RID3)

The RID3 register set to 0 each bits in receive interrupt control register (RIC3).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RDPD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RDPD	0b	R/W	Receive Descriptor Processed interrupt Disable 0b No change of RIC3.RDPE 1b Set RIC3.RDPE to 0b This bit supports interrupt enable. It controls set of RIC3.RDPE. This bit is always read as 0b.

31.4.1.38 Transmit Interrupt Status Register (TIS)

The TIS register indicates the states of the DMAC transmit interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TDPF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RCSRF	MFWF2	MFUF2	MFWF	MFUF	—	—	—	—	—	—	—	—	—	FTF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	TDPF	0b	R/W	Transmit Descriptor Processed Flag 0b No interrupt pending 1b Descriptor interrupt pending This bit indicates that a descriptor in transmit queue has been processed where DESCR.DIE is 0001b. *Note*: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when a descriptor in transmit queue has been processed where DESCR.DIE is 1.
15	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14	RCSRF	0b	R/W	Reference Clock Stop Req Flag 0b No interrupt pending 1b Reference Clock Stop Request occurred This bit indicates that Gigabit Ethernet Interface expects to stop “clk_miitx_gtx_refclk”. Whenever this flag is asserted, CPU has to stop ETH_PLL (PLL to generate “clk_miitx_gtx_refclk”). To stop ETH_PLL, refer to the document of System Controller. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when CCC.ERCS is 0b. This bit is set to 1b when Gigabit Ethernet Interface has expected to stop “clk_miitx_gtx_refclk”.

Bit	Bit Name	Initial Value	R/W	Description
13	MFWF2	0b	R/W	MAC status FIFO2 Warning Flag 0b No interrupt pending 1b Tx Status FIFO2 warning level has been reached This bit indicates that the warning level of the MAC status FIFO2 (12 out of 16 entries) has been reached. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when TCCR.MFEN2 is 0b. This bit is set to 0b when writing 1b to TCCR.MFR2. This bit is set to 1b when TCCR.MFEN2 is 1b, the MAC Status FIFO2 contains already 11 entries (TSR.MFFL2 is 11), and E-MAC detects an error during transmission of frame.
12	MFUF2	0b	R/W	MAC status FIFO2 Updated Flag 0b No interrupt pending 1b Tx Status FIFO2 has been updated This bit indicates that the MAC status FIFO2 has been updated when the E-MAC detects an error during transmission of frame. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when TCCR.MFEN2 is 0b. This bit is set to 0b when writing 1b to TCCR.MFR2. This bit is set to 1b when TCCR.MFEN2 is 1b and E-MAC detects an error during transmission of frame.
11	MFWF	0b	R/W	MAC status FIFO Warning Flag 0b No interrupt pending 1b Tx Status FIFO warning level has been reached This bit indicates that the warning level of the MAC status FIFO (12 out of 16 entries) has been reached. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when TCCR.MFEN is 0b. This bit is set to 0b when writing 1b to TCCR.MFR. This bit is set to 1b when frame with DESCR.MSR has been transmitted to TOE, TCCR.MFEN is 1b and the MAC Status FIFO contains already 11 entries (TSR.MFFL is 11).
10	MFUF	0b	R/W	MAC status FIFO Updated Flag 0b No interrupt pending 1b Tx Status FIFO has been updated This bit indicates that the MAC status FIFO has been updated after the TOE has transmitted a frame. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when TCCR.MFEN is 0b. This bit is set to 0b when writing 1b to TCCR.MFR. This bit is set to 1b when frame with DESCR.MSR has been transmitted to TOE and TCCR.MFEN is 1b.
9 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	FTF	0b	R/W	Frame Transmitted Flag 0b No interrupt pending 1b Frame transmitted by E-MAC This bit indicates that from transmit queue a frame is transmitted from DMAC. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when a frame from transmit queue has been transmitted from DMAC.

31.4.1.39 Transmit Interrupt Control Register (TIC)

The TIC register controls the DMAC transmit interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TDPE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RCSRE	MFWE2	MFUE2	MFWE	MFUE	—	—	—	—	—	—	—	—	—	FTE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	TDPE	0b	R/W	Transmit Descriptor Processed interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when TIS.TDPF is 1b. [Changing condition] This bit is set to 0b when writing 1b to TID.TDPD. This bit is set to 1b when writing 1b to TIE.TDPS.
15	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14	RCSRE	0b	R/W	Reference Clock Stop Req interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when TIS.RCSRF is 1b. [Changing condition] This bit is set to 0b when writing 1b to TID.RCSRD. This bit is set to 1b when writing 1b to TIE.RCSRS.
13	MFWE2	0b	R/W	MAC status FIFO2 Warning interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when TIS.MFWF2 is 1b. [Changing condition] This bit is set to 0b when writing 1b to TID.MFWD2. This bit is set to 1b when writing 1b to TIE.MFWS2.
12	MFUE2	0b	R/W	MAC status FIFO2 Updated interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when TIS.MFUF2 is 1b. [Changing condition] This bit is set to 0b when writing 1b to TID.MFUD2. This bit is set to 1b when writing 1b to TIE.MFUS2.

Bit	Bit Name	Initial Value	R/W	Description
11	MFWE	0b	R/W	MAC status FIFO Warning interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when TIS.MFWF is 1b. [Changing condition] This bit is set to 0b when writing 1b to TID.MFWD. This bit is set to 1b when writing 1b to TIE.MFWS.
10	MFUE	0b	R/W	MAC status FIFO Updated interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when TIS.MFUF is 1b. [Changing condition] This bit is set to 0b when writing 1b to TID.MFUD. This bit is set to 1b when writing 1b to TIE.MFUS.
9 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	FTE	0b	R/W	Frame Transmitted interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when TIS.FTF is 1b. [Changing condition] This bit is set to 0b when writing 1b to TID.FTD. This bit is set to 1b when writing 1b to TIE.FTS.

31.4.1.40 Transmit Interrupt Enable Register (TIE)

The TIE register set to 1 each bits in transmit interrupt control register (TIC).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TDPS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RCSRS	MFWS2	MFUS2	MFWS	MFUS	—	—	—	—	—	—	—	—	—	FTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	TDPS	0b	R/W	Transmit Descriptor Processed interrupt Set 0b No change of TIC.TDPE 1b Set TIC.TDPE to 1b This bit supports interrupt enable. It controls set of TIC.TDPE. This bit is always read as 0b.
15	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14	RCSRS	0b	R/W	Reference Clock Stop Req interrupt Set 0b No change of TIC.RCSRE 1b Set TIC.RCSRE to 1b This bit supports interrupt enable. It controls set of TIC.RCSRE. This bit is always read as 0b.
13	MFWS2	0b	R/W	MAC status FIFO2 Warning interrupt Set 0b No change of TIC.MFWE2 1b Set TIC.MFWE2 to 1b This bit supports interrupt enable. It controls set of TIC.MFWE2. This bit is always read as 0b.
12	MFUS2	0b	R/W	MAC status FIFO2 Updated interrupt Set 0b No change of TIC.MFUE2 1b Set TIC.MFUE2 to 1b This bit supports interrupt enable. It controls set of TIC.MFUE2. This bit is always read as 0b.
11	MFWS	0b	R/W	MAC status FIFO Warning interrupt Set 0b No change of TIC.MFWE 1b Set TIC.MFWE to 1b This bit supports interrupt enable. It controls set of TIC.MFWE. This bit is always read as 0b.
10	MFUS	0b	R/W	MAC status FIFO Updated interrupt Set 0b No change of TIC.MFUE 1b Set TIC.MFUE to 1b This bit supports interrupt enable. It controls set of TIC.MFUE. This bit is always read as 0b.
9 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	FTS	0b	R/W	Frame Transmitted interrupt Set 0b No change of TIC.FTE 1b Set TIC.FTE to 1b This bit supports interrupt enable. It controls set of TIC.FTE. This bit is always read as 0b.

31.4.1.41 Transmit Interrupt Disable Register (TID)

The TID register set to 0 each bits in transmit interrupt control register (TIC).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TDPD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RCSRD	MFWD2	MFUD2	MFWD	MFUD	—	—	—	—	—	—	—	—	—	FTD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	TDPD	0b	R/W	Transmit Descriptor Processed interrupt Disable 0b No change of TIC.TDPE 1b Set TIC.TDPE to 0b This bit supports interrupt enable. It controls set of TIC.TDPE. This bit is always read as 0b.
15	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14	RCSRD	0b	R/W	Reference Clock Stop Req interrupt Disable 0b No change of TIC.RCSRE 1b Set TIC.RCSRE to 0b This bit supports interrupt enable. It controls set of TIC.RSCRE. This bit is always read as 0b.
13	MFWD2	0b	R/W	MAC status FIFO2 Warning interrupt Disable 0b No change of TIC.MFWE2 1b Set TIC.MFWE2 to 0b This bit supports interrupt enable. It controls set of TIC.MFWE2. This bit is always read as 0b.
12	MFUD2	0b	R/W	MAC status FIFO2 Updated interrupt Disable 0b No change of TIC.MFUE2 1b Set TIC.MFUE2 to 0b This bit supports interrupt enable. It controls set of TIC.MFUE2. This bit is always read as 0b.
11	MFWD	0b	R/W	MAC status FIFO Warning interrupt Disable 0b No change of TIC.MFWE 1b Set TIC.MFWE to 0b This bit supports interrupt enable. It controls set of TIC.MFWE. This bit is always read as 0b.
10	MFUD	0b	R/W	MAC status FIFO Updated interrupt Disable 0b No change of TIC.MFUE 1b Set TIC.MFUE to 0b This bit supports interrupt enable. It controls set of TIC.MFUE. This bit is always read as 0b.
9 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	FTD	0b	R/W	Frame Transmitted interrupt Disable 0b No change of TIC.FTE 1b Set TIC.FTE to 0b This bit supports interrupt enable. It controls set of TIC.FTE. This bit is always read as 0b.

31.4.1.42 Common Interrupt Enable register (CIE)

The CIE register is used to control interrupts line.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CTIE	—	—	—	—	—	—	—	CRIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	CTIE	0b	R/W	Common Transmit Interrupt Enable 0b Disabled 1b Enabled This bit controls transmit related interrupt outputs of Gigabit Ethernet Interface. It has no influence on internal flagging in Gigabit Ethernet Interface SFR (e.g. ISS).
7 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CRIE	0b	R/W	Common Receive Interrupt Enable 0b Disabled 1b Enabled This bit controls receive related interrupt outputs of Gigabit Ethernet Interface. It has no influence on internal flagging in Gigabit Ethernet Interface SFR (e.g. ISS).

31.4.1.43 Interrupt Summary Status Register (ISS)

The ISS register gives a summary of the states of DMAC-related interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DPM15	DPM14	DPM13	DPM12	DPM11	DPM10	DPM9	DPM8	DPM7	DPM6	DPM5	DPM4	DPM3	DPM2	DPM1	RCSRM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MFWM2	MFUM2	—	RFWM	MFWM	MFUM	—	—	MM	EM	—	—	—	FTM	—	FRM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	DPMi	All 0	R	Descriptor Processed Mirror i 0b No interrupt pending 1b Descriptor interrupt pending Descriptor Processed Mirror i (i = 1 to 15) [Changing condition] This bit is set when any matching pair of DIC.DPEi enable and DIS.DPFi flag are both 1b.
16	RCSRM	0b	R	Reference Clock Stop Req Mirror 0b No interrupt pending 1b Reference Clock Stop Req interrupt pending This bit indicates that TIS.RCSRF is 1b and TIC.RCSRE is 1b. [Changing condition] This bit is updated when TIS.RCSRF or TIC.RCSRE changes.
15	MFWM2	0b	R	MAC status FIFO2 Warning Mirror 0b No interrupt pending 1b MAC status FIFO2 warning interrupt pending This bit indicates that TIS.MFWF2 is 1b and TIC.MFWE2 is 1b. [Changing condition] This bit is updated when TIS.MFWF2 or TIC.MFWE2 changes.
14	MFUM2	0b	R	MAC status FIFO2 Updated Mirror 0b No interrupt pending 1b MAC status FIFO2 updated interrupt pending This bit indicates that TIS.MFUF2 is 1b and TIC.MFUE2 is 1b. [Changing condition] This bit is updated when TIS.MFUF2 or TIC.MFUE2 changes.
13	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	RFWM	0b	R	Reception FIFO Warning Summary 0b No interrupt pending 1b Reception FIFO warning interrupt pending This bit indicates that RIS1.RFWF is 1b and RIC1.RFWE is 1b. [Changing condition] This bit is updated when RIS1.RFWF or RIC1.RFWE changes.

Bit	Bit Name	Initial Value	R/W	Description
11	MFWM	0b	R	MAC status FIFO Warning Mirror 0b No interrupt pending 1b MAC status FIFO warning interrupt pending This bit indicates that TIS.MFWF is 1b and TIC.MFWE is 1b. [Changing condition] This bit is updated when TIS.MFWF or TIC.MFWE changes.
10	MFUM	0b	R	MAC status FIFO Updated Mirror 0b No interrupt pending 1b MAC status FIFO updated interrupt pending This bit indicates that TIS.MFUF is 1b and TIC.MFUE is 1b. [Changing condition] This bit is updated when TIS.MFUF or TIC.MFUE changes.
9 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	MM	0b	R	MAC Mirror 0b No interrupt pending 1b MAC interrupt pending This bit indicates that there is a pending interrupt request issued by MAC. *Note*: The MAC is able to assert interrupt also in CONFIG mode.
6	EM	0b	R	Error Mirror 0b No interrupt pending 1b Error interrupt pending This bit indicates that at least one enabled flag in EIS is 1b or EIS.QFS is 1b. [Changing condition] This bit is updated when EIS or EIC changes.
5 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	FTM	0b	R	Frame Transmitted Mirror 0b No interrupt pending 1b Frame transmitted interrupt pending [Changing condition] This bit is set when any matching pair of TIC.FTE enable and TIS.FTF flag are both 1b. This bit is set when any matching pair of TIC.TDPE enable and TIS.TDPF flag are both 1b.
1	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	FRM	0b	R	Frame Received Mirror 0b No interrupt pending 1b Frame received interrupt pending [Changing condition] This bit is set when any matching pair of RIC0.FRE enable and RIS0.FRF flag are both 1b. This bit is set when any matching pair of RIC3.RDPE enable and RIS3.RDPF flag are both 1b.

31.4.2 E-MAC Registers

31.4.2.1 E-MAC operating mode register 1 (CXR20)

CXR20 sets the E-MAC operating mode.

Rewriting the bits in this register is prohibited when the transmission function is enabled (TPE = 1) and reception function is enabled (RPE = 1).

[Write restrictions]

Rewrite all the bits except for TPE and RPE when transmission function is disabled (TPE = 0) and reception function is disabled (RPE = 0).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TRCCM	RCPT	TCPT	RCSC	CXSER	DPAD	RZPF	TZPF	PFR	RXF	TXF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CER	—	—	—	—	—	RPE	TPE	LPM	—	—	DPM	PRM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	TRCCM	0b	R/W	TINT/RINT Counter Clear Mode Specifies the method of clearing TINT source counter (CXR40-43) and RINT source counter (CXR50-57). 1: Clears the register to 0 when reading the register. 0: Clears the register to 0 when writing to the register. (clears any write data to 0 at write)
25	RCPT	0b	R/W	Reception CRC Pass Through 1: CRC of received frame is transferred to TOE. RCSC (auto calculation of checksum of received frame data part) function is disabled at this time. 0: CRC of received frame is not transferred to TOE.
24	TCPT	0b	R/W	Transmit CRC Pass Through 1: CRC of transmit frame is not calculated automatically (hardware calculation). It is necessary to input the data with CRC added in AT port. 0: CRC of transmit frame is calculated automatically (hardware calculation). It is necessary to input the data without CRC added in AT port.
23	RCSC	0b	R/W	Reception Check Sum Calculation 1: Checksum of received frame data part is automatically calculated (hardware calculation). RCPT must be set to 0 at this time. 0: Checksum of received frame data part is not calculated automatically (hardware calculation).
22	CXSER	0b	R/W	Carrier extension Short Error Function for debug. It should always be set to 0.
21	DPAD	0b	R/W	Data Padding 1: Less than 60-byte data is transmitted without padding. 0: Less than 60-byte data is padded and transmitted as 60-byte data.

Bit	Bit Name	Initial Value	R/W	Description
20	RZPF	0b	R/W	Reception Zero PAUSE Frame 1: Enables to receive PAUSE frame with TIME parameter value being 0. 0: Disables receiving PAUSE frame with TIME parameter value being 0.
19	TZPF	0b	R/W	Transmit Zero PAUSE Frame 1: PAUSE frame of 0 Time parameter value is transmitted at asserting “Reception FIFO Critical Level Notification” from DMAC 0: PAUSE frame of 0 Time parameter value is not transmitted at asserting “Reception FIFO Critical Level Notification” from DMAC
18	PFR	0b	R/W	PAUSE Frame Receive 1: PAUSE frame is transferred to TOE. 0: PAUSE frame is not transferred to TOE.
17	RXF	0b	R/W	Receive Flow control mode 1: Receive flow control function (PAUSE frame reception) is enabled. 0: Receive flow control function (PAUSE frame reception) is disabled.
16	TXF	0b	R/W	Transmit Flow control mode 1: Transmit flow control function (PAUSE frame transmission) is enabled. 0: Transmit flow control function (PAUSE frame transmission) is disabled.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	CER	0b	R/W	CRC Error Used for debugging. It should always be set to 0.
11 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
8 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	RPE	0b	R/W	Receive Port Enable 1: Receive function of E-MAC is enabled. 0: Receive function of E-MAC is disabled. If 1 is changed to 0 while a frame is being received, the frame is completely received.
5	TPE	0b	R/W	Transmit Port Enable 1: Transmission function of E-MAC is enabled. 0: Transmission function of E-MAC is disabled. If 1 is changed to 0 while a frame is being transmitted, the frame is completely transmitted.
4	LPM	0b	R/W	Low Power Mode 1: E-MAC operates as Low Power Mode. 0: E-MAC operates as Normal Mode. (When LPM bit is set to “1”, DPM bit should be also set to “1”)
3 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	DPM	0b	R/W	Duplex Mode 1: Operates in full duplex mode. 0: Operates in half duplex mode.
0	PRM	0b	R/W	Promiscuous Mode 1: All the frames except for PAUSE frame are received. Self-addressed unicast, different address unicast, multicast, and broadcast frames are all transferred to TOE. PAUSE frame reception is controlled by PFR bit. 0: Self-addressed unicast, multicast, and broadcast frames are received, then transferred to TOE.

31.4.2.2 Maximum receive frame length register (CXR2A)

CXR2A sets the maximum receive frame length.

[Write restrictions]

If CXR20_bit6 is set to 1, it is not allowed to rewrite to this register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LEN_LMT[17:16]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LEN_LMT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17 to 0	LEN_LMT	H'0_0000	R/W	frame LENgth upper LiMiT value Sets the maximum receive frame length (Including CRC). If the receive frame length exceeds the set value, frame length error (RINT4) is asserted. < Set value > — < Check value > H'0_0000 to H'0_05EE — 1518 bytes H'0_05EF — 1519 bytes H'0_05F0 — 1520 bytes … H'0_07FF — 2047 bytes H'0_0800 — 2048 bytes … H'0_1000 — 4096 bytes … H'1_0000 — 65536 bytes … H'2_0000 to H'3_FFFF — 131072 bytes

31.4.2.3 Interrupt status register (CXR21)

CXR21 indicates interrupt status.

[Read restrictions]

When the PHY used does not have an INT signal, consider the read value of PHYI bit (bit3) as undefined.

[Write restrictions]

Writing 0 is invalid. 1 or 0 cannot be written to bit 3.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PFRI	PHYI	LINKI	—	FCI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	X	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	PFRI	0b	R/W	PAUSE Frame Retry Interrupt Interrupt source based on the PAUSE frame transmission retry count: Indicates that the PAUSE frame transmission is retried for the number of times specified using the CXR81 register. This bit can be cleared to 0 in the following way. (1) Read the CXR82 register to clear CXR82. (2) Write 1 to this bit.
3	PHYI	Xb	R/W	PHY Interrupt PHY chip interrupt source: Indicates status of 'PHY INT' of the E-MAC input/output signals. 1: Interrupt from PHY is asserted. 0: Interrupt from PHY is negated. Signal polarity of 'PHY INT' can be set using CXR2C. *Note*: When the PHY used does not have an INT signal, the read value of this bit should be undefined.
2	LINKI	0b	R/W	LINK Interrupt Link status change interrupt source: Indicates that on/off of bit 0 (LINK) of CXR2B has changed. Write 1 to clear this bit. *Note*: Before referring to this bit, determine whether to use PHY_LINK input signal or RGMII-Inband Status LINK bit as the link using SEL_LINK (bit 0) in the CXR31 register.
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	FCI	0b	R/W	False Carrier Interrupt Invalid carrier detected interrupt source: Indicates that an invalid carrier was detected in the PHY chip. Write 1 to clear this bit to 0.

31.4.2.4 Enables/disables interrupts register (CXR22)

CXR22 enables/disables interrupts.

It is possible to enable/disable interrupts for each interrupt source indicated in CXR21 (bit unit).

[Write restrictions]

If the PHY used does not have an INT signal, set PHYIM bit (bit 3) to 0. In the case of without using PHY_LINK signal and without using RGMII Inband Status function (case of setting '10' or '11' to SEL_LINK of CXR31), set PHYIM bit (bit 2) to 0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PFRIM	PHYIM	LINKIM	—	FCIM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	PFRIM	0b	R/W	PAUSE Frame Retry Interrupt Mask 1: Interrupt by PFRI bit in the CXR21 register is enabled. 0: Interrupt by PFRI bit in the CXR21 register is disabled.
3	PHYIM	0b	R/W	PHY Interrupt Mask 1: Interrupt by PHYI bit in the CXR21 register is enabled. 0: Interrupt by PHYI bit in the CXR21 register is disabled. *Note*: If the PHY used does not have an INT signal, set this bit to 0.
2	LINKIM	0b	R/W	LINK Interrupt Mask 1: Interrupt by LINKI bit in the CXR21 register is enabled. 0: Interrupt by LINKI bit in the CXR21 register is disabled.
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	FCIM	0b	R/W	False Carrier Interrupt Mask 1: Interrupt by FCI bit in the CXR21 register is enabled. 0: Interrupt by FCI bit in the CXR21 register is disabled.

31.4.2.5 PHY register access register (CXR23)

There are two methods to access the PHY register: using the E-MAC-MDIOC block and using the CXR23 register.

Which access method to be used can be set using the MDIOMOD register.

For PHY register access using CXR23, see **Section 31.5.14.3(2), PHY Register Access Method (Using CXR23)** and IEEE Standard 802.3, 2000 Edition “22.2.4.5 Management frame structure.”

[Write restrictions]

Writing bit 3 is invalid.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	MDI	MDO	MM	MDC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	X	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	MDI	Xb	R/W	Management Data Input
2	MDO	0b	R/W	Management Data Output
1	MM	0b	R/W	Management Data Management mode The write value should be 1. The read value should be 0.
0	MDC	0b	R/W	Management Clock

31.4.2.6 Link status register (CXR2B)

CXR2B indicates link status.

[Write restrictions]

Write disabled

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SOFT_LINK	SIGNAL_LINK	INBAND_LINK	LINK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	X	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	SOFT_LINK	0b	R	Software LINK Indicates the LINK ON/OFF status of software LINK setting (CXR2G) 1: Link ON (Link Up) 0: Link OFF (Link Down) This bit indicates the value of CXR2G.
2	SIGNAL_LINK	0b	R	LINK signal input Indicates the LINK ON/OFF status of the PHY_LINK input signal. 1: Link ON (Link Up) 0: Link OFF (Link Down) *Note*: Before referring to this bit, set the polarity of the PHY_LINK input signal using PHYLINKP (bit 1) in the CXR2C register.
1	INBAND_LINK	Xb	R	Inband Status LINK Indicates the link bit of RGMII Inband status. 1: Link ON (Link Up) 0: Link OFF (Link Down) *Note*: If the PHY used does not support Inband-status, the read value of this bit should be undefined.
0	LINK	0b	R	LINK status Indicates link status. 1: Link ON (Link Up) 0: Link OFF (Link Down) *Note*: Before referring to this bit, determine whether to use the PHY_LINK input signal, RGMII-Inband Status LINK bit or software LINK as the link using SEL_LINK (bit3, bit0) in the CXR31 register. In addition, to use the PHY_LINK input signal, set the polarity of PHY_LINK input using PHYLINKP (bit 1) in the CXR2C register.

31.4.2.7 Polarity set register (CXR2C)

CXR2C sets the polarity of ‘PHY_LINK’ and ‘PHY_INT’ signals.

[Write restrictions]

Rewrite to this register with bits 2 and 3 in CXR22 being 0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PHYLINKP	PHYIP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	PHYLINKP	0b	R/W	PHY_LINK Polarity Sets polarity of input signal ‘avb_miimg_link’. 1: High active (LINK ON [LINK UP] status with high) 0: Low active (LINK ON [LINK UP] status with low)
0	PHYIP	0b	R/W	PHY Interrupt Polarity Sets polarity of input signal ‘avb_miimg_int’. 1: High active (interrupt status with high) 0: Low active (interrupt status with low)

31.4.2.8 In-Band Status set register (CXR31)

CXR31 sets whether to apply In-Band Status function*1 of RGMII-IF to the E-MAC operating mode.

It is possible to select the configured value by software for Link status.

Note 1. In-Band Status function (optional) is a notification function from PHY to E-MAC about Link Status (Up/Down), transmission speed (10 M/100 M/1 G) and Duplex (Half/Full) via RGMII-IF.

[Write restrictions]

Rewrite to this register with transmission function disabled (CXR20 TPE = 0) and reception function also disabled (CXR20 RPE = 0). If the PHY used does not support In-band Status, set ‘H’000_0000’ to this register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SEL_LINK[1]	SEL_SPEED	SEL_DUPLEX	SEL_LINK[0]
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	SEL_LINK[1]	0b	R/W	Select LINK See bit0 of this register.
2	SEL_SPEED	0b	R/W	Select Speed*1,*2 ‘1’: The information from In-band Status is used for setting the E-MAC operation speed. ‘0’: CXR2D register value is used for setting the E-MAC operation speed.
1	SEL_DUPLEX	0b	R/W	Select Duplex*1,*2 ‘1’: The information from In-band Status is used for setting Full/Half Duplex. ‘0’: CXR20 DPM bit value is used for setting Full/Half Duplex.
0	SEL_LINK[0]	0b	R/W	Select LINK*1,*2 SEL_LINK [1:0] ‘11’: The information from CXR20 TPE bit value is used for setting the LINK.*3 The status of CXR20 TPE bit is set to CXR2B LINK bit. When this value is set, set ‘0’ (disable LINK interrupt) to CXR22 LINKIM bit. ‘10’: The information from CXR2B SOFTWR_LINK bit value is used for setting the LINK.*3 The status of CXR2G SOFT_LINK bit is set to CXR2B LINK bit. When this value is set, set ‘0’ (disable LINK interrupt) to CXR22 LINKIM bit. ‘01’: The information from In-band Status is used for setting the LINK. ‘00’: The information from the PHY_LINK input signal is used for setting the LINK.

Note 1. If the PHY does not support In-band Status is used or If MII-IF is used, do not set value using In-band Status.

Note 2. If the device driver software is developed as common for MII-IF and RGMII-IF, not using this register is recommended.

Note 3. When Low Power Mode E-MAC is used, it needs to know the status of PHY as Link Up (Link ON). But, when it does not have the information to judge the status of LINK (because MII-IF is used and PHY_LINK signal is not used), set ‘10’ or ‘11’.
In the case of SEL_LINK [1:0] = ‘10’, CXR2B SOFT_LINK is used. So, software has to check the LINK status by register of

PHY and set that information to CXR2G.
In the case of SEL_LINK [1:0] = ‘11’, CXR20 TPE is used as LINK status.

31.4.2.9 In-Band Status indicate register (CXR32)

CXR32 indicates In-Band-Status of RGMII-IF.

In-Band Status function (option) is the function that the PHY notifies the E-MAC of link status (Up/Down), transfer rate (10 M/100 M/1 G) and duplex (full or half duplex), using the RGMII interface signal.

[Write restrictions]

Write disabled

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	INBAND_SPEED		INBAND_DUPLEX	INBAND_LINK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	X	X	X	X
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 2	INBAND_SPEED	XXb	R	In-Band Status Speed*1 ‘10’: Indicates that the speed of RGMII In-Band status is 1 Gbps. ‘01’: Indicates that the speed of RGMII In-Band status is 100 Mbps. ‘00’: Indicates that the speed of RGMII In-Band status is 10 Mbps.
1	INBAND_DUPLEX	Xb	R	In-Band Status Duplex*1 ‘1’: Indicates that the Duplex bit of RGMII In-Band status is Full Duplex. ‘0’: Indicates that the Duplex bit of RGMII In-Band status is Half Duplex.
0	INBAND_LINK	Xb	R	In-Band Status LINK*1 ‘1’: Indicates that LINK of RGMII In-Band status is LINK ON (LINK UP). ‘0’: Indicates that LINK of RGMII In-Band status is LINK OFF (LINK DOWN).

Note 1. If the PHY used does not support Inband-status, the read value of those bits should be undefined.

31.4.2.10 Mode indicate register (CXR33)

CXR33 indicates the speed, duplex, and link operating mode of the E-MAC.

[Write restrictions]

Write disabled

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SPEED[1:0]		DUPLEX	LINK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 2	SPEED	00b	R	Speed ‘10’: Transfer rate mode of E-MAC is 1 Gbps. ‘01’: Transfer rate mode of E-MAC is 100 Mbps. ‘00’: Transfer rate mode of E-MAC is 10 Mbps.
1	DUPLEX	0b	R	Duplex ‘1’: Duplex mode of E-MAC is Full Duplex. ‘0’: Duplex mode of E-MAC is Half Duplex.
0	LINK	0b	R	LINK ‘1’: LINK status is Link On (Link Up). ‘0’: LINK status is Link Off (Link Down).

31.4.2.11 PHY interface select register (CXR35)

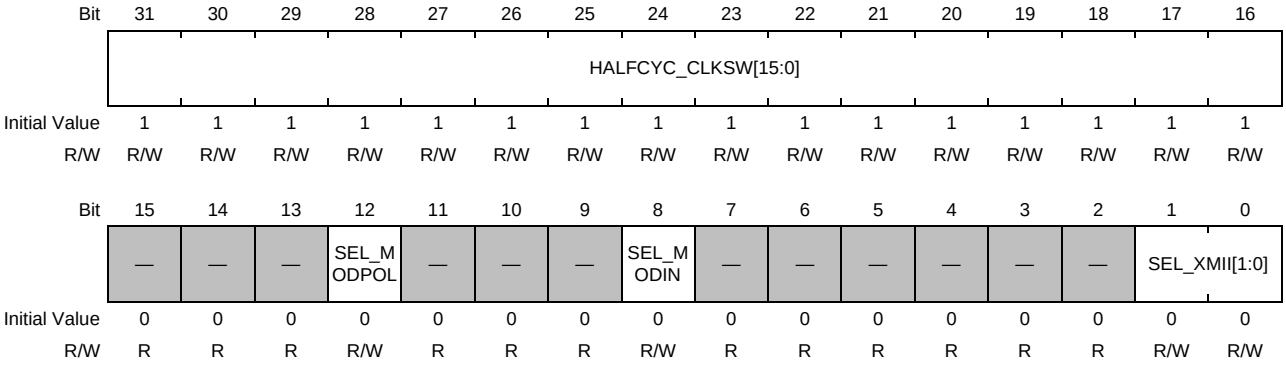
CXR35 selects RGMII or MII interface.

[Write restrictions]

(1) After release reset, make write-access to this register before making write-access to other registers (except MDIOMOD). Even if not need to change the value of this register, make write-access to this register at least one time. Because RGMII/MII MODE is recognized by accessing this register.

(2) If need to re-write the value of this register after making write-access to other register, reset to E-MAC is needed.

(3) If need to re-write the value of this register without making write-access to other register, wait the quadruple time which is set at HALFCYC_CLKSW.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	HALFCYC_CLKSW	H'FFFF	R/W	A clock stop time at the clock switching The wait time (over 20 µs) is needed when PHY interface is selected at initializing Gigabit Ethernet Interface. Set the number of clk_chi (APB clock) cycle to be half of wait time (over 10 us). H'0000: 1 cycle of clk_chi H'0001: 2 cycles of clk_chi H'0002: 3 cycles of clk_chi … H'03E8: 1000 cycles of clk_chi (If clk_chi is 100 MHz (10 ns), half of wait time will be 10 us) H'07D0: 2000 cycles of clk_chi (If clk_chi is 200 MHz (5 ns), half of wait time will be 10 us) … H'FFFF: 65536 cycles of clk_chi (default value)
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	SEL_MODPOL	0b	R/W	polar selection of RGMII/MII Mode input signal '0': (default value) E-MAC recognizes RGMII-IF is used. '1': E-MAC recognizes MII-IF is used.
11 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	SEL_MODIN	0b	R/W	method for selection of RGMII/MII '0': SEL_XMII field of this register is used for RGMII/MII selection (default value) '1': Reserved
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
1 to 0	SEL_XMII	00b	R/W	Selection of PHY interface to be used '00': RGMII interface is used (default value) '01': Reserved '10': MII interface is used '11': Reserved *Note*: The case of SEL_MODIN (bit8) = '0', this field is used to select PHY interface. The case of SEL_MODIN (bit8) = '1', this field is not used and Mode signals which is provided from out of E-MAC is used to select PHY interface.

Note: This register needs to write access at least one time.

[A] The case which CXR35 SEL_XMII is used for the selection of RGMII/MII in APB Clock 100 MHz.

(1) To use RGMII interface, Set 'H'03E8_0000' to this register.

(2) To use MII interface, Set 'H'03E8_0002' to this register.

31.4.2.12 PHY interface indicate register (CXR36)

CXR36 indicates the status of RGMII/MII Interface.

[Write restrictions]

Write disabled

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STS_XMII[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1 to 0	STS_XMII	00b	R	Status of selected PHY interface ‘00’: RGMII interface ‘01’: Reserved ‘10’: MII interface ‘11’: Reserved

31.4.2.13 PAUSE frame register 1 (CXR71)

CXR71 sets the timer value for Auto PAUSE frame.

[Write restrictions]

Set a value other than all 0 when transmission system flow control is enabled. (CXR20_bit16 = 1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APFTP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	APFTP	H'0000	R/W	Auto Pause Frame Time Parameter The value set in this register is used as the Timer value for PAUSE frame during Auto PAUSE Frame transmission. *Note*: Set the value of this register to other than 0000 when transmission system flow control is enabled (CXR20_bit 16 = 1). <set value> <bit time> H'0000 — H'0001 512 × 1 bit time H'0002 512 × 2 bit time H'0003 512 × 3 bit time … H'FFFF 512 × 65535 bit time <transfer rate> <1 bit time> 1000 Mbps 1 ns 100 Mbps 10 ns 10 Mbps 100 ns

31.4.2.14 PAUSE frame register 2 (CXR72)

CXR72 starts the transmission of Manual PAUSE frames and sets the timer value for Manual PAUSE frame.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MPFTP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	MPFTP	H’0000	R/W	Manual Pause Frame Time Parameter When a value is written to this register, PAUSE frame transmission is started using the value as the timer value. <set value> <bit time> H’0000 — H’0001 512 × 1 bit time H’0002 512 × 2 bit time H’0003 512 × 3 bit time … H’FFFF 512 × 65535 bit time <transfer rate> <1 bit time> 1000 Mbps 1 ns 100 Mbps 10 ns 10 Mbps 100 ns

31.4.2.15 PAUSE frame register 3 (CXR8A)

CXR8A is a transmit counter for PAUSE frame.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFTXC[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	PFTXC	H'0000	R/W	Pause Frame TX Counter Indicates the number of times of transmission of Auto PAUSE frames and Manual PAUSE frames in total. This register is cleared to 0 when read.

31.4.2.16 PAUSE frame register 4 (CXR80)

CXR80 is a receive counter for PAUSE frame.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFRXC[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	PFRXC	H'0000	R/W	Pause Frame RX Counter Indicates the number of PAUSE frames received when receive system flow control is enabled (CXR20_bit17 = 1). This register is cleared to 0 when read.

31.4.2.17 PAUSE frame register 5 (CXR81)

CXR81 sets the upper limit of the number of autonomous transmissions of PAUSE frame.

[Write restrictions]

Rewriting this register is prohibited when 1 is set to CXR20_bit 5.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFRTULMT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	PFRTULMT	H’0000	R/W	Pause Frame Retry Upper LiMiT Sets the number of times of Auto PAUSE frame transmission (upper limit) from the time of asserting “Reception FIFO Critical Level Notification” from DMAC assertion to negation. <set value> <Number of transmissions> H’0000 No limit H’0001 1 time H’0002 2 times H’0003 3 times … H’FFFF 65535 times

31.4.2.18 PAUSE frame register 6 (CXR82)

CXR82 counter indicates the number of Auto PAUSE frame transmissions from “Reception FIFO Critical Level Notification” from DMAC assertion to negation.

[Write restrictions]

Write disabled

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFRTC[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	PFRTC	H’0000	R	Pause Frame Retry Counter Indicates the number of times of Auto PAUSE frame transmission from “Reception FIFO Critical Level Notification” from DMAC assertion to negation. This register is cleared to 0 when read and at the start of “Reception FIFO Critical Level Notification” from DMAC assertion.

31.4.2.19 E-MAC operating mode register 2 (CXR2D)

CXR2D sets the operating mode of the E-MAC.

Rewriting this register bits is prohibited with transmission function enabled (CXR20<TPE> = 1) and reception function enabled (CXR20<RPE> = 1).

[Write restrictions]

Rewriting this register is prohibited when 1 is set to CXR20_bit 5 or 6.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	SPEED[1:0]		—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
5 to 4	SPEED	00b	R/W	Transmit SPEED Set transfer rate.
3 to 0	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

SPEED[1] bit5	SPEED[0] bit4	Transmit Speed
0	0	10 Mbps
0	1	100 Mbps
1	0	1000 Mbps
1	1	Reserved

31.4.2.20 Software LINK status register (CXR2G)

CXR2G indicates the status of Software LINK.

[Write restrictions]

Before setting ‘1’ to this register, set SEL_LINK of CXR31 whether using PHY_LINK input signal, using LINK bit of RGMII-Inband Status or using LINK setting by software.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SOFT_LINK	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	SOFT_LINK	0b	R/W	Software LINK This bit is used when the LINK status is set by software. ‘1’: Link ON (Link Up) ‘0’: Link OFF (Link Down) *Note*: Before setting ‘1’ to this bit, set SEL_LINK of CXR31 whether using PHY_LINK input signal, using LINK bit of RGMII-Inband Status or using LINK setting by software. Additionally, if PHY_LINK input signal is used, set the polar of PHY_LINK input signal by PHYLINKP of CXR2C.
2 to 0	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

31.4.2.21 Mac Address register 1 (CXR24)

CXR24 set the universal MAC address (upper 32 bits).

[Write restrictions]

Rewriting this register is prohibited when 1 is set to CXR20_bit 5 or 6.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	UMADR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UMADR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	UMADR	H'0000_0000	R/W	Upper MAC Address Upper-32bit MAC address

31.4.2.22 Mac Address register 2 (CXR25)

CXR25 set the universal MAC address (lower 16 bits).

[Write restrictions]

Rewriting this register is prohibited when 1 is set to CXR20_bit 5 or 6.

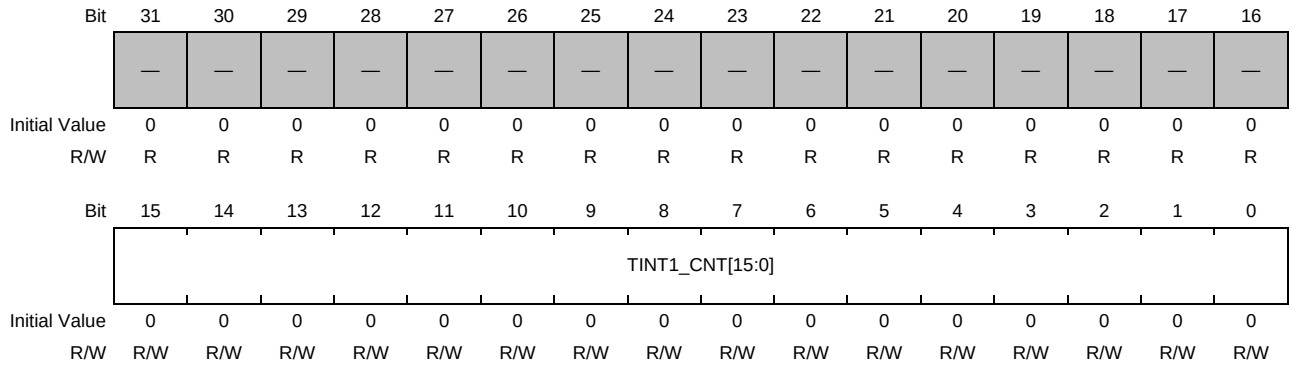
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LMADR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	LMADR	H'0000	R/W	Lower MAC Address Lower-16bit MAC address

31.4.2.23 TINT1 counter register (CXR40)

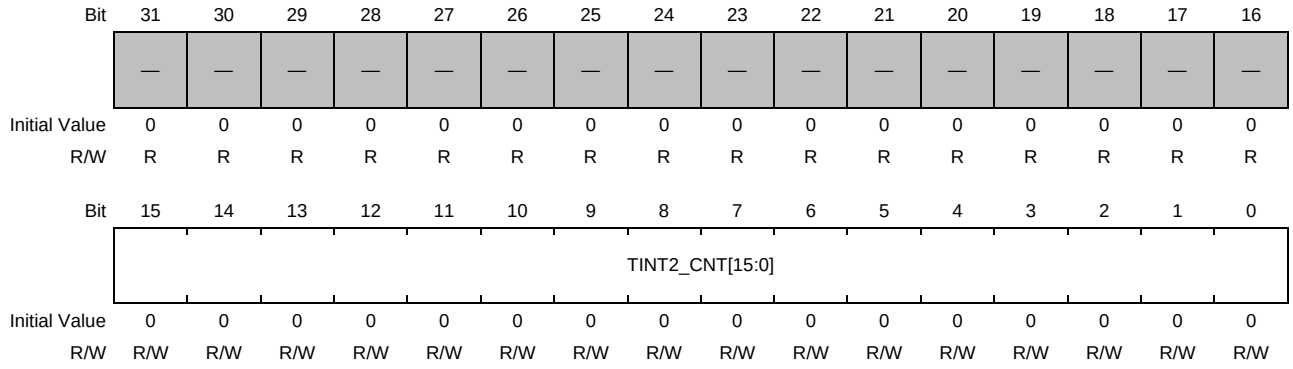
CXR40 indicates the number of occurrences of TINT1.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	TINT1_CNT	H'0000	R/W	TINT1 Counter Indicates the number of occurrences of TINT1. TINT1: Frame transmission timeout When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

31.4.2.24 TINT2 counter register (CXR41)

CXR41 indicates the number of occurrences of TINT2.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	TINT2_CNT	H'0000	R/W	TINT2 Counter Indicates the number of occurrences of TINT2. TINT2: Delayed collision detection When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

31.4.2.25 TINT3 counter register (CXR42)

CXR42 indicates the number of occurrences of TINT3.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TINT3_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	TINT3_CNT	H'0000	R/W	TINT3 Counter Indicates the number of occurrences of TINT3. TINT3: Transmission is terminated in error with TINT3 generated under the following conditions: Delay collision occurs at the end of transmit frame, the receive frame at the time of the collision continues until the start of next frame transmission, but does not continue until the end of the transmission. However, these conditions do not occur in normal operation.

31.4.2.26 RINT1 counter register (CXR50)

CXR50 indicates the number of occurrences of RINT1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RINT1_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT1_CNT	H'0000	R/W	RINT1 Counter Indicates the number of occurrences of RINT1. RINT1: Receive frame CRC error When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

31.4.2.27 RINT2 counter register (CXR51)

CXR51 indicates the number of occurrences of RINT2.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RINT2_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT2_CNT	H'0000	R/W	RINT2 Counter Indicates the number of occurrences of RINT2. RINT2: Frame reception error When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

31.4.2.28 RINT3 counter register (CXR52)

CXR52 indicates the number of occurrences of RINT3.

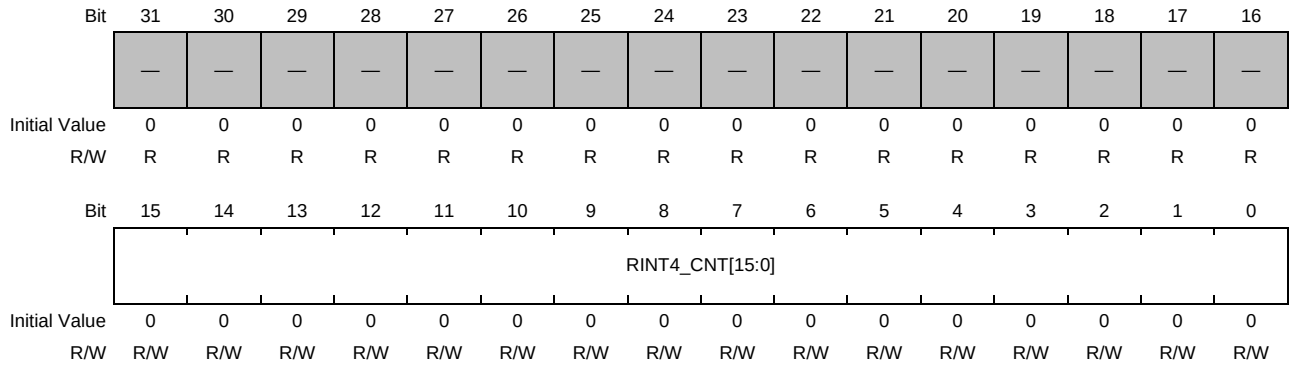
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RINT3_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT3_CNT	H'0000	R/W	RINT3 Counter Indicates the number of occurrences of RINT3. RINT3: Erroneous frame length (less than 64 bytes frame was received.) When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

31.4.2.29 RINT4 counter register (CXR53)

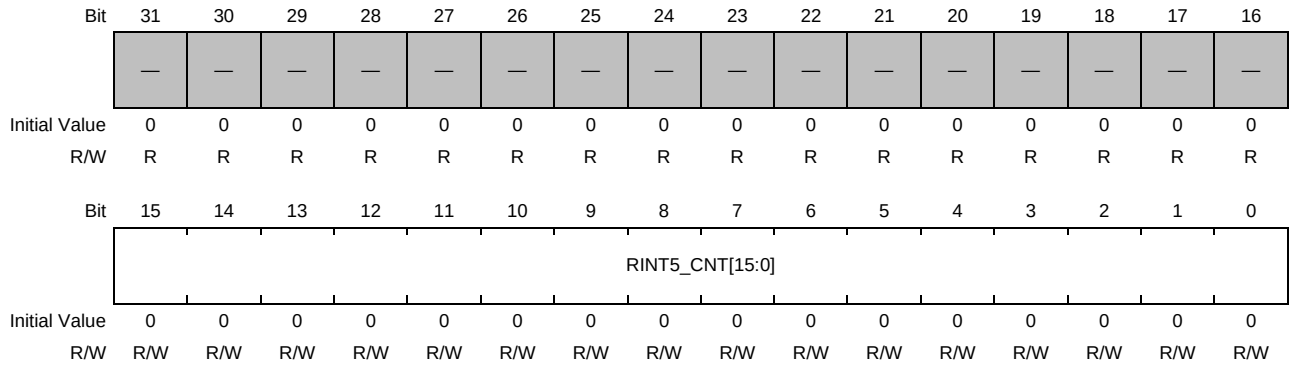
CXR53 indicates the number of occurrences of RINT4.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT4_CNT	H'0000	R/W	RINT4 Counter Indicates the number of occurrences of RINT4. RINT4: Erroneous frame length {A frame longer than the maximum frame length was received. Maximum frame length (including CRC) can be specified using the LEN_LMT field in the CXR2A register.} When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

31.4.2.30 RINT5 counter register (CXR54)

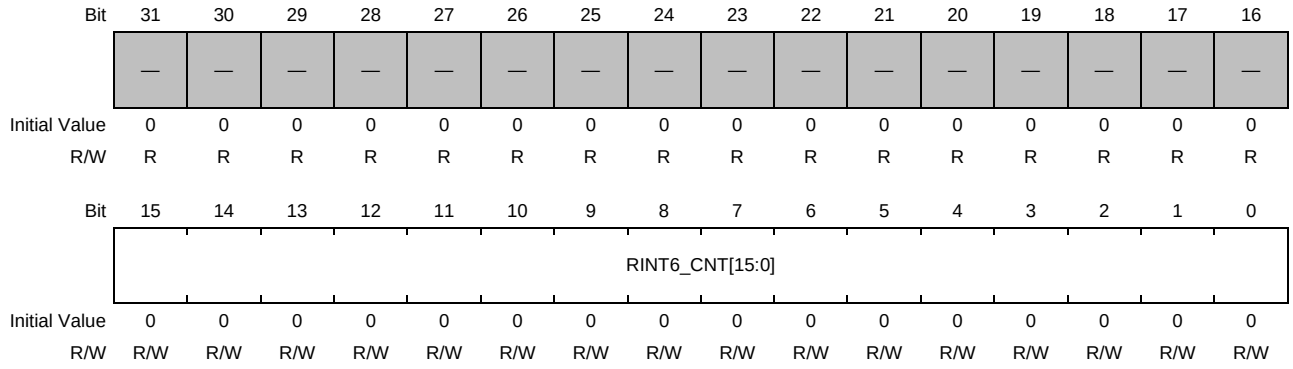
CXR54 indicates the number of occurrences of RINT5.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT5_CNT	H'0000	R/W	RINT5 Counter Indicates the number of occurrences of RINT5. RINT5: Fractional bit error When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data)

31.4.2.31 RINT6 counter register (CXR55)

CXR55 indicates the number of occurrences of RINT6.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT6_CNT	H'0000	R/W	RINT6 Counter Indicates the number of occurrences of RINT6. RINT6: Carrier extension lost When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

31.4.2.32 RINT7 counter register (CXR56)

CXR56 indicates the number of occurrences of RINT7.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RINT7_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT7_CNT	H'0000	R/W	RINT7 Counter Indicates the number of occurrences of RINT7. RINT7: Carrier extension error When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

31.4.2.33 RINT8 counter register (CXR57)

CXR57 indicates the number of occurrences of RINT8.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RINT8_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT8_CNT	H'0000	R/W	RINT8 Counter Indicates the number of occurrences of RINT8. RINT8: Multicast frame received When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

31.4.2.34 MDIO status register (MDIOSTS)

MDIOSTS indicates the status of PHY register access (MDIO access).

[Write restrictions]

Write disabled

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BSY
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	BSY	0b	R	PHY register access Busy flag 0: Access enabled 1: Busy Do not write to MDIOCMD, MDIOADR, MDIODAT or MDIOMOD registers when this bit is1.

31.4.2.35 MDIO command register (MDIOCMD)

MDIOCMD specifies the commands for PHY register access (MDIO access).

[Write restrictions]

Do not write to this register when the BSY bit in the MDIOSTS register is 1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OP[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1 to 0	OP	00b	R/W	Operation Code for PHY register access 00: Setting prohibited 01: WRITE 10: READ 11: Setting prohibited

31.4.2.36 MDIO address register (MDIOADR)

MDIOADR specifies PHY address and PHY register address.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	A[9:5]					A[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9 to 5	A[9:5]	00000b	R/W	PHY address 5 bits
4 to 0	A[4:0]	00000b	R/W	PHY register address 5 bits

31.4.2.37 MDIO data register (MDIODAT)

MDIODAT specifies PHY write data and indicates read data.

[Write restrictions]

Do not write to this register when the BSY bit in the MDIOSTS register is 1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	D[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	D[15:0]	H'0000	R/W	PHY register data When writing: Write data When reading: Read data

31.4.2.38 MDIO mode register (MDIOMOD)

MDIOMOD sets the operating mode of the MDIO controller.

[Write restrictions]

Writing to this bit is prohibited during PHY register access.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SEL_MDIO_N	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CYC_OCLK[7:0]							
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	SEL_MDIO_N	1b	R/W	Selection for PHY register access (MDIO access selection) 0: Uses MDIO controller for PHY register access. 1: Uses CXR23 register for PHY register access. (Initial value)
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	CYC_OCLK	H'FF	R/W	MDC (MDIO control clock) half cycle (number of cycles for system clock) H'00: 7 cycles H'01: 7 cycles ... H'06: 7 cycles H'07: 8 cycles H'08: 9 cycles H'09: 10 cycles ... H'FE: 255 cycles H'FF: 256 cycles (Initial value) MDC half cycle should be set to be 200 ns or more. [Calculation example] 27 cycles at a 133-MHz Clock (1 cyc = 7.5 ns), 200 ns or more -> Setting value is H'1A (26d) 7.5 × 27 = 202.5 ≥ 200 [Setting example] //66.66 MHz (15 ns): Setting value ≥ H'0D (13d) (14 cyc × 15 ns = 210 ns) //133.3 MHz (7.5 ns): Setting value ≥ H'1A (26d) (27 cyc × 7.5 ns = 202.5 ns) //266.6 MHz (3.75 ns): Setting value ≥ H'35 (53d) (54 cyc × 3.75 ns = 202.5 ns)

31.4.2.39 Low Power Mode register 1 (LPTXMOD1)

LPTXMOD1 sets the operating mode of the Low Power Transmission.

[Write restrictions]

Write to this register when the CXR20_bit6 = ‘0’ and CXR20_bit5 = ‘0’. Recommended value is H’0000_0000 (Initial value).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	GCYC_TCSTOP[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MCYC_TCSTOP[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 16	GCYC_TCSTOP	H’000	R/W	Number of minimum clock cycles until stopping RGMII TXC (1 Gbps) 0: 10 cycles (Initial value) 1 to 8: (Setting prohibited) 9: 9 cycles 10: 10 cycles 11: 11 cycles (skip the rest)
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	MCYC_TCSTOP	H’000	R/W	Number of minimum clock cycles until stopping RGMII TXC (100 Mbps) 0: 10 cycles (Initial value) 1 to 8: (Setting prohibited) 9: 9 cycles 10: 10 cycles 11: 11 cycles (skip the rest)

31.4.2.40 Low Power Mode register 2 (LPTXMOD2)

LPTXMOD2 sets the operating mode of the Low Power Transmission.

[Write restrictions]

Write to this register when the CXR20_bit6 = ‘0’ and CXR20_bit5 = ‘0’. When set ‘1’ to this register, check the configuration of Clock stoppable of PHY is ‘1’.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	MAC_TLPI_TYPE	—	—	—	STP_TXC_LPI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	MAC_TLPI_TYPE	0b	R/W	Transmission LPI Operation Type of MAC ‘1’: MAC TX LPI Slave Mode E-MAC does not transmit LPI code during LPI period. But using LPI Wakeup parameters (LPTXGTH3/LPTXMTH3) to make the delay time of transmitting such as for PAUSE Frame. When making PHY control the operation of LPI (Low Power Idle), set ‘1’ to this bit. ‘0’: MAC TX LPI Master Mode E-MAC controls the operation of LPI. (Initial value)
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STP_TXC_LPI	0b	R/W	Setting of stopping LPI TXC (this bit is used only RGMII Mode) ‘1’: The clock for RGMII transmitting (fet_miitx_gtx_clk) is stopped during output of LPI code. ‘0’: The clock for RGMII transmitting (fet_miitx_gtx_clk) is not stopped during output of LPI code. (Initial value) Set ‘1’ to this bit when TX Clock stop capable (Note1) of PHY is ‘1’ *Note 1.* See the Bit 6 Clock Stop Capable (MAC may stop clock during LPI) of PCS Status register in PHY MMD extended register (MMD address 3, REG address 1) *Note 2.* In the case of CCC.ERCS = 1 (The “clk_miitx_gtx_refclk” stop function is enable), even if this bit was set to ‘0’, the clock for RGMII transmitting (fet_miitx_gtx_clk) is stopped during the stopping period of “clk_miitx_gtx_refclk”. Because “fet_miitx_gtx_clk” is generated from “clk_miitx_gtx_refclk”.

31.4.2.41 RGMII Low Power parameter register 1 (LPTXGTH1)

LPTXGTH1 sets the change timing to Low Power Idle Transmission Mode at 1 Gbps.

[Write restrictions]

Write to this register when the CXR20_bit6 = ‘0’, CXR20_bit5 = ‘0’. Recommended value is H’0000_0000 (Initial value).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPTX_GSTH1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPTX_GSTH1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

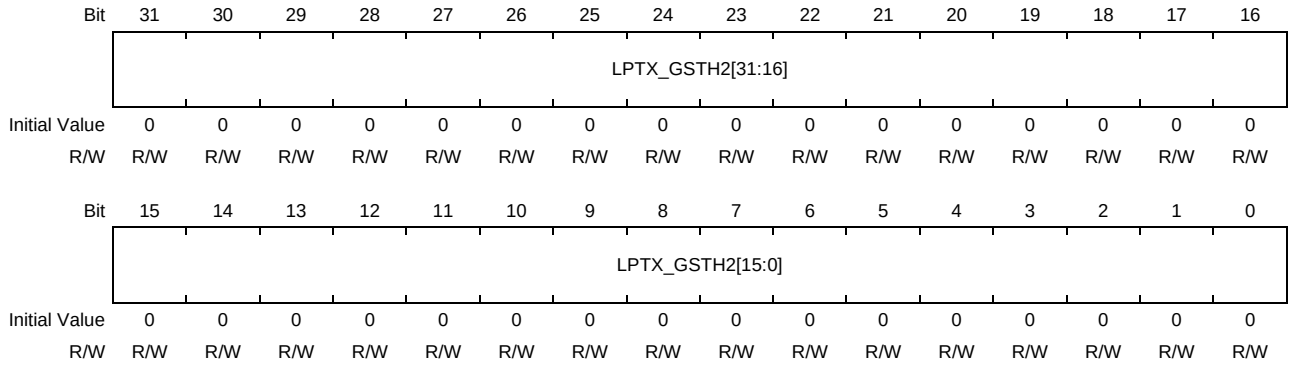
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_GSTH1	H’0000_0000	R/W	Threshold value of LPI code transmission standby status (1 Gbps) (Unit is 32 ns) 0: (Initial value) When MAC does not receive transmission request during 1 ms, MAC starts transmitting LPI code. 1 to 7: Setting prohibited ... 32: When MAC does not receive transmission request during 1024 ns. MAC starts transmitting LPI code. ... 31250: When MAC does not receive transmission request during 1 ms, MAC starts transmitting LPI code. ... (skip the rest) *Note*: This value is used by internal logic and does not mean indicating accurate time on PHY IF.

31.4.2.42 RGMII Low Power parameter register 2 (LPTXGTH2)

LPTXGTH2 sets the change timing to Low Power Idle Transmission Mode after enabled function of transmitting LPI code at 1Gbps.

[Write restrictions]

Write to this register when the CXR20_bit6 = ‘0’, CXR20_bit5 = ‘0’. Recommended value is H’0000_0000 (Initial value).



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_GSTH2	H’0000_0000	R/W	Threshold value of starting LPI code transmission standby status (1 Gbps) (Unit is 32 ns) 0: (Initial value) MAC will be ready for transmitting LPI code after 1.01 s later from LinkUp/LPM bit of CXR20 is set to ‘1’. 1 to 7: Setting prohibited ... 32: MAC will be ready for transmitting LPI code after 1024 ns later from LinkUp/LPM bit of CXR20 is set to ‘1’. ... 31250000: MAC will be ready for transmitting LPI code after 1 s later from LinkUp/LPM bit of CXR20 is set to ‘1’. ... 31562500: MAC will be ready for transmitting LPI code after 1.01 s later from LinkUp/LPM bit of CXR20 is set to ‘1’. ... 100000000: MAC will be ready for transmitting LPI code after 3.2 s later from LinkUp/LPM bit of CXR20 is set to ‘1’. ... (skip the rest) *Note*: This value is used by internal logic and does not mean indicating accurate time on PHY IF. Recommended setting value In the standard, the behavior of PHY is undefined if PHY receives LPI code within 1 s after Link Up. Therefore, the recommended value was determined as 1.01 s. (1 s plus 1% margin) The following is reason why “LPM bit of CXR20 is set to ‘1’” was include to the condition. Intenal Clock (32 ns:1 Gbps, 320 ns:100 Mbps) is used to measure the time after Link Up. But Internal Clock will change during transmission rate is un-determined. So, it is difficult to measure the time after Link Up exactly. Therefore, “LPM bit of CXR20 is set to ‘1’” was include to the condition as determined point of transmission rate of Low Power Mode.

31.4.2.43 RGMII Low Power parameter register 3 (LPTXGTH3)

LPTXGTH3 sets the operation return from Low Power Idle Transmission Mode at 1 Gbps.

[Write restrictions]

Write to this register when the CXR20_bit6 = ‘0’, CXR20_bit5 = ‘0’. Recommended value is H’0000_0000 (Initial value). Do not set 262144 (H’0004_0000) or more to this register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPTX_GRTH[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPTX_GRTH[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_GRTH	H'0000_0000	R/W	Transmission Low Power resume time (1 Gbps) (Unit 32 ns) 0: (Initial value) After received transmit request, MAC transmits Normal Idle code during 17.184 µs. And then MAC returns to Normal Mode from Low Power Mode. 1 to 7: Setting prohibited ... 32: After received transmit request, MAC transmits Normal Idle code during 1024 ns And then MAC returns to Normal Mode from Low Power Mode. ... 516: After received transmit request, MAC transmits Normal Idle code during 16.512 µs. And then MAC returns to Normal Mode from Low Power Mode. 521: After received transmit request, MAC transmits Normal Idle code during 16.672 µs. And then MAC returns to Normal Mode from Low Power Mode. 532: After received transmit request, MAC transmits Normal Idle code during 17.024 µs. And then MAC returns to Normal Mode from Low Power Mode. 537: After received transmit request, MAC transmits Normal Idle code during 17.184 µs. And then MAC returns to Normal Mode from Low Power Mode. ... 1000: After received transmit request, MAC transmits Normal Idle code during 17.184 µs. And then MAC returns to Normal Mode from Low Power Mode. ... 262144 or more: Reserved (Setting prohibited) (skip the rest) *Note*: This value is used by internal logic and does not mean indicating accurate time on PHY IF. Recommended setting value The basic value of this field is LPI WakeUp Time (Tw_sys_tx) with 1% margin. In the case of 1000BASE-T, Tw_sys_tx is 16.5 us. But some of PHY needs 1 us for Wakeup Time. Therefore, Recommended value was determined as 17.184 us. The setting value of this field will be the overhead time return from Low Power Mode. Do not set the value will be under the 16.5 µs (Minimum Wakeup Time which is defined by standard) to this field. In addition, if PHY has the LPI transmission control function, PHY also make overhead time same as Wake-Up Time.

31.4.2.44 RGMII Low Power parameter register 4 (LPTXGTH4)

LPTXGTH4 sets the minimum period for Low Power Idle Transmission Mode at 1 Gbps.

[Write restrictions]

Write to this register when the CXR20_bit6 = '0', CXR20_bit5 = '0'. Recommended value is H'0000_0000 (Initial value).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPTX_GCTH[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPTX_GCTH[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_GCTH	H'0000_0000	R/W	Minimum Transmission Low Power time (1 Gbps) (Unit is 32 ns) 0: (Initial value) After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 1024 ns. 1 to 7: Setting prohibited 8: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 256 ns. ... 32: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 1024 ns. ... 6469: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 207.008 µs (Sleep Time + α). ... 10000: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 320 µs. ... (skip the rest) *Note*: This value is used by internal logic and does not mean indicating accurate time on PHY IF. Recommended setting value Recommended value of this field is 1 us (1024 ns) for 1000BASE-T. This value based on followings: [1] This value should be sufficiency longer than 72 ns [2] This value should be sufficiency shorter than Wake Up Time (16.5 µs). And selected the value which is close to geometric mean of [1] and [2].

31.4.2.45 MII Low Power parameter register 1 (LPTXMTH1)

LPTXMTH1 sets the change timing to Low Power Idle Transmission Mode at 100 Mbps.

[Write restrictions]

Write to this register when the CXR20_bit6 = ‘0’, CXR20_bit5 = ‘0’. Recommended value is H’0000_0000 (Initial value).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPTX_MSTH1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPTX_MSTH1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_MSTH1	H’0000_0000	R/W	Threshold value of LPI code transmission standby status (100 Mbps) (Unit is 320 ns) 0: (Initial value) When MAC does not receive transmission request during 1 ms, MAC starts transmitting LPI code. 1 to 7: Setting prohibited 8: When MAC does not receive transmission request during 10240 ns, MAC starts transmitting LPI code. ... 3125: When MAC does not receive transmission request during 1 ms, MAC starts transmitting LPI code. ... (skip the rest) *Note*: This value is used by internal logic and does not mean indicating accurate time on PHY IF.

31.4.2.46 MII Low Power parameter register 2 (LPTXMTH2)

LPTXMTH2 sets the change timing to Low Power Idle Transmission Mode after enabled function of transmitting LPI code at 100 Mbps.

[Write restrictions]

Write to this register when the CXR20_bit6 = ‘0’, CXR20_bit5 = ‘0’. Recommended value is H’0000_0000 (Initial value).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPTX_MSTH2[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPTX_MSTH2[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

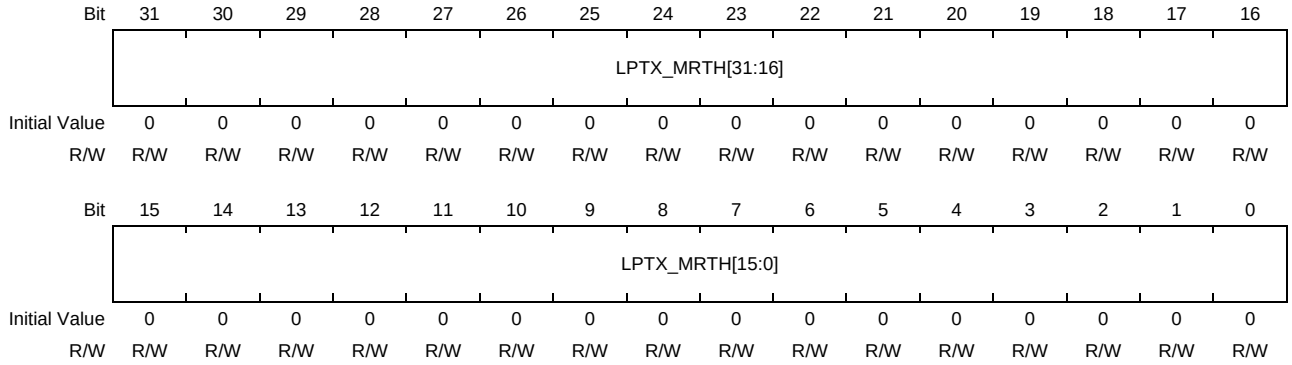
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_MSTH2	H’0000_0000	R/W	Threshold value of starting LPI code transmission standby status (100 Mbps) (Unit is 320 ns) 0: (Initial value) MAC will be ready for transmitting LPI code after 1.01 s later from Link Up or LPM bit of CXR20 is set to ‘1’. 1 to 7: Setting prohibited ... 32: MAC will be ready for transmitting LPI code after 10240 ns later from LinkUp/LPM bit of CXR20 is set to ‘1’. ... 3125000: MAC will be ready for transmitting LPI code after 1.0 s later from LinkUp/LPM bit of CXR20 is set to ‘1’. ... 3156250: MAC will be ready for transmitting LPI code after 1.01 s from LinkUp/LPM bit of CXR20 is set to ‘1’. ... 10000000: MAC will be ready for transmitting LPI code after 3.2 s later from LinkUp/LPM bit of CXR20 is set to ‘1’. ... (skip the rest) *Note*: This value is used by internal logic and does not mean indicating accurate time on PHY IF. Recommended setting value In the standard, the behavior of PHY is undefined if PHY receives LPI code within 1 s after Link Up. Therefore, the recommended value was determined as 1.01 s. (1 s plus 1% margin) The following is reason why “LPM bit of CXR20 is set to ‘1’” was include to the condition. Internal Clock (32 ns:1 Gbps, 320 ns:100 Mbps) is used to measure the time after Link Up. But Internal Clock will change during transmission rate is un-determined. So, it is difficult to measure the time after Link Up exactly. Therefore, “LPM bit of CXR20 is set to ‘1’” was include to the condition as determined point of transmission rate of Low Power Mode.

31.4.2.47 MII Low Power parameter register 3 (LPTXMTH3)

LPTXMTH3 sets the operation return from Low Power Idle Transmission Mode at 100 Mbps.

[Write restrictions]

Write to this register when the CXR20_bit6 = '0', CXR20_bit5 = '0'. Recommended value is H'0000_0000 (Initial value). Do not set 262144 (H'0004_0000) or more to this register.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_MRTH	H'0000_0000	R/W	Transmission Low Power resume time (100 Mbps) (Unit 320 ns) 0: (Initial value) After received transmit request, MAC transmits Normal Idle code during 30.40 µs. And then MAC returns to Normal Mode from Low Power Mode. 1 to 7: Setting prohibited ... 32: After received transmit request, MAC transmits Normal Idle code during 10240 ns And then MAC returns to Normal Mode from Low Power Mode. ... 94: After received transmit request, MAC transmits Normal Idle code during 30.08 µs And then MAC returns to Normal Mode from Low Power Mode. 95: After received transmit request, MAC transmits Normal Idle code during 30.40 µs And then MAC returns to Normal Mode from Low Power Mode. ... 262144 or more: Reserved (Setting prohibited) *Note*: This value is used by internal logic and does not mean indicating accurate time on PHY IF. Recommended setting value The basic value of this field is LPI WakeUp Time (Tw_sys_tx) with 1% margin. In the case of 100BASE-TX, Tw_sys_tx is 30 µs. Therefore, recommended value was determined as 30.4 µs. The setting value of this field will be the overhead time return from Low Power Mode. Do not set the value under the 30 µs (Minimum Wakeup Time which is defined by standard) to this field.

31.4.2.48 MII Low Power parameter register 4 (LPTXMTH4)

LPTXMTH4 sets the minimum period for Low Power Idle Transmission Mode at 100 Mbps

[Write restrictions]

Write to this register when the CXR20_bit6 = ‘0’, CXR20_bit5 = ‘0’. Recommended value is H’0000_0000 (Initial value).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPTX_MCTH[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPTX_MCTH[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_MCTH	H’0000_0000	R/W	Minimum Transmission Low Power time (100 Mbps) (Unit is 320 ns) 0: (Initial value) After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 3200 ns. 1 to 7: Setting prohibited 8: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 2560 ns. ... 10: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 3200 ns. ... 844: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 270.08 µs (Sleep Time+α). ... 1000: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 320 µs. ... (skip the rest) *Note*: This value is used by internal logic and doesn’t mean indicating accurate time on PHY IF. Recommended setting value Recommended value of this field is 3.2 µs (3200 ns) for 100BASE-TX. This value based on followings: [1] his value should be sufficiency longer than 360 ns [2] This value should be sufficiency shorter than Wake Up Time (30 µs). And selected the value which is close to geometric mean of [1] and [2].

31.4.3 TOE Registers

31.4.3.1 Checksum operating mode register (CSR0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FIFOCAP[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RBP	TBP	—	—	RPE	TPE	—	—	—	CCM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17 to 16	FIFOCAP	00b	R/W	FIFO CAPACITY TOE FIFO Capacity 00b: 2 Kbytes (512 Word × 32bit) 01b: 4 Kbytes (1024 Word × 32bit) 10b: 8 Kbytes (2048 Word × 32bit) 11b: 16 Kbytes (4096 Word × 32bit) Write to this field when TPE is ‘0’.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	RBP	0b	R/W	Receive Bypass Enable Bypass Enable of TOE Rx function. 0b: Rx Function is not bypassed. 1b: Rx Function is bypassed. (for debug) *Note*: This bit is used only for debug. Do not set 1b during normal operation. When this bit is set to “1", all Rx Functions of TOE are bypassed. Therefore, TOE sends Rx frame from E-MAC to DMAC even if RPE is ‘0’. Write to this bit when RPE is ‘0’. When this bit is set to “0", set E-MAC as CRC Pass Through Mode (CXR20 RCPT = “1”). When this bit is set to “1”, set E-MAC as Non-CRC Pass Through Mode (CXR20 RCPT = “0”).
8	TBP	0b	R/W	Transmit Bypass Enable Bypass Enable of TOE Tx Function 0b: Tx Function is not bypassed. 1b: Tx Function is bypassed (for debug) *Note*: This bit is used only for debug. Do not set 1b during normal operation. When this bit is set to “1", all Tx Functions of TOE are bypassed. Therefore, TOE sends Tx frame from DMAC to E-MAC even if TPE is ‘0’. Since Tx Frame does not go through TOE FIFO, there is no latency. Write to this bit when TPE is ‘0’.
7 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
5	RPE	0b	R/W	Receive Port Enable Enable of Frame Output Function to DMAC. 0b: Output Function is disabled. 1b: Output Function is enabled. When value is changed from ‘1’ to ‘0’ during TOE is outputting any frames, the value of this bit will be ‘0’ after finished outputting. Therefore, ‘1’ is read as read value during TOE is outputting frames.
4	TPE	0b	R/W	Transmit Port Enable Enable of Frame Output Function to E-MAC. 0b: Output Function is disable. 1b: Output Function is enabled. Procedure of setting ‘0’ to this bit. (1) Set ‘0’ to TPE. (2) Continue to read TPE until it will be read as ‘0’. (TPE will be ‘0’ when TOE FIFO will be empty) (3) If read value is ‘0’, procedure of setting ‘0’ is finished.
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CCM	0b	R/W	Counter Clear Mode This bit designates the clear method of Interrupt Status Counters (CSR20, CSR30, CSR31, CSR32) 1b: When each Counter Registers are read, they are cleared to ‘0’. 0b: When each Counter Registers are written any value, they are cleared to ‘0’. Write to this bit when TPE is ‘0’ and RPE is ‘0’.

31.4.3.2 Tx Frame Checksum Enable register (CSR1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	TDHD	TAHD	TROUT	THOP	—	TICMP6	TUDP6	TTCP6	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TICMP4	TUDP4	TTCP4	—	—	—	TIP4
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27	TDHD	0b	R/W	Transmit Destination Option Header Enable When the frame from DMAC has a Destination Option header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are ‘0’, the configuration of this bit is ignored. Write to this bit when TPE is ‘0’.
26	TAHD	0b	R/W	Transmit Authentication Header Enable When the frame from DMAC has an AH header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are ‘0’, the configuration of this bit is ignored. Write to this bit when TPE is ‘0’.
25	TROUT	0b	R/W	Transmit Routing Header Enable When the frame from DMAC has a Routing header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are ‘0’, the configuration of this bit is ignored. Write to this bit when TPE is ‘0’.
24	THOP	0b	R/W	Transmit Hop-by-Hop Option Header Enable When the frame from DMAC has a Hop-by-Hop Option header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are ‘0’, the configuration of this bit is ignored. Write to this bit when TPE is ‘0’.
23	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	TICMP6	0b	R/W	Transmit ICMP Checksum in IPv6 Enable Enable of IPv6 ICMP Checksum Calculation in the frame from DMAC. 0b: Disable 1b: Enable When this bit is ‘1’, the value of bits [27:24] are valid. Write to this bit when TPE is ‘0’.

Bit	Bit Name	Initial Value	R/W	Description
21	TUDP6	0b	R/W	Transmit UDP Checksum in IPv6 Enable Enable of IPv6 UDP Checksum Calculation in the frame from DMAC. 0b: Disable 1b: Enable When this bit is ‘1’, the value of bits [27:24] are valid. Write to this bit when TPE is ‘0’.
20	TTCP6	0b	R/W	Transmit TCP Checksum in IPv6 Enable Enable of IPv6 TCP Checksum Calculation in the frame from DMAC. 0b: Disable 1b: Enable When this bit is ‘1’, the value of bits [27:24] are valid. Write to this bit when TPE is ‘0’.
19 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	TICMP4	0b	R/W	Transmit ICMP Checksum in IPv4 Enable Enable of IPv4 ICMP Checksum Calculation in the frame from DMAC. 0b: Disable 1b: Enable Write to this bit when TPE is ‘0’.
5	TUDP4	0b	R/W	Transmit UDP Checksum in IPv4 Enable Enable of IPv4 UDP Checksum Calculation in the frame from DMAC. 0b: Disable 1b: Enable Write to this bit when TPE is ‘0’.
4	TTCP4	0b	R/W	Transmit TCP Checksum in IPv4 Enable Enable of IPv4 TCP Checksum Calculation in the frame from DMAC. 0b: Disable 1b: Enable Write to this bit when TPE is ‘0’.
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	TIP4	0b	R/W	Transmit IPv4 Checksum Enable Enable of IPv4 Header Checksum Calculation in the frame from DMAC. 0b: Disable 1b: Enable Write to this bit when TPE is ‘0’.

31.4.3.3 Rx Frame Checksum Enable register (CSR2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	RDHD	RAHD	RROUT	RHOP	—	RICMP6	RUDP6	RTCP6	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RICMP4	RUDP4	RTCP4	—	—	—	RIP4
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27	RDHD	0b	R/W	Receive Destination Option Header Enable When the frame from E-MAC has a Destination Option header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are ‘0’, the configuration of this bit is ignored. Write to this bit when RPE is ‘0’.
26	RAHD	0b	R/W	Receive Authentication Header Enable When the frame from E-MAC has an AH header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are ‘0’, the configuration of this bit is ignored. Write to this bit when RPE is ‘0’.
25	RROUT	0b	R/W	Receive Routing Header Enable When the frame from E-MAC has a Routing header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are ‘0’, the configuration of this bit is ignored. Write to this bit when RPE is ‘0’.
24	RHOP	0b	R/W	Receive Hop-by-Hop Option Header Enable When the frame from E-MAC has a Hop-by-Hop Option header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are ‘0’, the configuration of this bit is ignored. Write to this bit when RPE is ‘0’.
23	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	RICMP6	0b	R/W	Receive ICMP Checksum in IPv6 Enable Enable of IPv6 ICMP Checksum Calculation in the frame from E-MAC. 0b: Disable 1b: Enable When this bit is ‘1’, the value of bits [27:24] are valid. Write to this bit when RPE is ‘0’.

Bit	Bit Name	Initial Value	R/W	Description
21	RUDP6	0b	R/W	Receive UDP Checksum in IPv6 Enable Enable of IPv6 UDP Checksum Calculation in the frame from E-MAC. 0b: Disable 1b: Enable When this bit is '1', the value of bits [27:24] are valid. Write to this bit when RPE is '0'.
20	RTCP6	0b	R/W	Receive TCP Checksum in IPv6 Enable Enable of IPv6 TCP Checksum Calculation in the frame from E-MAC. 0b: Disable 1b: Enable When this bit is '1', the value of bits [27:24] are valid. Write to this bit when RPE is '0'.
19 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	RICMP4	0b	R/W	Receive ICMP Checksum in IPv4 Enable Enable of IPv4 ICMP Checksum Calculation in the frame from E-MAC. 0b: Disable 1b: Enable Write to this bit when RPE is '0'.
5	RUDP4	0b	R/W	Receive UDP Checksum in IPv4 Enable Enable of IPv4 UDP Checksum Calculation in the frame from E-MAC. 0b: Disable 1b: Enable Write to this bit when RPE is '0'.
4	RTCP4	0b	R/W	Receive TCP Checksum in IPv4 Enable Enable of IPv4 TCP Checksum Calculation in the frame from E-MAC. 0b: Disable 1b: Enable Write to this bit when RPE is '0'. These bits are read as 0. The write value should be always 0.
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RIP4	0b	R/W	Receive IPv4 Checksum Enable Enable of IPv4 Header Checksum Calculation in the frame from E-MAC. 0b: Disable 1b: Enable Write to this bit when RPE is '0'.

31.4.3.4 Tx Extended Header Number register (CSR3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	TNUEXHED[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	TNUEXHED	H’8	R/W	Transmit Support Number of Extension Header Transmit Support Number of Extension Header in the frame from DMAC H’0: 16 H’1: 1 … H’E: 14 H’F: 15 If the number of Extension Header in the IPv6 frame is over the configuration of this field, TOE ascribes this frame to unsupported. Write to this field when TPE is ‘0’.

31.4.3.5 Rx Extended Header Number register (CSR4)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	RNUEXHED[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	RNUEXHED	H’8	R/W	Receive Support Number of Extension Header Receive Support Number of Extension Header in the frame from E-MAC. H’0: 16 H’1: 1 … H’E: 14 H’F: 15 If the number of Extension Header in the IPv6 frame is over the configuration of this field, TOE ascribes this frame to unsupported. Write to this field when RPE is ‘0’.

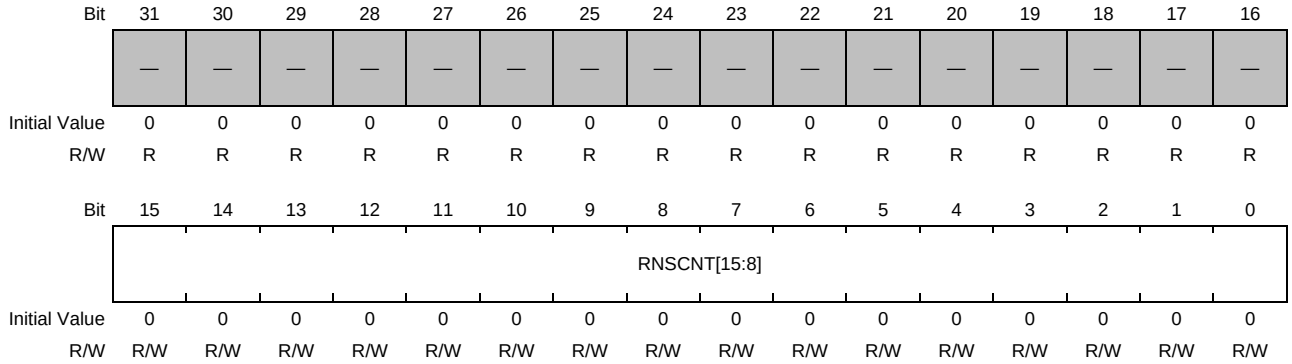
31.4.3.6 Unsupported Tx Frame Counter register (CSR20)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TNSCNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	TNSCNT	H'0000	R/W	Transmit Not Support Frame Counter Counter for Transmitted Unsupported Frames When CSR0_bit0 is '1', this register is cleared to '0' by read access. When CSR0_bit0 is '0', this register is cleared to '0' by write access of any data. The condition of counting up is OR of followings. • TYPE/TPID is Unsupported • Not ("MF = 0" and "Fragment Offset = H'0000") • Protocol is Unsupported • Next Header of IPv6 frame is Unsupported • The number of Next Header in IPv6 frame is 9 or more • Checksum calculation for a particular frame is disabled by CSR1 • The Frame Length is shorter than the result of Frame Analysis by TOE.

31.4.3.7 Unsupported Rx Frame Counter register (CSR30)



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RNSCNT	H'0000	R/W	Receive Not Support Frame Counter Counter for Received Unsupported Frames When CSR0_bit0 is '1', this register is cleared to '0' by read access. When CSR0_bit0 is '0', this register is cleared to '0' by write access of any data. The condition of counting up is OR of followings. • TYPE/TPID is Unsupported • Not ("MF = 0" and "Fragment Offset = H'0000") • Protocol is Unsupported • Next Header of IPv6 frame is Unsupported • The number of Next Header in IPv6 frame is 9 or more • Checksum calculation for a particular frame is disabled by CSR2 • The Frame Length is shorter than the result of Frame Analysis by TOE.

31.4.3.8 Rx IPv4 Header checksum Error Counter register (CSR31)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R4ECNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	R4ECNT	H'0000	R/W	Receive IPv4 Header Checksum Error Counter Counter for the number of IPv4 Header Checksum Error in the frames from E-MAC. When CSR0_bit0 is '1', this register is cleared to '0' by read access. When CSR0_bit0 is '0', this register is cleared to '0' by write access of any data. The condition of counting up is following. When the frame which has IPv4 Checksum Error comes, TOE counts up this register.

31.4.3.9 Rx TCP/UDP/ICMP checksum Error Counter register (CSR32)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RTUECNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RTUECNT	H'0000	R/W	Receive TCP/UDP/ICMP Checksum Error Counter Counter for the number of TCP/UDP/ICMP Checksum Error in the frames from E-MAC. When CSR0_bit0 is '1', this register is cleared to '0' by read access. When CSR0_bit0 is '0', this register is cleared to '0' by write access of any data. The condition of counting up is following. When the frame which has TCP/UDP/ICMP Checksum Error comes, TOE counts up this register.

31.4.3.10 Filter Operation/Auto Response Enable register (CSFR00)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	arp_ns_abt_en	—	—	reply_fifo_en	reg_fifo_en	true_false	ieee_len_fen	ana_stp_fen	v6_ana_protocol_fen	ns_dis_fen	ns_dup_fen	ns_uni_nopt_fen	ns_uni_fen
Initial Value	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

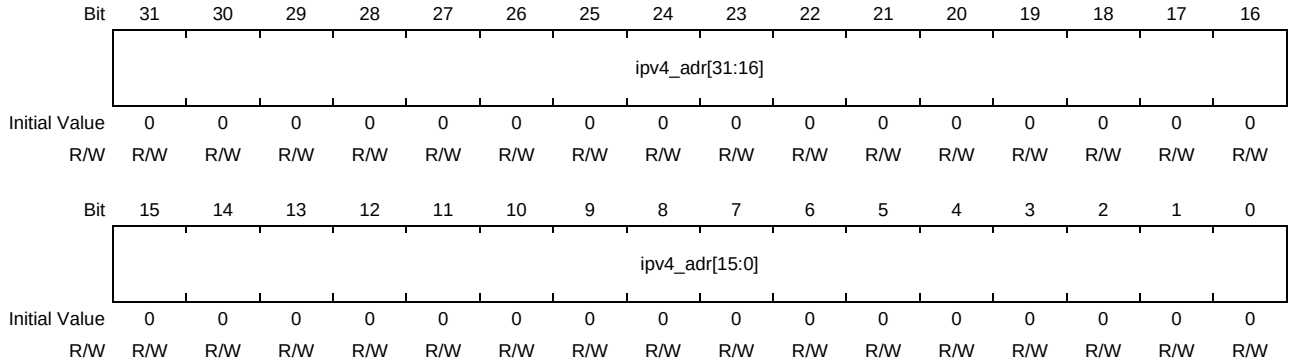
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ns_mul_fen	arp_noreq_fen	garp_self_fen	garp_other_fen	arp_self_fen	arp_other_fen	mac_da_mul_fen	mac_da_bro_fen	mac_da_uni_fen	v6_udp_pt_fen	v4_udp_pt_fen	v6_protocol_fen	v4_protocol_fen	type_fen	sby_mode[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	arp_ns_abt_en	1b	R/W	ARP ns frame Abort Enable Configuration of Transferred to DMAC or Discarded for the frame which was stored to REPLY FIFO of REG FIFO. 0b: Transferred to DMAC 1b: Discarded (Aborted)
27 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	reply_fifo_en	0b	R/W	Reply FIFO Enable Enable of Auto Response Function for ARP REQ and Neighbor Solicitations 0b: Disable 1b: Enable
24	reg_fifo_en	0b	R/W	Reg FIFO Enable Enable of Interrupt which occurs by reception of ARP REQ and Neighbor Solicitations. 0b: Disable 1b: Enable
23	true_false	0b	R/W	True False mode Discarding condition of the frame by filter. 0b: Discard by unmatch to filter condition 1b: Discard by match to filter condition
22	ieee_len_fen	0b	R/W	IEEE Length Filter Enable Filtering Operation for the IEEE802.3 Length Frame. 0b: Non-Filtering 1b: Filtering
21	ana_stp_fen	1b	R/W	Analysis stop Filter Enable Filtering Operation for the case of stopping frame analysis due to impossible analysis. 0b: Non-Filtering 1b: Filtering
20	v6_ana_protocol_fen	0b	R/W	IPv6 Analysis protocol Filter Enable Filtering Operation for the case of stopping frame analysis because Protocol No is not match to configurated value by v6_ana_protocol_0-15. 0b: Non-Filtering 1b: Filtering
19	ns_dis_fen	0b	R/W	Neighbor Solicitation discarded Filter Enable Filtering Operation for the invalid Neighbor Solicitations. 0b: Non-Filtering 1b: Filtering

Bit	Bit Name	Initial Value	R/W	Description
18	ns_dup_fen	0b	R/W	Neighbor Solicitation Duplicate Filter Enable Filtering Operation for the IP Duplicate Detection Neighbor Solicitations of Local Station. 0b: Non-Filtering 1b: Filtering
17	ns_uni_nopt_fen	0b	R/W	Neighbor Solicitation Unicast no option Filter Enable Filtering Operation for the Unicast Neighbor Solicitations of Local Station. (Except “Option Type=1” or No “Option Type”) 0b: Non-Filtering 1b: Filtering
16	ns_uni_fen	0b	R/W	Neighbor Solicitation Unicast Filter Enable Filtering Operation for the Unicast Neighbor Solicitations of Local Station. (Option Type=1) 0b: Non-Filtering 1b: Filtering
15	ns_mul_fen	0b	R/W	Neighbor Solicitation Multicast Filter Enable Filtering Operation for the Multicast Neighbor Solicitations of Local Station. 0b: Non-Filtering 1b: Filtering
14	arp_noreq_fen	0b	R/W	ARP No request Filer Enable Filtering Operation for the ARP frame except ARP request. 0b: Non-Filtering 1b: Filtering
13	garp_self_fen	0b	R/W	GARP Self Filter Enable Filtering Operation for the GARP request of Local Station. 0b: Non-Filtering 1b: Filtering
12	garp_other_fen	0b	R/W	GARP Other Filter Enable Filtering Operation for the GARP request of Other Station. 0b: Non-Filtering 1b: Filtering
11	arp_self_fen	0b	R/W	ARP Self Filter Enable Filtering Operation for the ARP request of Local Station. 0b: Non-Filtering 1b: Filtering
10	arp_other_fen	0b	R/W	ARP Other Filter Enable Filtering Operation for the ARP request of Other Station. 0b: Non-Filtering 1b: Filtering
9	mac_da_mul_fen	0b	R/W	MAC DA Multicast Filter Enable Filtering Operation by MAC DA. (multicast) 0b: Non-Filtering 1b: Filtering
8	mac_da_bro_fen	0b	R/W	MAC DA Broadcast Filter Enable Filtering Operation by MAC DA. (broadcast) 0b: Non-Filtering 1b: Filtering
7	mac_da_uni_fen	0b	R/W	MAC DA Unicast Filter Enable Filtering Operation by MAC DA. (unicast) 0b: Non-Filtering 1b: Filtering

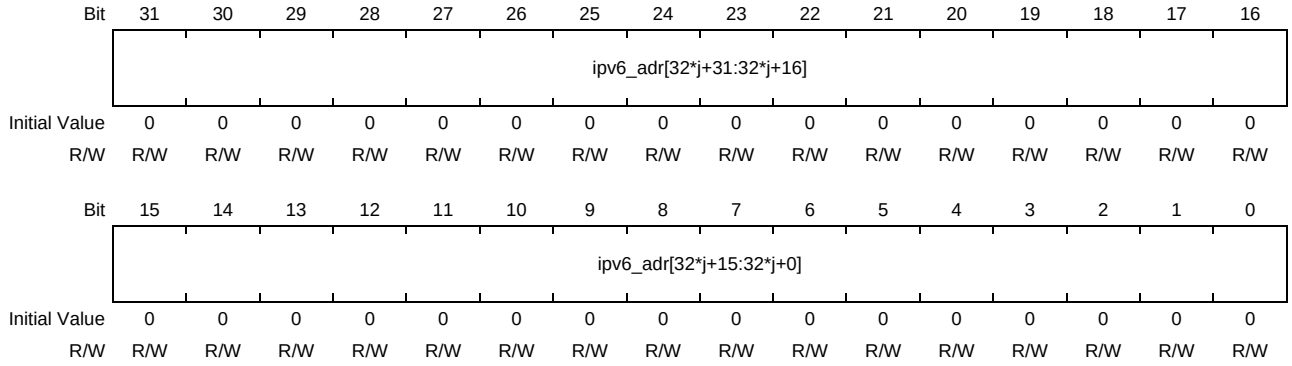
Bit	Bit Name	Initial Value	R/W	Description
6	v6_udp_pt_fen	0b	R/W	IPv6 UDP Port Filter Enable Filtering Operation by IPv6 UDP port number. (destination) 0b: Non-Filtering 1b: Filtering
5	v4_udp_pt_fen	0b	R/W	IPv4 UDP Port Filter Enable Filtering Operation by IPv4 UDP port number. (destination) 0b: Non-Filtering 1b: Filtering
4	v6_protocol_fen	0b	R/W	IPv6 protocol Filter Enable Filtering Operation by IPv6 Protocol number. 0b: Non-Filtering 1b: Filtering
3	v4_protocol_fen	0b	R/W	IPv4 protocol Filter Enable Filtering Operation by IPv4 Protocol number. 0b: Non-Filtering 1b: Filtering
2	type_fen	0b	R/W	Type Filter Enable Filtering Operation by Ethernet TYPE/TPID. 0b: Non-Filtering 1b: Filtering
1 to 0	sby_mode	00b	R/W	Standby Mode Filtering Operation Mode 00b: Non-Filtering 1*b: Filtering

31.4.3.11 Local IPv4 address register (CSFR01)



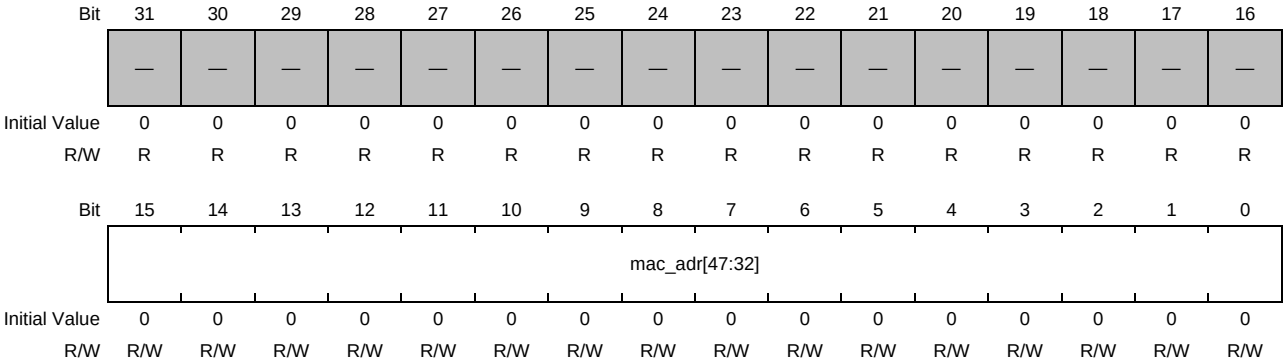
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ipv4_adr	H'0000_0000	R/W	IPv4 address IPv4 Address of Local Station Write to this field when CSFR00.reply_fifo_en = "0", reg_fifo_en = "0" and sby_mode = "00b".

31.4.3.12 Local IPv6 address register i (i = 0 to 3) (CSFR02_i)



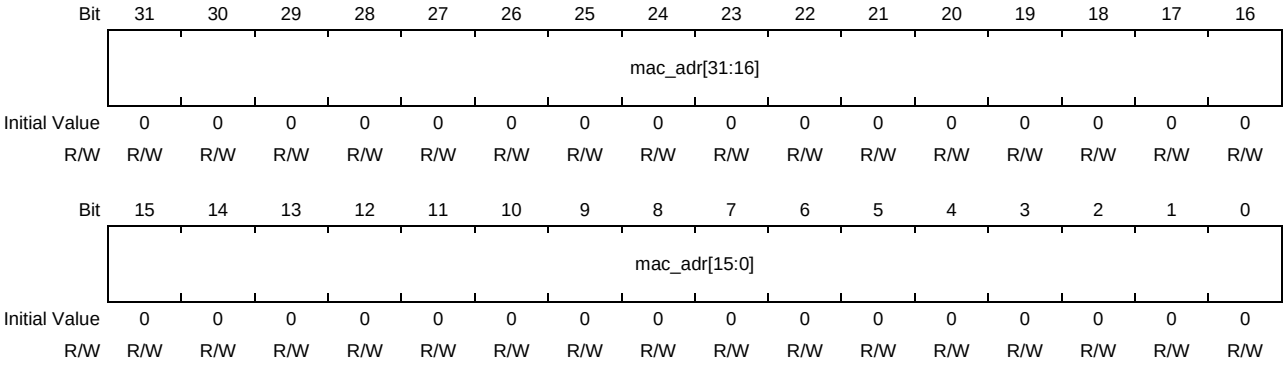
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ipv6_adr	H'0000_0000	R/W	IPv6 Address IPv6 Address[32*j+31:32*j] of Local Station The relationship between 'i' and 'j' is the following. i = 0 -> j = 3 i = 1 -> j = 2 i = 2 -> j = 1 i = 3 -> j = 0 Write to this field when CSFR00.reply_fifo_en = "0", reg_fifo_en = "0" and sby_mode = "00b".

31.4.3.13 Upper Local MAC address register (CSFR03_U)



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	mac_adr	H'0000	R/W	MAC Address MAC Address [47:32] of Local Station Write to this field when CSFR00.reply_fifo_en = "0", reg_fifo_en = "0" and sby_mode = "00b".

31.4.3.14 Lower Local MAC address register (CSFR03_L)



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	mac_adr	H'0000_0000	R/W	MAC Address MAC Address [32:0] of Local Station Write to this field when CSFR00.reply_fifo_en = "0", reg_fifo_en = "0" and sby_mode = "00b".

31.4.3.15 IPv6 no_next_header Protocol Number register (CSFR04)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	no_next_header[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	no_next_header	H'3B	R/W	Protocol Number of IPv6 no_next_header Usually, do not need to change from Initial values (59). Write to this field when CSFR00.sby mode is '00b'.

31.4.3.16 Ethernet Type Condition register i (i = 0 to 3) (CSFR10_i)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	type_2*i+1[15:8]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	type_2*I[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	type_2*i+1	H'0000	R/W	Ethernet Type 2*i+1 Configuration of the Ethernet TYPE for Filter Condition and Interrupt Condition. Write to this field when CSFR10. type_2*i+1_en is "0".
15 to 0	type_2*i	H'0000	R/W	Ethernet Type 2*i Configuration of the Ethernet TYPE for Filter Condition and Interrupt Condition. Write to this field when CSFR10. type_2*i_en is "0".

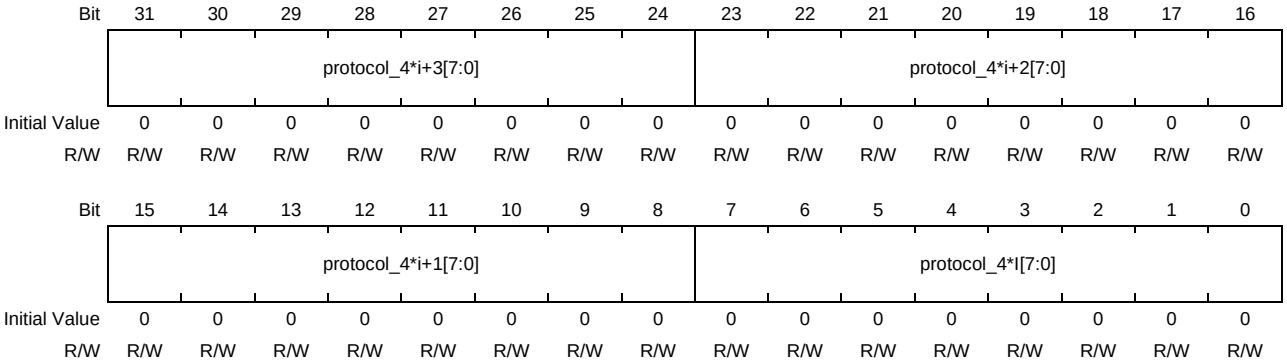
31.4.3.17 Ethernet Type Condition Enable register (CSFR10)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	type_7_en	type_6_en	type_5_en	type_4_en	type_3_en	type_2_en	type_1_en	type_0_en
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	type_7_en	0b	R/W	type_7 enable Enable for the value of type_7 which is configured in CSFR10_i. 0b: Disable 1b: Enable
6	type_6_en	0b	R/W	type_6 enable Enable for the value of type_6 which is configured in CSFR10_i. 0b: Disable 1b: Enable
5	type_5_en	0b	R/W	type_5 enable Enable for the value of type_5 which is configured in CSFR10_i. 0b: Disable 1b: Enable
4	type_4_en	0b	R/W	type_4 enable Enable for the value of type_4 which is configured in CSFR10_i. 0b: Disable 1b: Enable
3	type_3_en	0b	R/W	type_3 enable Enable for the value of type_3 which is configured in CSFR10_i. 0b: Disable 1b: Enable
2	type_2_en	0b	R/W	type_2 enable Enable for the value of type_2 which is configured in CSFR10_i. 0b: Disable 1b: Enable
1	type_1_en	0b	R/W	type_1 enable Enable for the value of type_1 which is configured in CSFR10_i. 0b: Disable 1b: Enable
0	type_0_en	0b	R/W	type_0 enable Enable for the value of type_0 which is configured in CSFR10_i. 0b: Disable 1b: Enable

31.4.3.18 Protocol Condition register i (i = 0 to 3) (CSFR11_i)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	protocol_4*i+3	H'00	R/W	protocol No 4*i+3 Configuration of the Protocol No for Filter Condition and Interrupt Condition. Write to this field when CSFR11. protocol_4*i+3_en is "0".
23 to 16	protocol_4*i+2	H'00	R/W	protocol No 4*i+2 Configuration of the Protocol No for Filter Condition and Interrupt Condition. Write to this field when CSFR11. protocol_4*i+2_en is "0".
15 to 8	protocol_4*i+1	H'00	R/W	protocol No 4*i+1 Configuration of the Protocol No for Filter Condition and Interrupt Condition. Write to this field when CSFR11. protocol_4*i+1_en is "0".
7 to 0	protocol_4*i	H'00	R/W	protocol No 4*i Write to this field when CSFR11. protocol_4*i_en is "0".

31.4.3.19 Protocol Condition Enable register (CSFR11)

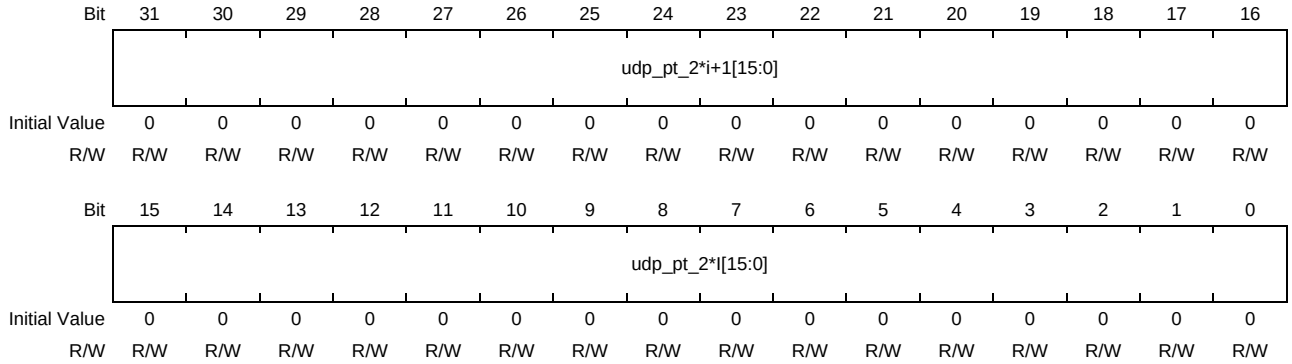
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	protocol_15_en	protocol_14_en	protocol_13_en	protocol_12_en	protocol_11_en	protocol_10_en	protocol_9_en	protocol_8_en	protocol_7_en	protocol_6_en	protocol_5_en	protocol_4_en	protocol_3_en	protocol_2_en	protocol_1_en	protocol_0_en
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15	protocol_15_en	0b	R/W	protocol_15 enable Enable for the value of protocol_15 which is configured in CSFR11_i. 0b: Disable 1b: Enable
14	protocol_14_en	0b	R/W	protocol_14 enable Enable for the value of protocol_14 which is configured in CSFR11_i. 0b: Disable 1b: Enable
13	protocol_13_en	0b	R/W	protocol_13 enable Enable for the value of protocol_13 which is configured in CSFR11_i. 0b: Disable 1b: Enable
12	protocol_12_en	0b	R/W	protocol_12 enable Enable for the value of protocol_12 which is configured in CSFR11_i. 0b: Disable 1b: Enable
11	protocol_11_en	0b	R/W	protocol_11 enable Enable for the value of protocol_11 which is configured in CSFR11_i. 0b: Disable 1b: Enable
10	protocol_10_en	0b	R/W	protocol_10 enable Enable for the value of protocol_10 which is configured in CSFR11_i. 0b: Disable 1b: Enable
9	protocol_9_en	0b	R/W	protocol_9 enable Enable for the value of protocol_9 which is configured in CSFR11_i. 0b: Disable 1b: Enable
8	protocol_8_en	0b	R/W	protocol_8 enable Enable for the value of protocol_8 which is configured in CSFR11_i. 0b: Disable 1b: Enable
7	protocol_7_en	0b	R/W	protocol_7 enable Enable for the value of protocol_7 which is configured in CSFR11_i. 0b: Disable 1b: Enable

Bit	Bit Name	Initial Value	R/W	Description
6	protocol_6_en	0b	R/W	protocol_6 enable Enable for the value of protocol_6 which is configured in CSFR11_i. 0b: Disable 1b: Enable
5	protocol_5_en	0b	R/W	protocol_5 enable Enable for the value of protocol_5 which is configured in CSFR11_i. 0b: Disable 1b: Enable
4	protocol_4_en	0b	R/W	protocol_4 enable Enable for the value of protocol_4 which is configured in CSFR11_i. 0b: Disable 1b: Enable
3	protocol_3_en	0b	R/W	protocol_3 enable Enable for the value of protocol_3 which is configured in CSFR11_i. 0b: Disable 1b: Enable
2	protocol_2_en	0b	R/W	protocol_2 enable Enable for the value of protocol_2 which is configured in CSFR11_i. 0b: Disable 1b: Enable
1	protocol_1_en	0b	R/W	protocol_1 enable Enable for the value of protocol_1 which is configured in CSFR11_i. 0b: Disable 1b: Enable
0	protocol_0_en	0b	R/W	protocol_0 enable Enable for the value of protocol_0 which is configured in CSFR11_i. 0b: Disable 1b: Enable

31.4.3.20 UDP Por Condition register i (i = 0 to 11) (CSFR12_i)



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	udp_pt_2*i+1	H'0000	R/W	UDP port No. Configuration of the UDP Port No for Filter Condition and Interrupt Condition. Write to this field when CSFR12. udp_pt_2*i+1_en is "0".
15 to 0	udp_pt_2*i	H'0000	R/W	UDP port No. Configuration of the UDP Port No for Filter Condition and Interrupt Condition. Write to this field when CSFR12. udp_pt_2*i_en is "0".

31.4.3.21 UDP Port Condition Enable register (CSFR12)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	udp_pt_23_en	udp_pt_22_en	udp_pt_21_en	udp_pt_20_en	udp_pt_19_en	udp_pt_18_en	udp_pt_17_en	udp_pt_16_en
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	udp_pt_15_en	udp_pt_14_en	udp_pt_13_en	udp_pt_12_en	udp_pt_11_en	udp_pt_10_en	udp_pt_9_en	udp_pt_8_en	udp_pt_7_en	udp_pt_6_en	udp_pt_5_en	udp_pt_4_en	udp_pt_3_en	udp_pt_2_en	udp_pt_1_en	udp_pt_0_en
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	udp_pt_23_en	0b	R/W	udp_pt_23 enable Enable for the value of udp_pt_23 which is configured in CSFR12_i. 0b: Disable 1b: Enable
22	udp_pt_22_en	0b	R/W	udp_pt_22 enable Enable for the value of udp_pt_22 which is configured in CSFR12_i. 0b: Disable 1b: Enable
21	udp_pt_21_en	0b	R/W	udp_pt_21 enable Enable for the value of udp_pt_21 which is configured in CSFR12_i. 0b: Disable 1b: Enable
20	udp_pt_20_en	0b	R/W	udp_pt_20 enable Enable for the value of udp_pt_20 which is configured in CSFR12_i. 0b: Disable 1b: Enable
19	udp_pt_19_en	0b	R/W	udp_pt_19 enable Enable for the value of udp_pt_19 which is configured in CSFR12_i. 0b: Disable 1b: Enable
18	udp_pt_18_en	0b	R/W	udp_pt_18 enable Enable for the value of udp_pt_18 which is configured in CSFR12_i. 0b: Disable 1b: Enable
17	udp_pt_17_en	0b	R/W	udp_pt_17 enable Enable for the value of udp_pt_17 which is configured in CSFR12_i. 0b: Disable 1b: Enable
16	udp_pt_16_en	0b	R/W	udp_pt_16 enable Enable for the value of udp_pt_16 which is configured in CSFR12_i. 0b: Disable 1b: Enable
15	udp_pt_15_en	0b	R/W	udp_pt_15 enable Enable for the value of udp_pt_15 which is configured in CSFR12_i. 0b: Disable 1b: Enable

Bit	Bit Name	Initial Value	R/W	Description
14	udp_pt_14_en	0b	R/W	udp_pt_14 enable Enable for the value of udp_pt_14 which is configured in CSFR12_i. 0b: Disable 1b: Enable
13	udp_pt_13_en	0b	R/W	udp_pt_13 enable Enable for the value of udp_pt_13 which is configured in CSFR12_i. 0b: Disable 1b: Enable
12	udp_pt_12_en	0b	R/W	udp_pt_12 enable Enable for the value of udp_pt_12 which is configured in CSFR12_i. 0b: Disable 1b: Enable
11	udp_pt_11_en	0b	R/W	udp_pt_11 enable Enable for the value of udp_pt_11 which is configured in CSFR12_i. 0b: Disable 1b: Enable
10	udp_pt_10_en	0b	R/W	udp_pt_10 enable Enable for the value of udp_pt_10 which is configured in CSFR12_i. 0b: Disable 1b: Enable
9	udp_pt_9_en	0b	R/W	udp_pt_9 enable Enable for the value of udp_pt_9 which is configured in CSFR12_i. 0b: Disable 1b: Enable
8	udp_pt_8_en	0b	R/W	udp_pt_8 enable Enable for the value of udp_pt_8 which is configured in CSFR12_i. 0b: Disable 1b: Enable
7	udp_pt_7_en	0b	R/W	udp_pt_7 enable Enable for the value of udp_pt_7 which is configured in CSFR12_i. 0b: Disable 1b: Enable
6	udp_pt_6_en	0b	R/W	udp_pt_6 enable Enable for the value of udp_pt_6 which is configured in CSFR12_i. 0b: Disable 1b: Enable
5	udp_pt_5_en	0b	R/W	udp_pt_5 enable Enable for the value of udp_pt_5 which is configured in CSFR12_i. 0b: Disable 1b: Enable
4	udp_pt_4_en	0b	R/W	udp_pt_4 enable Enable for the value of udp_pt_4 which is configured in CSFR12_i. 0b: Disable 1b: Enable
3	udp_pt_3_en	0b	R/W	udp_pt_3 enable Enable for the value of udp_pt_3 which is configured in CSFR12_i. 0b: Disable 1b: Enable
2	udp_pt_2_en	0b	R/W	udp_pt_2 enable Enable for the value of udp_pt_2 which is configured in CSFR12_i. 0b: Disable 1b: Enable

Bit	Bit Name	Initial Value	R/W	Description
1	udp_pt_1_en	0b	R/W	udp_pt_1 enable Enable for the value of udp_pt_1 which is configured in CSFR12_i. 0b: Disable 1b: Enable
0	udp_pt_0_en	0b	R/W	udp_pt_0 enable Enable for the value of udp_pt_0 which is configured in CSFR12_i. 0b: Disable 1b: Enable

31.4.3.22 Upper MAC DA unicast address Condition register (CSFR13_U)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	mac_da_uni[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	mac_da_uni	H'0000	R/W	MAC DA unicast address Configuration of the MAC DA unicast Address [47:32] for Filter Condition and Interrupt Condition. Write to this field when CSFR15.mac_da_uni_en = "0".

31.4.3.23 Lower MAC DA unicast address Condition register (CSFR13_L)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	mac_da_uni[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	mac_da_uni[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	mac_da_uni	H'0000_0000	R/W	MAC DA Unicast address Configuration of the MAC DA unicast Address [31:0] for Filter Condition and Interrupt Condition. Write to this field when CSFR15.mac_da_uni_en = "0".

31.4.3.24 Upper MAC DA broadcast address Condition register (CSFR14_U)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	mac_da_bro[47:40]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	mac_da_bro	H'FFFF	R/W	MAC DA Broadcast address Configuration of the MAC DA broadcast Address [47:32] for Filter Condition and Interrupt Condition. Write to this field when CSFR15.mac_da_bro_en = "0".

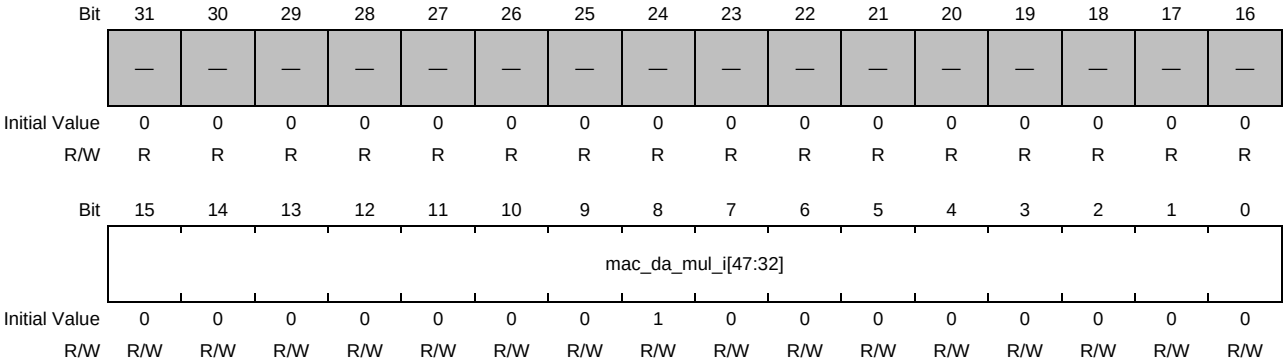
31.4.3.25 Lower MAC DA broadcast address Condition register (CSFR14_L)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	mac_da_bro[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	mac_da_bro[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

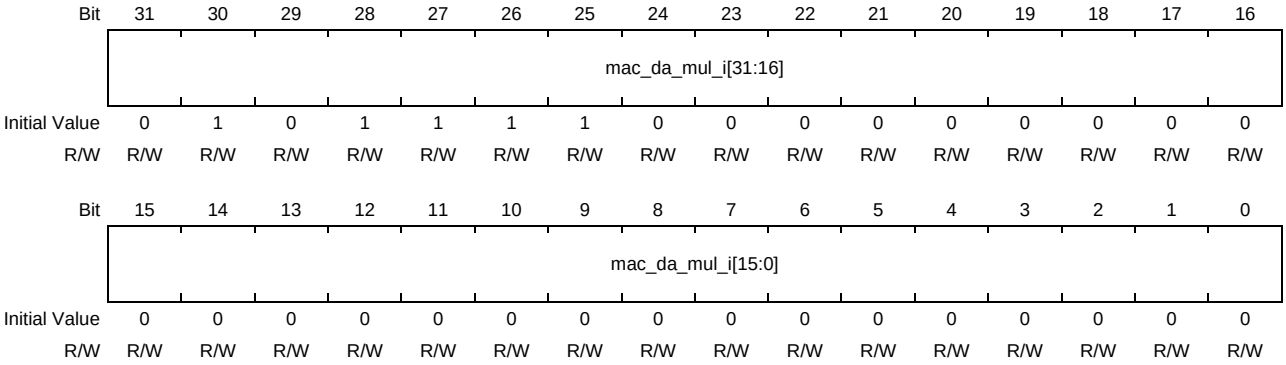
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	mac_da_bro	H'FFFF_FFFF	R/W	MAC DA Broadcast address Configuration of the MAC DA broadcast Address [31:0] for Filter Condition and Interrupt Condition. Write to this field when CSFR15.mac_da_bro_en = "0".

31.4.3.26 Upper MAC DA multicast address Condition register i (i = 0 to 19) (CSFR15_U_i)



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	mac_da_mul_i	H'0100	R/W	MAC DA Multicast address Configuration of the MAC DA multicast Address [47:32] for Filter Condition and Interrupt Condition. Write to this field when CSFR15.mac_da_mul_i_en = "0".

31.4.3.27 Lower MAC DA multicast address Condition register i (i = 0 to 19) (CSFR15_L_i)



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	mac_da_mul_i	H'5E00_0000	R/W	MAC DA Multicast address Configuration of the MAC DA multicast Address [31:0] for Filter Condition and Interrupt Condition. Write to this field when CSFR15.mac_da_mul_i _en = "0".

31.4.3.28 MAC DA Condition Enable register (CSFR15)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	mac_da_uni_en	—	—	—	mac_da_bro_en	—	—	—	—	mac_da_mul_19_en	mac_da_mul_18_en	mac_da_mul_17_en	mac_da_mul_16_en
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

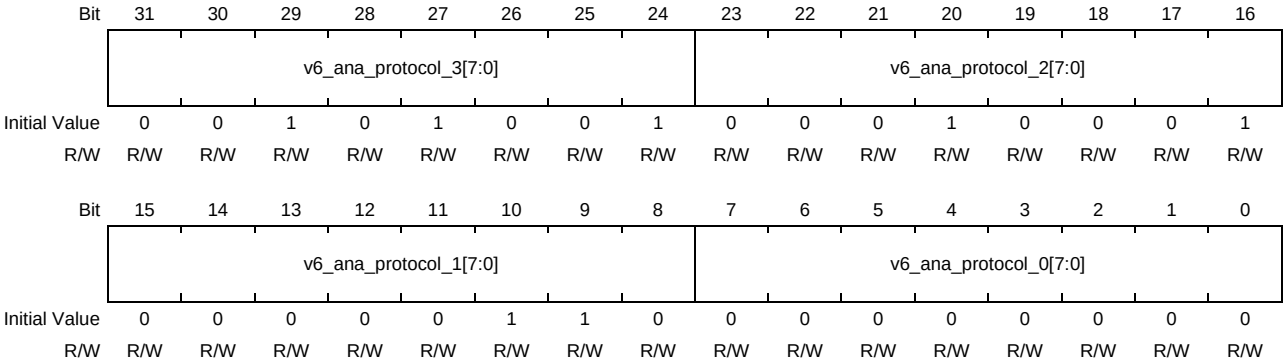
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	mac_da_mul_15_en	mac_da_mul_14_en	mac_da_mul_13_en	mac_da_mul_12_en	mac_da_mul_11_en	mac_da_mul_10_en	mac_da_mul_9_en	mac_da_mul_8_en	mac_da_mul_7_en	mac_da_mul_6_en	mac_da_mul_5_en	mac_da_mul_4_en	mac_da_mul_3_en	mac_da_mul_2_en	mac_da_mul_1_en	mac_da_mul_0_en
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	mac_da_uni_en	0b	R/W	Mac_da_uni enable Enable for the value of mac_da_uni which is configured in CSFR13_U/L. 0b: Disable 1b: Enable
27 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	mac_da_bro_en	0b	R/W	Mac_da_bro enable Enable for the value of mac_da_bro which is configured in CSFR14_U/L. 0b: Disable 1b: Enable
23 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	mac_da_mul_19_en	0b	R/W	mac_da_mul_19 enable Enable for the value of mac_da_mul_19 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
18	mac_da_mul_18_en	0b	R/W	mac_da_mul_18 enable Enable for the value of mac_da_mul_18 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
17	mac_da_mul_17_en	0b	R/W	mac_da_mul_17 enable Enable for the value of mac_da_mul_17 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
16	mac_da_mul_16_en	0b	R/W	mac_da_mul_16 enable Enable for the value of mac_da_mul_16 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
15	mac_da_mul_15_en	0b	R/W	mac_da_mul_15 enable Enable for the value of mac_da_mul_15 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable

Bit	Bit Name	Initial Value	R/W	Description
14	mac_da_mul_14_en	0b	R/W	mac_da_mul_14 enable Enable for the value of mac_da_mul_14 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
13	mac_da_mul_13_en	0b	R/W	mac_da_mul_13 enable Enable for the value of mac_da_mul_13 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
12	mac_da_mul_12_en	0b	R/W	mac_da_mul_12 enable Enable for the value of mac_da_mul_12 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
11	mac_da_mul_11_en	0b	R/W	mac_da_mul_11 enable Enable for the value of mac_da_mul_11 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
10	mac_da_mul_10_en	0b	R/W	mac_da_mul_10 enable Enable for the value of mac_da_mul_10 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
9	mac_da_mul_9_en	0b	R/W	mac_da_mul_9 enable Enable for the value of mac_da_mul_9 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
8	mac_da_mul_8_en	0b	R/W	mac_da_mul_8 enable Enable for the value of mac_da_mul_8 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
7	mac_da_mul_7_en	0b	R/W	mac_da_mul_7 enable Enable for the value of mac_da_mul_7 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
6	mac_da_mul_6_en	0b	R/W	mac_da_mul_6 enable Enable for the value of mac_da_mul_6 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
5	mac_da_mul_5_en	0b	R/W	mac_da_mul_5 enable Enable for the value of mac_da_mul_5 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
4	mac_da_mul_4_en	0b	R/W	mac_da_mul_4 enable Enable for the value of mac_da_mul_4 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
3	mac_da_mul_3_en	0b	R/W	mac_da_mul_3 enable Enable for the value of mac_da_mul_3 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
2	mac_da_mul_2_en	0b	R/W	mac_da_mul_2 enable Enable for the value of mac_da_mul_2 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable

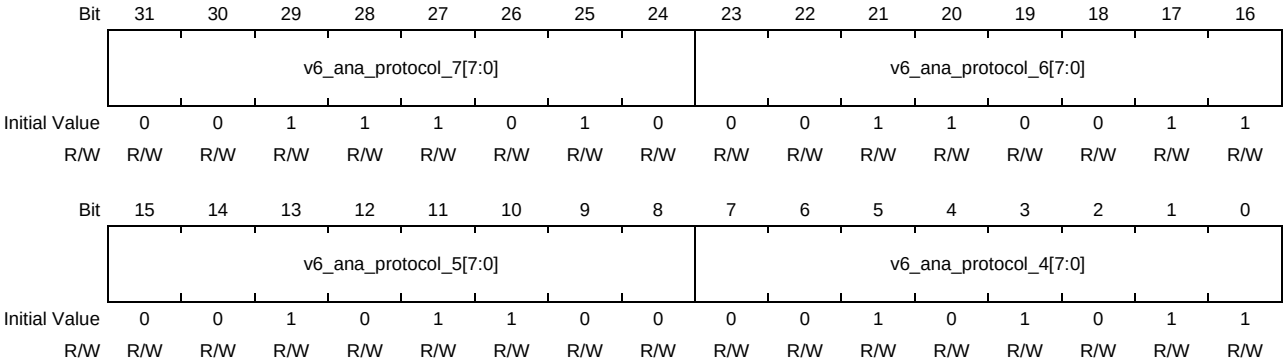
Bit	Bit Name	Initial Value	R/W	Description
1	mac_da_mul_1_en	0b	R/W	mac_da_mul_1 enable Enable for the value of mac_da_mul_1 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
0	mac_da_mul_0_en	0b	R/W	mac_da_mul_0 enable Enable for the value of mac_da_mul_0 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable

31.4.3.29 IPv6 analysis Protocol Condition register 0 (CSFR16_0)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	v6_ana_protocol_3	H'29	R/W	IPv6 Analyzable Protocol No 3 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'29 IPV6 (41). Write to this field when CSFR16. v6_ana_protocol_3_en is '0b'.
23 to 16	v6_ana_protocol_2	H'11	R/W	IPv6 Analyzable Protocol No 2 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'11 UDP (17). Write to this field when CSFR16. v6_ana_protocol_2_en is '0b'.
15 to 8	v6_ana_protocol_1	H'06	R/W	IPv6 Analyzable Protocol No 1 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'06 TCP (6). Write to this field when CSFR16. v6_ana_protocol_1_en is '0b'.
7 to 0	v6_ana_protocol_0	H'00	R/W	IPv6 Analyzable Protocol No 0 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'00 Hop-by-Hop (0). Write to this field when CSFR16. v6_ana_protocol_0_en is '0b'.

31.4.3.30 IPv6 analysis Protocol Condition register 1 (CSFR16_1)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	v6_ana_protocol_7	H'3A	R/W	IPv6 Analyzable Protocol No.7 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'3A ICMPv6 (58). Write to this field when CSFR16. v6_ana_protocol_7_en is '0b'.
23 to 16	v6_ana_protocol_6	H'33	R/W	IPv6 Analyzable Protocol No.6 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'33 Authentication (51). Write to this field when CSFR16. v6_ana_protocol_6_en is '0b'.
15 to 8	v6_ana_protocol_5	H'2C	R/W	IPv6 Analyzable Protocol No.5 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'2C Flag (44). Write to this field when CSFR16. v6_ana_protocol_5_en is '0b'.
7 to 0	v6_ana_protocol_4	H'2B	R/W	IPv6 Analyzable Protocol No.4 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'2B Routing (43). Write to this field when CSFR16. v6_ana_protocol_4_en is '0b'.

31.4.3.31 IPv6 analysis Protocol Condition register 2 (CSFR16_2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	v6_ana_protocol_8[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	v6_ana_protocol_8	H'3C	R/W	IPv6 Analyzable Protocol No 8 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'3C Destination Options (60). Write to this field when CSFR16. v6 ana protocol 8 en is '0b'.

31.4.3.32 IPv6 analysis Protocol Condition Enable register (CSFR16)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	v6_ana_protocol_8_en	v6_ana_protocol_7_en	v6_ana_protocol_6_en	v6_ana_protocol_5_en	v6_ana_protocol_4_en	v6_ana_protocol_3_en	v6_ana_protocol_2_en	v6_ana_protocol_1_en	v6_ana_protocol_0_en
Initial Value	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	v6_ana_protocol_8_en	1b	R/W	v6_ana_protocol_8 enable Enable for the value of v6_ana_protocol_8 which is configured in CSFR16_i. 0b: Disable 1b: Enable
7	v6_ana_protocol_7_en	1b	R/W	v6_ana_protocol_7 enable Enable for the value of v6_ana_protocol_7 which is configured in CSFR16_i. 0b: Disable 1b: Enable
6	v6_ana_protocol_6_en	1b	R/W	v6_ana_protocol_6 enable Enable for the value of v6_ana_protocol_6 which is configured in CSFR16_i. 0b: Disable 1b: Enable
5	v6_ana_protocol_5_en	1b	R/W	v6_ana_protocol_5 enable Enable for the value of v6_ana_protocol_5 which is configured in CSFR16_i. 0b: Disable 1b: Enable
4	v6_ana_protocol_4_en	1b	R/W	v6_ana_protocol_4 enable Enable for the value of v6_ana_protocol_4 which is configured in CSFR16_i. 0b: Disable 1b: Enable
3	v6_ana_protocol_3_en	1b	R/W	v6_ana_protocol_3 enable Enable for the value of v6_ana_protocol_3 which is configured in CSFR16_i. 0b: Disable 1b: Enable
2	v6_ana_protocol_2_en	1b	R/W	v6_ana_protocol_2 enable Enable for the value of v6_ana_protocol_2 which is configured in CSFR16_i. 0b: Disable 1b: Enable
1	v6_ana_protocol_1_en	1b	R/W	v6_ana_protocol_1 enable Enable for the value of v6_ana_protocol_1 which is configured in CSFR16_i. 0b: Disable 1b: Enable
0	v6_ana_protocol_0_en	1b	R/W	v6_ana_protocol_0 enable Enable for the value of v6_ana_protocol_0 which is configured in CSFR16_i. 0b: Disable 1b: Enable

31.4.3.33 Wake Up Interrupt Status register (CSFR20)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	trs_dmac	—	—	—	—	—	—	—	ieee_len	ns_dis	ns_dup	ns_uni_nopt	ns_uni
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ns_mul	ana_stp	v6_ana_protocol	arp_noreq	garp_self	garp_other	arp_self	arp_other	v6_udp_pt	v4_udp_pt	v6_protocol	v4_protocol	type	mac_da_mul	mac_da_bro	mac_da_uni
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	trs_dmac	0b	R/W	Transmit to DMAC Wake Up Interrupt which occurs by transmitting Frame from TOE to DMAC without discarding of filtering ‘1’ Write: Clear ‘0’ Write: Invalid
27 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	ieee_len	0b	R/W	IEEE Length filter Wake Up Interrupt which occurs by reception of IEEE802.3 Length Frame ‘1’ Write: Clear ‘0’ Write: Invalid
19	ns_dis	0b	R/W	Neighbor Solicitation discarded filter Wake Up Interrupt which occurs by reception of Invalid Neighbor Solicitations ‘1’ Write: Clear ‘0’ Write: Invalid
18	ns_dup	0b	R/W	Neighbor Solicitation Duplicate filter Wake Up Interrupt which occurs by reception of IP Duplicate Detection Neighbor Solicitations. ‘1’ Write: Clear ‘0’ Write: Invalid
17	ns_uni_nopt	0b	R/W	Neighbor Solicitation Unicast no option filter Wake Up Interrupt which occurs by reception of unicast Neighbor Solicitations (Except “Option Type=1” or No “Option Type”). ‘1’ Write: Clear ‘0’ Write: Invalid
16	ns_uni	0b	R/W	Neighbor Solicitation Unicast filter Wake Up Interrupt which occurs by reception of unicast Neighbor Solicitations (“Option Type=1”). ‘1’ Write: Clear ‘0’ Write: Invalid
15	ns_mul	0b	R/W	Neighbor Solicitation Multicast filter Wake Up Interrupt which occurs by reception of multicast Neighbor Solicitations. ‘1’ Write: Clear ‘0’ Write: Invalid

Bit	Bit Name	Initial Value	R/W	Description
14	ana_stp	0b	R/W	Analysis stop filter Wake Up Interrupt which occurs by reception of Un-analyzable Frames. ‘1’ Write: Clear ‘0’ Write: Invalid
13	v6_ana_protocol	0b	R/W	IPv6 Analysis protocol filter Wake Up Interrupt which occurs by reception of Un-analyzable IPv6 Protocol No. ‘1’ Write: Clear ‘0’ Write: Invalid
12	arp_noreq	0b	R/W	ARP no request filer Wake Up Interrupt which occurs by reception of ARP (Except ARP REQ). ‘1’ Write: Clear ‘0’ Write: Invalid
11	garp_self	0b	R/W	GARP self-filter Wake Up Interrupt which occurs by reception of GARP REQ for Local Station. ‘1’ Write: Clear ‘0’ Write: Invalid
10	garp_other	0b	R/W	GARP other filter Wake Up Interrupt which occurs by reception of GARP REQ for Other Station. ‘1’ Write: Clear ‘0’ Write: Invalid
9	arp_self	0b	R/W	ARP self-filter Wake Up Interrupt which occurs by reception of ARP REQ for Local Station. ‘1’ Write: Clear ‘0’ Write: Invalid
8	arp_other	0b	R/W	ARP other filter Wake Up Interrupt which occurs by reception of ARP REQ for Other Station. ‘1’ Write: Clear ‘0’ Write: Invalid
7	v6_udp_pt	0b	R/W	IPv6 UDP port filter Wake Up Interrupt which occurs by match of IPv6 UDP Port No. ‘1’ Write: Clear ‘0’ Write: Invalid
6	v4_udp_pt	0b	R/W	IPv4 UDP port filter Wake Up Interrupt which occurs by match of IPv4 UDP Port No. ‘1’ Write: Clear ‘0’ Write: Invalid
5	v6_protocol	0b	R/W	IPv6 protocol filter Wake Up Interrupt which occurs by match of IPv6 Protocol No. ‘1’ Write: Clear ‘0’ Write: Invalid
4	v4_protocol	0b	R/W	IPv4 protocol filter Wake Up Interrupt which occurs by match of IPv4 Protocol No. ‘1’ Write: Clear ‘0’ Write: Invalid
3	type	0b	R/W	Type filter Wake Up Interrupt which occurs by match of Ether Type. ‘1’ Write: Clear ‘0’ Write: Invalid
2	mac_da_mul	0b	R/W	MAC DA multicast filter Wake Up Interrupt which occurs by match of Multicast MAC DA. ‘1’ Write: Clear ‘0’ Write: Invalid

Bit	Bit Name	Initial Value	R/W	Description
1	mac_da_bro	0b	R/W	MAC DA broadcast filter Wake Up Interrupt which occurs by match of Broadcast MAC DA. ‘1’ Write: Clear ‘0’ Write: Invalid
0	mac_da_uni	0b	R/W	MAC DA unicast filter Wake Up Interrupt which occurs by match of Unicast MAC DA. ‘1’ Write: Clear ‘0’ Write: Invalid

NOTE

- From b[0] to b[20] are set to “1” when TOE received the Rx Frame which matches to Filter Conditions even if that frame has any error (such as CRC error, short length error or etc.). Therefore, use these bits only for Debug.
- b [28] is set to “1” only when TOE sent a normal Rx Frame to DMAC.

31.4.3.34 Wake Up Interrupt Mask register (CSFR21)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	trs_dmac_m	—	—	—	—	—	—	—	ieee_len_m	ns_dis_m	ns_dup_m	ns_uni_nopt_m	ns_uni_m
Initial Value	0	0	0	1	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ns_mul_m	ana_stp_m	v6_ana_protocol_m	arp_noreq_m	garp_self_m	garp_other_m	arp_self_m	arp_other_m	v6_udp_pt_m	v4_udp_pt_m	v6_protocol_m	v4_protocol_m	type_m	mac_da_mul_m	mac_da_bro_m	mac_da_uni_m
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	trs_dmac_m	1b	R/W	Transmit to DMAC Mask Mask Configuration for Wake-Up Interrupt which occurs by transmitting Frame from TOE to DMAC without discarding of filtering 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
27 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	ieee_len_m	1b	R/W	IEEE Length filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of IEEE802.3 Length Frame 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
19	ns_dis_m	1b	R/W	Neighbor Solicitation discarded filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of Invalid Neighbor Solicitations 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
18	ns_dup_m	1b	R/W	Neighbor Solicitation Duplicate filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of IP Duplicate Detection Neighbor Solicitations. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
17	ns_uni_nopt_m	1b	R/W	Neighbor Solicitation Unicast no option filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of unicast Neighbor Solicitations (Except “Option Type=1” or No “Option Type”). 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
16	ns_uni_m	1b	R/W	Neighbor Solicitation Unicast filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of unicast Neighbor Solicitations (“Option Type=1”). 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
15	ns_mul_m	1b	R/W	Neighbor Solicitation Multicast filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of multicast Neighbor Solicitations. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)

Bit	Bit Name	Initial Value	R/W	Description
14	ana_stp_m	1b	R/W	Analysis stop filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of Un-analyzable Frames. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
13	v6_ana_protocol_m	1b	R/W	IPv6 Analysis protocol filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of Un-analyzable IPv6 Protocol No. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
12	arp_noreq_m	1b	R/W	ARP no request filer Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of ARP (Except ARP REQ). 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
11	garp_self_m	1b	R/W	GARP self-filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of GARP REQ for Local Station. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
10	garp_other_m	1b	R/W	GARP other filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of GARP REQ for Other Station. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
9	arp_self_m	1b	R/W	ARP self-filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of ARP REQ for Local Station. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
8	arp_other_m	1b	R/W	ARP other filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of ARP REQ for Other Station. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
7	v6_udp_pt_m	1b	R/W	IPv6 UDP port filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of IPv6 UDP Port No. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
6	v4_udp_pt_m	1b	R/W	IPv4 UDP port filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of IPv4 UDP Port No. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
5	v6_protocol_m	1b	R/W	IPv6 protocol filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of IPv6 Protocol No. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
4	v4_protocol_m	1b	R/W	IPv4 protocol filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of IPv4 Protocol No. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)

Bit	Bit Name	Initial Value	R/W	Description
3	type_m	1b	R/W	Type filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of Ether Type. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
2	mac_da_mul_m	1b	R/W	MAC DA multicast filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of Multicast MAC DA. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
1	mac_da_bro_m	1b	R/W	MAC DA broadcast filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of Broadcast MAC DA. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
0	mac_da_uni_m	1b	R/W	MAC DA unicast filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of Unicast MAC DA. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)

31.4.3.35 Auto Response Configuration register1 (CSFR30)

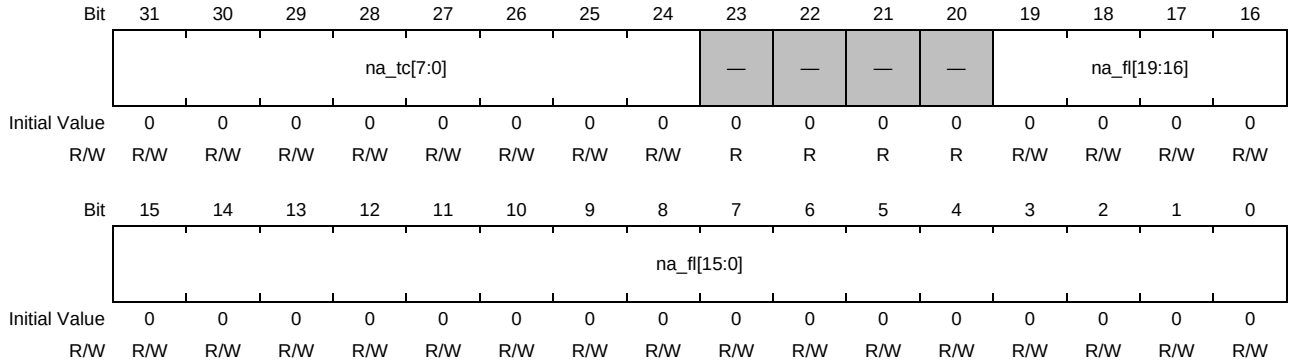
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	na_dup_of
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ns_dup_int_en	ns_uni_nopt_int_en	ns_uni_int_en	ns_mul_int_en	ns_dup_reply_en	—	ns_uni_reply_en	ns_mul_reply_en	—	garp_self_int_en	garp_other_int_en	arp_self_int_en	—	garp_self_reply_en	—	arp_self_reply_en
Initial Value	0	1	1	1	1	0	1	1	0	0	1	1	0	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	na_dup_of	1b	R/W	Neighbor Advertisement Duplicate Override flag Configuration of Override flag to Auto Response Neighbor Advertisements when the reception of IP Duplicate Detection Neighbor Solicitation for Local Station. Usually, do not change this bit. Keep initial values “1b”. 0b: Override flag is set as 0 1b: Override flag is set as 1
15	ns_dup_int_en	0b	R/W	Neighbor Solicitation Duplicate interrupt enable Enable of Interrupt which occurs by reception of IP Duplicate Detection Neighbor Solicitations for Local Station. 0b: Disable 1b: Enable
14	ns_uni_nopt_int_en	1b	R/W	Neighbor Solicitation Unicast no option interrupt enable Enable of Interrupt which occurs by reception of Unicast Neighbor Solicitations for Local Station (Except “Option Type=1” or No “Option Type”). 0b: Disable 1b: Enable
13	ns_uni_int_en	1b	R/W	Neighbor Solicitation Unicast interrupt enable Enable of Interrupt which occurs by reception of Unicast Neighbor Solicitations for Local Station (Option Type=1). 0b: Disable 1b: Enable
12	ns_mul_int_en	1b	R/W	Neighbor Solicitation Multicast interrupt enable Enable of Interrupt which occurs by reception of Multicast Neighbor Solicitations for Local Station. 0b: Disable 1b: Enable
11	ns_dup_reply_en	1b	R/W	Neighbor Solicitation Duplicate reply enable Enable of Auto Response for reception of IP Duplicate Detection Neighbor Solicitations for Local Station. 0b: Disable 1b: Enable
10	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
9	ns_uni_reply_en	1b	R/W	Neighbor Solicitation Unicast reply enable Enable of Auto Response for reception of Unicast Neighbor Solicitations (Option Type=1) for Local Station. 0b: Disable 1b: Enable
8	ns_mul_reply_en	1b	R/W	Neighbor Solicitation Multicast reply enable Enable of Auto Response for reception of Multicast Neighbor Solicitations for Local Station. 0b: Disable 1b: Enable
7	—	0b	R	Reserved T Whenever it is read, 0 is read. The written value will be ignored.
6	garp_self_int_en	0b	R/W	GARP self-interrupt enable Enable of Interrupt which occurs by reception of GARP REQ for Local Station. 0b: Disable 1b: Enable
5	garp_other_int_en	1b	R/W	GARP other interrupt enable Enable of Interrupt which occurs by reception of GARP REQ for Other Station. 0b: Disable 1b: Enable
4	arp_self_int_en	1b	R/W	ARP self-interrupt enable Enable of Interrupt which occurs by reception of ARP REQ for Local Station. 0b: Disable 1b: Enable
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	garp_self_reply_en	1b	R/W	GARP self reply enable Enable of Auto Response for reception of GARP REQ for Local Station. 0b: Disable 1b: Enable
1	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	arp_self_reply_en	1b	R/W	ARP self reply enable Enable of Auto Response for reception of ARP REQ for Local Station. 0b: Disable 1b: Enable

31.4.3.36 Auto Response Configuration register2 (CSFR31)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	na_tc	H'00	R/W	Neighbor Advertisement Traffic Class Traffic Class of Auto Response Neighbor Advertisements for Received Neighbor Solicitations for Local Station. Write to this field when CSFR00.reply_fifo_en is "0".
23 to 20	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
19 to 0	na_fl	H'0_0000	R/W	Neighbor Advertisement Flow Label Flow Label of Auto Response Neighbor Advertisements for Received Neighbor Solicitations for Local Station. Write to this field when CSFR00.reply_fifo_en is "0".

31.4.3.37 ARPREQ/Neighbor Solicitations Receive Interrupt Status register (CSFR40)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	reply_fifo_full	reg_fifo_full	arp_ns
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	reply_fifo_full	0b	R/W	Reply FIFO full This bit indicates that although TOE received ARPREQ or Neighbor Solicitations, but it was discarded because REPLY_FIFO is full. ‘1’ Write: Clear ‘0’ Write: Invalid
1	reg_fifo_full	0b	R/W	Reg FIFO full This bit indicates that although TOE received ARPREQ or Neighbor Solicitations, but it was discarded because REG_FIFO is full. ‘1’ Write: Clear ‘0’ Write: Invalid
0	arp_ns	0b	R/W	ARP Neighbor Solicitation This bit indicates that TOE received ARPREQ or Neighbor Solicitations and they are stored to REG_FIFO completely. ‘1’ Write: Clear ‘0’ Write: Invalid

31.4.3.38 ARPREQ/Neighbor Solicitations Receive Interrupt Mask register (CSFR41)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	reply_fifo_full_m	reg_fifo_full_m	arp_ns_m
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	reply_fifo_full_m	0b	R/W	Reply FIFO full Mask Mask Configuration for ARPREQ/Neighbor Solicitations Receive Interrupt which occurs by reply_fifo_full. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
1	reg_fifo_full_m	0b	R/W	Reg FIFO full Mask Mask Configuration for ARPREQ/Neighbor Solicitations Receive Interrupt which occurs by reg_fifo_full. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
0	arp_ns_m	0b	R/W	ARP Neighbor Solicitation Mask Mask Configuration for ARPREQ/Neighbor Solicitations Receive Interrupt which occurs by arp_ns. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)

31.4.3.39 Rx ARPREQ/Neighbor Solicitation Upper MAC SA register (CSFR42_U)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ns_option_type[7:0]								—	—	—	—	—	—	arp_ns_spec	rx_arp_ns
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	arp_sa_mac[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

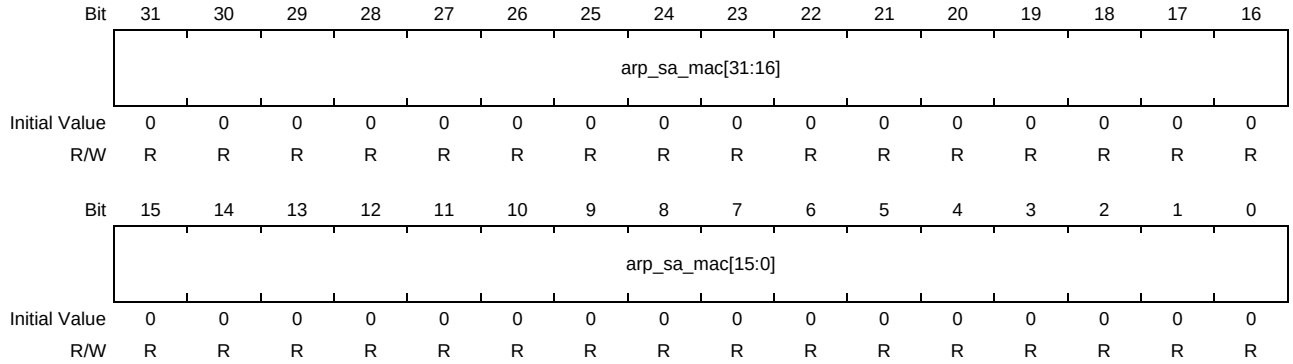
Bit	Bit Name	Initial Value	R/W	Description
31 to 5	ns_option_type	H'00	R	Neighbor Solicitations option type This field indicates Received Neighbor Solicitations Option Type. "H'00" means that Received Frame did not have Option. When rx_arp_ns = 0 (ARPREQ), this field is invalid.
23 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	arp_ns_spec	0b	R	ARP Neighbor Solicitations spec This field indicates Received ARPREQ/Neighbor Solicitations Frame Type. The Case of ARPREQ (rx_arp_ns = 0) 0b: ARPREQ 1b: GARPREQ The Case of Neighbor Solicitations (rx_arp_ns = 1) 0b: Multicast Neighbor Solicitations 1b: Unicast Neighbor Solicitations
16	rx_arp_ns	0b	R	ARP Neighbor Solicitations This field indicates which frame did TOE receive ARPREQ or Neighbor Solicitations. 0b: ARPREQ 1b: Neighbor Solicitations
15 to 0	arp_sa_mac	H'0000	R	ARP source Address MAC This field indicates the Source MAC Address [47:32] of Received ARPREQ/Neighbor Solicitation. When rx_arp_ns = 1 and ns_option_type = H'00, this field is invalid.

[Update timing of this register]

This register is updated when REG FIFO is empty and the interrupt of int_arp_ns_n caused by "Reception of ARPREQ or Neighbor Solicitations and Stored to REG FIFO" is asserted.

This register is updated when REG FIFO is not empty and the interrupt of int_arp_ns_n caused by "Reception of ARPREQ or Neighbor Solicitations and Stored to REG FIFO" is cleared by CSFR40.arp_ns.

31.4.3.40 Rx ARPREQ/Neighbor Solicitation Lower MAC SA register (CSFR42_L)



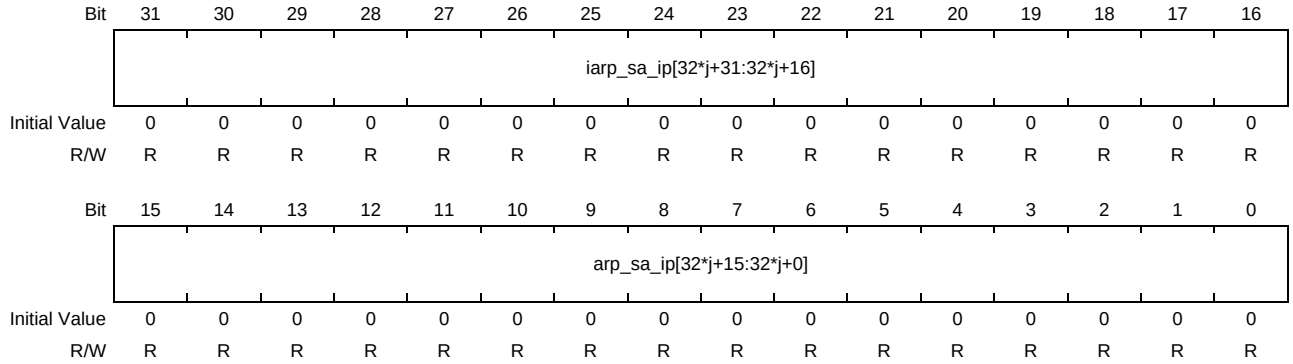
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	arp_sa_mac	H'0000_0000	R	ARP source Address MAC This field indicates the Source MAC Address [31:0] of Received ARPREQ/Neighbor Solicitation. When rx_arp_ns = 1 and ns_option_type = H'00, this field is invalid.

[Update timing of this register]

This register is updated when REG FIFO is empty and the interrupt of int_arp_ns_n caused by "Reception of ARPREQ or Neighbor Solicitations and Stored to REG FIFO" is asserted.

This register is updated when REG FIFO is not empty and the interrupt of int_arp_ns_n caused by "Reception of ARPREQ or Neighbor Solicitations and Stored to REG FIFO" is cleared by CSFR40.arp_ns.

31.4.3.41 Rx ARPREQ/Neighbor Solicitation IP SA register i (i = 0 to 3) (CSFR43_i)



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	arp_sa_ip	All 0	R	ARP source Address IP This field indicates the Source IP Address[32*j+31:32*j] of Received ARPREQ/Neighbor Solicitations. The relationship between ‘i’ and ‘j’ is the following. i = 0 -> j = 3 i = 1 -> j = 2 i = 2 -> j = 1 i = 3 -> j = 0 When rx_arp_ns = 0 (ARPREQ), Only Lower 32bit (CSFR43_3) is valid.

[Update timing of this register]

This register is updated when REG FIFO is empty and the interrupt of int_arp_ns_n caused by “Reception of ARPREQ or Neighbor Solicitations and Stored to REG FIFO” is asserted.

This register is updated when REG FIFO is not empty and the interrupt of int_arp_ns_n caused by “Reception of ARPREQ or Neighbor Solicitations and Stored to REG FIFO” is cleared by CSFR40.arp_ns.

31.5 Operation

The Gigabit Ethernet Interface consists of the following functional units:

- DMA transfer controller (DMAC):
  Handles DMA transfer between the data storage areas for reception and transmission in the URAM and the reception and transmission FIFO
- TCP/IP Offload Engine (TOE):
  Calculates Checksum by hardware. And also filters Ethernet Frames and responds specific frames automatically.
- MAC controller (E-MAC):
  Handles reception/transmission transfer from/to the RGMII or MII.

Using its direct memory access (DMA) function, the DMAC handles DMA transfer of frame data between the destinations for storing Ethernet frame data for transmission and reception in the URAM and the FIFO for reception and transmission. Data cannot be directly read from or written to the FIFO.

To handle DMA transfer, the DMAC requires information that includes the addresses for storage of data for transmission and received data. These data are referred to as descriptors. The DMAC reads data for transmission from the storage area for data to be transmitted according to the information in descriptors and writes received data to the storage area for received data accompanied by information in descriptors. The descriptors are placed in the URAM. Arranging multiple descriptors in descriptor lists allows the continuous reception or transmission of multiple Ethernet frames.

TOE calculates checksum of received frames from E-MAC. And then it outputted to DMAC. TOE also calculates checksum of transmission frames from DMAC. And then it outputted to E-MAC. When TOE received any frames from E-MAC, it analyzes them. By the configuration of registers, TOE filters received frames. If received frame does not correspond to the configured filter conditions by registers, it will be discarded. By using this analysis function, TOE is also able to respond to ARP Request and Neighbor Solicitation automatically.

The E-MAC supports a RGMII and MII, which provides an interface format for the externally connected PHY-LSI. The E-MAC transmits frames from TOE to the RGMII or MII. It also transmits frames from the RGMII or MII to TOE.

31.5.1 DMAC Operating Modes

Figure 31.2 illustrates the operating modes of the DMAC.

Transitions of DMAC operating mode are under the control of the items listed below.

- CPU operating mode (hardware reset)
- Configuration of the operating mode configuration bits (CCC.OPC) in the DMAC mode register

The operating mode can be confirmed by reading the operating mode status bits in the DMAC status register (CSR.OPS).

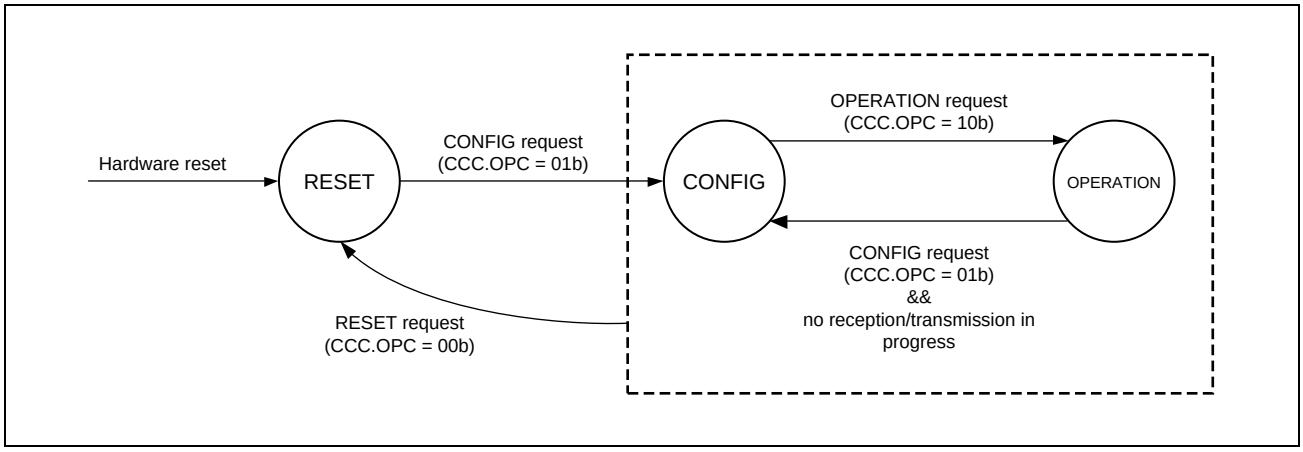


Figure 31.2 Operating Mode of DMAC

31.5.1.1 Operating Modes

(1) RESET mode

After a hardware reset, the DMAC enters RESET mode.

In RESET mode, only the DMAC operating mode control function is controllable and other functions are all stopped. This mode is designed for reduced power when the Ethernet function is not necessary.

(2) CONFIG mode

In CONFIG mode, various settings for the DMAC can be made.

The operation of most functions is stopped, and all status registers are initialized to their reset values.

(3) OPERATION mode

In OPERATION mode, all functions of the DMAC can operate. Ethernet communications can only proceed in this mode.

31.5.1.2 How to Set the Operating Mode

Set the operating mode configuration bits in the DMAC mode register (CCC.OPC) to select the operating mode. Furthermore, the operating mode can be checked by reading the operating mode status bits in the DMAC status register (CSR.OPS).

Transitions other than leaving from OPERATION mode are made after the value is written to the operating mode configuration bits (CCC.OPC).

For transitions leaving from OPERATION mode, it is not executed immediately because any transmission and reception in progress will be executed before leaving from OPERATION mode. To prevent any trouble, follow the "Stop Procedure" of **Section 31.5.13.4 Stop Procedure**.

31.5.1.3 Operating Mode Transitions Due to Hardware

The following hardware factors can also initiate transitions of the DMAC operating mode.

(1) Hardware reset

Resetting of the LSI chip leads to resetting of the entire Gigabit Ethernet Interface. The operating mode shifts to RESET mode.

31.5.2 Descriptors

31.5.2.1 Data Representation in URAM

The DMAC transfers data for transmission and received data to and from the application software via the URAM.

The memory in the URAM for use by the DMAC is configured with control structures referred to as descriptors and associated areas to which the frame data are allocated. Dividing the memory into a control area and data area allows the flexible allocation of frame data to the URAM. This enables sharing of the areas to which frame data are allocated and the use of non-contiguous areas. Frame data can be copied without using the CPU. Arbitration that ensures hardware and software access to the memory area is also available without access to registers of the DMAC.

Figure 31.3 shows an example of the memory maps for descriptors and the descriptor data area in the URAM.

A descriptor consists of its type (DESCR.DT), which controls the descriptor functions, a descriptor pointer (DESCR.DPTR) indicating the start address for storage of the frame data in the descriptor area, and the data size field (DESCR.DS), indicating the amount of frame data. Post-processing interrupt generation can be set up for each descriptor. Enabling and disabling of the interrupt is controlled by the descriptor processed interrupt enable bits (DESCR.DIE).

The descriptor may also hold information related to content. This information does not affect general descriptor functions. It provides information other than the frame data proper, such as on the state of reception.

For details, see **Section 31.5.3.2, Setting Up Reception Descriptors**, and **Section 31.5.4.1, Setting Up Transmission Descriptors**.

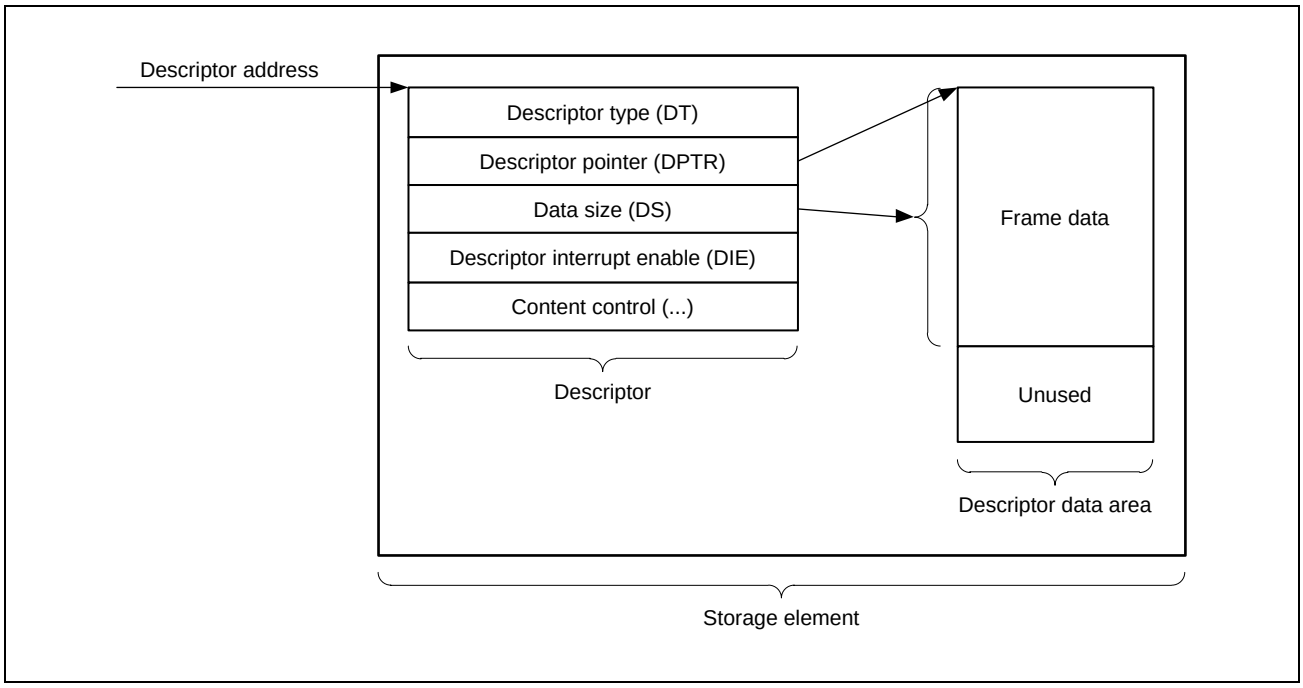


Figure 31.3 Outline of Storage Element Used for Receive and Transmit Queues

The descriptor must be aligned with a 32-bit boundary in the URAM.

Descriptors are configured of 64 bits.

The frame data must also be aligned with a 32-bit boundary in the URAM.

The amount of data in the frame is defined by the data size bits (DESCR.DS). In reception, these bits indicate the upper limit on the size of frames to be received. If the data size is not aligned with a 32-bit boundary, the bytes to the next 32-bit boundary in the data area will be an unused area.

31.5.2.2 Using Descriptor Chains in Queues

Transmission and reception descriptors in the URAM are grouped into queues. The queue is capable of controlling one or more frames. Accordingly, multiple descriptors can be assigned to one queue. A combination of multiple descriptors is referred to as a descriptor chain.

For a descriptor chain, the three general descriptor types listed below are defined. For details on these descriptor types, see **Section 31.5.2.6, Descriptor Type**.

- Descriptors that define frame data
- Descriptors that control the descriptor chain itself (e.g. LINK, EOS).
- Descriptors that arbitrate access by hardware or software

Figure 31.4 shows the two basic topologies for descriptor chains. In the simplified examples in the figure, all descriptors allocated to the chain are stored in the array.

- For a linear descriptor chain, the last descriptor in the array is a control descriptor indicating the end of the descriptors (e.g. EEMPTY).

- For a cyclic descriptor chain, the last descriptor in the array is a control descriptor that returns to the first descriptor in the array (e.g. LINK).

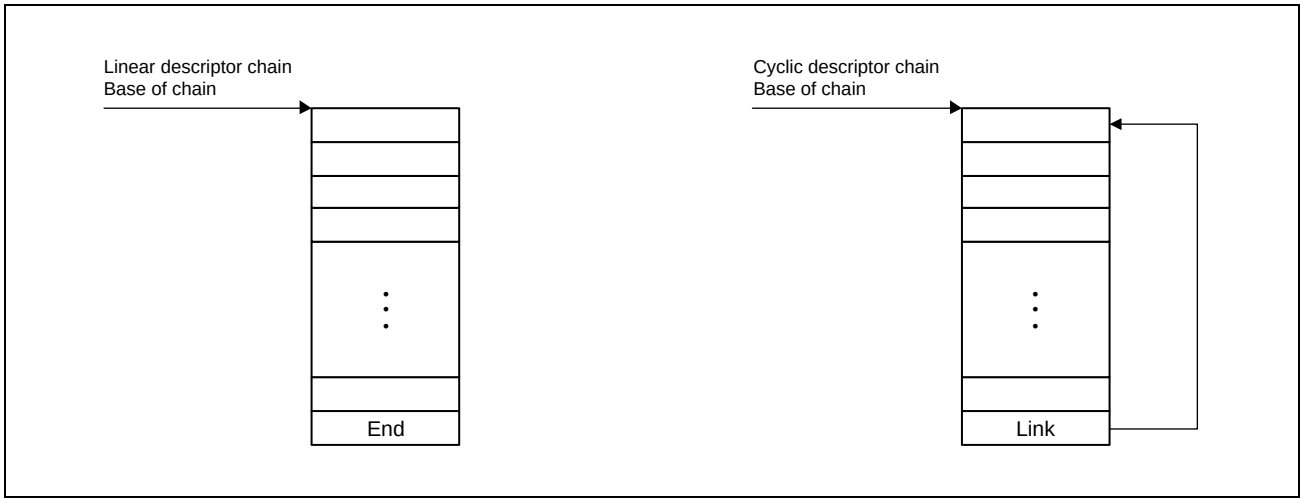


Figure 31.4 Outline of the Basic Descriptor Chains

The relationship between queue and descriptor chain is defined by the base addresses of chain. A queue is connected to one descriptor chain over one round of processing. There is also a method of switching to a different chain while in OPERATION mode.

There are no restrictions on the number of link descriptors and their locations within the chain. The last descriptor of a designed chain determines the topology.

Which chain structure should be used or which topology is suitable depend on the application. A description of how to design descriptor chains to suit various applications is given in **Section 31.5.3.2, Setting Up Reception Descriptors**, and **Section 31.5.4.1, Setting Up Transmission Descriptors**.

31.5.2.3 Descriptor Base Address Table

The base address table in the URAM contains the address of the first descriptor of all chains to be handled by the respective queues.

Entry 0 is used to access transmission queues. Entry 4 is used to access reception queue.

The configuration of entries in the base address table is the same as the configuration of link descriptors. We recommend using the descriptor type (DESCR.DT) LINKFIX. Processing of this link descriptor does not change it, so it does not require updating. The first descriptor of a chain performs hardware and software synchronization. If the application requires hardware and software synchronization for the base addresses, use the descriptor type (DESCR.DT) LINK.

The CPU is only capable of using LINKFIX and LINK as descriptor types (DESCR.DT) of descriptors in the base address table.

Set the location of the base address table in the URAM in the descriptor base address table register (DBAT).

Figure 31.5 shows an example of a base address table for controlling transmission and reception queues. The boxes to the right of the table represent descriptor chains with the desired topologies.

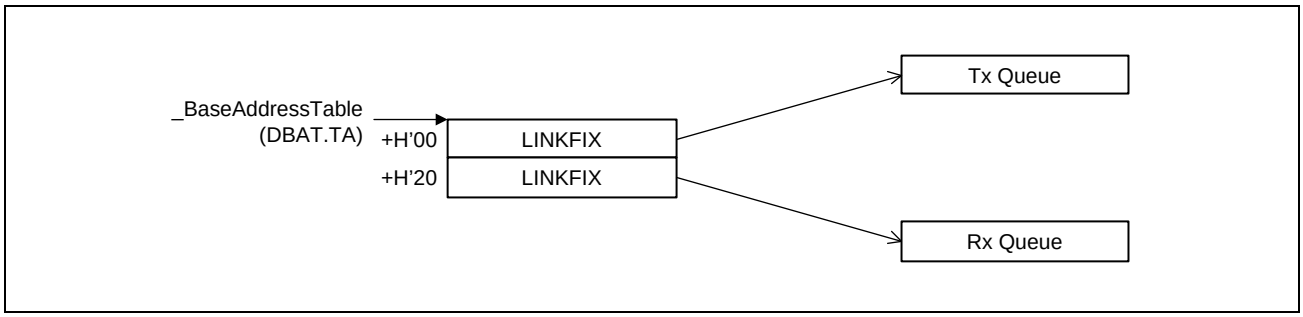


Figure 31.5 Example of a Base Address Table for Reception and Transmission Queues

31.5.2.4 Descriptor Chain Processing

The descriptor that is currently processed or will be processed when the related queue gets active is the current descriptor. The current descriptor address for use by a queue q can be checked in the current descriptor address register q (CDARq).

Current descriptors are stored in registers or in descriptors as described below in the given situations.

- In the descriptor base address table registers for all q queues (DBAT) (DBAT.TA+8*q) when the operating mode shifts to OPERATION mode.
- In the descriptor base address table register (DBAT) (DBAT.TA+8*q) when a base address load request is issued for a queue q by setting the corresponding bit (DLR.LBAq) in the descriptor base address load request register (DLR).
- In DESCR.DPTR for a link descriptor (LINK, LINKFIX) to be processed.
- After a descriptor has been processed, the current descriptor for the same queue is incremented by the size of the descriptors being handled by the queue. The DMAC updates the descriptor type and informs the CPU that the descriptor has been processed.

31.5.2.5 Descriptor Interrupts

A descriptor is able to issue a descriptor interrupt on completion of its processing. The setting of the descriptor interrupt enables bits (DESCR.DIE) in each descriptor selects disabling or generation of the descriptor interrupt.

The descriptor interrupt is a common resource that is shared between reception and transmission queues. Software control of the descriptor interrupt provides a flexible method of application-specific flag processing.

Figure 31.6 illustrates the way in which the DMAC generates descriptor interrupts (or sets bits in the descriptor interrupt status register (DIS.DPFi)). Processing of a descriptor with the value i in the descriptor interrupt enable bits (DESCR.DIE) leads to the corresponding bit in the descriptor interrupt status register (DIS.DPFi) being set.

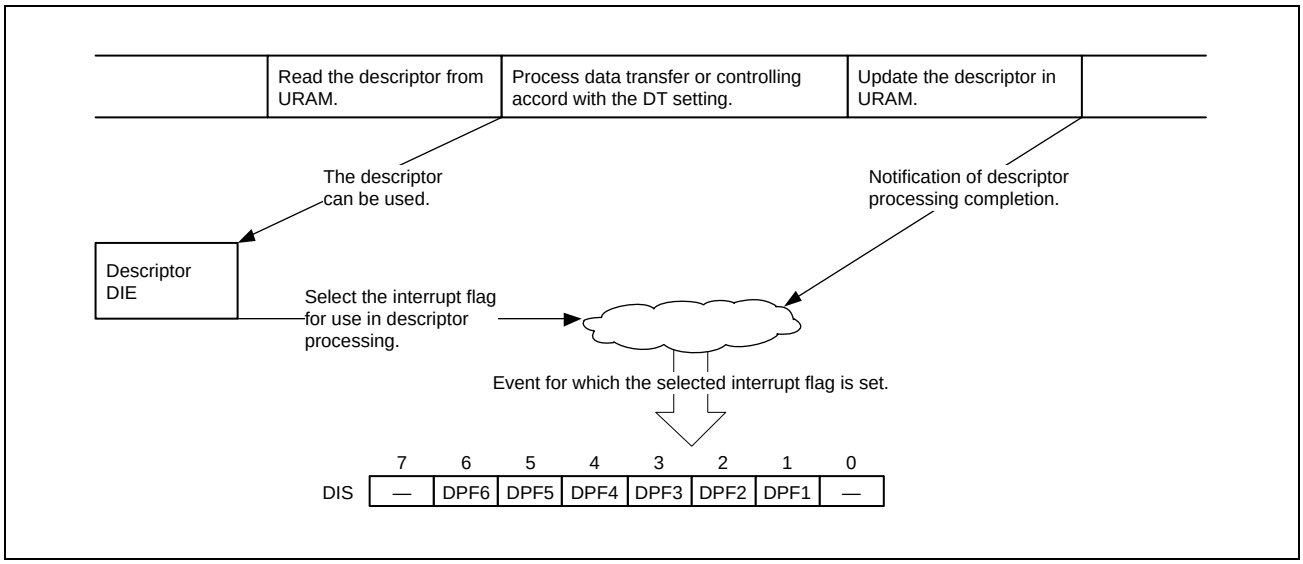


Figure 31.6 Method of Descriptor Interrupt Generation

31.5.2.6 Descriptor Type

The descriptor types (indicated by the DESCR.DT bits) supported by the DMAC fall into the following three categories.

- Definitions of frame data
- Control of descriptor chains
- Hardware and software arbitration

Table 31.7 is a summary of the descriptor types available for the DMAC. Entries under “Name” are the names of the descriptor types and the values under “DT” are the corresponding values to be set in the descriptor type field (DESCR.DT). A given descriptor may be handled differently according to whether it is in a transmission or reception queue, so the transmission and reception columns list the scopes of control and processing of the descriptor types.

The abbreviations defined below are used in the transmission and reception columns.

Definition of SW:

- The descriptor is processed by software.
- Software has access to and may modify the descriptor and descriptor data area.
- This descriptor cannot be changed by hardware (DMAC).

Definition of HW:

- The descriptor is processed by hardware (DMAC).
- Software must modify neither the descriptor nor the descriptor data area.
- Hardware (DMAC) processes this descriptor and subsequently changes the descriptor type.

Invalid:

- This descriptor type is not used in transfer in the given direction (transmission or reception).
- Do not write to this value to the descriptor type (DESCR.DT) field for transfer in the given direction.

- Hardware does not process these descriptor types in the cases listed as invalid. The current descriptor address (CDARq.CDA) will not be changed when processing of a queue for the given direction arrives at a descriptor with this type setting.

Table 31.7 Summary of Descriptor Types

Name	DT	Description	Reception	Transmission
Frame data				
FSTART	5	Frame Start The descriptor points to valid data for a frame. The frame starts with the given data and continues with that indicated by the next descriptor.	SW	HW
FMID	4	Frame Middle The descriptor points to valid data for a frame. The frame started with a previous descriptor and continues to the data indicated by the next descriptor.	SW	HW
FEND	6	Frame End The descriptor points to valid data for a frame. The frame continues from the previous descriptor and ends with the data indicated by in this descriptor.	SW	HW
FSINGLE	7	Frame Single The descriptor points to valid data for a complete frame.	SW	HW
Chain control				
LINK	8	Link Defines the next descriptor in the chain.	HW	HW
LINKFIX	9	Fixed Link Same as LINK, but not changed by DMAC after processing.	SW	SW
EOS	10	End Of Set Control element to split a descriptor chain. The chain stops and waits for user interaction.	HW	HW
HW/SW arbitration				
FEMPTY	12	Frame Empty A descriptor related to frame data but not containing valid data for a frame	HW	SW
DT13	13	Reserved	Invalid	Invalid
DT14	14	Reserved	Invalid	Invalid
FEMPTY_ND	15	Frame Empty No Data storage A descriptor related to frame data but not containing valid data for a frame. The descriptor is processed in the same way as FEMPTY, but data are not stored in the URAM.	HW	Invalid
LEMPTY	2	Link Empty A link descriptor for processing by the DMAC	SW	SW
EEMPTY	3	EOS Empty An EOS descriptor for processing by the DMAC	SW	SW
DT0	0	Reserved	Invalid	Invalid
DT1	1	Reserved	Invalid	Invalid
DT11	11	Reserved	Invalid	Invalid

31.5.2.7 Layout of General Descriptors in the URAM

The DMAC updates processed descriptors in the URAM. The field to be changed in the descriptor being updated depends upon whether the direction is transmission or reception and the queue mode. Other fields will not be changed. There are no restrictions on the values set in unused descriptor fields (indicated by "—" in the figure).

(1) Frame Data Descriptors

The allocation of bits in the frame data descriptors (FSTART, FMID, FEND, and FSINGLE) is shown below.
usable in both reception and transmission

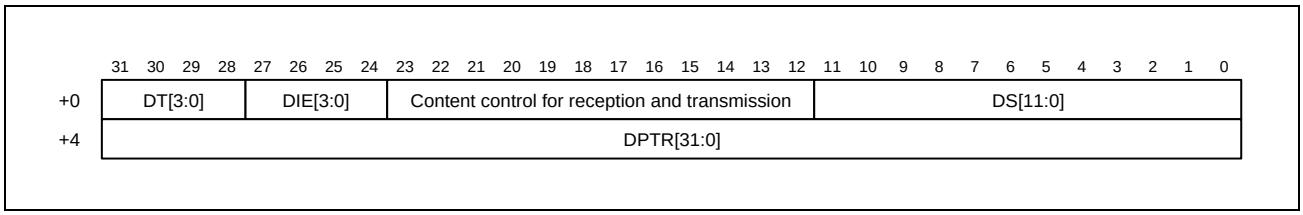


Figure 31.7 The allocation of bits in the frame data descriptors

Table 31.8 Contents of Frame Data Descriptors (DESCR)

Bit Name	Function
DT[3:0]	Descriptor Type 5: FSTART 4: FMID 6: FEND 7: FSINGLE For details, see **Section 31.5.3.2, Setting Up Reception Descriptors**, and **Section 31.5.4.1, Setting Up Transmission Descriptors**.
DIE[3:0]	Descriptor Interrupt Enable 0000b: Descriptor interrupt is disabled. 0001b to 1111b: The corresponding descriptor interrupt is generated (DIS.DPFi).
—	Content Control For details, see **Section 31.5.3.2, Setting Up Reception Descriptors**, and **Section 31.5.4.1, Setting Up Transmission Descriptors**.
DS[11:0]	Data Size Size of the data area/frame data for the descriptor (in bytes)
DPTR[31:0]	Descriptor Pointer Pointer to the data area for the descriptor

Note: Register an address aligned with a 32-bit boundary as the descriptor pointer (DESCR.DPTR).

(2) Hardware/Software Arbitration Descriptors (Only for Reception)

The allocation of bits in the descriptors for hardware/software arbitration (FEMPTY and FEMPTY_ND) is shown below.

The allocation of bits in the arbitration descriptors for use in reception is the same as in frame data descriptors.

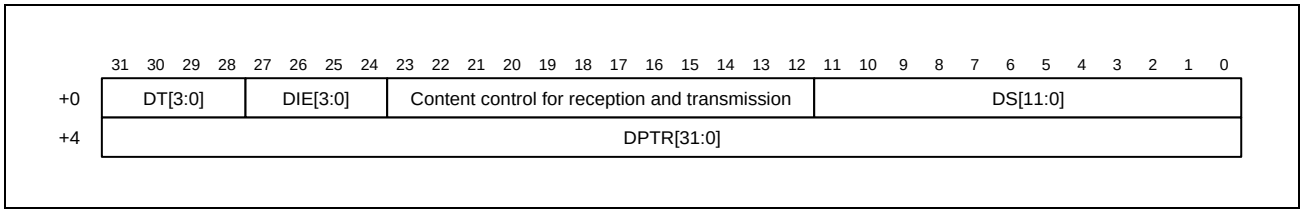


Figure 31.8 The allocation of bits in the arbitration descriptors

Table 31.9 Contents of Hardware/Software Arbitration Descriptors (DESCR)

Bit Name	Function
DT[3:0]	Descriptor Type 12: FEMPTY 15: FEMPTY_ND For details, see **Table 31.7, Summary of Descriptor Types**.
DIE[3:0]	Descriptor Interrupt Enable 0000b: Descriptor interrupt is disabled. 0001b to 1111b: The corresponding descriptor interrupt is generated (DIS.DPFi).
—	Content Control For details, see **Section 31.5.3.2, Setting Up Reception Descriptors**, and **Section 31.5.4.1, Setting Up Transmission Descriptors**.
DS[11:0]	Data Size Size of the data area/frame data for the descriptor (in bytes)
DPTR[31:0]	Descriptor Pointer Pointer to the data area for the descriptor

Note: Register an address aligned with a 32-bit boundary as the descriptor pointer (DESCR.DPTR).

In an FEMPTY descriptor, the descriptor type (DT), descriptor interrupt enable (DIE), data size (DS), and descriptor pointer (DPTR) fields are used.

In an FEMPTY_ND descriptor, the descriptor type (DT), descriptor interrupt enable (DIE), and data size (DS) are used.

(3) Link Descriptors

The allocation of bits in the link descriptors (LINK and LINKFIX) is shown below.

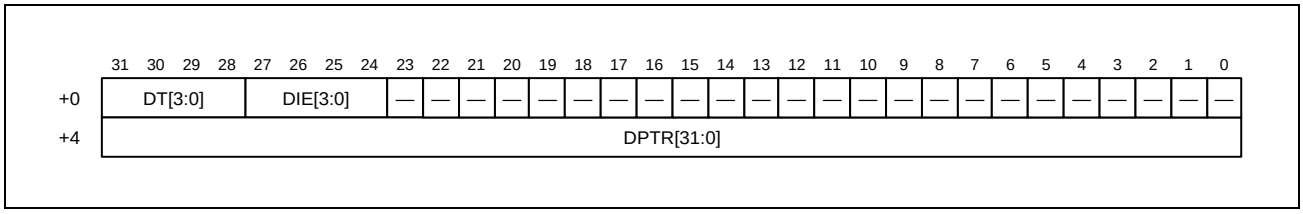


Figure 31.9 The allocation of bits in the link descriptors

Table 31.10 Contents of Link Descriptors (DESCR)

Bit Name	Function
DT[3:0]	Descriptor Type 8: LINK 9: LINKFIX For details, see **Table 31.7, Summary of Descriptor Types**.
DIE[3:0]	Descriptor Interrupt Enable 0000b: Descriptor interrupt is disabled. 0001b to 1111b: The corresponding descriptor interrupt is generated (DIS.DPFi).
DPTR[31:0]	Descriptor Pointer Pointer to the data area for the descriptor Register an address on a 32-bit boundary.

Note: Register an address aligned with a 32-bit boundary as the descriptor pointer (DESCR.DPTR).

(4) Other Descriptors

The allocation of bits in the other descriptors (EOS, FEMPTY (only for transmission), LEMPTY, and EEMPTY) is shown below.

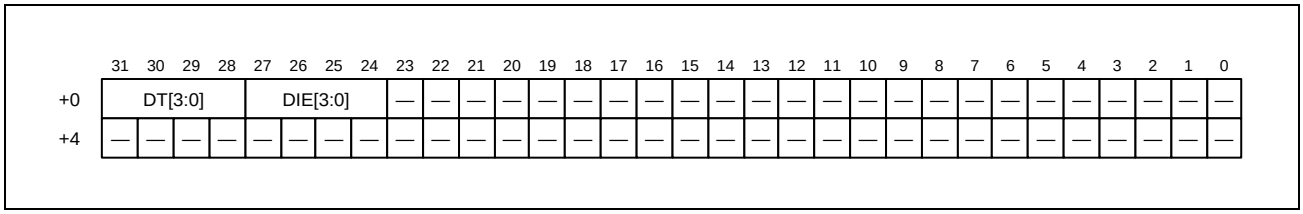


Figure 31.10 The allocation of bits in the other descriptors

Table 31.11 Contents of Other Descriptors (DESCR)

Bit Name	Function
DT[3:0]	Descriptor Type 10: EOS 12: FEMPTY (only for transmission) 2: LEMPTY 3: EEMPTY For details, see **Table 31.7, Summary of Descriptor Types**.
DIE[3:0]	Descriptor Interrupt Enable 0000b: Descriptor interrupt is disabled. 0001b to 1111b: The corresponding descriptor interrupt is generated (DIS.DPFi).

31.5.2.8 How to Use Frame Data Descriptors

In general, Ethernet frames are not of uniform length. The DMAC is capable of dividing frame data into multiple descriptors in order to minimize the memory capacity for frame data. This function allows processing of frames that are longer than the limit for descriptor data areas. Division can also be applied to frames on the basis of their data structures.

To handle frames and descriptors, four descriptor types (DESCR.DT) as FSTART, FEND, FMID and FSINGLE are defined.

Figure 31.11 shows the mapping of frame data by frame data descriptors. The descriptor data areas are allocated to the URAM. For frames that require division into four or more data areas, additional FMID descriptors can be added as required.

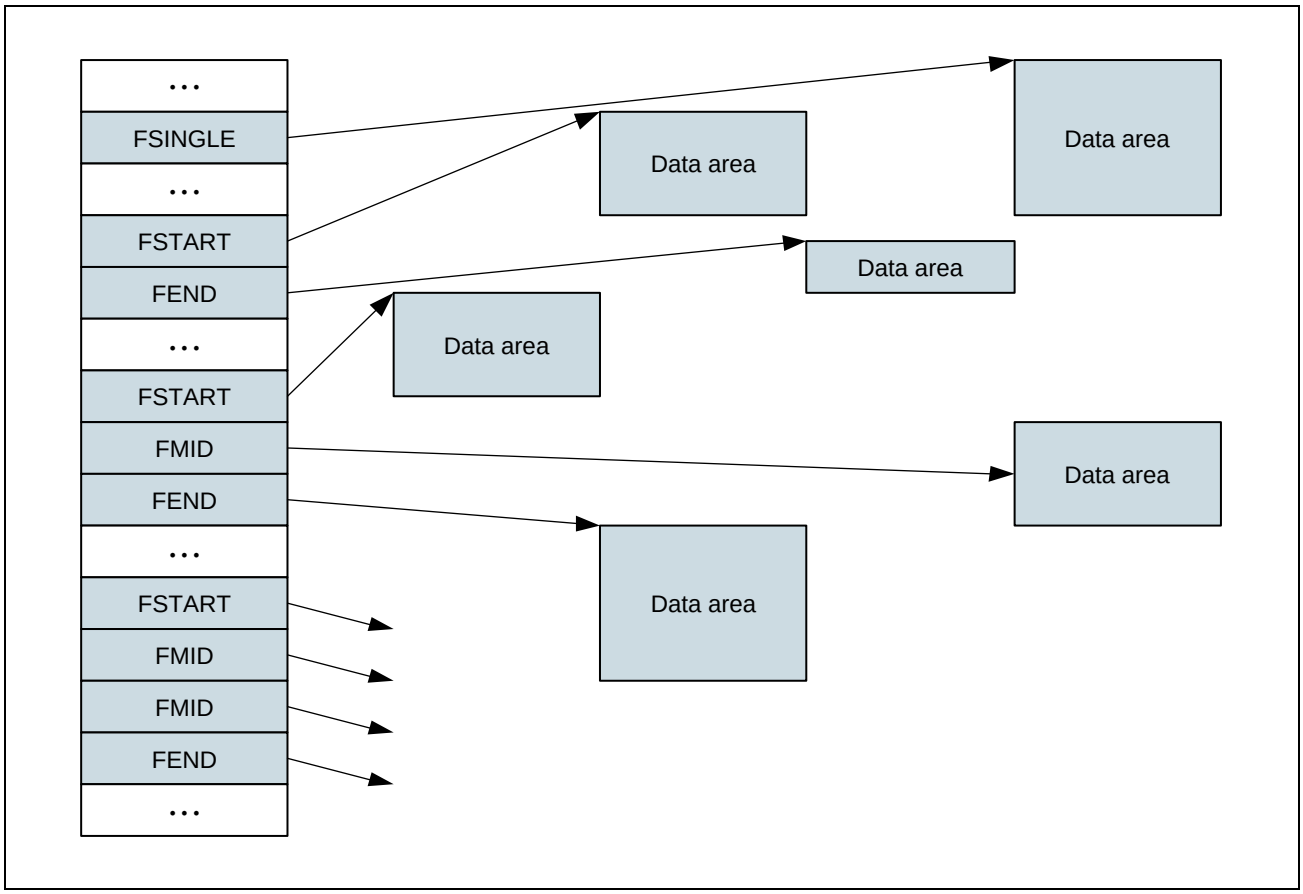


Figure 31.11 Mapping of Frame Data

For reception, the DMAC will store received frame data in the given area. If a received frame has more data than the descriptor data size (DESCR.DS), the DMAC will divide the data up.

For transmission, set the descriptor data size (DESCR.DS) to the actual data size. The DMAC modifies the descriptor type (DESCR.DT) to FEMPTY after processing the relevant descriptor. The data size (DESCR.DS) and descriptor pointer (DESCR.PTR) fields retain their settings.

31.5.2.9 How to Use Chain Control Descriptors

(1) Link Descriptors

The link descriptors can be used to set up cyclic descriptor chains (for details, see **Section 31.5.2.2, Using Descriptor Chains in Queues**)

After a LINK descriptor is processed, its descriptor type (DESCR.DT) is changed to LEMPTY. The descriptor pointer (DESCR.PTR) retains its setting.

After processing of a LINKFIX descriptor, the descriptor type (DESCR.DT) is not updated. Software can change the descriptor type (DESCR.DT), descriptor interrupt enable (DESCR.DIE), and descriptor pointer (DESCR.DPTR). However, DESCR.DT should not be modified by software. Take care to check the current descriptor address register (CDARq.CDA) before changing the descriptor pointer (DESCR.DPTR).

(2) EOS Descriptor

Use the EOS descriptor to divide a descriptor chain into various segments. The queue can continue even after an EOS descriptor.

In transmission, the response to an EOS descriptor is clearing of the transmit start request bit in the transmit configuration control register (TCCR.TSRQ) to 0.

In reception, the response is generation of a receive queue full interrupt (RIS2.QFF), although if the frame currently being received is being divided for storage (received data such as those where some storage is in FMID- or FEND-type frames), the data are not completely stored.

31.5.2.10 How to Use Hardware and Software Arbitration Descriptors

In hardware processing of descriptors, the empty descriptor types (FEMPTY, LEMPTY, and EEMPTY) are used to distinguish various descriptors. For software, they can be used to initiate checking for empty spaces, etc.

(1) FEMPTY and FEMPTY_ND

These descriptor types (DESCR.DT) are used for descriptors that do not contain effective data. Of these, only FEMPTY is used in transmission. The descriptor pointer of a FEMPTYxxx descriptor refers to a descriptor data area.

(2) LEMPTY

This descriptor type (DESCR.DT) is assigned to LINK descriptors after they have been processed. The descriptor pointer (DESCR.DPTR) of an LEMPTY descriptor still points to the linked descriptor.

(3) EEMPTY

This descriptor type (DESCR.DT) is assigned to EOS descriptors after they have been processed. The descriptor pointer (DESCR.DPTR) of an EEMPTY descriptor is not used.

31.5.2.11 Synchronization between Descriptor Access by Hardware and Software

The allocation of descriptor types (DESCR.DT) to the URAM can be used to set up the primary synchronization between hardware and software. By this, the number of CPU accesses to registers of Gigabit Ethernet Interface can be minimized and performance can be increased.

Basic concepts of synchronization:

- Each descriptor type in the set is exclusively for processing by hardware or software, depending on the direction of transfer (see **Table 31.7, Summary of Descriptor Types**).
- Software must not change a descriptor assigned to hardware processing (the hardware does not change descriptors assigned to software processing).

In the case of software processing, the software must process the information in the descriptor and the corresponding frame data before changing the descriptor type. If a descriptor type for hardware is set in DESCR.DT, the software should not change any part of the descriptor or of the corresponding frame data.

31.5.2.12 Tips for Optimizing Performance in Handling Descriptors

The following items are recommended as ways to ensure the optimal use of data structures in the URAM.

They are not requirements but using a different approach may increase the load on the system bus within the LSI chip.

- Register descriptors with 64-bit alignment.
- While in OPERATION mode, use LINKFIX instead of LINK whenever a descriptor need not be changed. Hardware modifies the descriptor type (DESCR.DT) fields of LINK descriptors.
- Frame data is accessed in blocks up to 128 bytes.
- The number of 128 byte borders (addresses H’x_xx00 and H’x_xx80) and frame data inside should be minimized.
- Design the descriptor chains in ways that minimize parallelism of processing. This helps in dividing the chains into segments allocated to different cache pages, and in arranging the different segments exclusively for access by software or hardware.
- Minimize the number of divided frames. This can reduce the overhead of descriptor handling.

31.5.3 Reception Control

The point of the DMAC is to transfer data between the TOE/E-MAC and URAM without intervention by the CPU.

DMAC needs descriptors that define the amounts of frame data to be stored and the locations. When the E-MAC received a frame, it is transferred to TOE and the conditions of reception is also transferred to DMAC as the E-MAC state. The checksum of received frame is calculated by TOE, the received frame is stored to Reception FIFO temporary. If TOE detects any errors from received frame (such as it was unsupported frame or checksum error), the error information is notified to DMAC. After that, the received frame is transferred to URAM by DMAC. At the time, MAC-Status and Error Information of TOE are written to Reception Descriptor by Write Back. For a description of how to set up descriptors for use in reception, see **Section 31.5.3.2, Setting Up Reception Descriptors**.

Figure 31.12 shows the reception data bus.

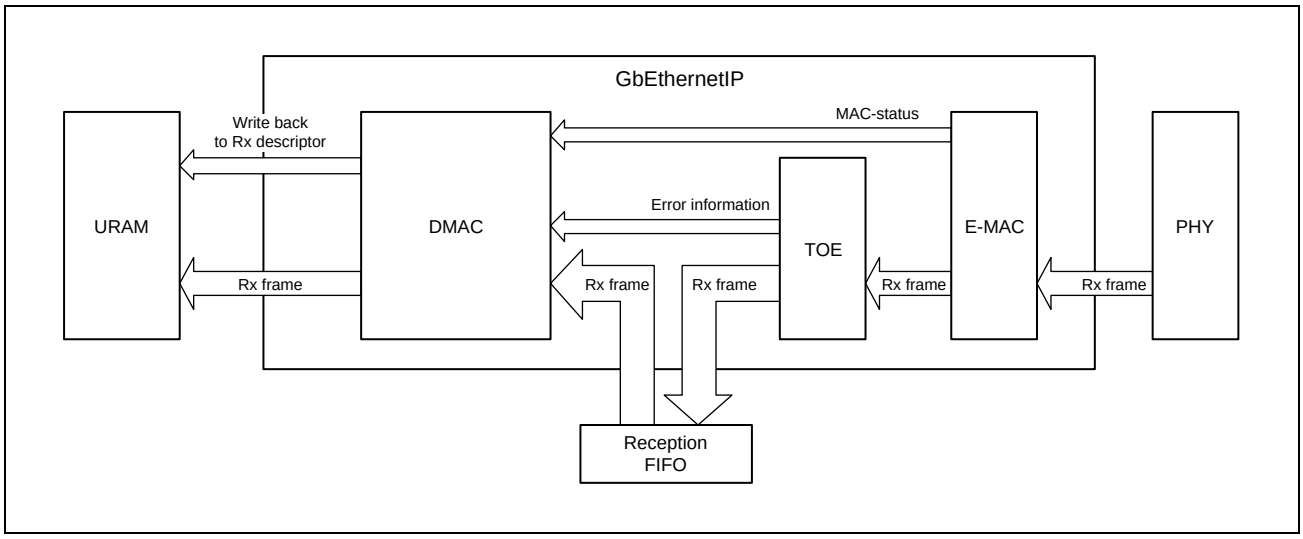


Figure 31.12 Mechanism of General Reception

31.5.3.1 Reception Queues

The DMAC stores all received frames in the URAM.

There is a condition for the DMAC to discard a received frame.

- Detection of an error during reception by the TOE and E-MAC
  - Whether error frames are discarded or stored in reception queue depends on the setting of the error frame enable bit in the receive configuration register (RCR.EFFS). If error frames are to be stored (RCR.EFFS = 1*[1]), they are always stored in queue. In this case, characteristics specific to the queue (e.g. truncation) will vary.

Note 1. RCR.EFFS = 1 is only for debug. Do not set this condition during normal operation.

The flowchart in **Figure 31.13** shows how the DMAC selects the reception queue in accord with the frame type. The result is storage of the frame in the proper queue or the frame being discarded.

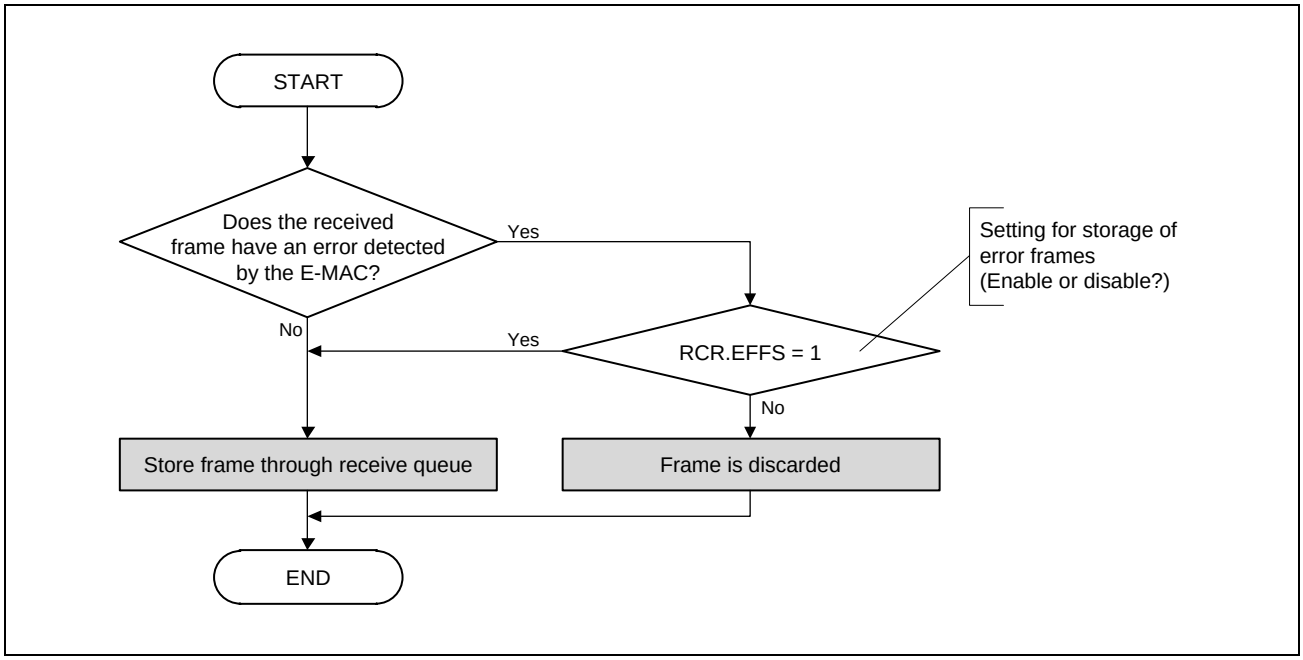


Figure 31.13 Mechanism of Reception Queue Selection

31.5.3.2 Setting Up Reception Descriptors

For reception, the descriptor mechanism is essentially as described in **Section 31.5.2, Descriptors**.

This section describes memory operations that are especially required in handling reception queues.

(1) Reception Descriptor Type

The type of a descriptor is defined by the descriptor type (DESCR.DT) field.

Table 31.12 shows the descriptor types used in reception.

Table 31.12 Descriptor Types in Reception

Descriptor Type (DESCR.DT)	Operation	Write-back
Frame Start (FSTART)	Condition for data not being stored in a reception queue: The RIS2.QFF bit indicates that queue r is full, and the received frame is not stored. Descriptor processing proceeds again in response to further reception.	Not changed
Frame Middle (FMID)	Same as FSTART	Not changed
Frame End (FEND)	Same as FSTART	Not changed
Frame Single (FSINGLE)	Same as FSTART	Not changed
Link (LINK)	Processing proceeds to the descriptor specified by DESCR.DPTR.	LEMPTY
Fixed Link (LINKFIX)	Same as LINK	Not changed
End Of Set (EOS)	A stop point defined by software has been reached. A descriptor of this type within a divided frame (writing of FMID or FEND) stops the frame being stored and the frame is lost. RIS2.QFF indicates that the frame has been lost. If this happens at the start of a frame (writing of FSTART or FSINGLE), storing of frames starts from the next descriptor. In either case, processing shifts to the next descriptor in the chain.	EEMPTY
Frame Empty (FEMPTY)	The descriptor can be used to store received data. Up to DESCR.DS bytes are stored in the descriptor data area. For details, see **Section 31.5.3.3(1), Storing Frame Data in the Descriptor Data Area**.	FSTART, FMID, FEND, or FSINGLE
Frame Empty No Data storage (FEMPTY_ND)	The descriptor can be used to store received data. Up to DESCR.DS bytes are captured from the Reception FIFO but not stored. After processing, DESCR.DS is written back as 0.	FSTART, FMID, FEND or FSINGLE
Link Empty (LEMPTY)	Same as FSTART	Not changed
EOS Empty (EEMPTY)	Same as FSTART	Not changed

(2) Configuration of Reception Frame Data Descriptors

Figure 31.14 shows the configuration of descriptors for use with reception queues. The reception-specific fields (DESCR.MSC, DESCR.PS, DESCR.EI, and DESCR.TI) are described in **Table 31.8**.

For the other fields and the descriptor types, see **Section 31.5.2.6, Descriptor Type**.

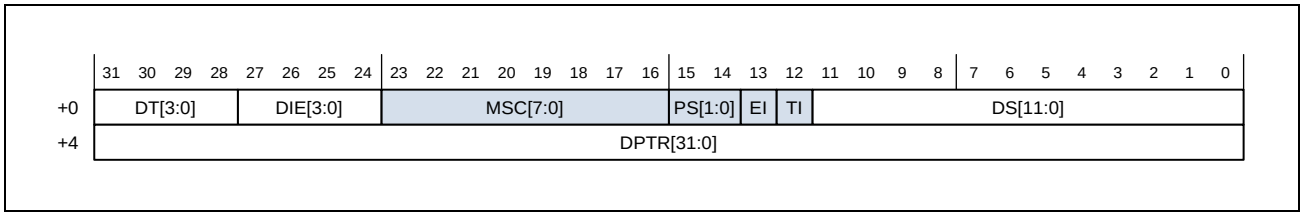


Figure 31.14 Configuration of Descriptor for a Received Frame

Table 31.13 Configuration of a Received Descriptor

Bit Name	Function
MSC	E-MAC/TOE Status Code These bits indicate errors in reception detected by the E-MAC and TOE. In the case of a divided frame, the bits are set to the same value within all descriptors for the frame data. Details of the bits are as follows. MSC [7]: Received frame has a multicast address. (E-MAC) MSC [6]: Check-sum error (TOE) MSC [5]: Received Unsupported frame (TOE) MSC [4]: Received frame has alignment error. (E-MAC) MSC [3]: Received frame is too long. (E-MAC) MSC [2]: Received frame is too short (E-MAC) MSC [1]: Error in frame reception (E-MAC) MSC [0]: Received frame has a CRC error. (E-MAC)
PS	Padding Selection 00b: Padding is not to be inserted.
EI	Error Indication This bit indicates the detection of an error in frame data while a frame was being stored. The bit is set to 1 for a descriptor in which an error has been detected. If the descriptor is for a divided frame, storage of the frame is aborted. 0: No error 1: Error is detected
TI	Truncation Indication This bit indicates whether frame data received from the E-MAC have been truncated before being stored. The bits are set to the same value within all frame data descriptors for a divided frame. 0: Data have not been truncated.

Note: The RCR.EFFS bit specifies whether frames with errors detected by the TOE or E-MAC are to be stored in the URAM or not. When the storing of error frames is disabled, error codes are not written to DESCR.MSC.

31.5.3.3 Reception Processing

After initialization, the DMAC can store received frames in the data area in the URAM as indicated by the descriptor. The DMAC continues to store received data in the URAM as long as space is available for descriptors and data areas.

If the Reception FIFO contains even one frame, storing is executed to the reception queue.

If there is even one empty data descriptor in a queue for which reception has started, the storage of frame data starts. Received frames for a queue that is already full (there is no empty frame descriptor) are discarded from the Reception FIFO.

(1) Storing Frame Data in the Descriptor Data Area

Frame data for storage are assumed to be in either of the two patterns described below.

- The data for an entire frame will fit in the descriptor data area.
  - In this case, the descriptor type (DESCR.DT) is FSINGLE.
- Frame data to be stored in the descriptor data area arrive in divided form.
  - In this case, FSTART is written to the descriptor type (DESCR.DT) bits of the first descriptor of the frame data to arrive and FMID and FEND are written to the type bits of descriptors for subsequent data.

The descriptor type is updated by the DMAC in the last step of descriptor processing, so software can always access the descriptor assigned to DESCR.DT.

The CPU can write FEMPTY (or FEMPTY_ND) directly to the descriptor type field after processing the stored element. Do not change the descriptor or any part of the descriptor data area after FEMPTY (or FEMPTY_ND) is written to DESCR.DT.

(1) Storing Frame Data for a Whole Single Frame

For a frame with an FSINGLE descriptor, all data for the frame are held at the position defined by DESCR.DPTR. DESCR.DS indicates the length of the received frame.

If DESCR.DS is bigger than the actual size of a received frame, the FSINGLE descriptor is stored in place of the FEMPTY or FEMPTY_ND descriptor after processing.

(2) Storing Frame Data as Divided Frames

Divided frames are handled in the same way as a single frame. A frame stored with divided descriptors must be recombined before use. DESCR.EI is only valid in the last descriptor of the sequence for a divided frame.

NOTE

If the data area size setting in DESCR.DS is not a multiple of four, the number of bytes set in DESCR.DS is fetched from the Reception FIFO and the remaining bytes are used as the next storage area.

After a received frame is divided into different descriptors, each storage element is handled separately, and the descriptor type is assigned by software after processing. Accordingly, an error frame (FEMPTYxxx instead of FMID or FEND) may exist while a descriptor chain is being processed. In such a case, the CPU must postpone processing of the error frame to the next trigger point.

(3) No Data are Stored

The application specification may lead to some types of received frames being unimportant (for example, when the application only requires stream data from the Ethernet frames). Storing frames in divided form makes separating out the unnecessary parts of Ethernet frames possible.

If part of a divided frame is not required, use the FEMPTY_ND descriptor for that part so that it is not stored in the URAM. Not storing the data negates the need for bandwidth on the data bus, improving the overall performance.

When an FEMPTY_ND descriptor is processed, DESCR.DS is set to 0. This brings the frame data section of the descriptor into agreement with the FEMPTY type. DESCR.DS = 0 is for the unique identification of the descriptor after writing.

(2) Flow of Reception Descriptor Processing

Constructing a descriptor chain requires software (see **Figure 31.15**).

In the example in the figure, the variable SWdescr (software descriptor pointer) is a structure to identify a descriptor being processed. SWdescr must be initialized after OPERATION mode is entered and a descriptor base address load request (DLR.LBAq) is executed (condition for starting the flow of software operations).

The frame_processing() function processes the stored data. The function can use SWdescr.DT to check whether processing of a frame is completed. How frame data are processed differs with the application, so create functions that handle processing in accord with the specification.

The processing section is common to all modes of reception. The number of frames processed in response to each trigger can be restricted. When multiple frames have to be processed in a batch, waiting for individual trigger boxes must be skipped for these frames.

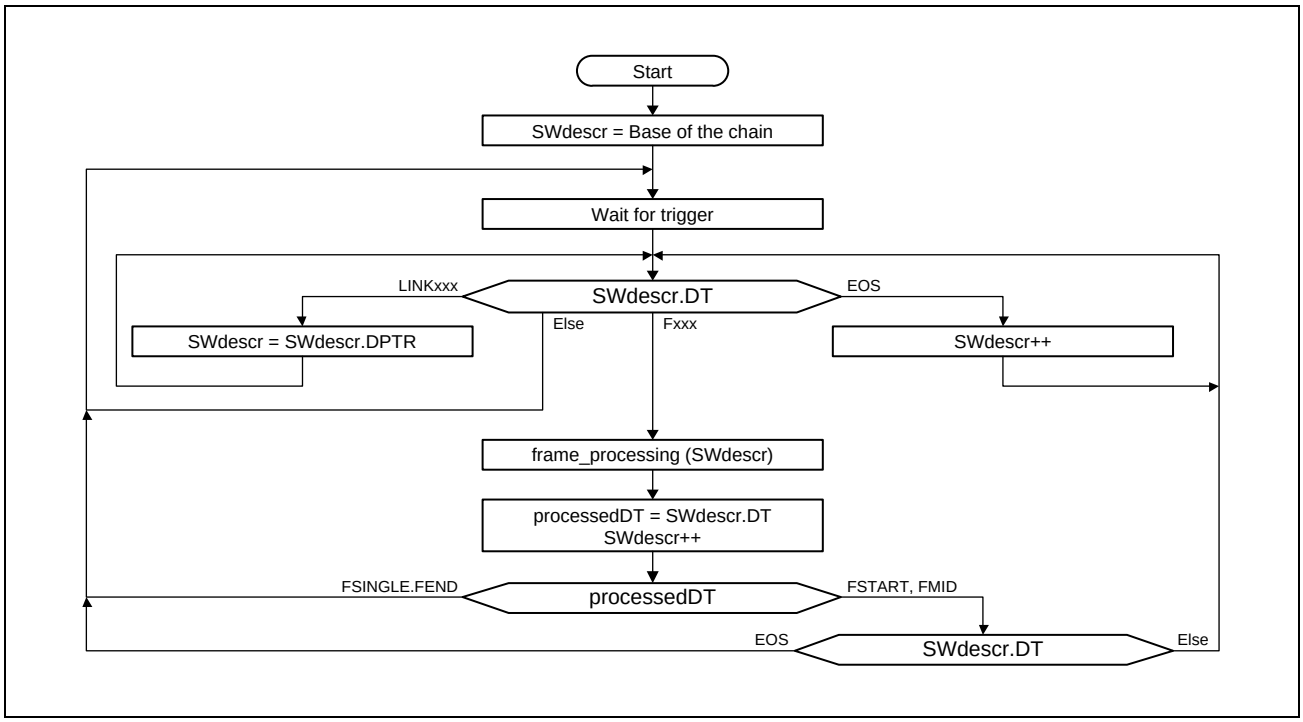


Figure 31.15 Flow of Reception Descriptor Processing

31.5.4 Transmission Control

Areas in the URAM for storing transmission descriptors must also be secured (for descriptors, see **Section 31.5.2, Descriptors**).

The DMAC fetches data from the URAM in accord with the procedure the descriptor describes. The descriptor also retains tag information once the frame has been fetched for transmission. The tag information is used to maintain the relationships between status for the software and the DMAC. The status information for transmitted frames remains accessible after their transmission is completed.

When DMAC started to transfer a frame, it reaches to TOE via Transmission FIFO and TOE FIFO. And after the calculation of transmission frame checksum by TOE, it is transferred to PHY by E-MAC. If any error information is detected by TOE or E-MAC, they are notified to DMAC and stored to MAC Status FIFO/MAC Status FIFO2. The oldest information in them is indicated to MFA/MFA2 of DMAC register.

Figure 31.16 shows the transmission data bus

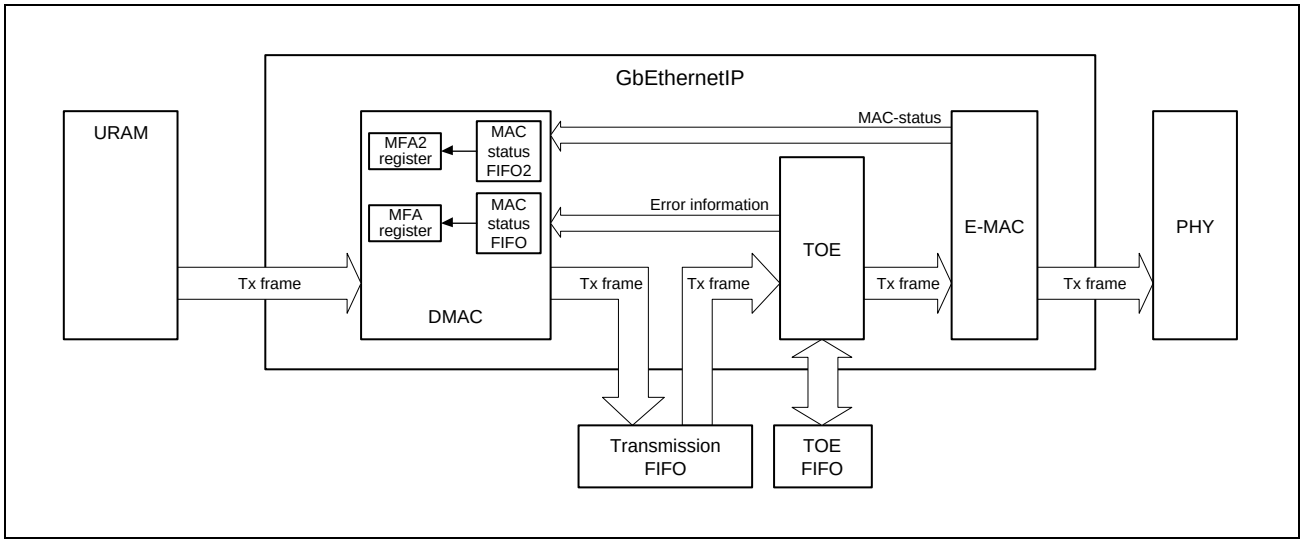


Figure 31.16 Mechanism of General Transmission

31.5.4.1 Setting Up Transmission Descriptors

(1) Transmission Descriptor Type

The type of a descriptor is defined by the descriptor type (DESCR.DT) field.

Table 31.14 shows the descriptor types used in transmission.

Table 31.14 Descriptor Types in Transmission

Descriptor Type (DESCR.DT)	Operation	Write-back
Frame Start (FSTART)	The DMAC fetches the first of the data for the divided frame and proceeds to processing of the next descriptor.	FEMPTY
Frame Middle (FMID)	The DMAC fetches the second or subsequent data for the divided frame and proceeds to processing of the next descriptor.	FEMPTY
Frame End (FEND)	The DMAC fetches the last of the data for the divided frame. The frame of data that has been fetched to the Transmission FIFO is ready for transmission by the E-MAC, and then the DMAC proceeds to processing of the next descriptor.	FEMPTY
Frame Single (FSINGLE)	The DMAC fetches the frame of data. The frame of data that has been fetched to the Transmission FIFO is ready for transmission by the E-MAC, and then the DMAC proceeds to processing of the next descriptor.	FEMPTY
Link (LINK)	Processing proceeds to the descriptor specified by DESCR.DPTR.	LEMPTY
Fixed Link (LINKFIX)	Same as LINK	Not changed
End Of Set (EOS)	This is a transmission stop point defined by software This leads to clearing of the transmit start request bit (TCCR.TSRQ), which stops transmission. When the TCCR.TSRQ is again set to 1 (a new transmission start request is issued), processing proceeds to the next descriptor.	EEMPTY
Frame Empty (FEMPTY)	No frame data are ready for transmission This leads to clearing of the transmit start request bit (TCCR.TSRQ), which stops transmission. When the TCCR.TSRQ is again set to 1 (a new transmission start request is issued), processing starts at this descriptor.	Not changed
Link Empty (LEMPTY)	Same as FEMPTY	Not changed
EOS Empty (EEMPTY)	Same as FEMPTY	Not changed

(2) Configuration of Transmission Frame Data Descriptors

Figure 31.17 shows the configuration of descriptors for use with transmission queues. The transmission-specific fields (DESCR.TSR, and DESCR.TAG) are described in **Table 31.15**.

For the other fields and the descriptor types, see **Section 31.5.2.6, Descriptor Type**.

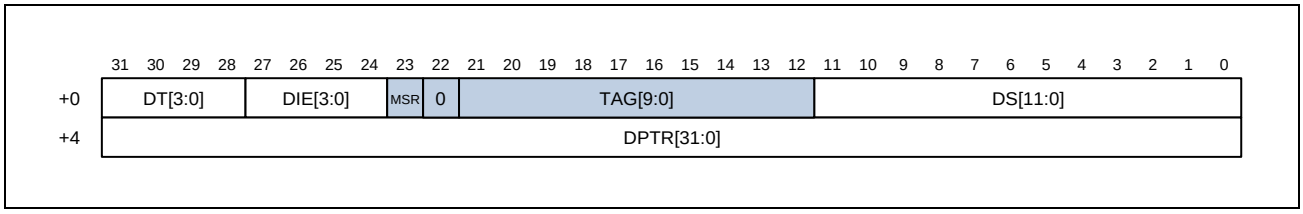


Figure 31.17 Configuration of Descriptor for a Transmitted Frame

Table 31.15 Configuration of a Transmission Descriptor

Bit Name	Function
MSR	MAC Status storage Request This bit configures if MAC transmission status of the frame is stored in MAC status FIFO. 0b: No status information is stored 1b: Status information is to be kept for this frame in MAC status FIFO This bit is considered only if DESCR.DT = FEND or FSINGLE. *Note*: If there is error during transmission of frame then status is always stored.
TAG	Frame Tag This TAG field is used to associate each frame data. Frame TAG is not required but is recommended. This bit is available while the current DESCR.DT is FEND or FSINGLE.

31.5.4.2 Transmission

(1) Transmitting Frames

Setting the transmit start request bit in the transmit configuration control register (TCCR.TSRQ) starts the transfer of frames from the corresponding transmission queue.

The descriptor in the current descriptor address (CDARq.CDA) for the queue (with q = 0) is read first.

If this descriptor is a descriptor for frame transmission (FSINGLE, etc.), the DMAC fetches the frame data from the data area indicated by the descriptor, writes FEMPTY back to the descriptor type (DESCR.DT) bits to indicate completion of this processing, then proceeds to processing of the next descriptor.

If the descriptor is not for transmission, processing is as dictated by the given descriptor (for these descriptors, see the descriptions in **Section 31.5.2, Descriptors**).

If a base address load request is issued for a descriptor chain while it is being processed (by setting 1 in the LBAq bit for transmission queue q that is currently being processed in the descriptor base address load request register, DLR), processing proceeds to the new descriptor chain. Changing the chain does not interrupt frame fetching, but note that frames that have not been fetched from the old chain remain where they are.

Figure 31.18 shows descriptor processing during transmission.

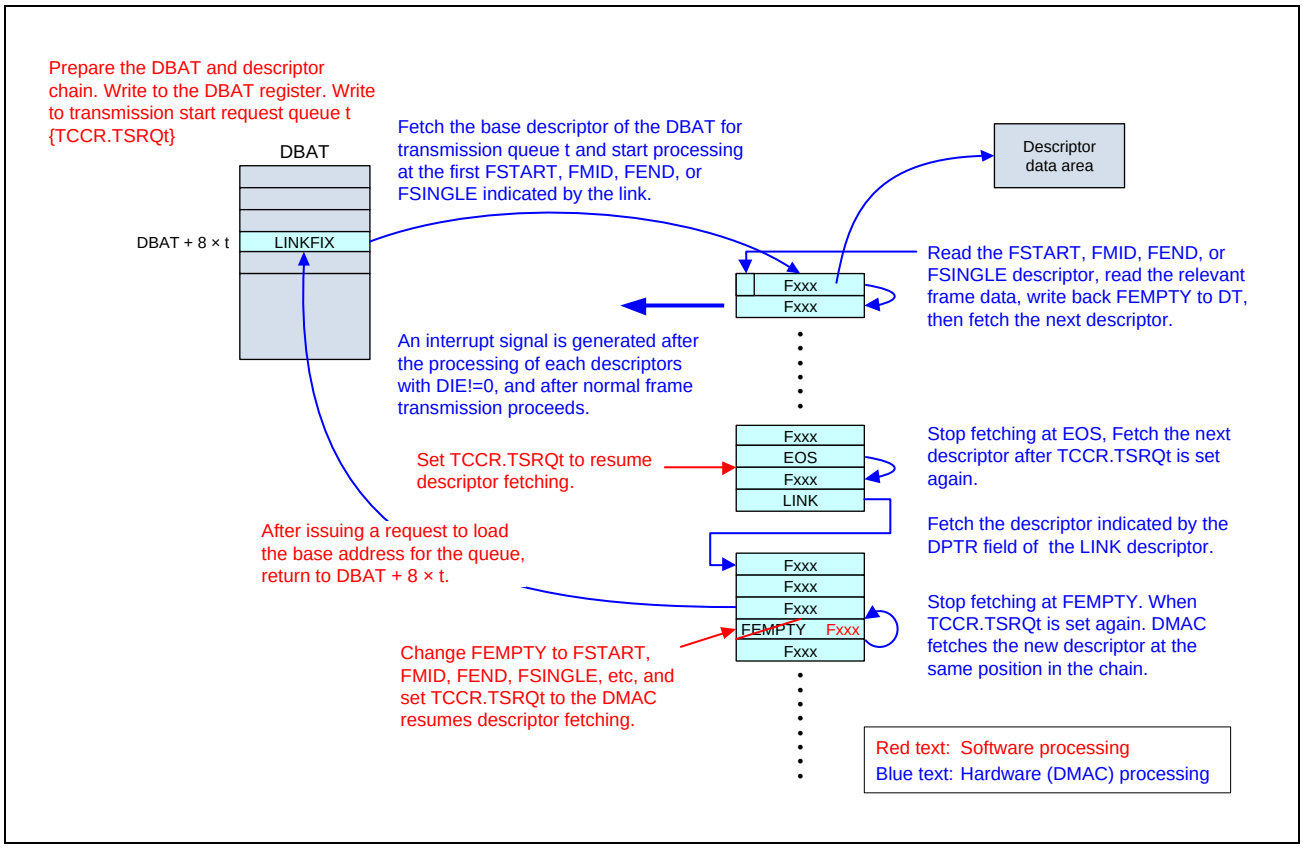


Figure 31.18 Descriptor Processing During Transmission

(2) Examples of Descriptor Usage

(1) Immediate Frame Transmission

Immediate frame transmission is a pattern in which fetching by the DMAC starts whenever software adds data to a queue. FEMPTY descriptors are used as stop points to keep the hardware and software in synchronization.

Create descriptor chains that have FEMTPY descriptors at the stop points.

Figure 31.19 shows the flow for software implementing this pattern. SW should read back written descriptor between changing FEMPTY descriptor before writing to TCCR.TSRQ.

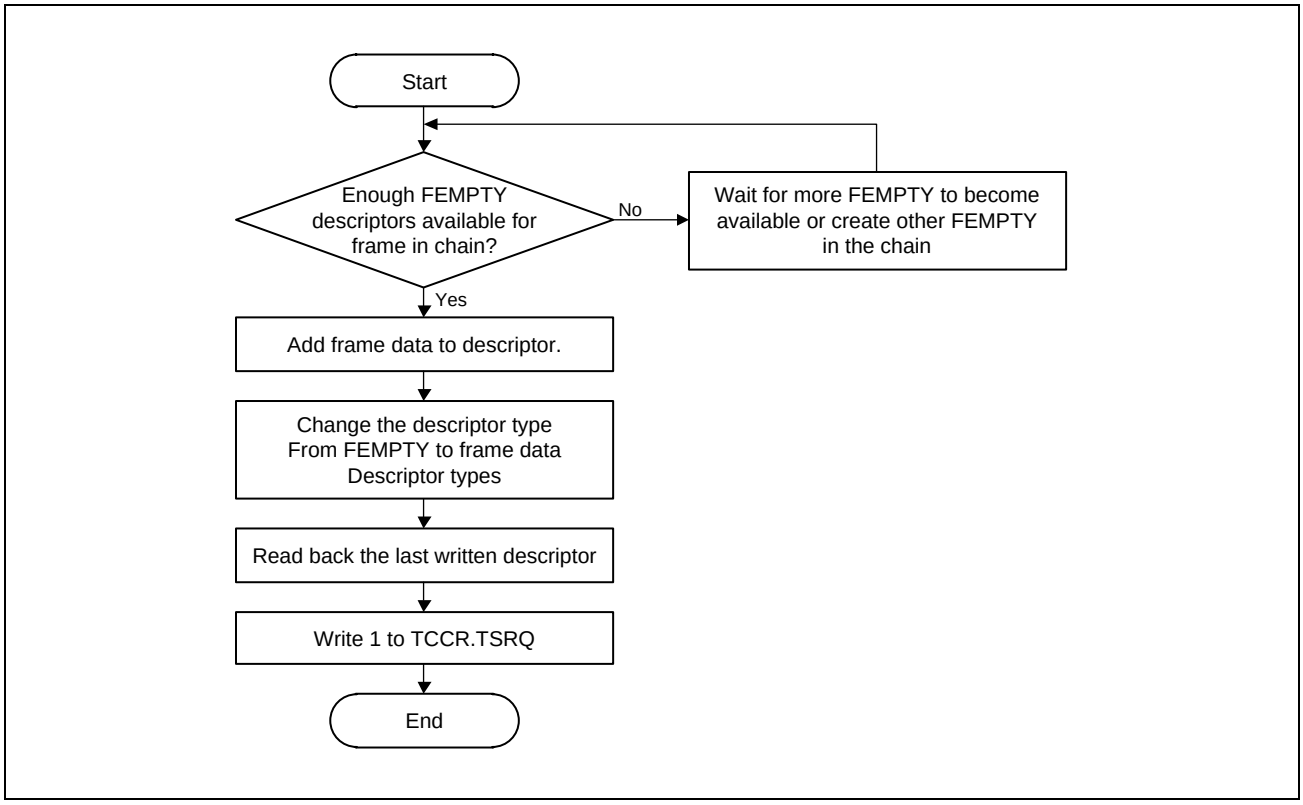


Figure 31.19 Software Flow for Immediate Frame Transmission

Figure 31.20 shows software and DMAC operations for immediate frame transmission.

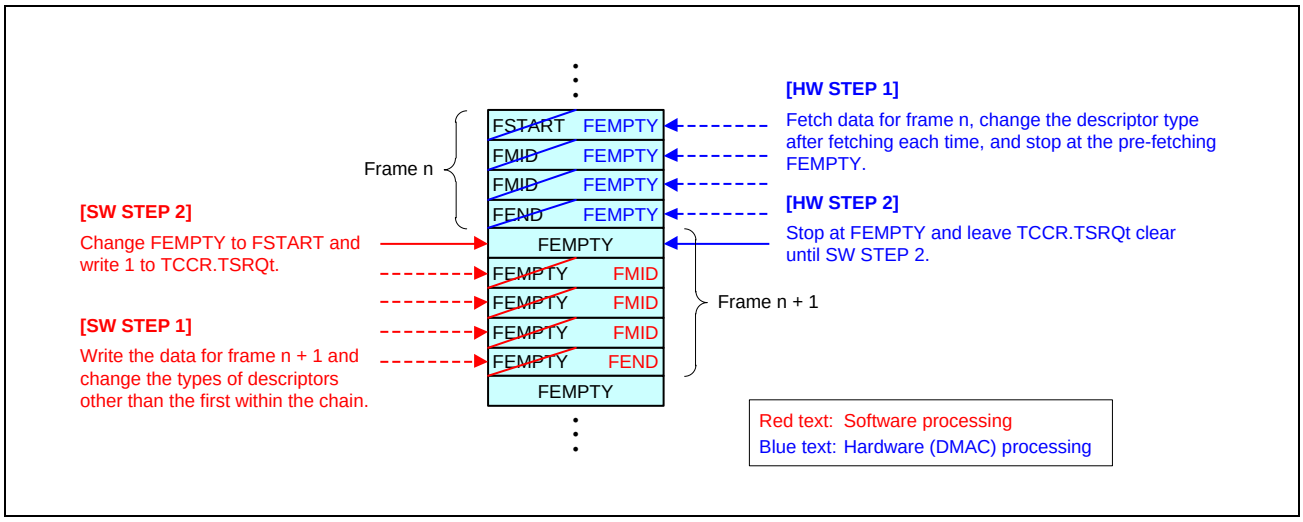


Figure 31.20 Software and DMAC Operations for Immediate Frame Transmission

(2) Frame Set Transmission with Changing of the Active Descriptor Chain

This pattern is used when data are transmitted with a delay for software control to secure bandwidth or for other reasons, rather than immediately transmitted. EOS descriptors are used for the stop points. Start by creating a descriptor chain that has a FEMPTY descriptor as its stop point.

Figure 31.21 shows the software flow in this pattern.

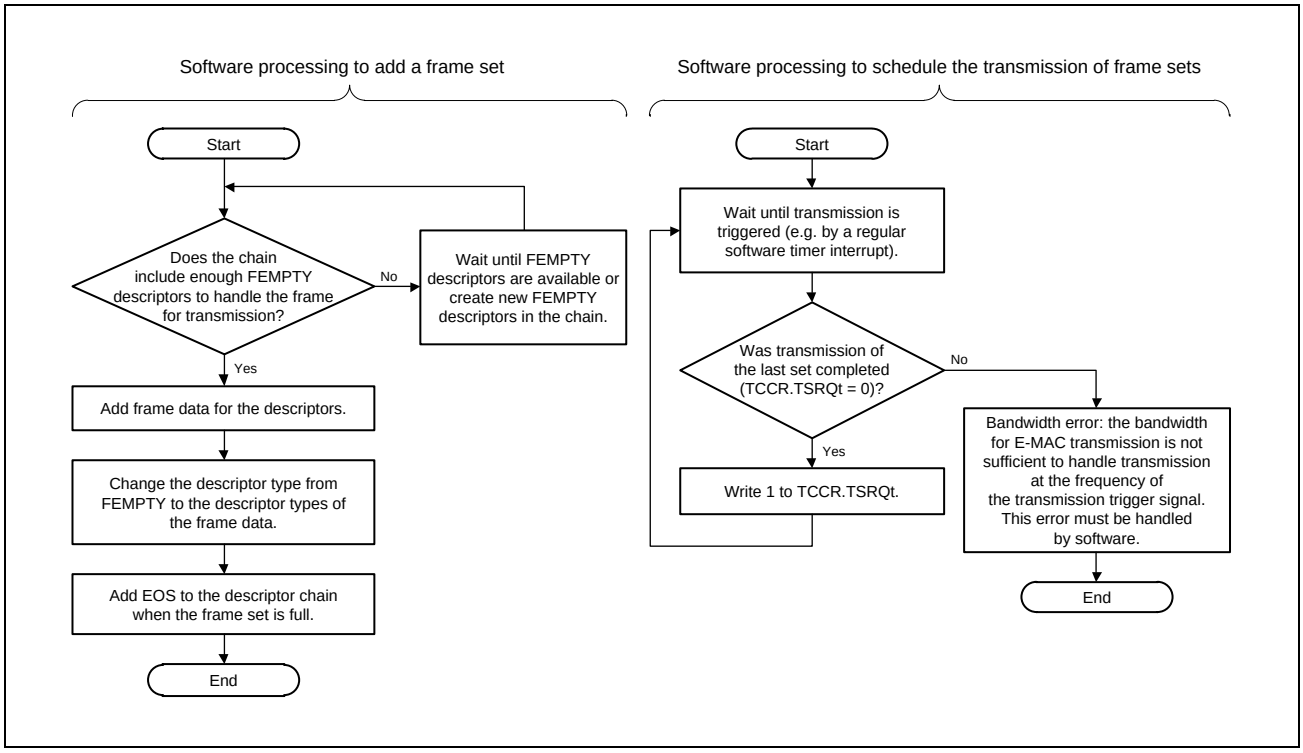


Figure 31.21 Software Flow for Frame Set Transmission with Changing of the Active Descriptor Chain

Figure 31.22 shows software and DMAC operations for frame set transmission.

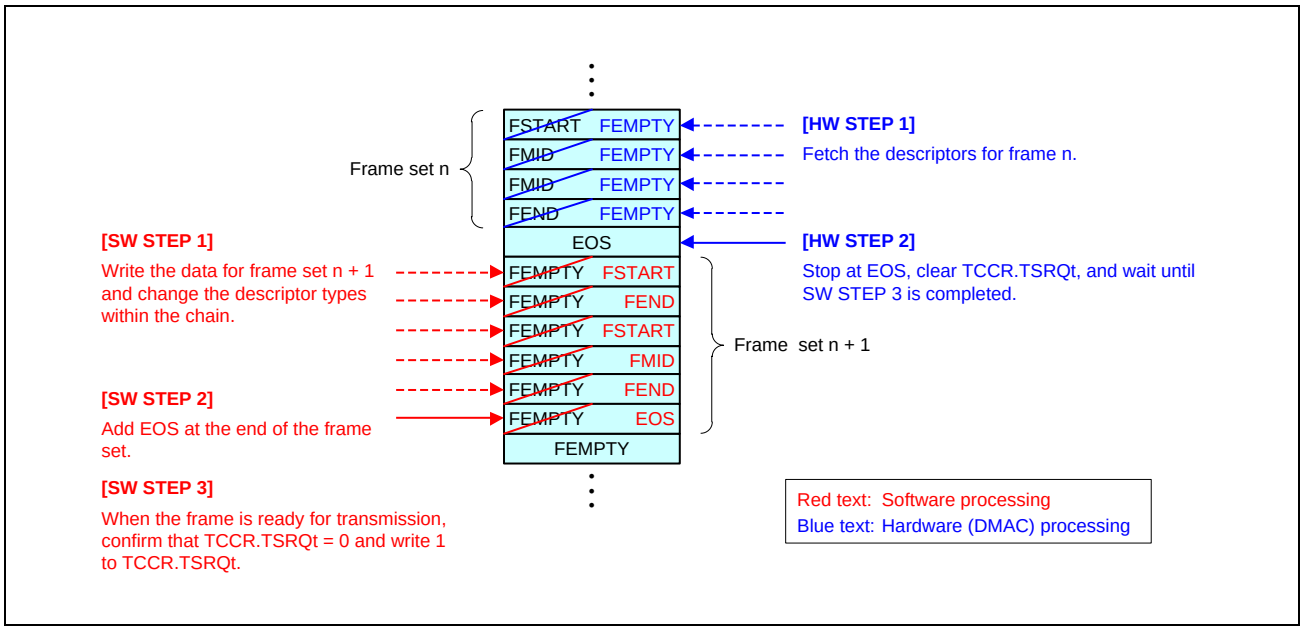


Figure 31.22 SW and DMAC Operations for Frame Set Transmission with Changing of the Active Descriptor Chain

(3) Frame Set Transmission Using a Shadow Descriptor Chain

This pattern is used when data are transmitted with a delay for software control to secure bandwidth or for other reasons, rather than immediately transmitted. Two or more descriptor chains are used. The chains are classified as active or shadow chains. EOS descriptors are used for the stop points.

Create descriptor chains that have FEMPTY descriptors at the stop points.

Figure 31.23 shows the flow for software implementing this pattern.

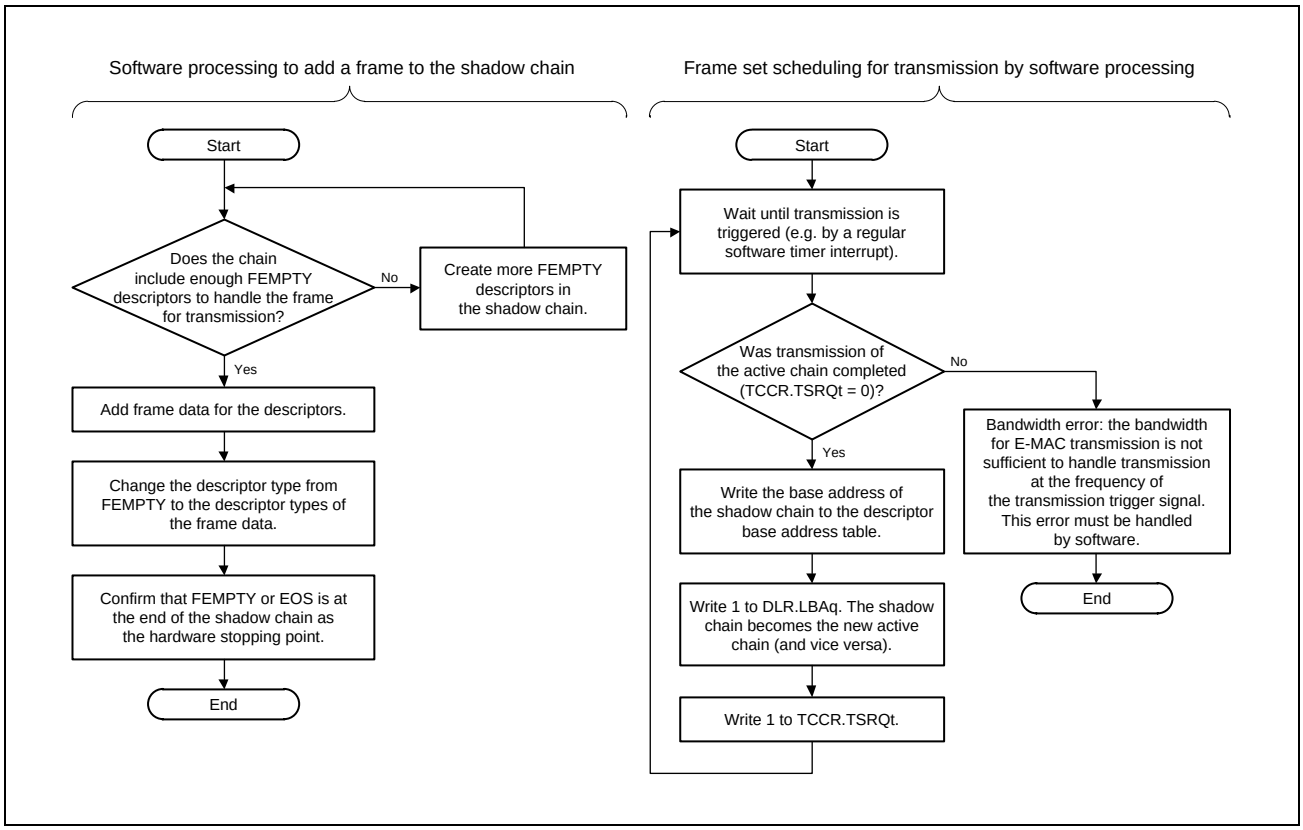


Figure 31.23 Software Flow for Frame Set Transmission Using the Shadow Descriptor Chain

Figure 31.24 shows software and DMAC operations for frame set transmission.

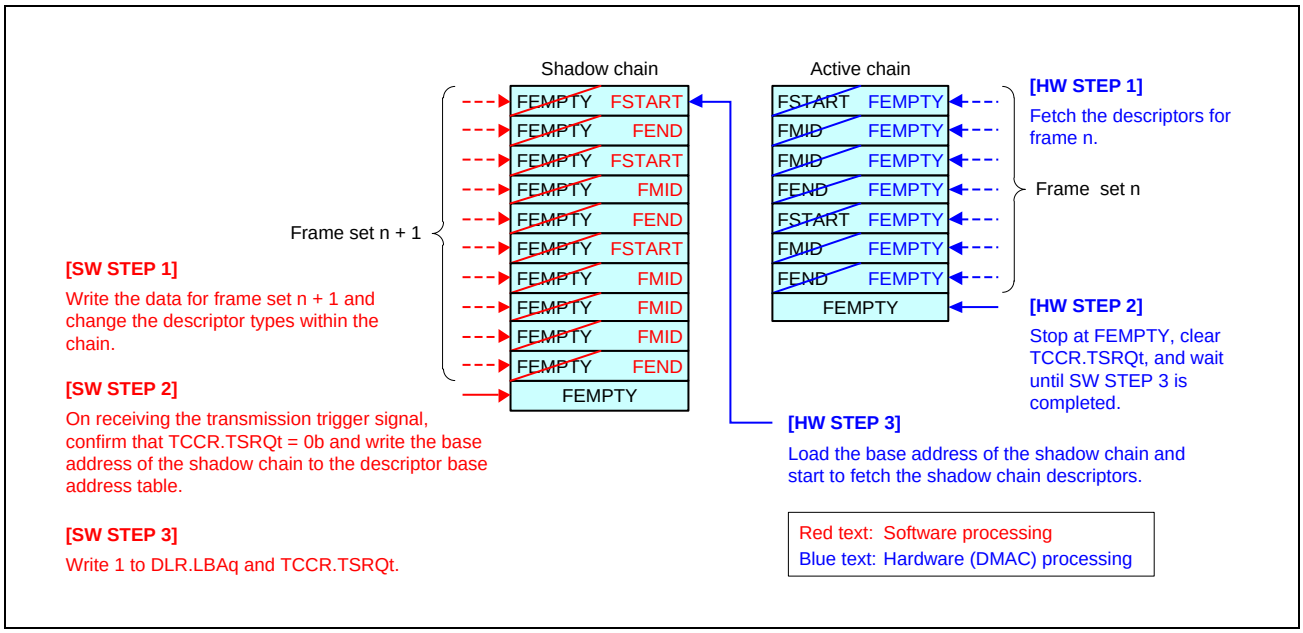


Figure 31.24 SW and DMAC Operations for Frame Set Transmission Using the Shadow Descriptor Chain

31.5.5 Integrity Checking

The DMAC can detect and identify errors produced in the processing of Ethernet frames and in the transfer of frame data for transmission and reception.

31.5.5.1 Integrity Checking in Reception

The aim of integrity checking in reception is preventing the storage of error frames in the URAM. If an error frame is stored in the URAM, software can get information to identify the frame as an error frame.

31.5.5.2 Integrity Checking in Transmission

The purpose of integrity checking in transmission is to prevent the transmission of broken frames.

Since transmission of a frame by the E-MAC can neither be stopped nor disabled once it has started, this check involves intensive monitoring for problems that can arise during fetching.

31.5.5.3 Monitoring in Both Reception and Transmission

(1) Errors in access to the URAM for reading of descriptors

The same descriptor may be processed again because the current descriptor address (CDARq.CDA) was not changed.

If this problem occurs in a divided frame, the sequence may be broken.

In reception:

- The received frame will be lost.
- The same problem will occur for the next frame of data received for the same queue.

In transmission:

- The transmit start request bit in the transmit configuration control register (TCCR.TSRQ) is set to 0.
- The frame will be lost from the Transmission FIFO.

Errors in access to read descriptors from the URAM are detected from the response signal of the AXI-Bus.

(2) Illegal configuration of a descriptor by an application

The same descriptor may be processed again because the current descriptor address (CDARq.CDA) was not changed.

If this problem occurs in a divided frame, the sequence may be broken.

In reception:

- The received frame will be lost.
- The same problem will occur for the next frame of data received for the same queue.

In transmission:

- The transmit start request bit in the transmit configuration control register (TCCR.TSRQ) is set to 0.
- The frame will be lost from the Transmission FIFO.

(3) Errors in access to the URAM for writing of descriptors

As in the case where no error occurs, the current descriptor address (CDARq.CDA) and the transmit start request bit in the transmit configuration control register (TCCR.TSRQ) are updated.

As DESCR.DT was not updated, hardware and software synchronization may have been destroyed.

Errors in access to write descriptors to the URAM are detected from the response signal of the AXI-Bus.

31.5.5.4 Monitoring in Reception

(1) Errors in access to the URAM for writing of data

- As in the case where no error occurs, the current descriptor address (CDARq.CDA) is updated.
- DESCR.EI is set to indicate incorrect contents.
- This problem occurring in a divided frame may break the descriptor sequence, making the queue unusable.

Errors in access to write data or descriptors to the URAM are detected from the response signal of the AXI-Bus.

(2) Error of the Reception FIFO

- Received frames are all invalidated.
- All frames stored as received frames are discarded. At this time, the number of frames and queue information cannot be captured.

If the Reception FIFO RAM faults, DMAC detects the fault as error of the Reception FIFO.

31.5.5.5 Monitoring in Transmission

(1) Errors in Access for Reading Data from the URAM

- Data that have already been fetched are discarded from the Transmission FIFO.
- When an error of this type occurs during processing of an FSINGLE or FEND descriptor:z
  As in the case where no error occurs, the current descriptor address (CDARq.CDA) and the transmit start bit in the transmit configuration control register (TCCR.TSRQ) are updated. Fetching resumes after the error frame.
- When an error of this type occurs during processing of an FSTART or FMID descriptor:
  The current descriptor address (CDARq.CDA) is not updated.
  The transmit start bit in the transmit configuration control register (TCCR.TSRQ) is set to 0.

Errors in access to read data from the URAM are detected from the response signal of the AXI-Bus.

(2) Overflow of the Transmission FIFO

- As in the case where no error occurs, the current descriptor address (CDARq.CDA) and the transmit start bit in the transmit configuration control register (TCCR.TSRQ) are updated. Fetching resumes after the error frame.
- The frame will be discarded from the FIFO.

(3) Damaged Data in the Transmission FIFO

- Fetching is not affected by damaged data.
- Since damaged data from the FIFO is only detected during frame transmission, an error frame may be transmitted.
- The information of MAC Status FIFO and MAC Status FIFO2 will be inconsistent.

If damaged data in the Transmission FIFO is an error due to the Transmission FIFO, this is detected by the DMAC.

31.5.6 Checksum Calculation

TOE calculates Checksum of received frames from E-MAC. And then it is outputted to DMAC. TOE also calculates Checksum of transmission frames from DMAC. And then it is outputted to E-MAC.

To use this function, configure E-MAC as CRC Pass Through Mode (CXR20.RCPT = 1).

TOE supports Checksum Calculation for following part of frames. See **Section 31.5.6.4, Supported Frame for Checksum Calculation** for about the details of Supported Frames.

- IPv4 Header
- IPv4 TCP/UDP/ICMP
- IPv6 TCP/UDP/ICMP

31.5.6.1 Checksum Calculation handling

(1) Reception Handling

The result of Checksum Calculation is attached to last 4 bytes of Ethernet Frames like **Figure 31.25**. And then the handled frames are transferred to memory by DMAC. If the frame does not have checksum error at the part of IPv4 Header or TCP/UDP/ICMP, the value of “H’0000” is attached to each part as the result of Checksum Calculation. The case of Unsupported Frame, the value of “H’FFFF” is attached. For example, if the part of IP Header is unsupported, “H’FFFF” is set to both field of IPv4 Header and TCP/UDP/ICMP. The case of IPv6, IPv4 Header field is always set to “H’FFFF”. And the set value of TCP/UDP/ICMP field depends on configuration for support conditions.

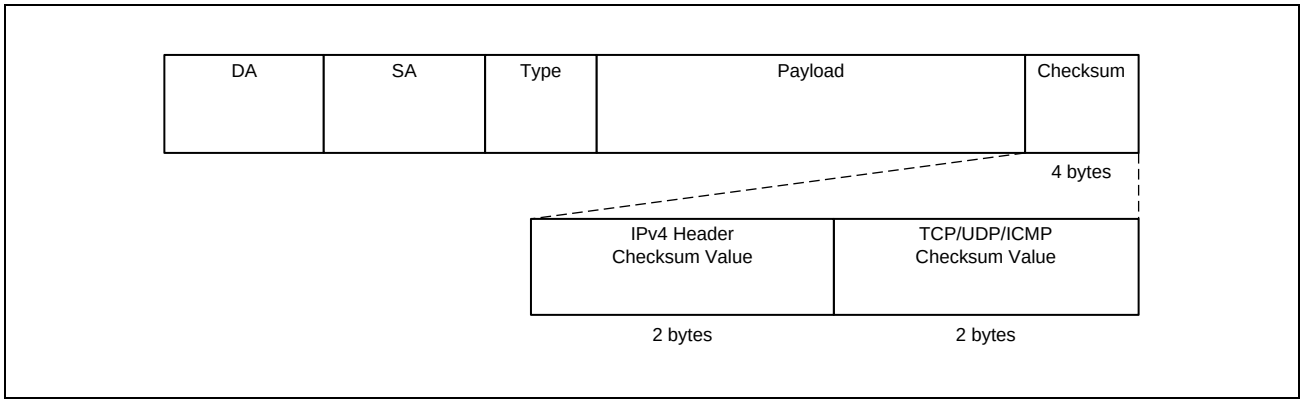


Figure 31.25 The Field of Checksum Attaching Field (Ex. Ethernet Frame of DIX)

(2) Transmission Handling

The result of Checksum Calculation is set to the Checksum field of each IPv4 Header/TCP/UDP/ICPM. For the Unsupported Frames, those fields are not changed. If a transmission frame is an UDP frame of IPv4 and its Checksum value in the UDP Header field is “H’0000”, TOE does not calculate Checksum for UDP part of this frame. Because Checksum for IPv4 UDP frame is defined as optional function in standards. If a transmission IPv4 UDP frame does not need to Checksum Value, set “H’0000” to the Checksum field in the UDP Header.

31.5.6.2 Error Notification

When TOE detects any errors by Checksum Calculation, the error information is notified to DMAC. The errors are detected by TOE are followings:

- Unsupported Transmission Frame Error (This error information is indicated to “MFA” register)
- Unsupported Reception Frame Error (This error information is stored to Descriptor by Write Back)
- Reception Frame has Checksum Error (This error information is stored to Descriptor by Write Back)

31.5.6.3 Individually Configuration by the frame type

The Enable/Disable setting of Checksum Calculation is configurable by CSR1 and CSR2 register. And CSR1 and CSR2 register are able to configure Enable/Disable setting individually for frame type. For example, it is possible to disable the checksum calculation for only UDP part of IPv4 frame.

The relationship between the Frame Type and Error Notification to DMAC are indicated at **Table 31.16** and **Table 31.17**. **Table 31.16** is for Transmission Frames. **Table 31.17** is for Reception Frames. The condition of both tables is that the checksum calculation for all frame type is enabled by CSR1 and CSR2 register.

If the checksum calculation of each Frame Type is configured as Disable individually by CSR1 and CSR2 register, the operation is followings. (Following descriptions are for the normal frames)

(1) Disabled Transmission Checksum Calculation

TOE notifies an Unsupported Transmission Frame Error to DMAC. And the Checksum Values which Type is configured as “Disable” are not changed.

(2) Disabled Reception Checksum Calculation

The frames which have Disabled configuration Types are handled as Unsupported Frame. So, the error notification to DAMC is same as the case of Unsupported Frames.

If the other checksum field which is configured as Enable has Checksum Error, Received Checksum Error is notified to DMAC. Received Unsupported Error also is notified. The value of “H’FFFF” is set to the checksum field which is configured as Disable.

Table 31.16 Relationship between Transmission Frame Type and Error Notification to DMAC

Transmission Frame Type						Checksum Operation		
VLAN-tag		IP		Transport protocol		Error Notification to DMAC	Checksum Value	
Number of tags	TPID	Protocol	Condition	Protocol	Condition	Unsupported Frame Error	IPv4 Header	TCP/UDP/ICMP
0 or 1	Nothing or H'8100	IPv4	Not Fragmented Packet	TCP/ICMP	—	Non-Asserted	Calculated Value	Calculated Value
				UDP	Checksum Field is not H'0000	Non-Asserted	Calculated Value	Calculated Value
					Checksum Field is H'0000	Non-Asserted	Calculated Value	Through (H'0000)
				Except TCP/ICMP/UDP	—	Asserted	Calculated Value	—
			Fragmented Packet	don't care	don't care	Asserted	Calculated Value	—
		IPv6	Non-Extended Header or The number of Supported Extended Header is from 1 to 8*1	TCP/ICMP	—	Non-Asserted	—	Calculated Value
				UDP	—	Non-Asserted	—	Calculated Value
				Except TCP/ICMP/UDP	—	Asserted	—	—
			The number of Supported Extended Header is 9 or more*1	don't care	don't care	Asserted	—	—
			Has Unsupported Extended Header*1	don't care	don't care	Asserted	—	—
		Except IPv4/IPv6	—	don't care	don't care	Asserted	—	—
1	Except H'8100	don't care	don't care	don't care	don't care	Asserted	—	—
2 or more	don't care	don't care	don't care	don't care	don't care	Asserted	—	—
Illegal Frames*2						Asserted	—	—

Note 1. See the **Section 31.5.6.4 Supported Frame for Checksum Calculation** for about the Supported Extended Headers.

Note 2. The Frame Length is shorter than the result of Frame Analysis by TOE.

Table 31.17 Relationship between Reception Frame Type and Error Notification to DMAC

Reception Frame Type						Field of Checksum Error	Checksum Operation			
VLAN-tag		IP		Transport protocol			Error Notification to DMAC		Checksum Value	
Number of tags	TPID	Protocol	Condition	Protocol	Condition		Checksum Error	Unsupported Frame Error	IPv4 Header	TCP/ UDP/ ICMP
0 or 1	Nothing or H'8100	IPv4	Not Fragmented Packet	TCP/ICMP	—	Non-Error	Non-Asserted	Non-Asserted	H'0000	H'0000
					—	IPv4 Header	Asserted	Non-Asserted	Calculated Value	H'0000
					—	TCP/ICMP	Asserted	Non-Asserted	H'0000	Calculated Value
					—	IPv4 Header and TCP/ICMP	Asserted	Non-Asserted	Calculated Value	Calculated Value
				UDP	Checksum Field is not H'0000	Non-Error	Non-Asserted	Non-Asserted	H'0000	H'0000
						IPv4 Header	Asserted	Non-Asserted	Calculated Value	H'0000
						UDP	Asserted	Non-Asserted	H'0000	Calculated Value
						IPv4 Header and UDP	Asserted	Non-Asserted	Calculated Value	Calculated Value
					Checksum Field is H'0000	Non-Error	Non-Asserted	Non-Asserted	H'0000	H'FFFF
						IPv4 Header	Asserted	Non-Asserted	Calculated Value	H'FFFF
				Except TCP/ ICMP/ UDP	—	Non-Error	Non-Asserted	Asserted	H'0000	H'FFFF
					—	IPv4 Header	Asserted	Asserted	Calculated Value	H'FFFF
			Fragmented Packet	don't care	don't care	Non-Error	Non-Asserted	Asserted	H'0000	H'FFFF
				don't care	don't care	IPv4 Header	Asserted	Asserted	Calculated Value	H'FFFF
		IPv6	Non-Extended Header or The number of Supported Extended Header is from 1 to 8*1	TCP/ICMP	—	Non-Error	Non-Asserted	Non-Asserted	H'FFFF	H'0000
					—	TCP/ICMP	Asserted	Non-Asserted	H'FFFF	Calculated Value
				UDP	Checksum Field is not H'0000	Non-Error	Non-Asserted	Non-Asserted	H'FFFF	H'0000
						UDP	Asserted	Non-Asserted	H'FFFF	Calculated Value
					Checksum Field is H'0000	—	Asserted	Non-Asserted	H'FFFF	H'FFFF
				Except TCP/ ICMP/ UDP	—	—	Non-Asserted	Asserted	H'FFFF	H'FFFF
			The number of Supported Extended Header is 9 or more*1	don't care	don't care	—	Non-Asserted	Asserted	H'FFFF	H'FFFF
			Has Unsupported Extended Header*1	don't care	don't care	—	Non-Asserted	Asserted	H'FFFF	H'FFFF
		Except IPv4/IPv6	—	don't care	don't care	—	Non-Asserted	Asserted	H'FFFF	H'FFFF
1	Except H'8100	don't care	don't care	don't care	don't care	—	Non-Asserted	Asserted	H'FFFF	H'FFFF
2 or more	don't care	don't care	don't care	don't care	don't care	—	Non-Asserted	Asserted	H'FFFF	H'FFFF
Illegal Frames*2						—	Non-Asserted	Asserted	H'FFFF	H'FFFF

Note 1. See the **Section 31.5.6.4 Supported Frame for Checksum Calculation** for about the Supported Extended Headers.

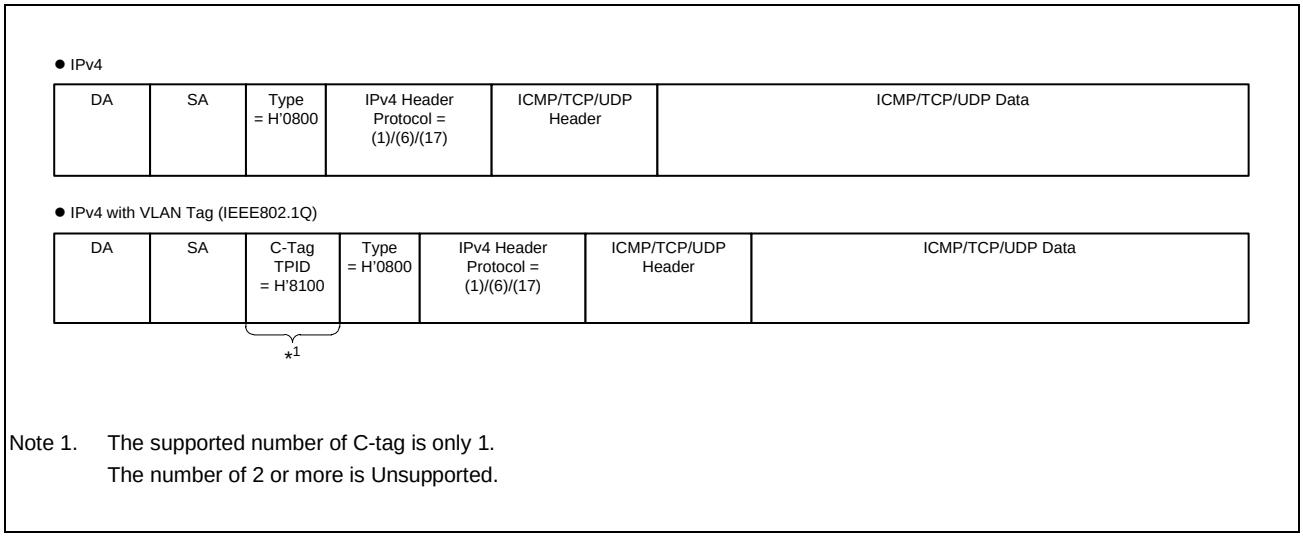
Note 2. The Frame Length is shorter than the result of Frame Analysis by TOE. Or the case of E-MAC detected Errors.

31.5.6.4 Supported Frame for Checksum Calculation

Following Frames are example of Supported/Unsupported Frame Formats for Checksum Calculation.

(1) For IPv4 Checksum Calculation

Figure 31.26 is Supported Frame Format for the Checksum Calculation of IPv4 Header and IPv4 TCP/UDP/ICMP.

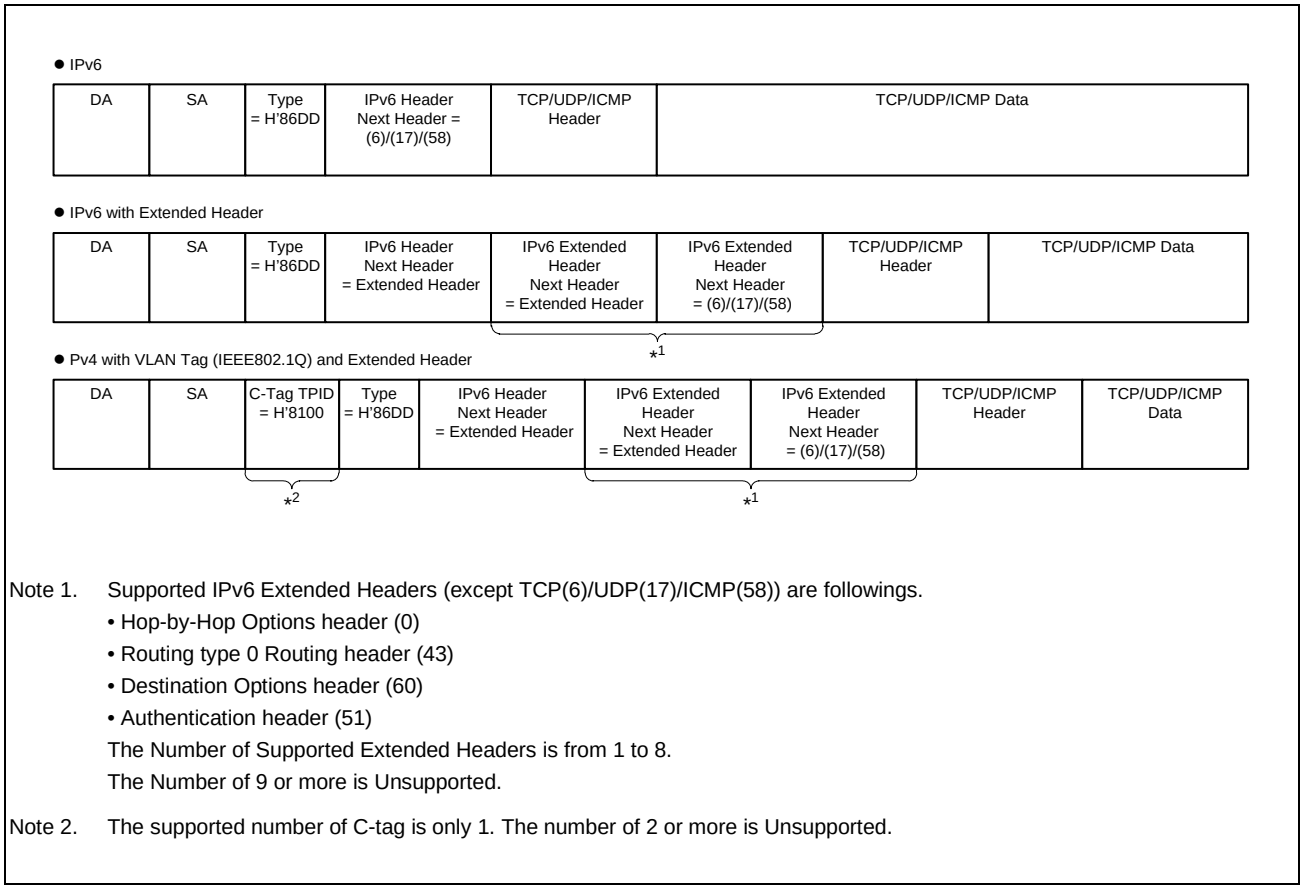


Note 1. The supported number of C-tag is only 1.
The number of 2 or more is Unsupported.

Figure 31.26 Supported Frame Format for IPv4 Checksum Calculation

(2) For IPv6 Checksum Calculation

Figure 31.27 is Supported Frame Format for the Checksum Calculation of IPv6 TCP/UDP/ICMP.



Note 1. Supported IPv6 Extended Headers (except TCP(6)/UDP(17)/ICMP(58)) are followings.
- Hop-by-Hop Options header (0)
- Routing type 0 Routing header (43)
- Destination Options header (60)
- Authentication header (51)

The Number of Supported Extended Headers is from 1 to 8.
The Number of 9 or more is Unsupported.

Note 2. The supported number of C-tag is only 1. The number of 2 or more is Unsupported.

Figure 31.27 Supported Frame Format for IPv6 Checksum Calculation

(3) Unsupported Frame Format (example)

Figure 31.28 is an Example Unsupported Frame Format.

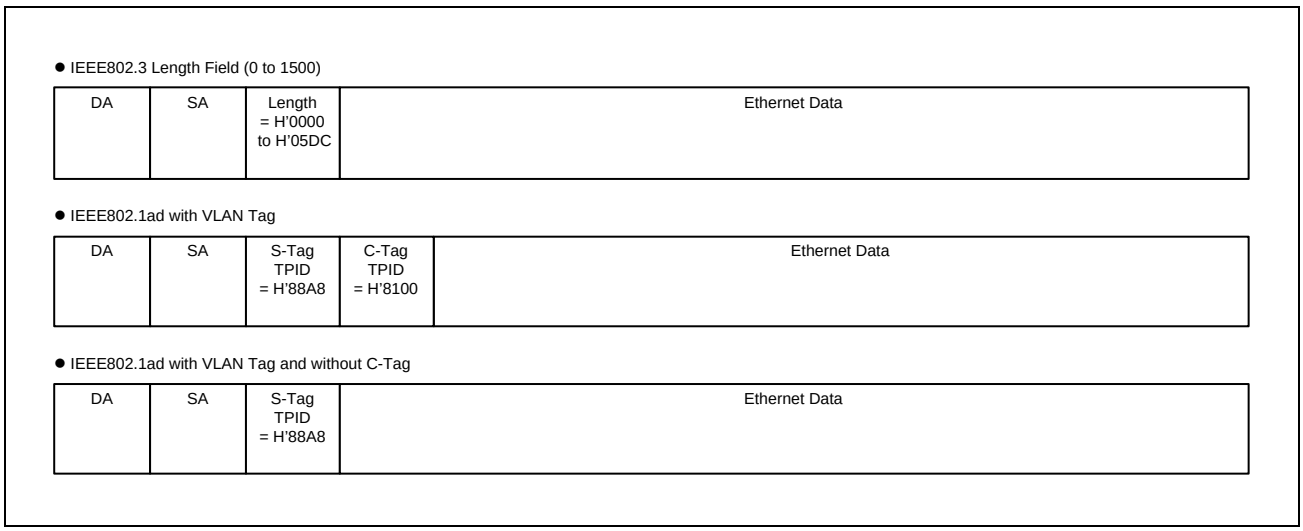


Figure 31.28 Unsupported Frame Format (Example)

31.5.6.5 Checksum Calculation Flow

Figure 31.29 and **Figure 31.30** are Flows for Checksum Calculation by TOE.

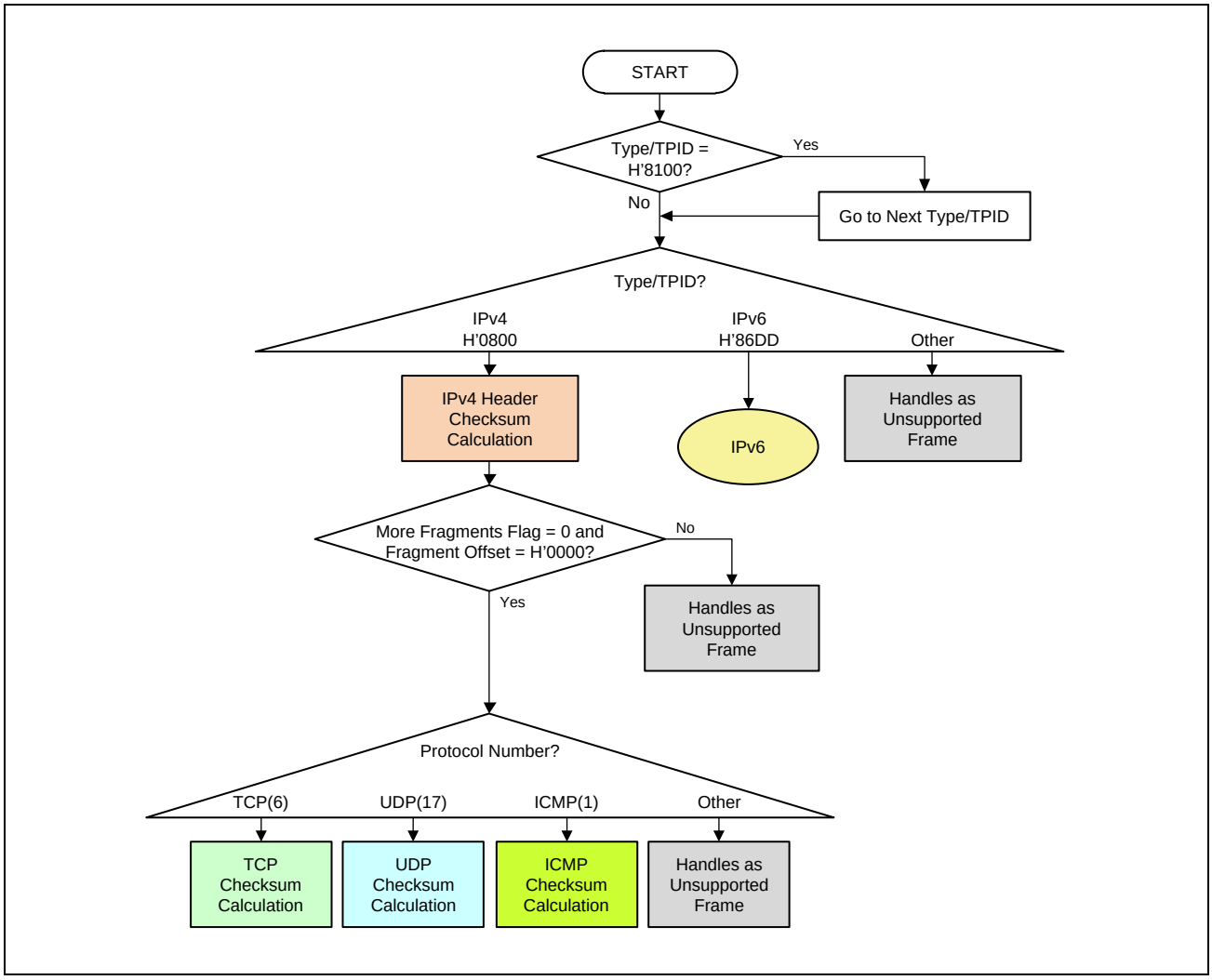


Figure 31.29 Checksum Calculation Flow (Part1)

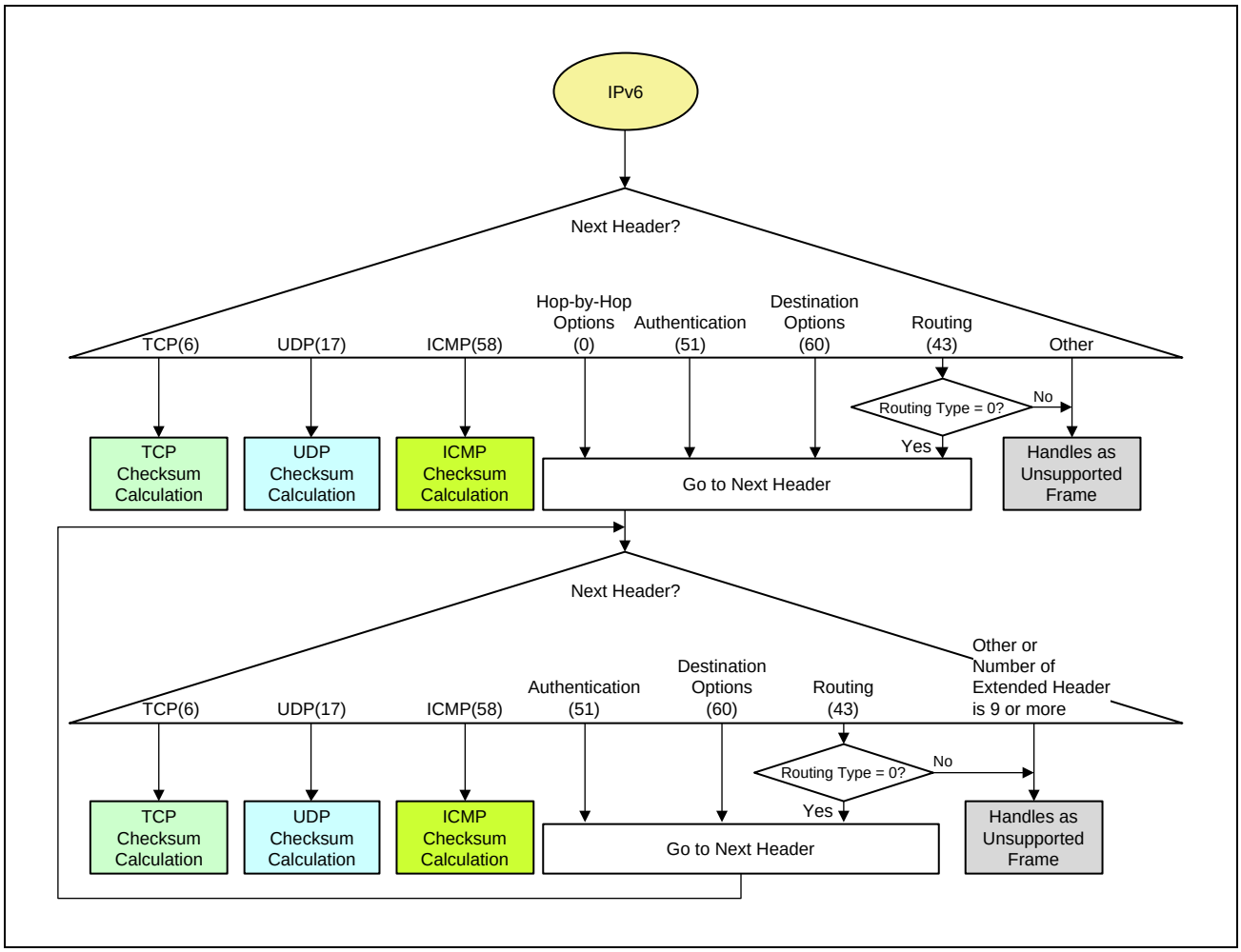


Figure 31.30 Checksum Calculation Flow (Part2)

31.5.7 Received Frame Filter

TOE has Received Frame Filters. When TOE received the frame which is not match to filters, TOE discards (abort) that frame. TOE is also able to assert an interrupt if TOE received the frame which is match to filters.

The Possible Configuration of Filter Condition are followings.

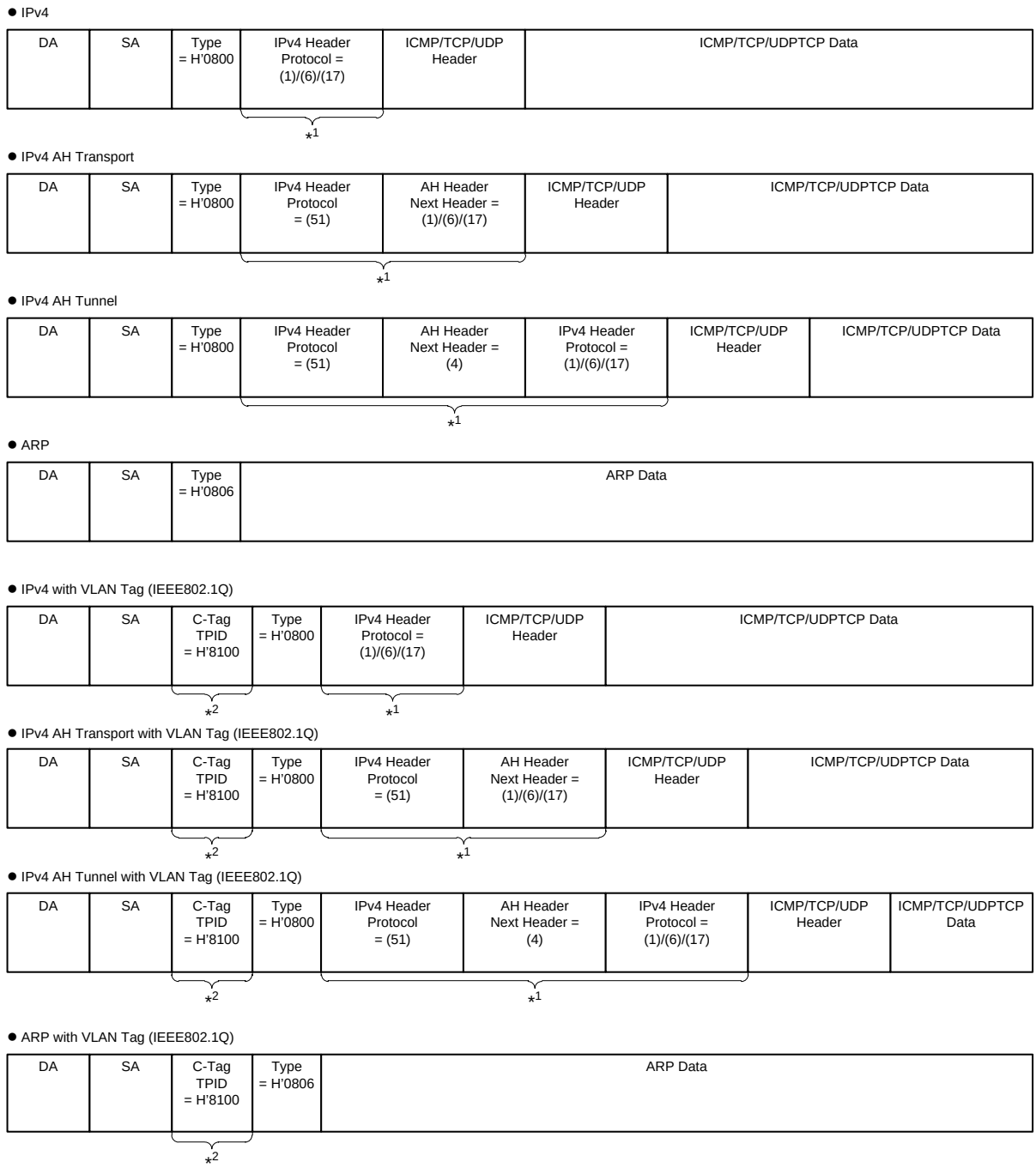
- Ethernet Type
- IP Protocol No.
- UDP destination port No.
- Destination MAC address (Unicast)
- Destination MAC address (Broadcast)
- Destination MAC address (Multicast)
- ARP Request of Local Station
- Neighbor Solicitation of Local Station
- Except IPv6 Next Header (Analyzable)

31.5.7.1 Supported Frame for Received Frame Filter

Followings are Supported /Unsupported Frame Formats by Received Frame Filter.

(1) For IPv4 Frame Filter

Figure 31.31 is Supported Frame format of IPv4.



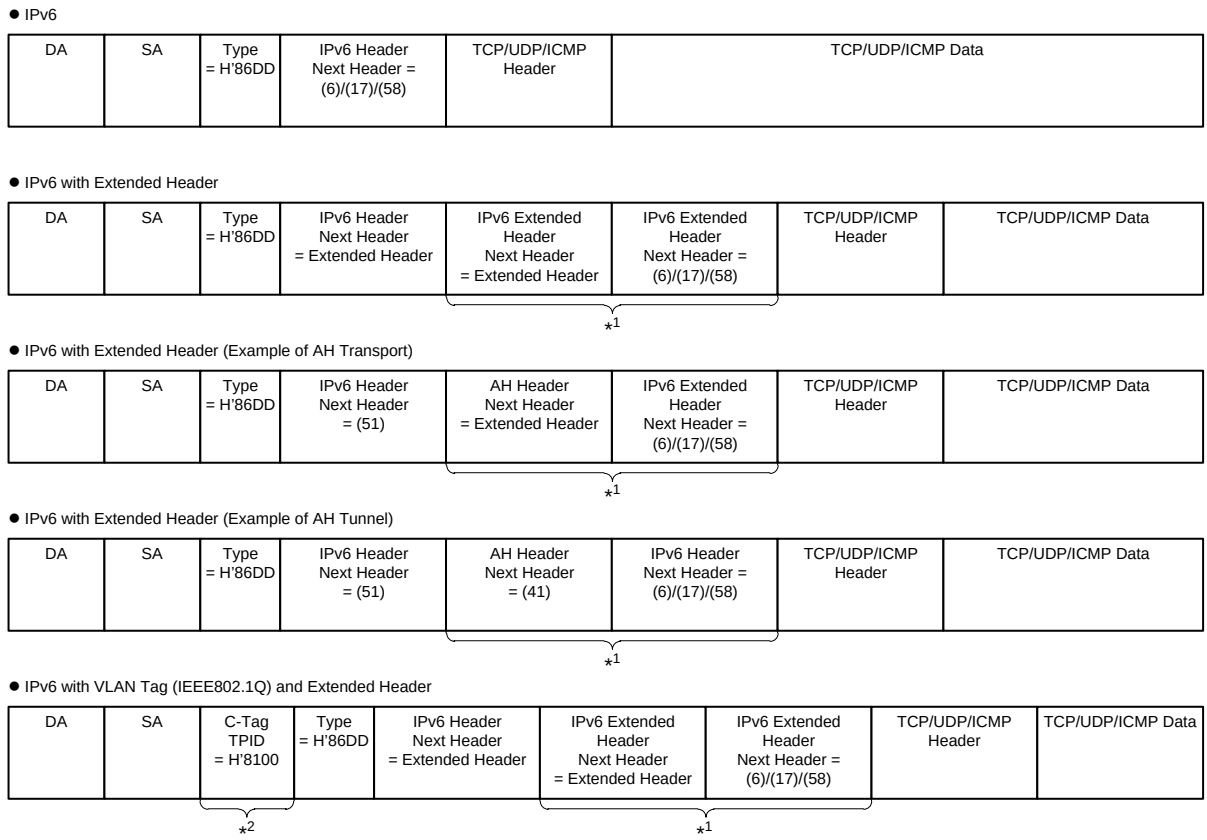
Note 1. Frame analysis is stopped if the Protocol No. is not ICMP (1), IPinIP (4), TCP (6), UDP (17) or AH (51)

Note 2. The supported number of C-tag is only 1.
The number of 2 or more is Unsupported.

Figure 31.31 Supported Frame of IPv4

(2) For IPv6 Frame Filter

Figure 31.32 is Supported Frame format of IPv6.



Note 1. Supported Extended Header (except TCP (6), UDP (17) and ICMP (58) are followings.
- Hop-by-Hop Options header (0)
- IPv6 (41)
- Routing type 0 Routing header (43)
- Flag (44)
- Authentication header (51)
- Destination Options header (60)

Others are Unsupported and Frame analysis is stopped.

Note 2. The supported number of C-tag is only 1.
The number of 2 or more is Unsupported.

Figure 31.32 Supported Frame Format of IPv6

(3) Unsupported Frame Format (Example)

Figure 31.33 is Example Unsupported Frame Format.

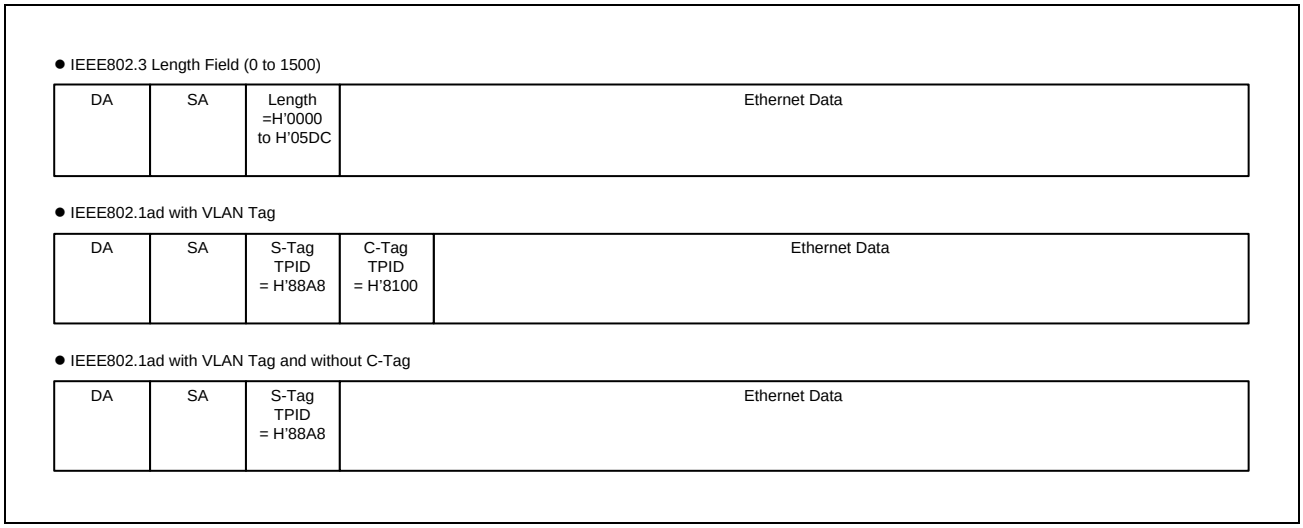


Figure 31.33 Example Unsupported Frame Format

31.5.7.2 Received Frame Filtering Flow

Figure 31.34 to **Figure 31.40** are Received Frame Filtering Flows.

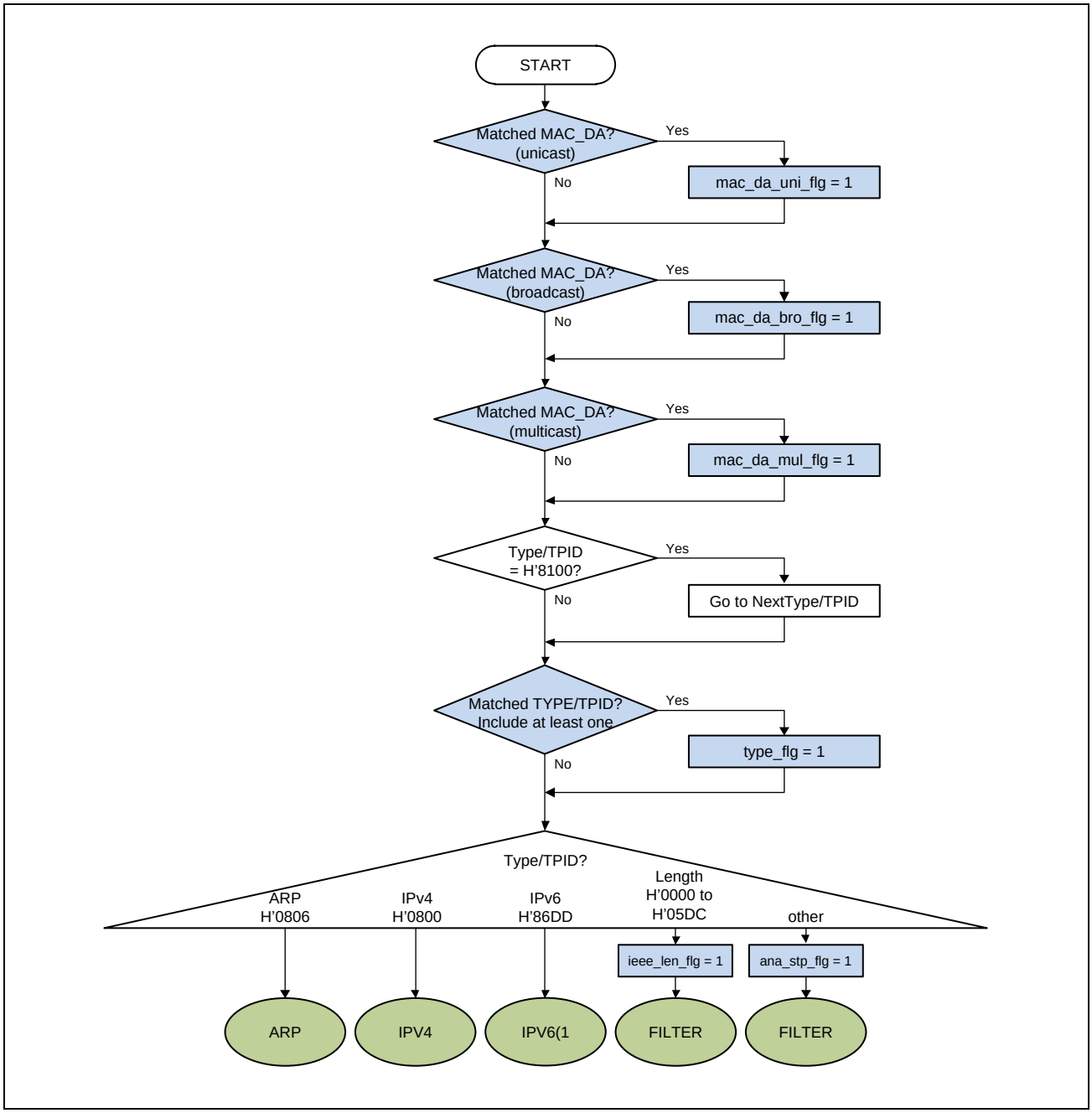


Figure 31.34 Received Frame Filtering Flow (Part1)

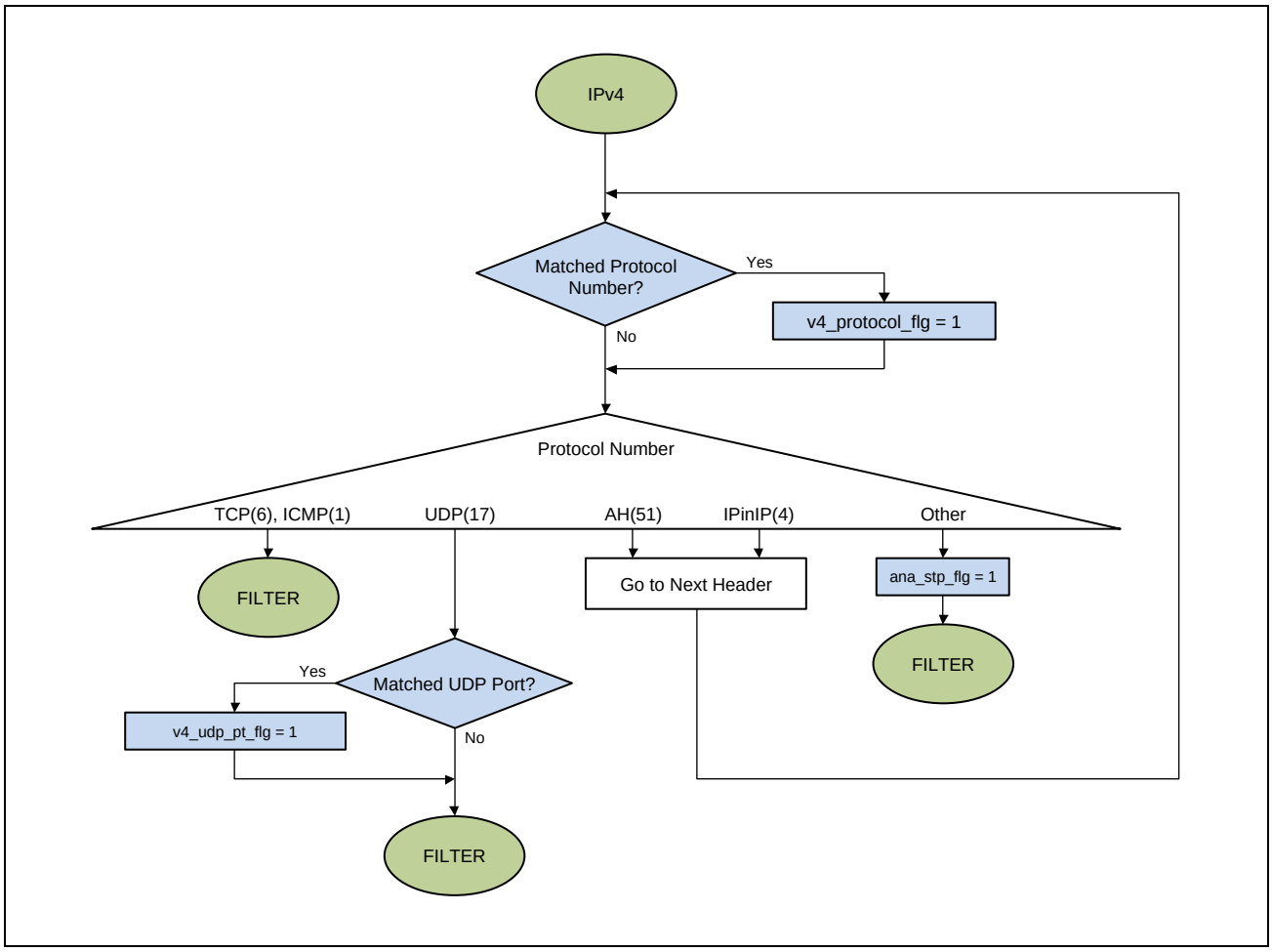


Figure 31.35 Received Frame Filtering Flow (Part2)

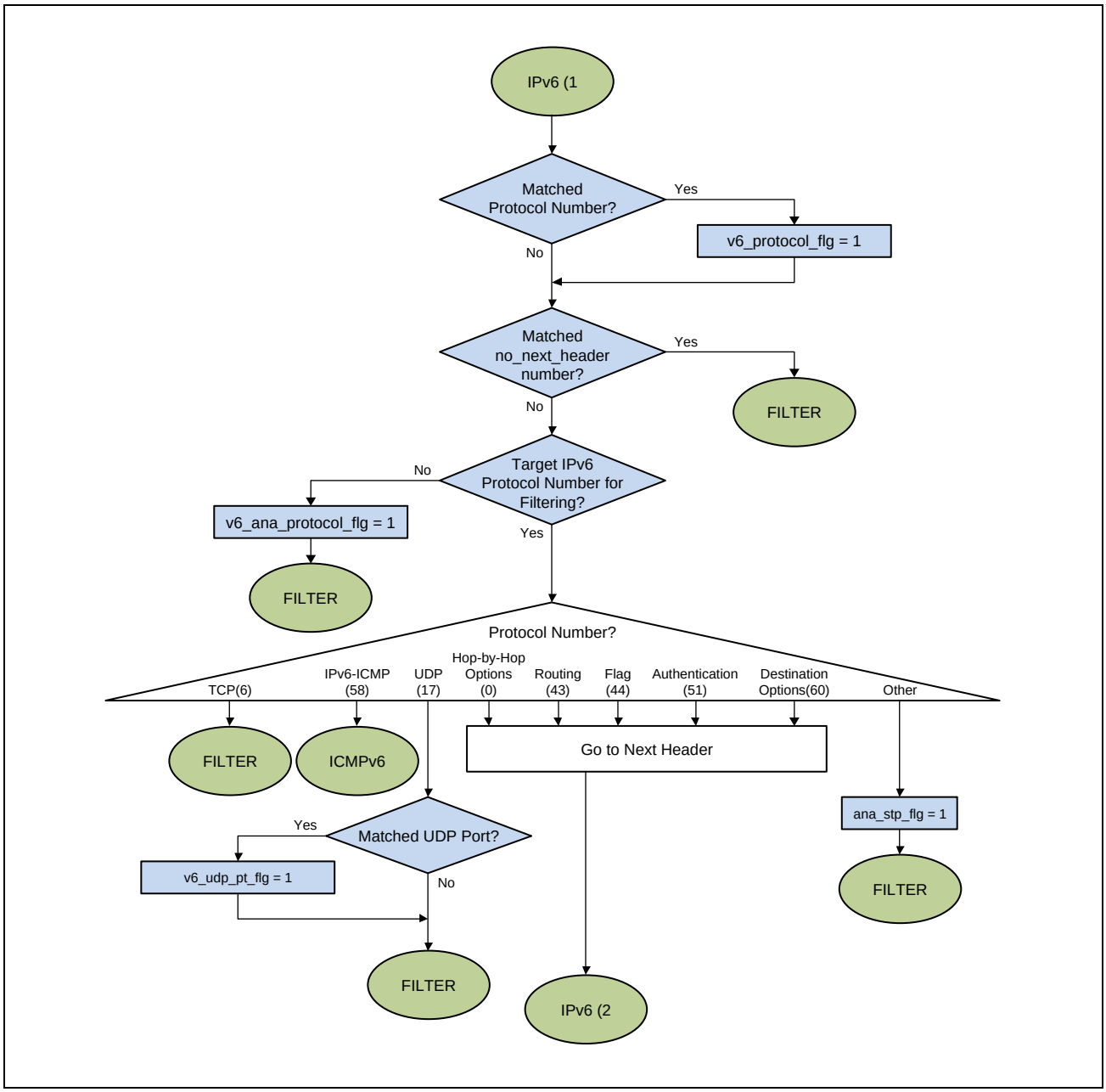


Figure 31.36 Received Frame Filtering Flow (Part3)

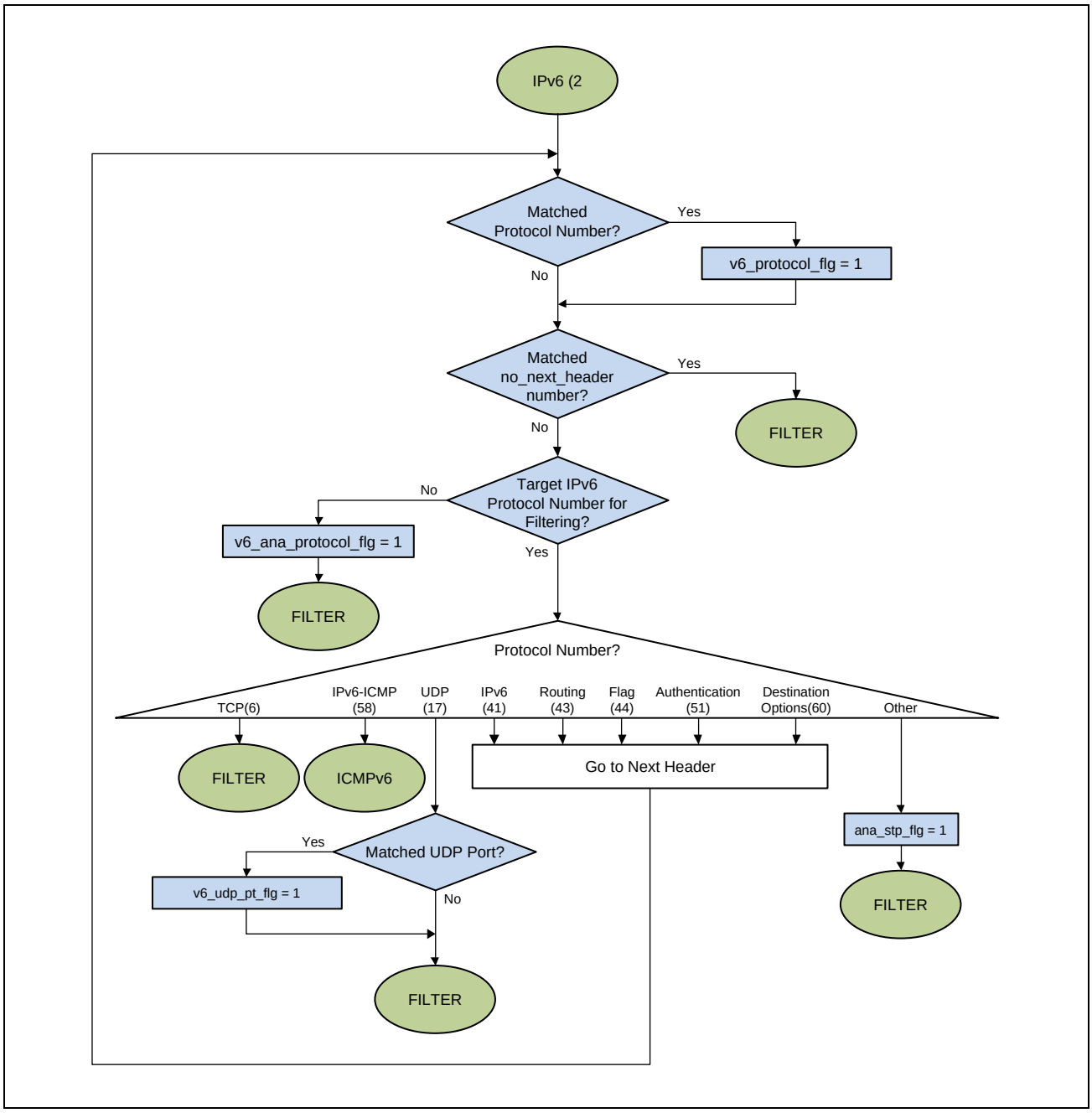


Figure 31.37 Received Frame Filtering Flow (Part4)

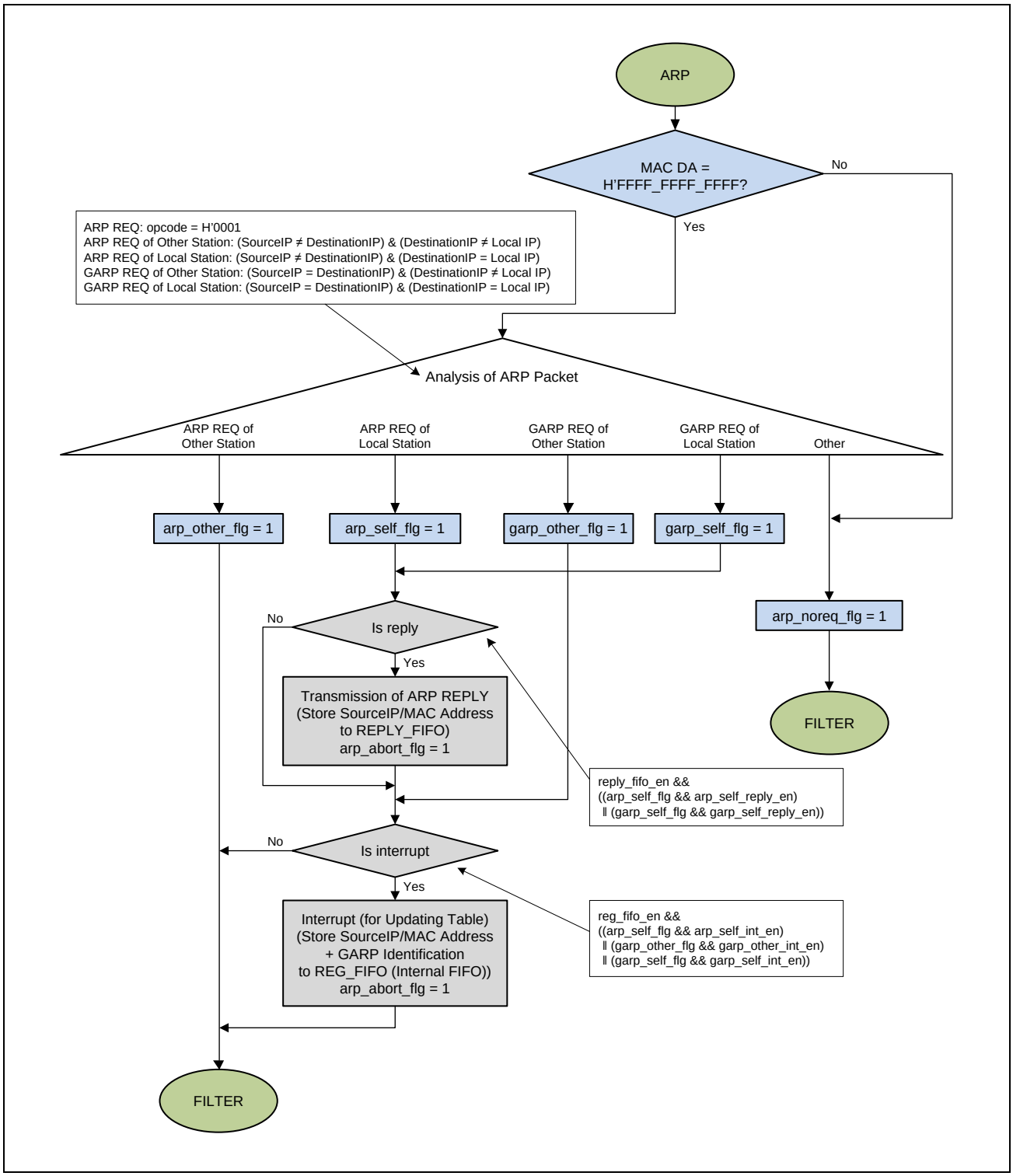


Figure 31.38 Received Frame Filtering Flow (Part5)

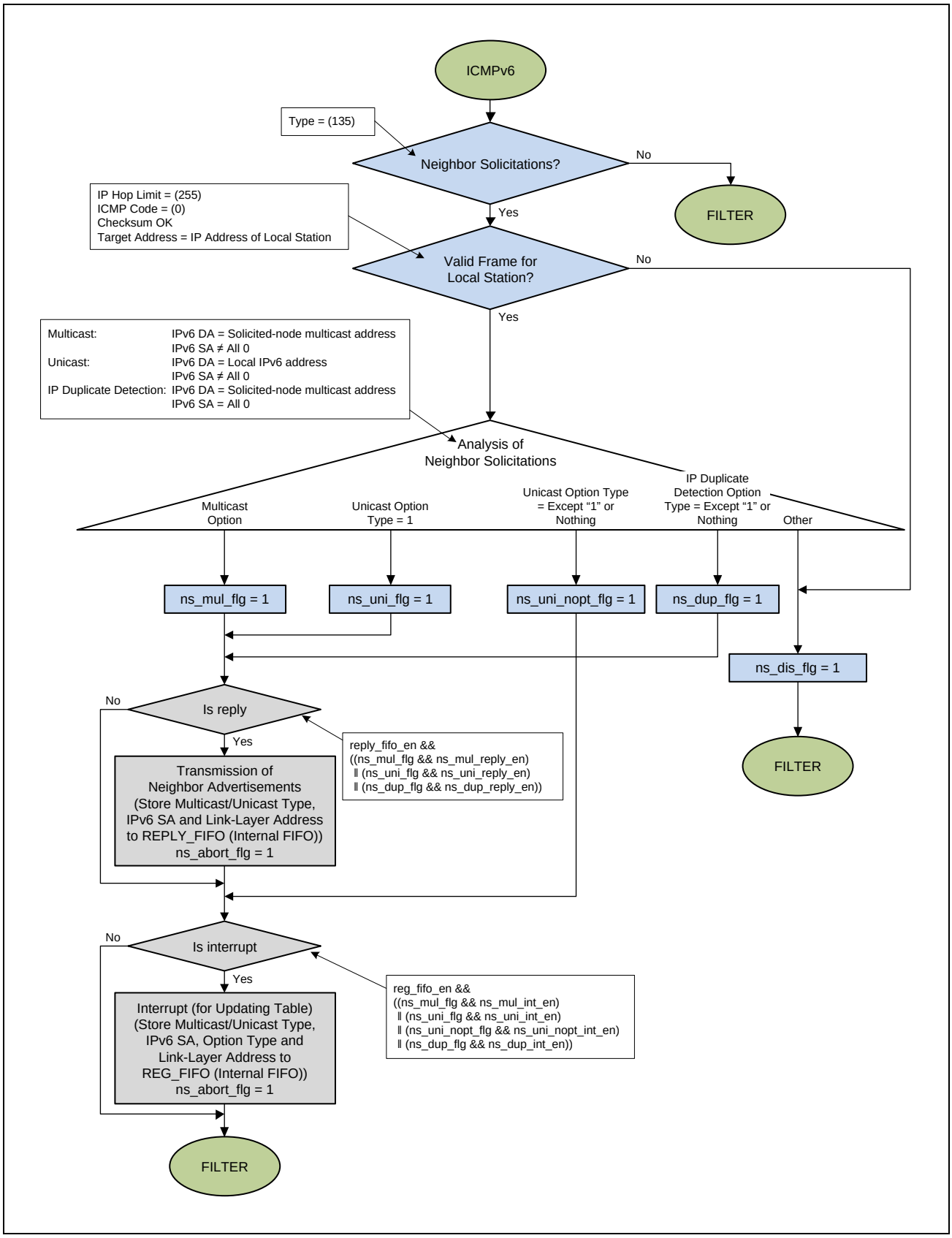


Figure 31.39 Received Frame Filtering Flow (Part6)

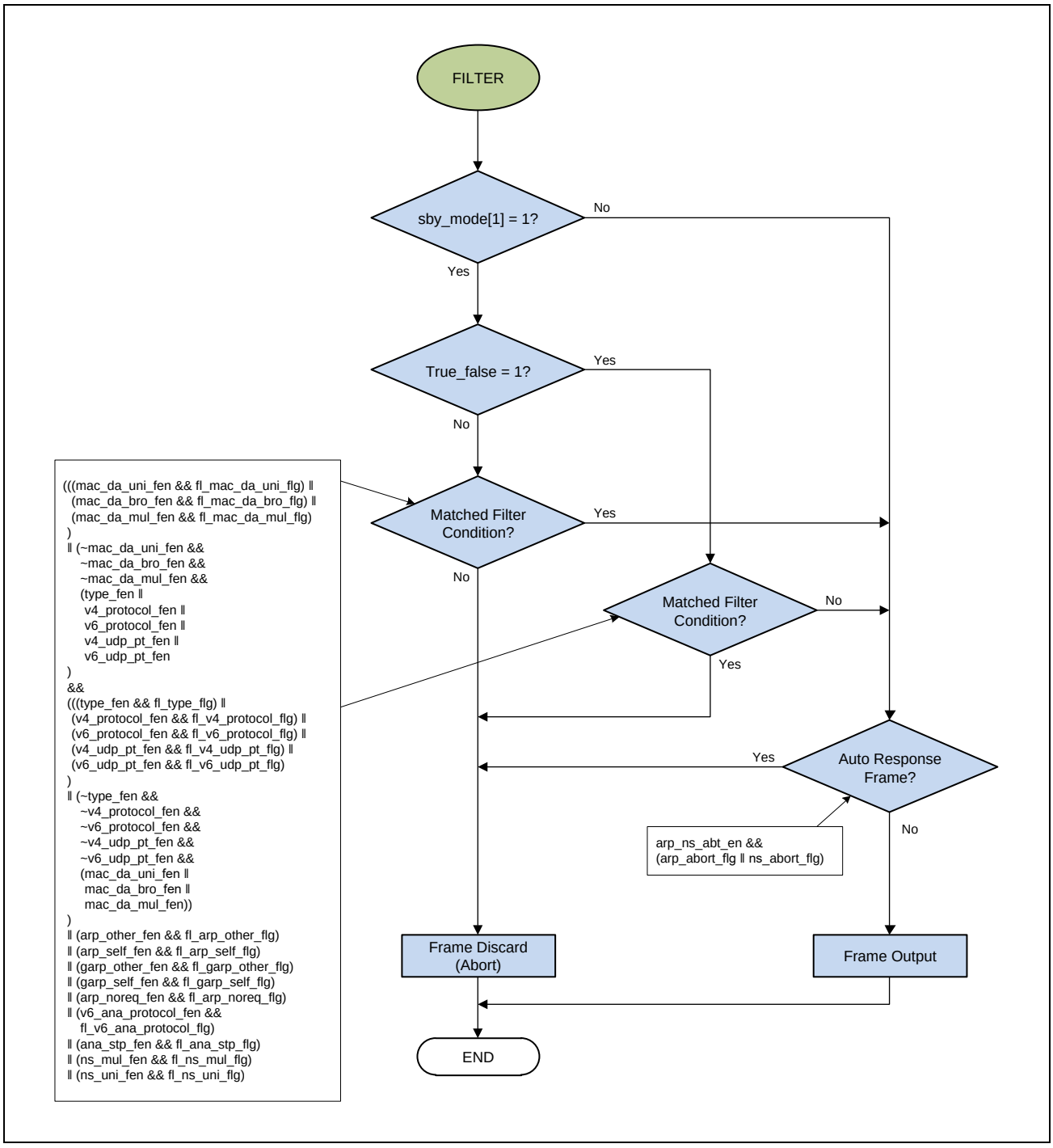


Figure 31.40 Received Frame Filtering Flow (Part7)

31.5.7.3 Filtering and Interrupts

Figure 31.41 is an overview for the function of Filtering and Interrupt.

TOE has flags for each filter conditions. When they match filter configurations of CSFR1*, each flag is set.

CSFR00 controls the action (discard or transmit) for the frames which are configured that Filtering is Enable. The filter condition relationship between “Match/Unmatch” and “Discard/Transmit” is configured by CSFR00.true_false.

The Enable/Disable configuration for discarding frame is controllable in a lump by CSFR00.sby_mode [1:0].

The flag status of each filter’s conditions is indicated to CSFR20 and outputted as interrupt. And the mask control for each interrupt is configured by CSFR21.

TOE has flags of each filter’s conditions for Filtering and Interrupt. The set condition of flags for each function is same. But clear condition is different. (The flags for Filtering are cleared at the top of frame. The flags for Interrupt are cleared by write access to the interrupt status registers.) And TOE also has Interrupt Flag which indicates the Received Frame was transmitted from TOE to DMAC without discarding by filtering conditions.

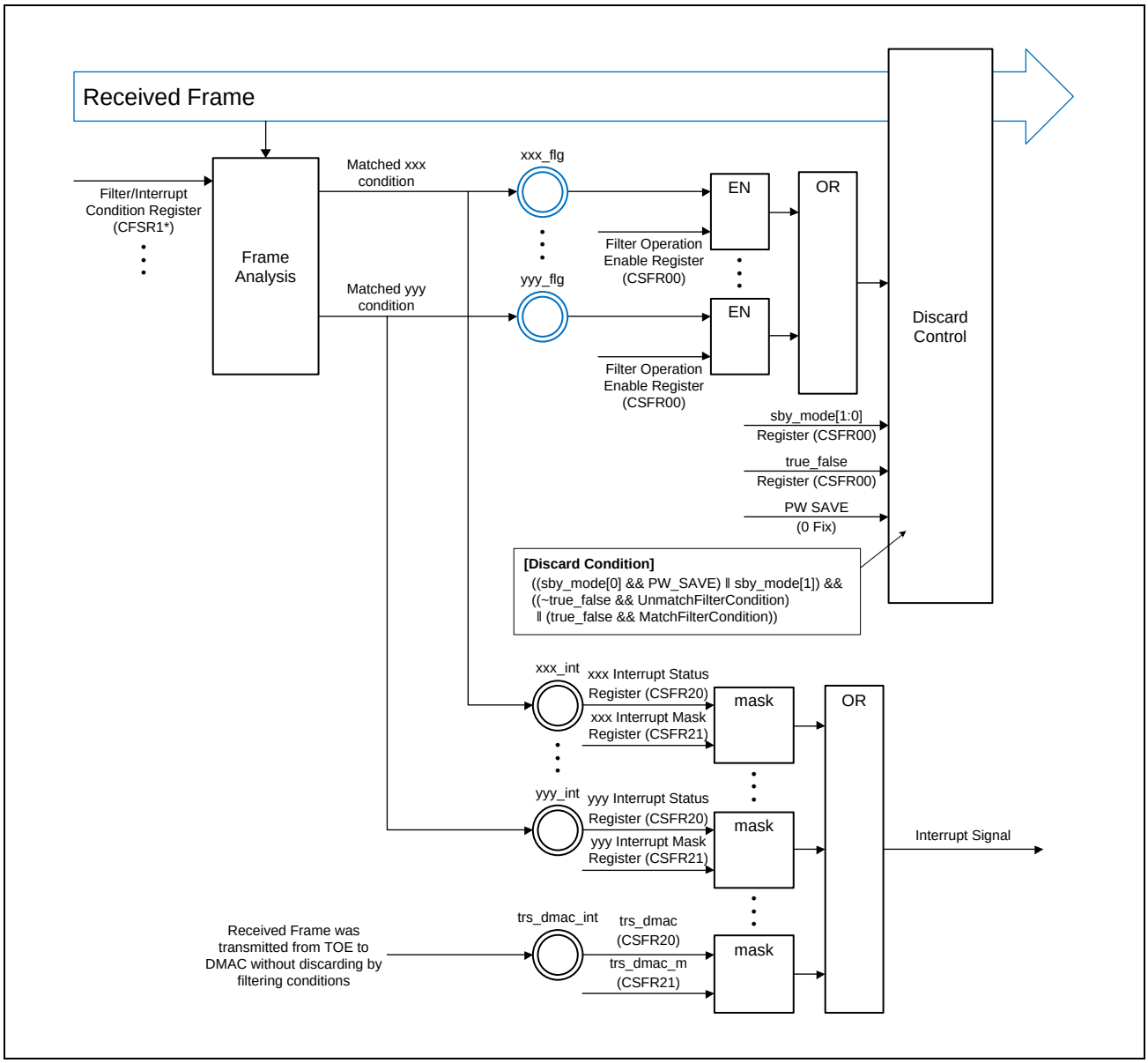


Figure 31.41 Overview of Filtering and Interrupt

31.5.8 Auto Response

TOE has following automatic responding functions.

- When TOE received ARP request for Local Station, it transmits ARP reply automatically.
- When TOE received Neighbor Solicitation for Local Station, it transmits Neighbor Advertisement automatically.

TOE also has following interrupt functions for updating ARP Table/Cache or responding function by software.

- When TOE received ARP request for Local Station, it asserts interrupt and indicates packet's information to registers. (CSFR4*)
- When TOE received Neighbor Solicitation for Local Station, it asserts interrupt and indicates packet's information to registers. (CSFR4*)

31.5.8.1 Auto Response Function for ARP REQ

Table 31.18 is operations for ARP request. See **Figure 31.38** for about the Frame Analysis Flow of ARP REQ.

Table 31.18 Operation after Received ARP REQ

	General Operation		TOE Operation		
ARP REQ	Transmit ARP REPLY	Update Table	REPLY FIFO	REG FIFO	Remarks Column
ARP for Other Station (SourceIP ≠ DestinationIP)	×	×	×	×	ARP REQ for Other Station
ARP for Local Station (SourceIP ≠ DestinationIP)	✓	✓	✓	✓	Operation of ARP for Local Station
GARP for Other Station (SourceIP = DestinationIP)	×	✓	×	✓	Request of updating ARP Table
GARP for Local Station (SourceIP = DestinationIP)	✓	×	✓	×	Response for IP Duplicate Detection

GARP:

REQ which Source IP Address and Destination IP Address are same.

REPLY FIFO:

It stores SourceIP/MAC Address to FIFO and transmits ARP REPLY by Hardware.
The information in this FIFO is cleared after TOE sends ARP REPLY.

REG FIFO:

It stores SourceIP/MAC Address + GARP identification to FIFO and asserts interrupt.
Software is expected to check registers (CSFR4*) and update ARP Table.
The information in this FIFO is cleared by writing "1" to CSRF40.arp_ns. At that time if this FIFO has next information, CSRF40.arp_ns is asserted again and CSFR4* are updated to next information.

Storing to REPLY FIFO and REG FIFO is controlled by followings.

Use followings for Enable/Disable Configuration in a lump.

CSFR00.reply_fifo_en.reg_fifo_en

Use followings for Enable/Disable Configuration of individually

CSFR30.arp_self_reply_en
.garp_self_reply_en
.arp_self_int_en
.garp_other_int_en
.garp_self_int_en

When CSFR00.arp_ns_abt_en = “1”, the ARP frame which was stored to REPLY FIFO or REG FIFO is not transferred to DMAC. It is discarded (aborted).

REPLY FIFO and REG FIFO is used as shared FIFO for the function of “Auto Response for Neighbor Solicitation” which is indicated in the chapter of 4.8.2. And the FIFO capacity is for 8 frames.

If TOE received ARP REQ when FIFO was full, its information is not stored to FIFO and the interrupt is asserted to notify the status of by reg_fifo_full or reply_fifo_full. If one-FIFO is already full and TOE receives ARP REQ which is stored to both FIFO, its information is not stored to both FIFO.

Table 31.19 Operation after Received Neighbor Solicitation

	General Operation		TOE Operation		
Neighbor Solicitations	Transmit Neighbor Advertisements	Update Cache	REPLY FIFO	REG FIFO	Remarks Column
Multicast for Local Station Option Type = 1 (SourceIP ≠ All 0)	✓	✓	✓	✓	—
Unicast for Local Station Option Type = 1 (SourceIP ≠ All 0)	✓	✓	✓	✓	—
Unicast for Local Station Option Type ≠ 1 or Nothing (SourceIP ≠ All 0)	✓	×	×	✓	Transmission of Neighbor Advertisements is executed by software via REG_FIFO.
IP Duplicate Detection for Local Station Option Type ≠ 1 or Nothing (SourceIP = All 0)	✓	×	✓	×	Response for IP Duplicate Detection
Other	×	×	×	×	Invalid

REPLY FIFO:

It stores SourceIP/MAC Address + multicast/unicast identification to FIFO and transmits Neighbor Advertisements by Hardware.
The information in this FIFO is cleared after TOE sends Neighbor Advertisements.

REG FIFO:

It stores SourceIP/MAC Address + multicast/unicast identification + OptionType to FIFO and asserts interrupt to CPU. Software is expected handling followings by interrupt status (CSFR4*).

- Updating cache.

- In the case of Unicast for Local Station(Option Type ≠ 1 or Nothing), transmits Neighbor Advertisements after checking Source MAC Address by referencing cache or Neighbor search.

The information in this FIFO is cleared by writing “1” to CSRF40.arp_ns. At that time, if this FIFO has next information, CSRF40.arp_ns is asserted again and CSFR4* are updated to next information.

TOE does not support anycast address. If anycast address is set to CSFR02_i register, TOE behaves same action the case of received unicast address. But the Random Delay Function for responding Neighbor Advertisement which is be defined in RFC2461 is not supported.

Storing to REPLY FIFO and REG FIFO is controlled by followings.

Use followings for Enable/Disable Configuration in a lump.

CSFR00.reply_fifo_en
.reg_fifo_en

Use followings for Enable/Disable Configuration of individually

CSFR30.ns_mul_reply_en
.ns_uni_reply_en
.ns_dup_reply_en
.ns_mul_int_en
.ns_uni_int_en
.ns_uni_nopt_int_en
.ns_dup_int_en

When CSFR00.arp_ns_abt_en = “1”, the Neighbor Solicitation frame which was stored to REPLY FIFO or REG FIFO is not transferred to DMAC. It is discarded (aborted).

REPLY FIFO and REG FIFO is used as shared FIFO for the function of “Auto Response for ARP REQ” which is indicated in the chapter of 4.8.1. And the FIFO capacity is for 8 frames.

If TOE received Neighbor Solicitation when FIFO was full, its information is not stored to FIFO and the interrupt is asserted to notify the status of by reg_fifo_full or reply_fifo_full. If one-FIFO is already full and TOE receives Neighbor Solicitations which is stored to both FIFO, its information is not stored to both FIFO.

31.5.9 Flow Control

Flow control conforming to the IEEE802.3x is possible during full-duplex operation. There are two methods of transmitting PAUSE frames that are used in flow control as given below.

To use this function, configure CCC.FCE and RCR.RFCL of DMAC.

(1) Auto transmission of PAUSE frame (See bit 16 in CXR20)

A PAUSE frame is automatically transmitted by E-MAC when DMAC notifies that the condition of Reception FIFO reached at Critical Level. The value set in the CXR71 parameter settings register is used as the timer value included in the PAUSE frame. After transmitting a PAUSE frame, if the “Reception FIFO Critical Level Notification” from DMAC is not negated by the time that is indicated in the Timer value, the PAUSE frame is sent again. The number of times of transmitting a PAUSE frame can be set from 1 to 65535 in the CXR81 register, and once the upper limit of the transmission count is reached (value set in CXR81), the PAUSE frame is not transmitted thereafter.

However, if the “Reception FIFO Critical Level Notification” from DMAC is negated, the transmission count counter is reset when the next “Reception FIFO Critical Level Notification” from DMAC is asserted and PAUSE frame transmission begins. It is also possible to not define the transmission count upper limit (transmitting for infinite times).

(2) Manual transmission of PAUSE frame (See CXR72)

The PAUSE frame can be transmitted by software instruction. PAUSE frame transmission is started by setting the Timer value in the CXR72 register. In this case, PAUSE frame is transmitted once only (one frame).

(3) PAUSE Timer value (See bits 19 and 20 in CXR20)

Control can be enabled or disabled for the PAUSE frames having PAUSE Timer value 0

a) Operation during transmission
If control is enabled and the “Reception FIFO Critical Level Notification” from DMAC is negated within the time indicated by the Timer value, the PAUSE frame with Timer value 0 is transmitted.
If control is disabled, the PAUSE frame with Timer value 0 is not transmitted.

b) Operation during reception
If control is enabled, a PAUSE frame with Timer value 0 is received and frame transmission waiting is cancelled.
If control is disabled, a PAUSE frame with Timer value 0 is received and the PAUSE frame is discarded.

(4) PAUSE frame reception (See the bit 17 in CXR20)

When a PAUSE frame is received, the next frame transmission is in wait mode till the time which is considered Timer value of PAUSE frame elapses. The transmission of the frame that is being transmitted continues. In addition, the PAUSE frame reception count is counted (See CXR80).

31.5.10 Low Power Idle (LPI)

E-MAC supports the function of Low Power Idle for 1 Gbps/100 Mbps transfer rate. Since LPI function for 10Mbps is not defined in standard*[1], E-MAC also does not support.

Note 1. Specifically, Standard of 10 BASE-Te whose voltage is lower than 10 BASE-T was defined. But a basic specification except about voltage (such as the rule of Code or Media Access Control Operation) is same as 10 BASE-T. (10 BASE-Te is supported by the cable of Cat5 or more)

(1) Output function of Low Power Idle Code (LPI Code)

When E-MAC does not receive any transmission request during over certain period, E-MAC outputs transmission LPI Code to PHY for notifying request to change into Low Power Mode. If E-MAC receives a transmission request (not includes the request for PAUSE frame) while outputting transmission LPI Code, E-MAC changes output Code from transmission LPI Code to Normal Idle Code. Then E-MAC resumes to transmit Frames after waiting certain period (WakeUp time of PHY).

The following time is configurable by register.
- The time from E-MAC does not receive any transmission request to E-MAC starts outputting transmission LPI Code.
- The time from E-MAC received transmission request to E-MAC starts outputting Normal Idle Code.

(2) Receive function of Low Power Idle Code (LPI Code)

When PHY receives a changing request into Low Power Mode from Link Partner, PHY starts outputting reception LPI Code to E-MAC. If E-MAC receives reception LPI Code, E-MAC does not send this Code to TOE. E-MAC outputs Normal Status to TOE same as receiving Normal Idle Code.

(3) Stop function of transmission clock (Only RGMII Mode)

When RGMII interface is used, E-MAC can stop the RGMII transmission clock (TXC) during outputting transmission LPI Code. This function is configurable by register. But the case of MII interface is used, to stop MII transmission clock (TX_CLK) which PHY outputs is prohibited.

(4) Handling function for stopping reception clock (Both RGMII and MII Mode)

E-MAC can accept to stop RGMII and MII reception clock (RXC/RX_CLK) while receiving reception LPI Code. To stop RGMII and MII reception clock (RXC/RX_CLK) is configurable by register of PHY. (See the datasheet of PHY)

[Simplified Full Duplex Mode]

E-MAC supports Simplified Full Duplex Mode which is used for Low Power Idle Function.

In the Full Duplex Mode, when E-MAC receives transmission request from Upper Module (such as DMAC), E-MAC starts transmitting frames without care of PHY status.

But in the Simplified Full Duplex Mode, E-MAC cares the PHY status and does not transmit any frames to PHY during both PHY status is Low Power Idle and PHY is changing status from Low Power Idle to Normal.

To use E-MAC as Simplified Full Duplex Mode, set ‘1’ to DMP bit and LPM bit of CXR20 register after checking Link Partner supports Low Power Idle Mode by Auto Negotiation*[1].

Note 1. Check PHY and Link Partner support Low Power Idle Mode before setting E-MAC as Low Power Idle Mode. And for the function of Low Power Idle Mode, registers area of PHY was expanded. Therefore, to use Low

Power Idle Mode, software has to be changed for the function of accessing PHY registers, PHY Auto Negotiation and initial setting of E-MAC.

31.5.11 Control of Reference Clock for Ethernet

Gigabit Ethernet Interface IP has a function to notify the status that whether Gigabit Ethernet Interface needs Reference Clock for Ethernet “clk_miitx_gtx_refclk” or not by “ref_clk_stop_en”. And this condition also is notified to CPU by interrupt signal “pif_int_n” (Case of interrupt is indicated on TIS.RCSRF and ISS.RCSRM).

This function is only enabled when the configurations of Gigabit Ethernet Interface are matched following all conditions.

- RGMII is selected as PHY IF.
- This function is enabled by register configuration. (CCC.ERCS = 1)
- E-MAC is configured as LowPowerMode. (CXR20.LPM = 1), Duplex = Full and TransmitSpeed = 1000 M/100 Mbps.

Figure 31.42 is a block diagram of this function.

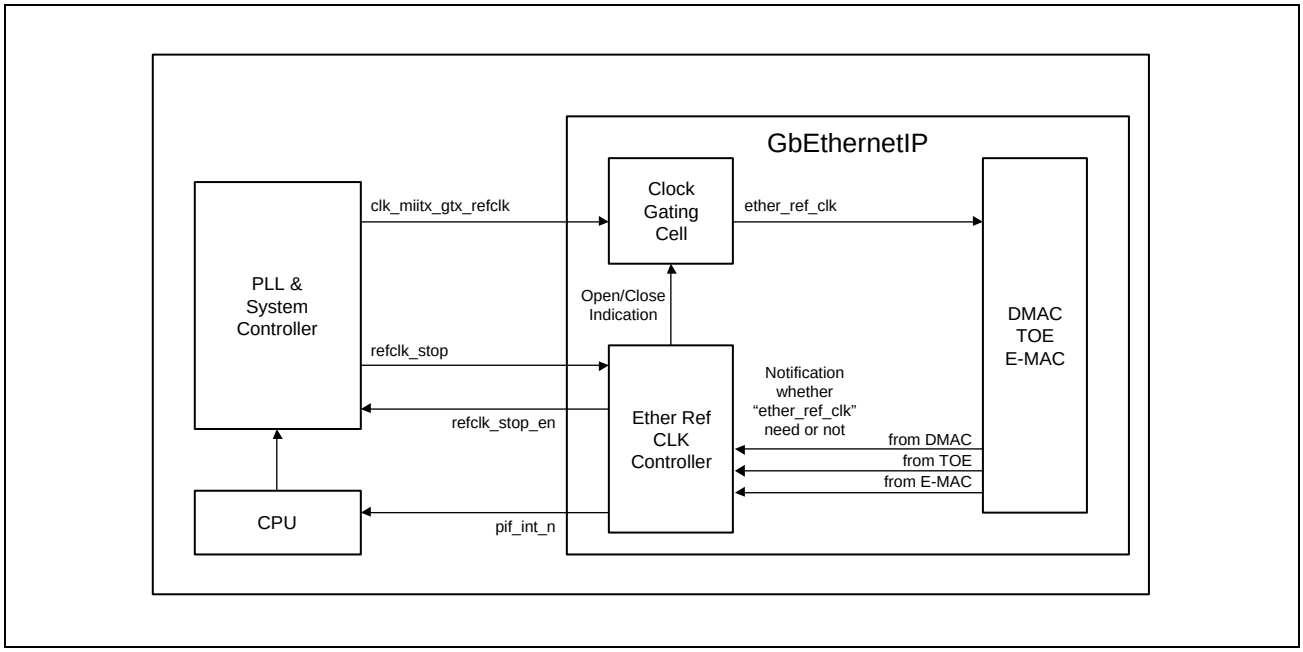


Figure 31.42 Block Diagram

As showing **Figure 31.42**, Gigabit Ethernet Interface has an “Ether Ref CLK Controller” which controls “ref_clk_stop_en”, “pif_int_n” and Open/Close Operation of Internal Clock Gating Cell for “clk_miitx_gtx_refclk”.

The state of Open/Close Operation of Internal Clock Gating Cell is indicated on CSR.RCSI.

NOTE

Whenever “pif_int_n” is asserted by this function, CPU has to instruct PLL & System Controller to stop “clk_miitx_gtx_refclk”.

31.5.11.1 Stop Procedure of Reference Clock for Ethernet

Stop procedure of “clk_miitx_gtx_refclk” is executed by following steps.

1. When DMAC and TOE do not have any Transmission Frames, they do not need “clk_miitx_gtx_refclk” anymore. So, they notify that status to Ether Ref CLK Controller.
2. After outputting Transmission LPI Code during configured time at LPTXG(M)TH4, E-MAC does not need “clk_miitx_gtx_refclk” anymore, So, it notifies that status to Ether Ref CLK Controller.
3. When Ether Ref CLK Controller is received the status that all of DMAC/TOE/E-MAC don’t need “clk_miitx_gtx_refclk” anymore, it asserts “ref_clk_stop_en” to notify about this condition to ETH_PLL & System Controller. Then Gigabit Ethernet Interface closes an internal Clock Gating Cell for “clk_miitx_gtx_refclk” and asserts “pif_int_n” to indicate this condition to CPU.
4. CPU instructs ETH_PLL & System Controller to stop “clk_miitx_gtx_refclk” and writes “0” to TIS.RCSRF to clears “pif_int_n”.
5. After PLL is stopped completely, ETH_PLL & System Controller asserts “ref_clk_stop” notifying PLL status to Ether Ref CLK Controller.

Figure 31.43 is a sequence of stopping “clk_miitx_gtx_refclk”.

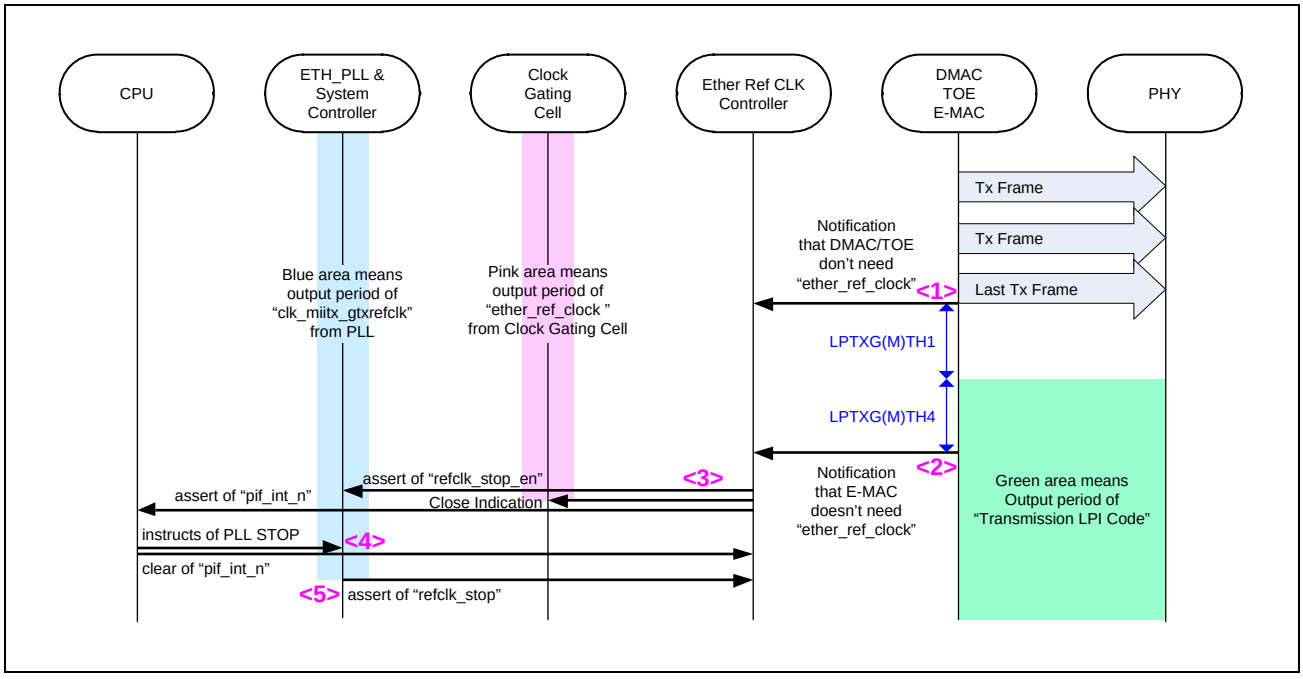


Figure 31.43 Sequence of stopping “clk_miitx_gtx_refclk”

31.5.11.2 Resume Procedure of Reference Clock for Ethernet

Resume procedure of “clk_miitx_gtx_refclk” is executed by following steps.

[Case1] Gigabit Ethernet Interface receives indication from CPU restarting transmission operation.

1. CPU indicates restarting transmitting operation to DMAC.
2. DMAC fetches a new transmission descriptor and gets a Transmission Frame from external Memory. Then, it notifies to Ether Ref CLK Controller that it needs “clk_miitx_gtx_refclk” for transmitting operation.
3. After Ether Ref CLK Controller is notified from DMAC that DMAC needs “clk_miitx_gtx_refclk”, it de-asserts “ref_clk_stop_en” to notify ETH_PLL & System Controller that Gigabit Ethernet Interface needs restart of PLL.
4. ETH_PLL & System Controller tries to restart PLL. After “clk_miitx_gtx_refclk” is resumed completely, it de-asserts “ref_clk_stop” to notify Ether Ref CLK Controller about PLL status.
5. By the change of “ref_clk_stop” from assert to de-assert, Ether Ref CLK Controller opens the internal Clock Gating Cell for ”clk_miitx_gtx_refclk”.
6. Since providing clock is resumed, Transmitting Function of DMAC/TOE/E-MAC is also resumed. Then, E-MAC outputs a Transmission Frame to PHY after outputting IDLE Code during the time which is configured at LPTXG(M)TH3 register.

Figure 31.44 is a sequence of resuming “clk_miitx_gtx_refclk” of Case1.

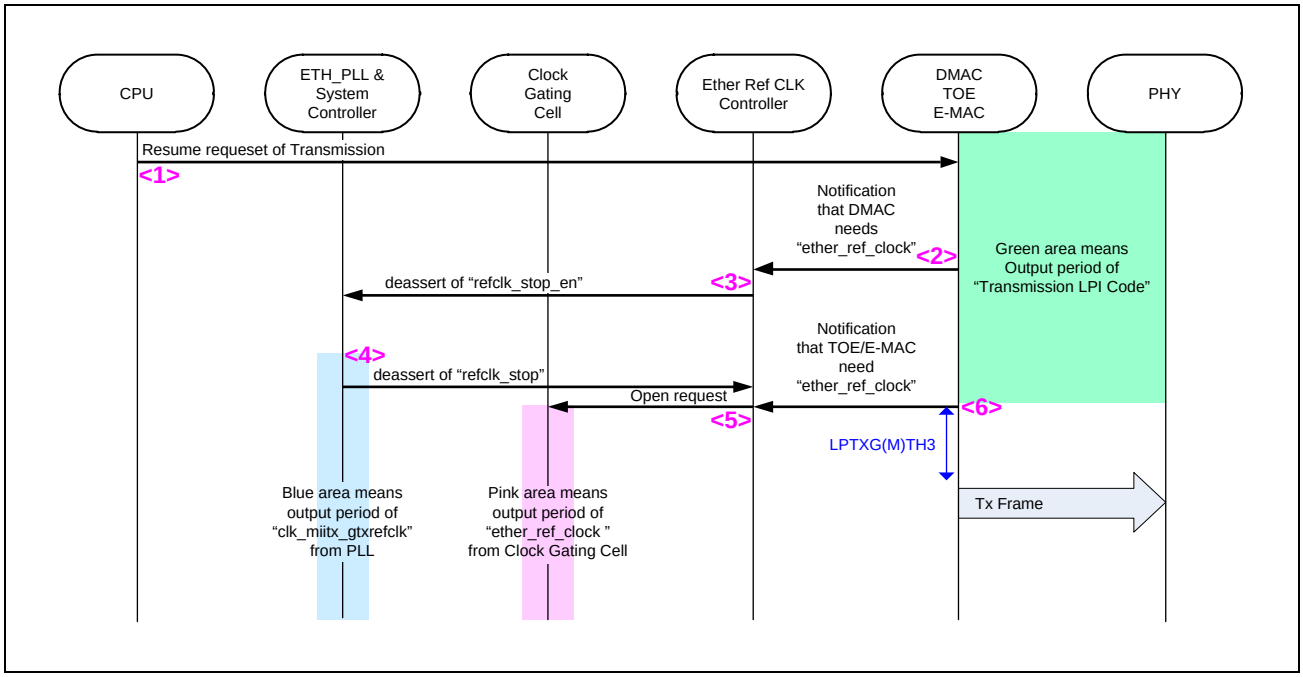


Figure 31.44 Sequence of resuming “clk_miitx_gtx_refclk” of Case1

[Case2] Gigabit Ethernet Interface receives a frame which needs to Automatic Response by H/W.

For about the function of Automatic Response by H/W, see **Section 31.5.8, Auto Response**.

1. Received a frame which needs to Automatic Response by H/W from PHY.
2. After received the frame, TOE notifies to Ether Ref CLK Controller that it needs “clk_miitx_gtx_refclk” for transmitting operation.
3. After Ether Ref CLK Controller is notified from TOE that TOE needs “clk_miitx_gtx_refclk”, it deasserts “ref_clk_stop_en” to notify ETH_PLL & System Controller that Gigabit Ethernet Interface needs restart of PLL.
4. ETH_PLL & System Controller tries to restart PLL. After “clk_miitx_gtx_refclk” is resumed completely, it de-asserts “ref_clk_stop” to notify Ether Ref CLK Controller about PLL status.
5. By the change of “ref_clk_stop” from assert to de-assert, Ether Ref CLK Controller opens the internal Clock Gating Cell for ”clk_miitx_gtx_refclk”.
6. Since providing clock is resumed, Transmitting Function of DMAC/TOE/E-MAC is also resumed. Then, E-MAC outputs an Automatic Response Frame to PHY after outputting IDLE Code during the time which is configured at LPTXG(M)TH3 register.

Figure 31.45 is a sequence of resuming “clk_miitx_gtx_refclk” of Case2.

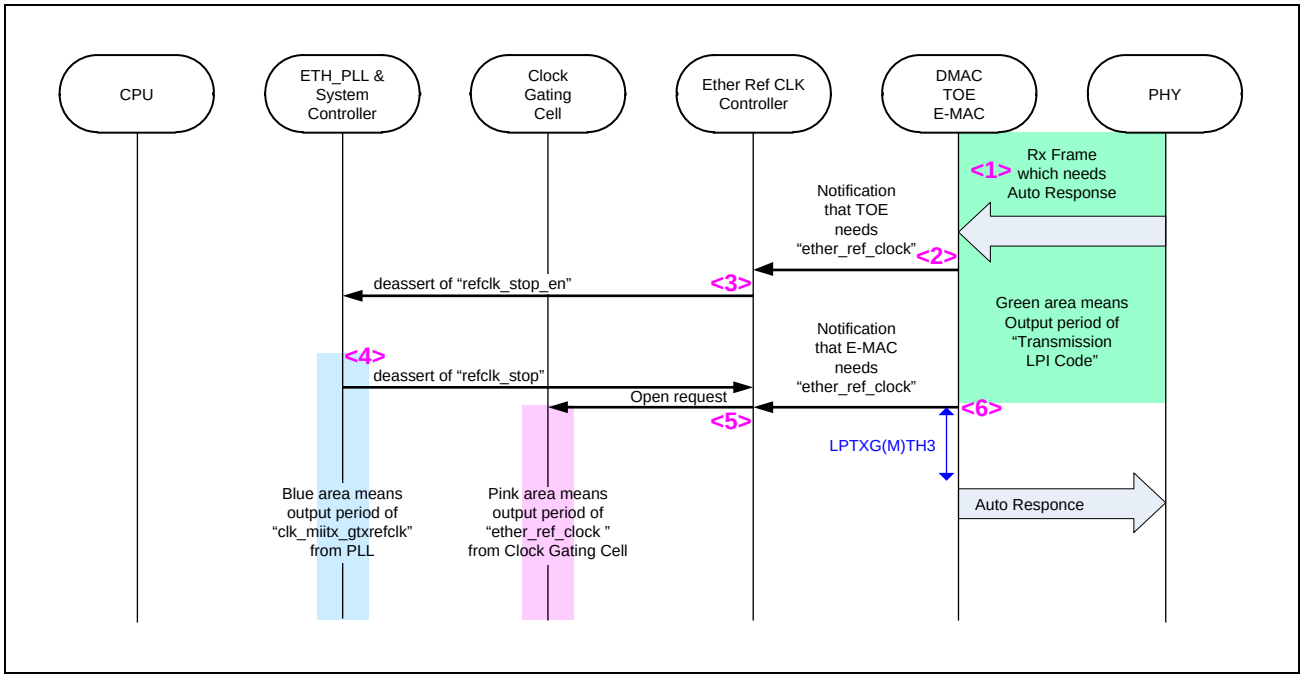


Figure 31.45 Sequence of resuming “clk_miitx_gtx_refclk” of Case2

31.5.12 Interrupts

The DMAC related interrupts include descriptor interrupts, error interrupts, reception interrupts and transmission interrupts. The states of a DMAC-related interrupt sources can be checked in the following registers.

- Descriptor interrupt status register (DIS)
- Error interrupt status register (EIS)
- Receive interrupt status register (RISi)
- Transmit interrupt status register (TIS)

The interrupts are controlled by the corresponding interrupt enable bits. However, the status flags operate independently of the settings of the enable bits.

The states of grouped interrupts can only be checked by reading the interrupt summary status register (ISS) and the queue full error interrupt summary bit in the error interrupt status register (EIS.QFS). This reduces the load on the CPU.

31.5.12.1 Transmit/Receive Data Management Interrupt

The management interrupt for transmission and reception is conveyed when the interrupt conditions corresponding to the following sources are satisfied.

- Receive frame interrupts in the receive interrupt status register 0 (RIS0.FRF)
- Descriptor processed interrupts in the receive interrupt status register 3 (RIS3.RDPF)
- Frame transmitted interrupts in the transmit interrupt status register (TIS.FTF)
- Descriptor processed interrupts in the transmit interrupt status register (TIS.TDPF)
- Descriptor processed interrupts in the descriptor interrupt status register (DIS.DPFi)

The general error interrupt state can be checked by reading the descriptor interrupt flag bits in the interrupt summary status register (ISS.DPMi) or the Reception FIFO warning interrupt summary bit (ISS.RFWM).

31.5.12.2 Error Management Interrupt

The error management interrupt is conveyed when interrupt conditions corresponding to the following sources are satisfied.

- Queue error interrupt in the error interrupt status register (EIS.QEF)
- E-MAC and TOE transmission/reception error interrupts in the error interrupt status register (EIS.MTEF, EIS.MREF, EIS.MFFF, EIS.MFFF2)
- Reception FIFO full interrupt in the receive interrupt status register 2 (RIS2.RFFF)
- Receive queue full interrupts in the receive interrupt status register 2 (RIS2.QFF0)

The general error interrupt state can be checked by reading the error interrupt summary bit in the interrupt summary status register (ISS.EM, ISS.MFWM2, ISS.MFUM2).

31.5.12.3 Other Management Interrupts

The other management interrupt is conveyed when interrupt conditions corresponding to the following sources are satisfied.

(1) Reception related interrupt

- Reception FIFO warning interrupt in the receive interrupt status register 1 (RIS1.RFWF)

(2) Transmission related interrupts

- MAC Status FIFO/FIFO2 Warning Flag (TIS.MFWF/TIS.MFWF2)
- MAC status FIFO/FIFO2 Updated Flag (TIS.MFUF/TIS.MFUF2)
- Reference Clock Stop Req Flag (TIS.RCSRF)

The general error interrupt state can be checked by reading the Reception FIFO warning error interrupt status bit in the interrupt summary status register (ISS.RFWM).

31.5.12.4 E-MAC Interrupt

The E-MAC interrupt is conveyed when the E-MAC interrupt source is generated.

- PAUSE Frame Retry Interrupt in the E-MAC status register (CXR21.PFRI)
- PHY Interrupt in the E-MAC status register (CXR21.PHYI)
- LINK Interrupt in the E-MAC status register (CXR21.LINKI)
- False Carrier Interrupt in the E-MAC status register (CXR21.FCI)

The general error interrupt state can be checked by reading the E-MAC interrupt summary bit in the interrupt summary status register (ISS.MM).

31.5.12.5 TOE Interrupt

The TOE interrupt is issued when the TOE interrupt source is generated.

- Wake Up interrupt in the TOE register (CSFR20)
- ARPREQ/Neighbor Solicitation Receive interrupt in the TOE register (CSFR40, CSFR42_*, CSFR43_*)

31.5.12.6 Interrupt

Table 31.20, **Table 31.21**, and **Table 31.22** shows interrupt for each interrupt status bit. Interrupts of DMAC and E-MAC are merged into the “pif_int_n” of output port. TOE has two independent output ports “int_fil_n” and “int_arp_ns_n”.

Table 31.20 Interrupt of DMAC

Interrupt trigger	Interrupt flag bit	Interrupt enable bit	Interrupt (Output port)
The descriptor in a reception or transmit queue has been processed where DESCR.DIE is i	DIS.DPFi (i = 1 to 15)	DIC.DPEi = 1	pif_int_n
In receive queue a frame is stored, and data is available to be processed by CPU.	RIS0.FRF	RIC0.FRE = 1	
The descriptor in reception queue has been processed where DESCR.DIE is 0001b.	RIS3.RDPF	RIC3.RDPE = 1 & CIE.CRIE = 1	
The descriptor in transmit queue has been processed where DESCR.DIE is 0001b.	TIS.TDPF	TIC.TDPE = 1 & CIE.CTIE = 1	
A frame is transmitted from transmit queue.	TIS.FTF	TIC.FTE = 1	
An error has been detected while processing receive or transmit queue.	EIS.QEF	EIC.QEE = 1	
The E-MAC has detected a fault during transmission.	EIS.MTEF	EIC.MTEE = 1	
The E-MAC has detected a fault during reception.	EIS.MREF	EIC.MREE = 1	
The MAC status FIFO2 is full.	EIS.MFFF2	EIC.MFFE2 = 1	
The MAC status FIFO is full.	EIS.MFFF	EIC.MFFE = 1	
A received frame was not completely stored in the Reception FIFO.	RIS2.RFFF	RIC2.RFFE = 1	
No sufficient space in receive queue to store completely a received frame.	RIS2.QFF0	RIC2.QFE = 1	
The Reception FIFO has reached warning level.	RIS1.RFWF	RIC1.RFWE = 1	
The MAC status FIFO has been reached warning level.	TIS.MFWF	TIC.MFWE = 1	
The MAC status FIFO has been updated.	TIS.MFUF	TIC.MFUE = 1	
The MAC status FIFO2 has been reached warning level.	TIS.MFWF2	TIC.MFWE2 = 1	
The MAC status FIFO2 has been updated.	TIS.MFUF2	TIC.MFUE2 = 1	
Reference Clock Stop Request has been notified	TIS.RCSRF	TIC.RCSRE = 1	

Table 31.21 Interrupt of E-MAC

Interrupt trigger	Interrupt flag bit	Interrupt enable bit	Interrupt (Output port)
PAUSE Frame Retry reached to threshold.	CXR21.PFRI	CXR22.PFRIM	pif_int_n
PHY Interrupt is detected.	CXR21.PHYI	CXR22.PHYIM	
LINK status change is detected.	CXR21.LINKI	CXR22.LINKIM	
False Carrier is detected.	CXR21.FCI	CXR22.FCIM	

Table 31.22 Interrupt of TOE

Interrupt trigger	Interrupt flag bit	Interrupt mask bit	Interrupt main enable bit	Interrupt (Output port)
Transmit Rx Frame to DMAC	CSFR20.trs_dmac	CSFR21. trs_dmac_m	—	int_fil_n
Reception of IEEE Length Frames.	CSFR20.ieee_len	CSFR21. ieee_len_m	—	
Reception of Invalid Neighbor Solicitations.	CSFR20.ns_dis	CSFR21.ns_dis_m	—	
Reception of IP Duplicate Detection Neighbor Solicitations.	CSFR20.ns_dup	CSFR21.ns_dup_m	—	
Reception of unicast Neighbor Solicitations. (Except “Option Type=1” or No “Option Type”)	CSFR20.ns_uni_nopt	CSFR21.ns_uni_nopt_m	—	
Reception of unicast Neighbor Solicitations. (“Option Type=1”)	CSFR20.ns_uni	CSFR21.ns_uni_m	—	
Reception of multicast Neighbor Solicitations.	CSFR20.ns_mul	CSFR21.ns_mul_m	—	
Reception of Un-analyzable Frames.	CSFR20.ana_stp	CSFR21.ana_stp_m	—	
Reception of Un-analyzable IPv6 Protocol No.	CSFR20.v6_ana_protocol	CSFR21.v6_ana_protocol_m	—	
Reception of ARP (Except ARP REQ).	CSFR20.arp_noreq	CSFR21.arp_noreq_m	—	
Reception of GARP REQ for Local Station.	CSFR20.garp_self	CSFR21.garp_self_m	—	
Reception of GARP REQ for Other Station.	CSFR20.garp_other	CSFR21.garp_other_m	—	
Reception of ARP REQ for Local Station.	CSFR20.arp_self	CSFR21.arp_self_m	—	
Reception of ARP REQ for Other Station.	CSFR20.arp_other	CSFR21.arp_other_m	—	
Match of IPv6 UDP Port No.	CSFR20.v6_udp_pt	CSFR21.v6_udp_pt_m	—	
Match of IPv4 UDP Port No.	CSFR20.v4_udp_pt	CSFR21.v4_udp_pt_m	—	
Match of IPv6 Protocol No.	CSFR20.v6_protocol	CSFR21.v6_protocol_m	—	
Match of IPv4 Protocol No.	CSFR20.v4_protocol	CSFR21.v4_protocol_m	—	
Match of Ether Type.	CSFR20.type	CSFR21.type_m	—	
Match of Multicast MAC DA.	CSFR20.mac_da_mul	CSFR21.mac_da_mul_m	—	
Match of Broadcast MAC DA.	CSFR20.mac_da_bro	CSFR21.mac_da_bro_m	—	
Match of Unicast MAC DA.	CSFR20.mac_da_uni	CSFR21.mac_da_uni_m	—	
REPLY_FIFO is full.	CSFR40.reply_fifo_full	CSFR41.reply_fifo_full_m	CSFR00.reply_fifo enable & at least one CSFR30.xx_reply_en	int_arp_ns_n
REG_FIFO is full.	CSFR40.reg_fifo_full	CSFR41.reg_fifo_full_m	CSFR00.reg_fifo_en & at least one CSFR30.xx_int_en	
Reception of ARPREQ or Neighbor Solicitations and Stored to REG FIFO.	CSFR40.arp_ns	CSFR41.arp_ns_m	CSFR00.reg_fifo_en & at least one CSFR30.xx_int_en	

31.5.13 Configuration Procedure

The following are Configuration Procedures. The meaning of “X” in Write Value, set the appropriate value to each bit for your system.

31.5.13.1 Set Up Procedure

Check the Configuration of Other Station by using Auto Negotiation Function of PHY before setting Transfer Rate (1 Gbps/100 Mbps/10 Mbps), Duplex Mode (Half or Full), Low Power Mode and Flow Control to E-MAC registers.

Action		Address	Write Value	Description
Hardware Reset				
↓				
DMAC.CSR	Read	H’00C	—	Check the read value is same as “H’0000_0001” (Reset Mode).
↓				
DMAC.CCC	Write	H’000	H’0000_0001	Set “Config Mode”
↓				
DMAC.CSR	Read	H’00C	—	Check the read value is same as “H’0000_0002” (Config Mode).
↓				
E-MAC Initial Configuration (Not necessary to change other E-MAC registers)				
E-MAC.CXR35	Write	H’540	H’XXXX_XXXX	Set the appropriate value for your system. And wait the double of your configuration value to this register.
E-MAC.CXR31	Write	H’530	H’0000_000X	Set the appropriate value for your system. (Only if In-Band Status of RGMII will be used)
E-MAC.CXR2C	Write	H’52C	H’0000_000X	Set the appropriate value for your system.
E-MAC.CXR2A	Write	H’508	H’0000_2000	Set Max Frame size as 8 KB.
E-MAC.CXR71	Write	H’554	H’0000_XXXX	Set the appropriate value for your system. (Only if Flow Control Function will be used)
E-MAC.CXR24	Write	H’5C0	H’XXXX_XXXX	Set the appropriate value for your system.
E-MAC.CXR25	Write	H’5C8	H’0000_XXXX	Set the appropriate value for your system.
E-MAC.CXR21	Write	H’510	H’0000_001F	Clear all Interrupt.
E-MAC.CXR22	Write	H’518	H’0000_000X	Set the appropriate value for your system.
E-MAC.CXR2D	Write	H’5B0	H’0000_00X0	Set the appropriate value for your system. (Only if In-Band Status of RGMII will not be used)
E-MAC.CXR2G	Write	H’5BC	H’0000_000X	Set the appropriate value for your system. (Only if Software LINK will not be used)
↓				
TOE Initial Configuration (Not necessary to change other TOE registers).				
TOE.CSR1	Write	H’804	H’0XX0_00XX	Set the appropriate value for your system.
TOE.CSR2	Write	H’808	H’0XX0_00XX	Set the appropriate value for your system.
TOE.CSFR01	Write	H’844	H’XXXX_XXXX	Set the appropriate value for your system.
TOE.CSFR02_i	Write	H’848+4*i	H’XXXX_XXXX	Set the appropriate value for your system. (i = 0 to 3)
TOE.CSFR03_U	Write	H’858	H’0000_XXXX	Set the appropriate value for your system.
TOE.CSFR03_L	Write	H’85C	H’XXXX_XXXX	Set the appropriate value for your system.
TOE.CSFR10_i	Write	H’870+4*i	H’XXXX_XXXX	Set the appropriate value for your system. (i = 0 to 3)
TOE.CSFR10	Write	H’880	H’0000_00XX	Set the appropriate value for your system.
TOE.CSFR11_i	Write	H’884+4*i	H’XXXX_XXXX	Set the appropriate value for your system. (i = 0 to 3)
TOE.CSFR11	Write	H’894	H’0000_XXXX	Set the appropriate value for your system.
TOE.CSFR12_i	Write	H’898+4*i	H’XXXX_XXXX	Set the appropriate value for your system. (i = 0 to 11)
TOE.CSFR12	Write	H’8C8	H’00XX_XXXX	Set the appropriate value for your system.
TOE.CSFR13_U	Write	H’8CC	H’0000_XXXX	Set the appropriate value for your system.

Action		Address	Write Value	Description
TOE.CSFR13_L	Write	H'8D0	H'XXXX_XXXX	Set the appropriate value for your system.
TOE.CSFR14_U	Write	H'8D4	H'0000_XXXX	Set the appropriate value for your system.
TOE.CSFR14_L	Write	H'8D8	H'XXXX_XXXX	Set the appropriate value for your system.
TOE.CSFR15_U_i	Write	H'8DC+8*i	H'0000_XXXX	Set the appropriate value for your system. (i = 0 to 19)
TOE.CSFR15_L_i	Write	H'8E0+8*i	H'XXXX_XXXX	Set the appropriate value for your system. (i = 0 to 19)
TOE.CSFR15	Write	H'97C	H'XX0X_XXXX	Set the appropriate value for your system.
TOE.CSFR20	Write	H'9A0	H'000F_FFFF	Clear all Interrupt.
TOE.CSFR21	Write	H'9A4	H'000X_XXXX	Set the appropriate value for your system.
TOE.CSFR30	Write	H'9B0	H'000X_XXXX	Set the appropriate value for your system.
TOE.CSFR31	Write	H'9B4	H'XX0X_XXXX	Set the appropriate value for your system.
TOE.CSFR00	Write	H'840	H'10XX_XXXX	Enable the appropriate Filter Function for your system. Disable of Auto Response Function.
↓				
DMAC Initial Configuration (Not necessary to change other DMAC registers).				
DMAC.DBAT	Write	H'004	H'XXXX_XXXX	Set the appropriate value for your system.
DMAC.RCR	Write	H'090	H'XXXX_000X	Set the appropriate value for your system.
DMAC.RTC	Write	H'0B4	H'0000_2000	Set Max Reception Frame size as 8 KB.
DMAC.CIE	Write	H'384	H'0000_0X0X	Set the appropriate value for your system.
DMAC.DIE	Write	H'450	H'0000_XXXX	Set the appropriate value for your system.
DMAC.EIE	Write	H'458	H'0000_0X0X	Set the appropriate value for your system.
DMAC.RIE0	Write	H'460	H'0000_000X	Set the appropriate value for your system.
DMAC.RIE1	Write	H'468	H'X000_0000	Set the appropriate value for your system.
DMAC.RIE2	Write	H'470	H'X000_000X	Set the appropriate value for your system.
DMAC.TIE	Write	H'478	H'000X_XX0X	Set the appropriate value for your system.
DMAC.RIE3	Write	H'488	H'0000_000X	Set the appropriate value for your system.
↓				
DMAC.CCC	Write	H'000	H'0X0X_0002	Set bit [1:0] = "10b" (Operation Mode) and bit25 (FCE)/bit18 (ERCS) = "the appropriate value for your system".
↓				
TOE.CSR0	Write	H'800	H'0000_0030	Set TPE and RPE = "1" and TOE FIFO=2 KB.
↓				
E-MAC.CXR20	Write	H'500	H'0XXX_XXXX	Set the appropriate value for your system. (The following are example)
			H'0203_0061	Rx CRC PassThrough, PAUSE, TPE and RPE = '1', Half Duplex (for 100 Mbps/10 Mbps), Promiscuous.
			H'0203_0063	Rx CRC PassThrough, PAUSE, TPE/RPE = '1', Full Duplex (for 10 Mbps), Promiscuous.
			H'0203_0073	Rx CRC PassThrough, PAUSE, TPE/RPE = '1', Simplified Full Duplex (for 1 Gbps/100 Mbps), Promiscuous.
↓				
DMAC.TCCR	Write	H'304	H'000X_0001	Set bit0 (TSRQ) = "1" and bit18 (MFEN2), bit16 (MFEN) = "the appropriate value to for your system".

31.5.13.2 System Mode Change Procedure (Normal -> Network Standby)

Action		Address	Write Value	Description
DMAC.TCCR	Read	H'304	—	Wait until bit0 (TSRQ) will be "0".
↓				
DMAC.CSR	Read	H'00C	—	Wait until bit16 (TPO) and bit20 (RPO) will be "0".
↓				
DMAC.CCC	Write	H'000	H'0X0X_0202	Set bit9 (RDFD) = "1" to stop fetch of Rx Descriptor. Not necessary to change other bits.
↓				
DMAC.CSR	Read	H'00C		Wait until bit9 (RDFDM) will be "1" and re-check bit20 (RPO) is "0".*1
↓				
TOE Configuration to enable Auto Response Function (Not necessary to change other registers)				
TOE.CSFR40	Write	H'9C0	H'0000_0007	Clear all Interrupt.
TOE.CSFR41	Write	H'9C4	H'0000_000X	Set the appropriate value for your system.
TOE.CSFR00	Write	H'840	H'1XXX_XXXX	Enable of the appropriate Auto Response Function for your system. Not necessary to change Filter Function.
↓				
Change System Mode from "Normal" to "Network Standby".				
↓				
Set Tx Descriptor and Tx Frame at Internal SRAM				
↓				
DMAC.DBAT	Write	H'004	H'XXXX_XXXX	Set the appropriate value for your system.
↓				
DMAC.DLR	Write	H'008	H'003F_FFFF	Set bit0 and bit4 = "1" loading DBAT to Tx and Rx queue.
↓				
DMAC.TCCR	Write	H'304	H'000X_0001	Set bit0 (TSRQ) = "1" and bit18 (MFEN2), bit16 (MFEN) = "the appropriate value to for your system".

Note 1. If bit9 (RDFDM) is "0" and bit20 (RPO) is "0", wait bit9 (RDFDM) will be "1" without accessing other registers.
If bit9 (RDFDM) is "1" and bit20 (RPO) is "0", the DMA transfer and fetch of Rx Descriptor have stopped.
So go to next step.
If bit9 (RDFDM) is "0" and bit20 (RPO) is "1", wait bit9 (RDFDM) will be "1" without accessing other registers.
If bit9 (RDFDM) is "1" and bit20 (RPO) is "1", the Reception Frame was stored to Reception FIFO.
So software has to cancel changing System mode. Set DMAC.CCC. bit9 (RDFD) = "0" immediately to resume fetch of Rx Descriptor.

31.5.13.3 System Mode Change Procedure (Network Standby -> Normal)

Action		Address	Write Value	Description
TOE.CSFR00	Write	H'840	H'10XX_XXXX	Disable of the Auto Response Function. Not necessary to change Filter Function.
↓				
DMAC.TCCR	Read	H'304	—	Wait until bit0 (TSRQ) will be "0".
↓				
DMAC.CSR	Read	H'00C	—	Wait until bit16 (TPO) will be "0".
↓				
Change System Mode from "Network Standby" to "Normal".				
↓				
Set Rx/Tx Descriptor and Tx Frame at External SDRAM.				
↓				
DMAC.DBAT	Write	H'004	H'XXXX_XXXX	Set the appropriate value for your system.
↓				
DMAC.DLR	Write	H'008	H'003F_FFFF	Load DBAT to Tx and Rx queue.
↓				
DMAC.CCC	Write	H'000	H'0X0X_0002	Set "bit9 (RDFD) = 0" to start Rx DMA Transfer for Rx Frame. Not necessary to change other bits.
↓				
DMAC.TCCR	Write	H'304	H'000X_0001	Set bit0 (TSRQ) = "1" and bit18 (MFEN2), bit16 (MFEN) = "the appropriate value to for your system".

31.5.13.4 Stop Procedure

Action		Address	Write Value	Description
Make “EOS” in the Tx Descriptor chain and wait until it is used. (Wait the interrupt of DIS.DPFi of this EOS Descriptor.)				
↓				
E-MAC.CXR20	Write	H’500	H’0XXX_XXXX	Set bit6 (RPE) = “0” to stop Rx Function of E-MAC. Not necessary to change other bits.
↓				
Wait following time depending on the Transfer speed mode. (Waiting time that the last Rx Frame is stored to Reception FIFO and the last Tx Frame is transmitted to PHY) • 1 Gbps : Wait 70 µs • 100 Mbps : Wait 700 µs • 10 Mbps : Wait 7 ms				
↓				
DMAC.CSR	Read	H’00C	—	Wait until bit20 (RPO) will be “0” (Wait Reception FIFO will be empty)
↓				
DMAC.CCC	Write	H’000	H’0X0X_0102	Set bit8 (DTSR) = “1” to stop URAM access. Not necessary to change other bits.
↓				
DMAC.CSR	Read	H’00C	—	Wait until bit8 (DTS) will be “1” (URAM access is stopping)
↓				
TOE.CSR0	Write	H’800	H’0000_0000	Set bit5 (RPE) = “0” and bit4 (TPE) = “0” to stop Rx and Tx Function of TOE. Not necessary to change other bits.
↓				
TOE.CSR0	Read	H’800	—	Wait until bit5 (RPE) and bit4 (TPE) will be “0”
↓				
E-MAC.CXR20	Write	H’500	H’0XXX_XXXX	Set bit5 (TPE) = “0” to stop Tx Function of E-MAC. Not necessary to change other bits.
↓				
E-MAC.CXR20	Read	H’500	—	Wait until bit6 (RPE) and bit5 (TPE) will be “0”
↓				
DMAC.CCC	Write	H’000	H’0X0X_0001	Set bit [1:0] = “01b” (Config Mode) and bit8 (DTSR) = “0”

Note: If the system needs software reset, set DMAC.CCC bit [1:0] = “00b” to be RESET Mode after “Stop Procedure” and retry “Set up Procedure”.

31.5.14 Connection to PHY-LSI

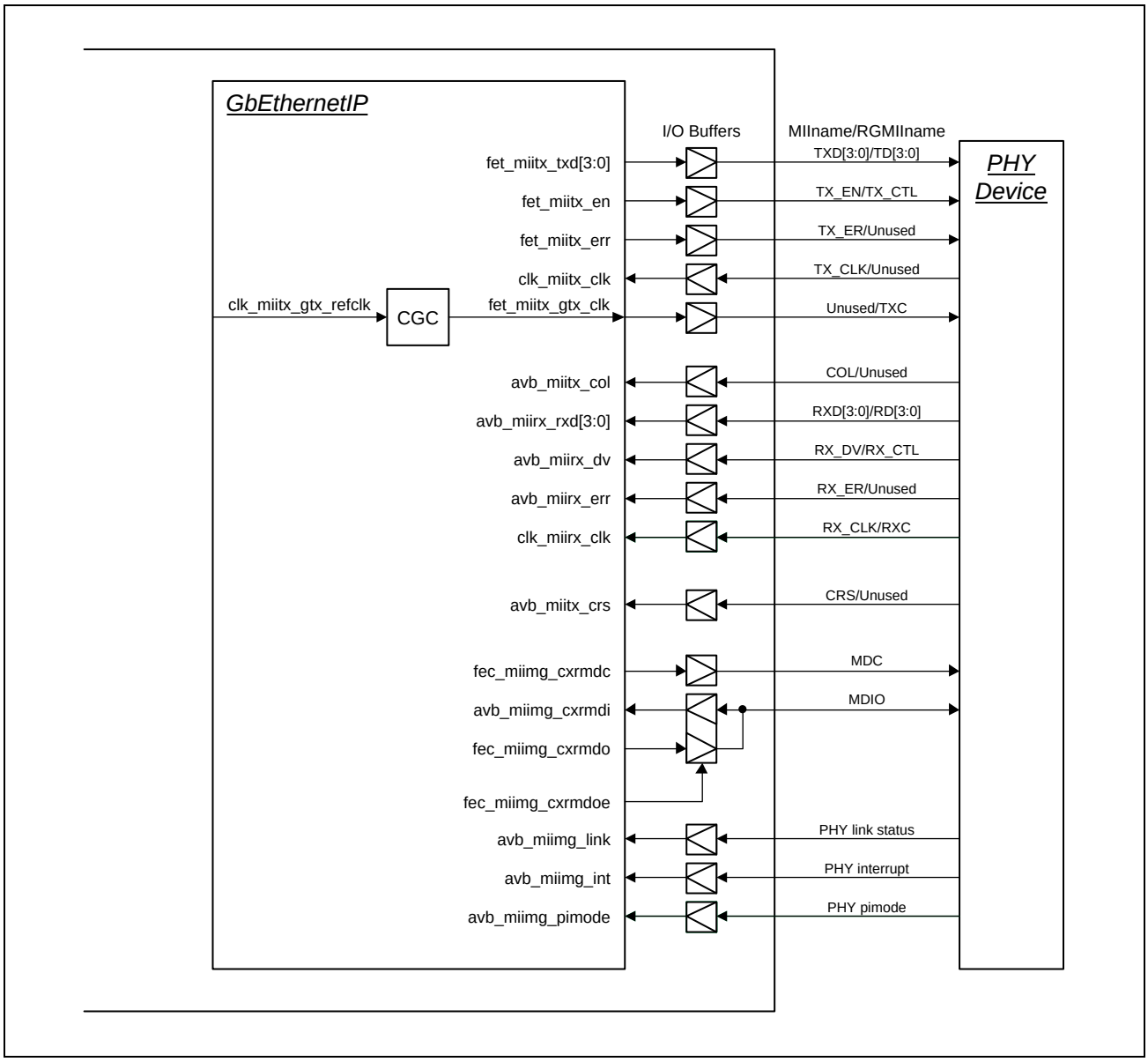


Figure 31.46 Connection to PHY-LSI

31.5.14.1 RGMII Frame Transmission/Reception Timing

Each RGMII frame transmission/reception timing is shown in **Figure 31.47** and **Figure 31.48**.

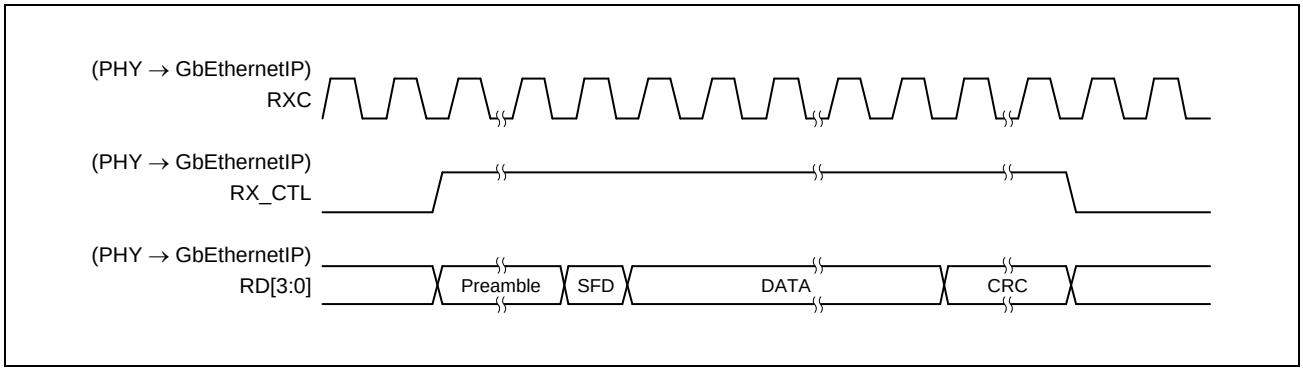


Figure 31.47 Normal Frame Reception Time Chart (RGMII)

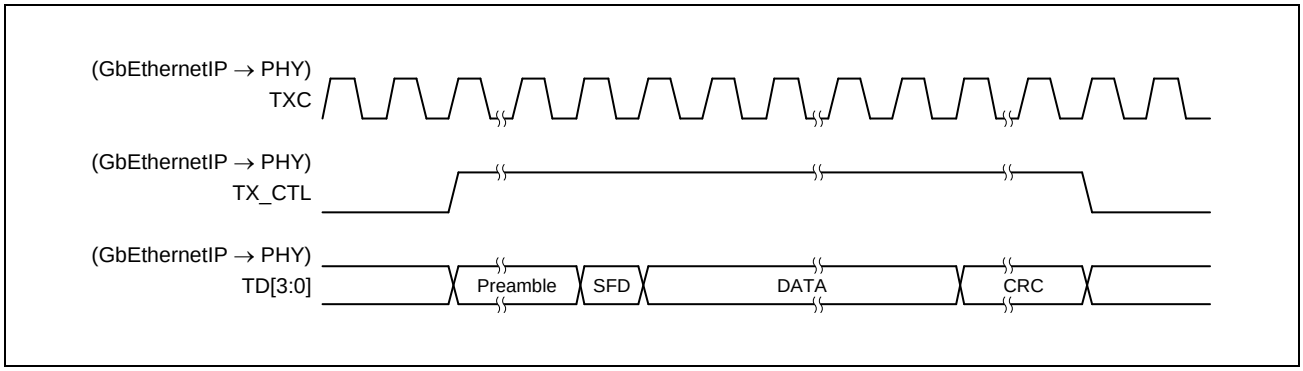


Figure 31.48 Normal Frame Transmission Time Chart (RGMII)

31.5.14.2 MII Frame Reception Timing

Each MII frame transmission/reception timing is shown in **Figure 31.49** and **Figure 31.50**.

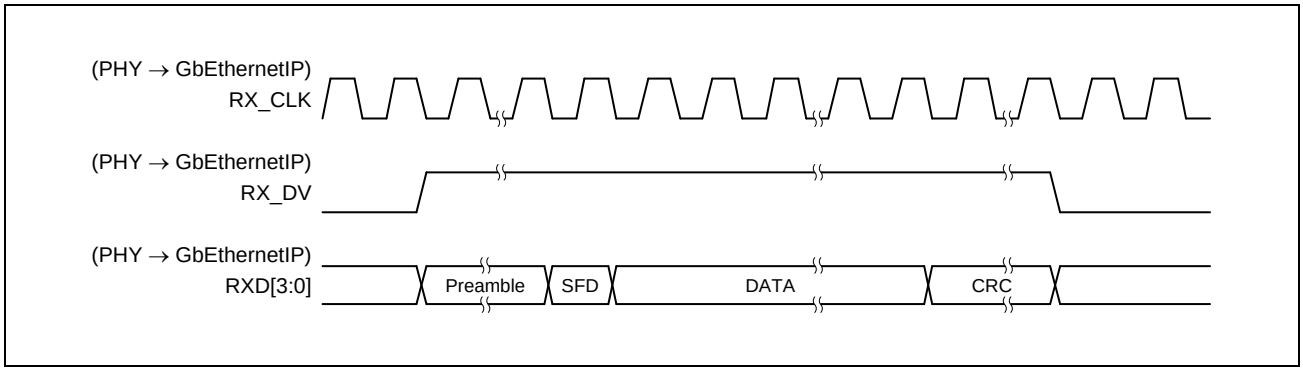


Figure 31.49 Normal Frame Reception Time Chart (MII)

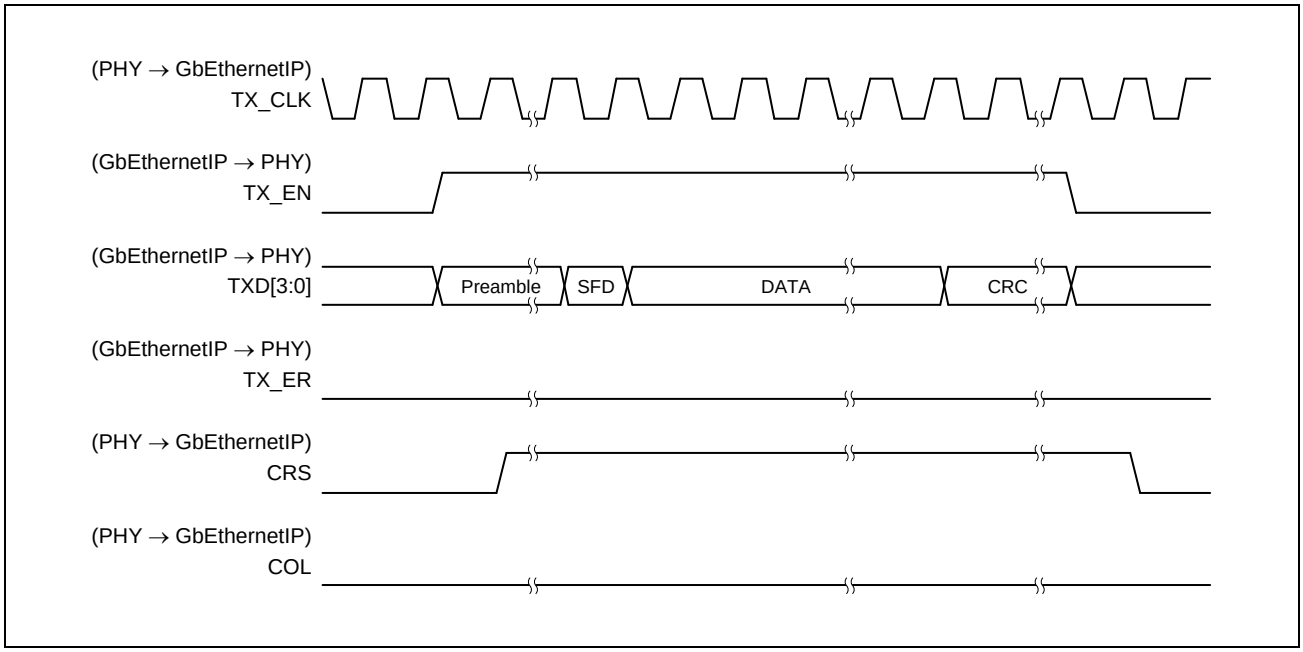


Figure 31.50 Normal Frame Transmission Time Chart (MII)

31.5.14.3 Accessing PHY Registers

There are two methods to access the PHY register: using the E-MAC-MDIOC block and using the CXR23 register.

Which access method to be used can be set using the MDIOMOD register.

(1) PHY Register Access Method (using MDIOC)

To access the PHY register from MDIOC, the SEL_MDIO bit in MDIOMOD register must be set to 0.

The following commands can be used for indirect access using MDIOC.

Table 31.23 List of MDIOC Commands

Command	OP code	Address	Data	Description
WRITE	“01”	A [9:0]	D [15:0]	Write PHY register
READ	“10”	A [9:0]	D [15:0]	Read PHY register

PHY Register Write Access Procedure (Using MDIOC)

The write access procedure for PHY register is given below.

(1) Read MDIOSTS register and confirm that BSY bit = 0.

(2) Set PHY address as (A [9:5]) and register address as (A [4:0]) in MDIOADR register.

(3) Set write data in MDIODAT register.

(4) Set OP field of MDIOCMD register to “01”.
{Here, the hardware sets BSY bit in MDIOSTS register to 1.}

(5) Read MDIOSTS register and wait till BSY bit = 0. After BSY becomes 0, write access is complete.
{After write access to PHY register completes, the BSY bit in MDIOSTS register is cleared to 0 by hardware.}

PHY Register Read Access Procedure (Using MDIOC)

The read access procedure for PHY Register is given below.

(1) Read MDIOSTS register and confirm that BSY bit = 0.

(2) Set PHY address as (A [9:5]) and register address as (A [4:0]) in MDIOADR register.

(3) Set OP field of MDIOCMD register to “10”.
{Here, the hardware sets BSY bit in MDIOSTS register to 1.}

(4) Read MDIOSTS register and wait till BSY bit = 0.
{After read access from PHY register completes and read data becomes available in MDIODAT register, hardware clears BSY bit in MDIOSTS register is cleared to 0 by hardware.}

(5) When the BSY bit in MDIOSTS register is cleared to 0, read data is acquired from MDIODAT register and access is complete.

(2) PHY Register Access Method (Using CXR23)

To access PHY register using CXR23 register, SEL_MDIO bit of MDIOMOD register must be set to 1. (The initial value can be used as is.)

PHY access using CXR23 is done by controlling High and Low of MDC and MDIO signals for every cycle. The relationship between CXR23 register access and MDC and MDIO signals is shown below.

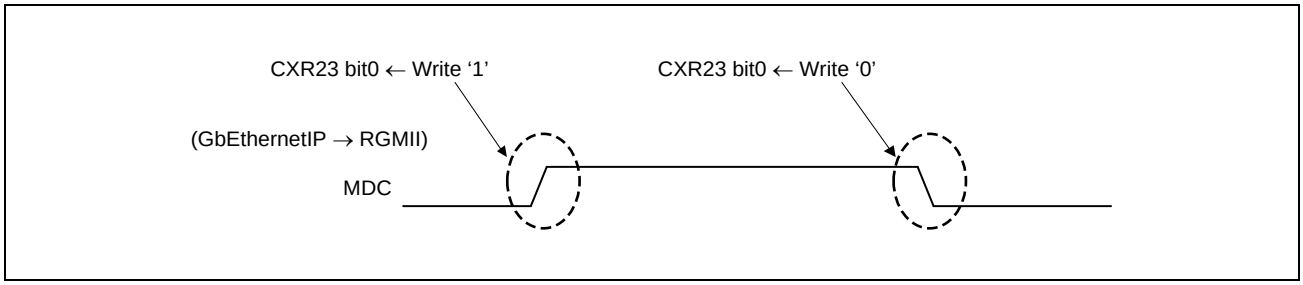


Figure 31.51 Relationship between CXR23 Write and MDC signal

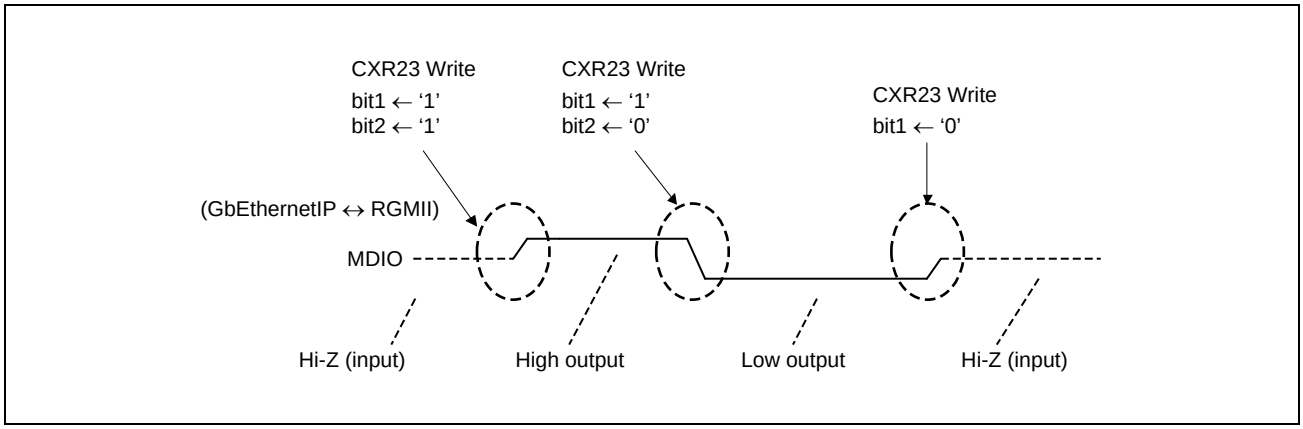


Figure 31.52 Relationship between CXR23 Write and MDIO (output) signal

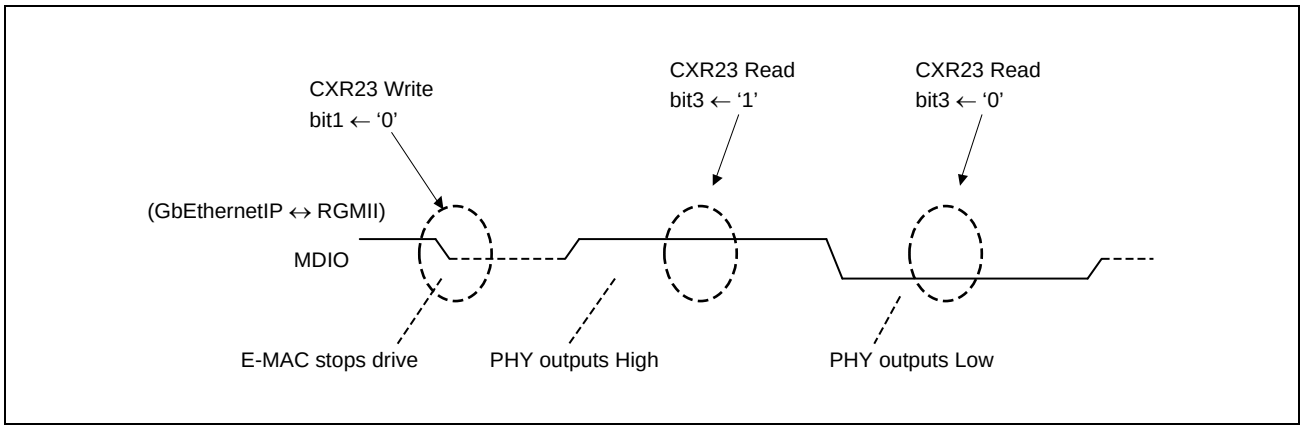


Figure 31.53 Relationship between MDIO input and CXR23 Read Values

PHY Register Write Access Procedure (Using CXR23)

To write PHY chip register, control MDC and MDIO signals based on the format given below.

(1) PHY Write Access Format

Table 31.24 PHY Write Access Format

#	Field name	I/O	Details	Remarks
1	Preamble	O	1 = 32 bits	Preamble
2	ST	O	“01”	Start
3	Op Code (Write)	O	“01”	Operation code for write
4	PHY Address	O	5 bits	PHY address (See data sheet of PHY LSI)
5	Reg Address	O	5 bits	Register address (See data sheet of PHY LSI)
6	Turn Around	O	“10”	Turnaround
7	Data (Write)	O	16 bits	Write data
8	IDLE	—	—	Idle status

(2) PHY Register Write Access Time chart

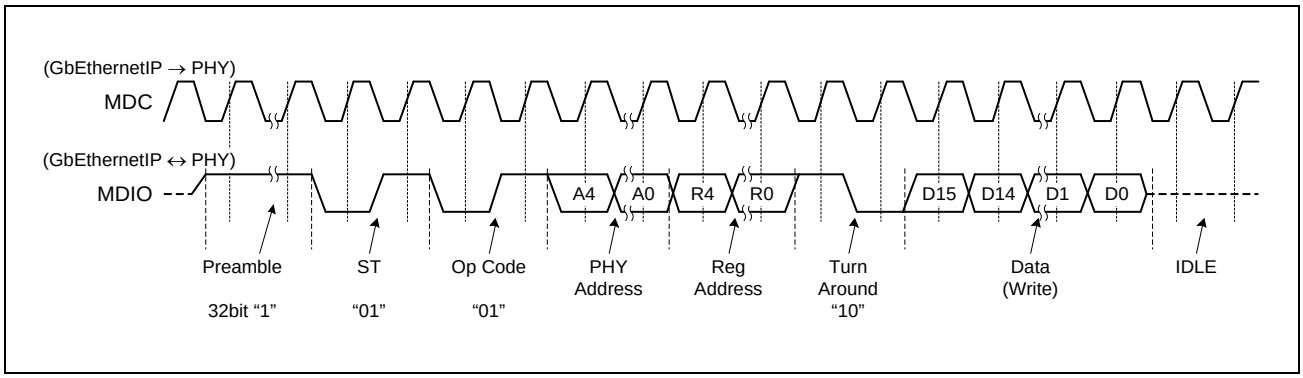


Figure 31.54 PHY Register Write Access Time Chart

(3) PHY Register Write Access Procedure

Table 31.25 PHY Register Write Access Procedure (1/2)

#	CXR23				CXR23 access	Description
	bit 3	bit 2	bit 1	bit 0		
0	0	0	0	0	Initial state	IDLE
1	0	1	1	1	CXR23 WRITE	Start of Write access (MDC rise)
2	0	1	1	0	CXR23 WRITE	Start of Write access (MDC fall)
3	0	1	1	1	CXR23 WRITE	Preamble1
4	0	1	1	0	CXR23 WRITE	
5	0	1	1	1	CXR23 WRITE	Preamble2
6	0	1	1	0	CXR23 WRITE	
7	0	1	1	1	CXR23 WRITE	Preamble3
8	0	1	1	0	CXR23 WRITE	
—	—	—	—	—	—	Repeat
64	0	1	1	0	CXR23 WRITE	
65	0	1	1	1	CXR23 WRITE	Preamble32
66	0	0	1	0	CXR23 WRITE	
67	0	0	1	1	CXR23 WRITE	ST
68	0	1	1	0	CXR23 WRITE	
69	0	1	1	1	CXR23 WRITE	ST
70	0	0	1	0	CXR23 WRITE	
71	0	0	1	1	CXR23 WRITE	Op Code (Write)
72	0	1	1	0	CXR23 WRITE	
73	0	1	1	1	CXR23 WRITE	Op Code (Write)
74	0	A4	1	0	CXR23 WRITE	PHY address A4 bit is output here
75	0	A4	1	1	CXR23 WRITE	PHY Address
76	0	A3	1	0	CXR23 WRITE	
77	0	A3	1	1	CXR23 WRITE	PHY Address
78	0	A2	1	0	CXR23 WRITE	
79	0	A2	1	1	CXR23 WRITE	PHY Address

Table 31.25 PHY Register Write Access Procedure (2/2)

#	CXR23				CXR23 access	Description
	bit 3	bit 2	bit 1	bit 0		
80	0	A1	1	0	CXR23 WRITE	
81	0	A1	1	1	CXR23 WRITE	PHY Address
82	0	A0	1	0	CXR23 WRITE	
83	0	A0	1	1	CXR23 WRITE	PHY Address
84	0	R4	1	0	CXR23 WRITE	REG address A4 bit is output here
85	0	R4	1	1	CXR23 WRITE	REG Address
86	0	R3	1	0	CXR23 WRITE	
87	0	R3	1	1	CXR23 WRITE	REG Address
88	0	R2	1	0	CXR23 WRITE	
89	0	R2	1	1	CXR23 WRITE	REG Address
90	0	R1	1	0	CXR23 WRITE	
91	0	R1	1	1	CXR23 WRITE	REG Address
92	0	R0	1	0	CXR23 WRITE	
93	0	R0	1	1	CXR23 WRITE	REG Address
94	0	1	1	0	CXR23 WRITE	Turn Around
95	0	1	1	1	CXR23 WRITE	Turn Around
96	0	0	1	0	CXR23 WRITE	Turn Around
97	0	0	1	1	CXR23 WRITE	Turn Around
98	0	D15	1	0	CXR23 WRITE	Write data D15 bit is output here
99	0	D15	1	1	CXR23 WRITE	Write Data
100	0	D14	1	0	CXR23 WRITE	
101	0	D14	1	1	CXR23 WRITE	Write Data
—	—	—	—	—	—	Repeat
112	0	D1	1	0	CXR23 WRITE	
113	0	D1	1	1	CXR23 WRITE	Write Data
114	0	D0	1	0	CXR23 WRITE	
115	0	D0	1	1	CXR23 WRITE	Write Data
116	0	0	0	0	CXR23 WRITE	IDLE
117	0	0	0	1	CXR23 WRITE	IDLE
118	0	0	0	0	CXR23 WRITE	IDLE
119	0	0	0	1	CXR23 WRITE	IDLE
120	0	0	0	0	CXR23 WRITE	IDLE
121	0	0	0	1	CXR23 WRITE	IDLE
122	0	0	0	0	CXR23 WRITE	IDLE End of write access

PHY Register Read Access Procedure (Using CXR23)

To read from the PHY chip registers, control MDC and MDIO signals in the format given below

(1) PHY Read Access Format

Table 31.26 PHY Read Access Format

#	Field name	I/O	Details	Remarks
1	Preamble	O	1 = 32 bits	Preamble
2	ST	O	“01”	Start
3	Op Code (read)	O	“10”	Operation code for read
4	PHY Address	O	5 bits	PHY address (See data sheet of PHY LSI)
5	Reg Address	O	5 bits	Register address (See data sheet of PHY LSI)
6	Turn Around	I	“X0”	Turnaround (input/output switching)
7	Data (Read)	I	16 bits	Read data
8	IDLE	—	—	Idle state

(2) PHY Register Read Access Time Chart

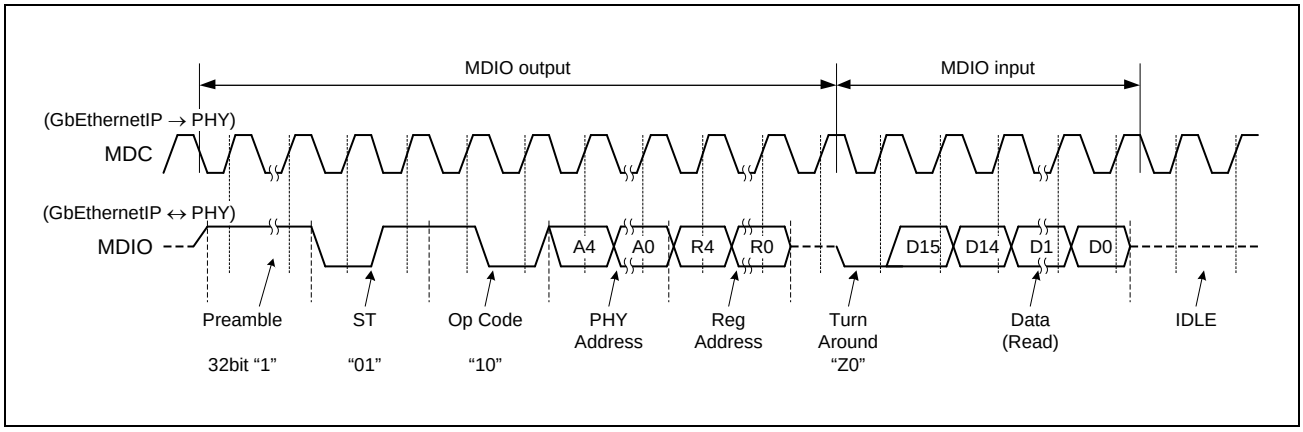


Figure 31.55 PHY Register Read Access Time Chart

(3) PHY Register Read Access Procedure

Table 31.27 PHY Register Read Access Procedure (1/2)

#	CXR23				CXR23 access	Description
	bit 3	bit 2	bit 1	bit 0		
0	0	0	0	0	Initial state	IDLE
1	0	1	1	1	CXR23 WRITE	Start of write access (MDC rise)
2	0	1	1	0	CXR23 WRITE	Start of write access (MDC fall)
3	0	1	1	1	CXR23 WRITE	Preamble1
4	0	1	1	0	CXR23 WRITE	
5	0	1	1	1	CXR23 WRITE	Preamble2
6	0	1	1	0	CXR23 WRITE	
7	0	1	1	1	CXR23 WRITE	Preamble3
8	0	1	1	0	CXR23 WRITE	
—	—	—	—	—	—	Repeat
64	0	1	1	0	CXR23 WRITE	
65	0	1	1	1	CXR23 WRITE	Preamble32
66	0	0	1	0	CXR23 WRITE	
67	0	0	1	1	CXR23 WRITE	ST
68	0	1	1	0	CXR23 WRITE	
69	0	1	1	1	CXR23 WRITE	ST
70	0	1	1	0	CXR23 WRITE	
71	0	1	1	1	CXR23 WRITE	Op Code (Read)
72	0	0	1	0	CXR23 WRITE	
73	0	0	1	1	CXR23 WRITE	Op Code (Read)
74	0	A4	1	0	CXR23 WRITE	PHY address A4 bit is output here
75	0	A4	1	1	CXR23 WRITE	PHY Address
76	0	A3	1	0	CXR23 WRITE	
77	0	A3	1	1	CXR23 WRITE	PHY Address
78	0	A2	1	0	CXR23 WRITE	
79	0	A2	1	1	CXR23 WRITE	PHY Address
80	0	A1	1	0	CXR23 WRITE	
81	0	A1	1	1	CXR23 WRITE	PHY Address
82	0	A0	1	0	CXR23 WRITE	
83	0	A0	1	1	CXR23 WRITE	PHY Address
84	0	R4	1	0	CXR23 WRITE	REG address A4 bit is output here
85	0	R4	1	1	CXR23 WRITE	REG Address
86	0	R3	1	0	CXR23 WRITE	
87	0	R3	1	1	CXR23 WRITE	REG Address
88	0	R2	1	0	CXR23 WRITE	
89	0	R2	1	1	CXR23 WRITE	REG Address
90	0	R1	1	0	CXR23 WRITE	
91	0	R1	1	1	CXR23 WRITE	REG Address
92	0	R0	1	0	CXR23 WRITE	
93	0	R0	1	1	CXR23 WRITE	REG Address
94	0	0	0	0	CXR23 WRITE	Turn Around (MDIO Hi-Z Output MDIO Output -> Input)

Table 31.27 PHY Register Read Access Procedure (2/2)

#	CXR23				CXR23 access	Description
	bit 3	bit 2	bit 1	bit 0		
95	0	0	0	1	CXR23 WRITE	Turn Around
96	0	0	0	0	CXR23 WRITE	Turn Around
97	0	0	0	1	CXR23 WRITE	Turn Around
98	0	0	0	0	CXR23 WRITE	Read Data Start
99	D15	X	X	X	CXR23 Read	Here Data bit15 in PHY register can be seen in bit3 in CXR23
100	0	0	0	1	CXR23 WRITE	
101	0	0	0	0	CXR23 WRITE	
102	D14	X	X	X	CXR23 Read	Data bit14 read
103	0	0	0	1	CXR23 WRITE	
104	0	0	0	0	CXR23 WRITE	
—	—	—	—	—	—	Repeat
141	D1	X	X	X	CXR23 Read	Data bit1 read
142	0	0	0	1	CXR23 WRITE	
143	0	0	0	0	CXR23 WRITE	
144	D1	X	X	X	CXR23 Read	Data bit0 read
145	0	0	0	1	CXR23 WRITE	Read Data End
146	0	0	0	0	CXR23 WRITE	IDLE
147	0	0	0	1	CXR23 WRITE	IDLE
148	0	0	0	0	CXR23 WRITE	IDLE
149	0	0	0	1	CXR23 WRITE	IDLE
150	0	0	0	0	CXR23 WRITE	IDLE
151	0	0	0	1	CXR23 WRITE	IDLE
152	0	0	0	0	CXR23 WRITE	IDLE End of read access

32. USB2.0

This LSI includes 1 channel USB2.0 OTG/DRD (Host/Function) interface and 1 channel USB2.0 Host interface.

This section describes the overview of USB2.0 and USB2.0 PHY control.

The detail function of USB2.0 Host controller and the common function both Host module and Function module are described in **Section 32A, USB 2.0 Host Module**.

The detail function of USB2.0 Function controller is described in **Section 32B, USB 2.0 Function Module**.

32.1 Features

This interface complies the following specifications.

- Universal Serial Bus Specification Revision 2.0
- On-The-Go and Embedded Host Supplement to the USB Revision 2.0 Specification Revision 2.0 plus errata and ecn*1
- Enhanced Host Controller Interface (EHCI) Specification for Universal Serial Bus Revision 1.0
- EHCI v1.1 Addendum*2
- Open Host Controller Interface (OHCI) Specification for USB Release 1.0a
- Battery Charging Specification Revision 1.2*3

Note 1. Session Request Protocol (SRP) and Host Negotiation Protocol (HNP) are not supported.
Note 2. Some EHCI v1.1 feature are not supported. (Refer to **Section 32A, USB 2.0 Host Module** for detail)
Note 3. DCP (Dedicated Charging Port) is not supported as downstream port.

32.1.1 Ch0: USB2.0 OTG/DRD (Host/Function) Interface

The following table shows features of this interface.

Table 32.1 Ch0: USB2.0 OTG/DRD (Host/Function) Interface Features

Function	Description
Host mode	• Support mode: High-Speed (480 Mbps)/Full-Speed (12 Mbps)/Low-Speed (1.5 Mbps) • Support Isochronous/Interrupt/Control/Bulk transfer modes • Support Isochroous/Interrupt high bandwidth transfer
Function mode	• Support mode: High-Speed (480 Mbps)/Full-Speed (12 Mbps) • Isochronous/Interrupt/Control/Bulk transfer • Up to 10 ch PIPE (includes default control PIPE)
Other functions	• OTG function (Rev2.0) • Battery Charging function • DRD (Dual-Role-Device) function (Static switch between Host and Function)

32.1.2 Ch1: USB2.0 HOST Interface

The following table shows features of this interface.

Table 32.2 Ch1: USB2.0 Host Interface Function Features

Features	Description
Host mode	• Support mode: High-Speed (480 Mbps)/Full-Speed (12 Mbps)/Low-Speed (1.5 Mbps) • Support Isochronous/Interrupt/Control/Bulk transfer modes • Support Isochroous/Interrupt high bandwidth transfer
Other function	• Battery Charging function

32.1.3 Block Diagram

The block diagram of USB interface is as follows.

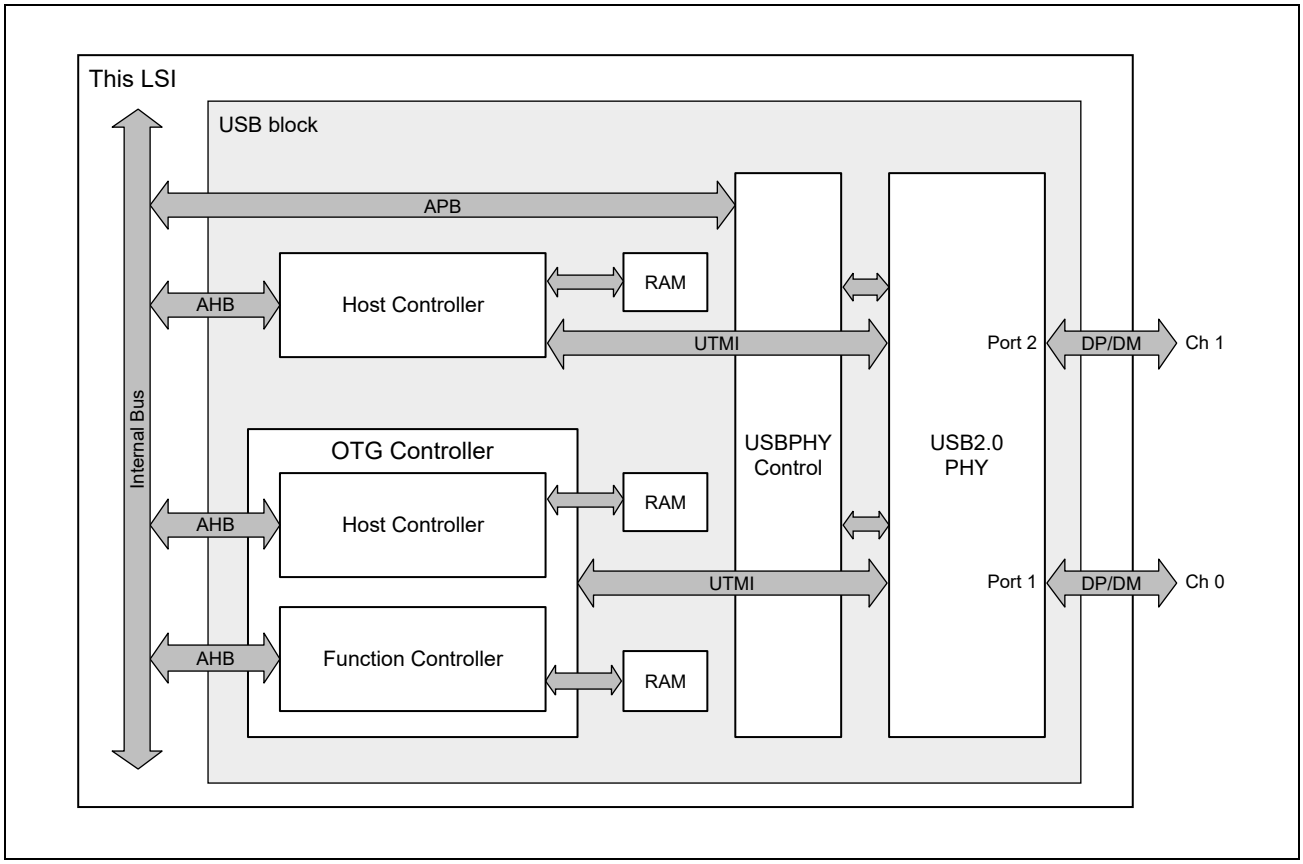


Figure 32.1 USB Interface Block Diagram

32.1.4 External Signal Pins

The following table shows external signal pins for USB block.

Table 32.3 External Signal Pins

Name	I/O	Description	Active Level	Initial Value	Handling in Not Used
USB0_DP	—	USB data pin Data+ (Ch0)	—	—	Open*1
USB0_DM	—	USB data pin Data− (Ch0)	—	—	Open*1
USB0_OVRCUR	I	Over current detection input (Ch0)	L	—	Pull-Up
USB0_VBUSEN	O	VBUS output enable (Ch0)	H	0b	Open
USB0_VBUSIN	I	Peripheral VBUS input (Ch0)	H	—	Pull-Down
USB0_OTG_EXICEN	O	External power IC control (Ch0)	H	0b	Open
USB0_OTG_ID	I	OTG ID input (Ch0)	—	—	Pull-Down
USB1_DP	—	USB data pin Data+ (Ch1)	—	—	Open*1
USB1_DM	—	USB data pin Data- (Ch1)	—	—	Open*1
USB1_OVRCUR	I	Over current detection input (Ch1)	L	—	Pull-Up
USB1_VBUSEN	O	VBUS output enable (Ch1)	H	0b	Open

Note 1. When it is not used permanently, connect to GND via a resistor of 10kΩ.

32.1.5 Power and GND Pins

The following table shows Power and GND pins for USB block.

Table 32.4 Power and GND pins

Name	Description
USB_VDD18	Internal Regulator Power (1.8 V), HS receiver power (1.8 V)
USB_VDD33	IO power (3.3 V)
USB_RREF	Reference voltage
VSS	GND

32.1.6 Interrupt

The following table shows interrupt list for USB block.

Table 32.5 USB Interrupt

Name	Description	Active Level	Type
U2H0_INT	USB2.0 Host AHB Interrupt (Ch0)	H	Level-Sensitive
U2H0_OHCI_INT	USB2.0 Host OHCI Interrupt (Ch0)	H	Level-Sensitive
U2H0_EHCI_INT	USB2.0 Host EHCI Interrupt (Ch0)	H	Level-Sensitive
U2H0_WAKEON_INT	USB2.0 Host EHCI Wakeup Interrupt (Ch0)	H	Level-Sensitive
U2H0_OBINT	USB2.0 Host OTG and Battery Charging Interrupt (Ch0)	H	Level-Sensitive
U2H1_INT	USB2.0 Host AHB Interrupt (Ch1)	H	Level-Sensitive
U2H1_OHCI_INT	USB2.0 Host OHCI Interrupt (Ch1)	H	Level-Sensitive
U2H1_EHCI_INT	USB2.0 Host EHCI Interrupt (Ch1)	H	Level-Sensitive
U2H1_WAKEON_INT	USB2.0 Host EHCI Wakeup Interrupt (Ch1)	H	Level-Sensitive
U2H1_OBINT	USB2.0 Host OTG and Battery Charging Interrupt (Ch1)	H	Level-Sensitive
U2P_IXL_INT	USB2.0 Function controller Interrupt (Ch0)	H	Edge-Triggered
U2P_INT_DMA[1:0]	USB2.0 Function controller DMA transaction complete interrupt (Ch0)	H	Level-Sensitive
U2P_INT_DMAERR	USB2.0 Function controller DMA error response interrupt (Ch0)	H	Level-Sensitive

32.2 Register Configuration

32.2.1 Host Controller Register

Refer to the **Section 32A, USB 2.0 Host Module**.

32.2.2 Function Controller Register

Refer to the **Section 32B, USB 2.0 Function Module**.

32.2.3 USBPHY Control Register

USBPHY Control Register mainly controls reset and power down of the USB/PHY.

Base Address: H'0_11E0_0000 (Cortex-A55 Address Space)
Base Address: H'41E0_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
Base Address: H'51E0_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

The following table shows USBPHY Control Register list.

Prohibit to write undefined area.

Table 32.6 USBPHY Control Register List

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Reset register	RESET	R/W	H'0000_0133	H'000	32
Clock control register	UCLKCTL	R/W	H'0340_0303	H'018	32
Direct power down setting register	UDIRPD	R/W	H'0000_0000	H'01C	32
VBUSEN control register	VBENCTL	R/W	H'0000_0000	H'03C	32
Connection control register	CON_CTRL	R/W	H'0000_0000	H'020	32
Clock status register	CLK_STAT	R	H'0000_0000	H'104	32

32.3 Register Descriptions

32.3.1 Reset Register (RESET)

This register controls USB/PHY reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	sel_pll reset	—	—	—	pllreset	—	—	sel_p2 reset	sel_p1 reset	—	—	phyrst_2	phyrst_1
Initial Value	0	0	0	0	0	0	0	1	0	0	1	1	0	0	1	1
R/W	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	sel_pllreset	0	R/W	Select USB/PHY PLL reset 1: register (by bit8: pllreset) 0: control by host controller (Ch0)
11 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	pllreset	1	R/W	PLL reset 1: PLL reset 0: deassert PLL reset
7, 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	sel_p2reset	1	R/W	Select USB/PHY Port2 reset 1: register(phyrst_2) 0: Control by host controller (Ch1)
4	sel_p1reset	1	R/W	Select USB/PHY Port1 reset 1: register(phyrst_1) 0: Control by host controller (Ch0)
3, 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	phyrst_2	1	R/W	USB/PHY reset (Port2) 1: USB/PHY reset 0: deassert USB/PHY reset Only when sel_p2reset=1, this bit is valid.
0	phyrst_1	1	R/W	USB/PHY reset (Port1) 1: USB/PHY reset 0: deassert USB/PHY reset Only when sel_p1reset=1, this bit is valid.

32.3.2 Clock Control Register (UCLKCTL)

This register controls USB/PHY clock.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	sel_sleepm	—	—	sleepm_2	sleepm_1	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	sel_suspendm	—	—	suspendm_2	suspendm_1	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1
R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	sel_sleepm	0	R/W	Select SLEEPM control 1: register (controlled by bit 24: sleepm_1 and bit 25: sleepm_2) 0: Control by controller
27, 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	sleepm_2	1	R/W	Sleep control (Port2) 1: Normal mode 0: Sleep mode
24	sleepm_1	1	R/W	Sleep control (Port1) 1: Normal mode 0: Sleep mode
23	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	—	1	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
21, 20	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
19, 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17, 16	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
15, 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
12	sel_suspendm	0	R/W	Select SUSPENDM control 1: register (controlled by bit 8: suspendm_1 and bit 9: suspendm_2) 0: control by controller
11, 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
9	suspend_2	1	R/W	Suspend control (Port2) 0: Suspend mode 1: Normal mode
8	suspend_1	1	R/W	Suspend control (Port1) 0: Suspend mode 1: Normal mode
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3, 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	—	All 1	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

32.3.3 Direct Power Down Setting Register (UDIRPD)

This register controls USB/PHY direct power down.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	sel_udirpd	—	—	—	dirpd
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	sel_udirpd	0	R/W	Select dirpd control 1: register (dirpd) 0: Control by host controller (Ch0)
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	dirpd	0	R/W	Power down control to reduce the power consumption of the USB/PHY when not using the USB function. 0: Normal operation 1: Power save

32.3.4 VBUSEN Control Register (VBENCTL)

This register controls VBUSEN connection.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	vbus_sel_0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	vbus_sel_0	0	R/W	Select VBUSEN control 0: Port Power bit of the EHCI/OHCI Operational register. 1: VBOUT bit of the VBUS Control register.

32.3.5 Connection Control Register (CON_CTRL)

This register controls USB/PHY connection.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	sel_connect	—	—	connect_2	connect_1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	sel_connect	0	R/W	Select USB connection notification control 1: register (controlled by bit0: connect_1 and bit1: connect_2) 0: Control by host controller
3, 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	connect_2	0	R/W	Notice of USB connection (port2) Used to notify UTM + core of USB connection status 0: USB not connected (single end receiver disabled) 1: USB not connected (single end receiver enabled)
0	connect_1	0	R/W	Notice of USB connection (port1) Used to notify UTM + core of USB connection status 0: USB not connected (single end receiver disabled) 1: USB not connected (single end receiver enabled)

32.3.6 Clock Status Register (CLK_STAT)

This register shows USB/PHY clock status.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	frclk60 lock	frclk48 lock	phy lock	—	—	plllock_ 2	plllock_ 1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	frclk60_lock	0	R	FRCLK60 LOCK status 0: UNLOCK 1: LOCK
5	frclk48_lock	0	R	FRCLK48 LOCK status 0: UNLOCK 1: LOCK
4	phylock	0	R	PHY PLL LOCK Status (Transceiver) 0: UNLOCK 1: LOCK
3, 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	plllock_2	0	R	PLL LOCK status (port2) 0: UNLOCK 1: LOCK
0	plllock_1	0	R	PLL LOCK status (port1) 0: UNLOCK 1: LOCK

32.4 Operation

32.4.1 USB/PHY Initialization

In the initial state, the reset of USB/PHY Port1 and Port2 is controlled by the register in the USBPHY Control Register, and the reset is asserted, so the control is switched to controller by the following initialization.

Also, in the initial state, the USB/PHY PLL Reset is controlled by the Host Controller in the OTG Controller, but it needs to be switched to controlled by the USBPHY Control Register.

- **Initialization when starting only Port1 of USB/PHY (when Port2 is not used)**

  Write H'0000_1133 to the reset register (RESET) of USBPHY Control Register, and then write H'0000_1022 to release the reset.
  (By the above register settings, USBPHY Control Register controls the PLL Reset and the reset is released. Port1 of USB/PHY is controlled by Host Controller, and Port2 is reset by USBPHY Control Register.)

- **Initialization when starting only Port2 of USB/PHY (when Port1 is not used)**

  Write H'0000_1133 to the reset register (RESET) of USBPHY Control Register, and then write H'0000_1011 to release the reset.
  (By the above register settings, USBPHY Control Register controls the PLL Reset and the reset is released. USB/PHY Port1 is reset by USBPHY Control Register, and port2 is controlled by Host Controller.)

- **Initialization when starting both Port1 and Port2 of USB/PHY**

  Write H'0000_1133 to the reset register (RESET) of USBPHY Control Register, and then write H'0000_1000 to release the reset.
  (By the above register settings, USBPHY Control Register controls the PLL Reset and the reset is released. USB/PHY Port1 is controlled by the Host Controller, and Port2 is controlled by the Host Controller.)

The common setting sequence both Host mode and Function mode including USB/PHY initialization is described in **Section 32A.9.1, Host/Peripheral Common Setting Sequence**.

32.4.2 Handling of Permanently Unused Pin

Take the following measures when USB interface is not used permanently.

32.4.2.1 USB/PHY Related Pins

(1) When both Port1 and Port2 are unused

Take the following measures for the USB pins and power supply.

- Connect the USB0_DP/DM and USB1_DP/DM to GND via a resistor of 10kΩ.
- Set the RREF to Open.
- Connect USB_VDD18 to a 1.8 V power supply. However, there is no need to separate the digital and analog power supplies.
- Connect USB_VDD33 to GND.
- Connect VSS to GND.

Also, perform the following control with software.

- Set the direct power down (via UDIRPD register). See **Section 32.4.3, Direct Power Down (DIRPD)** for more information.

(2) When either Port1 or Port2 is unused

Supply all power.

Since USB_VDD18/USB_VDD33 are common to 2 Port PHY, it is necessary to supply power even when one of the ports is not in use.

Make the following settings for unused port.

- Apply Port reset
  - Apply reset to unused USB/PHY port by controlling the Reset register (RESET).
  - Set sel_p1reset = 1 and phyrst_1 = 1 when USB/PHY Port1 is unused.
  - Set sel_p2reset = 1 and phyrst_2 = 1 when USB/PHY Port2 is unused.
- Power down setting
  - Set power down setting to unused USB/PHY port by controlling the Clock control register (UCLKCTL).
  - Set sel_suspendm=1 and suspendm_1=0 when USB/PHY Port1 is unused.
  - Set sel_suspendm=1 and suspendm_2=0 when USB/PHY Port2 is unused.
- Single-ended receiver stop
  - Stop single-ended receiver of unused USB/PHY port by controlling the Connection control register (CON_CTRL).
  - Set sel_connect=1 and connect_1=0 when USB/PHY Port1 is unused.
  - Set sel_connect=1 and connect_2=0 when USB/PHY Port2 is unused.

32.4.2.2 Handling of USB Controller Related Pins

(1) When Host controller is unused

The USB0_VBUSEN pin leaves open and fix the USB0_OVRCUR pin to 1 when USB/PHY port1 is unused.

The USB1_VBUSEN pin leaves open and fix the USB1_OVRCUR pin to 1 when USB/PHY port2 is unused.

(2) Function controller is unused

Fix the USB0_VBUSIN pin to 0 when USB/PHY port1 is unused.

32.4.3 Direct Power Down (DIRPD)

By directly controlling the power down signal of the USB/PHY with the USBPHY Control register, the USB/PHY can be put into a low power state regardless of the state of the controller.

When set to the direct power down state, all USB/PHY functions including the PLL will be stopped. Direct power down can only be used when the USB/PHY is unused (means both ports are unused).

Follow the procedure below to switch to / return to the direct power down mode.

For details on the USBPHY Control Register reset register and direct power down setting register used in the procedure below, refer to **Section 32.3.1, Reset Register (RESET)** and **Section 32.3.3, Direct Power Down Setting Register (UDIRPD)**, respectively.

32.4.3.1 Enter to Direct Power Down Mode

Follow the steps below to switch to direct power down mode.

(1) USB operation stop

Stop the operation of USB.

(2) Assert the direct power down signal.

It is asserted by setting H'0000_0010 in the direct power-down setting register (UDIRPD) and then setting H'0000_0011.

There are no timing restrictions when transitioning to direct power down. Immediately after setting, it shifts to power down mode.

32.4.3.2 Resume from Direct power down mode

Follow the steps below to recover from the direct power down mode.

(1) Assert a USB/PHY PLL reset.

The PLL reset is asserted by setting 1 to sel_pllreset bit of the Reset register (RESET) and then setting 1 to pllreset bit.

The reset period should be 1us or more.

(2) De-assert direct power down signal

Negate the direct power down signal after continuing to assert the PLL reset for more than 1us. It is negated by setting H'0000_0010 in the direct power down setting register (UDIRPD) and then setting H'0000_0000.

After negating the direct power down signal, keep asserting the PLL reset for at least 1µs.

(3) De-assert USB/PHY PLL reset

After negating the direct power down signal, continue to assert the PLL reset for 1us or more, and then negate the PLL reset.

The PLL reset is negated by clearing the pllreset bit in the reset register (RESET) and then clearing the sel_pllreset.

The timing chart when returning from the direct putter down is shown below.

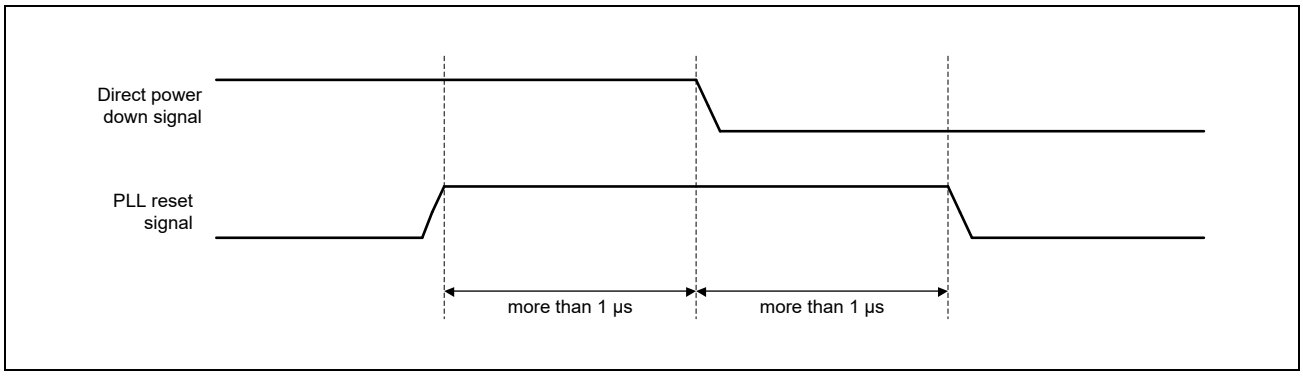


Figure 32.2 Resume from Direct Power Down Mode

32.5 Usage Notes

32.5.1 Precautions in Power Down Mode

When USB interface is unused, it can be set to power-down mode by register setting. For the setting, refer to **Section 32.3.3, Direct Power Down Setting Register (UDIRPD)**.

The precautions for power down mode are as follows.

- Once enter to power down, do not access the interface.
- Move to the power-down state after the USB is in unused state. If USB interface moves power-down state while it is operating, current may continue to flow to the USB side and it may damage device.

32A. USB 2.0 Host Module

32A.1 Overview

32A.1.1 Overview

This LSI has two USB 2.0 host/function modules. For each module, you can switch between the host mode and the peripheral mode by specifying the UCOM register setting. This section describes the circuits that are common to both modes, and the host controller itself.

32A.1.2 Features

This module has the following features:

Function	Description
Host function	• Supporting high-speed (480 Mbps), full-speed (12 Mbps), and low-speed (1.5 Mbps) transfers • Compliant with Open Host Controller Interface (OHCI) Specification for USB Revision 1.0a • Compliant with Enhanced Host Controller Interface (EHCI) Specification for USB Revision 1.1*1
Other functions	• Battery charging (compliant with Battery Charging Specification Revision 1.2)*2 • Dual-role-device function (static switching between the USB host and USB peripheral functions)

Note 1. Some functions (specifications) are not supported.

Note 2. The setting for battery charging is handled by the host controller even if the function controller is selected. This LSI does not support DCP (Dedicated Charging Port) as a Downstream Port.

32A.1.2.1 EHCI v1.1 functions

Conventional host controller modules comply with EHCI Specification Revision.1.0. Meanwhile, this module supports the additional functions listed below to comply with EHCI v1.1 Addendum.

Note, however, that this module supports only the functions marked with a circle "✓" in the table below.

Function Name	Support
Per-Port Change Events	✓
Shorter Periodic Frame List	✓
Hardware Prefetching	—
Link Power Management (LPM)	✓

Of the three features to support, the registers associated with "Per-Port Change Events" and "Shorter Periodic Frame List" are listed below. For more information on each register, refer to the appropriate register specification of the **Section 32A.2, Register Descriptions**.

With regard to "Link Power Management (LPM)", it is described in **Section 32A.1.2.2, Link Power Management (LPM) function**.

Table 32A.1 Per-Port Change Events

Relevant Register	Relevant Bit		Attribute	Function
HCCPARAMS (offset: H'108)	bit 18	Per-Port Change Event Capability	R	This bit indicates to support this function. This bit is fixed to 1b.
USBCMD (offset: H'120)	bit 15	Per-Port Change Events Enable	R/W	Per-Port event notification setting. Writing 1b to this bit enables the Per-Port event notification function.
USBSTS (offset: H'124)	bit 16	Port-1 Change Detect	R/W(1)	If a port-1 change event is detected, 1b is written to this bit.
USBINTR (offset: H'128)	bit 16	Port-1 Change Event Enable	R/W	Enable/Disable setting of the above "port-1 Change Detect" field. To reflect the port event on the above field for the port, set corresponding bit to 1b.

Table 32A.2 Shorter Periodic Frame List

Relevant Register	Relevant Bit		Attribute	Function
HCCPARAMS (offset: H'108)	bit 19	32-Frame Periodic List Capability	R	This bit indicates to support this function. This bit is fixed to 1b.
USBCMD (offset: H'120)	bit[3:2]	Frame List Size	R/W	This bit determines the Frame List Size. To determine 32-frames, set this field 11b.

32A.1.2.2 Link Power Management (LPM) function

This module supports the power management function conforming to “USB 2.0 Link Power Management Addendum to the Universal Serial Bus v2.0 Specification” (abbreviated as LPM).

When the peripheral device which supports LPM function is connected to this module, the device can be moved to suspend state faster than conventional by using LPM function.

*In using LPM function: 8 to 10 µs

The relevant registers are listed in the following. Refer to the appropriate register specification of each register detail in **Section 32A.2, Register Descriptions**.

Relevant Register	Relevant Bit		Attribute	Function
HCCPARAMS (offset: H’108)	bit 17	Link Power Management Capability	R	This bit indicates to support this function. This bit is fixed to 1b.
USBCMD (offset: H’120)	bit[27:24]	Host-Initiated Resume Duration	R/W	The minimum duration time of K-state drive in resume from LPM state.
PORTSC1 (offset: H’164)	bit[31:25]	Device Address [7:0]	R/W	The device address of the destination of LPM Token. In using LPM function, this setting is needed. Set the address of the device connected to the corresponding port before sending LPM Token.
	bit[24:23]	Suspend Status [1:0]	R	The device response to LPM Token.
	bit 9	Suspend using L1	R/W	In using LPM function, set this bit 1b.
	bit 7	Suspend	R/W	When this bit is set to 1b this module starts LPM Token transaction, in condition of “Suspend using L1 bit = 1b” and “Device address field = H’000”.
PORT_LPM_CTR1 (offset: H’320)	bit[7:4]	NYET_RETRY_CNT_P1[3:0]	R/W	The number of retry for NYET response of device in LPM transaction.
	bit 3	REMOTEWAKE_EN_P1	R/W	Setting of LPM RemoteWakeup permission. 0b: permitted (default) 1b: not permitted
	bit 2	SLEEP_INT_EN_P1	R/W	In LPM transaction, setting to generate an interrupt or not when received a response other than ACK. 0b: not to generate an interrupt (default) 1b: to generate an interrupt
	bit 1	RETRY_ENABLE_NYET_P1	R/W	In LPM transaction, setting to the host behavior when received NYET response from the device. 0b: not to Retry (default) 1b: to Retry
	bit 0	HIRD_SEL_P1	R/W	Setting the duration time of the K-state drive, in resume of LPM state.

32A.1.2.3 OTG function

Detects changing of the USB0_OTG_ID pin by U2H0_OBINT interrupt and switches the Host/Function role. Confirm the IDCHG_STA bit of **Section 32A.2.4.5(2), OTG-BC Interrupt Status Register** and the IDMON bit of **Section 32A.2.4.5(5), Line Control Port 1 Register**.

Host/Function role switching is performed by the OTG_PERI bit of **Section 32A.2.4.5(1), Common Control Register**.

OTG_PERI bit
0 : Host mode
1 : Function mode (Only CH0 can be set)

32A.1.2.4 Battery-charging function

The Control and monitoring of the Battery Charging I/F of the USBPHY is set in the UCOM register of this module. The detail of the UCOM register, refer to **Section 32A.2.4.5, UCOM Register**.

32A.1.2.5 Suspend extension function

This module implements the following two suspend extension functions to reduce power consumption by stopping PLL of USBPHY.

The detail of the register, refer to **Section 32A.2.4.4(6), Suspend Control Register**.

[Relevant register]

Suspend Control Register (offset: H'308)

[Functional specification]

Function		Relevant Bit	
[1]	Function to assert USBPHY SUSPENDM by asserting the Suspend bit in the OHCI/EHCI Operational Register	bit 31	SUSPENDM_ENABLE
[2]	Function to forcibly assert USBPHY SUSPENDM	bit 0	GLOBAL_SUSPENDM_P1

32A.1.3 Support of USB-Related Specifications

USB-Related Specification or Function			Support
Host function	High Speed	Bulk IN/OUT transfer	✓
		Control IN/OUT transfer	✓
		Isochronous IN/OUT transfer	✓
		Isochronous Highband transfer	✓
		Interrupt IN/OUT transfer	✓
	Full Speed	Bulk IN/OUT transfer	✓
		Control IN/OUT transfer	✓
		Isochronous IN/OUT transfer	✓
		Interrupt IN/OUT transfer	✓
	Low Speed	Control IN/OUT transfer	✓
		Interrupt IN/OUTtransfer	✓
	No. of hub connection stages	HS: 5 stages	✓
		FS: 5 stages	✓
	Support of EHCI V1.1	Hardware Prefetching	×
		Link Power Management	✓
		Per-Port Change Events	✓
		Shorter Periodic Frame List	✓
Battery-charging function (hereafter called the “BC function”)			✓
Dual role device * Function to statically switch between the host and peripheral modes			✓

Remarks: ✓: support
×: no support

32A.2 Register Descriptions

32A.2.1 Register Attributes

Table 32A.3 Register Attributes

Register Attribute	Description
R/W	Register bits can be read and written.
R/W(1)	Register bits can be read. A clear bit may be set by writing “1”; writing 0 to R/W(1) bits has no effect.
R/W(0)	Register bits can be read. A clear bit may be set by writing “0”; writing 1 to R/W(0) bits has no effect.
R	Register bits can only be read.
W	Register bits can only be written.
Reserved	Reserved bits are Read Only field.

32A.2.2 Base Address

Table 32A.4 Base Addresses for Each Channel of the USB Host Module

Channel	Base Address	
0	H’0_11E1_0000	(Cortex-A55 Address Space)
	H’41E1_0000	(Cortex-M33/Cortex-M33_FPU Address Space Secure)
	H’51E1_0000	(Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)
1	H’0_11E3_0000	(Cortex-A55 Address Space)
	H’41E3_0000	(Cortex-M33/Cortex-M33_FPU Address Space Secure)
	H’51E3_0000	(Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Note: Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

32A.2.3 Register Overview

Register Name	Abbreviated Name	R/W	Initial Value	Offset Address	Access Size	Remarks
HcRevision	HcRevision	R	H'0000_0000	H'000	32	OHCI Operation Registers
HcControl	HcControl	R/W	H'0000_0000	H'004	32	
HcCommandStatus	HcCommandStatus	R/W	H'0000_0000	H'008	32	
HcInterruptStatus	HcInterruptStatus	R/W	H'0000_0000	H'00C	32	
HcInterruptEnable	HcInterruptEnable	R/W	H'0000_0000	H'010	32	
HcInterruptDisable	HcInterruptDisable	R/W	H'0000_0000	H'014	32	
HcHCCA	HcHCCA	R/W	H'0000_0000	H'018	32	
HcPeriodCurrentED	HcPeriodCurrentED	R	H'0000_0000	H'01C	32	
HcControlHeadED	HcControlHeadED	R/W	H'0000_0000	H'020	32	
HcControlCurrentED	HcControlCurrentED	R/W	H'0000_0000	H'024	32	
HcBulkHeadED	HcBulkHeadED	R/W	H'0000_0000	H'028	32	
HcBulkCurrentED	HcBulkCurrentED	R/W	H'0000_0000	H'02C	32	
HcDoneHead	HcDoneHead	R	H'0000_0000	H'030	32	
HcFmInterval	HcFmInterval	R/W	H'0000_2EDF	H'034	32	
HcFmRemaining	HcFmRemaining	R	H'0000_2EDF	H'038	32	
HcFmNumber	HcFmNumber	R	H'0000_0000	H'03C	32	
HcPeriodicStart	HcPeriodicStart	R/W	H'0000_0000	H'040	32	
HcLSThreshold	HcLSThreshold	R/W	H'0000_0628	H'044	32	
HcRhDescriptorA	HcRhDescriptorA	R/W	H'0F00_0901	H'048	32	
HcRhDescriptorB	HcRhDescriptorB	R/W	H'0002_0000	H'04C	32	
HcRhStatus	HcRhStatus	R/W	H'0000_0000	H'050	32	
HcRhPortStatus1	HcRhPortStatus1	R/W	H'0000_0000	H'054	32	
HCIVERSION / CAPLENGTH	CAPL_VERSION	R	H'0110_0020	H'100	32	EHCI Capability Registers
HCSPARAMS	HCSPARAMS	R	H'0000_1191	H'104	32	
HCCPARAMS	HCCPARAMS	R	H'000E_0006	H'108	32	
HCSP_PORTROUTE	HCSP_PORTROUTE	R	H'0000_0000	H'10C	32	
USBCMD	USBCMD	R/W	H'0008_0B00	H'120	32	EHCI Operation Registers
USBSTS	USBSTS	R/W	H'0000_1000	H'124	32	
USBINTR	USBINTR	R/W	H'0000_0000	H'128	32	
FRINDEX	FRINDEX	R/W	H'0000_0000	H'12C	32	
CTRLDSSEGMENT	CTRLDSSEGMENT	R	H'0000_0000	H'130	32	
PERIODICLISTBASE	PERIODICLISTBASE	R/W	H'0000_0000	H'134	32	
ASYNCLISTADDR	ASYNCLISTADDR	R/W	H'0000_0000	H'138	32	
CONFIGFLAG	CONFIGFLAG	R/W	H'0000_0000	H'160	32	
PORTSC1	PORTSC1	R/W	H'0000_2000	H'164	32	
INT_ENABLE	INT_ENABLE	R/W	H'0000_0000	H'200	32	AHB Registers
INT_STATUS	INT_STATUS	R/W	H'0000_0000	H'204	32	
AHB_BUS_CTR	AHB_BUS_CTR	R/W	H'0000_0000	H'208	32	
USBCTR	USBCTR	R/W	H'0000_0002	H'20C	32	

Register Name	Abbreviated Name	R/W	Initial Value	Offset Address	Access Size	Remarks
Register Enable/Clock Gating Control	REGEN_CG_CTRL	R/W	H'0000_0000	H'304	32	Core Defined Registers
Suspend Control	SPD_CTRL	R/W	H'0000_0000	H'308	32	
Suspend/Resume Timer Setting	SPD_RSM_TIMSET	R/W	H'01F4_03E8	H'30C	32	
Overcurrent Detection/Sleep Timer Setting	OC_SLP_TIMSET	R/W	H'0C83_0D40	H'310	32	
SBRN/FLADJ/PORTWAKECAP	SBRN_FLADJ_PW	R/W	H'0003_2020	H'314	32	
PORT_LPM_CTRL1	PORT_LPM_CTRL1	R/W	H'0000_0000	H'320	32	
Common Control	COMMCTRL	R/W	H'8000_0000	H'800	32	OTG/BC Module Control Register
OTG-BC Interrupt Status	OBINTSTA	R/W	H'0000_0001	H'804	32	OTG/BC Interrupt Status Register
OTG-BC Interrupt Enable	OBINTEN	R/W	H'0000_0000	H'808	32	OTG/BC Interrupt Enable Register
VBUS Control	VBCTRL	R/W	H'0001_0000	H'80C	32	OTG VBUS Control Register
Line Control Port 1	LINECTRL1	R/W	H'0000_0000	H'810	32	OTG USB Bus Control Register (Port 1)
BC Control Port 1	BCCTRL1	R/W	H'0300_0000	H'820	32	Battery Charging Control Register (Port 1)

32A.2.4 Description of Registers

32A.2.4.1 OHCI Operational Register

(1) HcRevision Register

Abbreviated name of register: HcRevision

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	Revision							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved The write value should always be 0.
7 to 0	Revision	All 0	R	This field indicates the version of HCI specifications implemented in this host controller module. Because this module conforms to OHCI standard 1.0a, H'10 is indicated.

(2) HcControl Register

Abbreviated name of register: HcControl

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RWC	—	HCFS		BLE	CLE	IE	PLE	CBSR	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved The write value should always be 0.
9	RWC	0	R/W	Remote Wakeup Connected This bit indicates whether the host controller supports remote wakeup signaling. Software should set this bit to 1b in the initialization sequence if it is required to support remote wakeup. Note that this bit can be initialized at hardware reset only. 0b: The remote wakeup is not supported. 1b: The remote wakeup is supported
8	—	0	R	Reserved The write value should always be 0.
7, 6	HCFS	All 0	R/W	Host Controller Functional State This field indicates the operating state of the host controller. 00b: USB Reset 01b: USB Resume 10b: USB Operational 11b: USB Suspend When the state is changed to USB Operational, the host controller module starts SOF transmission at 1 ms boundary. This field is basically controlled by software, but it can be controlled by the host controller in the state of USB Suspend only. If the host controller (in the state of USB Suspend) detects a Remote Wakeup signal from the connecting device, the state in this field is changed to USB Resume. Note that the reset value of this field differs between hardware reset and software reset. Hardware reset: 00b (USB Reset) Software reset: 11b (USB Suspend)
5	BLE	0	R/W	BulkList Enable This bit sets whether the bulk list processing is performed for the next frame. The setting value of this bit is enabled from the next frame. When you correct the bulk list, this bit must be 0b. 0b: The processing of the Bulk list is disabled. 1b: The processing of the Bulk list is enabled
4	CLE	0	R/W	ControlList Enable This bit sets whether the control list processing is performed for the next frame. The setting value of this bit is enabled from the next frame. When you correct the control list, this bit must be 0b. 0b: The processing of the Control list is disabled. 1b: The processing of the Control list is enabled.

Bit	Bit Name	Initial Value	R/W	Description
3	IE	0	R/W	Isochronous Enable This bit sets whether the isochronous ED processing is performed. The setting value of this bit is enabled from the next frame. If the host controller module detects isochronous ED (F = 1) during the periodic list processing, it checks the bit and determines whether to perform isochronous ED processing. 1b: The processing of the isochronous ED is continued. 0b: The periodic list processing is stopped, and the bulk/control list processing is started. 0b: The processing of the isochronous ED is disabled. 1b: The processing of the isochronous ED is enabled.
2	PLE	0	R/W	Periodic List Enable This bit indicates whether the periodic list processing is performed for the next frame. The setting value of this bit is enabled from the next frame. The host controller module checks this bit before starting the periodic list processing. 0b: The processing of the periodic list is disabled. 1b: The processing of the periodic list is enabled.
1, 0	CBSR	All 0	R/W	Control Bulk Service Ratio This field defines the service ratio of the control transfer and bulk transfer. When the periodic list is processed, the service ratio defined in this field is used for transfer.

CBSR	No. of Control EDs Over Bulk EDs Served
00b	1:1
01b	2:1
10b	3:1
11b	4:1

(3) HcCommandStatus Register

Abbreviated name of register: HcCommandStatus

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOC	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	BLF	CLF	HCR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved The write value should always be 0.
17, 16	SOC	All 0	R	Scheduling Overrun Count This field counts the number of scheduling overrun. This field is initialized to 00b, and is counted up every time scheduling overrun is detected. After the field is incremented to 11b, it returns to 00b. Even if the SO (Scheduling Overrun) bit in the HcInterrupt Status register is set, this field is counted up when scheduling overrun is detected.
15 to 3	—	All 0	R	Reserved The write value should always be 0.
2	BLF	0	R/W	BulkList Filled This bit indicates whether TD exists in the bulk list. To add TD to ED in the bulk list, set this bit to 1b. The host controller checks this bit when it starts processing of the head ED in the bulk list. If this bit is set to 0b, the host controller does not start the list processing. If this bit is set to 1b, the host controller starts processing of the bulk list, and sets this bit to 0b. When the host controller finds TD in the bulk list, it sets this bit to 1b again, and continues processing of the bulk list. When the host controller finishes the list processing, this bit is set to 0b. However, if TD is not found in the bulk list, or if this bit is not set to 1b, this bit remains to be 0b, and the list processing stops. To rebuild the list and start the list processing, before you set the BLE bit of the HcControl register and start the list processing, you need to set this bit. 0b: TD does not exist in the bulk list. 1b: TD exists in the bulk list.

Bit	Bit Name	Initial Value	R/W	Description
1	CLF	0	R/W	ControlList Filled This bit indicates whether TD exists in the control list. To add TD to ED in the control list, set this bit to 1b. The host controller checks this bit when it starts processing of the head ED in the control list. If this bit is set to 0b, the host controller does not start the processing of the control list. If this bit is set to 1b, the host controller starts processing of the control list, and set this bit to 0b. When the host controller finds TD in the control list, it sets this be to 1b again, and continues the list processing. When the host controller finishes the list processing, this bit is set to 0b. However, if TD is not found in the control list, or if this bit is not set to 1b, this bit remains to be 0b, and the list processing stops. To rebuild the list and start the list processing, before you set the CLE bit of the HcControl register and start the list processing, you need to set this bit. 0b: TD does not exist in the control list. 1b: TD exists in the control list.
0	HCR	0	W	Host Controller Reset This bit is used to start OHCI software reset for the host controller. When this bit is set to 1b, the operating status of the host controller is changed to USB Suspend regardless of the functional state of the host controller. Also, the most OHCI Operational registers and OHCI control circuits are initialized. When the software reset finishes, the host controller clears this bit to 0b.

(4) HcInterruptStatus Register

Abbreviated name of register: HcInterruptStatus

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RHSC	FNO	UE	RD	SF	WDH	SO
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved The write value should always be 0.
6	RHSC	0	R/W	Root Hub Status Change Interrupt bit that indicates that the state of the HcRhStatus register or HcRhPortStatus1 register has changed. When the Root Hub status is changed, the host controller sets this bit to 1b. Writing 1b to this bit clears the interrupt. 0b: The status of root hub has not changed. 1b: The status of root hub has changed.
5	FNO	0	R/W	Frame Number Overflow Interrupt bit that indicates that the MSB in the FrameNumber (FN) field of theHcFmNumber register has changed. When the MSB in the Frame Number field is changed from 0 to 1, or from 1 to 0, this bit is set after HccaFrameNumber is updated. Writing this bit to 1b clears the interrupt. 0b: The overflow of frame number has not occurred. 1b: The overflow of frame number has occurred.
4	UE	0	R/W	Unrecoverable Error Interrupt bit that indicates that a system error that is not related to the USB (for example, an error on the system bus) has been detected. Writing this bit to 1b clears the interrupt. 0b: The unrecoverable error has not occurred. 1b: The unrecoverable error has occurred.
3	RD	0	R/W	Resume Detected Interrupt bit that indicates that Resume has been detected. When the host controller detects the Resume signal (RemoteWakeup) from an USB device, it sets this bit to 1b. This bit is not set when the Resume signal is sent by setting the HCFS field to USB Resume. Writing this bit to 1b clears the interrupt. 0b: The host controller has not detected resume signaling (RemoteWakeup). 1b: The host controller has detected resume signaling (RemoteWakeup).

Bit	Bit Name	Initial Value	R/W	Description
2	SF	0	R/W	Start of Frame Interrupt bit that indicates that Hcca Frame Number was updated when each frame started. The host controller sends an SOF packet and updates HccaFrameNumber at the same time, and sets this bit to 1b. Writing this bit to 1b clears the interrupt. 0b: The host controller has not started new frame. 1b: The host controller has started new frame.
1	WDH	0	R/W	Writeback DoneHead Interrupt bit that indicates that the host controller has written the contents of HcDoneHead to HccDoneHead. The host controller sets this bit to 1b immediately after it updates HccaDoneHead. Note that HccaDoneHead is not updated until this bit is cleared. Writing this bit to 1b clears the interrupt. This bit must be cleared only after the contents of HccaDoneHead are saved. 0b: The write back to HccaDoneHead has not occurred. 1b: The write back to HccaDoneHead has occurred.
0	SO	0	R/W	Scheduling Overrun Interrupt bit that indicates that overrun of the USB schedule occurred. When USB scheduling overrun occurs, the host controller updates HccaFrameNumberl, and sets this bit to 1b. When this bit is set, the SchedulingOverrunCount field of the HcCommandStatus register is also incremented. Writing this bit to 1b clears the interrupt. 0b: The scheduling overrun has not occurred. 1b: The scheduling overrun has occurred.

(5) HcInterruptEnable Register

Abbreviated name of register: HcInterruptEnable

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MIE	OCE	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RHSCE	FNOE	UEE	RDE	SFE	WDHE	SOE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	MIE	0	R/W	Master Interrupt Enable This bit sets whether each Interrupt Enable setting (which is set in HcInterruptEnable [30:0]) is enabled. If this bit is set to 0b, all OHCI interrupts are masked. To clear this bit, set the MID bit (bit 31) of the HcInterruptDisable register to 1b. 0b: All interrupts are disabled. 1b: Interrupts that are set to 1b are enabled.
30	OCE	0	R/W	OC (Ownership Change) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes OC (OwnershipChange). To clear this bit, set the OCD bit (bit 30) of the HcInterruptDisable register to 1b. 0b: OC (OwnershipChange) interrupt is disabled. 1b: OC (OwnershipChange) interrupt is enabled.
29 to 7	—	All 0	R	Reserved The write value should always be 0.
6	RHSCE	0	R/W	RHSC (RootHub Status Change) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes RHSC (RootHub Status Change). To clear this bit, set the RHSCD bit (bit 6) of the HcInterruptDisable register to 1b. 0b: RHSC (RootHub Status Change) interrupt is disabled. 1b: RHSC (RootHub Status Change) interrupt is enabled.
5	FNOE	0	R/W	FNO (Frame Number Overflow) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes FNO (Frame Number Overflow). To clear this bit, set the FNOD bit (bit 5) of the HcInterruptDisable register to 1b. 0b: FNO (Frame Number Overflow) interrupt is disabled. 1b: FNO (Frame Number Overflow) interrupt is enabled.
4	UEE	0	R/W	UE (Unrecoverable Error) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes UE (Unrecoverable Error). To clear this bit, set the UED bit (bit 4) of the HcInterruptDisable register to 1b. 0b: UE (Unrecoverable Error) interrupt is disabled. 1b: UE (Unrecoverable Error) interrupt is enabled.
3	RDE	0	R/W	RD (Resume Detect) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes RD (Resume Detect). To clear this bit, set the RDD bit (bit 3) of the HcInterruptDisable register to 1b. 0b: RD (Resume Detect) interrupt is disabled. 1b: RD (Resume Detect) interrupt is enabled.

Bit	Bit Name	Initial Value	R/W	Description
2	SFE	0	R/W	SF (Start of Frame) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes SF (Start of Frame). To clear this bit, set the SFD bit (bit 2) of the HcInterruptDisable register to 1b. 0b: SF (Start of Frame) interrupt is disabled. 1b: SF (Start of Frame) interrupt is enabled.
1	WDHE	0	R/W	WDH (Writeback DoneHead) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes WDH (Writeback DoneHead). To clear this bit, set the WDHDD bit (bit 1) of the HcInterruptDisable register to 1b. 0b: WDH (Writeback DoneHead) interrupt is disabled. 1b: WDH (Writeback DoneHead) interrupt is enabled.
0	SOE	0	R/W	SO (Scheduling Overrun) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes SO (Scheduling Overrun). To clear this bit, set the SOD bit (bit 0) of the HcInterruptDisable register to 1b. 0b: SO (Scheduling Overrun) interrupt is disabled. 1b: SO (Scheduling Overrun) interrupt is enabled.

(6) HcInterruptDisable Register

Abbreviated name of register: HcInterruptDisable

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MID	OCD	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W(1)	R/W(1)	R	R	R	R	R	R	R	R	R	R	R	R	R	R

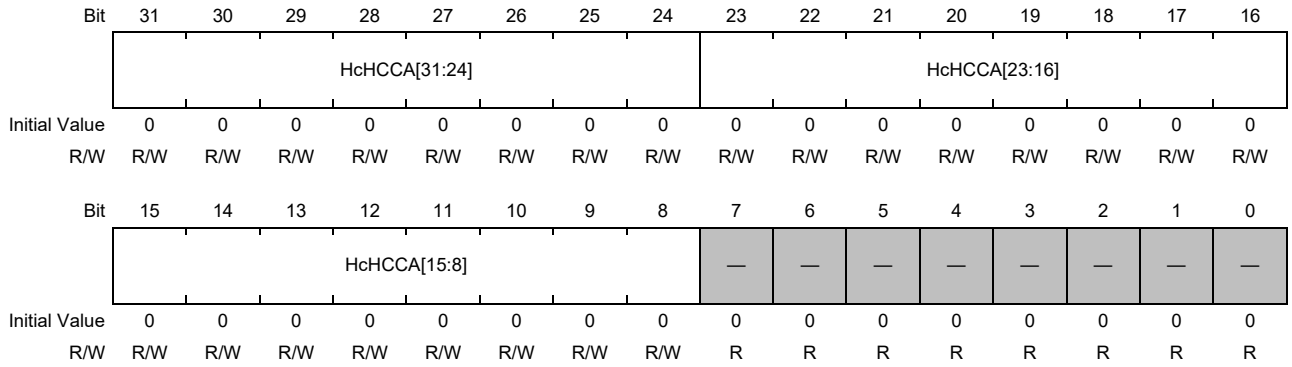
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RHSCD	FNOD	UED	RDD	SFD	WDHD	SOD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)

Bit	Bit Name	Initial Value	R/W	Description
31	MID	0	R/W (1)	Master Interrupt Disable This bit sets whether the Enable setting (for each interrupt) that is set by HcInterruptEnable[30:0] is disabled. If this bit is set to 1b, the MIE bit (bit 31) of the HcInterruptEnable register is cleared to 0b, and all OHCI interrupts are masked. Writing 0b to this bit is ignored.
30	OCD	0	R/W (1)	OC (Ownership Change) Interrupt Disable bit. If this bit is set to 1b, the OCE bit (bit 30) of the HcInterruptEnable register is cleared to 0b, and OC (Ownership Change) is excluded from the interrupt source. Writing 0b to this bit is ignored.
29 to 7	—	All 0	R	Reserved The write value should always be 0.
6	RHSCD	0	R/W (1)	RHSC (RootHub Status Change) Interrupt Disable bit. If this bit is set to 1b, the RHSCE bit (bit 6) of the HcInterruptEnable register is cleared to 0b, and RHSC (RootHub Status Change) is excluded from the interrupt source. Writing 0b to this bit is ignored.
5	FNOD	0	R/W (1)	FNO (Frame Number Overflow) Interrupt Disable bit. If this bit is set to 1b, the FNOE bit (bit 5) of the HcInterruptEnable register is cleared to 0b, and FNO (Frame Number Overflow) is excluded from the interrupt source. Writing 0b to this bit is ignored.
4	UED	0	R/W (1)	UE (Unrecoverable Error) Interrupt Disable bit. If this bit is set to 1b, the UEE bit (bit 4) of the HcInterruptEnable register is cleared to 0b, and UE (Unrecoverable Error) is excluded from the interrupt source. Writing 0b to this bit is ignored.
3	RDD	0	R/W (1)	RD (Resume Detected) Interrupt Disable bit. If this bit is set to 1b, the RDE bit (bit 3) of the HcInterruptEnable register is cleared to 0b, and RD (Resume Detected) is excluded from the interrupt source. Writing 0b to this bit is ignored.
2	SFD	0	R/W (1)	SF (Start of Frame) Interrupt Disable bit. If this bit is set to 1b, the SFE bit (bit 2) of the HcInterruptEnable register is cleared to 0b, and SF (Start of Frame) is excluded from the interrupt source. Writing 0b to this bit is ignored.
1	WDHD	0	R/W (1)	WDH (Writeback DoneHead) Interrupt Disable bit. If this bit is set to 1b, the WDHE bit (bit 1) of the HcInterruptEnable register is cleared to 0b, and WDH (Writeback DoneHead) is excluded from the interrupt source. Writing 0b to this bit is ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	SOD	0	R/W (1)	SO (Scheduling Overrun) Interrupt Disable bit. If this bit is set to 1b, the SOE bit (bit 1) of the HcInterruptEnable register is cleared to 0b, and SO (Scheduling Overrun) is excluded from the interrupt source. Writing 0b to this bit is ignored.

(7) HcHCCA Register

Abbreviated name of register: HcHCCA



Bit	Bit Name	Initial Value	R/W	Description
31 to 8	HcHCCA	All 0	R/W	This field sets the base address (of the RAM) that is assigned as the Host Controller Communication Area. This field must be set at initialization. The host controller requests (as HCCA) 256-byte area from the base address specified in this field.
7 to 0	—	All 0	R	Reserved The write value should always be 0.

(8) HcPeriodicCurrentED Register

Abbreviated name of register: HcPeriodCurrentED

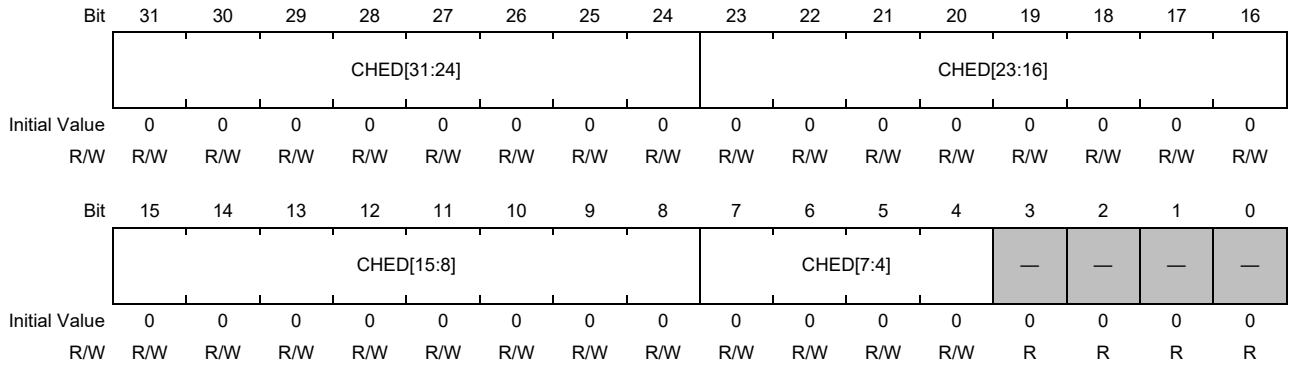
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PECD[31:24]								PECD[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PECD[15:8]								PECD[7:4]				—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	PECD	All 0	R	Period Current ED This pointer indicates the physical address of ED in the periodic list that is currently processed. The host controller updates this pointer when the list processing of the periodic ED finishes.
3 to 0	—	All 0	R	Reserved The write value should always be 0.

(9) HcControlHeadED Register

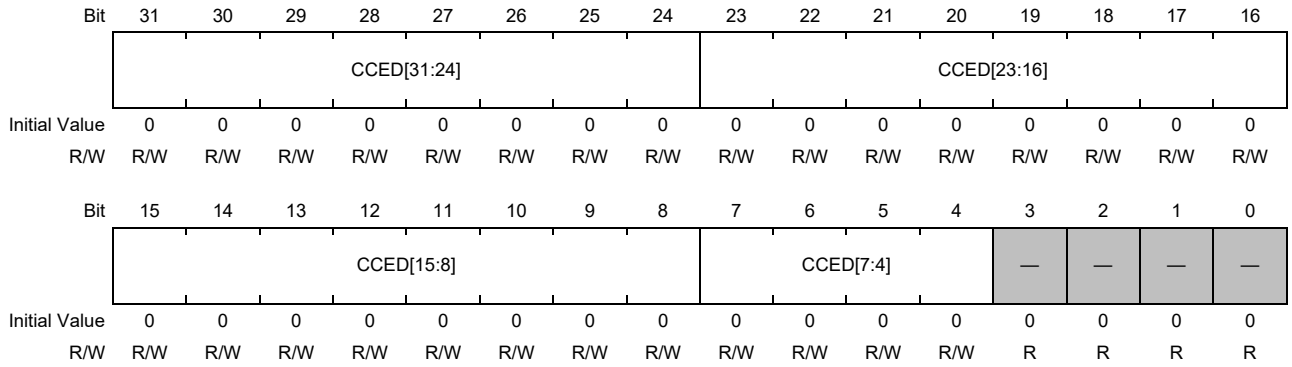
Abbreviated name of register: HcControlHeadED



Bit	Bit Name	Initial Value	R/W	Description
31 to 4	CHED	All 0	R/W	Control Head ED This field specifies the physical address of the head ED of the control list. These bits must be set for control transfer before the CLE bit of the HcControl register is set. The host controller starts processing of the control list from the HcBulkHeadED pointer.
3 to 0	—	All 0	R	Reserved The write value should always be 0.

(10) HcControlCurrentED Register

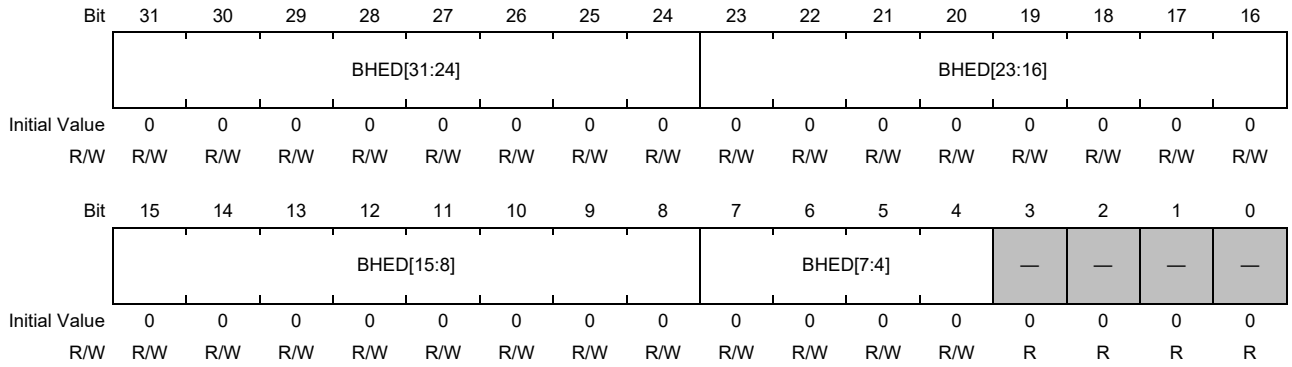
Abbreviated name of register: HcControlCurrentED



Bit	Bit Name	Initial Value	R/W	Description
31 to 4	CCED	All 0	R/W	Control Current ED This pointer indicates the physical address of the ED that is currently processed in the control list. After the current ED processing finishes, this pointer proceeds to the next ED. The host controller continues the list processing until the end of the frame. When the end of the control list is reached, the host controller checks the ControlListFilled bit of the HcCommandStatus register. If the corresponding bit is set to 1b, the contents of the HcControlHeadED field are copied to the HcControlCurrentED field, and the ControlListFilled bit is cleared. If the corresponding bit is set to 0b, nothing is performed. Update of this register is allowed only when the ControlListEnable bit of the HcControl register is cleared. If the ControlListEnable bit is set to 1b, the value of this register is only read. This register is initially set to H'0 to indicate the end of the control list.
3 to 0	—	All 0	R	Reserved The write value should always be 0.

(11) HcBulkHeadED Register

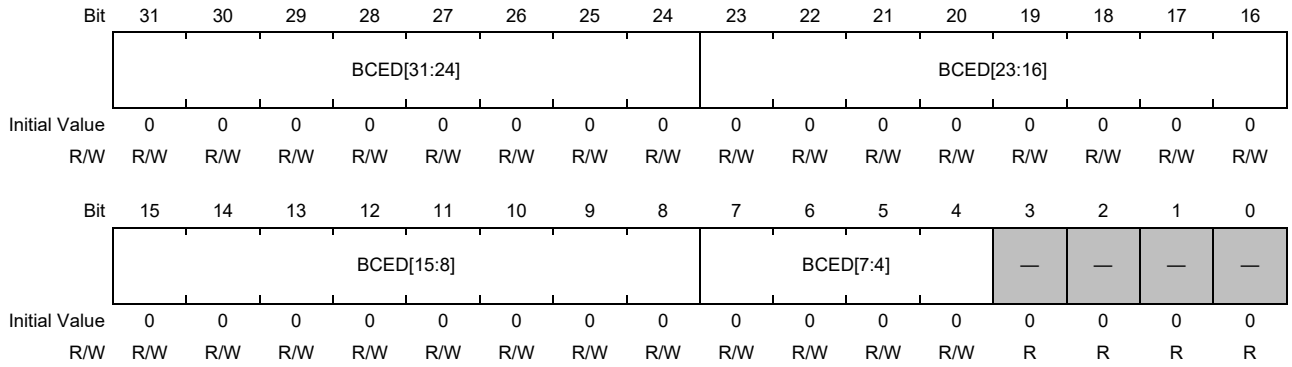
Abbreviated name of register: HcBulkHeadED



Bit	Bit Name	Initial Value	R/W	Description
31 to 4	BHED	All 0	R/W	BulkHead ED This field specifies the physical address of the head ED of the bulk list. This field must be set for bulk transfer before the BLE bit of the HcControl register is set. The host controller starts the processing of the control list from the HcBulkHeadED pointer.
3 to 0	—	All 0	R	Reserved The write value should always be 0.

(12) HcBulkCurrentED Register

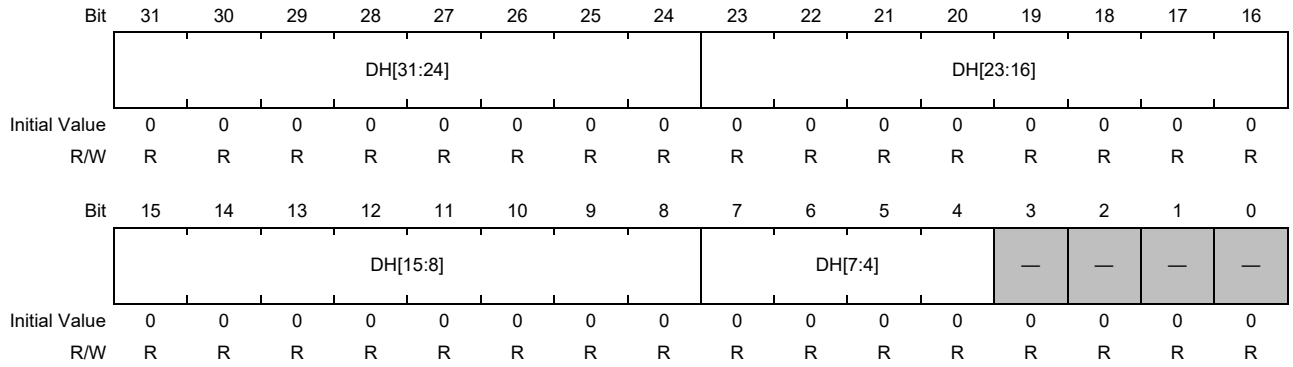
Abbreviated name of register: HcBulkCurrentED



Bit	Bit Name	Initial Value	R/W	Description
31 to 4	BCED	All 0	R/W	Bulk Current ED This pointer indicates the physical address of the ED that is currently processed in the bulk list. After the current ED processing finishes, this pointer proceeds to the next ED. The host controller continues the list processing until the end of the frame. When the end of the bulk list is reached, the host controller checks the ControlListFilled bit of the HcCommandStatus register. If the corresponding bit is set to 1b, the contents of the HcBulkHeadED field are copied to the HcBulklCurrentED field, and the ControlListFilled bit is cleared. If the corresponding bit is set to 0b, nothing is performed. Update of this register is allowed only when the ControlListEnable bit of the HcControl register is cleared. If the ControlListEnable bit is set to 1b, the value of this register is only read. This register is initially set to H'0 to indicate the end of the bulk list.
3 to 0	—	All 0	R	Reserved The write value should always be 0.

(13) HcDoneHead Register

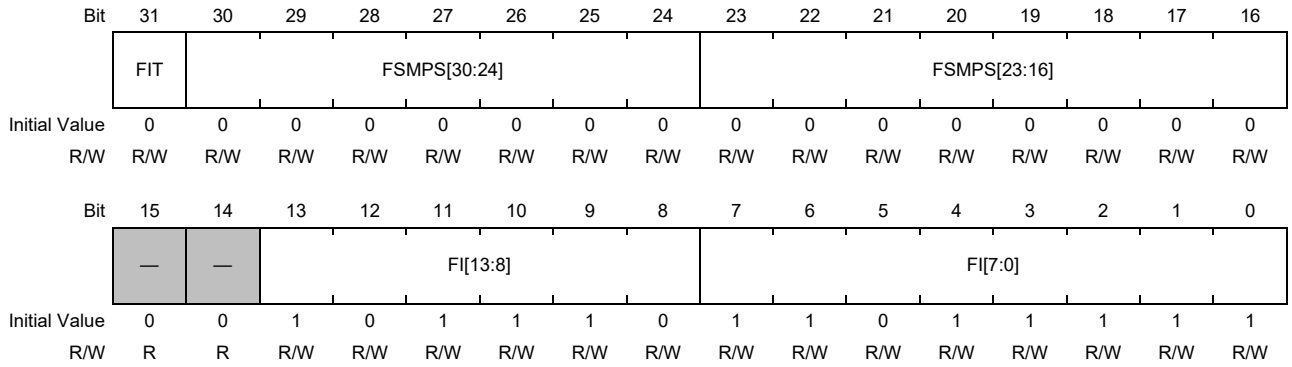
Abbreviated name of register: HcDoneHead



Bit	Bit Name	Initial Value	R/W	Description
31 to 4	DH	All 0	R	Done Head This field indicates the physical address of the HcDoneHead of the host controller. The physical address of the TD that lately finished and is added to the Done queue. After the TD processing finishes, the host controller writes the contents of the HcDoneHead to the NextTD field of the TD. At the same time, the host controller overwrites the contents of HcDoneHead with the TD address. After the host controller writes the contents of this register into HCCA, it sets 0b to this register. Then, the WritebackDoneHead bit of the HcInterruptStatus register is set to 1b.
3 to 0	—	All 0	R	Reserved The write value should always be 0.

(14) HcFmInterval Register

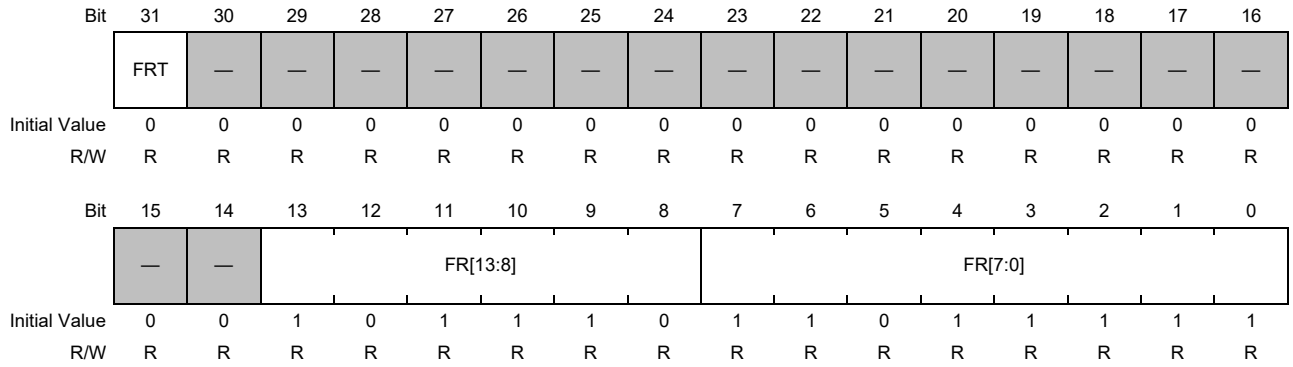
Register abbreviation: HcFmInterval



Bit	Bit Name	Initial Value	R/W	Description
31	FIT	0	R/W	Frame Interval Toggle This bit is used to synchronize the frame setting value with the hardware (host controller). When the FI field is updated, the toggle value is written to this bit. When the FI field value is applied to the FR field of the HcFmRemaining register, the value of this bit is also applied to the FRT bit of the HcFmRemaining register. By checking the toggle value in the FRT bit of the cFmRemaining register, the software can check whether the value in the FI field has been applied to the FR field of the HcFmRemaining register.
30 to 16	FSMPS	All 0	R/W	FS Largest Data Packet This field sets the maximum amount of data (bits) that the host controller can send and receive without the fear of schedule overrun. The host controller compares the current frame position and the setting value in this field to determine the length (of the frame) that is ready to be transferred. This value differs depending on the capacity of the system bus and other reasons, estimate the value and set it to this field. *Note*: The maximum setting value for this field is H'2778. Do not set any value that is larger than H'2778.
15, 14	—	All 0	R	Reserved The write value should always be 0.
13 to 0	FI	H'2EDF	R/W	Frame Interval This field is used to set the length of the frame (bit time) used for Full Speed. Set the value of this field to "H'2EDF" so that 1 frame (= 1 ms) of USB standard is satisfied.

(15) HcFmRemaining Register

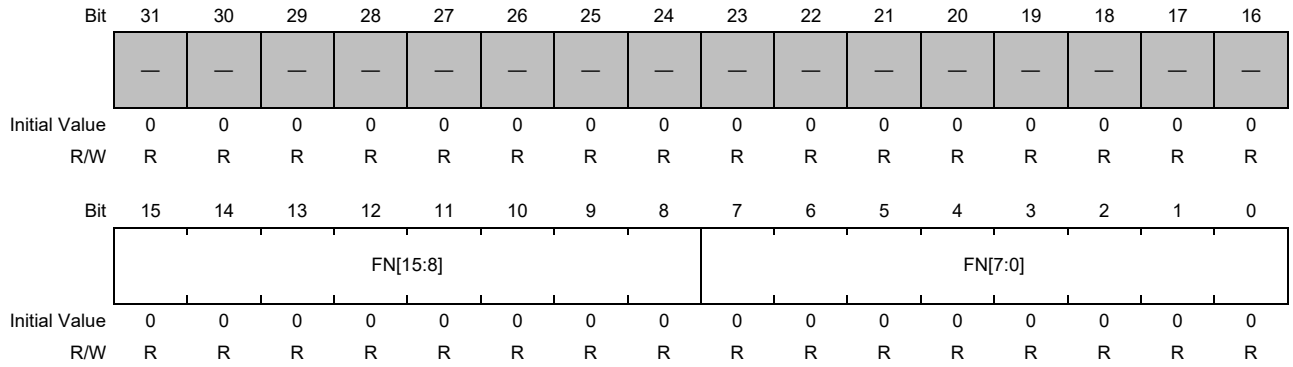
Register abbreviation: HcFmRemaining



Bit	Bit Name	Initial Value	R/W	Description
31	FRT	0	R	Frame Remaining Toggle This bit is used to synchronize the frame setting value with the hardware (host controller). When the FR (Frame Remaining) field is set to H'0000, the host controller copies the FI (Frame Interval) field value to the FR field, and copies the FIT (Frame Interval Toggle) bit value to this bit. This bit can be used to check that the FI field of the HcFmInterval register has been correctly copied to the FR field.
30 to 14	—	All 0	R	Reserved The write value should always be 0.
13 to 0	FR	H'2EDF	R	Frame Remaining This field indicates the current frame value for 14-bit down counter. The value in this field counts down as time passes. When the value becomes H'0000, the value of FI (Frame Interval) of the HcFmInterval register is loaded. When the state of the host controller is changed to the USB Operational state, the host controller reloads the value in the FI (Frame Interval) field of the HcFmInterval register, and the new value is used from the next SOF.

(16) HcFmNumber Register

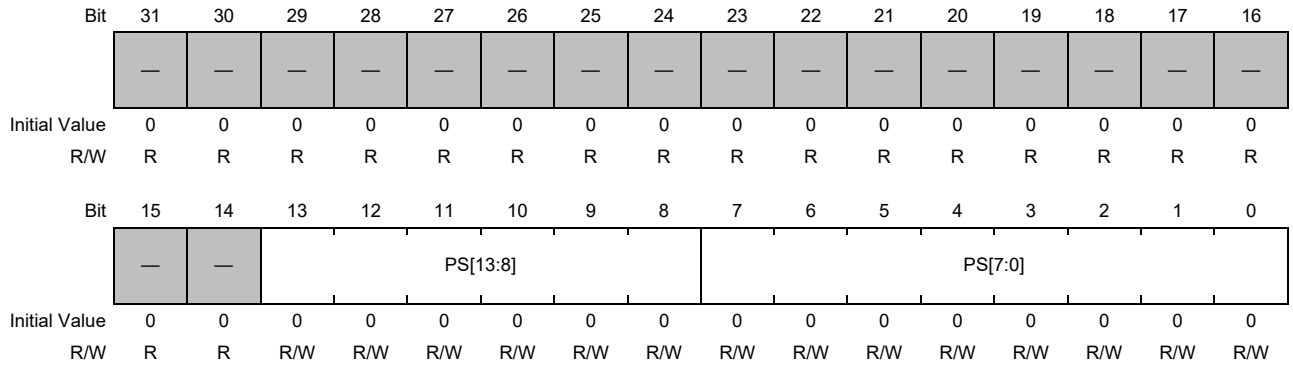
Register abbreviation: HcFmNumber



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved The write value should always be 0.
15 to 0	FN	All 0	R	Frame Number This field indicates the number of passed frames. When the HcFmRemaining register is reloaded, this field is incremented. If this field is reached to H'FFFF, the value is rolled over to H'0000. When the state of the host controller is changed to the USB Operational state, this field is automatically incremented. After the host controller increments the frame number at the frame boundary and sends SOF, the contents of this field are written to HCCA. This is performed before the host controller reads the first ED of the frame. After writing to HCCA, the host controller sets the SF bit of the HcInterruptStatus register.

(17) HcPeriodicStart Register

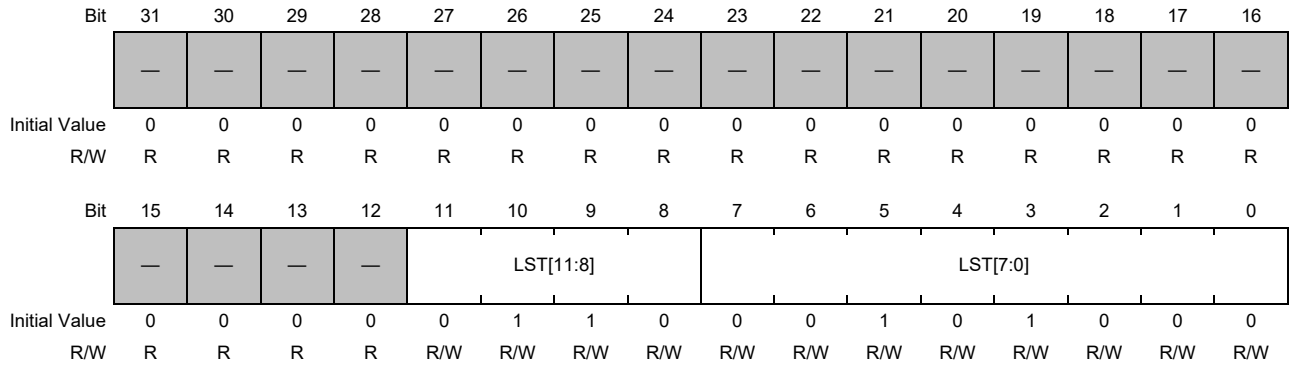
Register abbreviation: HcPeriodicStart



Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved The write value should always be 0.
13 to 0	PS	All 0	R/W	Periodic Start This field indicates the time the host controller starts the periodic list processing in the frame. Estimate an appropriate value, and set the value to this field at the initial setting of the host controller. OHCI standard recommends that you set this setting value to about 90% of the FI field value of the HcFmInterval register. The recommended value is H'2A2F. When the value in the FR field of the HcFmRemaining register reaches the value set to this field, the periodic list processing is given priority over the control/bulk list processing. Therefore, after the currently running control or bulk transfer finishes, the host controller starts the Interrupt list processing.

(18) HcLSThreshold Register

Register abbreviation: HcLSThreshold



Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved The write value should always be 0.
11 to 0	LST	H'628	R/W	LS Threshold This field indicates the threshold value of whether transfer is available for the remaining time of the LS transfer frame. If the value of the FR field of the FmRemaining register is larger than the value set to this field, the host controller can start LS transfer.

(19) HcRhDescriptorA Register

Register abbreviation: HcRhDescriptorA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	POTPGT								—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	NOCP	OCPM	DT	NPS	PSM	NDP							
Initial Value	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1
R/W	R	R	R	R/W	R/W	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	POTPGT	H'0F	R/W	PowerOn To PowerGood Time This field indicates the time for waiting until software can access the root hub port after the power is supplied to the port. The unit of time is 2 ms. Therefore, the wait time is POTPGT × 2 ms.
23 to 13	—	All 0	R	Reserved The write value should always be 0.
12	NOCP	0	R/W	No OverCurrent Protection This bit sets whether the overcurrent function of the root hub is supported. If this bit is set to 0b, the overcurrent state is reported depending on the setting of the OCPM bit. 0b: The overcurrent state is supported. 1b: The overcurrent state is not supported. If you do not require the overcurrent function, set this bit to 1b once the module is released from module standby. For details, see **Section 32A.9.1, Host/Peripheral Common Setting Sequence**.
11	OCPM	1	R/W	OverCurrent Protection Mode This bit sets how to report the overcurrent state of the root hub. If this bit is reset, this bit must indicate the same mode as the PSM (Power Switching Mode) bit. This bit is valid only when the NOCP (NoOverCurrent Protection) bit is cleared (0b). 0b: The overcurrent state is collectively reported for all ports. 1b: The overcurrent state is reported for each port.
10	DT	0	R	Device Type This bit indicates that the root hub is not a composite device. This bit is always 0b because the root hub is not allowed to be a composite device.
9	NPS	0	R/W	No Power Switching This bit sets how to control the port power. If this bit is set to 0b, the PSM bit is used to set whether the power control is collectively performed for all ports or is performed for each port. 0b: The port power can be switched between on and off. 1b: The power is always on while the host controller is running.
8	PSM	1	R/W	Power Switching Mode This bit sets how to control the power switch for each port of the root hub. This bit is valid only when the NPS bit is 0b. 0b: The power of all ports is collectively controlled. 1b: The power of ports is controlled for each port. If the PPCM (PortPower Control Mask) bit of the HcRhDescriptorB register is set, each port responds only to the Set/ClearPortPower command. If the PPCM bit is cleared, each port is controlled by the Set/ClearGlobalPower command.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	NDP	H'01	R	Number Downstream Ports This field indicates the number of downstream ports supported by the root hub.

(20) HcRhDescriptorB Register

Register abbreviation: HcRhDescriptorB

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PPCM	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DR	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved The write value should always be 0.
17	PPCM	1	R/W	PortPower Control Mask This bit sets the port power control command when the PSM (Power Switching Mode) bit of the HcRhDescriptorA register is set. If this bit is 0b, the Global Power Control command (Set/ClearGlobalPower) is used for control. If this bit is 1b, the Port Power Control command (Set/ClearPortPower) is used for control. If the PSM bit is 0b, this bit is ignored.
16 to 2	—	All 0	R	Reserved The write value should always be 0.
1	DR	0	R/W	Device Removable This bit indicates whether each port of the root hub is removable. If this bit is 0b, the connected device is removable. If this bit is 1b, the connected device is not removable.
0	—	0	R	Reserved The write value should always be 0.

(21) HcRhStatus Register

Register abbreviation: HcRhStatus

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRWE	—	—	—	—	—	—	—	—	—	—	—	—	—	OCIC	LPSC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W(1)	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRWE	—	—	—	—	—	—	—	—	—	—	—	—	—	OCI	LPS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	CRWE	0	W	Clear RemoteWakeup Enable This bit is used to clear the DRWE bit. If this bit is set to 1b, DRWE (Device RemoteWakeup Enable) bit can be cleared. Writing 0b to this bit has no effect.
30 to 18	—	All 0	R	Reserved The write value should always be 0.
17	OCIC	0	R/W (1)	OverCurrent Indicator Change This bit is used to report the change in the OCI bit. If there is any change in the OCI bit, the host controller sets this bit to 1b. If 1b is written to this bit while this bit is set to 1b, this bit can be cleared. Writing 0b to this bit has no effect. 0b: There is no change in the Overcurrent state. 1b: There is a change in the Overcurrent state.
16	LPSC	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Local Power Status Change This bit is always read as 0b because the Local Power Status is not supported. [Write] Set Global Power If this bit is set to 1b, the power to the ports is turned on. The ports whose power is turned on are determined by the settings of the PSM (Power Switching Mode) bit and the PPCM (Port Power Control Mask) bit of the HcRhDescriptorA register.

written to this bit	PSM	PPCM[N]	Description
0	—	—	Setting ignored
1	0	—	1b is set to the PPS bit.
	1	0	1b is set to the PPS bit.
		1	Setting ignored

Writing 0b to this bit has no effect.

15	DRWE	0	R/W	[Read] Device RemoteWakeup Enable This bit sets whether the RemoteWakeup event includes the CSC (Connect Status Change) bit. If this bit is set to 1b, the CSC bit of the HcRhPortStatus register becomes valid as the Resume event. If the CSC bit is changed to 1b, the state is changed from USB Suspend to USB Resume, and the Resume detection interrupt occurs. 0b: Connect Status Change is not the source of RemoteWakeup. 1b: Connect Status Change is the source of RemoteWakeup. [Write] Set RemoteWakeup Enable This bit is used to set the DRWE bit. Writing 0b to this bit has no effect.

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>14 to 2</td><td>—</td><td>All 0</td><td>R</td><td>Reserved
The write value should always be 0.</td></tr>
<tr><td>1</td><td>OCI</td><td>0</td><td>R</td><td>Over Current Indicator
This bit is used to report the overcurrent state in the global overcurrent detection mode (OCPM bit = 0b).
This bit always indicates 0b when overcurrent for each port is reported (when OPCM bit = 1b).
0b: The port state is normal.
1b: The port is in the overcurrent state.</td></tr>
<tr><td>0</td><td>LPS</td><td>0</td><td>R/W</td><td>The meaning of this bit differs depending on whether the operation is read or write.
[Read] Local Power Status
This bit is always read as 0b because the Local Power Status is not supported.
[Write] Clear Global Power
If this bit is set to 1b, the power to the ports is turned off.
The ports whose power is turned off are determined by the settings of the PSM (Power Switching Mode) bit and PPCM (Port Power Control Mask) bit of the HcRhDescriptorA register.
<table>
<tr><th>written to this bit</th><th>PSM</th><th>PPCM[N]</th><th>Description</th></tr>
<tr><td>0</td><td>—</td><td>—</td><td>Setting ignored.</td></tr>
<tr><td>1</td><td>0</td><td>—</td><td>The PPS bit is cleared to 0b.</td></tr>
<tr><td></td><td>1</td><td>0</td><td>The PPS bit is cleared to 0b.</td></tr>
<tr><td></td><td></td><td>1</td><td>Setting ignored.</td></tr>
</table>
Writing 0b to this bit has no effect.</td></tr>
</table>

(22) HcRhPortStatus[1:NDP] Register

Register abbreviation: HcRhPortStatus1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	PRSC	OCIC	PSSC	PESC	CSC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	LSDA	PPS	—	—	—	PRS	POCI	PSS	PES	CCS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved The write value should always be 0.
20	PRSC	0	R/W (1)	Port Reset Status Change This bit indicates that port reset (bus reset) has finished. The host controller sets this bit when 10 ms port reset (bus reset) finishes. 0b: Port reset has not finished, or the PRS (Port Reset Status) bit is not changed. 1b: Port reset has finished.
19	OCIC	0	R/W (1)	Port OverCurrent Indicator Change This bit is set when the overcurrent state of the port is detected. This bit is set when the host controller changed the POCI bit value. This bit is valid only under the setting that the overcurrent state is reported for each port (OCPM bit = 1b). If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. 0b: The overcurrent state has not changed. 1b: The overcurrent state has changed (POCI bit changed).
18	PSSC	0	R/W (1)	Port Suspend Status Change This bit indicates that the Resume sequence finished. This sequence includes 20 ms of the Resume signal, LS EOP, and 3 ms of resynchronization delay. If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. If the PRSC (Port Reset Status Change) bit is set, this bit is cleared. 0b: The Resume sequence has not finished. 1b: The Resume sequence has finished.
17	PESC	0	R/W (1)	Port Enable Status Change This bit indicates that the PES (Port Enable Status) bit was changed. If a hardware event clears the PES bit, this bit is set to 1b. If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. 0b: The PES (Port Enable Status) bit has not changed. 1b: The PES (Port Enable Status) bit has changed.

Bit	Bit Name	Initial Value	R/W	Description
16	CSC	0	R/W (1)	Connect Status Change This bit indicates that the CCS (Current Connect Status) bit was changed. If the Connect/Disconnect event occurs, this bit is set to 1b. If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. If a request (Port Reset/Port Enable/Port Suspend) is received during the Disconnect status, this bit is set for reevaluation of device connection confirmation. 0b: The CCS (Current Connect Status) bit has not changed. 1b: The CCS (Current Connect Status) bit has changed.
15 to 10	—	All 0	R	Reserved The write value should always be 0.
9	LSDA	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Low Speed Device Attached This bit indicates the speed of the device connected to the port. This bit is valid only when the CCS (Current Connect Status) bit is set. 0b: Full Speed device is connected. 1b: Low Speed device is connected. [Write] Clear Port Power This bit is used to turn off the power of the port when the port is power controlled. Writing 1b to this bit turns off the port power. Writing 0b to this bit has no effect.
8	PPS	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Power Status This bit indicates the power status of the port. This bit is cleared when the overcurrent is detected. 0b: Port power is off. 1b: Port power is on. [Write] Set Port Power This bit is used to turn on the power of the port when the port is power controlled. Writing 1b to this bit turns on the port power. Writing 0b has no effect.
7 to 5	—	All 0	R	Reserved The write value should always be 0.
4	PRS	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Reset Status This bit indicates the port reset (bus reset) status. When 10 ms of port reset finishes, the PRSC (Port Reset Status Change) bit is set and this bit is cleared. When the CCS bit is cleared (when no device is connected), this bit cannot be set. 0b: Not during port reset 1b: During port reset [Write] Set Port Reset This bit is used to issue a port reset (bus reset) to the downstream port. Writing 1b to this bit starts 10 ms of port reset. Writing 0b to this bit has no effect. If CCS is cleared, this bit cannot be set. Instead, CSC is set to 1b. This is performed to report to ports to which no device is connected, that port reset was performed.

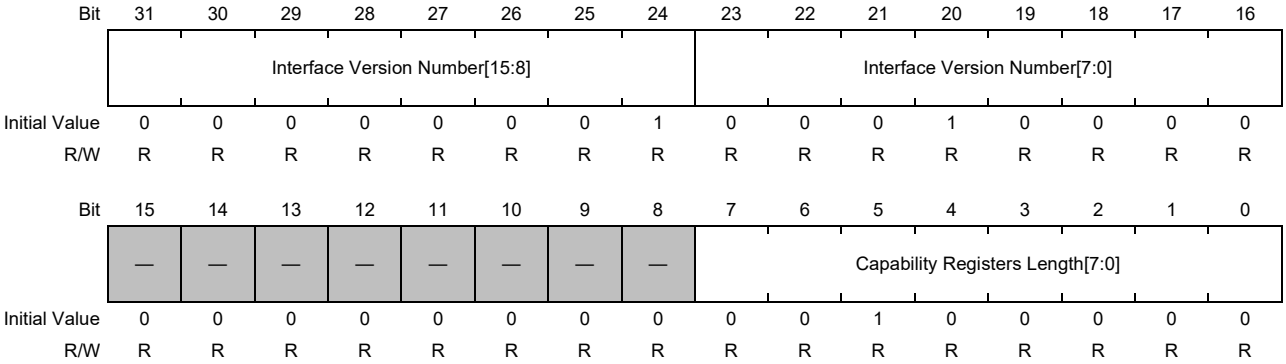
Bit	Bit Name	Initial Value	R/W	Description
3	POCI	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port OverCurrent Indicator This bit indicates that the downstream port is in the overcurrent status. This bit is valid only when the setting is specified that the overcurrent status is reported for each port (OCPM bit = 1b). On the other hand, if the setting is specified that the overcurrent status is reported for all ports, this bit is always read as 0b. 0b: The port is in normal status. 1b: The port is in overcurrent status. [Write] Clear Suspend Status This bit is used to finish the Suspend status and start the Resume sequence. Writing 1b to this bit starts the Resume sequence. Writing 0b has no effect. The Resume sequence starts only when PSS (Port Suspend Status) is set.
2	PSS	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Suspend Status This bit indicates that the port status is in Suspend or Resume sequence. 0b: The port is not in Suspend status. 1b: The port is in Suspend status. When the CCS bit is cleared (when no device is connected), this bit cannot be set. This bit is cleared under the following conditions: • When the Resume sequence finished, and the PSSC (Port Suspend Status Change) bit is set • When the port reset finished, and the PRSC (Port Reset Status Change) bit is set • When the host controller is in the USB Resume state [Write] Set Port Suspend This bit is used to change the port status to Suspend. Writing 1b to this bit changes the status to Suspend. Writing 0b to this bit has no effect. If CCS is cleared, this bit cannot be set. Instead, CSC is set to 1b. This is performed to report to the ports to which no device is connected that the Suspend command was issued.
1	PES	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Enable Status This bit indicates whether the port status is Enable or Disable. If the host controller detects a bus error (for example, overcurrent state, disconnect, port power off, and babble error), it clears this bit. Then, the PESC (PortEnableStatusChange) bit is set. When the CCS bit is cleared (when no device is connected), this bit cannot be set. This bit is set “when port reset finished and the PRSC bit is set” or “when the port status becomes Suspend and the PRSC bit is set”. 0b: The port status is Disable. 1b: The port status is Enable. [Write] Set Port Enable This bit is used to set the PES bit. Writing 1b to this bit changes the port status to Enable. Writing 0b to this bit has no effect. If CCS is cleared, this bit cannot be set. Instead, CSC is set to 1b. This is performed to report that the status of the ports to which no device is connected was tried to be changed to Enable.

Bit	Bit Name	Initial Value	R/W	Description
0	CCS	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Current Connect Status The current connection status of the downstream port is applied to this bit. 0b: No device is connected. 1b: A device is connected. [Write] Clear Port Enable This bit is used to clear the PES (Port Enable Status) bit. Writing 1b to this bit changes the port status to Disable. Writing 0b to this bit has no effect.

32A.2.4.2 EHCI Controller Capability Register

(1) HCIVERSION/CAPLENGTH Register

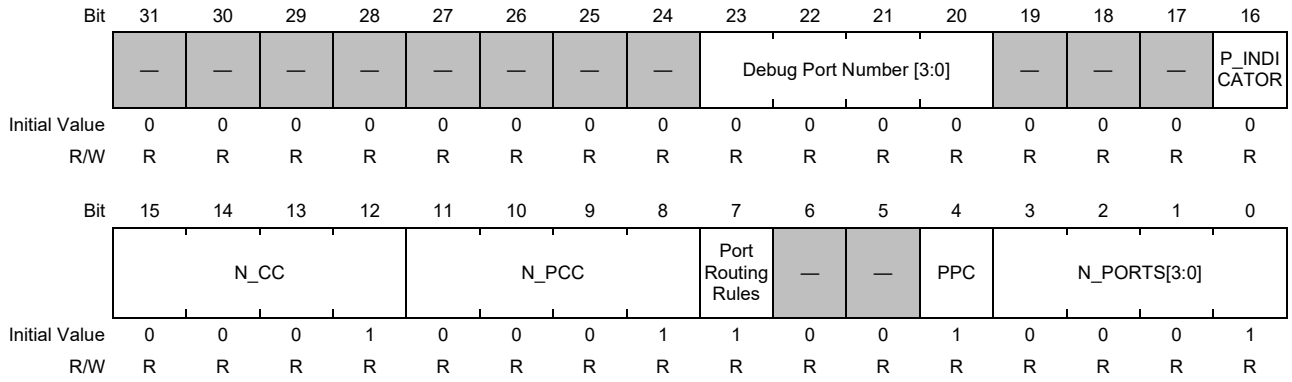
Register abbreviation: CAPL_VERSION



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	Interface Version Number	H'0110	R	This field indicates the EHCI version supported by the host controller. H'0110 is indicated because this host controller supports EHCI Rev1.1.
15 to 8	—	All 0	R	Reserved The write value should always be 0.
7 to 0	Capability Registers Length	H'20	R	This field is used as an offset that is added to the base address to find the start address of the EHCI Operational register. H'20 is indicated because the EHCI Operation register of this module starts from H'20.

(2) HCSPARAMS Register

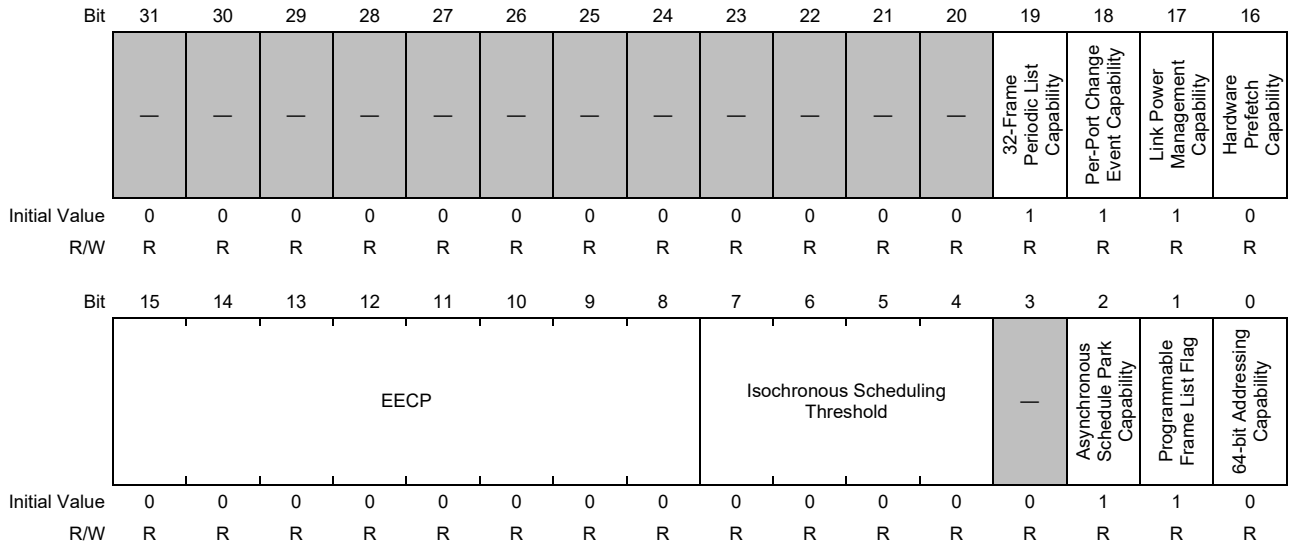
Register abbreviation: HCSPARAMS



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved The write value should always be 0.
23 to 20	Debug Port Number	All 0	R	This field indicates that the host controller port is a debug port. 0000b is indicated because this module does not have a debug port.
19 to 17	—	All 0	R	Reserved The write value should always be 0.
16	P_INDICATOR	0	R	Port Indicators This bit indicates whether the host controller supports port indicator control. 0b is indicated because this module does not support port indicator control.
15 to 12	N_CC	H'1	R	Number of Companion Controller This field indicates the number of OHCI host controllers implemented in this module. H'1 is indicated because this module has one OHCI host controller.
11 to 8	N_PCC	H'1	R	Number of Ports per Companion Controller This field indicates the number of ports supported by an OHCI host controller. The setting value of the Port_no field of the PCI Configuration EXT1 register is applied to this field.
7	Port Routing Rules	1	R	This bit indicates how individual ports are mapped to the OHCI host controller. This bit indicates 1b because, in this module, the contents of the HCSP_PORTROUTE register show the mapping method.
6, 5	—	All 0	R	Reserved The write value should always be 0.
4	PPC	1	R	Port Power Control This bit indicates how the port power of this module is controlled. 0b: The port power is always on. 1b: The PP bit of the PORTSC register controls the port power.
3 to 0	N_PORTS	H'1	R	This field indicates the number of physical downstream ports used by this module. 0001b: 1 Port

(3) HCCPARAMS Register

Register abbreviation: HCCPARAMS



Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved The write value should always be 0.
19	32-Frame Periodic List Capability	1	R	This bit indicates 1b because this module supports 32-Frame Periodic List defined in EHCI V1.1. This means that the Frame List Size field of the USBCMD register is set to 11b, so this module supports 32-Frame Periodic List.
18	Per-Port Change Event Capability	1	R	This bit indicates 1b because this module supports event detection function for ports that are defined in EHCI V1.1. This means that this module supports event detection function for ports, and is associated with the Pre-Port Change Event Enable field of the USBCMD register, Port-1 Change Detect field of the USBSTS register, and Port-1 Change Interrupt Enable field of the USBINT register. If this bit is 0b, the corresponding fields of the above registers are treated as Reserved.
17	Link Power Management Capability	1	R	This bit indicates 1b because this module supports LPM (Link Power Management) defined in EHCI V1.1. This means that this module supports LPM L1 state, and is controlled by the Suspend using L1 bit, Suspend Status bit, and Device Address field of the PORTSC register. If this bit is 0b, the corresponding bits and field of the above PORTC register are treated as Reserved.
16	Hardware Prefetch Capability	0	R	This bit indicates 0b because this module does not support the hardware prefetch function defined in EHCI V1.1.
15 to 8	EECP	All 0	R	This field indicates the offset address of the EHCI Extend Capabilities Registers. This field indicates H'00 because this module does not use EHCI Extend Capabilities Registers.
7 to 4	Isochronous Scheduling Threshold	All 0	R	This field indicates H'0 because this module does not support caches with isochronous data structure for the entire frame.
3	—	0	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	Asynchronous Schedule Park Capability	1	R	This bit indicates whether the Park mode is supported for High Speed QH (Queue Head) in an asynchronous schedule. This bit indicates 1b because this module supports the above function.
1	Programmable Frame List Flag	1	R	This bit indicates the setting for the available frame list size. This bit indicates 1b for this module. If this bit is set to 1b, bit [3:2] (Frame List Size) of the USBCMD register can be used to set the available frame list size, and the frame list size smaller than 4 Kbyte is configurable.
0	64-bit addressing Capability	0	R	This bit indicates which type of memory pointer the data structure uses (32 bit address memory pointer or 64 bit address memory pointer). This bit indicates 0b for this module because this module has the data structure that uses 32 bit address memory pointer. 64 bit address is not supported.

(4) HCSP-PORTROUTE Register

Register abbreviation: HCSP_PORTROUTE

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Companion Port Route[31:24]								Companion Port Route[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

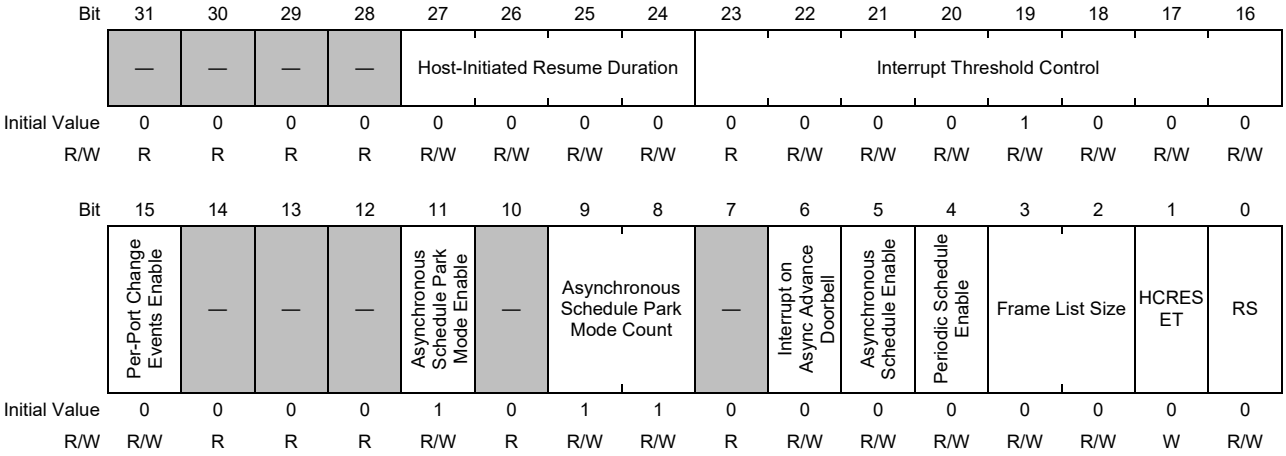
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Companion Port Route[15:8]								Companion Port Route[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Companion Port Route[31:0]	All 0	R	This field indicates the ports controlled by the OHCI host controller. This field indicates 0b for this module because this module has one OHCI host controller.

32A.2.4.3 HCI Operational Register

(1) USBCMD Register

Register abbreviation: USBCMD

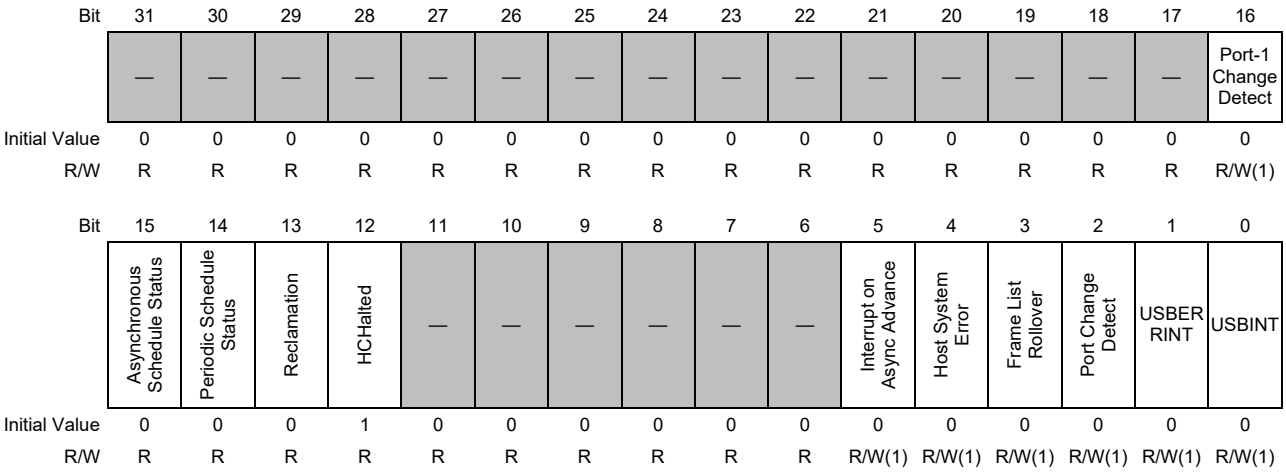


Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved The write value should always be 0.
27 to 24	Host-Initiated Resume Duration	All 0	R/W	This field indicates the minimum time of the K-state drive while the host controller resumes from the LPM (L1) state. The value in this field is sent to the connected device that has the LPM function via the HIDR field in the bmAttributes field of the LPM token. The encoded value of this field is defined as the name of the HIRD field in the LPM Token. Specifically, H'0 means 50 µs, and if the value is incremented by 1, 75 µs is incremented. For example, H'1 means 125 µs, and H'4 means 1175 µs.
23 to 16	Interrupt Threshold Control	H'08	R, R/W	This field indicates the maximum rate until the host controller issues an interrupt. Note that the setting values other than the following values are not guaranteed. Value / Maximum Interrupt Interval H'00 Reserved H'01 1 micro-frame H'02 2 micro-frames H'04 4 micro-frames H'08 8 micro-frames (default, equals to 1 ms) H'10 16 micro-frames (2 ms) H'20 32 micro-frames (4 ms) H'40 64 micro-frames (8 ms) Do not set H'00 to this field while the Halted bit is 0b.
15	Per-Port Change Events Enable	0	R/W	This field is used to enable the event report function of the ports defined by the Port-1 Change Detect field of the USBSTS register and the Port-1 Change Detect Enable field of the USBINTR register. 0b: The event report function of the ports is disabled. 1b: The event report function of the ports is enabled.
14 to 12	—	All 0	R	Reserved The write value should always be 0.
11	Asynchronous Schedule Park Mode Enable	1	R/W	This bit enables or disables the Asynchronous Schedule Park mode. 0b: The Park mode is disabled. 1b: The Park mode is enabled.

Bit	Bit Name	Initial Value	R/W	Description
10	—	0	R	Reserved The write value should always be 0.
9, 8	Asynchronous Schedule Park Mode Count	11b	R/W	This field sets the number of transactions that the host controller can serially execute for one QH (Queue Head) fetch in an asynchronous schedule. The valid value range is H'1 to H'3. This field is valid when bit 11 (Asynchronous Schedule Park Mode Enable) is 1b. Do not set H'0 to this field.
7	—	0	R	Reserved The write value should always be 0.
6	Interrupt on Async Advance Doorbell	0	R/W	This bit is used as the doorbell. In an asynchronous schedule processing, if you want an interrupt to occur before proceeding to the next QH (Queue), set this bit to 1b. After a QH processing normally finishes, the host controller clears this bit to 0b, and sets bit 5 (Interrupt on Async Advance bit) of the USBSTS register to 1b. If bit 5 (Interrupt on Async Advance Enable bit) of the UBINTR register is set to 1b, an interrupt occurs at the next interrupt timing. If the asynchronous schedule is disabled, do not write 1b to this bit.
5	Asynchronous Schedule Enable	0	R/W	This bit sets whether the host controller proceeds to the asynchronous list processing or skip the processing. 0b: The asynchronous list processing is skipped. 1b: Use the ASYNCLISTADDR register to proceed the asynchronous list processing.
4	Periodic Schedule Enable	0	R/W	This bit sets whether the host controller proceeds or skips the periodic list processing. 0b: The periodic list processing is skipped. 1b: Use the PERIODICLISTBASE register to proceed the periodic list processing.
3, 2	Frame List Size	0	R/W	This field specifies the frame list size. The setting value of this field determines the size of the Frame List Current index of the FRINDEX register. Value: 00b — the number of frames in Frame List: 1024 frames (default) 01b: 512 frames 10b: 256 frames 11b: 32 frames
1	HCRESET	0	W	Host Controller Reset This bit is used to initialize the EHCI circuit of the host controller. If this bit is set to 1b, the host controller initializes the internal pipelines, counters, and state machines, and communications on the USB immediately stop. At this time, port reset is not issued to the downstream ports. Reset by this bit has no effect on the registers other than the EHCI Operational register. The EHCI Operational register is initialized, and the port owner returns to OHCI. Software must be reset to return the host controller to the operational state. When this reset processing finishes, the host controller sets this bit to 0b. Writing 0b to this bit cannot stop the reset processing. If the HCHalted bit of the USBSTS register is set to 0b, do not set 1b to this bit.
0	RS	0	R/W	Run/Stop This bit is used to run or stop the EHCI host controller. If this bit is set to 1b, the host controller starts operation. As long as this bit is set to 1b, the host controller continues running. If this bit is set to 0b, the host controller finishes the currently executing transaction and some other transactions, and then is changed to the Halt status. The HCHalted bit of the USBSTS register indicates that the host controller finished the transaction processing and entered into the stop status. If the host controller is in a status other than Halt (the HCHalted bit of the USBSTS register is 1b), do not write 1b to this bit.

(2) USBSTS Register

Register abbreviation: USBSTS



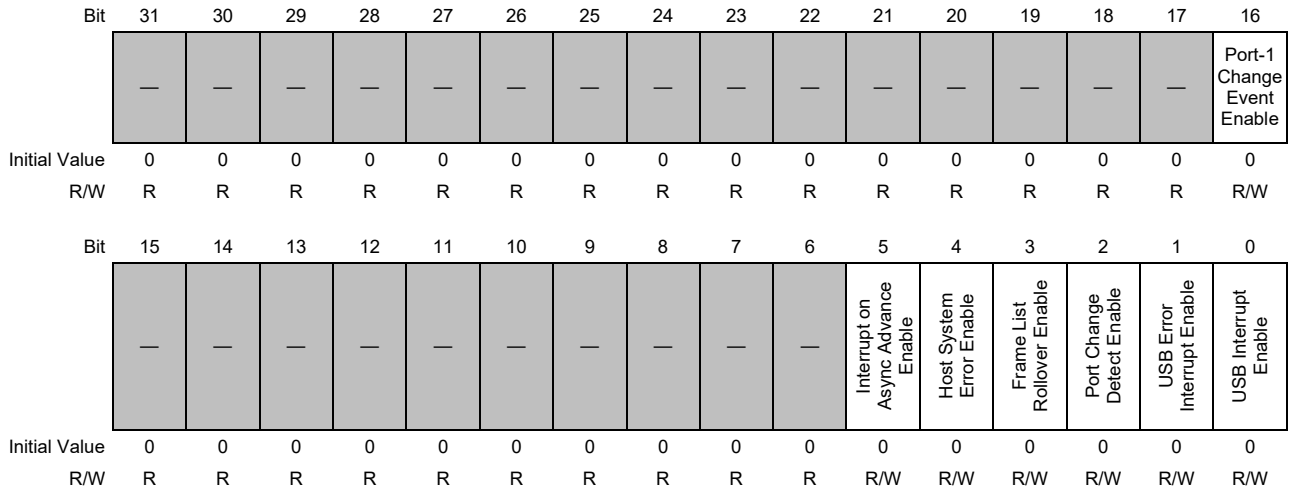
Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved The write value should always be 0.
16	Port-1Change Detect	0	R/W (1)	If this bit is set to 1b, it indicates that a change in the port status was detected. This bit is used only when the Per-Port Change Events Enable bit of the USBCMD register is set to 1b.
15	Asynchronous Schedule Status	0	R	This bit indicates the current status of the asynchronous schedule. 0b: The asynchronous schedule is disabled. 1b: The asynchronous schedule is enabled. If this bit and bit 5 (Asynchronous Schedule Enable) of the USBCMD register have the same value, the asynchronous schedule is enabled (1b) or disabled (0b).
14	Periodic Schedule Status	0	R	This bit indicates the current status of the periodic schedule. 0b: The periodic schedule is disabled. 1b: The periodic schedule is enabled. If this bit and bit 4 (Periodic Schedule Enable) of the USBCMD register have the same value, the periodic schedule is enabled (1b) or disabled (0b).
13	Reclamation	0	R	This bit is used to detect an empty asynchronous schedule. If this bit is 1b, the asynchronous schedule is empty. After reset or when a QH (H = 1) is fetched, the host controller clears this bit to 0b. Also, when the host controller executes an asynchronous transaction or detects a start event, it sets this bit to 1b. If this bit is 0b and a QH (H = 1) is fetched, the host controller is entered into the Async Sched Sleeping mode.
12	HCHalted	1	R	If the Run/Stop bit of the USBCMD register is 1b, this bit indicates 0b. If the software or host controller sets the Run/Stop bit to 0b, the host controller stops operation, and sets 1b to this bit. 0b: The EHCI host controller is running. 1b: The EHCI host controller is stopped.
11 to 6	—	All 0	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5	Interrupt on Async Advance	0	R/W (1)	This bit indicates the Async Advance Interrupt status. After the host controller fetches the QH, it checks bit 6 (Interrupt on Async Advance Doorbell [IAAD] bit) of the USBCMD Register. If the IAAD bit is set to 1b, the host controller clears the IAAD bit after the QH processing normally finishes, and sets this bit. If bit 5 (Interrupt on Async Advance Enable bit) of the UBINTR register is set to 1b, an interrupt due to this source will occur at the next interrupt timing after 1b is set to this bit. If HCD writes 1b to this bit, this bit can be cleared. Writing 0b to this bit has no effect. 0b: Async Advance Interrupt not occurred. 1b: Async Advance Interrupt status is detected.
4	Host System Error	0	R/W (1)	This bit is set to 1b if a serious error occurs in the host controller. If this error occurs, the host controller clears the Run/Stop bit of the USBCMD register to 0b so that the subsequent schedules are not executed. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: No system error occurred. 1b: A system error occurred.
3	Frame List Rollover	0	R/W (1)	If a frame list rollover occurs, the host controller sets this bit to 1b. Specifically, when the Frame Index field of the FRINDEX register is returned to H'000 from the maximum value (rollover), the host controller sets this bit to 1b. The maximum value (the value at which rollover occurs) depends on the Frame List Size field of the USBCMD register. For example, if the Frame List Size is 1024 frame, rollover occurs every time FRIMDEX [13] toggles. If the Frame List Size is 512 frame, rollover occurs every time FRINDEX [12] toggles. If the Frame List Size is 256 frame, rollover occurs every time FRINDEX [11] toggles. Writing 1b to this bit can clears this bit. Writing 0b to this bit has no effect. 0b: The frame list is not returned to H'000. 1b: The frame list is returned to H'000 (rollover occurred).
2	Port Change Detect	0	R/W (1)	This bit indicates that the port status has changed. Among the ports for which the Port Owner bit of the PORTSC1 register is set to 0b, if any port satisfies one of the following conditions, the host controller sets this bit to 1b: • Connect or Disconnect status of a device is detected, and the Connect Status Change bit of the PORTSC1 register is changed from 0 to 1. • A change of the Enable status of the port is detected, and the Port Enable/Disable Change bit of the PORTSC1 register is changed from 0 to 1. • The overcurrent state is detected, and the Over-current Change bit of the PORTSC1 register is changed from 0 to 1. • J-K transition is detected on a port in the Suspend status, and the Force Port Resume bit of the PORTSC1 register is changed from 0 to 1. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect.
1	USBERRINT	0	R/W (1)	USB Error Interrupt This bit indicates that a USB transaction finished with an error. When a USB transaction finishes with an error, the host controller sets this bit to 1b. If 1b is set to the IOC bit of qTD at which an error interrupt occurred, 1b is set to both of this bit and USBINT bit. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: USB transaction is normal. 1b: USB transaction finished with an error.

Bit	Bit Name	Initial Value	R/W	Description
0	USBINT	0	R/W (1)	USB Interrupt This bit indicates that a USB transfer has finished. The host controller sets this bit to 1b if one of the following conditions is satisfied: • A USB transfer has finished. • A short packet is received. Even if a USB transfer finished with an error, if IOC (Interrupt On Complete) of the TD is set to 1b, this bit is set to 1b. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: A USB transfer has not finished. 1b: A USB transfer has finished.

(3) USBINTR Register

Register abbreviation: USBINTR

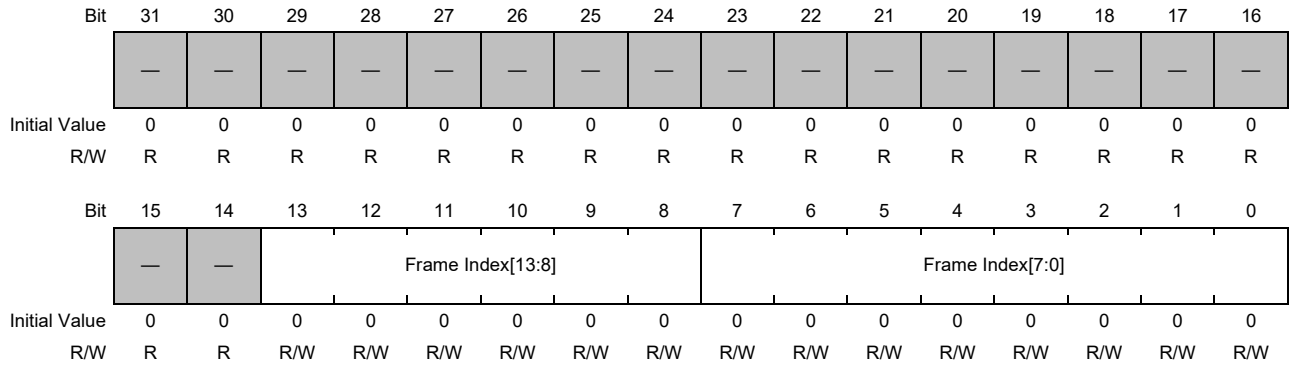


Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved The write value should always be 0.
16	Port-1 Change EventEnable	0	R/W	If this bit is 1b and the Port Change Detect bit of the USBSTS register is set to 1b, the host controller issues an interrupt.
15 to 6	—	All 0	R	Reserved The write value should always be 0.
5	Interrupt on Async Advance Enable	0	R/W	This bit sets whether bit 5 (Interrupt on Async Advance [IAA] bit) of the USBSTS register is enabled or disabled. If the IAA bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the IAA bit).
4	Host System Error Enable	0	R/W	This bit sets whether bit 4 (Host System Error [HSE] bit) of the USBSTS register is enabled or disabled. If the HSE bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the HSE bit).
3	Frame List Rollover Enable	0	R/W	This bit sets whether the bit 3 (Frame List Rollover [FLR] bit) of the USBSTS register is enabled or disabled. If the FLR bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the FLR bit).
2	Port Change Detect Enable	0	R/W	This bit sets whether bit 2 (Port Change Detect [PCD] bit) of the USBSTS register is enabled or disabled. If the USBERRINT bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the PCD bit).
1	USB Error Interrupt Enable	0	R/W	This bit sets whether bit 1 (USBERRINT bit) of the USBSTS register is enabled or disabled. If the USBERRINT bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the USBERRINT bit).

Bit	Bit Name	Initial Value	R/W	Description
0	USB Interrupt Enable	0	R/W	This bit sets whether bit 0 (IUSBINT bit) of the USBSTS register is enabled or disabled. If the USBINT bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the USBINT bit).

(4) FRINDEX Register

Register abbreviation: FRINDEX



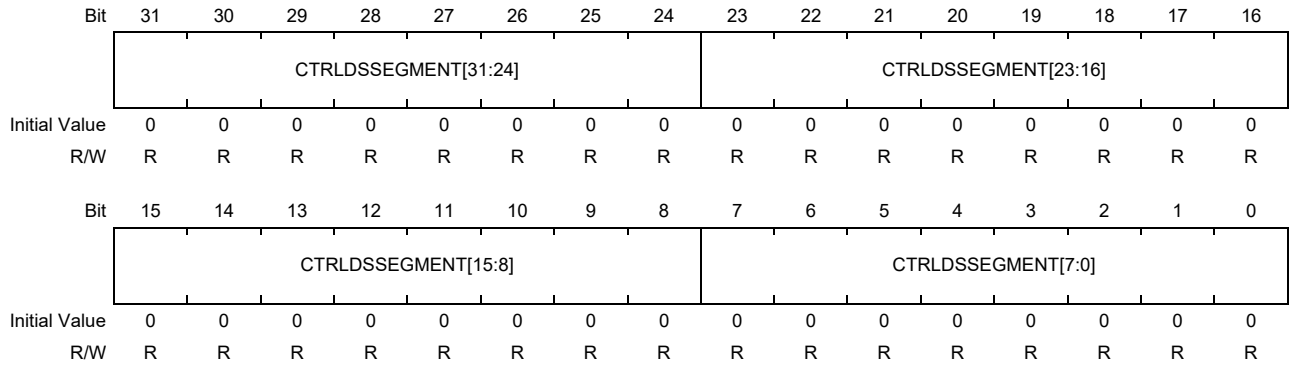
Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved The write value should always be 0.
13 to 0	Frame Index[13:0]	All 0	R/W	This field is used by the host controller to add an index to the periodic frame list. The value in this field is incremented at the end of a micro frame. Bit [N:3] of this field is used as the Frame List Current index. This means that, before the next index arrives, the current frame list is accessed 8 times. The value for N is determined, as follows, by the setting value of bit [3:2] (Frame List Size field) of the USBCMD register.

Frame List Size	Number of Frames	N
00b	1024	12
01b	512	11
10b	256	10
11b	32	12

Access this register only when the host controller is in stop status (bit 12 [HCHalted] = 1b). The setting value of this field is applied to the SOF frame number of the SOF token.

(5) CTRLDSSEGMENT Register

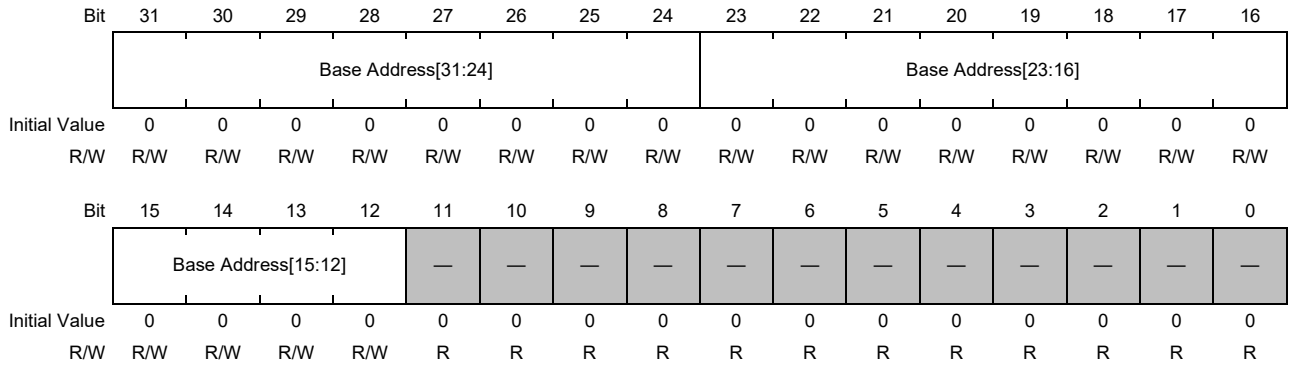
Register abbreviation: CTRLDSSEGMENT



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CTRLDSSEGMENT	All 0	R	This register is not used because this module does not support 64-bit address method. Therefore, HCD must not access this register.

(6) PERIODICLISTBASE Register

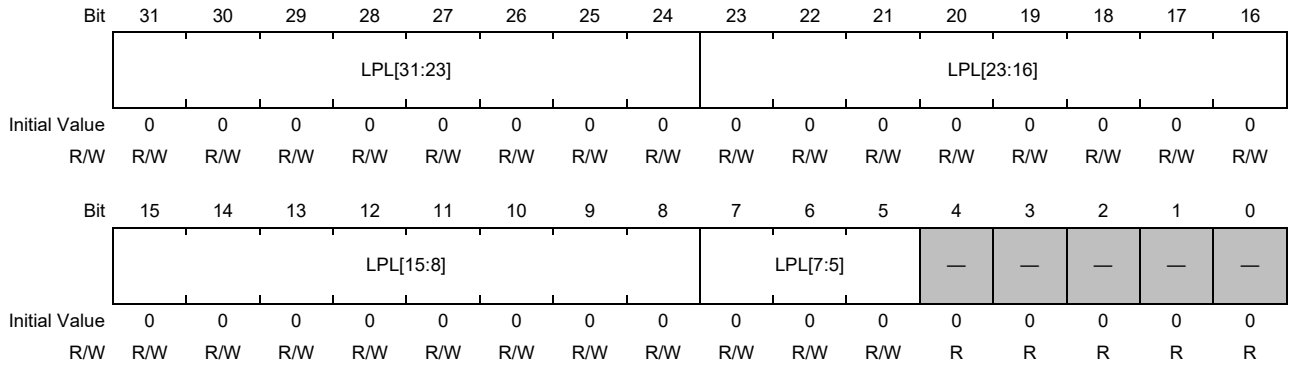
Register abbreviation: PERIODICLISTBASE



Bit	Bit Name	Initial Value	R/W	Description
31 to 12	Base Address	All 0	R/W	This field indicates the head address of the periodic frame list on the system memory. The host controller loads the contents of this register before starting the list processing. The host controller determines the frame list to be processed by using this field and Frame Index of the FRINDEX register. Align the address of the periodic frame list by 4 Kbyte. Normal operation is not guaranteed if any of these bits are changed during operation.
11 to 0	—	All 0	R	Reserved The write value should always be 0.

(7) ASYNCLISTADDR Register

Register abbreviation: ASYNCLISTADDR



Bit	Bit Name	Initial Value	R/W	Description
31 to 5	LPL	All 0	R/W	Link Pointer Low This field indicates the address (on the system memory) of the Asynchronous Queue Head to be processed next time. Align the address of Asynchronous Queue Head by 32 byte.
4 to 0	—	All 0	R	Reserved The write value should always be 0.

(8) CONFIGFLAG Register

Register abbreviation: CONFIGFLAG

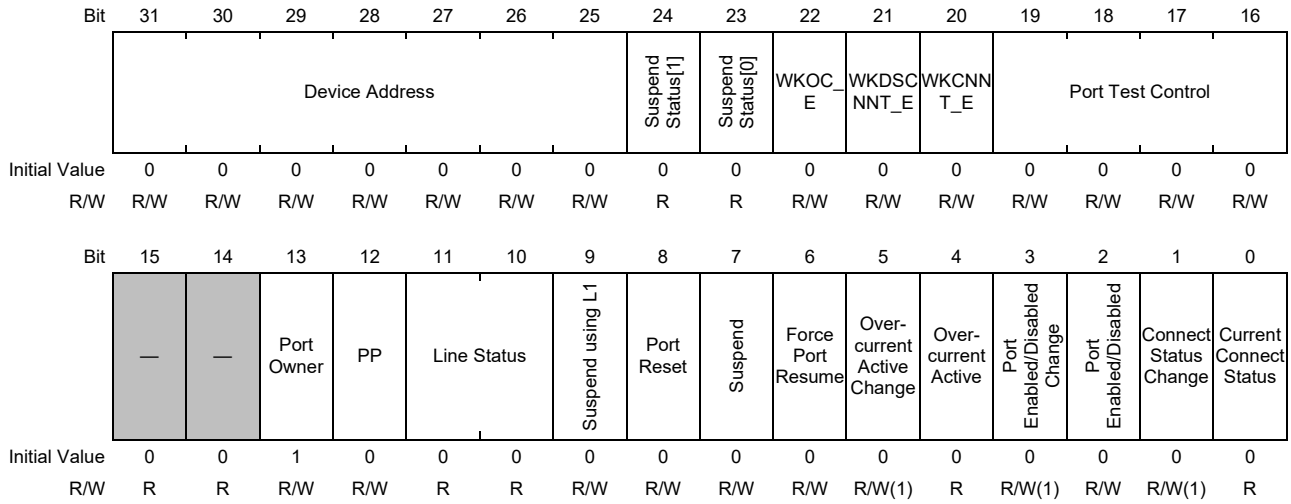
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved The write value should always be 0.
0	CF	0	R/W	Configuration Flag This bit controls which of OHCI or EHCI is routed by the port routing control circuit by default. At the end of the host controller configuration, this bit is set to 1b. 0b: The port routing control circuit routes each port to the OHCI host controller by default. 1b: The port routing control circuit routes each port to the EHCI host controller by default.

(9) PORTSC1Register

Register abbreviation: PORTSC1



Bit	Bit Name	Initial Value	R/W	Description
31 to 25	Device Address	All 0	R/W	This field indicates the USB device address (7 bits) of the device connected to the down port. This address is used when the LPM Token is sent. If the value of this filed is H'00, it means that no device that needs to use this field has been connected.
24, 23	Suspend Status	All 0	R	This field indicates the response from the connected device to the LPM Token (L1 transition request). 00b: The device succeeded transition to the L1 state (ACK received from the device). 01b: The device has not changed to the L1 state (NYET received from the device). 10b: The device does not support the L1 state transition (STALL received from the device). 11b: Other response (for example, Timeout error) Change this field only when the Suspend bit is 0b.
22	WKOC_E	0	R/W	Wake on Overcurrent Enable By writing 1b to this bit, the overcurrent state can be detected as an EHCI Wakeup event. If bit 12 (PP [Port Power] bit) is 0b, this bit becomes 0b.
21	WKDSCNNT_E	0	R/W	Wake on Disconnect Enable By writing 1b to this bit, device disconnection can be detected as an EHCI Wakeup event. If bit 12 (PP [Port Power] bit) is 0b, this bit becomes 0b.
20	WKCNNT_E	0	R/W	Wake on Connect Enable By writing 1b to this bit, device connection can be detected as an EHCI Wakeup event. If bit 12 (PP [Port Power] bit) is 0b, this bit becomes 0b.

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>19 to 16</td><td>Port Test Control</td><td>All 0</td><td>R/W</td><td>This field is controlled by the test mode.
If the value of this field is other than 0000b, it indicates that this module is running in the test mode.
<table>
<tr><th>Value</th><th>Test mode</th></tr>
<tr><td>0000b</td><td>Test mode not enabled</td></tr>
<tr><td>0001b</td><td>Test J_STATE</td></tr>
<tr><td>0010b</td><td>Test K_STATE</td></tr>
<tr><td>0011b</td><td>Test SE0_NAK</td></tr>
<tr><td>0100b</td><td>Test Packet</td></tr>
<tr><td>0101b</td><td>Test FORCE_ENABLE</td></tr>
<tr><td>Other</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>15, 14</td><td>—</td><td>All 0</td><td>R</td><td>Reserved
The write value should always be 0.</td></tr>
<tr><td>13</td><td>Port Owner</td><td>1</td><td>R/W</td><td>This bit indicates which of OHCI or EHCI has the port ownership.
0b: EHCI has the port ownership.
1b: OHCI has the port ownership.
When bit 0 (Configure Flag bit) of the CONFIGFLA register is changed from 0b to 1b, this bit becomes 0b. If bit 0 (Configure Flag bit) of the CONFIGFLA register is 0b, this bit becomes 1b.
If the connected device is not a High Speed device, this bit is set to 1b to transfer the port ownership to OHCI.</td></tr>
<tr><td>12</td><td>PP</td><td>0</td><td>R/W</td><td>Port Power
This bit controls power supply to the port.
If this bit is 0b, power is not supplied to the port. Therefore, the port does not function, and does not recognize connection and disconnection.
If overcurrent is detected while this bit is set to 1b, the host controller clears this bit to 0b, and the power supplied to the port is stopped.
Note: As described later, if the PPC bit is 0b, this bit is fixed to 1b, so the power supplied to the port is not stopped.
The function of this bit differs depending on the value of bit 4 (PPC [Port Power Control] bit) of the HCSPARAMS register.
<table>
<tr><th>PPC</th><th>PP</th><th>Operation</th><th></th></tr>
<tr><td>0b</td><td>1b</td><td>R</td><td colspan="2">This bit is fixed to 1b, and the power is always supplied to the port.</td></tr>
<tr><td>1b</td><td>0b/1b</td><td>R/W</td><td colspan="2">Whether or not power is supplied to the port depends on the setting of this bit.</td></tr>
<tr><td></td><td></td><td></td><td>0b</td><td>Power is not supplied to the port.</td></tr>
<tr><td></td><td></td><td></td><td>1b</td><td>Power is supplied to the port.</td></tr>
</table></td></tr>
<tr><td>11, 10</td><td>Line Status</td><td>All 0</td><td>R</td><td>This field indicates the logical level of D+/D- lines of the current USB bus. (bit 11: DP / bit 10: DM)
This field is used to detect an LS device before starting a sequence for port reset and port enable.
Therefore, this bit is valid only when bit 3 (Port Enable/Disable bit) is 0b and bit 0 (Current Connect Status bit) is 1b.
<table>
<tr><th>bit 11</th><th>bit 10</th><th>USB bus status</th><th>Description</th></tr>
<tr><td>0b</td><td>0b</td><td>SE0</td><td>The device is not an LS device.</td></tr>
<tr><td></td><td></td><td></td><td>EHCI port reset is executing.</td></tr>
<tr><td>1b</td><td>0b</td><td>J-state</td><td>The device is not an LS device.</td></tr>
<tr><td></td><td></td><td></td><td>EHCI port reset is executing.</td></tr>
<tr><td>0b</td><td>1b</td><td>K-state</td><td>An LS device was connected.</td></tr>
<tr><td></td><td></td><td></td><td>The port ownership is transferred from EHCI to OHCI.</td></tr>
<tr><td>1b</td><td>1b</td><td>Undefined</td><td>The device is not an LS device.</td></tr>
<tr><td></td><td></td><td></td><td>EHCI port reset is executing.</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>9</td><td>Suspend using L1</td><td>0</td><td>R/W</td><td>Suspend using L1 (LPM) control bit.
If 1b is written to the Suspend bit (bit 7) while this bit is 1b, the state of this module is changed to the LPM state.
Writing to this bit is possible only when the Suspend bit (bit 7) is 0b.
0b: Suspend using L2
1b: Suspend using L1 (LPM)
If this bit is 1b and the Device Address field is other than H’0000, when the Suspend bit is set to 1b, the host controller generates an LPM Token to change the state to the L1 state.
If this bit is 0b, the host controller operates as L2 Suspend.</td></tr>
<tr><td>8</td><td>Port Reset</td><td>0</td><td>R/W</td><td>This bit indicates the reset status of the port.
0b: The port is not being reset.
1b: The port is being reset.
If 1b is written to this bit while this bit is 0b, the bus reset sequence defined in USB 2.0 standard starts. To finish the bus reset sequence, 0b must be written to this bit. Note that this bit must remain 1b long time enough to guarantee that the bus reset sequence defined in USB 2.0 standard will be complete.
If bit 12 (HCHalted) of the USBSTS register is 1b, do not set this bit to 1b.
If any of the PP (Port Power) bit, Port Owner bit, and Current Connect Status bit is in the following status, this bit becomes 0b.
Note: Even if 1b is written to this bit, the bus reset sequence does not start.
• PP (Port Power) bit = 0b
• Port Owner bit = 1b
• Current Connect Status bit = 0b</td></tr>
<tr><td>7</td><td>Suspend</td><td>0</td><td>R/W</td><td>This bit indicates the Suspend control and status of the port.
This bit and bit 2 (Port Enabled/Disabled bit) indicate the following port status.
<table>
<tr><th>Port Enabled</th><th>Suspend</th><th>Port Status</th></tr>
<tr><td>0b</td><td>—</td><td>Disable</td></tr>
<tr><td>1b</td><td>0b</td><td>Enable</td></tr>
<tr><td></td><td>1b</td><td>Suspend</td></tr>
</table>
To change the port state to L1 or L2 Suspend, set this bit to 1b.
Whether the host controller supports L1 Suspend state or L2 Suspend state depends on the value in the Suspend Using L1 bit.
In the Suspend state, data transfer to the downstream port is blocked by this port (except for port reset). If this bit is set to 1b during data transfer, blocking of transfer data does not occur until the current data transfer finishes.
Writing 0b to this bit has no effect.
This bit can be set to 1b only when all the following conditions are satisfied: “PP (Port Power) bit = 1b”, “Port Owner bit = 0b”, and “Current Connect Status bit = 1b”. If any of the following conditions is satisfied, the host controller clears this bit:
• “Resume is complete” is detected.
• The PR bit is set to 1b when PR (Port Reset) bit is 0b.
• The port owner is 1b (OHCI).
• The PP (Port Power) bit is set to 0b.
• The Port Enabled/Disabled bit is set to 0b.</td></tr>
</table>

Bit	Bit Name	Initial Value	R/W	Description
6	Force Port Resume	0	R/W	This bit indicates that the Resume state of the port is detected. 0b: Resume signal is not detected or output. 1b: Resume (K-state) is detected or output. When the port is in Suspend state, if the host controller detects transition of the state from J to K (if RemoteWakeup is detected from the connected device), it sets this bit to 1b. The host controller also sets the Port Change Detect bit or Port-1 Changes Detect bit of the USBSTS register to 1b. If this bit is set to 1b, the host controller does not set 1b to the Port Change Detect bit and Port-1 Changes Detect bit of the USBSTS register. While this bit is 1b, the Resume signal (FS K) is driven onto the USB bus. For L2 transition, this bit must be cleared to 0b after an appropriate time has passed. By writing 0b to this bit while this bit is 1b, the port status is recovered to be the HS Idle status. This bit remains 1b until the port is recovered. The host controller must finish transition to the HS Idle state within 2msec since this bit is cleared to 0b. On the other hand, for L1 transition, the host controller sends a Resume signal at necessary timing, and this bit is cleared to 0b at Resume recovery. Note that the software sets the length of the Resume signal driven by the host controller, by using the Host-Initiated Resume Duration field of the USBCMD register. If the Port Power (PP) bit is 0b, this bit becomes 0b.
5	Over-current Change	0	R/W (1)	This bit indicates that bit 4 (Over-current Active bit) has changed. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: Over-current Active bit has not changed. 1b: Over-current Active bit has changed.
4	Over-current Active	0	R	This bit indicates the overcurrent status of the port. If the host controller detects overcurrent, it disables the port, and set this bit to 1b. After the overcurrent state is released, the host controller automatically clears this bit from 1b to 0b. 0b: The port is not in overcurrent state. 1b: The port is in overcurrent state.
3	Port Enable/ Disable Change	0	R/W (1)	This bit indicates that the host controller detected frame babble. If the Host Controller detects frame babble, it disables the port and sets this bit to 1b. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. If the Port Power (PP) bit is 0b, this bit becomes 0b. 0b: Frame babble has not occurred. 1b: Frame babble is detected.
2	Port Enabled/ Disabled	0	R/W	This bit indicates the Enable/Disable status of the port. The host controller resets the port, and enables the port if the connected device is recognized as an HS device, and sets this bit to 1b. The software cannot set this bit to 1b. If the host controller detects disconnection of a device or other errors, it disables the port, and clears this bit to 0b. The port also becomes disabled when 0b is written to this bit. If the port is disabled, data transfer to the downstream port is blocked except for port reset. If Port Test Control [3:0] = 0101b (Test FORCE_ENABLE), the port becomes enabled, and this bit is set to 1b. If the Port Power (PP) bit is 0b, this bit becomes 0b. 0b: The port is disabled. 1b: The port is enabled.
1	Connect Status Change	0	R/W (1)	This bit indicates that bit 0 (Current Connect Status bit) has changed. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. If the Port Power (PP) bit is 0b, this bit becomes 0b. 0b: The Current Connect Status bit has no change. 1b: The Current Connect Status bit has changed.

Bit	Bit Name	Initial Value	R/W	Description
0	Current Connect Status	0	R	This bit indicates the connection status of the port. If the host controller detects connection of a device, it sets this bit to 1b. Also, if Port Test Control [3:0] = 0101b (Test FORCE_ENABLE), the Host Controller sets this bit to 1b even if no device is connected. On the other hand, if the host controller detects disconnection of a device, it sets this bit to 0b. If the Port Power (PP) bit is 0b, or the Port Owner (PO) bit is 0b, this bit becomes 0b. 0b: No device is connected to the port. 1b: A device is connected to the port.

32A.2.4.4 AHB Bridge Register

(1) INT_ENABLE Register

Abbreviated name of register: INT_ENABLE

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	WAKEON_INTEN	UCOM_INTEN	USBH_INTBEN	USBH_INTAEN	AHB_INTEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved The write value should always be 0.
4	WAKEON_INTEN	0	R/W	This bit enables or disables bit 4 (WAKEON_INT) of the INT_STATUS register. 0b: disable 1b: enable
3	UCOM_INTEN	0	R/W	This bit enables or disables bit 3 (UCOM_INT) of the INT_STATUS register. 0b: disable 1b: enable
2	USBH_INTBEN	0	R/W	This bit enables or disables bit 2 (USBH_INTB) of the INT_STATUS register. 0b: disable 1b: enable
1	USBH_INTAEN	0	R/W	This bit enables or disables bit 1 (USBH_INTA) of the INT_STATUS register. 0b: disable 1b: enable
0	AHB_INTEN	0	R/W	This bit enables or disables bit 0 (AHB_INT) of the INT_STATUS register. 0b: disable 1b: enable

(2) INT_STATUS Register

Abbreviated name of register: INT_STATUS

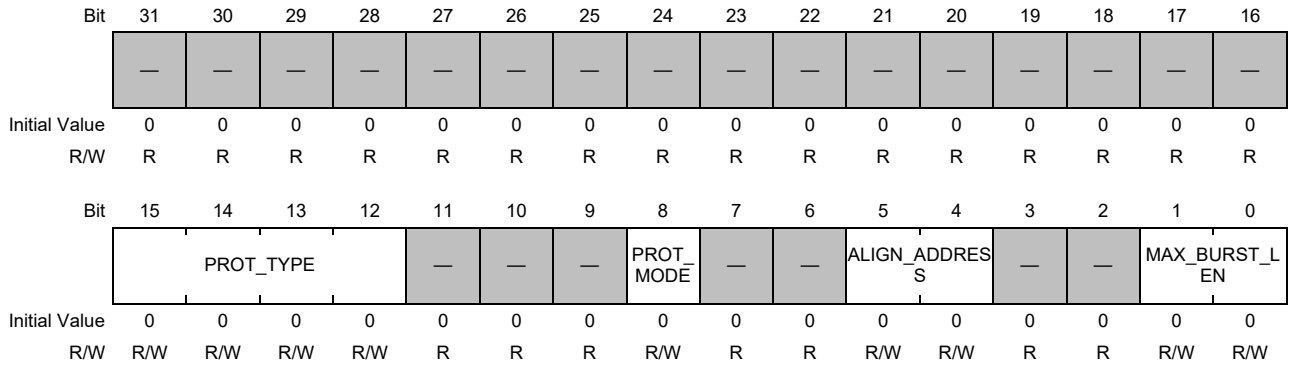
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	WAKEON_INT	UCOM_INT	USBH_INTB	USBH_INTA	AHB_INT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W(1)	R	R	R	R/W(1)

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved The write value should always be 0.
4	WAKEON_INT	0	R/W (1)	This bit indicates the state of WAKEON interrupt from the HOST module. Writing 1b to this bit can clear this bit. 0b: No WAKEON interrupt 1b: WAKEON interrupt
3	UCOM_INT	0	R	This bit indicates the state of interrupt from the UCOM register. To clear the interrupt, use the UCOM2 Register. 0b: No UCOM register interrupt 1b: UCOM register interrupt
2	USBH_INTB	0	R	This bit indicates the state of EHCI interrupt. To clear the interrupt, use the USBSTS Register (of the EHCI Operational Register). 0b: No INTB interrupt 1b: INTB interrupt
1	USBH_INTA	0	R	This bit indicates the state of OHCI interrupt. To clear the interrupt, use the HcInterruptStatus Register (of the OHCI Operational Register). 0b: No INTA interrupt 1b: INTA interrupt
0	AHB_INT	0	R/W (1)	This bit indicates that a BUS Master error occurred. Writing 1b to this bit can clear this bit. 0b: No bus error occurred. 1b: A bus error occurred.

(3) AHB_BUS_CTR Register

Abbreviated name of register: AHB_BUS_CTR



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved The write value should always be 0.
15 to 12	PROT_TYPE	All 0	R/W	This field sets MHPROT [3:0] used when the BUS Master interface issues a transfer request. bit 15 0b: Cache disabled. 1b: Cache enabled. bit 14 0b: Buffer disabled. 1b: Buffer enabled. bit 13 0b: User access 1b: Privileged access bit 12 0b: Operation code 1b: Data
11 to 9	—	All 0	R	Reserved The write value should always be 0.
8	PROT_MODE	0	R/W	This bit selects the mode of MHPROT [3:0] used when the Master interface issues a transfer request. 0b: The value of PROT_TYPE is output as MHPROT [3:0]. 1b: When a DMA transfer is performed, MHPROT [3:0] is set to 0000b if the final burst is performed, or MHPROT [3:0] is set to the PROT_TYPE value if another burst transfer is performed.
7, 6	—	All 0	R	Reserved The write value should always be 0.
5, 4	ALIGN_ADDRESS	All 0	R/W	This field sets the address boundary used when the BUS Master interface issues a burst transfer. 00b: A burst transfer is issued so that not to exceed 1-Kbyte boundary. 01b: A burst transfer is issued so that not to exceed 64-byte boundary. 10b: A burst transfer is issued so that not to exceed 32-byte boundary. (The maximum burst length is INCR8. This is because, if INCR16 is used, 32-byte boundary is exceeded.) 11b: A burst transfer is issued so that not to exceed16 byte boundary. (The maximum burst length is INCR4. This is because, if the length is at least INCR8, 16-byte boundary is exceeded.)
3,2	—	All 0	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	MAX_BURST_LEN	All 0	R/W	This field selects the maximum burst length used when the BUS Master interface issues a transfer request. 00b: INCR16 01b: INCR8 10b: INCR4 11b: SINGLE

(4) USBCTR Register

Abbreviated name of register: USBCTR

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DIRPD	PLL_RST	USBH_RST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved The write value should always be 0.
2	DIRPD	0	R/W	USBPHY Standby Mode Control 0b: USBPHY normal operating mode 1b: USBPHY standby mode *Note*: When the USB module is not in use, setting this bit to 1 reduces power consumption by the USBPHY module. Only set it to 1 when the USB module is not in use. In transitions from USBPHY standby mode to USBPHY normal operating mode, assert the reset signal for the USBPHY module for at least 1 µs before the transition.
1	PLL_RST	1	R/W	This bit controls resetting of the USBPHY module. 0b: The USBPHY reset is released. 1b: The USBPHY module is reset.
0	USBH_RST	0	W	Software reset to this module. Setting this bit to 1b resets this module entirely. This bit is always read as 0b. *Note*: Set this bit only when the BUS Master interface of this module is not running. Access to this module becomes valid 10 CLK (internal bus clock [P1ϕ]) after this bit is written. 0b: Nothing occurs. 1b: Reset is issued to this module.

(5) Register Enable/Clock Gating Control Register

Abbreviated name of register: REGEN_CG_CTRL

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NONUSE_CLK_MSK	—	HOST_CLK_MSK	PERI_CLK_MSK	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	NONUSE_CLK_MSK	0	R/W	This bit is used to mask the clock for the unused host or peripheral controller, depending on the setting value of the OTG_PERI bit of the COMMCTRL register. 0b: Do not mask the clock. 1b: Mask the clock. For details, see **Section 32A.3.1.2, Specifications of NONUSE_CLK_MSK operation**.
30	—	0	R	Reserved The write value should always be 0.
29	HOST_CLK_MSK	0	R/W	This bit is used to forcibly mask clock supply to the host controller. 0b: Do not mask the clock supply. 1b: Mask the clock supply. For details, see **Section 32A.3.1.3, Specifications of PERI_CLK_MSK and HOST_CLK_MSK operations**.
28	PERI_CLK_MSK	0	R/W	This bit is used to forcibly mask the clock supply to the peripheral controller. 0b: Do not mask the clock supply. 1b: Mask the clock supply. For details, see **Section 32A.3.1.3, Specifications of PERI_CLK_MSK and HOST_CLK_MSK operations**.
27 to 24	—	All 0	R	Reserved The write value should always be 0.
23 to 0	—	All 0	R	Reserved The write value should always be 0.

(6) Suspend Control Register

Abbreviated name of register: SPD_CTRL

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SUSPENDM_ENABLE	SLEEPM_ENABLE	—	—	—	—	—	—	WKCNNT_ENABLE	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R

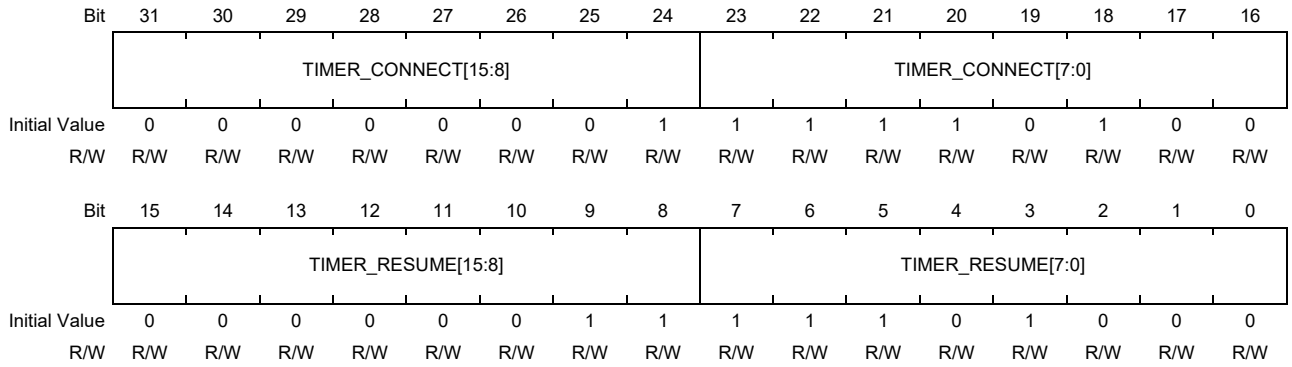
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GLOBAL_SUSPENDM_P1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	SUSPENDM_ENABLE	0	R/W	The setting of this bit is valid only in the Host mode (when the OTG_PERI bit of the COMMCTRL register is 0b). This bit is used to place USBPHY into the suspend status (the state in which PHY built-in PLL is stopped) when this module is changed to the USB Suspend mode. If this bit is set to 1b, the Suspend related bits of the OHCI/EHCI Operational registers below are set. If this module is changed to the USB Suspend mode, USBPHY is placed into the suspend status. [This function is valid for the following OHCI/EHCI Operational registers] EHCI: bit [7] (Suspend bit) of the PORTSC1 register OHCI: bit [2] (PSS bit) of the HcRhPortStatus register OHCI: bit [7:6] (HCFS field) of the HcControl register
30	SLEEPM_ENABLE	0	R/W	The setting of this bit is valid only in the Host mode (when the OTG_PERI bit of the COMMCTRL register is 0b). This bit is used to place USBPHY into the sleep status when the LPM function is used to change to the L1 Suspend mode. (In the sleep status, the PHY built-in PLL is running, but the 60MHz clock from USBPHY is gated.) This bit is valid when "Suspend using L1" of the POTSC register is 1b. If this bit is set to 1b, an L1 transition request is issued to the device. If the request is accepted, USBPHY is placed into the sleep status after the following EHCI register bit is set to 1b. [This function is valid for the following OHCI/EHCI Operational register] EHCI: bit [7] (Suspend bit) of the PORTSC1 register
29 to 24	—	All 0	R	Reserved The write value should always be 0.
23	WKCNNT_ENABLE	0	R/W	The setting of this bit is valid only in the Host mode (the OTG_PERI bit of the COMMCTRL register is 0b). If this bit is set to 1b, when a device disconnect occurs while USBPHY is in the suspend or sleep status, the suspend or sleep status is released. This bit is valid only when the SUSPENDM_ENABLE bit (bit 31) or SLEEP_ENABLE bit (bit 30) is 1b. *Note*: To set 1b to the SUSPENDM_ENABLE bit (bit 31) or SLEEPM_ENABLE bit (bit 30), as the general rule, set this bit to 1b. If, in the above case, this bit is not set to 1b, the suspend or sleep status cannot be released even if a device disconnect occurs.
22 to 1	—	All 0	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	GLOBAL_SUSPENDM_P1	0	R/W	The setting of this bit is valid regardless of the value of the OTG_PERI bit of the COMMCTRL register. This bit is used to forcibly place USBPHY into the suspend status (in which the PHY built-in PLL is stopped). If this bit is set to 1b, USBPHY is placed into the suspend status, regardless of the operating status and port status of the host controller. *Note*: – Do not set this bit to 1b during data transfer. We recommend that you set this bit to 1b after “stopping the EHCI/OHCI list processing” and “placing the port in the disabled status”. – If the SUSPENDM_ENABLE bit (bit 31) is 1b, do not set this bit to 1b.

(7) Suspend/Resume Timer Setting Register

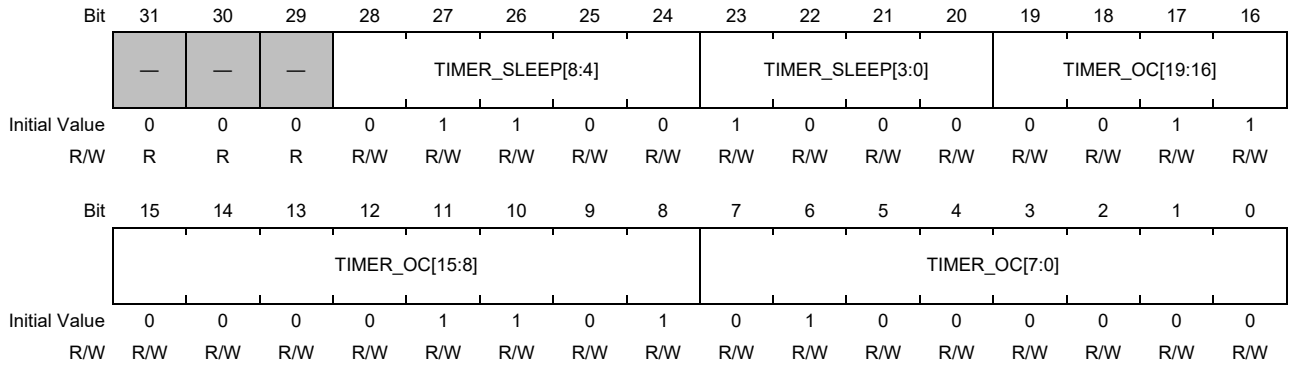
Abbreviated name of register: SPD_RSM_TIMSET



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TIMER_CONNECT[15:0]	H'01F4	R/W	This field indicates the timer value used by USBPHY to detect Device Connect/Disconnect of the device in the suspend status (in which the PHY built-in PLL is stopped) when the SUSPENDM_ENABLE bit of the SPD_CTRL register is set to 1b. When USBPHY is in suspend status, whether Connect/Disconnect occurs is judged by the internal bus clock (P1ϕ). According to the internal bus clock (P1ϕ) frequency, specify the setting so that this timer value becomes at least 2.5 μs. 1 bit = 1 cycle (μs) (Setting guideline) For 100 MHz: At least H'FA
15 to 0	TIMER_RESUME[15:0]	H'03E8	R/W	This field indicates the timer value used by USBPHY to detect RemoteWakeup signal from the device in suspend status (in which the PHY built-in PLL is stopped) when the SUSPENDM_ENABLE bit of the SPD_CTRL register is set to 1b. When USBPHY is in suspend status, whether the RemoteWakeup signal or not is judged by the internal bus clock (P1ϕ). According to the internal bus clock (P1ϕ) frequency, specify the setting so that this timer value becomes at least 5 μs. 1 bit = 1 cycle (μs) (Setting guideline) For 100 MHz: At least H'1F4

(8) Overcurrent Detection/Sleep Timer Setting Register

Abbreviated name of register: OC_SLP_TIMSET



Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved The write value should always be 0.
28 to 20	TIMER_SLEEP[8:0]	H'0C8	R/W	This field indicates the timer value used by USBPHY when the SLEEPM_ENABLE bit of the SPD_CTRL register is set to 1b. This timer value is used to measure the time for detecting RemoteWakeup reception and Resume-K drive time during sleep status (in which the PHY built-in PLL is running, but 60MHz clock from the USBPHY is gated). When USBPHY is in sleep status, whether the RemoteWakeup signal or not is judged by the internal bus clock (P1ϕ). According to the internal bus clock (P1ϕ) frequency, specify the setting so that this timer value becomes 1 µs. 1 bit = 1 cycle (µs) (Setting guideline) For 100 MHz: H'064
19 to 0	TIMER_OC[19:0]	H'30D40	R/W	This field indicates the timer value used for overcurrent detection. If the overcurrent input (OVRCUR) set in this field is continuously asserted (0b) for the duration set in this register, this module determines that overcurrent occurred. According to the internal bus clock (P1ϕ) frequency, specify the setting so that this timer value becomes at least 1 ms. 1 bit is 1 cycle (µs). (Setting guideline) For 100 MHz and 1 ms: At least H'1_86A0

(9) SBRN_FLADJ_PORTWAKECAP Register

Abbreviated name of register: SBRN_FLADJ_PW

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PORTWAKECAP[15:8]								PORTWAKECAP[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FLADJ[7:0]								SBRN[7:0]							
Initial Value	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	PORTWAKECAP[15:0]	H'0003	R/W	This field is used to mask the ports (of the connected device) that are used for Wakeup event. Operation on this field has no effect on the operation of the HOST module.
15 to 8	FLADJ[7:0]	H'20	R, R/W	This field adjusts the length of one micro frame by 16HS bit time unit. The initial value indicates H'20 (60000d HS bit time).
7 to 0	SBRN[7:0]	H'20	R	This field indicates the Serial Bus Release Number. The fixed value "H'20" is indicated.

(10) PORT_LPM_CTR1 Register

Abbreviated name of register: PORT_LPM_CTRL1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	NYET_RETRY_CNT_P1[3:0]				REMOTEWAKE_EN_P1	SLEEP_INT_EN_P1	RETRY_ENABLE_NYET_P1	HIRD_SEL_P1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved The write value should always be 0.
7 to 4	NYET_RETRY_CNT_P1[3:0]	All 0	R/W	This field sets the number of retries that are allowed when the response from the device in the LPM transaction was NYET. The setting value of this bit is valid if the RETRY_ENABLE_NYET_P1 bit (bit 1) is 1b. 0000b: No retry. 0001b to 1111b: Retries the set number of times. (MAX: 15 retries)
3	REMOTEWAKE_EN_P1	0	R/W	This bit is used to indicate the value of the RemoteWakeup bit of the LPM Token. 0b: RemoteWakeup is supported. 1b: RemoteWakeup is not supported.
2	SLEEP_INT_EN_P1	0	R/W	This bit is used to enable an interrupt to occur when a status other than ACK is received in the LPM transaction. The Per-Port Change interrupt can occur. 0b: No interrupt occurs. 1b: An interrupt occurs.
1	RETRY_ENABLE_NYET_P1	0	R/W	This bit is used to set the behavior of the host controller when a response from the device is NYET in an LPM transaction. 0b: No retry. 1b: Retries are made.

Bit	Bit Name	Initial Value	R/W	Description
0	HIRD_SEL_P1	0	R/W	This bit sets the time for K drive for when recovered from the Sleep state. Based on the setting values of this bit and EHCI USBCMD Register bit [27:24], the K drive time is determined as follows.

USBCMD Register	HIRD_SEL_P1 (Setting value of this bit)	
bit [27:24]	0b	1
0000b	75 µs	50 µs
0001b	100 µs	125 µs
0010b	150 µs	200 µs
0011b	250 µs	275 µs
0100b	350 µs	350 µs
0101b	450 µs	425 µs
0110b	950 µs	500 µs
0111b	1950 µs	575 µs
1000b	2950 µs	650 µs
1001b	3950 µs	725 µs
1010b	4950 µs	800 µs
1011b	5950 µs	875 µs
1100b	6950 µs	950 µs
1101b	7950 µs	1025 µs
1110b	8950 µs	1100 µs
1111b	9950 µs	1175 µs

32A.2.4.5 UCOM Register

(1) Common Control Register

Abbreviated name of register: COMMCTRL

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OTG_PERI	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	OTG_PERI	1	R/W	This bit specifies whether this module is set to the host mode or function mode. 0b: Host mode 1b: Function mode (Only CH0 can be set)
30 to 0	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value.

(2) OTG-BC Interrupt Status Register

Abbreviated name of register: OBINTSTA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DPMONCHG_STA	DMMONCHG_STA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W(1)	R/W(1)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CHGDETCHG1_STA	—	PDDETCHG1_STA	—	—	OCINT_STA	IDCHG_STA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R/W(1)	R	R/W(1)	R	R	R/W(1)	R/W(1)

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved The write value should always be 0.
17	DPMONCHG_STA	0	R/W (1)	This bit is set if the DPMON bit of the LINECTRL1 register has changed. 0b: The DPMON bit of the LINECTRL1 register has not changed. 1b: The DPMON bit of the LINECTRL1 register has changed.
16	DMMONCHG_STA	0	R/W (1)	This bit is set if the DMMON bit of the LINECTRL1 register has changed. 0b: The DMMON bit of the LINECTRL1 register has not changed. 1b: The DMMON bit of the LINECTRL1 register has changed.
15 to 7	—	All 0	R	Reserved The write value should always be 0.
6	CHGDETCHG1_STA	0	R/W (1)	This bit is set if the CHGDETSTS bit of the BCCTRL1 register has changed from 0b to 1b. 0b: The CHGDETSTS bit of the BCCTRL1 register has not changed from 0b to 1b 1b: The CHGDETSTS bit of the BCCTRL1 register has changed from 0b to 1b.
5	—	0	R	Reserved The write value should always be 0.
4	PDDETCHG1_STA	0	R/W (1)	This bit is set if the PDDETSTS bit of the BCCTRL1 register has changed from 0b to 1b. 0b: The PDDETSTS bit of the BCCTRL1 register has not changed from 0b to 1b. 1b: The PDDETSTS bit of the BCCTRL1 register has changed from 0b to 1b.
3	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	OCINT_STA	0	R/W (1)	This bit is set if the OVRCUR pin is asserted. 0b: The OVRCUR pin is not asserted (and remains 1b). 1b: The OVRCUR pin is asserted (and became 0b).
0	IDCHG_STA	1	R/W (1)	This bit is set if the input value from the OTG_ID pin has changed. 0b: There is no change in the OTG_ID pin. 1b: There is a change in the OTG_ID pin. *Note*: The initial value is 1. Before using this bit, clear the status.

(3) OTG-BC Interrupt Enable Register

Abbreviated name of register: OBINTEN

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DPMONCHG_EN	DMMONCHG_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CHGDETCHG1_EN	—	PDDETCHG1_EN	—	—	OCINT_EN	IDCHG_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved The write value should always be 0.
17	DPMONCHG_EN	0	R/W	DPMONCHG_STA bit interrupt enable 0b: Interrupt via the DPMONCHG_STA bit is disabled. 1b: Interrupt via the DPMONCHG_STA bit is enabled.
16	DMMONCHG_EN	0	R/W	DMMONCHG_STA bit interrupt enable 0b: Interrupt via the DMMONCHG_STA bit is disabled. 1b: Interrupt via the DMMONCHG_STA bit is enabled.
15 to 7	—	All 0	R	Reserved The write value should always be 0.
6	CHGDETCHG1_EN	0	R/W	CHGDETCHG1_STA bit interrupt enable 0b: Interrupt via the CHGDETCHG1_STA bit is disabled. 1b: Interrupt via the CHGDETCHG1_STA bit is enabled.
5	—	0	R	Reserved The write value should always be 0.
4	PDDETCHG1_EN	0	R/W	PDDETCHG1_STA bit interrupt enable 0b: Interrupt via the PDDETCHG1_STA bit is disabled. 1b: Interrupt via the PDDETCHG1_STA bit is enabled.
3	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	OCINT_EN	0	R/W	OCINT_STA bit interrupt enable 0b: Interrupt via the OCINT_STA bit is disabled. 1b: Interrupt via the OCINT_STA bit is enabled.
0	IDCHG_EN	0	R/W	IDCHG_STA bit interrupt enable 0b: Interrupt via the IDCHG_STA bit is disabled. 1b: Interrupt via the IDCHG_STA bit is enabled.

(4) VBUS Control Register

Abbreviated name of register: VBCTRL

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	VGPUO	—	—	—	VBOUT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved The write value should always be 0.
29	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value.
28 to 22	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value.
21	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value.
20 to 17	—	All 0	R	Reserved The write value should always be 0.
16	—	1	R	Reserved The write value should always be 1.
15 to 5	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value.
4	VGPUO	0	R/W	The level corresponding to the inverse of this bit (in terms of positive logic) is output from the OTG_EXICEN pin. This bit is used, for example, for control of the external power IC.
3 to 1	—	All 0	R	Reserved The write value should always be 0.
0	VBOUT	0	R/W	This bit is one of the VBUSEN control bits. This bit is used to assert VBUS by controlling the external power IC. 0b: VBUS output disable 1b: VBUS output enable If overcurrent occurs, this bit is automatically cleared to 0.

(5) Line Control Port 1 Register

Abbreviated name of register: LINECTRL1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	DPRPU_EN	DP_RPU	DPRPD_EN	DP_RPD	DMRPD_EN	DM_RPD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	DPMON	DMMON	—	IDMON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved
25, 24	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23, 22	—	All 0	R	Reserved
21	DPRPU_EN	0	R/W	This bit enables the DP_RPU (bit 20) setting for pullup control of D+ in port 1. 0: DP_RPU (bit 20) setting for pullup control of D+ in port 1 is disabled. 1: DP_RPU (bit 20) setting for pullup control of D+ in port 1 is enabled.
20	DP_RPU	0	R/W	This bit controls pullup of D+ in port 1. This control is enabled only when DPRPU_EN (bit 21) = 1. Note that if this bit is cleared to 0 when DPRPU_EN (bit 21) = 1, D+ is in the floating state while no device drives the USB bus. Note also that setting DPRPD_EN (bit 19) and DP_RPD (bit 18) to enabled is prohibited while this bit is set to enabled - that is, enabling both pullup and pulldown of D+ together is prohibited. 0: Pullup of D+ in port 1 is disabled. 1: Pullup of D+ in port 1 is enabled.
19	DPRPD_EN	0	R/W	This bit enables DP_RPD (bit 18) to control USB bus (DP) 15 kΩ Pulldown resister. 0b: Control of DP-side 15 kΩ Pulldown resister by DP_RPD (bit 18) is disabled. 1b: Control of DP-side 15 kΩ Pulldown resister by DP_RPD (bit 18) is enabled.
18	DP_RPD	0	R/W	This bit controls USB bus (DP) 15 kΩ Pulldown resister when DPRPD_EN (bit 19) = 1b. 0b: DP-side 15 kΩ Pulldown resister is OFF. 1b: DP-side 15 kΩ Pulldown resister is ON.
17	DMRPD_EN	0	R/W	This bit enables DM_RPD (bit 16) to control USB bus (DM) 15 kΩ Pulldown resister. 0b: Control of DM-side 15 kΩ Pulldown resister by DM_RPD (bit 16) is disabled. 1b: Control of DM-side 15 kΩ Pulldown resister by DM_RPD (bit 16) is enabled.
16	DM_RPD	0	R/W	This bit controls USB bus (DM) 15 kΩ Pulldown resister when DMRPD_EN (bit 17) = 1b. 0b: DM-side 15 kΩ Pulldown resister is OFF. 1b: DM-side 15 kΩ Pulldown register is ON.
15 to 4	—	All 0	R	Reserved The write value should always be 0.
3	DPMON	0	R	This bit indicates the value of USB bus DP.
2	DMMON	0	R	This bit indicates the value of USB bus DM.
1	—	0	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	IDMON	0	R	This bit indicates the value of the OTG_ID pin.

(6) BC Control Port 1 Register

Abbreviated name of register: BCCTRL1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PDDETSTS	CHGDETSTS	—	—	DCPMODE	VDMSRCE	IDPSINKE	VDPSRCE	IDMSINKE	IDPSRCE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	H'00_C000	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value.
9	PDDETSTS	0	R	This bit indicates the USBPHY Portable Device Detect signal state.
8	CHGDETSTS	0	R	This bit indicates the USBPHY Charging Downstream Port Detect signal state.
7, 6	—	All 0	R	Reserved The write value should always be 0.
5	DCPMODE	0	R/W	If USBPHY is used as DCP (Dedicated Charging Port), this bit is set to 1b.
4	VDMSRCE	0	R/W	This bit controls the USBPHY built-in VDM_SRC circuit. If this bit is set to 1b, VDM_SRC goes ON, and the DM pin is driven.
3	IDPSINKE	0	R/W	This bit controls the USBPHY built-in Portable Device Detect circuit. If this bit is set to 1b, Portable Device detection is enabled.
2	VDPSRCE	0	R/W	This bit controls the USBPHY built-in VDP_SRC circuit. If this bit is set to 1b, VDP_SRC goes ON, and the DP pin is driven.
1	IDMSINKE	0	R/W	This bit controls the USBPHY built-in Charging Downstream Port Detect circuit. If this bit is set to 1b, Charging Downstream Port detection is enabled.
0	IDPSRCE	0	R/W	This bit controls the USBPHY built-in IDP_SRC circuit. If this bit is set to 1b, IDP_SRC goes ON, and the DP pin is driven.

32A.3 Clock Signals

32A.3.1 Clock Gating Specifications

32A.3.1.1 Overview of clock gating

Because this module has a feature of switching the host controller and peripheral controller, clock supply to the unused controller might not be necessary.

Therefore, three clock gating control bits are allocated in the Register Enable/Clock Gating Control Register for the purpose of reducing power consumption by implementing clock gating for the “circuits that do not require clock supply temporarily.”

The clock gating control bits can be controlled to gate the clock that is supplied to the host controller or the peripheral controller.

[Target Register]

Register Enable/Clock Gating Control Register (offset:H’304)

[Functional specification]

bit	Symbol	Functional specification
31	NONUSE_CLK_MSK	Gating clocks for unused host controller or peripheral controller.
29	HOST_CLK_MSK	Gating clocks for host controller.
28	PERI_CLK_MSK	Gating clocks for peripheral controller.

32A.3.1.2 Specifications of NONUSE_CLK_MSK operation

This function automatically gates the clock to the unused host controller or the unused peripheral controller.

This function is enabled if the NONUSE_CLK_MSK bit (bit 31) is set to 1b.

If it is no problem whether the clock supply to the unselected function is stopped, use this function.

The following table shows the operating specifications of this function, based on the setting of the OTG_PERI bit (bit 31) of the Common Control Register (offset: H’800).

Clock Gating Register Setting			Host/Peripheral Switching Setting	Gating Target	
			Register		
NONUSE_CLK_MSK	HOST_CLK_MSK	PERI_CLK_MSK	OTG_PERI	Host Controller	Function Controller
1	0	0	0		✓
			1	✓	

[Note on using the NONUSE_CLK_MSK bit]

If the NONUSE_CLK_MSK bit is used, as the general rule, set the HOST_CLK_MSK/PERI_CLK_MSK bit to 0b.

If the HOST_CLK_MSK/PERI_CLK_MSK bit is 1b, the effect of clock gating becomes logical OR of each bit.

32A.3.1.3 Specifications of PERI_CLK_MSK and HOST_CLK_MSK operations

This function forcibly gates the clock to the host controller and function controller.

The following table shows the operating specifications of this function.

Clock Gating Register Setting			Host/Peripheral Switching Setting	Gating Target	
			Register		
NONUSE_CLK_MSK	HOST_CLK_MSK	PERI_CLK_MSK	OTG_PERI	Host Controller	Function Controller
0	1	0	—	✓	
	0	1	—		✓
	1	1	0	✓	✓

Note: If the HOST_CLK_MSK/PERI_CLK_MSK bit is used, as the general rule, set the NONUSE_CLK_MSK bit to 0b.
If the NONUSE_CLK_MSK bit is set t0 1b, the effect of clock gating becomes logical OR of each bit.

32A.4 Interrupt Sources

32A.4.1 Interrupt Signals

This module has the five interrupt signals listed below.

Interrupt Source Name	Interrupt Type	Pulse/ Level	Active Level
U2H_INT	BUS Master interrupt signal. This signal is asserted when a bus error occurs in the BUS Master. Interrupt control is performed by the AHB Bridge Register.	Level	H
U2H_OHCI_INT	OHCI interrupt signal This signal is asserted during FS/LS transfer, when data transfer finishes or when the change of the USB bus state is detected. Interrupt control is performed by the OHCI Operational Register.	Level	H
U2H_EHCI_INT	EHCI interrupt signal. This signal is asserted during HS transfer, when data transfer finishes or when the change of the USB bus state is detected. Interrupt control is performed by the EHCI Operational Register.	Level	H
U2H_WAKEON_INT	EHCI Wakeup interrupt signal. This signal is asserted by an EHCI Wakeup event. Interrupt control is performed by the EHCI Operational Register.	Level	H
U2H_OBINT	OTG/Battery Charging interrupt signal. This signal is asserted by OTG or Battery Charging related event. Interrupt control is performed by the UCOM2 Register.	Level	H

32A.4.2 Interrupt Sources and Control

32A.4.2.1 U2H_INT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the register below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset:H'200) bit[0] (AHB_INTEN)

[Interrupt source]

A bus error (MHRESP = 1b) occurs in the AHB master.

[Clearing the interrupt]

To clear the interrupt, write 1b to the relevant bit in the register below.

AHB Bridge Register INT_STATUS Register (offset:H'204) bit[0] (AHB_INT)

32A.4.2.2 U2H_OHCI_INT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset:H'200) bit[1] (USBH_INTAEN)

OHCI Operational Register HcInterruptEnable Register (offset:H'010) bit[31], bit[6:0]*[1]

Note 1. Enable the bit required as the assertion source.

[Interrupt source]

Interrupt Source		Registers That Require Interrupt Enable Setting							
			HcInterruptEnable						
		USBH_INTAEN	bit [31] MIE	bit [6] RHSCE	bit [5] FNOE	bit [3] ROE	bit [2] SFE	bit [1] WDHE	bit [0] SOE
1	Device connection is detected.	✓	✓	✓					
2	Device disconnection is detected.	✓	✓	✓					
3	Port power is OFF (excluding overcurrent detection).	✓	✓	✓					
4	Babble error is detected during a USB transfer.	✓	✓	✓					
5	Resume is complete.	✓	✓	✓					
6	Overcurrent is detected.	✓	✓	✓					
7	Bus reset is complete.	✓	✓	✓					
8	When the HcRhDescritorB Register DR bit is 1b, OHCI becomes "USB Operational" (HCFS[1:0] bit = 10b) or "USB Suspend" (HCFS[1:0] bit = 11b).	✓	✓	✓					
9	When no device is connected (CCS bit = 0b), 1 is written to bit [0] (Clear Port Enable) of the OHCI HcRhPort Status register.	✓	✓	✓					
10	When no device is connected (CCS bit = 0b), 1 is written to bit [1] (Set Port Enable) of the OHCI HcRhPort Status register.	✓	✓	✓					
11	When no device is connected (CCS bit = 0b), 1 is written to bit [2] (Set Port Suspend) of the OHCI HcRhPort Status register.	✓	✓	✓					
12	When no device is connected (CCS bit = 0b), 1 is written to bit [3] (Clear Suspend Status) of the OHCI HcRhPort Status register.	✓	✓	✓					
13	When no device is connected (CCS bit = 0b), 1 is written to bit [4] (Set Port Reset) of the OHCI HcRhPort Status register.	✓	✓	✓					
14	When no device is connected (CCS bit = 1b), the port power is turned off.	✓	✓	✓					
15	The MSB of bit [15:0] (Frame Number) of the HcFmNumber register has changed.	✓	✓		✓				
16	RemoteWakeup signal (Resume signal) is detected from a device.	✓	✓			✓			
17	HccaFrameNumber is updated. (Almost the same meaning as SOF is sent.)	✓	✓				✓		
18	A transfer finishes (including an error), and the host module updated HccaDoneHead.	✓	✓					✓	
19	USB schedule overrun occurred for the frame.	✓	✓						✓

[Clearing the interrupt]

To clear the interrupt, write 1b to the bit corresponding to the interrupt source in the register below to clear the interrupt.

OHCI Operational Register HcInterruptStatus Register (offset:H'00C) bit[6:0]

32A.4.2.3 U2H_EHCI_INT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.
AHB Bridge Register INT_ENABLE Register (offset:H'200) bit[2] (USBH_INTBEN)

EHCI Operational Register USBINTR Register (offset:H'128) bit[16], bit[5:0]*1

Note 1. Enable the bit required as the assertion source.

Also, control the relevant bit(s) in the register below as required.

EHCI Operational Register USBCMD Register (offset:H'120) bit[15], bit[6]

[Interrupt source]

Interrupt Source		Registers That Require Interrupt Enable Setting								
			USBCMD		USBINTR					
			bit [15]	bit [6]	bit [16]	bit [5]	bit [3]	bit [2]	bit [1]	bit [0]
		USBH_INTBEN	Per-Port Change Event	Doorbell	Port-1 Change Event	Async Advance	Frame List Rollover	Port Change Detect	USB ERRINT	USB INT
[1]	Device connection is detected.	✓						✓		
[2]	Device disconnection is detected.	✓						✓		
[3]	Overcurrent is detected.	✓						✓		
[4]	RemoteWakeup signal (Resume Signal) is detected from a device.	✓								
[5]	Babble status of the USB bus is detected.	✓						✓		
[6]	USB transfer with "qTD IOC = 1b" normally finishes.	✓								✓
[7]	Short packet is received.	✓								✓
[8]	USB transfer finished with an error. (Retry transfer failed three times. A bubble error was detected. STALL was received.)	✓							✓	
[9]	The QH processing normally finished while the USBCMD Register bit [6] (Interrupt on Async Advance Doorbell) is 1b.	✓		✓		✓				
[10]	The FRINDEX Register FrameIndex bit returned from the maximum value to H'000 (rollover detected).	✓					✓			
[11]	The Port Change Detect event (interrupt source 1 to 5) was detected.	✓	✓		✓					

[Clearing the interrupt]

To clear the interrupt, write 1b to the bit corresponding to the interrupt source in the register below to clear the interrupt.

EHCI Operational Register USBSTS Register (offset:H'124) bit[16], bit[5:0]

32A.4.2.4 U2H_WAKEON_INT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset:H'200) bit[4] (WAKEON_INTEN)

EHCI Operational Register PORTSC[1:2] Register (offset:H'164/H'168) bit[22:20]*1

Note 1. Enable the bit required as the assertion source.

[Interrupt source]

Interrupt Source		Registers That Require Interrupt Enable Setting			
			PORTSC		
		WAKEON_INTEN	bit [22] WKOC_E	bit [21] WKDSCNNT_E	bit [20] WKCNNT_E
1	Device connection is detected.	✓			✓
2	Device disconnection is detected.	✓		✓	
3	Overcurrent is detected.	✓	✓		
4	RemoteWakeup signal (Resume Signal) is detected from a device.	✓			

[Clearing the interrupt]

To clear the interrupt, write 1b to the relevant bit in the register below.

AHB Bridge Register INT_STATUS Register (offset:H'204) bit[4] (WAKEON_INT)

32A.4.2.5 U2H_OBINT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset:H'200) bit[3] (UCOM_INTEN)

UCOM Register OTG-BC Interrupt Enable Register (offset:H'808) bit[17:16], bit[6], bit[4], bit[1:0]*[1]

Note 1. Enable the bit required as the assertion source.

[Interrupt source]

Interrupt Source	Registers That Require Interrupt Enable Setting		
		OTG-BC Interrupt Enable	
		bit [17]	bit [16]
	UCOM_INTEN	DPMONCHG_EN	DMMONCHG_EN
1 The DP pin has changed.	✓	✓	
2 The DM pin has changed.	✓		✓

Interrupt Source	Registers That Require Interrupt Enable Setting		
		OTG-BC Interrupt Enable	
		bit [6]	bit [4]
	UCOM_INTEN	CHGDETCHG1_EN	PDDETCHG1_EN
1 The portable device detection signal has changed.	✓		✓
2 The charging port detection signal has changed.	✓	✓	

Interrupt Source	Registers That Require Interrupt Enable Setting		
		OTG-BC Interrupt Enable	
		bit [1]	bit [0]
	UCOM_INTEN	OCINT_EN	IDCHG_EN
1 Overcurrent is detected. (The change of the OVRCUR pin from 1 to 0b is detected.)	✓	✓	
2 The OTG_ID pin has changed.	✓		✓

[Clearing the interrupt]

To clear the interrupt, write 1b to the bit corresponding to the interrupt source in the register below to clear the interrupt.

UCOM Register OTG-BC Interrupt Status Register (offset:H'804) bit[17:16], bit[6], bit[4], bit[1:0]

32A.4.3 Timing of De-asserting Interrupt Signals

After the register access to clear an interrupt source, it may take time to begin clearing the interrupt triggered by the interrupt source. Therefore, take a measure to prevent the false recognition of interrupts during the period from the end of the register clear access until the next interrupt is recognized.

32A.5 Power-Saving Function

This module controls power consumption by using the following two methods:

1. Controlling the SUSPENDM/SLEEPM pin of USBPHY
2. Using clock gating to stop the clock to the host controller or peripheral controller.

32A.5.1 Controlling the SUSPENDM and SLEEPM Pins of the USBPHY

You can expect the following power-saving effects by asserting the SUSPENDM and SLEEPM pins of the USBPHY:

- Reducing the power consumption by the USBPHY
- Reducing the power consumption by the host core by stopping the clocks from the USBPHY

As described at **Section 32A.1.2.5, Suspend extension function**, by default, the SUSPENDM and SLEEPM pins of the USBPHY are not asserted even when EHCI and OHCI are put into the Suspended state.

See **Section 32A.2.4.4(6), Suspend Control Register** and control the relevant registers appropriately.

32A.5.2 Controlling the Clock-Gating Function

See **Section 32A.2.4.4(5), Register Enable/Clock Gating Control Register** and **Section 32A.3.1, Clock Gating Specifications**, then control the relevant registers appropriately.

32A.6 Battery Charging

32A.6.1 Support of charging port

The charging port refers to a port that supplies power in compliance with the Battery Charging Specification. The charging port is usually installed on the host controller.

Charging ports can be generally classified by function as described below.

Type	Function
CDP (Charging Downstream Port)	Downstream port that can supply power in compliance with the Battery Charging Specification. This type of charging port detects a portable device, and after the handshake (for Battery Charging) finishes, it proceeds to the usual device connect sequence (and operates as the usual host).
DCP (Dedicated Charging Port)	Port that provides only power supply function in compliance with the Battery Charging Specification. This type of charging port does not operate as the usual host.
SDP (Standard Downstream Port)	Standard Downstream Port that is not compliant with the Battery Charging Specification. This type of charging port supplies power in the range conforming to the conventional USB 2.0 standard, and operates as the usual host.

32A.6.2 Support of portable device

A portable device refers to a device that is supplied power (or that requests power supply) in compliance with the Battery Charging Specification. The portable port is usually installed in the peripheral controller.

32A.7 Bus Master

32A.7.1 Functional specifications of the bus master

32A.7.1.1 Supported bus master functions

Function as the BUS MASTER	Status
Residual burst after an error response is received	The transfer is not stopped.
1 Kbyte boundary processing	The fixed length burst (INCRx) across 1 Kbyte boundary is not performed.

32A.7.1.2 Issuing requests for different types of bus transfer

MHTRANS[1:0]	MHSIZE[2:0]	MHWRITE	MHBURST[2:0]	Reply	Remarks
IDLE (00b)	—	—	—	—	—
BUSY (01b)	—	—	—	—	Not issued.
NONSEQ (10b)	32-bit (010b)	WRITE	SINGLE INCR4 INCR8 INCR16	OKAY/ERROR	32-bit transfer is issued. A response error is reported by an interrupt, and the transfer is not stopped.
		READ			
	8-bit (000b) 16-bit (001b)	WRITE	SINGLE	OKAY/ERROR	8-bit/16-bit transfer is issued (SINGLE only). A response error is reported by an interrupt.
	other than the above	—	—	—	Not issued.
SEQ (11b)	32-bit (010b)	WRITE	INCR4 INCR8 INCR16	OKAY/ERROR	32-bit transfer is issued. A response error is reported by an interrupt, and the transfer is not stopped.
		READ		OKAY/ERROR	
	other than the above	—	—	—	Not issued.

32A.7.1.3 Supported responses

Response type	Response	Remarks
OKAY	Enable	Supported.
ERROR	Enable	A response error is reported by an interrupt, and termination of transfer (Early Burst Termination) is not performed.

32A.7.1.4 Protection control information

The value of MHPROT[3:0] can be set in the PROT_TYPE bits (bits [15:12]) in the AHB_BUS_CTR Register (offset: H'208).

Also, when the PROT_MODE bit (bit 8) in the AHB_BUS_CTR Register is set, only the last data transfer in an EHCI/OCHI DMA transfer can be handled as a non-buffered transfer, and other transfers can be handled as buffered transfers.

32A.7.1.5 Maximum burst length

The maximum burst length can be selected from SINGLE, INCR4, INCR8, and INCR16 by using the MAX_BURST_LEN bits (bit[1:0]) in the AHB_BUS_CTR Register (offset: H'208). The maximum burst length is common to reading and writing.

32A.7.1.6 Boundary of transfer data

The value of MHADDR[31:0] bits does not exceed 1 KB boundary during a burst transfer. Also, you can change the address boundary for burst transfer to 16, 32, or 64 bytes by writing a value to the ALIGN_ADDRESS bits (bit[5:4]) in the AHB_BUS_CTR Register (offset: H'208).

32A.7.1.7 Start address of fixed-length INCR burst transfer

The following table lists the values of the lower bits of MHADDR to be applied when a fixed-length INCR burst transfer starts.

ALIGN_ADDRESS Setting	Fixed length Start Address of INCR Burst		
	INCR4	INCR8	INCR16
00b (Aligned at the 1-Kbyte boundary)	MHADDR[9:0] = H'000 H'004 H'008 H'00C H'010 : H'3D0 H'3D4 H'3D8 H'3DC H'3E0 H'3E4 H'3E8 H'3EC H'3F0	MHADDR[9:0] = H'000 H'004 H'008 H'00C H'010 : H'3C8 H'3CC H'3D0 H'3D4 H'3D8 H'3DC H'3E0	MHADDR[9:0] = H'000 H'004 H'008 H'00C H'010 : H'3B8 H'3BC H'3C0
01b (Aligned at the 16-byte boundary)	MHADDR[3:0] = H'0	-- (Not issued)	-- (Not issued)
10b (Aligned at the 32-byte boundary)	MHADDR[4:0] = H'00 H'04 H'08 H'0C H'10	MHADDR[4:0] = H'00	-- (Not issued)
11b (Aligned at the 64-byte boundary)	MHADDR[5:0] = H'00 H'04 H'08 H'0C H'10 H'14 H'18 H'1C H'20 H'24 H'28 H'2C H'30	MHADDR[5:0] = H'00 H'04 H'08 H'0C H'10 H'14 H'18 H'1C H'20	MHADDR[5:0] = H'00

32A.8 Overcurrent Control and VBUS Control

32A.8.1 OVRCUR/VBUSEN pin

Overcurrent detection on the USB port and port power (VBUS) control are performed by the external power IC connected to this module.

This module pin	Input/Output	Level	Description
OVRCURI	Input	L	Overcurrent status was detected.
		H	Overcurrent status was not detected.
VBUSEN	Output	L	Port Power (VBUS) OFF
		H	Port Power (VBUS) ON

32A.8.2 Overcurrent detection timer setting

This module detects overcurrent when the OVRCUR pin remains asserted (0b) for a set period.

The period over which assertion of the OVRCUR pin is required (“vercurrent detection time”) can be set in the following register.

Register	Overcurrent detection/sleep timer setting register (offset: H’310)
Bits	TIMER_OC[19:0]
Initial value	H’3_0D40

The overcurrent detection time can be converted from the setting value of the above register, taking that one bit equal to the internal bus clock (P1ϕ) cycle.

Therefore, set the value of the above register at initial configuration, considering “the internal bus clock (P1ϕ) frequency to be used” and “the overcurrent detection time you want to specify”.

32A.8.3 Port Power (VBUS) control specifications

VBUSEN can be controlled by the Port Power bit of the EHCI/OHCI Operational register or by the VBOUT bit of the VBUS Control Register. Which of the above bit controls VBUSEN is determined by the VBUSEN control (VBENCTL) register in USBPHY Control. Refer to **Section 32.3.4, VBUSEN Control Register (VBENCTL)**. When using as Function, VBUSEN should be 0.

The following table describes control examples by the Port Power bit of the EHCI/OHCI Operational register.

Situation		Register		bit
EHCI control		PORTSC1 (offset: H'164)		bit 12 (PP)
OHCI control	Global control*1	ON setting	HcRhStatus Register (offset: H'050)	bit 16 (Set Global Power)
		OFF setting		bit 0 (Clear Port Status)
	Selective control*2	ON setting	HcRhPortStatus1 Register (offset: H'054)	bit 8 (Set Port Power)
		OFF setting		bit 9 (Clear Port Power)

Note 1. Global control refers to the status in which the register settings are as follows:
HcRhDescriptorA Register (offset: H'048) bit 8 (PSM) = 0b
or
HcRhDescriptorA Register (offset: H'048) bit 8 (PSM) = 1b
and
HcRhDescriptorB Register (offset: H'04C) bit 17 (PPCM[1]) = 0b

Note 2. Selective control refers to the status in which the register settings are as follows:
HcRhDescriptorA Register (offset: H'048) bit 8 (PSM) = 1b
and
HcRhDescriptorB Register (offset: H'04C) bit 17 (PPCM[1]) = 1b

However, if the register settings are as follows, the VBUSEN pin is always asserted (1b), and the Port Power (VBUS) becomes ON, regardless of the OVRCUR pin's status.

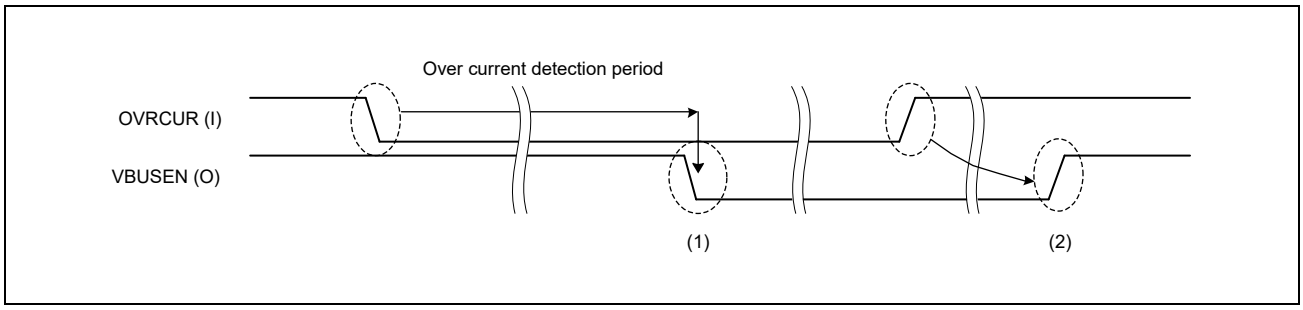
Specify the settings as necessary (for example, your system needs that VBUS is always ON).

EHCI Operational	OHCI Operational				
HCSPARAMS (offset: H'104)	HcRhDescriptorA (offset: H'048)			HcRhDescriptorB (offset: H'04C)	Pin Operation
PPC (bit 4)	NOCP (bit 12)	NPS (bit 9)	PSM (bit 8)	PPCM[1] (bit 17)	When the OVRCUR pin is asserted (0b)
0	—	—	—	—	Fixed to 1b
—	1	—	—	—	Fixed to 1b
—	—	1	—	—	Fixed to 1b
1	0	0	0	—	0b
			1	0	
				1	

32A.8.4 Timing Chart for Overcurrent Detection and Recovery

The figure below shows the assertion/de-assertion timings of the OVRCUR and VBUSEN pins signals at overcurrent detection and recovery.

Note that this timing chart is on the assumption that the changes of register settings to fix the Port Power bit to asserted state (see **Section 32A.8.3, Port Power (VBUS) control specifications**) have not been made.



1. When the OVRCUR pin is kept asserted (0b) for the overcurrent detection time, this module determines the occurrence of overcurrent, and then de-asserts the VBUSEN pin (0b).
2. After the overcurrent status has been resolved, and de-assertion of the OVRCUR pin (1b) is confirmed, 1b is written to the Port Power bit described in **Section 32A.8.3, Port Power (VBUS) control specifications** to turn on the Port Power (Vbus).

NOTE

Before the Port Power bit is set by firmware, be sure to check that the OVRCUR pin has been deasserted.

32A.9 Procedure for Setting this Module

32A.9.1 Host/Peripheral Common Setting Sequence

The following shows the necessary sequence common to both host and peripheral modes.

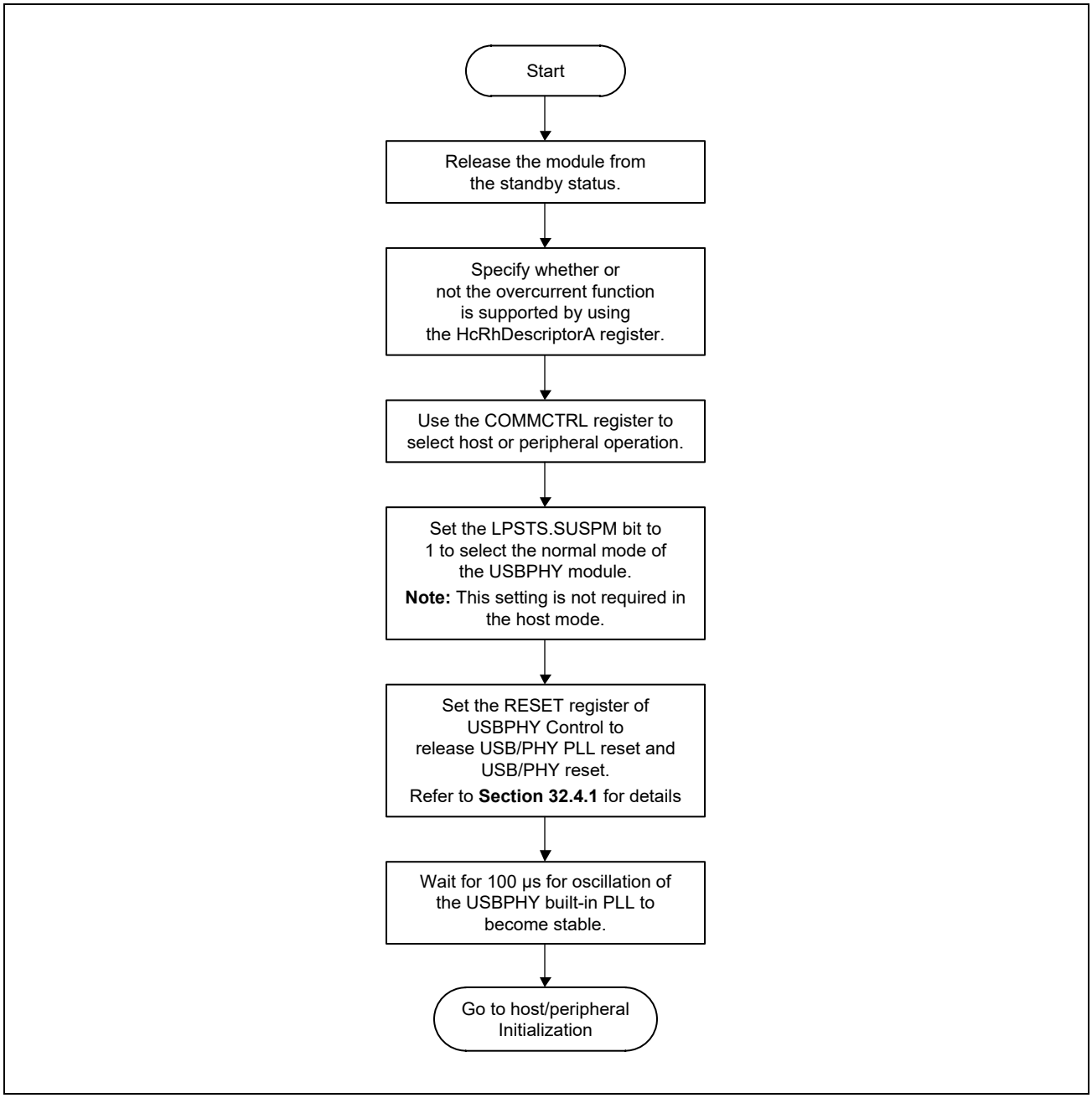


Figure 32A.1 Sequence Common to Both Host and Peripheral Modes

32A.9.2 Initialization Sequence

The following shows the initialization sequence in the Host mode

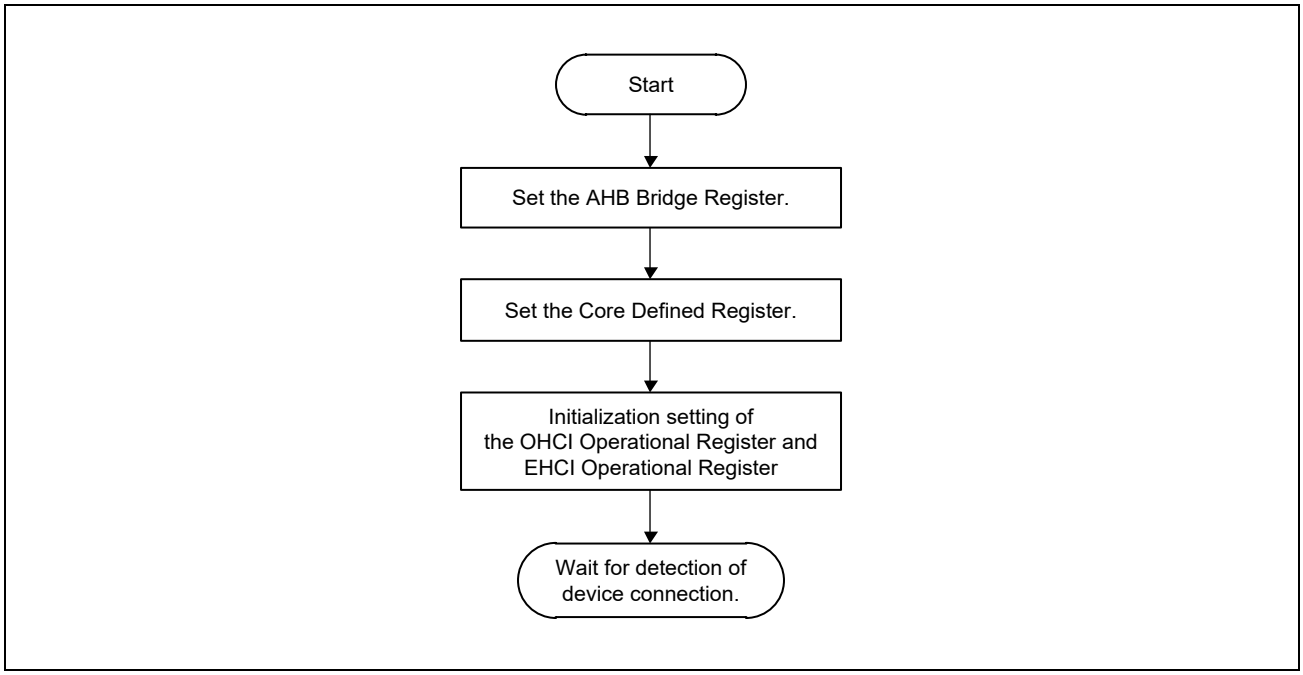


Figure 32A.2 Initialization Sequence

32A.9.3 Flow of Error Handling

While operating this module, if the operation falls into an abnormal state and recovery to the normal flow seems difficult, perform the following steps for reset.

[When an abnormality occurs while EHCI is running]

1. Write 1b to the HCRESET bit (bit 1) of the EHCI Operational Register USBCMD Register (offset: H'120) to execute EHCI software reset.
2. Re-initialize the EHCI Operational Register.

[When an abnormality occurs while OHCI is running]

1. Write 1b to the HCR bit (bit 0) of the OHCI Operational Register HcCommandStatus Register (offset: H'008) to execute OHCI software reset.
2. Re-initialize the OHCI Operational Register.

32B. USB 2.0 Function Module

32B.1 Overview

32B.1.1 Overview

This LSI has two channels of USB 2.0 host/function module. Switching between the host and peripheral functions for each channel is possible by setting the UCOM register.

The setting for battery charging is handled by the host controller even if the peripheral controller is selected.

This chapter describes the peripheral controller. For details on the host/peripheral common circuit and battery charging, see **Section 32A, USB 2.0 Host Module**.

This module is a universal serial bus (USB) controller that has peripheral functions.

This module supports high-speed and full-speed transfer defined by the Universal Serial Bus Specification Revision 2.0.

This module supports all transfer types defined in the USB Specification. This module incorporates 8 Kbytes of buffer memory for data transfer, and can use a maximum of 10 pipes. You can assign any endpoint number to any pipe other than pipe 0 in conformity to the peripheral equipment or system to communicate with.

32B.1.2 Features

32B.1.2.1 Peripheral controller supporting high-speed USB

(1) On-chip peripheral USB controller

32B.1.2.2 Support of all types of USB transfer

(1) Supporting all types of USB transfer, including isochronous transfer

(2) Control transfer

(3) Bulk transfer

(4) Interrupt transfer (high-bandwidth transfers not supported)

(5) Isochronous transfer (high-bandwidth transfers not supported)

32B.1.2.3 Bus interface

(1) Includes a two-channel DMA interface

32B.1.2.4 Pipe configuration

(1) 8 Kbytes of buffer memory for USB communications for each channel

(2) Up to 10 pipes (including the default control pipe) selectable for each channel

(3) Programmable pipe configuration

(4) Any endpoint number assignable to pipes other than pipe 0

Table 32B.1 Pipe Settings

PIPE	Transfer Type	Double buffer	Continuous transfer mode	Buffer size
PIPE0	Control	—	—	Fixed to 64 bytes/ 256 bytes (CNTMD = 1)
PIPE1	Iso/Bulk	✓	✓ (Bulk only)	Up to 2 Kbytes
PIPE2	Iso/Bulk	✓	✓ (Bulk only)	Up to 2 Kbytes
PIPE3	Bulk	✓	✓	Up to 2 Kbytes
PIPE4	Bulk	✓	✓	Up to 2 Kbytes
PIPE5	Bulk	✓	✓	Up to 2 Kbytes
PIPE6	Int	—	—	Fixed to 64 bytes
PIPE7	Int	—	—	Fixed to 64 bytes
PIPE8	Int	—	—	Fixed to 64 bytes
PIPE9	Int	—	—	Fixed to 64 bytes

32B.1.2.5 Features of peripheral functions

(1) Support of high-speed transfer (at 480 Mbps) and full-speed transfer (at 12 Mbps)

(2) Automatic recognition of high-speed or full-speed operation based on automatic response to the reset handshake

(3) Control transfer stage monitoring function

(4) Device state monitoring function

(5) Automatic response to SET_ADDRESS request

(6) NAK response interrupt (NRDY)

(7) SOF interpolation

32B.1.2.6 Features of DMA transfer

DMA transaction mode: Fetching in both register and link modes are supported.

Interrupt: Level is supported.

Transfer size: A transfer size from 1 to 128 bytes can be selected separately for the transfer source and transfer destination.

Skip (scatter/gather) function: The access size and skip size can be specified separately for the transfer source and destination.

Suspend function: A running DMA transaction can be suspended temporarily.

Interval function: The interval of DMA transfers can be specified to control the bus occupancy.

32B.1.2.7 Other functions

(1) Byte endian swap function to support both big endian and little endian as data formats (when using only CFIFO)

(2) Transfer ending function using a transaction counter

(3) SOF pulse output function

(4) BRDY interrupt event notification timing change function (BFRE)

(5) Function (SHTNAK) to set NAK in the response PID when transfer ends

(6) Support of the Link Power Management (LPM) ECN, making available a new low-power-consumption state (L1 state)

32B.1.3 Overview of Functions

32B.1.3.1 Automatic recognition of USB transfer speed

This module automatically recognizes USB transfer speed.

(1) Methods of FIFO buffer memory access

This module supports the two types of access described below to the FIFO buffer memory for USB data transfer.

(a) Access from the CPU

Specify a FIFO port address, and then write data to or read data from the FIFO buffer memory.

(b) Direct memory access (DMA)

Selecting a pipe window and setting the DMA control registers enables writing data to or reading data from the FIFO buffer memory.

32B.1.3.2 USB event

This module notifies the event in USB operation by issuing an interrupt.

You can specify whether to enable notification by interrupt for individual interrupt types and sources through software settings.

32B.1.3.3 USB data transfer

This module performs all types of USB data transfer: control transfer, bulk transfer, interrupt transfer, and isochronous transfer. The following pipe resources are available for individual transfer types:

(1) One pipe dedicated to control transfer

(2) Four pipes dedicated to interrupt transfer

(3) Three pipes dedicated to bulk transfer

(4) Two pipes selectively used for bulk or isochronous transfer

For each pipe, specify the settings, including transfer type, endpoint number, and maximum packet size, required for USB transfer according to the system.

This module can incorporate up to 8 Kbytes of buffer memory. For the pipes dedicated to bulk transfer and those selectively used for bulk transfer or isochronous transfer, allocate buffer memory and specify a buffer operating mode and other necessary settings according to the system. Setting the buffer operating mode enables high-speed data transfers with fewer interrupts to be performed by using double-buffering and continuous transfer of data packets.

32B.1.3.4 SOF pulse output function

This module has a function to output an SOF pulse to indicate the timing of SOF packet transmission. This module asserts a SOF pulse output signal when an SOF packet is received. This module outputs pulses at regular intervals based on an SOF interpolation timer even when an SOF packet is damaged.

32B.1.4 Restriction matter and Notes

32B.1.4.1 Restriction matter

(1) Restrictions on the USB specifications

It’s no support About below of USB 2.0 specification

- HighBandWidth transfer is no support

(2) Restrictions on DMA Master

- DAD = 1 (destination address fixed) and cannot use skip transfer of destination side. When forwarding by such setting, movement is unsettled. Should not do such transmission.
- SAD = 1 (source address fixed), and cannot use skip transfer of source side. When forwarding by such setting, movement is unsettled. Should not do such transmission.
- DAD = 1 (destination address fixed), and cannot use the beat unalign transfer of destination side. When forwarding by such setting, movement is unsettled. Should not do such transmission.
- SAD = 1 (source address fixed), and cannot use beat unalign transfer of source side. When forwarding by such setting, movement is unsettled. Should not do such transmission.
- When REQD =1, SBE = 1 (sweep mode) and compulsion discharge function cannot use.

32B.1.4.2 Notes

(1) DMA transfer and passing problem of interrupt signal in DMA Master construction.

(a) Overview

A USBFDMAmn interrupt (m, n = 0, 1) might occur before the last data of DMA transaction is written to the write-target device.

(b) Plan to avoid

A plan to avoid the above problem is indicated below.

Plan to avoid 1) HPROT is set as non bufferable.

According to the DMA mode, set 0 in the DPR[2] bit in the CHEXT_n register or the LDPR[2] bit in the DCTRL register, set the HPROT signal as non bufferable, and then perform the DMA transaction.

If all transfers are set to non bufferable, transfer efficiency might fall.

In such cases, set the most part of transaction as bufferable and perform a transfer, and then perform the last transfer by setting a register set or descriptor as non bufferable.

(2) Setting for battery charging when the peripheral controller is selected

The setting for battery charging is handled by the host controller even if the peripheral controller is selected.

32B.2 Registers

How to read the register table

(1) Bit number:

(2) State after reset: Initial state of the register that occurs immediately after a reset “Power on Reset” indicates the initial state at power-on reset.
The state after USB reset is the initial state of the register that occurs when this module detects a USB bus reset.
Significant points regarding reset operation are indicated in notes.
“—” indicates that a user’s setting is retained without this module operation having been performed. “X” indicates that the value is undefined.

(4) Access condition: The condition to be met when this module accesses the register for an operation.

R: Reading only
W: Writing only
R/W: Reading or writing
R(0): 0-reading only
W(1): 1-writing only

(5) Name: Bit symbol and bit name

(6) Function: Description of functions.

<Example of description>

(1) Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	A bit	B bit	C bit	—	—	—	—	—	—	—	—	—	—	—	—
(2) Initial Value	X	0	0	0												
R/W	X	0	—	—												

Bit	Bit Name	Initial Value	R/W	Description
15	—			Nothing is assigned. Fix this bit to 0.
14	A bit		R/W	AAA enable 0: Disables operation 1: Enables operation
13	B bit		R	BBB operation 0: Outputs low-level signal 1: Outputs high-level signal
12	C bit		R(0)/ W(1)	CCC control 0: ..... 1: .....
	(5)		(4)	(6)

32B.2.1 Base Address

Table 32B.2 Base addresses for each channel of the USB function module

Channel number	Base address
0	H'0_11E2_0000 (Cortex-A55 Address Space)
	H'41E2_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
	H'51E2_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Note: Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

32B.2.2 List of Registers

Table 32B.3 lists the registers of this module.

Table 32B.3 List of Registers (1/3)

Address	Symbol	Name	Access unit
000	SYSCFG0	System Configuration Control Register 0	16-bit
002	SYSCFG1	System Configuration Control Register 1	16-bit
004	SYSSTS0	System Configuration Status Register	16-bit
008	DVSTCTR0	Device Control Register 0	16-bit
00C	TESTMODE	Test Mode Register	16-bit
014	CFIFO	CFIFO Port Register	8-/16-/32-bit
016			
020	CFIFOSEL	CFIFO Port Select Register	16-bit
022	CFIFOCTR	CFIFO Port Control Register	16-bit
028	D0FIFOSEL	D0FIFO Port Select Register	16-bit
02A	D0FIFOCTR	D0FIFO Port Control Register	16-bit
02C	D1FIFOSEL	D1FIFO Port Select Register	16-bit
02E	D1FIFOCTR	D1FIFO Port Control Register	16-bit
030	INTENB0	Interrupt Enable Register 0	16-bit
036	BRDYENB	BRDY Interrupt Enable Register	16-bit
038	NRDYENB	NRDY Interrupt Enable Register	16-bit
03A	BEMPENB	BEMP Interrupt Enable Register	16-bit
03C	SOFCFG	SOF Output Configuration Register	16-bit
040	INTSTS0	Interrupt Status Register 0	16-bit
046	BRDYSTS	BRDY Interrupt Status Register	16-bit
048	NRDYSTS	NRDY Interrupt Status Register	16-bit
04A	BEMPSTS	BEMP Interrupt Status Register	16-bit
04C	FRMNUM	Frame Number Register	16-bit
04E	UFRMNUM	Micro Frame Number Register	16-bit
050	USBADDR	USB Address Register	16-bit
054	USBREQ	USB Request Type Register	16-bit
056	USBVAL	USB Request Value Register	16-bit
058	USBINDX	USB Request Index Register	16-bit
05A	USBLENG	USB Request Length Register	16-bit
05C	DCPCFG	DCP Configuration Register	16-bit

Table 32B.3 List of Registers (2/3)

Address	Symbol	Name	Access unit
05E	DCPMAXP	DCP Max. Packet Size Register	16-bit
060	DCPCTR	DCP Control Register	16-bit
064	PIPESEL	Pipe Window Select Register	16-bit
068	PIPECFG	Pipe Configuration Register	16-bit
06A	PIPEBUF	Pipe Buffer Setting Register	16-bit
06C	PIPEMAXP	Pipe Max. Packet Size Register	16-bit
06E	PIPEPERI	Pipe Cycle Control Register	16-bit
070	PIPE1CTR	PIPE1 Control Register	16-bit
072	PIPE2CTR	PIPE2 Control Register	16-bit
074	PIPE3CTR	PIPE3 Control Register	16-bit
076	PIPE4CTR	PIPE4 Control Register	16-bit
078	PIPE5CTR	PIPE5 Control Register	16-bit
07A	PIPE6CTR	PIPE6 Control Register	16-bit
07C	PIPE7CTR	PIPE7 Control Register	16-bit
07E	PIPE8CTR	PIPE8 Control Register	16-bit
080	PIPE9CTR	PIPE9 Control Register	16-bit
090	PIPE1TRE	PIPE1 Transaction Counter Enable Register	16-bit
092	PIPE1TRN	PIPE1 Transaction Counter Register	16-bit
094	PIPE2TRE	PIPE2 Transaction Counter Enable Register	16-bit
096	PIPE2TRN	PIPE2 Transaction Counter Register	16-bit
098	PIPE3TRE	PIPE3 Transaction Counter Enable Register	16-bit
09A	PIPE3TRN	PIPE3 Transaction Counter Register	16-bit
09C	PIPE4TRE	PIPE4 Transaction Counter Enable Register	16-bit
09E	PIPE4TRN	PIPE4 Transaction Counter Register	16-bit
0A0	PIPE5TRE	PIPE5 Transaction Counter Enable Register	16-bit
0A2	PIPE5TRN	PIPE5 Transaction Counter Register	16-bit
0D0	DEVADD0	Device Address 0 Configuration Register	16-bit
0D2	DEVADD1	Device Address 1 Configuration Register	16-bit
0D4	DEVADD2	Device Address 2 Configuration Register	16-bit
0D6	DEVADD3	Device Address 3 Configuration Register	16-bit
0D8	DEVADD4	Device Address 4 Configuration Register	16-bit
0DA	DEVADD5	Device Address 5 Configuration Register	16-bit
0DC	DEVADD6	Device Address 6 Configuration Register	16-bit
0DE	DEVADD7	Device Address 7 Configuration Register	16-bit
0E0	DEVADD8	Device Address 8 Configuration Register	16-bit
0E2	DEVADD9	Device Address 9 Configuration Register	16-bit
0E4	DEVADDA	Device Address A Configuration Register	16-bit
100	LPCTRL	Low Power Control Register	16-bit
102	LPSTS	Low Power Status Register	16-bit
104	PHYFUNCTR	PHY Function Control Register	16-bit
10A	PHYOTGCTR	PHY OTG Control Register	16-bit
144	PL1CTRL1	Peripheral L1 Control Register 1	16-bit
146	PL1CTRL2	Peripheral L1 Control Register 2	16-bit
400	N0SA_0	Next0 Source Address Register 0	32-bit
404	N0DA_0	Next0 Destination Address Register 0	32-bit

Table 32B.3 List of Registers (3/3)

Address	Symbol	Name	Access unit
408	N0TB_0	Next0 Transaction Byte Register 0	32-bit
40C	N1SA_0	Next1 Source Address Register 0	32-bit
410	N1DA_0	Next1 Destination Address Register 0	32-bit
414	N1TB_0	Next1 Transaction Byte Register 0	32-bit
418	CRSA_0	Current Source Address Register 0	32-bit
41C	CRDA_0	Current Destination Address Register 0	32-bit
420	CRTB_0	Current Transaction Byte Register 0	32-bit
424	CHSTAT_0	Channel Status Register 0	32-bit
428	CHCTRL_0	Channel Control Register 0	32-bit
42C	CHCFG_0	Channel Configuration Register 0	32-bit
430	CHITVL_0	Channel Interval Register 0	32-bit
434	CHEXT_0	Channel Extension Register 0	32-bit
438	NXLA_0	Next Link Address Register 0	32-bit
43C	CRLA_0	Current Link Address Register 0	32-bit
440	N0SA_1	Next0 Source Address Register 1	32-bit
444	N0DA_1	Next0 Destination Address Register 1	32-bit
448	N0TB_1	Next0 Transaction Byte Register 1	32-bit
44C	N1SA_1	Next1 Source Address Register 1	32-bit
450	N1DA_1	Next1 Destination Address Register 1	32-bit
454	N1TB_1	Next1 Transaction Byte Register 1	32-bit
458	CRSA_1	Current Source Address Register	32-bit
45C	CRDA_1	Current Destination Address Register 1	32-bit
460	CRTB_1	Current Transaction Byte Register 1	32-bit
464	CHSTAT_1	Channel Status Register 1	32-bit
468	CHCTRL_1	Channel Control Register 1	32-bit
46C	CHCFG_1	Channel Configuration Register 1	32-bit
470	CHITVL_1	Channel Interval Register 1	32-bit
474	CHEXT_1	Channel Extension Register 1	32-bit
478	NXLA_1	Next Link Address Register 1	32-bit
47C	CRLA_1	Current Link Address Register 1	32-bit
600	SCNT_0	Source Continuous Register 0	32-bit
604	SSKP_0	Source Skip Register 0	32-bit
608	DCNT_0	Destination Continuous Register 0	32-bit
60C	DSKP_0	Destination Skip Register 0	32-bit
620	SCNT_1	Source Continuous Register 1	32-bit
624	SSKP_1	Source Skip Register 1	32-bit
628	DCNT_1	Destination Continuous Register 1	32-bit
62C	DSKP_1	Destination Skip Register 1	32-bit
700	DCTRL	DMA Control Register	32-bit
704	DSCITVL	Descriptor Interval Register	32-bit
710	DSTAT_EN	DMA Status EN Register	32-bit
714	DSTAT_ER	DMA Status ER Register	32-bit
718	DSTAT_END	DMA Status END Register	32-bit
71C	DSTAT_TC	DMA Status TC Register	32-bit
720	DSTAT_SUS	DMA Status SUS Register	32-bit

32B.2.3 System Configuration Control Registers

32B.2.3.1 System Configuration Control Register 0 [SYSCFG0] <Address: H'000>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CNEN	HSE	—	DRPD	DPRPU	—	—	—	USBE
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
8	CNEN	0	R/W	This bit prohibits or enables single-ended receiver operation. 0: Single-ended receiver operation prohibited 1: Single-ended receiver operation enabled
7	HSE	0	R/W	This bit prohibits or enables High-Speed operation. 0: High-Speed operation prohibited (Full-Speed) 1: High-Speed operation enable (The controller detects the communication speed.)
6	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
5	DRPD	1	R/W	D+/D- line resistor control Set this bit to 0 to use this module. For details, see Control of USB Data Bus Resistors.
4	DPRPU	0	R/W	D+ line resistor control This bit prohibits or enables D+ line pull-up for the peripheral controller function. For details, see Control of USB Data Bus Resistors. 0: Pull Up prohibited 1: Pull Up enabled
3 to 1	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
0	USBE	0	R/W	USB block operation prohibited This bit prohibits or enables USB block operation. 0: USB block operation prohibited 1: USB block operation enabled

Note: Data can be written to and read from this register even while the UTMI+PHY clock is stopped. However, if a value is set while the UTMI+PHY clock is stopped, the corresponding function takes effect after the oscillation of UTMI+PHY clock starts.

(1) Single-ended receiver operation enable (CNEN) bit

Setting this bit enables the single-ended receiver to operate. This bit is intended to prevent damage by inrush current that can be caused when the single-ended receiver in unattached status goes floating. This bit also allows the LNST bit to be referenced.

Set this bit when VBUS is detected as the result of a VBUS interrupt. Clear this bit when VBUS is removed.

(2) High-speed operation enable (HSE) bit

Setting this bit enables the high-speed operation. When this bit is 1, this module performs a high-speed or full-speed operation according to the result of reset handshake.

- When the HSE bit is 0, this module performs a full-speed operation.
- When the HSE bit is 1, this module executes the reset handshake protocol, and then automatically performs a high-speed or full-speed operation according to the result of reset handshake.

Rewriting the value of this bit must be done when the DPRPU bit is 0.

(3) D+/D- line resistor control (DRPD or DPRPU) bit

Table 32B.4 shows available settings of the resistors for the USB data bus. Use the DPRPU bit to select the USB data bus resistors.

Table 32B.4 Control of USB Data Bus Resistors

Setting		Control of USB Data Bus Resistors		
DRPD	DPRPU	D- Line	D+ Line	Remarks
0	0	Open	Open	
0	1	Open	Pull-Up	Specify the settings as shown in the left.
1	0	Pull-Down	Pull-Down	Initial state (When power on reset is canceled)
1	1	Pull-Down	Pull-Up	Setting prohibited

(a) D+ pull-up resistor control (DPRPU) bit

Setting this bit enables this module to notify the USB host of attaching by pulling up the D+ line voltage to 3.3 V. Clearing this bit enables this module to let the USB host know that the device has been detached by stopping pulling up the D+ line voltage.

(4) USB block operation enable (USBE) bit

This bit enables or disables the operation of the USB block of this module.

When this bit is changed from 1 to 0, this module initializes the bits shown in **Table 32B.5**.

Table 32B.5 Register Bits That Are Initialized by Writing 0 to the USBE Bit

Register Name	Bit Name
SYSSTS0	LNST
DVSTCTR0	RHST
INTSTS0	DVSQ
USBADDR	USBADDR
USBREQ	bRequest bmRequestType
USBVAL	wValue
USBINDX	wIndex
USBLENG	wLength

Note: Changing the value of this bit must be done when the SUSPENDM bit is 1 and after the oscillation of UTMI+PHY clock starts.

32B.2.3.2 System Configuration Control Register 1 [SYSCFG1] <Address: H'002>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	BWAIT					
Initial Value	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
11 to 8	—	H'F	R	Nothing is assigned to these bits. Fix these bits to 0.
7, 6	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
5 to 0	BWAIT	H'0F	R/W	CPU bus access wait specification These bits specify the number of wait cycles for the access to this module. 000000: 0 wait cycles (2 access cycles) : 000010: 2 wait cycles (4 access cycles) : 000100: 4 wait cycles (6 access cycles) : 001111: 15 wait cycles (17 access cycles) (default) : 111111: 63 wait cycles (65 access cycles)

(1) CPU bus access wait specification (BWAIT) bits

These bits specify the wait cycles for the access to the HPB.

The following restriction is placed on the cycle of the access to the registers at address H'04 or after of this module: Restriction on wait cycle: The cycle of continuous accesses to registers of this module must be at least 67 ns.

To comply with this restriction, you must control the number of wait cycles with the internal bus clock (P1ϕ) frequency.

The default of wait cycles is 17 clock cycles (maximum limit). Select an optimum setting.

This setting is also applied to accesses to FIFO port registers. The maximum speeds of accesses to FIFO ports are as follows:

MBW = 10 (32-bit access width): Max 60 MBytes/sec
MBW = 01 (16-bit access width): Max 30 MBytes/sec
MBW = 00 (8-bit access width): Max 15 MBytes/sec

32B.2.4 System Configuration Status

32B.2.4.1 System Configuration Status Register (SYSSTS0) <Address: H'004>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LNST	
Initial Value	—	—	0	0	0	0	0	0	0	0	0	0	0	—	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 2	—	See above	R	Nothing is assigned to these bits. Fix these bits to 0.
1, 0	LNST	All 0	R	USB Line status monitor The USB line status is displayed. *Note*: See the detailed description.

(1) Line status monitor (LNST) bits

Table 32B.6 shows the line status of the USB data bus of this module. This module monitors the line status (status of the D+ and D- lines) of the USB data bus in the LNST bits of the SYSSTS0 register.

Referencing the LNST bits must be done only after the USBE bit is set and attaching is performed (the DPRPU bit is set).

Table 32B.6 Line Status of USB Data Bus

LNST [1]	LNST [0]	Full-Speed operation	High-Speed operation	Chirp operation
0	0	SE0	Squelch	Squelch
0	1	J State	Unsquelch	Chirp J
1	0	K State	Invalid	Chirp K
1	1	SE1	Invalid	Invalid

Note: Chirp: State in which high-speed operation is enabled (HSE = 1) and the reset handshake protocol is being executed
Squelch: SE0 or idle state
Unsquelch: High-speed J or high-speed K state
Chirp J: Chirp J State
Chirp K: Chirp K State

32B.2.5 USB Signal Control Registers

32B.2.5.1 Device State Control Register 0 [DVSTCTR0] <Address: H'008>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	WKUP	—	—	—	—	—	RHST		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	0	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R/W(1)	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved. When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
14 to 9	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
8	WKUP	0	R/W (1)	Remote wakeup output This bit prohibits or enables Remote wakeup (resume signal output). 0: Remote wakeup signal is not output. 1: Remote wakeup signal is output.
7 to 3	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
2 to 0	RHST	All 0	R	Reset handshake This bit indicates the reset handshake status. *Note*: See the detailed description.

(1) Remote wakeup (resume signal output) enable (WKUP) bit

When this bit is set, this module outputs the remote wakeup signal to the USB.

This module manages the time of remote wakeup signal output. When the WKUP bit is set, this module outputs the K state for 10 ms, and then clears the WKUP bit.

The USB Specification requires the USB idle state to be retained for at least 5 ms before the remote wakeup signal is sent. Therefore, even if the WKUP bit is set immediately after the suspended state is detected, this module waits for 2 ms, and then outputs the K state.

Writing 1 to the WKUP bit must be done only when the device is in the suspended state (DVSQ = 1xx) and remote wakeup is allowed by the USB host.

When setting the WKUP bit, do not stop the internal clock even if the device is in the suspended state. (Write 1 to the WKUP bit when the SUSPM bit is 1.)

When the WKUP bit is set at a transition to the L1 state, this module outputs the K state for 50 µs, and then clears the WKUP bit. In the L1 state, setting the WKUP bit must be done only when the DVSQ[4] bit is 1.

(2) Reset handshake status (RHST) bits

This module outputs the result of reset handshake to this bit. **Table 32B.7** lists the results of reset handshake.

Table 32B.7 Reset Handshake Status

Bus State	Value of RHST Bit
Powered or disconnected state	000
Reset handshake in process	100
Full-speed connection	010
High-speed connection	011

If the HSE bit is 1, the RHST bits indicate 100 when this module detects a USB bus reset. Then, after this module has output Chirp K, these bits indicate 011 when this module detects Chirp JK from the USB host three times. If the status is not fixed to High-Speed within 2.5 ms after Chirp K is output, these bits indicate 010.

If the HSE bit is 0, the RHST bits indicate 010 when this module detects a bus reset.

After this module has detected a USB reset, a DVST interrupt occurs when the value of the RHST bits is fixed to 010 or 011.

32B.2.6 Test Mode Register

32B.2.6.1 USB Test Mode Register [TESTMODE] <Address: H'00C>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UTST			
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
14	—	0	R	Reserved. When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
13, 12	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
11	—	0	R	Reserved. When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
10 to 4	—	H'10	R	Nothing is assigned to these bits. Fix these bits to 0.
3 to 0	UTST	All 0	R/W	Test mode See the detailed description.

(1) Test mode (UTST) bits

When a value is written to these bits, this module outputs a USB test signal during high-speed operation.

Table 32B.8 lists the test modes of this module.

Table 32B.8 List of test mode operation

Test mode	Value of UTST Bits
Normal operation	0000
Test_J	0001
Test_K	0010
Test_SE0_NAK	0011
Test_Packet	0100
Test_Force_Enable	—
Reserved	0101 to 0111

Write a value to these bits according to the SetFeature request sent from the USB host during high-speed communication. When these bits contain a value from 0001 to 0100, this module does not enter the suspended state.

To perform a normal USB communication after setting a test mode, perform Power on reset.

32B.2.7 FIFO Port Registers

32B.2.7.1 CFIFO Port Register [CFIFO] <Address: H'014>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIFOPORT (Low)															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

32B.2.7.2 CFIFO Port Register [CFIFO] <Address: H'016>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FIFOPORT (High)															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	FIFOPORT	All 0	R/W	FIFO port These bits are accessed to read received data from the FIFO buffer or to write send data to the FIFO buffer.

(1) FIFO port control bits (for FIFOPORT)

The send/receive buffer memory of this module has a FIFO structure (FIFO buffer). Use FIFO port registers to access the FIFO buffer. The FIFO port consists of the port register (CFIFO) to read data from and write data to the FIFO buffer, the register (CFIFOSEL) to select the pipe to be allocated to the FIFO port, and the control register (CFIFOCTR).

Individual FIFO ports have the following features:

1. The CFIFO port must be used to access the FIFO buffer through the DCP.
2. When functions specific to FIFO ports are used, the pipe number (selected pipe) specified in the CURPIPE bits cannot be changed.
3. The registers configured for a FIFO port do not affect any other FIFO ports.
4. The FIFO buffer memory can be accessed by either the CPU or SIE. Access by the CPU is not possible while the SIE has the right of access to the FIFO buffer memory.

(2) FIFO port bits (CFIFO)

When one of these registers is accessed, this module accesses the FIFO buffer allocated to the pipe number specified in the CURPIPE bits in the corresponding pipe select register (CFIFOSEL).

These registers can be accessed only when the FRDY bit of the respective control registers (CFIFOCTR) is 1 (or when the UCL_Dx_DREQ output is asserted by this module).

The valid bits of these registers vary depending on the values of the NBW and BIGEND bits. The valid bits are shown in **Table 32B.9** to **Table 32B.11**.

Table 32B.9 Endian Operation in 32-Bit Access (When MBW = 10)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	N + 3 address	N + 2 address	N + 1 address	N + 0 address
1	N + 0 address	N + 1 address	N + 2 address	N + 3 address

Table 32B.10 Endian Operation in 16-Bit Access (When MBW = 01)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	Writing: invalid Reading: prohibited*1		even-numbered address	odd-numbered address
1	even-numbered address	odd-numbered address	Writing: invalid Reading: prohibited*1	

Table 32B.11 Endian Operation in 8-Bit Access (When MBW = 00)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	Writing: invalid Reading: prohibited*1			Writing: valid Reading: valid
1	Writing: valid Reading: valid	Writing: invalid Reading: prohibited*1		

Note 1. Reading words or bytes from an invalid register is prohibited.

32B.2.7.3 CFIFO Port Select Register [CFIFOSEL] <Address: H'020>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	—	—	MBW		—	BIGEND	—	—	ISEL	—	CURPIPE			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R(0)/W	R	R	R/W	R/W	R	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	RCNT	0	R/W	Read count mode This bit specifies DTLN read mode for the CFIFOCTR register. 0: Clears the DTLN bits when all received data is read. 1: Decrements the value of the DTLN bits each time received data is read.
14	REW	0	R(0)/W	Buffer pointer rewind Set this bit to 1 to rewind the buffer pointer. 0: Does not rewind the buffer pointer. 1: Rewinds the buffer pointer.
13, 12	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
11, 10	MBW	All 0	R/W	CFIFO port access bit width This bit specifies the bit width for access to the CFIFO port. 00: 8-bit width 01: 16-bit width 10: 32-bit width 11: Setting prohibited
9	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
8	BIGEND	0	R/W	FIFO port byte endian control This bit specifies the byte endian of the CFIFO port. 0: Little endian 1: Big endian
7, 6	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
5	ISEL	0	R/W	FIFO port access direction with DCP selected This bit specifies the FIFO port access direction when DCP is selected for the CURPIPE bits. 0: Selects reading of buffer memory. 1: Selects writing of buffer memory.
4	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
3 to 0	CURPIPE	All 0	R/W	FIFO port access pipe specification This bit specifies the pipe number used for accessing the CFIFO port. 0000: DCP 0001: PIPE1 0010: PIPE2 ↓ 1000: PIPE8 1001: PIPE9 ↓ 1110: PIPE14 1111: PIPE15

(1) Read count mode (RCNT) bit

When this bit is 0, this module clears the DTLN bits in the CFIFOCTR register when all received data has been read from the FIFO buffer allocated to the pipe (selected pipe) specified in the CURPIPE bits (or, in the case of double-buffer configuration, when all relieved data has been read from one buffer).

With this bit is 1, this module decrements the value of the DTLN bits in the CFIFOCTR register each time received data is read from the FIFO buffer allocated to the specified pipe.

(2) Buffer pointer rewind (REW) bit

When this bit is set during data reading from the FIFO buffer while the selected pipe is in the receiving direction, reading can be restarted from the first data in the FIFO buffer (or, in the case of a double-buffer configuration, rereading can be started from the first data in the FIFO buffer being read).

Do not set this bit at the same time as changing the value of the CURPIPE bits. Before setting this bit, always check that the FRDY bit is 1.

If you want to redo writing to the FIFO buffer from the first data in the FIFO buffer when the selected PIPE is in the sending direction, use the BCLR bit.

(3) CFIFO port access bit width (MBW) bits

These bits are used to specify the bit width for the access to the CFIFO port.

If you start reading after setting a value in these bits when the pipe specified in the CURPIPE bits is in the receiving direction, do not change the value of these bits until data is all read.

Also, to set a value in these bits when the pipe specified in the CURPIPE bits is in the receiving direction, temporarily change the original value of the CURPIPE bits to a different value, and then set the values of the CURPIPE and MBW bits at the same time.

For how to change the value of the CURPIPE bits, see the description of the CURPIPE bits.

When the pipe specified in the CURPIPE bits is in the sending direction, you cannot change the bit width from 8 bits to 16 bits or 32 bits or from 16 bits to 32 bits while writing to the buffer memory is in process.

Even with the 16-bit width or 32-bit width setting, you can write data also to odd bytes by using byte access control.

(4) FIFO port byte endian control (BIGEND) bit

This bit is used to specify the byte endian of the CFIFO port.

For details see **Section 32B.2.7.2(1), FIFO port control bits (for FIFOPORT)**.

(5) FIFO port access direction with DCP selected (ISEL) bit

To change the value of this bit when the specified pipe is DCP, write a value to this bit, read the bit, and then check that the written value is the same as the read value before proceeding to the next processing.

If the value of the bit is changed in the middle of an access to the FIFO buffer, the access is held. Therefore, the same access can be resumed after these bits are restored to the original value.

Set this bit simultaneously with setting of the CURPIPE bits.

(6) FIFO port access pipe specification (CURPIPE) bits

These bits are used to specify the pipe number of the pipe through which to read or write data via the CFIFO port.

If you change the value of these bits, write a desired value to these bits, read these bits, and then check that the written value is the same as the read value before proceeding to the next processing.

If the value of these bits is changed in the middle of an access to the FIFO buffer, the access is held. Therefore, the same access can be resumed after these bits are restored to the original value.

32B.2.7.4 D0FIFO Port Select Register [D0FIFOSEL] <Address:H'028>
D1FIFO Port Select Register [D1FIFOSEL] <Address:H'02C>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	DCLRM	DREQE	MBW		—	—	—	—	—	—	CURPIPE			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R(0)/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	RCNT	0	R/W	Read count mode This bit specifies Dx_FIFOCTR DTLN read mode. 0: Clears the DTLN bits when all received data is read. 1: Decrements the value of the DTLN bits each time received data is read.
14	REW	0	R(0)/W	Buffer pointer rewind Set this bit to 1 to rewind the buffer pointer. 0: Does not rewind the buffer pointer. 1: Rewinds the buffer pointer.
13	DCLRM	0	R/W	Automatic buffer memory clear mode after reading data through the specified pipe This bit prohibits or enables automatic buffer memory clear after data is read through the specified pipe. 0: Automatic FIFO buffer clear prohibited 1: Automatic FIFO buffer clear enabled
12	DREQE	0	R/W	UCL_Dx_DREQ output enable This bit prohibits or enables the output of the UCL_Dx_DREQ signal. 0: Output prohibited 1: Output enabled
11, 10	MBW	All 0	R/W	DxFIFO port access bit width This bit specifies the bit width for access to the DxFIFO port. 00: 8-bit width 01: 16-bit width 10: 32-bit width 11: Setting prohibited
9 to 4	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
3 to 0	CURPIPE	All 0	R/W	FIFO port access pipe specification 0000: No specification 0001: PIPE1 0010: PIPE2 ↓ 1000: PIPE8 1001: PIPE9

(1) Read count mode (RCNT) bit

When this bit is 0, this module clears the DTLN bits in the DxFIFOCTR register when all received data has been read from the FIFO buffer allocated to the pipe (selected pipe) specified in the CURPIPE bits (or, in the case of double-buffer configuration, when all relieved data has been read from one buffer).

With this bit is 1, this module decrements the value of the DTLN bits in the DxFIFOCTR register each time received data is read from the FIFO buffer allocated to the specified pipe.

(2) Buffer pointer rewind (REW) bit

When this bit is set during data reading from the FIFO buffer while the selected pipe is in the receiving direction, reading can be restarted from the first data in the FIFO buffer (or, in the case of a double-buffer configuration, rereading can be started from the first data in the FIFO buffer being read).

Do not set this bit at the same time as changing the value of the CURPIPE bits. Before setting this bit, always check that the FRDY bit is 1.

If you want to redo writing to the FIFO buffer from the first data in the FIFO buffer when the selected PIPE is in the sending direction, use the BCLR bit.

(3) Automatic FIFO buffer clear enable (DCLRM) bit

This bit is used to enable or disable the mode to automatically clear the FIFO buffer memory after reading data from the specified pipe. With this bit set, this module performs the “BCLR = 1” processing on the FIFO buffer, if a zero-length packet is received when the FIFO buffer allocated to the specified pipe is empty or if data reading ends because a short packet is received when the BFRE bit is 1.

Always clears this bit when you use this module with the BRDYM bit set.

(4) UCL_Dx_DREQ output enable (DREQE) bit

This bit is used to enable or disable the output of the UCL_Dx_DREQ signal.

When enabling the UCL_Dx_DREQ signal, set this bit always after setting a value in the CURPIPE bits. When changing the value of the CURPIPE bits, change the value always after clearing this bit.

(5) DxFIFO port access bit width (MBW) bits

These bits are used to specify the bit width for the access to the DxFIFO port. For details, see **Section 32B.2.7.3(3), CFIFO port access bit width (MBW) bits**.

(6) FIFO port access pipe specification (CURPIPE) bits

These bits are used to specify the pipe number through which to read or write data via the DxFIFO port.

If you change the value of these bits, write a desired value to these bits, read these bits, and then check that the written value is the same as the read value before proceeding to the next processing.

Do not specify the same pipe number for the CURPIPE bits in the CFIFOSEL, D0FIFOSEL, and D1FIFOSEL registers. If the value of these bits is changed in the middle of an access to the FIFO buffer, the access is held. Therefore, the same access can be resumed after these bits are restored to the original value.

32B.2.7.5 CFIFO Port Control Register [CFIFOCTR] <Address: H'022>
D0FIFO Port Control Register [D0FIFOCTR] <Address: H'02A>
D1FIFO Port Control Register [D1FIFOCTR] <Address: H'02E>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BVAL	BCLR	FRDY	—	DTLN											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W(1)	R(0)/W(1)	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	BVAL	0	R/W(1)	Buffer memory valid flag Specify 1 for this bit when writing to the FIFO buffer on the CPU side through the pipe specified in CURPIPE (current pipe) ends. 0: Invalid 1: Writing complete
14	BCLR	0	R(0)/W(1)	CPU buffer clear Specify 1 for this bit to clear the FIFO buffer on the CPU side of the current pipe. 0: Invalid 1: CPU buffer memory clear
13	FRDY	0	R	FIFO port ready This bit indicates whether the FIFO port can be accessed. 0: The FIFO port cannot be accessed. 1: The FIFO port can be accessed.
12	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
11 to 0	DTLN	All 0	R	Receive data length These bits indicate the length of receive data.

(1) Buffer memory valid flag (BVAL)

When the pipe (selected pipe) specified in the CURPIPE bits is in the sending direction, this bit must be set in the cases descried below. This module switches the FIFO buffer from the CPU side to the SIE side to enable data sending.

(1) To send short packets, set this bit when data writing ends.

(2) To send zero length packets, set this bit before writing data to the FIFO buffer.

(3) Set this bit after writing, to the pipe in continuous transfer mode, the data of which the size is a positive integral multiple of the maximum packet size and less than the buffer size.

When the maximum packet size of data is written to the pipe in non-continuous transfer mode, this module sets this bit to switch the FIFO buffer from the CPU side to the SIE side and enable data sending.

When this bit and the BCLR bit are set at the same time when the specified pipe is in the sending direction, this module clears the data that has been written so far and enables zero-length packets to be sent.

Setting this bit must be done only when the FRDY bit in the corresponding port control register is 1. If you want to check the value of the FRDY bit after setting this bit, wait at least 80 ns after setting this bit, and then reference the FRDY bit. Do not set this bit when the specified pipe is in the receiving direction.

(2) CPU buffer clear (BCLR) bit

When this bit is set, this module clears the FIFO buffer on the CPU side among the FIFO buffers allocated to the specified pipe.

Even if the two FIFO buffers in a double-buffer configuration are allocated to the specified pipe and both buffers can be read, this module clears only one of the two buffers.

If this bit is set when the specified pipe is the DCP, this module clears the FIFO buffer regardless of whether the FIFO buffer is on the CPU side or it is on the SIE side. To clear the buffer on the SIE side, set this bit always after setting the PID bits for DCP to “NAK”.

When the specified pipe is other than the DCP, setting this bit must be done only when the FRDY bit in the corresponding port control register is 1. If you want to check the value of the FRDY bit after setting this bit, wait at least 80 ns after setting this bit, and then reference the FRDY bit.

(3) FIFO port ready (FRDY) bit

This bit indicates whether the FIFO port can be accessed from the CPU. This bit is operated by this module.

In the cases described below, even when this module sets this bit, data cannot be read from the FIFO port because the FIFO buffer does not contain data to be read. In these cases, set the BCLR bit to clear the FIFO buffer to enable the next sending and receiving of data.

(1) A zero-length packet has been received while the FIFO buffer allocated to the specified pipe is empty.

(2) A short packet has been received and data reading has ended while the BFRE bit is 1.

(4) Receive data length (DTLN) bits

These bits indicate the length of receive data. These bits are operated by this module. During reading of the FIFO buffer, the value of these bits varies depending on the value of the RCNT bit as described below.

(1) When the RCNT bit is 0:
This module indicates a receive data length by using these bits until the CPU ends reading all received data from one FIFO buffer.
When the BFRE bit is 1, this module retains the receive data length until the BCLR bit is set even if reading of received data has ended.

(2) When the RCNT bit is 1:
This module decrements the value of these bits each time the CPU reads data.
(The value is decremented by 1 when the MBW bits are 00, by 2 when the MBW bits are 01 or by 4 when the MBW bits are 10.)

When the CPU ends reading from one FIFO buffer, this module clears these bits. If reading of one of the FIFO buffers in a double-buffer configuration, however, ends before reading of received data from the other FIFO buffer ends, these bits indicate the receive data length for the other FIFO buffer at the end of reading from the FIFO buffer of which reading ends earlier.

When these bits are read during reading of the FIFO buffer when the RCNT bit is 1, this module updates the value of these bits within 150 ns after a cycle of read access to the corresponding FIFO port.

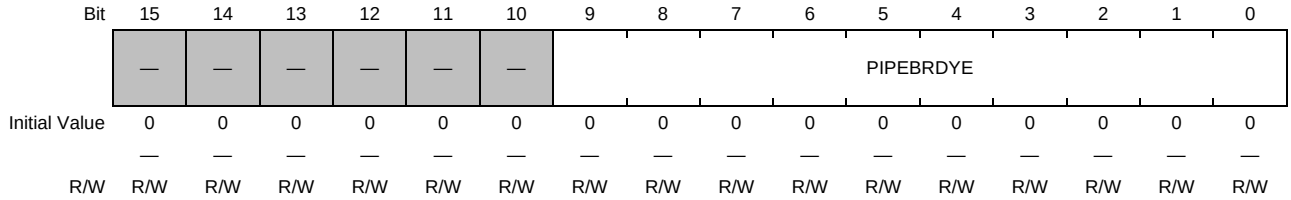
32B.2.8 Interrupt Enable Registers (INTENBx, BRDYENB, NRDYENB, BEMPENB)

32B.2.8.1 Interrupt Enable Register 0 [INTENB0] <Address: H'030>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBSE	RSME	SOFE	DVSE	CTRE	BEMPE	NRDYE	BRDYE	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	VBSE	0	R/W	VBUS Interrupt Enable This bit prohibits or enables a USB interrupt when a VBINT interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
14	RSME	0	R/W	Frame number update interrupt enable Resume interrupt enable This bit prohibits or enables a USB interrupt when a RESM interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
13	SOFE	0	R/W	This bit prohibits or enables a USB interrupt when a SOF interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
12	DVSE	0	R/W	Device state transition interrupt enable This bit prohibits or enables a USB interrupt when a DVST interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
11	CTRE	0	R/W	Control transfer stage transition interrupt enable This bit prohibits or enables a USB interrupt when a CTRT interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
10	BEMPE	0	R/W	Buffer empty interrupt enable This bit prohibits or enables a USB interrupt when a BEMP interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
9	NRDYE	0	R/W	Buffer not ready response interrupt enable This bit prohibits or enables a USB interrupt when an NRDY interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
8	BRDYE	0	R/W	Buffer ready interrupt enable This bit prohibits or enables a USB interrupt when a BRDY interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
7 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

32B.2.8.2 BRDY Interrupt Enable Register [BRDYENB] <Address: H'036>



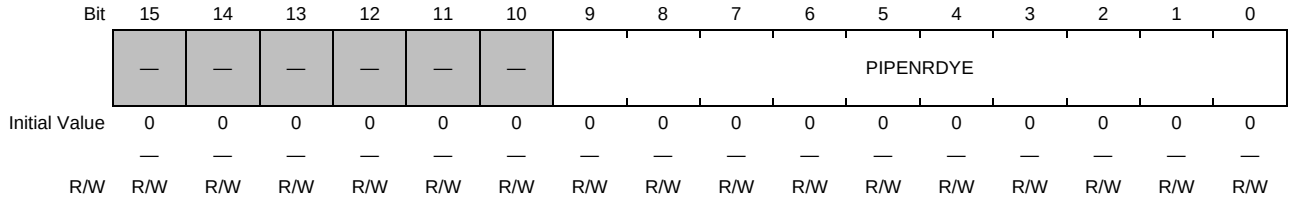
Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R/W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
9 to 0	PIPEBRDYE	All 0	R/W	Pipe BRDY interrupt enable These bits prohibit or enable the BRDY bit to be set when a BRDY interrupt to a pipe is detected. 0: Interrupt output prohibited 1: Interrupt output enabled

(1) Pipe BRDY interrupt enable (PIPEBRDYE) bits

When this module detects a BRDY interrupt to the pipe for which is set 1 in this register, this module sets the corresponding PIPEBRDY bit in the BRDYSTS register, sets the BRDY bit in the INTSTS0 register, and asserts the interrupt.

When at least one of the PIPEBRDY bits in the BRDYSTS register is 1 and software changes the corresponding interrupts enable bit in this register changes from 0 to 1, this module asserts the interrupt.

32B.2.8.3 NRDY Interrupt Enable Register [NRDYENB] <Address: H'038>



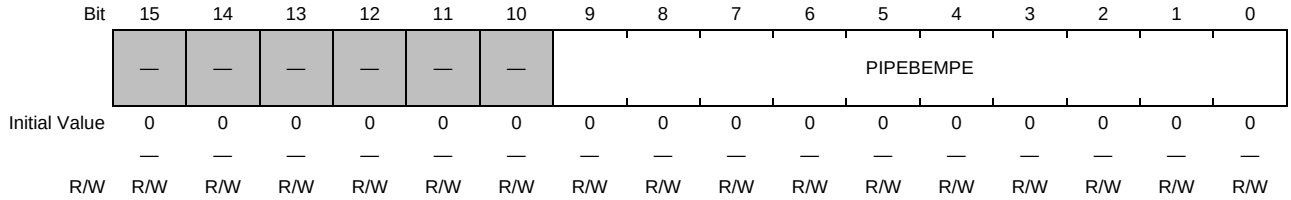
Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R/W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
9 to 0	PIPENRDYE	All 0	R/W	Pipe NRDY interrupt enable These bits prohibit or enable the NRDY bit to be set when a BRDY interrupt to a pipe is detected. 0: Interrupt output prohibited 1: Interrupt output enabled

(1) Pipe NRDY interrupt enable (PIPENRDYE) bits

When this module detects a BRDY interrupt to the pipe for which is set 1 in this register, this module sets the corresponding PIPENRDY bit in the NRDYSTS register, sets the NRDY bit in the INTSTS0 register, and asserts the interrupt.

When at least one of the PIPENRDY bits in the NRDYSTS register is 1 and the corresponding interrupts enable bit in this register changes from 0 to 1, this module asserts the interrupt.

32B.2.8.4 BEMP Interrupt Enable Register [BEMPENB] <Address: H'03A>



Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R/W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
9 to 0	PIPEBEMPE	All 0	R/W	Pipe BEMP interrupt enable These bits prohibit or enable the BEMP bit to be set when a BRDY interrupt to a pipe is detected 0: Interrupt output prohibited 1: Interrupt output enabled

Note: Bit numbers correspond to pipe numbers.

(1) Pipe BEMP interrupt enable (PIPEBEMPE) bits

When this module detects a BEMP interrupt to the pipe for which is set 1 in this register, this module sets the corresponding PIPEBEMP bit in the BEMPSTS register, sets the BEMP bit in the INTSTS0 register, and asserts the interrupt.

When at least one of the PIPEBEMP bits in the BEMPSTS register is 1 and the corresponding interrupts enable bit in this register changes from 0 to 1, this module asserts the interrupt.

32B.2.9 SOF Control Register

32B.2.9.1 SOF Pin Configuration Register [SOFCFG] <Address: H'03C>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	BRDYM	—	—	SOFM		—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
6	BRDYM	0	R/W	PIPEBRDY interrupt status clear timing This bit specifies the timing at which the PIPEBRDY interrupt is to be cleared. 0: Software clears the status. 1: Hardware clears the status by reading from or writing to the FIFO buffer. This bit can be set only during initialization (before communication). The setting cannot be changed after communication.
5, 4	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
3, 2	SOFM	All 0	R/W	SOF function setting These bits are used to select SOF pulse output mode. 00: SOF output disabled 01: SOF output in units of 1 ms 10: µSOF output in units of 125 µs 11: Reserved
1, 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

32B.2.10 Interrupt Status

32B.2.10.1 Interrupt Status Register 0 [INTSTS0] <Address: H'040>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBINT	RESM	SOFR	DVST	CTRT	BEMP	NRDY	BRDY	VBSTS	DVSQ			VALID	CTSQ		
Initial Value	0	0	0	0	0	0	0	0	—	0	0	0	0	0	0	0
	—	—	—	1	—	—	—	—	—	0	0	1	—	—	—	—
R/W	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R	R	R	R	R	R	R	R/W(0)	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	VBINT	0	R/W(0)	Change detection interrupt status This bit indicates the VBUS change detection interrupt status. 0: No VBUS interrupt is generated. 1: A VBUS interrupt is generated.
14	RESM	0	R/W(0)	Resume interrupt status This bit indicates the resume detection interrupt status. 0: No resume interrupt is generated. 1: A resume interrupt is generated.
13	SOFR	0	R/W(0)	Frame number update interrupt status This bit indicates the frame number update interrupt status. 0: No SOF interrupt is generated. 1: A SOF interrupt is generated.
12	DVST	0	R/W(0)	Device state transition interrupt status This bit indicates the device state transition interrupt. 0: No device state transition interrupt is generated. 1: A device state transition interrupt is generated.
11	CTRT	0	R/W(0)	Control transfer stage transition interrupt status This bit indicates the status of a control transfer stage transition interrupt. 0: No control transfer stage transition interrupt is generated. 1: A control transfer stage transition interrupt is generated.
10	BEMP	0	R	BEMP interrupt status This bit indicates the BEMP interrupt status. 0: No BEMP interrupt is generated. 1: A BEMP interrupt is generated.
9	NRDY	0	R	NRDY interrupt status This bit indicates the NRDY interrupt status. 0: No NRDY interrupt is generated. 1: An NRDY interrupt is generated.
8	BRDY	0	R	BRDY interrupt status This bit indicates the BRDY interrupt status. 0: No BRDY interrupt is generated. 1: A BRDY interrupt is generated.
7	VBSTS	—	R	VBUS input status This bit indicates the VBUS pin input status. 0: The VBUS pin is at the low level. 1: The VBUS pin is at the high level.

Bit	Bit Name	Initial Value	R/W	Description
6 to 4	DVSQ	All 0	R	Device state These bits indicate the device state. 000: Powered state 001: Default state 010: Address state 011: Configured state 1xx: Suspended state
3	VALID	0	R/W(0)	USB request reception This bit indicates whether USB request reception is detected. 0: Not detected 1: A setup packet is received.
2 to 0	CTSQ	All 0	R	Control transfer stage These bits indicate the control transfer stage. 000: Idle or setup stage 001: Control reading data stage 010: Control reading status stage 011: Control writing data stage 100: Control writing status stage 101: Control writing (No Data) status stage 110: Control transfer sequence error 111: Reserved

Note: When you want to clear the status indicated by the VBINT, RESM, SOFR, DVST, or CTRT bit, write 0 to only the bit to be cleared and 1 to other bits. Do not write 0 to any status bit that is currently 0.

Note: This module detects a status change indicated by the VBINT or RESM bit in this register even while the clock is stopped (the SUSPM bit is 0), and reports an interrupt corresponding to the status bit if the interrupt is enabled. Clearing the interrupt status must be done after the clock enabled.

(1) VBUS change interrupt status (VBINT) bit

This module sets this bit when it detects a change of the level of the VBUS pin input (from the high level to low level, or vice versa). This module indicates the level of the VBUS pin input by the VBSTS bit. When a VBINT interrupt occurs, read the VBSTS bit several times to check for consistency and remove chattering.

(2) Resume interrupt status (RESM) bit

This module sets this bit when it is in the suspended state (DVSQ bits are 1XXb) and detects a falling edge of the signal at the DP pin.

(3) Frame number update interrupt status (SOFR) bit

This module sets this bit under the following conditions:

This module sets this bit when the frame number is updated. (The frame number update interrupt is monitored at intervals of 1 ms.)

This module detects an SOFR interrupt based on SOF interpolation even when an SOF packet from the USB host is damaged.

(4) Device state transition interrupt status (DVST) bit

When this module detects a change of the device state, this module updates the value of the DVSQ bits, and sets this bit. When a device state transition interrupt occurs, clear the interrupt status before this module detects the next device state transition.

(5) Control transfer stage transition interrupt status (CTRT) bit

When this module detects a transition of control transfer stage, this module updates the value of the CTSQ bits and sets this bit.

When a control transfer stage transition interrupt occurs, clear the interrupt status before this module detects the next transition of control transfer stage.

(6) Buffer empty interrupt status (BEMP) bit

This module sets this bit when at least one of the PIPEBEMP bits in the BEMPSTS register corresponding to the pipes for which the PIPEBEMPE bit in the BEMPENB register is set (that is, when this module detects a BEMP interrupt to at least one of the pipes for which is enabled BEMP interrupt notification).

For the conditions to assert the PIPEBEMP status signal, see the description of the BEMPSTS register.

This module clears this bit when writes 0 to all the PIPEBEMP bits corresponding to the pipes for which a BEMP interrupt has been enabled by setting the PIPEBEMPE bit.

Cannot clear this bit even by writing 0.

(7) Buffer not-ready interrupt status (NRDY) bit

This module sets this bit when at least one of the PIPENRDY bits in the BNRDYSTS register corresponding to the pipes for which the PIPENRDYE bit in the NRDYENB register is set (that is, when this module detects an NRDY interrupt to at least one of the pipes for which is enabled NRDY interrupt notification).

For the conditions to assert the PIPENRDY status signal, see the description of the NRDYSTS register.

This module clears this bit when writes 0 to all the PIPENRDY bits corresponding to the pipes for which a NRDY interrupt has been enabled by setting the PIPENRDYE bit.

Software cannot clear this bit even by writing 0.

(8) Buffer ready interrupt status (BRDY) bit

This module sets this bit when at least one of the PIPEBRDY bits in the BRDYSTS register corresponding to the pipes for which the PIPEBRDYE bit in the BRDYENB register is set (that is, when this module detects an BRDY interrupt to at least one of the pipes for which is enabled BRDY interrupt notification).

For the conditions to assert the PIPEBRDY status signal, see the description of the BRDYSTS register.

This module clears this bit when writes 0 to all the PIPEBRDY bits corresponding to the pipes for which a BRDY interrupt has been enabled by setting the PIPEBRDYE bit.

Software cannot clear this bit even by writing 0.

32B.2.10.2 BRDY Interrupt Status Register [BRDYSTS] <Address: H'046>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPEBRDY									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	— R/W	— R/W	— R/W	— R/W	— R/W	— R/W	— R/W(0)	— R/W(0)	— R/W(0)	— R/W(0)	— R/W(0)	— R/W(0)	— R/W(0)	— R/W(0)	— R/W(0)	— R/W(0)

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R/W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
9 to 0	PIPEBRDY	All 0	R/W(0)	Pipe BRDY interrupt status These bits indicate the BRDY interrupt status of each pipe. 0: No interrupt is generated. 1: An interrupt is generated.

Note: Bit numbers correspond to pipe numbers.

Note: To clear the interrupt status indicated by a bit in this register when the BRDYM bit is 0, write 0 to only the bit to be cleared and 1 to other bit.

Note: When the BRDYM bit is 0, clearing the BRDY interrupt status must be done always before the next access to the FIFO.

(1) Pipe BRDY interrupt status (PIPEBRDY) bit

When this module detects a BRDY interrupt to a pipe, this module sets the corresponding PIPEBRDY bit in the BRDYSTS register. At that time, if already set the corresponding bit in the BRDYENB register, this module sets the BRDY bit in the INTSTS0 register, and asserts the interrupt.

Conditions to generate and clear a BRDY interrupt vary depending on the values of the BRDYM bit, and the BFRE bit for each pipe.

(a) BRDYM = 0 and BFRE = 0

When the BRDYM and BFRE bits are 0, the BRDY interrupt is generated to indicate that the FIFO port has become ready for access.

Under the conditions described below, this module generates an internal BRDY interrupt request trigger, and sets the PIPEBRDY bit corresponding to the pipe for which the request trigger is generated.

1. For the pipe in the sending direction

(a) When the DIR bit changes from 0 to 1

(b) When this module has ended sending packets through a pipe when data writing by the CPU to the FIFO buffer allocated to that pipe is disabled (when the value read from the BSTS bit is 0)
In continuous transfer mode, a request trigger is generated when data has all been sent from one FIFO buffer.

(c) When, in a double-buffer configuration, one FIFO buffer is empty when writing to the other FIFO buffer has ended
If sending to one FIFO buffer has ended during writing to the other FIFO buffer, no request trigger is generated until the ongoing writing to the other FIFO buffer ends.

(d) When this module flushes the FIFO buffer allocated to the pipe of which the transfer type is isochronous.

(e) When the state of the FIFO buffer is changed from the write-disabled state to the write-enabled state by writing 1 to the ACLRM bit

No request trigger is generated when the pipe is the DCP (in other words, when data is sent by a control transfer).

2. For the pipe in the receiving direction

(a) When this module has ended receiving packets through a pipe and reading from the FIFO buffer is enabled when data reading by the CPU from the FIFO buffer allocated to that pipe is disabled (when the value read from the BSTS bit is 0)
No request trigger is generated for the transaction that involves a data PID mismatch.
In continuous transmission/reception mode, no request trigger is generated when the data size is the maximum packet size and the FIFO buffer still has a free space.
If a short packet is received, a request trigger is generated even when the FIFO buffer has a free space.
When the transaction counter is used, a request trigger is generated when the specified number of packets have been received.
In that case, the request trigger is generated even if the FIFO buffer has a free space.

(b) When, in a double-buffer configuration, one FIFO buffer is in the read-enabled state when reading from the other FIFO buffer has ended
If receiving from one FIFO buffer has ended during reading from the other FIFO buffer, no request trigger is generated until the ongoing reading from the other FIFO buffer ends.

This interrupt does not occur during the communication at the status stage of a control transfer.

The PIPEBRDY interrupt status of the corresponding pipe can be cleared to "0" by writing "0" to the corresponding bit. When clearing a PIPEBRDY bit by writing 0, write 1 to all the PIPEBRDY bits corresponding to other pipes.

Clearing the pipe BRDT interrupt status must be done always before the next access to the FIFO buffer.

(b) When BRDYM = 0 and BFRE = 1

When the BRDYM bit is 0 and the BFRE bit is 1, this module determines that a BRDY interrupt occurs when all the data for a transfer has been read through a receiving pipe, and sets the PIPEBRDY bit corresponding to the pipe.

This module determines that the last data in a transfer has been received when one of the following conditions is met:

1. A short packet or a zero-length packet has been received.
2. The transaction counter (TRNCNT bits) is used, and as many packets as the value of the TRNCNT bits have been received.

When one of the above conditions is met and reading of the relevant data has ended, this module determines that all the data in a transfer has been read.

If a zero-length packet is received when the FIFO buffer is empty, this module determines that all the data in a transfer has been read when the FRDY bit is set and the DTLN bits are cleared in the corresponding FIFO Port Control Register. To start the next transfer in that case, write 1 to the BCLR bit in the corresponding FIFOCTR.

When the BRDYM bit is 0 and the BFRE bit is 1, this module does not detect any BRDY interrupt to the pipe in the sending direction.

The PIPEBRDY interrupt status of the corresponding pipe can be cleared to “0” by writing “0” to the corresponding bit. When clearing a PIPEBRDY bit by writing 0, write 1 to the PIPEBRDY bits corresponding to other pipes.

In this mode, do not change the value of the BFRE bit until all the processing for a transfer ends.

If you need to change the value of the BFRE bit during the transfer, set the ACLRM bit to clear all the FIFO buffers for the specified pipe.

(c) When BRDYM = 1 and BFRE = 0

When the BRDYM bit is 1 and the BFRE bit is 0, the values of individual PIPEBRDY bits interlock with the values of the BSTS bits for individual pipes. In other words, this module sets or clears the BRDY interrupt status of a pipe according to the state of the FIFO buffer allocated to the pipe.

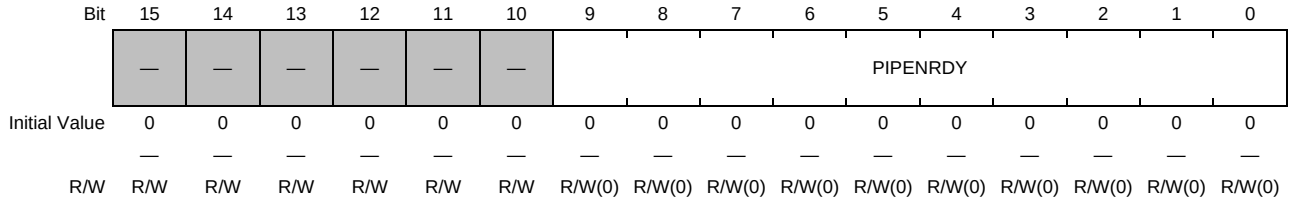
(a) For the pipe in the sending direction
This module sets the PIPEBRDY bit for the pipe when data can be written to the FIFO port or clears the PIPEBRDY bit when data cannot be written to the FIFO port.
The BRDY interrupt signal, however, is not asserted even when the sending pipe is write-enabled if the pipe is the DCP.

(b) For the pipe in the receiving direction
This module sets the PIPEBRDY bit for the pipe when data can be read from the FIFO port or clears the PIPEBRDY bit when all data has been read (that is, when data reading from the FIFO port is disabled).
If a zero-length packet is received when the FIFO buffer is empty, the PIPEBRDY bit corresponding to the specified pipe is kept being set and the BRDY interrupt signal is kept being asserted until sets the BCLR bit.

When the BRDYM bit is 1 and the BFRE bit is 0, this module cannot clear any PIPEBRDY bit.

When the BRDYM bit is 1, all the BFRE bits (for all pipes) must be 0.

32B.2.10.3 NRDY Interrupt Status Register [NRDYSTS] <Address: H'048>



Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R/W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
9 to 0	PIPENRDY	All 0	R/W(0)	Pipe NRDY interrupt status These bits indicate the NRDY interrupt status of each pipe. 0: No interrupt is generated. 1: An interrupt is generated.

Note: Bit numbers correspond to pipe numbers.

Note: To clear the interrupt status indicated by a bit in this register, write 0 only to the bit and 1 to all other bit.

(1) Pipe NRDY interrupt status (PIPENRDY) bit

When this module issues an internal NRDY interrupt request for a pipe of which the PID is set to BUF, this module sets the PIPENRDY bit corresponding to the pipe in the NRDYSTS register. At that time, if the NRDYENB register bit corresponding to the pipe is set, this module sets the NRDY bit in the INTSTS0 register, and asserts the interrupt.

This module issues an internal NRDY interrupt request for individual pipes under the conditions described below. This module does not issue any interrupt request at the status stage of a control transfer.

(a) For the pipe in the sending direction

(1) When an IN token is received in a situation where the PID bits corresponding to the specified pipe are 01 (BUF) and the FIFO buffer does not contain send data
When an IN token is received, this module issues an NRDY interrupt request, and sets the PIPENRDY bit.
If the transfer type of the pipe for which the interrupt is generated is isochronous transfer, this module sends a zero- length packet, and sets the OVRN bit.

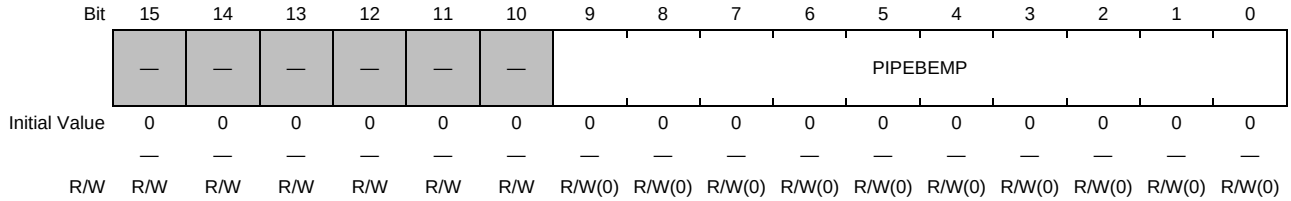
(b) For the pipe in the receiving direction

(1) When an OUT token is received in a situation where the PID bits corresponding to the specified pipe are 01 (BUF) and the FIFO buffer is full
If the transfer type of the pipe for which the interrupt is generated is isochronous transfer, this module issues an NRDY interrupt request, sets the PIPENRDY bit, and sets the OVRN bit.
If the transfer type of the pipe for which the interrupt is generated is not isochronous transfer, this module issues an NRDY interrupt request when sending an NAK Handshake signal after receiving the data that follows the OUT token, and sets the PIPENRDY bit.
Note, however, that this module does not issue an NRDY interrupt request when resending data (when a DATA-PID mismatch has occurred).
This module does not issue an NRDY interrupt request also when an error has occurred in a data packet.

(2) When a PING token is received in a situation where the PID bits corresponding to the specified pipe are 01 (BUF) and the FIFO buffer is full
When a PING token is received, this module issues an NRDY interrupt request, and sets the PIPENRDY bit.

(3) When the transfer type of the specified pipe is isochronous transfer, the PID bits corresponding to the specified pipe are 01 (BUF), and data has not been received normally within an interval frame
When an SOF token is received, this module issues an NRDY interrupt request, and sets the PIPENRDY bit for the specified pipe.

32B.2.10.4 BEMP Interrupt Status Register [BEMPSTS] <Address: H'04A>



Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R/W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
9 to 0	PIPEBEMP	All 0	R/W(0)	Pipe BEMP interrupt status These bits indicate the BEMP interrupt status of each pipe. 0: No interrupt is generated. 1: An interrupt is generated.

Note: Bit numbers correspond to pipe numbers.

Note: To clear the interrupt status indicated by a bit in this register, write 0 only to the bit and 1 to all other bit.

(1) Pipe BEMP interrupt status (PIPEBEMP) bit

When this module detects a BEMP interrupt to a pipe of which the PID is set to BUF, this module sets the PIPEBEMP bit corresponding to the pipe in the BEMPSTS register. At that time, if the BEMPENB register bit corresponding to the pipe is set, this module sets the BEMP bit in the INTSTS0 register, and asserts the interrupt.

This module issues an internal BEMP interrupt request for individual pipes under the conditions described below.

1. When data sending (including sending of zero-length packets) to a pipe in the sending direction has ended and the FIFO buffer allocated to the pipe is empty
   In a single-buffer configuration, this module generates a BRDY interrupt at the same time as issuing an internal BEMP interrupt request for the pipes other than the DCP.
   Note, however, that this module does not issue the internal BEMP interrupt request in the following cases:

(a) When software (DMAC) has already started writing to the CPU-side FIFO buffer, in a double-buffer configuration, when sending data for one buffer ends

(b) When the buffer is cleared (to empty the buffer) by writing 1 to the ACLRM or BCLR bit

(c) During an IN transfer (sending zero-length packets) at the status stage of a control transfer

2. For the pipe in the receiving direction
   When the data larger than the specified maximum packet size has been received normally
   In that case, this module issues a BEMP interrupt request, sets the PIPEBEMP bit for the specified pipe, discards the received data, and changes the value of the PID bits for the specified pipe to 11 (STALL).
   Then, this module returns a STALL packet.
   Note, however, that this module does not issue the internal BEMP interrupt request in the following cases:

(a) When a CRC or bit stuff error has been detected in the received data

(b) When a SETUP transaction is being executed

Writing 0 to this bit clears the interrupt status.

Writing 1 to this bit causes no effect.

32B.2.11 Frame Number Registers (FRMNUM, UFRMNUM)

32B.2.11.1 Frame Number Register [FRMNUM] <Address: H'04C>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OVRN	CRCE	—	—	—	FRNM										
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W(0)	R/W(0)	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	OVRN	0	R/W(0)	Overrun/underrun detection status This bit indicates whether an overrun or underrun is detected in the pipe being used to perform an isochronous transfer. 0: No error 1: An error occurred.
14	CRCE	0	R/W(0)	CRC error detection status This bit indicates the CRC error detection status for the pipe being used to perform an isochronous transfer. 0: No error 1: An error occurred.
13 to 11	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
10 to 0	FRNM	All 0	R	Frame number These bits indicate the latest frame number.

Note: The OVRN bit is intended for debugging. When designing a system, design transfer timings appropriately to prevent buffer overruns and underruns.

(1) Overrun/underrun detection status (OVRN) bit

This module sets this bit when this module detects an overrun or underrun in a pipe of which the transfer type is isochronous transfer.

When this module detects an overrun or underrun, this module issues an internal NRDY interrupt request. For details, see **Section 32B.2.10.3(1), Pipe NRDY interrupt status (PIPENRDY) bit**.

Software can clear this bit by writing 0 to this bit. If the CRCE bit should not be cleared together when this bit is cleared, write H'40.

When the peripheral controller function is selected

This module sets this bit in the following cases:

1. When an IN token is received although writing send data to the FIFO has not ended, in the case of a pipe that is in the sending direction and performs an isochronous transfer
2. When an OUT token is received although the free space of FIFO buffer is less than the size of one FIFO buffer, in the case of a pipe that is in the sending direction and performs an isochronous transfer

(2) CRC error detection status (CRCE) bit

This module sets this bit when this module detects a CRC or bit stuff error in a pipe of which the transfer type is isochronous transfer.

Software can clear this bit by writing 0 to this bit. If the OVRN bit should not be cleared together when this bit is cleared, write H’80.

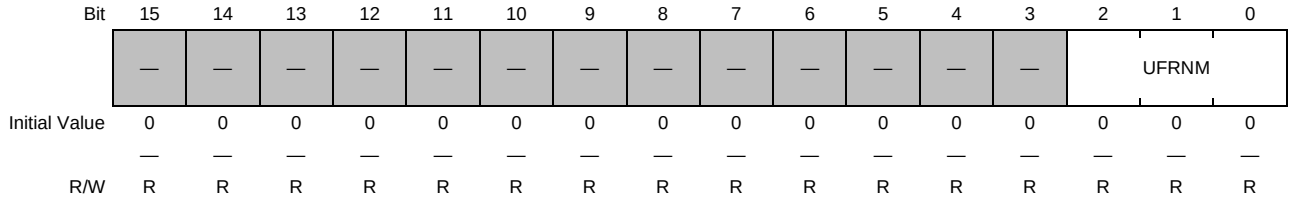
When this module detects a CRC error, this module issues an internal NRDY interrupt request. For details, see **Section 32B.2.10.3(1), Pipe NRDY interrupt status (PIPENRDY) bit**.

(3) Frame number (FRNM) bits

This module updates the value of these bits each time an SOF packet is received (once per 1 ms), and indicates the latest frame number in these bits.

When reading these bits, read them twice and check for consistency.

32B.2.11.2 Micro Frame Number Register [UFRMNUM] <Address: H'04E>



Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Nothing is assigned to these bits. Fix these bits to 0.
14 to 3	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
2 to 0	UFRNM	All 0	R	Micro frame These bits indicate the micro frame number.

(1) Micro frame number (UFRNM) bit

In high-speed transfer mode, this module writes the micro frame number to these bits. In a mode other than high-speed transfer mode, this module writes H'00 to these bits.

When reading these bits, read them twice and check for consistency.

32B.2.12 USB Address

32B.2.12.1 USB Address Register [USBADDR] <Address: H'050>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	USBADDR						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
10 to 8	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
7	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
6 to 0	USBADDR	All 0	R	USB address These bits indicate the USB address allocated by the host.

(1) USB address (USBADDR) bits

When this module has received and normally processed a SetAddress request, this module writes the received USB address to these bits.

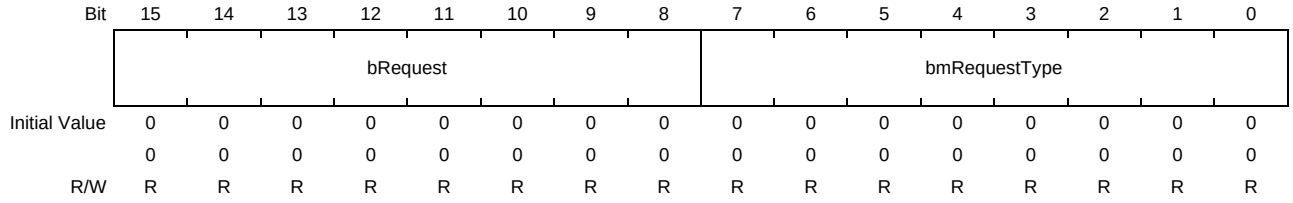
When this module detects a USB bus reset, this module writes H'00 to these bits.

32B.2.13 USB Request Registers

USB request registers are used to store control transfer setup requests.

These registers store the values set in the received USB requests.

32B.2.13.1 USB Request Type Register [USBREQ] <Address: H'054>



Bit	Bit Name	Initial Value	R/W	Description
15 to 8	bRequest	All 0	R	Request Value of USBRequestbRequest
7 to 0	bmRequestType	All 0	R	Request type Value of USBRequestbmRequestType

(1) USB request (bRequest) bits

These bits indicate the value of the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

(2) USB request type (bRmRequestType) bits

These bits indicate the value of the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

32B.2.13.2 USB Request Value Register [USBVAL] <Address: H'056>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	wValue															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	wValue	All 0	R	Value Value of the wValue field in a USB request

(1) Value (wValue) bits

These bits indicate the value of the wValue field in a USB request.

These bits indicate the value of wValue field in the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

32B.2.13.3 USB Request Index Register [USBINDX] <Address: H'058>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	wIndex															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	wIndex	All 0	R	Index Value of the wIndex field in a USB request

(1) Index (wIndex) bits

These bits indicate the value of the wIndex field in a USB request.

These bits indicate the value of wIndex field in the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

32B.2.13.4 USB Request Length Register [USBLENG] <Address: H'05A>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	wLength															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	wLength	All 0	R	Length Value of the wLength field in a USB request

(1) Length (wLength) bits

These bits indicate the value of the wLength field in a USB request.

These bits indicate the value of wLength field in the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

32B.2.14 DCP Configuration

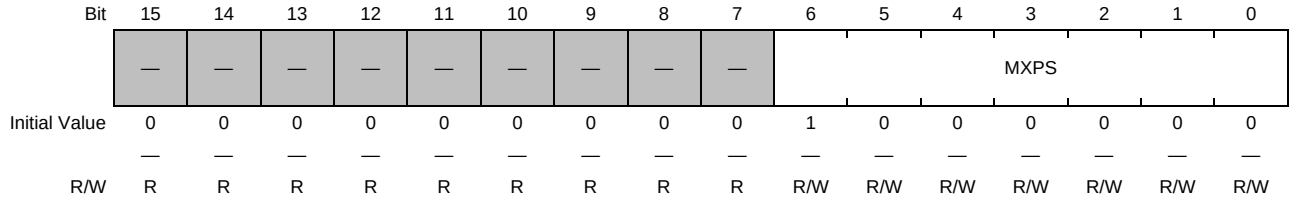
When performing a data communication by a control transfer, use the default control pipe (DCP).

32B.2.14.1 DCP Configuration Register [DCPCFG] <Address: H'05C>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CNTMD	SHTNAK	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	— R	— R	— R	— R	— R	— R	— R	— R/W	— R/W	— R	— R	— R	— R	— R	— R	— R

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
8	CNTMD	0	R/W	Continuous transfer mode This bit specifies whether to use the default control pipe to perform communication in continuous transfer mode. 0: Non-continuous transfer mode 1: Continuous transfer mode
7	SHTNAK	0	R/W	Pipe disable at transfer end When the default control pipe is in the receiving direction, this bit specifies whether to change the PID setting to NAK when the transfer ends. 0: Continues the pipe when the transfer ends 1: Disables the pipe when the transfer ends
6 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

32B.2.14.2 DCP Max. Packet Size Register [DCPMAXP] <Address: H’05E>



Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
6 to 0	MXPS	H’40	R/W	Maximum packet size These bits specify the maximum data payload size (maximum packet size) for the DCP.

(1) Maximum packet size (MXPS) bits

These bits are used to specify the maximum data payload size (maximum packet size) for the DCP.

The default is H’40 (64 bytes).

The value of the MXPS bits must conform to the USB Specification.

Writing a value to the MXPS bits must be done when the PID is set to NAK and no value is set in the CURPIPE bits. If you need to change the value of these bits after changing the PID setting for the specified pipe from BUF to NAK, check the PBUSY bit is 0 before changing the value of these bits. If, however, this module has changed the PID setting from BUF to NAK, the value of the PBUSY bit need not be checked.

When the MXPS bits are all 0, data must not be written to the FIFO buffer, and the PID setting must not be changed to BUF.

32B.2.14.3 DCP Control Register [DCPCTR] <Address: H'060>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	—	—	—	—	—	—	SQCLR	SQSET	SQMON	PBUSY	—	—	CCPL	PID	
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0
R/W	R	R	R	R	R	R	R	R(0)/W(1)	R(0)/W(1)	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	Buffer status This bit indicates the accessibility status of the DCP FIFO buffer. 0: Buffer access is not possible. 1: Buffer access is possible.
14 to 9	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
8	SQCLR	0	R(0)/W(1)	Toggle bit clear This bit can set DATA0 as the expected value of the sequence toggle bit for the next transaction in DCP transfer. 0: Writing disabled 1: DATA0 specified
7	SQSET	0	R(0)/W(1)	Toggle bit set This bit can set DATA1 as the expected value of the sequence toggle bit for the next transaction in DCP transfer. 0: Writing disabled 1: DATA1 specified
6	SQMON	1	R	Sequence toggle bit monitor This bit indicates the expected value of the sequence toggle bit for the next transaction in DCP transfer. 0: DATA0 1: DATA1
5	PBUSY	0	R	Pipe busy This bit indicates whether the specified pipe is being used in the USB bus. 0: The specified pipe is not used in the USB bus. 1: The specified pipe is being used in the USB bus.
4 to 3	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
2	CCPL	0	R/W	Control transfer end enable Setting this bit permits the status stage of the control transfer to end. 0: Does not permit the control transfer to end. 1: Permits the control transfer to end.
1, 0	PID	All 0	R/W	Response PID These bits control responses from this module in control transfer. 00: NAK response 01: BUF response (according to the buffer status) 10: STALL response 11: STALL response

(1) Buffer status (BSTS) bit

This bit indicates whether the CPU can access the FIFO buffer allocated to the DCP. This bit is operated by this module. The meaning of this bit varies as follows depending on the value of the ISEL bit:

1. When ISEL = 0: This bit indicates whether receive data can be read from the FIFO buffer.
2. When ISEL = 1: This bit indicates whether send data can be written to the FIFO buffer.

(2) Sequence toggle bit clear (SQCLR) bit

When software sets this bit, this module specifies DATA0 as the expected value of the sequence toggle bit for the specified pipe. This module always clears this bit.

Do not set the SQCLR and SQSET bits at the same time.

Setting this bit must be done when the PID is set to NAK and no value is set in the CURPIPE bits.

If you need to set this bit after changing the PID setting for the specified pipe from BUF to NAK, check, that the PBUSY bit is 0 before setting this bit. If, however, this module has changed the PID setting from BUF to NAK, the value of the PBUSY bit need not be checked.

(3) Sequence toggle bit set (SQSET) bit

When software sets this bit, this module specifies DATA1 as the expected value of the sequence toggle bit for the specified pipe. This module always clears this bit.

Do not set the SQCLR and SQSET bits at the same time.

Setting this bit must be done when the PID is set to NAK and no value is set in the CURPIPE bits.

If you need to set this bit after changing the PID setting for the specified pipe from BUF to NAK, check, that the PBUSY bit is 0 before setting this bit. If, however, this module has changed the PID setting from BUF to NAK, the value of the PBUSY bit need not be checked.

(4) Sequence toggle bit monitor (SQMON) bit

This bit indicates the expected value of the sequence toggle bit for the specified pipe. This bit is operated by this module. When a transaction ends normally, this module toggles this bit.

This module, however, does not toggle this bit if a DATA-PID mismatch occurs during a transfer in the receiving direction.

When a SETUP packet is received normally, this module sets this bit (to specify DATA1 as the expected value).

This module does not reference this bit in an IN or OUT transaction at the status stage. Also, this module does not toggle this bit even when the transaction ends normally.

(5) Pipe busy (PBUSY) bit

This module changes this bit from 0 to 1 when a USB transaction using the specified pipe starts. This module changes this bit from 1 to 0 when the transaction ends.

Reading this bit after software sets the PID to NAK enables you to check whether you can change pipe settings.

(6) Control transfer end enable (CCPL) bit

When software sets this bit when the PID of the specified pipe is BUF, this module ends the status stage of the ongoing control transfer.

In other words, in a control read transfer, this module sends an ACK handshake in response to an OUT transaction request from the USB host, and, in a control write or no-data control transfer, this module sends a zero-length packet in response to an IN transaction request from the USB host. If, however, a SetAddress request is detected, this module performs automatic response throughout the period from the setup stage to the end of the status stage regardless of the value of this bit.

When a new SETUP packet is received, this module changes this bit from 1 to 0.

When the VALID bit is 1, software cannot set this bit.

(7) Response PID (PID) bits

The setting of these bits must be changed from NAK to BUF when the data stage or status stage of a control transfer is executed.

This module changes the value of these bits in the following cases:

1. This module changes the value of these bits to 00 (NAK) when it receives a SETUP packet. At that time this module sets the VALID bit. Software cannot change the value of these bits until it clears the VALID bit.
2. When set these bits to 01 (BUF), this module changes the value of these bits to 11 (STALL) when it receives the data exceeding the specified maximum packet size.
3. This module changes the value of these bits to 1x (STALL) when it detects a sequence error in a control transfer.
4. This module changes the value of these bits to 00 (NAK) when it detects a USB reset.

This module does not reference these bits during SetAddress request processing (automatic processing).

32B.2.15 Pipe Configuration Registers (PIPESEL, PIPECFG, PIPEBUF, PIPEMAXP, PIPEPERI)

To configure pipes 1 to 15, use the PIPESEL, PIPECFG, PIPEBUF, PIPEMAXP, PIPEPERI, PIPExCTR, PIPExTRE, and PIPExTRN registers.

Select the pipes to be used by using the PIPESEL register, and then configure functions of individual pipes by using the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers. Note that you can use the PIPExCTR, PIPExTRE, and PIPExTRN registers for setting regardless of the pipe selection by the PIPESEL register.

32B.2.15.1 Pipe Window Select Register [PIPESEL] <Address: H'064>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PIPESEL			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
3 to 0	PIPESEL	All 0	R/W	Pipe window select These bits specify a pipe for registers at addresses H'68 to H'6E. 0000: No selection 0001: PIPE1 0010: PIPE2 0011: PIPE3 0100: PIPE4 0101: PIPE5 0110: PIPE6 0111: PIPE7 1000: PIPE8 1001: PIPE9

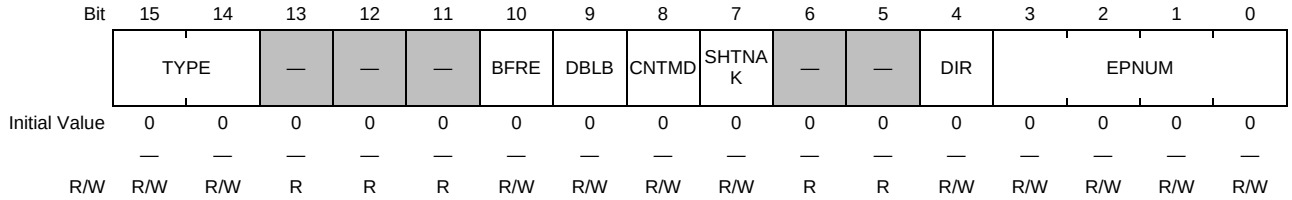
Note: When the PIPESEL bits are 0000, all bits of the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers are cleared. When the PIPESEL bits are 0000, writing to the registers at addresses H'68 to H'6E is ignored.

(1) Pipe Window select (PIPESEL) bits

When software a value from 0001 to 1111 to these bits, this module indicate the pipe information and settings corresponding to the registers at addresses H'68 to H'6E. After a pipe is selected by these bits, the values set in the areas at addresses H'68 to H'6E are applied, by this module, to the transfer operation using the selected pipe.

When writes 0000 to these bits, this module writes 0 to all bits of the registers at addresses H'68 to H'6E. Then, writing to the areas at addresses H'68 to H'6E is ignored.

32B.2.15.2 Pipe Configuration Register [PIPECFG] <Address: H'068>



Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TYPE		—	—	—	BFRE	DBLB	CNTMD	SHTNAK	—	—	DIR	EPNUM			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	TYPE	All 0	R/W	Transfer type These bits specify the transfer type of the pipe specified in the PIPESEL bit. 00: The pipe cannot be used. 01: Bulk transfer 10: Interrupt transfer 11: Isochronous transfer
13 to 11	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
10	BFRE	0	R/W	BRDY interrupt operation specification This bit specifies the timing at which this module notifies a BRDY interrupt relating to the specified pipe. 0: A BRDY interrupt is notified when data is sent or received. 1: A BRDY interrupt is notified when reading of data is completed.
9	DBLB	0	R/W	Double-buffer mode This bit specifies a single or double FIFO buffer to be used by the specified pipe. 0: Single buffer 1: Double buffer
8	CNTMD	0	R/W	Continuous transfer mode This bit specifies whether to use the specified pipe to perform communication in continuous transfer mode. 0: Non-continuous transfer mode 1: Continuous transfer mode
7	SHTNAK	0	R/W	Pipe disable at transfer end When the specified pipe is in the receiving direction, this bit specifies whether to change the PID setting to NAK when the transfer ends. 0: Continues the pipe when the transfer ends. 1: Disables the pipe when the transfer ends.
6, 5	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
4	DIR	0	R/W	Transfer direction This bit specifies the transfer direction of the specified pipe. 0: Receiving direction 1: Sending direction
3 to 0	EPNUM	All 0	R/W	Endpoint number These bits specify the endpoint number of the specified pipe.

(1) Transfer type (TYPE) bits

These bits are used to specify the USB transfer type of the pipe (selected pipe) specified in the PIPESEL bits.

Table 32B.12 lists pipes and the transfer types specifiable in these bits.

Table 32B.12 Selected Pipes and the Transfer Types Specifiable in the TYPE Bits

Selected Pipe	TYPE Bits	USB Transfer Type
PIPE1 or PIPE2	01 or 11	Bulk or isochronous transfer
PIPE3 to PIPE5	01	Bulk transfer
PIPE6 to PIPE9	10	Interrupt transfer

Always specify a value other than 00 in these bits for a selected pipe before setting the PID of the selected pipe to BUF (to start USB communication using the selected pipe).

The value of these bits for a selected pipe must be done while the PID of the selected pipe is NAK. If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(2) BRDY interrupt operation specification (BFRE) bit

This bit is valid when the selected PIPE is PIPE1 to PIPE5.

When set this bit and been using the selected pipe in the receiving direction (in other words, the DIR bit is 0), this module detects the end of transfer (when it occurs) and generates a BRDY interrupt when reading of the last packet ends.

If a BRDY interrupt occurs with the above settings, software must write 1 to the BCLR bit. The FIFO buffer allocated to the selected pipe remains unready for reception until 1 is written to the BCLR bit.

When set this bit and been using the selected pipe in the sending direction (in other words, the DIR bit is 1), this module does not generate any BRDY interrupt.

For details, see the description of the PIPEBRDY interrupt status bit.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(3) Double-buffer mode (DBLB) bit

This bit is valid when the selected pipe is PIPE1 to PIPE5.

When set this bit for a selected pipe, this module allocates, to the selected pipe, two FIFO buffers, each of which has the FIFO buffer size specified in the BUFSIZE bits in the PIPEBUF register.

The size of the FIFO buffer this module allocates to the selected pipe is as follows:

(BUFSIZE + 1) × 64 × (DBLB + 1) [bytes]

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(4) Continuous transfer mode (CNTMD) bit

This bit is valid when the selected pipe is PIPE1 to PIPE5, and the transfer type of the selected pipe is bulk transfer.

This module determines whether data sending from or receiving in the FIFO buffer allocated to the selected pipe has ended according to the value of this bit in the way described in **Table 32B.13**.

Table 32B.13 How to Determine the End of Data Sending from or Receiving in the FIFO Buffer According to the Value of the CNTMD Bit

CNTMD Bit Setting Value	How to Determine Whether Reading or Sending is Enabled
0	Condition for enabling reading from the FIFO buffer when the receiving direction is set ("DIR = 0"): This module receives one packet
	Condition for enabling sending from the FIFO buffer when the sending direction is set ("DIR = 1"): Either of the following conditions (1) and (2) is met: (1) Data for the maximum packet size is written to the FIFO buffer. (2) Data for the short packet (including the case of zero-byte data) is written to the FIFO buffer and then 1 is written in the BVAL bit.
1	Conditions for enabling reading from the FIFO buffer when the receiving direction is set ("DIR = 0"): (1) The number of bytes in data received in the FIFO buffer allocated to the selected pipe becomes equal to the number of allocated bytes ((BUFSIZE + 1) * 64). (2) This module receives a short packet other than a zero-length packet. (3) This controller receives a zero-length packet when the FIFO buffer allocated to the selected pipe already contains data. (4) Packets are received as many times as the value of the transaction counter set for the selected pipe.
	Condition for enabling sending from the FIFO buffer when the sending direction is set ("DIR = 1"): One of the following conditions (1) to (3) is met: (1) The amount of written data becomes equal to the size of one FIFO buffer allocated to the selected pipe. (2) Data for less than the size of one FIFO buffer allocated to the selected pipe (including the case of zero-byte data) is written to the FIFO buffer and then 1 is written in the BVAL bit. (3) Data for less than the size of one FIFO buffer allocated to the selected pipe (including the case of zero-byte data) is written to the FIFO buffer, and a transfer end signal is asserted at the same time of the last writing.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(5) PIPE disable at transfer end (SHTNAK) bit

This bit is valid when the selected pipe is PIPE1 to PIPE5 and is in the receiving direction.

When set this bit for a selected pipe in the receiving direction, this module changes the PID of the selected pipe to NAK when this module determines the end of data transfer to the selected pipe. This module determines the end of transfer when one of the following conditions (1) and (2) is met:

(1) This module has normally received short packet data (including zero-length packets).

(2) When using a transaction counter, this module has normally received as many packets as the value set in the transaction counter.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

For the pipes in the sending direction, this bit must be cleared.

(6) Transfer direction (DIR) bit

When writes 0 to this bit for a selected pipe, this module uses the selected pipe in the receiving direction. When software writes 1 to this bit for the selected pipe, this module uses the selected pipe in the sending direction.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(7) Endpoint number (EPNUM) bits

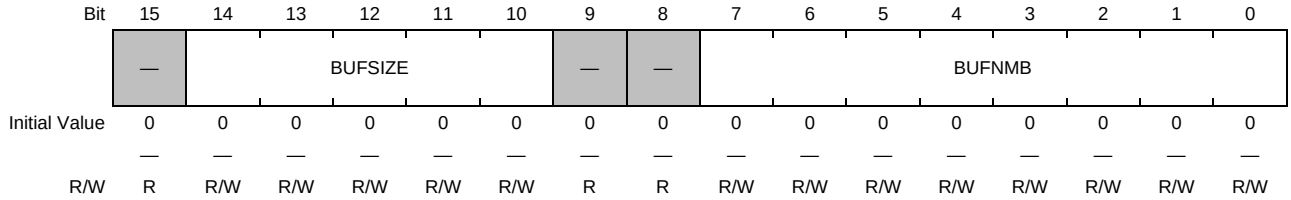
These bits are used to specify the endpoint number of the endpoint of a selected pipe. Note that specifying 0000 in these bits for a pipe means that the pipe is not used.

Changing the value of these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

The combination of the values of the DIR bit and EPNUM bits for a pipe must be unique among those for all pipes. (The setting "EPNUM = 000" [the selected pipe is not used] can be duplicated for multiple pipes.)

32B.2.15.3 Pipe Buffer Setting Register [PIPEBUF] <Address: H'06A>



Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
14 to 10	BUFSIZE	All 0	R/W	Buffer size These bits specify the size of the FIFO buffer for the pipe specified in the PIPESEL bit. H'00: 64 bytes H'01: 128 bytes ... (H'1F: 2 Kbytes)
9, 8	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
7 to 0	BUFNMB	All 0	R/W	Buffer number These bits specify the FIFO buffer number of the specified pipe. (H'4 to H'7F)

Note: Changing values of these register bits for a selected pipe must be done when sets the PID of the selected pipe to NAK, and specifies no pipe in the CURPIPE bits.

Note: If you change values of these register bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the values of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(1) Buffer size (BUFSIZE) bits

These bits are used to specify the size of the FIFO buffer to be allocated to the selected pipe. Specify the FIFO buffer size in units of blocks. One block has 64 bytes.

When set the DBLB bit for a selected pipe, this module allocates, to the selected pipe, two FIFO buffers, each of which has the FIFO buffer size specified in these bits.

The size of the FIFO buffer this module allocates to the selected pipe is as follows:

(BUFSIZE + 1) × 64 × (DBLB + 1) [bytes]

The following value can be specified in these bits:

(1) Any value from H'0 to H'1F when the selected pipe is PIPE1 to PIPE5.

(2) H'0 only when the selected pipe is PIPE6 to PIPE9.

In continuous transfer mode (CNTMD = 1), specify an integral multiple of the maximum packet size in the BUFSIZE bits.

(2) Buffer number (BUFNMB) bits

These bits are used to specify the block number of the first block in the FIFO buffer to be allocated to the selected pipe. This module allocates the following blocks of FIFO buffer to the selected pipe:

Block with block number “BUFNMB” to the block with block number “BUFNMB + (BUFSIZE + 1) × (DBLB + 1) - 1”

The value of these bits must be H’04 to H’7F. Note, however, that the following rules must be observed: Value " H’00" is exclusively used for DCP.

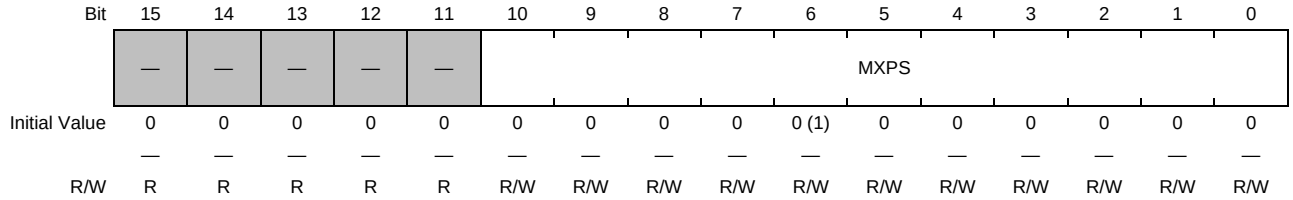
Value “H’04” is exclusively used for PIPE6. When, however, PIPE6 is not used, this value can be used for another pipe. If PIPE6 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets H’04 in the BUFNMB bits for PIPE6.

Value “H’05” is exclusively used for PIPE7. When, however, PIPE7 is not used, this value can be used for another pipe. If PIPE7 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets H’05 in the BUFNMB bits for PIPE7.

Value “H’06” is exclusively used for PIPE8. When, however, PIPE8 is not used, this value can be used for another pipe. If PIPE8 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets H’06 in the BUFNMB bits for PIPE8.

Value “H’07” is exclusively used for PIPE9. When, however, PIPE9 is not used, this value can be used for another pipe. If PIPE9 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets H’07 in the BUFNMB bits for PIPE9.

32B.2.15.4 Pipe Maximum Packet Size Register [PIPEMAXP] <Address: H’06C>



Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
10 to 0	MXPS	*Note	R/W	Maximum packet size These bits specify the maximum data payload size (maximum packet size) for the specified pipe. For PIPE6 to PIPE8, a value from H’1 to H’40 (bytes) can be set.

Note: The initial value of the MXPS bits is H’000 when no pipe is specified in the PIPESEL bits in the PIPESEL register or H’040 when a pipe is specified in the PIPESEL bits.

(1) Maximum packet size (MXPS) bits

These bits are used to specify the maximum data payload size (maximum packet size) for the selected pipe.

The initial value of these bits is H’40 (64 bytes).

- For PIPE1 and PIPE2, a value from H’1 (1 byte) to H’400 (1024 bytes) can be specified.
- For PIPE3 to PIPE5, H’8 (8 bytes), H’10 (16 bytes), H’20 (32 bytes), H’40 (64 bytes), or H’200 (512 bytes) can be specified. (Bits [2:0] are excluded.)
- For PIPE6 to PIPE9, a value from H’1 (1 byte) to H’40 (64 bytes) can be specified.

The value of the MXPS bits for individual transfer type must conform to the USB Specification.

Setting a value in these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check the PBUSY bit is 0 before changing the value of thee bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

When the MXPS bits are all 0, data must not be written to the FIFO buffer, and the PID setting must not be changed to BUF.

32B.2.15.5 Pipe Cycle Control Register [PIPEPERI] <Address: H'06E>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	IFIS	—	—	—	—	—	—	—	—	—	IITV		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
12	IFIS	0	R/W	Isochronous IN buffer flush This bit specifies whether to perform a buffer flush when the pipe specified in the PIPESEL bit is used for isochronous IN transfer. 0: Does not perform a buffer flush. 1: Performs a buffer flush.
11 to 3	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
2 to 0	IITV	All 0	R/W	These bits specify the transfer interval of the specified pipe. The value to be specified is the frame timing multiplied by an n-th power of 2.

(1) Isochronous IN buffer flush (IFIS) bit

When the selected pipe is used for isochronous IN transfer, this bit is used to specify this module automatically clears the FIFO buffer if this module fails to receive the IN token from the USB host in a (micro) frame sent at intervals specified in the IITV bits.

In double-buffer mode (DBLB = 1), this module clears only the data in one buffer used earlier than the other.

This module clears the FIFO buffer when it receives an SOF packet immediately after the (micro) frame in which the IN token hag to be received. Even if the SOF packet is corrupted, this module clears the FIFO buffer in the same timing to receive the SOF packet by the use of the internal interpolation function.

(2) Interval error detection interval (IITV) bits

These bits specify the interval of interval error detection for the selected pipe. The value to be specified is the frame timing multiplied by an n-th power of 2.

Setting a value in these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of thee bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

If you change the value specified in these bits to another after a USB communication, change the PID to NAK, and then set the ACLRM bit to initialize the interval timer before changing the value.

For PIPE3 to PIPE5 these bits are ignored. Write 0 to all these bits corresponding to PIPE3 to PIPE5.

You can specify a value in these bits when the transfer type of the selected pipe is isochronous.

(a) When the selected pipe is used for isochronous OUT transfer

If this module does not receive any data packet in the (micro) frame sent at intervals specified in the IITV bits, this module generates an NRDT interrupt.

This module generates an NRDY interrupt also if this module cannot receive data because an error, e.g., CRC error, has occurred in a data packet or because the FIFO buffer is full (such a situation might result if, for example, software [DMAC] delays in reading data from the FIFO buffer).

This module generates the NRDY interrupt when it receives an SOF packet. Even if the SOF packet is corrupted, this module generates the NRDY interrupt in the same timing to receive the SOF packet by the use of the internal interpolation function.

When, however, the value of the IITV bits is not 0, this module generates the NRDY interrupt every time an SOF packet is received at the specified intervals after interval counting starts.

If the PID of the selected pipe is changed to NAK after the interval timer starts, this module does not generate the NRDY interrupt even when it receives an SOF packet.

The condition for starting interval counting varies by the value of the IITV bits.

(a) When IITV = 0: Interval counting starts when the PID of the selected pipe is changed to BUF.

(Micro) frame	SOF	SOF	SOF	OUT	DATA0	SOF	OUT	DATA0
Setting of PID bits	NAK	NAK	BUF			BUF		
Whether token reception is expected (0: reception expected —: non-reception expected)	—	—	0			0		
Start of interval counting			↑					

Figure 32B.1 Relationship between (Micro) Frames and Whether Token Reception Is Expected When IITV = 0

(b) When IITV is not 0: Interval counting starts at the end of the first normal reception of data packet after the PID of the selected pipe is changed to BUF.

(Micro) frame	SOF	SOF	SOF	OUT	DATA0	SOF	SOF	OUT	DATA0	SOF	SOF	OUT	DATA0
Setting of PID bits	NAK	BUF	BUF			BUF	BUF			BUF	BUF		
Whether token reception is expected (0: reception expected —: non-reception expected)	—	—	0			—	0			—	0		
Start of interval counting			↑										

Figure 32B.2 Relationship between (Micro) Frames and Whether Token Reception Is Expected When IITV = 1

(b) When the selected pipe is used for isochronous IN transfer

The IITV bits are used in combination with the setting of the IFIS bit to 1. When the IFIS bit is 0, this module sends a data packet in response to a received token regardless of the value of the IITV bits.

When the IFIS bit is 1, if this module does not receive any IN token in the (micro) frame sent at intervals specified in the IITV bits although the FIFO buffer has sendable data, this module clears the FIFO buffer.

This module clears the FIFO buffer also when it cannot receive an IN token normally because of a bus error, e.g., CRC error.

This module clears the FIFO buffer when it receives an SOF packet. Even if the SOF packet is corrupted, this module clears the FIFO buffer in the same timing to receive the SOF packet by the use of the internal interpolation function.

The condition for starting interval counting varies by the value of the IITV bits. (The condition is the same as that for isochronous OUT transfer.)

The interval counter is cleared when one of the following conditions (1) to (3) is met:

(1) This module is reset (then, also the IITV bits are cleared).

(2) The ACLRM bit is set.

(3) This module detects a USB bus reset.

32B.2.16 Pipe Control Registers (PIPExCTR)

32B.2.16.1 PIPE1 Control Register [PIPE1CTR] <Address: H'070>
PIPE2 Control Register [PIPE2CTR] <Address: H'072>
PIPE3 Control Register [PIPE3CTR] <Address: H'074>
PIPE4 Control Register [PIPE4CTR] <Address: H'076>
PIPE5 Control Register [PIPE5CTR] <Address: H'078>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	INBUFM	—	—	—	ATREPM	ACLRM	SQCLR	SQSET	SQMON	PBUSY	—	—	—	PID	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0
R/W	R	R	R	R	R	R/W	R/W	R(0)/W(1)	R(0)/W(1)	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	Buffer status This bit indicates the FIFO buffer status of the specified pipe. 0: Buffer access is not possible. 1: Buffer access is possible.
14	INBUFM	0	R	Transmit buffer monitor When the specified pipe is in the sending direction, this bit indicates the FIFO buffer status of the specified pipe. 0: The FIFO buffer does not contain data that can be sent. 1: The FIFO buffer contains data that can be sent.
13 to 11	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
10	ATREPM	0	R/W	Automatic response mode This bit prohibits or enables automatic response of the specified pipe. 0: Automatic response prohibited 1: Automatic response enabled. (A zero-length packet response is sent during transmission. For reception, an NAK response is sent and an NRDY interrupt is generated.)
9	ACLRM	0	R/W	Automatic buffer clear mode This bit prohibits or enables automatic buffer clear mode for the specified pipe. 0: Prohibited 1: Enabled (all buffers are initialized)
8	SQCLR	0	R(0)/W(1)	Toggle bit clear Specify 1 in this bit to clear the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA0. 0: Writing disabled 1: DATA0 specified
7	SQSET	0	R(0)/W(1)	Toggle bit set Specify 1 in this bit to set the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA1. 0: Writing disabled 1: DATA1 specified
6	SQMON	0	R	Sequence toggle bit monitor This bit indicates the expected value of the sequence toggle bit for the next transaction in the specified pipe. 0: DATA0 1: DATA1

Bit	Bit Name	Initial Value	R/W	Description
5	PBUSY	0	R	Pipe busy This bit indicates whether the specified pipe is being used in the USB bus. 0: The specified pipe is not used in the USB bus. 1: The specified pipe is being used in the USB bus.
4 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1, 0	PID	All 0	R/W	Response PID These bits specify the response method for the next transaction in the specified pipe. 00: NAK response 01: BUF response (according to the buffer status) 10: STALL response 11: STALL response

(1) Buffer status (BSTS) bit

This bit indicates whether the CPU can access the FIFO buffer allocated to the selected pipe. This bit is operated by this module.

The meaning of this bit varies as follows depending on the value of the values of the DIR, BFRE, and DCLRM bits:

Table 32B.14 BSTS Bit Operations

DIR bit Setting Value	BFRE bit Setting Value	DCLRM bit Setting Value	Meaning of the BSTS Bit
0	0	0	This bit indicates 1 when reading of received data from the FIFO buffer becomes possible, and indicates 0 when reading data has finished.
		1	Setting prohibited
	1	0	This bit indicates 1 when reading of received data from the FIFO buffer becomes possible. This bit indicates 0 when 1 is written in the BCLF bit after reading data has finished.
		1	This bit indicates 1 when reading of received data from the FIFO buffer becomes possible, and indicates 0 when reading data has finished.
1	0	0	This bit indicates 1 when writing of send data in the FIFO buffer becomes possible, and indicates 0 when writing data has finished.
		1	Setting prohibited
	1	0	Setting prohibited
		1	Setting prohibited

(2) Transmit buffer monitor (INBUFM) bit

When the selected pipe is in the sending direction (DIR = 1), this module sets this bit when software (or the DMAC) has finished writing data to at least one FIFO buffer.

This module clears this bit when this module finishes sending all data from the FIFO buffer to which the data has been written. In double-buffer mode (DBLB = 1), this module clears this bit when this module has finished sending all data from the two FIFO buffers and software (or the DMAC) has not yet finished writing data to one FIFO buffer.

When the selected pipe is in the receiving direction (DIR = 0), this bit indicates the same value as that of the BSTS bit.

(3) Automatic response mode (ATREPM) bit

This bit can be set when the transfer type of the selected pipe is bulk transfer.

When this bit is 1, this module responds to tokens sent from the USB host as described below.

1. When the selected pipe is used for bulk IN transfer (TYPE = 01 and DIR = 1)
   When the ATREPM bit is 1 and the PID of the selected pipe is BUF, this module responds to an IN token by sending a zero-length packet.
   Each time this module receives ACK from the USB host (the sequence of one transaction is receiving an IN token, sending a zero-length packet, and then receiving ACK), this module updates (toggles) the sequence toggle bit (DATA- PID).
   This module does not generate BRDY and BEMP interrupts.

2. When the selected pipe is used for bulk OUT transfer (TYPE = 01 and DIR = 0)
   When the ATREPM bit is 1 and the PID of the selected pipe is BUF, this module responds to an OUT token (or a PING token) by sending an NAK response and generates an NRDY interrupt.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

To perform a USB communication with this bit set, the FIFO buffer must be empty. No data must be written to the FIFO buffer during the USB communication with this bit set.

When the transfer type of the selected pipe is isochronous transfer, this bit must always be 0.

(4) Automatic buffer clear mode (ACLRM) bit

When you need to clear the whole FIFO buffer allocated to the selected pipe, write 1 and 0 successively to the ACLRM bit.

Table 32B.15 shows the buffer contents this module clears when 1 and 0 are written successively to the ACLRM bit.

Table 32B.16 shows the cases that require this processing.

Table 32B.15 Buffer Contents This Core Clears When the ACLRM Bit is Set

No.	Contents to be Cleared by Setting the ACLRM Bit
(1)	Whole contents of the FIFO buffer allocated to the specified pipe (if the double buffer is set, both FIFO buffers are cleared)
(2)	If the transfer type of the specified pipe is Isochronous transfer, the interval count value is cleared.

Table 32B.16 Cases Requiring the ACLRM Bit to be Set

No.	Cases when Data Clear is Required
(1)	The whole contents of the FIFO buffer allocated to the specified pipe needs to be cleared.
(2)	The interval count value needs to be reset.
(3)	The value of the BFRE bit is changed.
(4)	The value of the DBLB bit is changed.
(5)	Forced termination of the transaction count function is performed.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(5) Sequence toggle bit clear (SQCLR) bit

When software sets this bit, this module specifies DATA0 as the expected value of the sequence toggle bit for the selected pipe. This module always clears this bit.

Writing 1 to the SQCLR bit must be done when the PID of the selected pipe is NAK.

If you write 1 to this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(6) Sequence toggle bit set (SQSET) bit

When software sets this bit, this module specifies DATA1 as the expected value of the sequence toggle bit for the selected pipe. This module always clears this bit.

Writing 1 to the SQSET bit must be done when the PID of the selected pipe is NAK.

If you write 1 to this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(7) Sequence toggle bit monitor (SQMON) bit

This bit indicates the expected value of the sequence toggle bit for the selected pipe. This bit is operated by this module.

When the transfer type of the selected pipe is other than isochronous transfer, this module toggles this bit when a transaction ends normally. This module, however, does not toggle this bit if a DATA-PID mismatch occurs during a transfer in the receiving direction.

(8) Pipe busy (PBUSY) bit

This module changes this bit from 0 to 1 when a USB transaction using the selected pipe starts. This module changes this bit from 1 to 0 when the transaction ends normally.

Reading this bit after software sets the PID to NAK enables you to check whether you can change pipe settings.

(9) Response PID (PID) bits

These bits are used to specify, the type of response of this module for individual pipes.

The default of PID is NAK. When the selected pipe is used for USB transfers, the PID setting muse be changed to BUF. For the basic operations (without communication packet errors involved) of this module depending on the value of the PID bits, see **Table 32B.17**.

If you have changed the PID of a selected pipe from BUF to NAK while the selected pipe is performing a USB communication, check, that the PBUSY bit is 0 to confirm that the USB transfer through the selected pipe has actually changed to the NAK status. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

This module changes the value of the PID bits in the following cases:

1. When the selected pipe is in the receiving direction and set the SHTNAK bit for the selected pipe, this module sets the PID to NAK when this module recognizes the end of a transfer.
2. When this module has received a data packet of which the payload size is larger than the maximum packet size, this module sets the PID to STALL (PID = 11).
3. If this module detects a USB bus reset, this module sets the PID to NAK.

To change the PID from NAK (PID = 00) to STALL, write 10 to the PID bits. To change the PID from BUF (PID = 01) to STALL, write 11 to the PID bits.

To change the PID from STALL (PID = 11) to NAK, write 10 to the PID bits once, and then write 00 to the PID bits. To change the PID from STALL to BUF, change the PID to NAK once, and then change it to BUF.

Table 32B.17 Core Operations Depending on the PID Setting

PID bit Setting Value	Transfer Type (TYPE Bit Setting Value)	Transfer Direction (DIR Bit Setting Value)	Operation of This Core
00 (NAK)	Bulk ("TYPE = 01"), or Interrupt ("TYPE = 10")	Independent of the setting value	Sends a NAK response for a token from the USB host.
	Isochronous ("TYPE = 11")	Receiving direction ("DIR = 0")	Does not respond to a token from the USB host.
		Sending direction ("DIR = 1")	Sends a zero-length packet for a token from the USB host.
01 (BUF)	Bulk ("TYPE = 01")	Receiving direction ("DIR = 0")	For an OUT token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module receives data and then sends an ACK or NYET response. If receiving data is not possible, this module sends an NAK response. For a PING Token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module sends an ACK response. If receiving data is not possible, this module sends an NAK response.
	Interrupt ("TYPE = 10")	Receiving direction ("DIR = 0")	For an OUT token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module receives data and then sends an ACK response. If receiving data is not possible, this module sends an NAK response.
	Bulk ("TYPE = 01") or Interrupt ("TYPE = 10")	Sending direction ("DIR = 1")	If the corresponding FIFO buffer is available for sending data, this module sends data in response to a token from the USB host. If sending data is not possible, this module sends an NAK response.
	Isochronous ("TYPE = 11")	Receiving direction ("DIR = 0")	For an OUT token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module receives data. If receiving data is not possible, this module discards data.
		Sending direction ("DIR = 1")	If the corresponding FIFO buffer is available for sending data, this module sends data in response to a token from the USB host. If sending data is not possible, this module sends a zero-length packet.
10 (STALL) or 11 (STALL)	Bulk ("TYPE = 01") or Interrupt ("TYPE = 10")	Independent of the setting value	Sends a STALL response for a token from the USB host.
	Isochronous ("TYPE = 11")	Independent of the setting value	Does not respond to a token from the USB host.

32B.2.16.2 PIPE6 Control Register [PIPE6CTR] <Address: H'07A>
PIPE7 Control Register [PIPE7CTR] <Address: H'07C>
PIPE8 Control Register [PIPE8CTR] <Address: H'07E>
PIPE9 Control Register [PIPE9CTR] <Address: H'080>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	—	—	—	—	—	ACLRM	SQCLR	SQSET	SQMON	PBUSY	—	—	—	PID	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0
R/W	R	R	R	R	R	R	R/W	R(0)/W(1)	R(0)/W(1)	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	Buffer status This bit indicates the FIFO buffer status of the specified pipe. 0: Buffer access is not possible. 1: Buffer access is possible.
14 to 10	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
9	ACLRM	0	R/W	Automatic buffer clear mode This bit prohibits or enables automatic buffer clear mode for the specified pipe. 0: Automatic buffer clear mode prohibited 1: Automatic buffer clear mode enabled (all buffers are initialized)
8	SQCLR	0	R(0)/W(1)	Toggle bit clear Specify 1 in this bit to clear the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA0. 0: Disabled 1: DATA0 specified
7	SQSET	0	R(0)/W(1)	Toggle bit set Specify 1 in this bit to set the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA1 0: Disabled 1: DATA1 specified
6	SQMON	0	R	Toggle bit monitor This bit indicates the expected value of the sequence toggle bit for the next transaction in the specified pipe. 0: DATA0 1: DATA1
5	PBUSY	0	R	Pipe busy This bit indicates whether the specified pipe is being used in the USB bus. 0: The specified pipe is not used in the USB bus. 1: The specified pipe is being used in the USB bus.
4 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1, 0	PID	All 0	R/W	Response PID These bits specifies the response method for the next transaction in the specified pipe 00: NAK response 01: BUF response (according to the buffer status) 10: STALL response 11: STALL response

(1) Buffer status (BSTS) bit

See **Section 32B.2.16.1(2), Transmit buffer monitor (INBUFM) bit**.

(2) Automatic buffer clear mode (ACLRM) bit

When you need to clear the whole FIFO buffer allocated to the selected pipe, write 1 and 0 successively to the ACLRM bit.

Table 32B.18 shows the buffer contents this module clears when 1 and 0 are written successively to the ACLRM bit.

Table 32B.19 shows the cases that require this processing.

Table 32B.18 Buffer Contents This Core Clears when the ACLRM Bit is Set

No.	Contents Cleared by ACLRM Bit Operation
(1)	All contents of the FIFO buffer allocated to the selected pipe

Table 32B.19 Cases Requiring the ACLRM Bit to be Set

No.	Contents Cleared by ACLRM Bit Operation
(1)	When clearing contents of the FIFO buffer allocated to the selected pipe
(2)	When resetting the interval counter
(3)	When the value of the BFRE bit is changed
(4)	When the transaction count function is terminated forcibly

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(3) Sequence toggle bit clear (SQCLR) bit

See **Section 32B.2.16.1(5), Sequence toggle bit clear (SQCLR) bit**.

(4) Sequence toggle bit set (SQSET) bit

See **Section 32B.2.16.1(6), Sequence toggle bit set (SQSET) bit**.

(5) Sequence toggle bit monitor (SQMON) bit

See **Section 32B.2.16.1(7), Sequence toggle bit monitor (SQMON) bit**.

(6) Pipe busy (PBUSY) bit

See **Section 32B.2.16.1(8), Pipe busy (PBUSY) bit**.

(7) Response PID (PID) bits

See **Section 32B.2.16.1(9), Response PID (PID) bits**.

32B.2.17 Transaction Counters (PIPExTRE)

32B.2.17.1 PIPE1 Transaction Counter Enable Register [PIPE1TRE] <Address: H'090>
PIPE2 Transaction Counter Enable Register [PIPE2TRE] <Address: H'094>
PIPE3 Transaction Counter Enable Register [PIPE3TRE] <Address: H'098>
PIPE4 Transaction Counter Enable Register [PIPE4TRE] <Address: H'09C>
PIPE5 Transaction Counter Enable Register [PIPE5TRE] <Address: H'0A0>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TRENB	TRCLR	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R/W	R(0)/W(1)	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
9	TRENB	0	R/W	Transaction counter enable This bit enables or disables the transaction counter. 0: Disables the transaction counter. 1: Enables the transaction counter.
8	TRCLR	0	R(0)/W(1)	Transaction counter clear This bit clears the transaction counter to 0. To clear the counter, write 1 to this bit. 0: Invalid 1: Clears the current transaction counter.
7 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

Note: Changing values of these register bits for a selected pipe must be done when the PID of the selected pipe is NAK.
If you change the value of a bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of the bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(1) Transaction counter enable (TRENB) bit

When software sets this bit for the selected pipe in the receiving direction after specifying a total number of packets in the TRNCNT bits, this module performs the following control when it finishes receiving the same number of packets as the total number specified in the TRNCNT bits:

1. When the continuous transfer mode is used (CNTMD = 1), this module switches the FIFO buffer to the CPU side at the end of reception even if the FIFO buffer is not full.
2. When the SHTNAK bit is 1, this module changes the PID of the selected pipe to NAK when it finishes receiving the same number of packets as the total number specified in the TRNCNT bits.
3. When the DENDE bit is 1 and the PKTMD bit is 0, this module asserts the DEND signal when reading the last data after having received the same number of packets as the total number specified in the TRNCNT bits.
4. When the BFRE bit is 1, this module asserts the BRDY interrupt signal when it finishes reading the last data after having received the same number of packets as the total number specified in the TRNCNT bits.

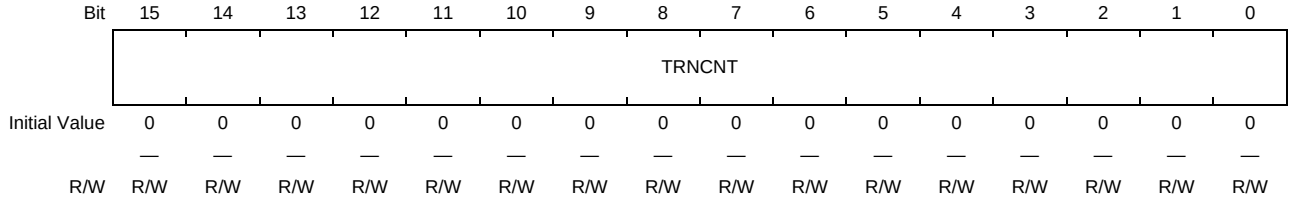
For the pipe in the sending direction, write 0 to this bit (TRENB bit). When not using the transaction count function, write 0 to this bit.

When using the transaction count function, specify a value in the TRNCNT bits before writing 1 to this bit. Also, write 1 to this bit before receiving the first packet among those to be counted by the transaction count function.

(2) Transaction counter clear (TRCLR) bit

When software sets this bit for a selected pipe, this module clears the current value of the transaction counter corresponding to the selected pipe, and then clears this bit.

32B.2.17.2 PIPE1 Transaction Counter Register [PIPE1TRN] <Address: H'092>
PIPE2 Transaction Counter Register [PIPE2TRN] <Address: H'096>
PIPE3 Transaction Counter Register [PIPE3TRN] <Address: H'09A>
PIPE4 Transaction Counter Register [PIPE4TRN] <Address: H'09E>
PIPE5 Transaction Counter Register [PIPE5TRN] <Address: H'0A2>



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TRNCNT	All 0	R/W	When writing: Specifies the total number of packets to be received by the pertinent pipe (number of transactions). When reading: Indicates the specified number of transactions if TRENB is 0. Indicates the number of the currently counted transaction if TRENB is 1.

(1) Transaction counter (TRNCNT) bits

When software writes 1 to the TRENB bit after setting the total number of packets to be received for the selected pipe in the receiving direction, this module performs the control described in **Section 32B.2.17.1(1), Transaction counter enable (TRENB) bit**.

When the TRENB bit is 0, this module indicates, by these bits, the number of transactions set.

When the TRENB bit is 1, this module indicates, by these bits, the current number of transactions counted.

This module increments the value of the TRNCNT bits by 1 when the status of reception meets all the following conditions (a) to (c):

(a) The TRENB bit is 1.

(b) When a packet is received, the value of the TRCNT bits is not equal to "current count + 1."

(c) The payload size of received packets has reached the value of the MXPS bits.

This module clears the TRNCNT bits to 0 when any of the following conditions (1) to (3) is met:

(1) All the following conditions (a) to (c) are met:

(a) The TRENB bit is 1.

(b) When a packet is received, the value of the TRCNT bits is equal to "current count + 1. "

(c) The payload size of received packets has reached the value of the MXPS bits.

(2) Both of the following conditions (a) and (b) are met:

(a) The TRENB bit is 1.

(b) A short packet has been received.

(3) The following condition is met:

(a) Written 1 to the TRCLR bit.

For the pipe in the sending direction, write 0 to these bits (TRNCNT bits). When not using the transaction count function, write 0 to these bits.

Changing the value of these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the TRENB bit is 0.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

If you change the value of these bits, write 1 to the TRCLR bit before writing 1 to the TRENB bit.

32B.2.18 Low Power Control Register

32B.2.18.1 Low Power Control Register [LPCTRL] <Address: H'100>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HWUPM	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved Nothing is assigned to these bits. Fix these bits to 0.
7	HWUPM	0	R/W	0: Resumes the PHY from the low-power mode while the internal bus clock (P1ϕ) is operating. 1: Enables resume from the low-power mode while the internal bus clock (P1ϕ) is stopped.
6 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

(1) HWUPM

This bit is used to specify whether to enable resumption from the low-power mode even while the internal bus clock (P1ϕ) is stopped.

0: Disables resumption while the internal bus clock (P1ϕ) is stopped.

1: Enables resumption while the internal bus clock (P1ϕ) is stopped.

This bit specifies whether to detect resume signaling while the internal bus clock (P1ϕ) is stopped. Whether to resume is controlled by the L1EXTMD bit. To resume from the low-power mode (LPM L1 state) while the internal bus clock (P1ϕ) is stopped, set both this bit and the L1EXTMD bit.

32B.2.19 Low Power Status Register

32B.2.19.1 Low Power Status Register [LPSTS] <Address: H'102>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SUSPM	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
14	SUSPM	0	R/W	USBPHY Suspend M control This bit controls the Suspend M signal to the USBPHY. 0: USBPHY suspend mode 1: USBPHY normal mode
13 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

(1) USBPHY SuspendM control (SUSPM) bit

This bit is used to control the SuspendM signal to the USBPHY. By default, the value of this bit is 0 and the USBPHY is in suspend mode. To operate this module, write 1 to this bit.

When the SUSPM bit is 0 (that is, when the UTMI clock is stopped), data cannot be written to this module, but can only be read from this module. Note, however, that data can be written to the registers listed in **Table 32B.20**.

Table 32B.20 Registers That Allow Writing when the SUSPM Bit is 0

Address	Register Name
H'000	SYSCFG0
H'002	BUSWAIT
H'100	LPCTRL
H'102	SUSPMODE

Note that the values written to the SYSCFG0 register while the USBPHY clock is stopped (SUSPM = 0) will be applied after the USBPHY clock starts (SUSPM = 1).

When the L1EXTMD bit is 0, this bit (SUSPM bit) is controlled (set or cleared) by software. When L1EXTMD bit is 1, this bit is controlled by software for the transition to the L1 or L2 state, and controlled by hardware for resumption from the L1 or L2 state, regardless of the level (L1 or L2).

32B.2.20 PHY Function Control Register

32B.2.20.1 PHY Function Control Register [PHYFUNCTR] <Address: H'104>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SusMon	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
14	SusMon	0	R	This bit allows reading of the status of the Suspend M signal.
13 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

(1) SusMon

SuspendM monitor bit (read only)

The status of the Suspend M signal can be read.

32B.2.21 PHY_OTG Control Register (PHYOTGCTR)

32B.2.21.1 PHY_OTG Control Register [PHYOTGCTR] <Address: H'10A>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DmPuDwn	DpPuDwn	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
10	DmPuDwn	1	R	Dm Pulldown monitor bit
9	DpPuDwn	1	R	Dp Pulldown monitor bit
8 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

(1) DmPuDwn

DmPulldown monitor bit (read only)

0: 15-kΩ Pulldown resistor control on the DM side is disabled.

1: 15-kΩ Pulldown resistor control on the DM side is enabled.

(2) DpPuDwn

DpPulldown monitor bit (read only)

0: 15-kΩ Pulldown resistor control on the DP side is disabled.

1: 15-kΩ Pulldown resistor control on the DP side is enabled.

32B.2.22 Peripheral L1 Control Register 1 (PL1CTRL)

32B.2.22.1 Peripheral L1 Control Register 1 [PL1CTRL1] <Address: H'144>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	L1EXTMD	—	—	HIRDTHR[3:0]				DVSQ[3:0]				L1NEGOMD	L1RESPMD[1:0]		L1RESPEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
14	L1EXTMD	0	R/W	USBPHY control mode on resumption from the L1 state This bit controls the USBPHY resume operation on resumption from the L1 state. 0: Does not set the Suspend M bit when the Host K signal is received. 1: Sets the Suspend M bit when the Host K signal is received
13, 12	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
11 to 8	HIRDTHR[3:0]	All 0	R/W	L1 response negotiation threshold HIRD threshold to be used for the L1NEGOMD bit The format is the same as the HIRD field of the HL1CTRL register.
7	DVSQ[3]	0	R	DVSQ extension bit This bit combined with the device state (DVSQ[2:0]) bit indicates the L1 state. 0000b: Powered state 0001b: Default state 0010b: Address state 0011b: Configured state 01xxb: Suspended state 10xxb: L1 state
6 to 4	DVSQ[2:0]	All 0	R	These bits mirror the DVSQ[2:0] bits in INTSTS0.
3	L1NEGOMD	0	R/W	L1 response negotiation control This bit sets the negotiation function using for the HIRD value. 0: Returns an ACK response if the received HIRD value is larger than the value of the HIRDTHR[3:0] bits. In other cases (including same values), an NYET response is returned. 1: Returns an ACK response if the received HIRD value is smaller than the value of the HIRDTHR[3:0] bits. In other cases (including same values), an NYET response is returned. This bit is valid only when the value of the L1RESPMD[1:0] bit is 11b.
2, 1	L1RESPMD[1:0]	All 0	R/W	L1 response mode These bits specify how to respond to an LPM token. 00b: NYET 01b: ACK 10b: STALL 11b: Response according to the value of the L1NEGOMD bit
0	L1RESPEN	0	R/W	L1 response enable This bit enables an L1 response. 0: Does not support LPM. 1: Supports LPM.

(1) L1 EXT mode (L1EXTMD) bit

This bit specifies how to control the SuspendM bit upon receiving the Host K signal when the USBPHY is stopped by setting the SuspendM bit in the L1 state.

0: Does not set the SuspendM bit when this module is resumed from the L1 state.

1: Sets the SuspendM bit when this module is resumed from the L1 state.

NOTES

1. The Host K period lasts a minimum of 50 µs. Therefore, the USBPHY might be unable to be resumed within the Host K period if the software settings for resume same as those for the suspend state are applied. Because the initial value of this bit is controlled by software, set this bit at initialization when the L1 state is to be supported.
2. For the transition to the L1 state, the SuspendM bit is controlled by software regardless of the value of this bit.
3. When this bit is set, the SuspendM bit will be set also at resumption from the L2 state.

(2) HIRD negotiation threshold (HIRDTHR[3:0]) bits

These bits specify the value of HIRD threshold to be used for the negotiation specified by the L1NEGOMD bit.

The format of the value is the same as that of the HIRD field in the HL1CTRL register.

(3) Device state extension (DVSQ[3]) bit

This bit is used as the fourth bit for the device state (DVSQ) bits.

0000b: Powered state

0001b: Default state

0010b: Address state

0011b: Configured state

01xxb: Suspended state

10xxb: L1 state

(4) Device status (DVSQ[2:0]) bits

These bits mirror the DVSQ[2:0] bits in the Interrupt Status Register (INTSTS0).

(5) L1 negotiation mode (L1NEGOMD) bit

This bit is used to specify the negotiation function using the HIRD value.

0: Returns an ACK response when the received HIRD value is larger than the value in HIRDTHR[3:0] bits, or returns an NYET response in other cases.

1: Returns an ACK response when the received HIRD value is smaller than the value in HIRDTHR[3:0] bits, or returns an NYET response in other cases.

This bit is valid only when the value of L1RESPMD[1:0] bits is 11b.

(6) L1 response mode (L1RSPMD[1:0]) bits

When the L1RSPED bit is set, this module respond to an LPM token according to the value of these bits. These bits specify how to respond to the LPM token.

00b: NYET

01b: ACK

10b: STALL

11b: Response according to the value of L1NEGOMD bit

(7) L1 response enable (L1RSPEN) bit

When this bit is 0, this module does not respond to the LPM token it receives. When this bit is 1, this module responds to the LPM token (it receives) according to the value of the LPMRESPMD[1:0] bits.

32B.2.23 Peripheral L1 Control Register 2

32B.2.23.1 Peripheral L1 Control Register 2 [PL1CTRL2] <Address: H'146>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	RWEMON	HIRDMON[3:0]				—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
12	RWEMON	0	R/W	This bit reflects the value of the RWE bit in the LPM token received last.
11 to 8	HIRDMON[3:0]	All 0	R/W	These bits reflect the value of the HIRD field in the LPM token received last.
7 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

(1) RWE value monitor (RWEMON) bit

This bit is referenced to monitor the value of the RWE bit in a received LPM token.

This bit reflects the value of the RWE bit in the LPM token received last.

(2) HIRD value monitor (HIRDMON) bits

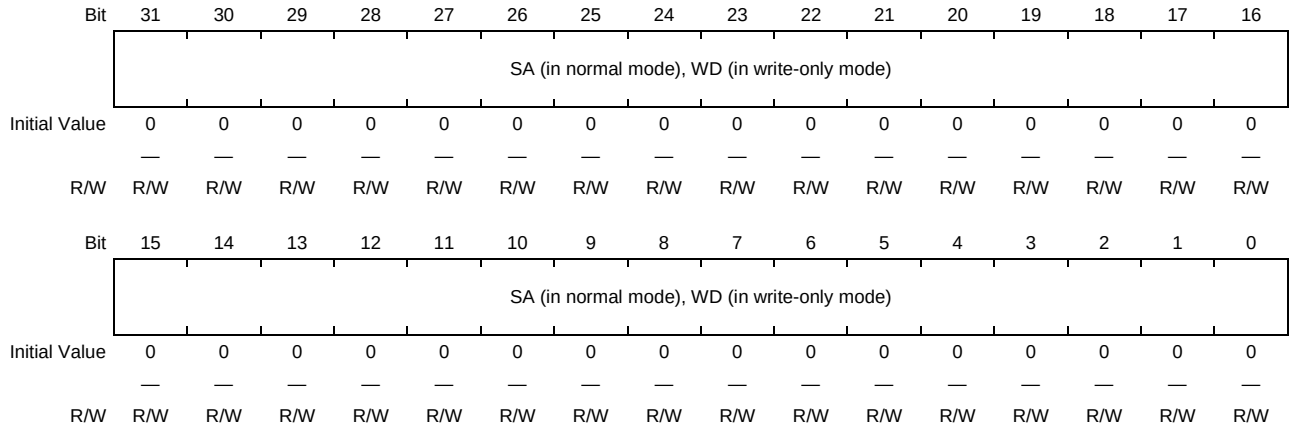
These bits are referenced to monitor the value of the HIRD field in a received LPM token.

These bits reflect the value of the HIRD field in the LPM token received last.

32B.3 Next Register Set

32B.3.1 Next Source Address Register n

32B.3.1.1 Next0 Source Address Register ch0 [N0SA_0] <Address: H'400>
Next1 Source Address Register ch0 [N1SA_0] <Address: H'40C>
Next0 Source Address Register ch1 [N0SA_1] <Address: H'440>
Next1 Source Address Register ch1 [N1SA_1] <Address: H'44C>

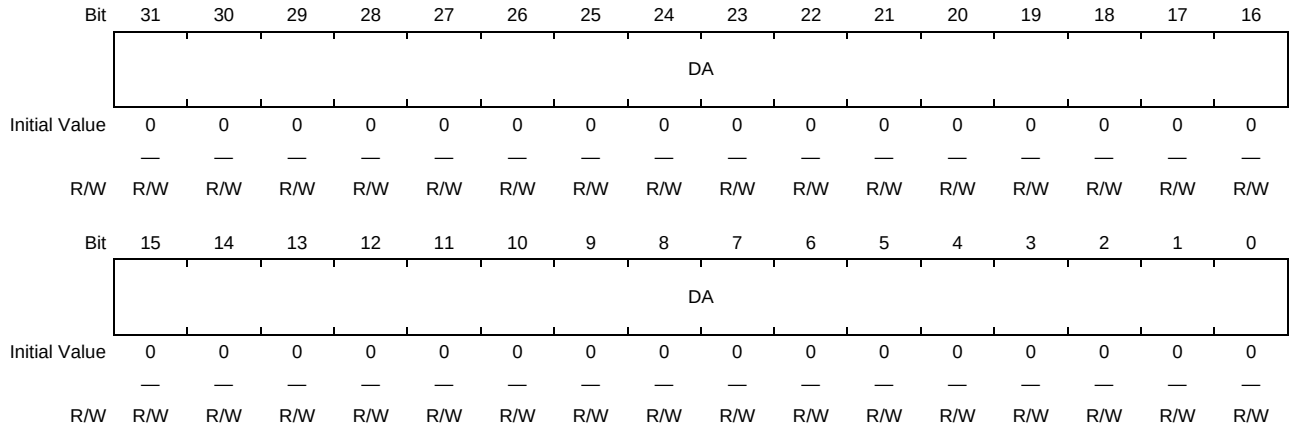


Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SA (in normal mode)	All 0	R/W	Source Address These bits specify the start address of the DMA transfer source.
	WD (in write-only mode)	All 0	R/W	Write Data These bits specify the write data in write-only mode.

Note: In a transfer in link mode, the data in the N0SA_n register is overwritten with descriptor read data.

32B.3.2 Next Destination Address Register n

32B.3.2.1 Next0 Destination Address Register ch0 [N0DA_0] <Address: H'404>
Next1 Destination Address Register ch0 [N1DA_0] <Address: H'410>
Next0 Destination Address Register ch1 [N0DA_1] <Address: H'444>
Next1 Destination Address Register ch1 [N1DA_1] <Address: H'450>

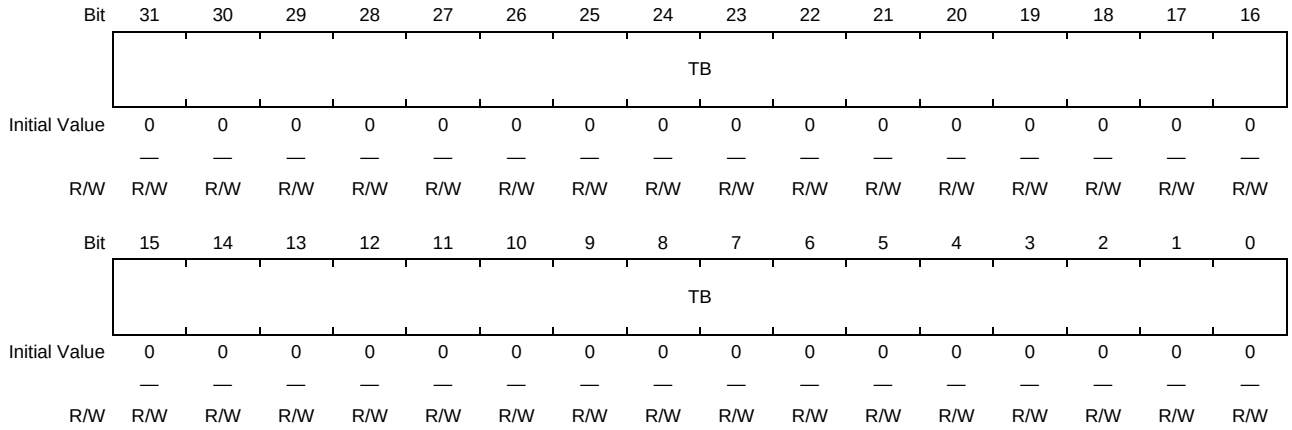


Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DA	All 0	R/W	Destination Address These bits specify the start address of the DMA transfer destination.

Note: In a transfer in link mode, the data in the N0DA_n register is overwritten with descriptor read data.

32B.3.3 Next Transaction Byte Register n

32B.3.3.1 Next0 Transaction Byte Register ch0 [N0TB_0] <Address: H'408>
Next1 Transaction Byte Register ch0 [N1TB_0] <Address: H'414>
Next0 Transaction Byte Register ch1 [N0TB_1] <Address: H'448>
Next1 Transaction Byte Register ch1 [N1TB_1] <Address: H'454>



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TB	All 0	R/W	Transaction Byte These bits specify the total number of transfer bytes. *Note*: Do not start a DMA transaction with 0 set in these bits.

Note: The N0TB_n register is overwritten by the descriptor read data during link mode transfer.

32B.4 Current Register Set

32B.4.1 Current Source Address Register

32B.4.1.1 Current Source Address Register ch0 [CRSA_0] <Address: H'418>
Current Source Address Register ch1 [CRSA_1] <Address: H'458>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRSA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRSA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRSA	All 0	R	Current Source Address Register This register indicates the read address for the next DMA transaction. The value of this register is incremented automatically while DMA transactions are in process. (When the SAD bit in the CHCFG_n register is 1, the value of this register is fixed. When the WONLY bit in the CHCFG_n register is 1, the value of this register is undefined.) The initial value of this register is loaded from the following register: In register mode: A transfer source address is loaded from the Next0/1 Register Set. In link mode: A transfer source address is loaded from the descriptor. (The descriptor read data is input to the N0SA_n register, and is loaded into the CRSA_n register when a transfer starts.) The value of this register is incremented when a read transfer ends. Read this register after DMA stops (that is, after the TACT bit in the CHSTAT_n register is cleared). (Handle the value read during DMA only as a reference value.)

32B.4.2 Current Destination Address Register

32B.4.2.1 Current Destination Address Register ch0 [CRDA_0] <Address: H'41C>
Current Destination Address Register ch1 [CRDA_1] <Address: H'45C>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRDA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRDA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRDA	All 0	R	Current Destination Address Register This register indicates the write address for the next DMA transaction. The value of this register is incremented automatically while DMA transactions are in process. (When the DAD bit in the CHCFG_n register is 1, the value of this register is fixed.) The initial value of this register is loaded from the following register: In register mode: A transfer destination address is loaded from the Next0/1 Register Set. In link mode: A transfer destination address is loaded from the descriptor. (The descriptor read data is input to the N0DA_n register, and is loaded into the CRDA_n register when a transfer starts.) The value of this register is incremented when a write transfer ends. Read this register after DMA stops (that is, after the TACT bit in the CHSTAT_n register is cleared). (Handle the value read during DMA only as a reference value.)

32B.4.3 Current Transaction Byte Register

32B.4.3.1 Current Transaction Byte Register ch0 [CRTB_0] <Address: H'420>
Current Transaction Byte Register ch1 [CRTB_1] <Address: H'460>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRTB															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRTB															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRTB	All 0	R	Current Transaction Byte Register This register indicates the remaining number of transfer bytes in the ongoing DMA transaction. The value of this register is decremented automatically while the DMA transaction is in process. The initial value of this register is loaded from the following register: In register mode: The number of transfer bytes is loaded from the Next0/1 Register Set. In link mode: The number of transfer bytes is loaded from the descriptor. (The descriptor read data is input to the N0TB_n register, and is loaded into the CRTB_n register when a transfer starts.) The value of this register is decremented when a write transfer ends. Read this register after DMA stops (that is, after the TACT bit in the CHSTAT_n register is cleared). (Handle the value read during DMA only as a reference value.)

32B.5 Channel Register Set

32B.5.1 Channel Status Register n

32B.5.1.1 Channel Status Register ch0 [CHSTAT_0] <Address: H'424>
Channel Status Register ch1 [CHSTAT_1] <Address: H'464>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DNUM								—	—	—	—	—	SWPRQ	DMARQM	INTM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MODE	DER	DW	DL	SR	TC	END	ER	SUS	TACT	RQST	EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	DNUM	All 0	R	Data Number These bits indicate the amount of valid data in the buffer. The indicated amount of data is the amount of the data that was read from the source but has not yet been written to the destination. (Unit: byte) Incrementing condition: • A DMA read transfer ends. Decrementing condition: • A DMA write transfer ends. Clearing conditions: • A condition for clearing the EN bit is met. • "1" is written to the SWRST bit in the CHCTRL_n register.
23 to 19	—	All 0	R	Reserved area. Write 0 to these bits. When these bits are read, 0 is read.
18	SWPRQ	0	R	Sweep Request This bit indicates the state of sweep request. This bit indicates the state of software sweep request (which has been activated by the SETSSWPRQ bit in the CHCTRL_n register). 1: The sweep request signal has been asserted. 0: The sweep request signal has not been asserted. Setting condition: • The SETSSWPRQ bit in the CHCTRL_n register is asserted. Clearing conditions: • The buffer becomes empty because of sweep. • "1" is written to the CLRHSWPRQM bit in the CHCTRL_n register. • "1" is written to the SWRST bit in the CHCTRL_n register.

Bit	Bit Name	Initial Value	R/W	Description
17	DMARQM	0	R	DMAREQ Mask This bit indicates the state of temporary masking of the DMA transfer request from the USB control. 1: The request is temporarily masked. 0: The request is released from temporary masking. Setting condition: ● The SETDMARQM bit in the CHCTRL_n register is set. Clearing conditions: ● “1” is written to the CLRDMARQM bit in the CHCTRL_n register. ● “1” is written to the SWRST bit in the CHCTRL_n register.
16	INTM	0	R	Interrupt Mask This bit indicates the state of temporary masking of the output from the USBFDMAmn interrupt. 1: The output is temporarily masked. 0: The output is released from temporary masking. Setting condition: ● “1” is written to the SETINTM bit in the CHCTRL_n register. Clearing conditions: ● “1” is written to the CLRINTM bit in the CHCTRL_n register. ● “1” is written to the SWRST bit in the CHCTRL_n register.
15 to 12	—	All 0	R	Reserved area. Write 0 to these bits. When these bits are read, 0 is read.
11	MODE	0	R	DMA Mode This bit indicates the DMA mode. The indicated value is the value of the DMS bit in the CHCFG_n register. 0: Register mode 1: Link mode
10	DER	0	R	Descriptor Error This bit indicates whether the data read from the descriptor is invalid (LV = 0) (regardless of the value of the DIM bit in the CHCFG_n register). 0: No descriptor error has occurred. 1: A descriptor error has occurred. Setting condition: ● In link mode, when the DRRP bit in the CHCFG_n register is 0, the LV bit value read from the descriptor is 0. Clearing conditions: ● “1” is written to the CLRDER bit in the CHCTRL_n register. ● “1” is written to the SWRST bit in the CHCTRL_n register.
9	DW	0	R	Descriptor WriteBack This bit indicates whether data is being written back to the descriptor. If a bus error occurs during write-back to the descriptor, this bit retains 1. 0: Status other than write-back to the header in link mode 1: (When the ER bit in the CHSTAT_n register is 0) Data is being written back to the header in link mode. (When the ER bit in the CHSTAT_n register is 1) A bus error has occurred during write-back to the header in link mode. Setting condition: ● Write-back to the header is started in link mode. Clearing conditions: ● Write-back to the header in link mode ends with an OK response. ● “1” is written to the SWRST bit in the CHCTRL_n register. If the bit retains 1 because of an error response, this bit can be cleared only by setting the SWRST bit.

Bit	Bit Name	Initial Value	R/W	Description
8	DL	0	R	Descriptor Load This bit indicates whether data is being read from the descriptor. If a bus error occurs during descriptor reading, this bit retains 1. 0: Status other than descriptor reading 1: (When the ER bit is 0) Descriptor reading is in process in link mode. (When the ER bit is 1) A bus error has occurred during descriptor reading in link mode. Setting condition: ● Descriptor reading is started in link mode. Clearing conditions: ● Descriptor reading in link mode ends with an OK response. ● “1” is written to the SWRST bit in the CHCTRL_n register. If the bit retains 1 because of an error response, this bit can be cleared only by setting the SWRST bit.
7	SR	0	R	Selected Register Set In register mode, this bit indicates the register set that is selected. 0: Next0 Register Set 1: Next1 Register Set Setting condition: ● The RSEL bit in the CHCFG_n register is set. Clearing condition: ● The RSEL bit in the CHCFG_n register is cleared.
6	TC	0	R	Terminal Count This status bit indicates whether the DMA transaction has ended. This bit is set only when the TCM bit in the CHCFG_n register is 0. 0: The DMA transfer has not ended. 1: The DMA transfer has ended. Setting conditions: ● In register mode, transfer of the total number of transfer bytes specified in the CRTB bits ends. ● In link mode, when the WBD bit in the header of the descriptor is 1, transfer of the total number of transfer bytes specified in the CRTB bits ends. ● In link mode, when the WBD bit in the header of the descriptor is 0, write-back to the descriptor ends. Clearing conditions: ● “1” is written to the CLRTC bit in the CHCTRL_n register. ● “1” is written to the SWRST bit in the CHCTRL_n register.
5	END	0	R	USBFDMAmn Interrupted This bit indicates whether the DMA transaction has ended and an USBFDMAmn interrupt has occurred. 0: The DMA transfer has not ended. 1: The DMA transfer has ended. Setting conditions: ● The condition for setting the TC bit is met, and the DEM bit in the CHCFG_n register is 0. ● In link mode, when the descriptor is read, the LV bit in the header is 0, and the DRRP and DIM bits in the CHCFG_n register are 0. Clearing conditions: ● “1” is written to the CLREND bit in the CHCTRL_n register. ● “1” is written to the SWRST bit in the CHCTRL_n register.

Bit	Bit Name	Initial Value	R/W	Description
4	ER	0	R	Error This bit indicates whether an error response has been received and a DMAERR interrupt has occurred during the DMA transfer. 0: No error response has been received. 1: An error response has been received. Setting condition: ● An error response is received in a bus cycle. Clearing condition: ● “1” is written to the SWRST bit in the CHCTRL_n register.
3	SUS	0	R	Suspend This bit indicates whether the channel is suspended. For details, see **Section 32B.9.13.8(2), Suspension**. 0: Channel_n is not suspended. 1: Channel_n is suspended. Setting condition: ● “1” is written to the SETSUS bit in the CHCTRL_n register during the DMA transfer using channel_n, and, thereby, the inside of the channel is suspended. Clearing conditions: ● “1” is written to the CLRSUS bit in the CHCTRL_n register. ● “1” is written to the CLREN bit in the CHCTRL_n register. ● A condition for clearing the EN bit in the CHSTAT_n register.
2	TACT	0	R	Transaction Active This bit indicates whether the DMAC is operating. This bit is used to check whether the channel is stopped fully. For details, see **Section 32B.9.13.8, Transfer state**. 0: The DMA in channel_n is stopped. 1: The DMA in channel_n is operating. Setting condition: ● “1” is written to the SETEN bit in the CHCTRL_n register (to start descriptor reading or wait for a DMA request). Clearing condition: ● The internal state is the idle state (the EN bit in the register has been cleared, and all transfers have ended).
1	RQST	0	R	Request This bit indicates whether a transfer request has been received. 0: No DMA transfer request has been received. 1: A DMA transfer request has been received. Setting conditions: ● “1” is written to the STG bit in the CHCTRL_n register. ● A DMA transfer request is received from the USB control. Clearing conditions: ● “1” is written to the SWRST bit in the CHCTRL_n register. ● “1” is written to the CLRRQ bit in the CHCTRL_n register. ● In single transfer mode (the TM bit in the CHCFG_n register is 0), a transfer is executed on the side specified by the REQD bit in the CHCFG_n register ● In register mode, all DMA transactions are complete (the REN bit in the CHCFG_n register is 0). ● In link mode, the DMA transfer of the last descriptor (LE = 1) ends. ● In link mode, a DMA transfer is stopped during descriptor reading (the LV bit is 0 and the DRRP bit in the CHCFG_n register is 0). ● In link mode, when the DEM bit in the CHCFG_n register is 0, a DMA transaction ends. ● The master interface receives a bus error signal.

Bit	Bit Name	Initial Value	R/W	Description
0	EN	0	R	Enable This bit indicates whether the operation of DMA channel n is enabled or stopped. 0: Operation is stopped. 1: Operation is enabled. Setting conditions: • “1” is written to the SETEN bit in the CHCTRL_n register. Clearing conditions: • “1” is written to the SWRST bit in the CHCTRL_n register. • “1” is written to the CLREN bit in the CHCTRL_n register. • An error response is received during transfer. • In register mode, all DMA transactions are completed (the REN bit in the CHCFG_n register is 0). • In link mode, DMA transfer of the last descriptor (the LE bit is 1) ends (if the WBD bit is 0, write- back to the descriptor ends). • In link mode, reading of a descriptor is stopped (the LV bit is 0 and the DRRP bit in the CHCFG_n register is 0).

Note 1. When the ER bit in the CHSTAT_n register is set, treat the corresponding series of transfers as invalid transactions.

Note 2. To interrupt a DMA transaction, mask or clear transfer requests or clear the EN bit in the CHSTAT_n register. (For the procedure to interrupt, see **Section 32B.9.13.8(3), Transfer suspension**.)

Note 3. If the DMA transfer request from the USB control and the transfer request by software (setting the STG bit in the CHCFG_n register) are used together, the cause of activating the request that takes effect cannot be identified. Therefore, design the system so that only one type of transfer requests is used.

Note 4. When using the transfer request by software, operate the STG bit for a new transfer request only after the DMA transfer requested last ends (after checking the end of the last DMA transfer by referencing the Current Register Set or another method).

32B.5.2 Channel Control Register n

32B.5.2.1 Channel Control Register ch0 [CHCTRL_0] <Address: H'428>
Channel Control Register ch1 [CHCTRL_1] <Address: H'468>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CLRDMARQM	SETDMARQM	CLRINTM	SETINTM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	— R	— R	— R	— R	— R	— R	— R	— R	— R	— R	— R	— R	— R/W	— R/W	— R/W	— R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SETSSWPRQ	—	SETREN	—	—	CLRSUS	SETUSS	CLRDER	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	— R	— R/W	— R	— R/W	— R	— R	— R/W	— R/W	— R/W	— R/W	— R/W	— R/W	— R/W	— R/W	— R/W	— R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved area. Write 0 to these bits. When these bits are read, 0 is read
19	CLRDMARQM	0	R/W	Clear DMAREQ Mask Writing 1 to this bit releases the DMA transfer requests from the USB control from temporary masking. Writing 1 to this bit also clears the DMARQM bit in the CHSTATn register. When this bit is read, 0 is read. 1: Releases the DNA transfer requests from masking set by using the SETDMARQM bit. 0: Has no effect on operation.
18	SETDMARQM	0	R/W	SET DMAREQ Mask Writing 1 to this bit temporarily masks the DMA transfer requests from the USB control. Writing 1 to this bit also sets the DMARQM bit in the CHSTATn register. When this bit is read, 0 is read. 1: Masks the DMA transfer requests from the USB control. 0: Has no effect on operation.
17	CLRINTM	0	R/W	Clear Interrupt Mask Writing 1 to this bit releases the USBFDMAmn interrupt from masking. Writing 1 to this bit also clears the INTM bit in the CHSTATn register. Releasing the INT_DMA[n] pin output from masking when the LVINT bit in the DCTRL register and the END bit in the CHSTAT_n register are 1 activates the INT_DMA[n] pin output. (The pin output is not activated if the LVINT bit is 0.) When this bit is read, 0 is read. 1: Releases the pin output from masking set by using the SETINTM bit. 0: Has no effect on operation.
16	SETINTM	0	R/W	SETINTMSet Interrupt Mask Writing 1 to this bit temporarily masks the USBFDMAmn interrupt. Writing 1 to this bit also sets the INTM bit in the CHSTATn register. When this bit is read, 0 is read. 1: Masks the USBFDMAmn interrupt. 0: Has no effect on operation.
15	—	0	R	Reserved area. Write 0 to this bit. When this bit are read, 0 is read.

Bit	Bit Name	Initial Value	R/W	Description
14	SETSSWPRQ	0	R/W	Set Software Sweep Request Writing 1 to this bit sweeps out the data stored in the buffer to the destination (see **Section 32B.9.13.3(1), Forced software sweeping request**). When this bit is read, 0 is read. 1: Writes, to the destination, the data that is stored in the buffer and has not yet been written to the destination. 0: Has no effect on operation. If the destination asserts a hardware request (REQD = 1), the sweep operation cannot be used.
13	—	0	R	Reserved area. Write 0 to this bit. When this bit are read, 0 is read.
12	SETREN	0	R/W	Set Register Set Enable Writing 1 to this bit sets the REN bit in the CHCFG_n register. When this bit is read, 0 is read. 1: Sets the REN bit in the CHCFG_n register. 0: Has no effect on operation.
11, 10	—	All 0	R	Reserved area. Write 0 to these bits. When these bits are read, 0 is read.
9	CLRSUS	0	R/W	Clear Suspend Writing 1 to this bit when the SUS bit in the CHSTAT_n register is 1 releases the ongoing DMA transfer from the suspended state. When this bit is read, 0 is read. 1: Releases the ongoing DMA transfer from the suspended state. 0: Has no effect on operation.
8	SETSUS	0	R/W	Set Suspend Writing 1 to this bit when the EN bit in the CHSTAT_n register is 1 suspends the ongoing DMA transfer. When this bit is read, 0 is read. 1: Suspends the ongoing DMA transfer. 0: Has no effect on operation.
7	CLRDER	0	R/W	Clear DER Writing 1 to this bit clears the DER bit in the CHSTAT_n register. Writing 1 to this bit also clear the USBFDMAmn interrupt. When this bit is read, 0 is read. 1: Clears the DER bit. 0: Has no effect on operation.
6	CLRTC	0	R/W	Clear TC Writing 1 to this bit clears the TC bit in the CHSTAT_n register. When this bit is read, 0 is read. 1: Clears the TC bit. 0: Has no effect on operation.
5	CLREND	0	R/W	Clear End Writing 1 to this bit clears the END bit in the CHSTAT_n register. Writing 1 to this bit also clear the USBFDMAmn interrupt. When this bit is read, 0 is read. 1: Clears the END bit. 0: Has no effect on operation.
4	CLRRQ	0	R/W	Clear Request Writing 1 to this bit clears the RQST bit in the CHSTAT_n register. When this bit is read, 0 is read. 1: Clears the RQST bit in the CHSTAT_n register. 0: Has no effect on operation.

Bit	Bit Name	Initial Value	R/W	Description
3	SWRST	0	R/W	Software Reset Writing 1 to this bit clears individual bits in the CHSTAT_n register (for the bits to be cleared, see the description of each bit). Setting this bit must be done when the EN and TACT bits are 0. When this bit is read, 0 is read. 1: Clears individual bits in the CHSTAT_n register. 0: Has no effect on operation.
2	STG	0	R/W	Software Trigger Writing 1 to this bit makes software set an internal transfer request. If this bit and the SWRST bits are set at the same time, clearing by the SWRST bit takes priority. When this bit is read, 0 is read. 1: Makes software set a transfer request (set the RQST bit in the CHSTAT_n register). 0: Has no effect on operation.
1	CLREN	0	R/W	Clear Enable Writing 1 to this bit clears the EN bit in the CHSTAT_n register (for details, see **Section 32B.9.13.8(3) Transfer suspension**). When this bit is read, 0 is read. 1: Disables DMA transfers (clears the EN bit in the CHSTAT_n register). 0: Has no effect on operation.
0	SETEN	0	R/W	Set Enable Writing 1 to this bit enables DMA transfers in DMA channel n. If this bit and the SWRST bits are set at the same time, clearing by the SWRST bit takes priority, and DMA transfers do not start. When this bit is read, 0 is read. 1: Enables DMA transfers (sets the EN bit in the CHSTAT_n register). 0: Has no effect on operation.

Note: Temporary masking (using the CLRDMARQM, and SETDMARQM bits) of the DMA transfer requests from the USB control applies to only the resources for channel n. Setting the SETDMARQM bit for channel n does not affect the operation of channel m.

32B.5.3 Channel Configuration Register

32B.5.3.1 Channel Configuration Register ch0 [CHCFG_0] <Address: H'42C>
Channel Configuration Register ch1 [CHCFG_1] <Address: H'46C>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMS	REN	RSW	RSEL	SBE	DIM	TCM	DEM	WONLY	—	DAD	SAD	DDS			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDS				DRRP	—	—	—	—	—	—	—	REQD	—	—	SEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	DMS	0	R/W	DMA Mode Select This bit specifies the DMA mode to be used. 0: Register mode (default) 1: Link mode
30	REN	0	R/W	Register Set Enable This bit specifies whether to successively perform, after a DMA transaction ends, another DMA transaction using the Next Register Set selected by the RSEL bit. This bit is valid only in register mode. 0: Does not perform the DMA transaction successively. 1: Performs the DMA transaction successively. Setting conditions: • "1" is written to this bit. • "1" is written to the SETREN bit in the CHCTRL_n register. Clearing conditions: • "0" is written to this bit. • The REN bit is 1, and a DMA transaction ends. To re-set the REN bit during a transaction, we recommend to use the SETREN bit in the CHCTRL_n register.
29	RSW	0	R/W	Register Select Switch This bit specifies whether to automatically invert the RSEL bit after a DMA transaction ends. This bit is valid only in register mode. 0: Does not invert the RSEL bit after a DMA transaction ends. (Default) 1: Inverts the RSEL bit after a DMA transaction ends.
28	RSEL	0	R/W	Register Set Select This bit is used to select the Next Register Set to be used for the next DMA transaction. This bit is valid only in register mode. When the RSW bit is 1, this bit is inverted automatically at the end of a DMA transaction. 0: Uses the Next0 Register Set. (Default) 1: Uses the Next1 Register Set. Transition condition: A DMA transaction ends with the RSW bit set.

Bit	Bit Name	Initial Value	R/W	Description
27	SBE	0	R/W	Sweep Buffer Enable This bit specifies whether to sweep (write) the data already read and stored in the buffer and stop transfer when the EN bit in the CHSTAT_n register is cleared during a DMA transaction. The sweep mode can be used only when the REQD bit is 0. 0: Stops transfer without sweeping out the buffer. (Default) 1: Stops transfer after sweeping out the buffer.
26	DIM	0	R/W	Descriptor Interrupt Mask This bit specifies whether to mask the USBFDMAmn interrupt when the LV bit value read from descriptor header is 0. 0: Does not mask the USBFDMAmn interrupt. (Default) 1: Mask the USBFDMAmn interrupt.
25	TCM	0	R/W	DMATC Mask This bit is used to mask the DMATC signal, which is sent from the DMAC to USB control. When this bit is 1 at the time the DMATC signal is to be output, the DMATC signal is not asserted. Also, the TC bit in the CHSTAT_n register is not asserted in that case. In register mode, this bit is automatically cleared. In link mode, this bit is not cleared automatically. Use this bit when you control DMA transfers by software. 0: Does not mask the DMATC signal. (Default) 1: Masks the DMATC signal. Clearing condition: A DMA transaction ends with the TCM bit set.
24	DEM	0	R/W	USBFDMAmn Mask When this bit is 1 at the time the USBFDMAmn interrupt is not asserted. Also, the END bit in the CHSTAT_n register is not asserted in that case. In register mode, this bit is not cleared automatically. In link mode, this bit is automatically cleared. 0: Does not mask the USBFDMAmn interrupt. (Default) 1: Masks the USBFDMAmn interrupt. Clearing condition: A DMA transaction ends with the DEM bit set.
23	WONLY	0	R/W	Write Only Mode This bit is used to switch the transfer operation mode to the write-only mode (see **Section 32B.9.12.2, Write only mode**). 0: Normal operation (default) 1: Write-only mode
22	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
21	DAD	0	R/W	Destination Address Direction This bit specifies the direction of counting the transfer-destination address in DMA channel n. If the transfer destination is on the USB control side, write 1 (fixed) to this bit. 0: Incrementing (default) 1: Fixed When the transfer destination uses the skip mode or is beat-unaligned, do not specify 1 (fixed) in this bit.
20	SAD	0	R/W	Source Address Direction This bit specifies the direction of counting the transfer-source address in DMA channel n. If the transfer source is on the USB control side, write 1 (fixed) to this bit. 0: Incrementing (default) 1: Fixed When the transfer source uses the skip mode or is beat-unaligned, do not specify 1 (fixed) in this bit.

Bit	Bit Name	Initial Value	R/W	Description
19 to 16	DDS[3:0]	All 0	R/W	Destination Data Size These bits specify the size of DMA transfer data. When the transfer destination is on the USB control side, select the normal mode. Use bit 3 to switch between the normal and skip modes. 0: Normal mode (default) 1: Skip mode Use bits 2 to 0 to specify the size of transfer data. (For specifiable values, see **Table 32B.21**.) 000: 8 bits (default) 001: 16 bits 010: 32 bits 011: 64 bits 100: 128 bits 101: 256 bits 110: 512 bits 111: Setting prohibited
15 to 12	SDS[3:0]	All 0	R/W	Source Data Size These bits specify the size of DMA transfer data. Use bit 3 to switch between the normal and skip modes. 0: Normal mode (default) 1: Skip mode Use bits 2 to 0 to specify the size of transfer data. (For specifiable values, see **Table 32B.21**.) 000: 8 bits (default) 001: 16 bits 010: 32 bits 011: 64 bits 100: 128 bits 101: 256 bits 110: 512 bits 111: Setting prohibited
11	DRRP	0	R/W	Descriptor Read Repeat This bit switches the operation to be performed when the LV value in the header read from the descriptor is 0. (See **section 32B.9.12.1(2)(a), Operation flow of link mode**.) • 0: This module sets the DER bit in the CHSTAT_n register, and then stops descriptor reading. (Default) • 1: This module keeps reading the same descriptor until the LV value changes to 1, and, when the LV value becomes 1, starts the DMA transfer using the values in the descriptor. The interval of descriptor reading is controlled by using the DSCITVL register.
10 to 7	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
6, 5	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 1.
4	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
3	REQD	0	R/W	Request Direction This bit specifies whether the USB control is on the transfer source side or it is on the transfer destination side. 0: The USB control is on the transfer source side. (Default) 1: The USB control is on the transfer destination side.
2, 1	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
0	SEL	0	R/W	Terminal Select This bit selects the FIFO channel to be used on the USB control side. 0: D0FIFO 1: D1FIFO

The range of values specifiable in the SDS[2:0] and DDS[2:0] bits depends on the data bus width, number of implemented buffer stages, and whether the address to be accessed is beat-aligned or not (beat-unaligned). The table below shows the range of specifiable values.

Table 32B.21 Range of Sizes That Can Be Specified in SDS and DDS Bits

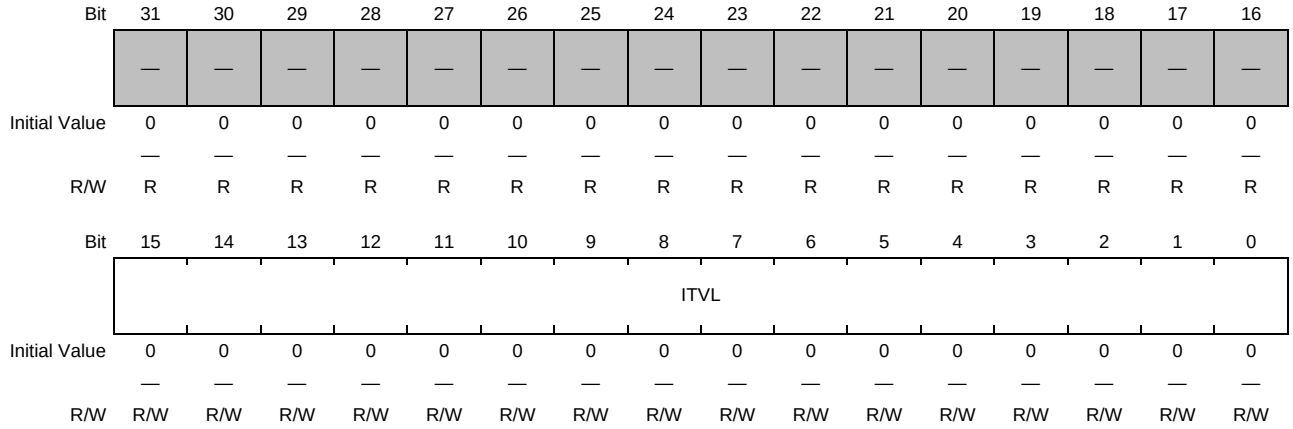
Transfer Address	REQD	SDS[2:0]	DDS[2:0]
Beat aligned	0	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.	8 to 32 bits (000 to 010) 128 to 512 bits (100 to 110)
	1	8 to 32 bits (000 to 010) 128 to 512 bits (100 to 110)	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.
Beat unaligned	0	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.	8 to 32 bits (000 to 010) 128 to 256 bits (100 to 101)
	1	8 to 32 bits (000 to 010) 128 to 256 bits (100 to 101)	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.

Note: When the destination is beat-unaligned with REQD = 0 or the source is beat-unaligned with REQD = 1, specify values in the range of specifiable values for beat-unaligned transfer addresses in both of the SDS[2:0] and DDS[2:0] bits.
Even if the transfer source and destination are beat-aligned when a DMA transaction starts, they might become beat-unaligned in the middle of the transaction when a skip transfer is used. If this might occur, perform settings in the first place on the assumption that the transfer source and destination are beat-unaligned.
If software cannot determine whether the transfer source and/or destination is beat-aligned, use values in the range of specifiable values for beat-unaligned transfer addresses.

32B.5.4 Channel Interval Register n

For details, see **Section 32B.9.13.6, Interval Count Function**.

32B.5.4.1 Channel Interval Register ch0 [CHITVL_0] <Address: H'430>
Channel Interval Register ch1 [CHITVL_1] <Address: H'470>



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
15 to 0	ITVL	All 0	R/W	Interval These bits specify the DMA transfer interval.

32B.5.5 Channel Extension Register n

32B.5.5.1 Channel Extension Register ch0 [CHEXT_0] <Address: H'434>
Channel Extension Register ch1 [CHEXT_1] <Address: H'474>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DPR				—	—	—	—	SPR			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
11 to 8	DPR	All 0	R/W	Destination PROT These bits specify the value to be output to the MHPROT[3:0] pin in a DMA write transfer. The initial value of these bits is H'0.
7 to 4	—	All 0	R/W	Nothing is assigned to these bits. Fix these bits to 0.
3 to 0	SPR	All 0	R/W	Source PROT These bits specify the value to be output to the MHPROT[3:0] pin in a DMA read transfer. The initial value of these bits is H'0.

32B.6 Link Register Set

When software sets a descriptor address in the NXLA_n register and starts the DMAC, hardware loads the value set in the NXLA_n register into the CRLA_n register. Then, the descriptor is read, and the DMAC starts a DMA transaction according to the values read from the descriptor. The value in the NXLA_n register is automatically updated to the Next Link Address value read from the descriptor, and the updated value is used as the descriptor address in the next DMA transaction.

32B.6.1 Next Link Address Register n (NXLA_n)

32B.6.1.1 Next Link Address Register ch0 [NXLA_0] <Address: H'438>
Next Link Address Register ch1 [NXLA_1] <Address: H'478>

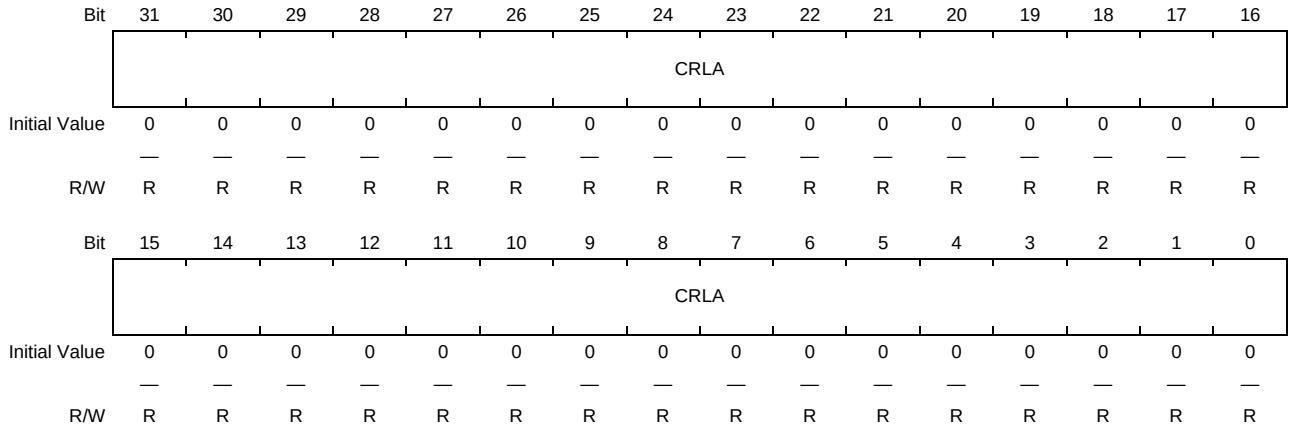
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NXLA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NXLA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	NXLA	All 0	R/W	Next Link Address
1, 0		All 0	R	These bits specify the link-destination address. Upper two bits are fixed to 0, and only the address aligned with a word boundary can be set.

32B.6.2 Current Link Address Register n (CRLA_n)

32B.6.2.1 Current Link Address Register ch0 [CRLA_0] <Address: H'43C>
Current Link Address Register ch1 [CRLA_1] <Address: H'47C>



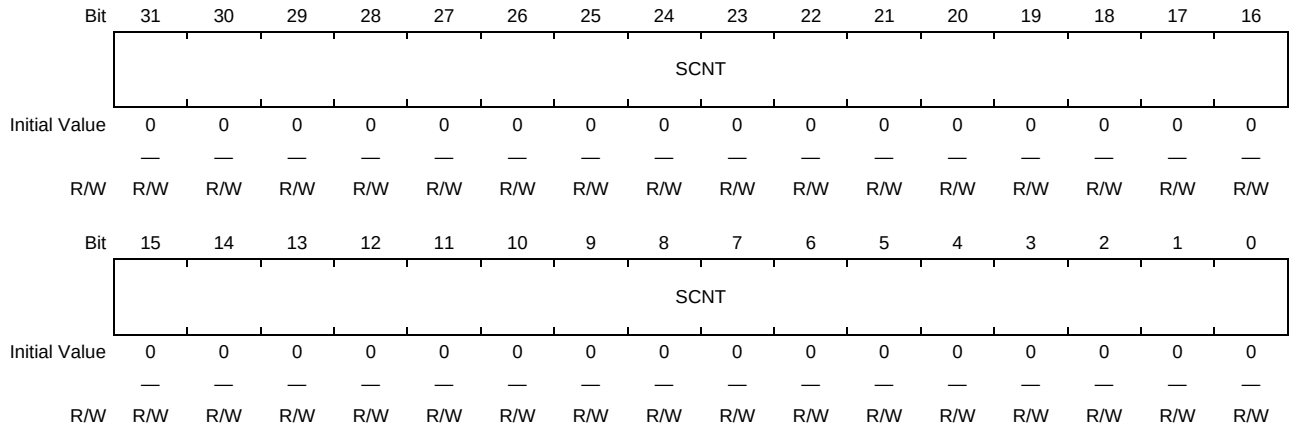
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRLA	All 0	R	Current Link Address These bits indicate the address of the descriptor being used for current transaction.

32B.7 Skip Register Set

This register set is used to specify settings for a skip (scatter/gather) transfer.

32B.7.1 Source Continuous Register n

32B.7.1.1 Source Continuous Register ch0 [SCNT_0] <Address: H'600>
Source Continuous Register ch1 [SCNT_1] <Address: H'620>

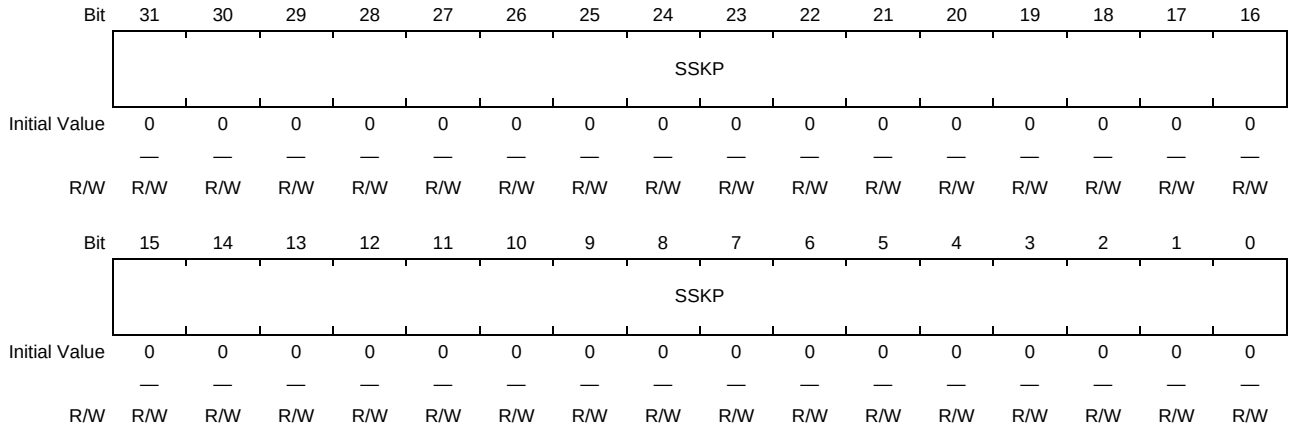


Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SCNT	All 0	R/W	Source Continuous These bits specify the size of the space to be accessed continuously by source address access. (Unit: byte)

Note: This register is used in pair with the SSKP_n register (see **Figure 32B.3**). To use this mode, set the SDS[3] bits in the CHCFG_n register to 1. To perform a skip transfer on the transfer source side, the SAD bit in the CHCFG_n register must not be set to 1 (Fixed). Do not perform a skip transfer with the SCNT bits set to 0.

32B.7.2 Source Skip Register n

32B.7.2.1 Source Skip Register ch0 [SSKP_0] <Address: H'604>
Source Skip Register ch1 [SSKP_1] <Address: H'624>



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SSKP	All 0	R/W	Source Skip These bits specify the size of area to be skipped in a source address access. (Unit: byte)

Note: This register is used in pair with the SCNT_n register (see **Figure 32B.3**). To use this mode, set the SDS[3] bits in the CHCFG_n register to 1. To perform a skip transfer on the transfer source side, the SAD bit in the CHCFG_n register must not set to 1 (Fixed).

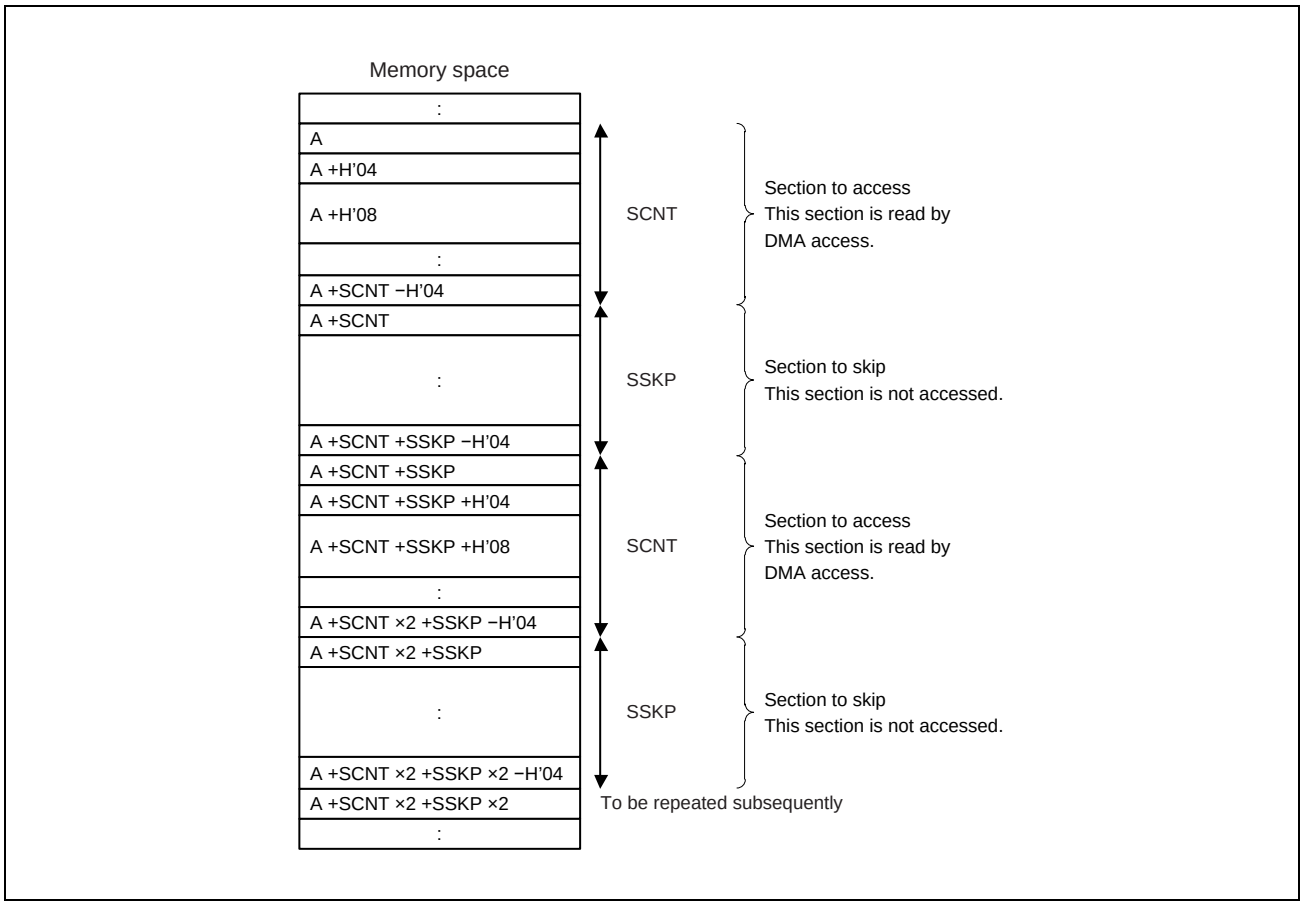
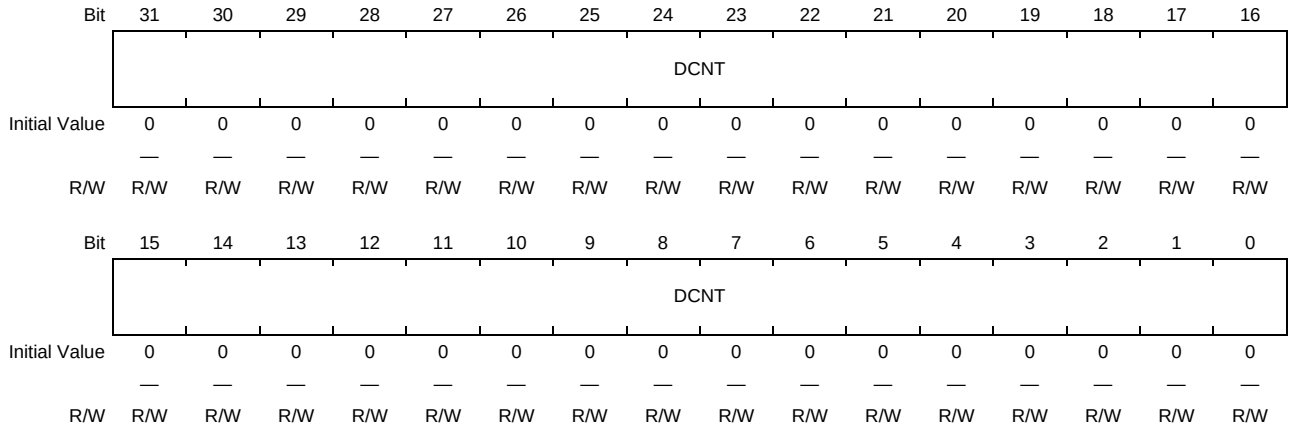


Figure 32B.3 Relationship between SSKP and SCNT

You can specify values of the SCNT and SSKP bits regardless of the source address and the value of the SDS field in the CHCFG_n register. The DMAC performs access based on the size specified in the SDS field, and fetches only valid data into the buffer (see **Section 32B.9.14.1(1), Read access**).

32B.7.3 Destination Continuous Register n

32B.7.3.1 Destination Continuous Register ch0 [DCNT_0] <Address: H'608>
Destination Continuous Register ch1 [DCNT_1] <Address: H'628>

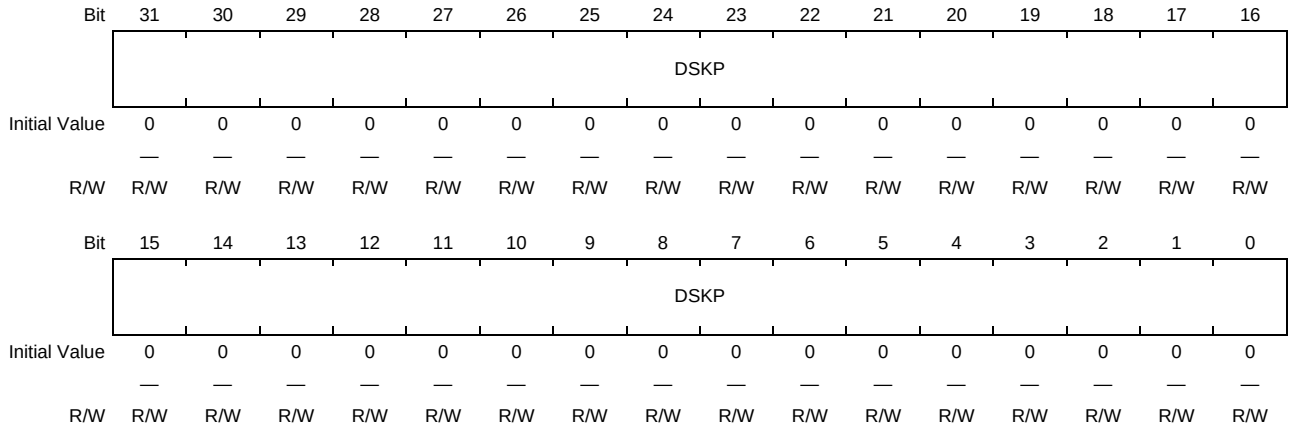


Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DCNT	All 0	R/W	Destination Continuous These bits specify the size of the space to be accessed continuously by destination address access. (Unit: byte)

Note: This register is used in pair with the DSKP_n register (see **Figure 32B.4**). To use this mode, set the DDS[3] bits in the CHCFG_n register to 1. To perform a skip transfer on the transfer destination side, the DAD bit in the CHCFG_n register must not be set to 1 (Fixed). Do not perform a skip transfer with the DCNT bits set to 0.

32B.7.4 Destination Skip Register n

32B.7.4.1 Destination Skip Register ch0 [DSKP_0] <Address: H'60C>
Destination Skip Register ch1 [DSKP_1] <Address: H'62C>



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DSKP	All 0	R/W	Destination Skip These bits specify the size of area to be skipped in a destination address access. (Unit: byte)

Note: This register is used in pair with the DCNT_n register (see **Figure 32B.4**). To use this mode, set the DDS[3] bits in the CHCFG_n register to 1. To perform a skip transfer on the transfer destination side, the DAD bit in the CHCFG_n register must not be set to 1 (Fixed).

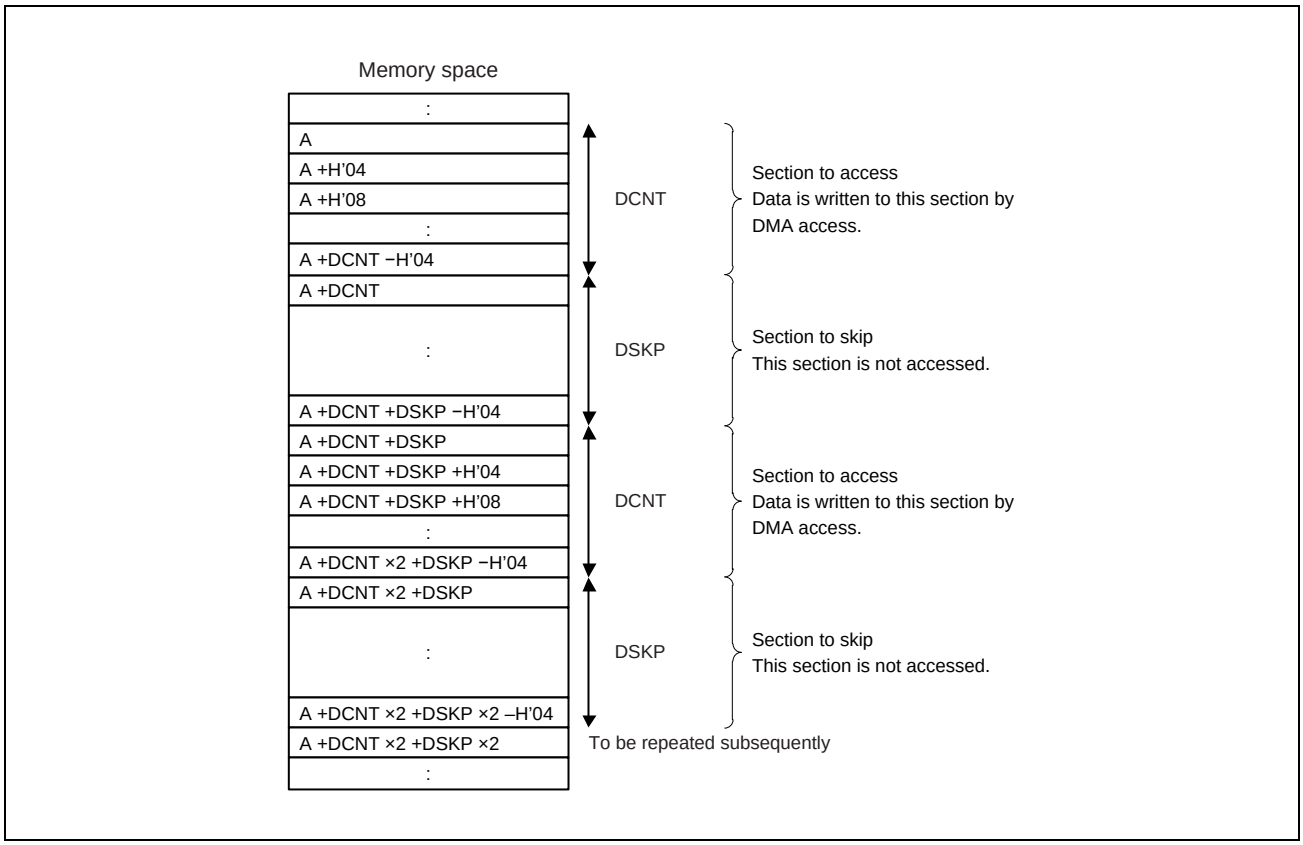


Figure 32B.4 Relationship between DSKP and DCNT

You can specify values of the DCNT and DSKP bits regardless of the destination address and the value of the DDS field in the CHCFG_n register. The DMAC performs write access to only the specified space that has a combined size not more than the size specified in the DDS field (see **Section 32B.9.14.1(1), Read access**).

32B.8 DMA Register Set

The registers described below are shared by all channels.

32B.8.1 DMA Control Register

32B.8.1.1 DMA Control Register [DCTRL] <Address: H'700>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	LWPR				—	—	—	—	LDPR			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LVINT	PR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
27 to 24	LWPR	All 0	R/W	Link WriteBack PROT These bits specify the value to be output to the MHPROT[3:0] pin at write-back to the descriptor in link mode.
23 to 20	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
19 to 16	LDPR	All 0	R/W	Link Descriptor PROT These bits specify the value to be output to the MHPROT[3:0] pin at reading of the descriptor in link mode.
15 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1	LVINT	0	R/W	Level Interrupt To use this module, be sure to set this bit to 1.
0	PR	0	R/W	Priority This bit specifies the transfer priority control mode (see **Section 32B.9.13.2, DMA channel priority control**). 0: Fixed priority mode 1: Round-robin mode

32B.8.2 Descriptor Interval Register n

32B.8.2.1 Descriptor Interval Register [DSCITVL] <Address: H'704>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DITVL								—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
15 to 8	DITVL	All 0	R/W	Descriptor Interval These bits specify the interval of descriptor read operation. The descriptor will be re-read at intervals of "value of DITVL x 256."
7 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

32B.8.3 DMA Control Register

32B.8.3.1 DMA Status EN Register [DSTAT_EN] <Address: H'710>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EN1	EN0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1	EN1	0	R	This bit indicates the state of the EN bit for DMA channel 1.
0	EN0	0	R	This bit indicates the state of the EN bit for DMA channel 0.

32B.8.4 DMA Status ER Register

32B.8.4.1 DMA Status ER Register [DSTAT_ER] <Address: H'714>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ER1	ER0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1	ER1	0	R	This bit indicates the state of the ER bit for DMA channel 1.
0	ER0	0	R	This bit indicates the state of the ER bit for DMA channel 0.

32B.8.5 DMA Status END Register

32B.8.5.1 DMA Status END Register [DSTAT_END] <Address: H'718>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	END1	END0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1	END1	0	R	This bit indicates the state of the END bit for DMA channel 1.
0	END0	0	R	This bit indicates the state of the END bit for DMA channel 0.

32B.8.6 DMA Status TC Register

32B.8.6.1 DMA Status TC Register [DSTAT_TC] <Address: H'71C>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TC1	TC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1	TC1	0	R	This bit indicates the state of the TC bit for DMA channel 1.
0	TC0	0	R	This bit indicates the state of the TC bit for DMA channel 0.

32B.8.7 DMA Status SUS Register

32B.8.7.1 DMA Status SUS Register [DSTAT_SUS] <Address: H'720>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SUS1	SUS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1	SUS1	0	R	This bit indicates the state of the SUS bit for DMA channel 1.
0	SUS0	0	R	This bit indicates the state of the SUS bit for DMA channel 0.

32B.9 Functions

32B.9.1 System Control and Oscillation Control

This chapter describes register manipulations required to perform initial setup of this module. This chapter also describes the registers required to control power consumption.

For the parts of the sequence that are required in both host and peripheral modes, see **Section 32A.9.1, Host/Peripheral Common Setting Sequence**.

32B.9.1.1 USB data bus resistor control

This module controls switchover between the pull-up resistors for the D+ signal and the pull-down resistors for the D- signal of the USBPHY. Use the DPRPU bit and DRPD bit in the SYSCFG0 register to set pull-up or pull-down of each signal.

Recognize that a connection to the USB host is established, and then set 1 for the DPRPU bit in the SYSCFG0 register to pull up the D+ signal.

After connected to the host, this module automatically switches the resistor when the state changes to reset handshake, suspend, or resume.

If 0 is set for the DPRPU bit in the SYSCFG0 register during communication with the host, the pull-up resistor (or termination resistor) for the USB data line is disabled. This can notify the host controller of disconnection of a device.

32B.9.2 Interrupt Function

32B.9.2.1 Overview of Interrupt Function (other than DMA Master)

The following shows a list of interrupt functions of this module. An interrupt is notified as U2P_IXL_INT. Check the status register to identify the interrupt factor. The interrupt from this module is not asserted when the supply of the internal bus clock (P1ϕ) is stopped.

Table 32B.22 List of Interrupt functions

Bit	Interrupt name	Interrupt factor	Related status
VBINT	VBUS interrupt	The status change of the VBUS input pin is detected. (Changes from L to H and from H to L are detected.)	VBSTS
RESM	Resume interrupt	In the suspended state, a change of the USB bus status is detected (from J-State to K-State or from J-State to SE0).	—
SOFR	Frame number update interrupt	If SOFRM is 0: An SOF packet with a different frame number is received. If SOFRM is 1: An SOF with µframe number 0 cannot be received due to a problem such as packet corruption.	—
DVST	Device state transition interrupt	A transition of a device state is detected. USB bus reset detected Suspended state detected Set Address request received Set Configuration request received	DVSQ
CTRT	Control transfer stage transition interrupt	A transition of a control transfer stage is detected. Setup stage completed ControlWrite transfer status stage transition ControlRead transfer status stage transition Control transfer completed Control transfer sequence error	CTSQ
BEMP	Buffer empty interrupt	All data in the buffer memory is sent and the buffer becomes empty. A packet exceeding the maximum packet size is received.	PIPEBEMP
NRDY	Buffer not ready interrupt	A token is received when the PID setting is BUF and the buffer memory is not available for sending and receiving data. A CRC error or bit stuff error occurs when data is received in isochronous transfer. An interval error occurs when data is received in isochronous transfer.	PIPENRDY
BRDY	Buffer ready interrupt	The buffer becomes ready (available for reading or writing data).	PIPEBRDY

The following shows the relationship between interrupts of this module.

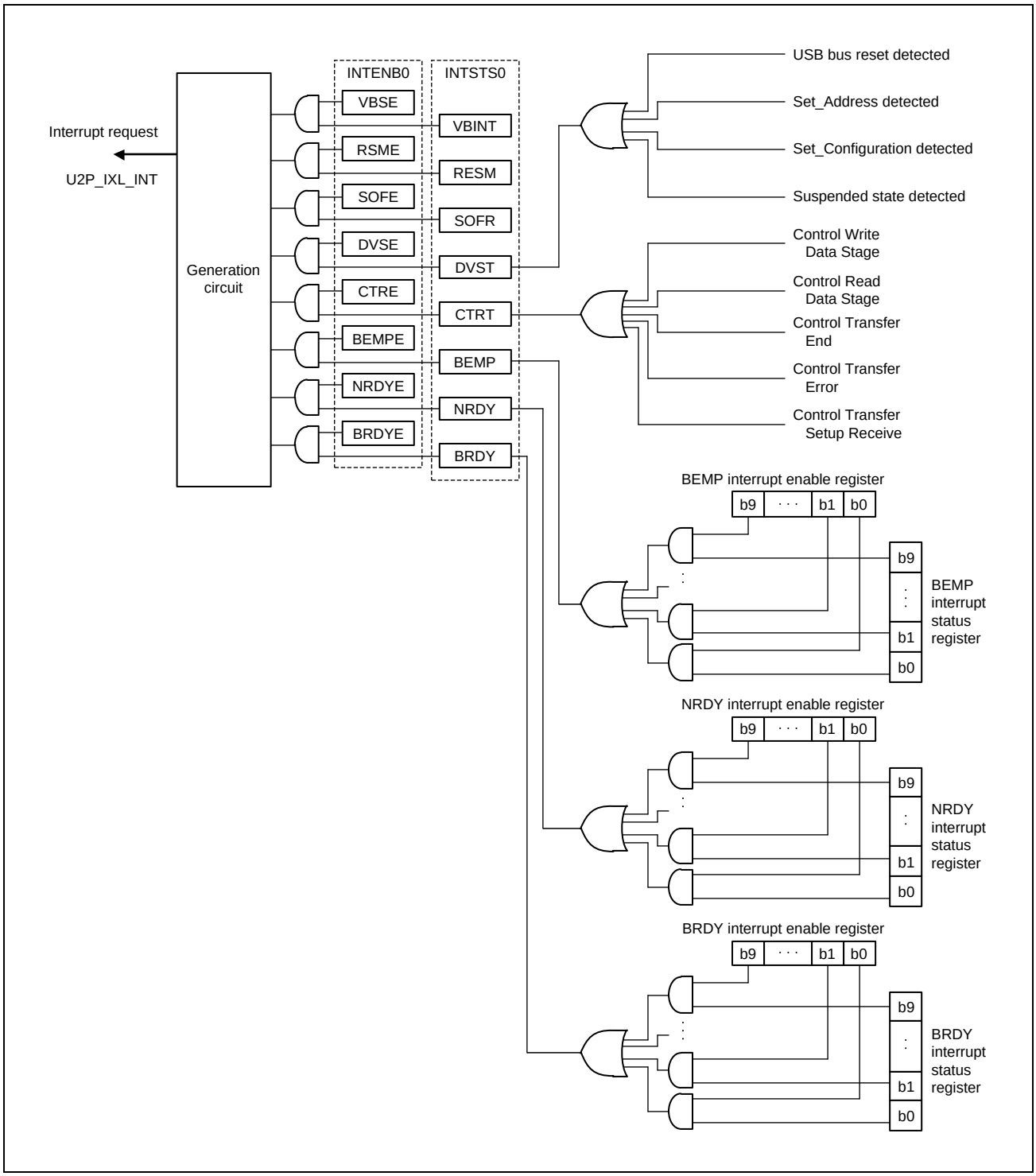


Figure 32B.5 Interrupt Association Diagram

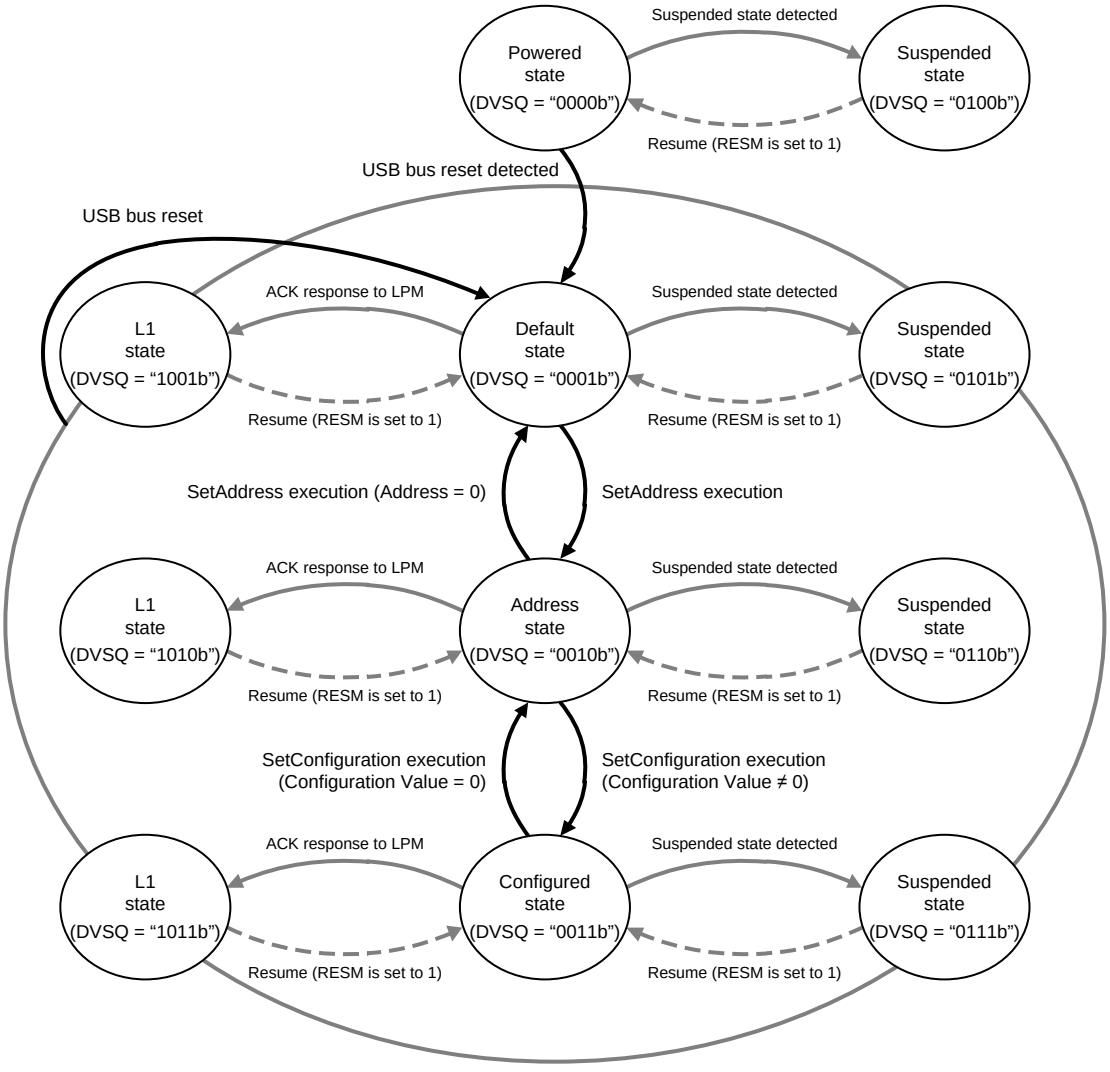
32B.9.2.2 Device State Transition Interrupts

Figure 32B.6 shows the device state transition diagram of this module.

This module manages device states and generates device state transition interrupts. However, resumption from the suspended state (resume signal detection) is detected by a resume interrupt.

Device state transition interrupts can be enabled and prohibited by using the INTENB0 register. A device state for which a transition has occurred can be checked in the DVSQ bits in the INTSTS0 register.

To trigger a transition to the default state, a device state transition interrupt is generated after a reset hand-shake protocol ends.



Note: When a transition indicated by a solid line occurs, the DVST bits are set to 1.
When a transition indicated by a broken line occurs, the RESM bit is set to 1.

Figure 32B.6 Device State Transition Diagram

32B.9.2.3 Control Transfer Stage Transition Interrupts

Figure 32B.7 shows the control transfer stage transition diagram of this module. This module manages a sequence of control transfers, and generates a control transfer stage transition interrupt. Control transfer stage transition interrupts can be enabled and prohibited by using the INTENB0 register. A transfer stage for which a transition has occurred can be checked in the CTSQ bits in the INTSTS0 register.

The following describes sequence errors that can occur during control transfer. If an error occurs, the PID bits in the DCPCTR register are set to 1X (STALL).

(1) For Control Read transfer

(a) An OUT or PING token is received in a situation where data has not been transferred yet for an IN token of the data stage.

(b) An IN token is received in the status stage.

(c) A data packet "DATAPID = DATA0" is received in the status stage.

(2) For Control Write transfer

(a) An IN token is received in a situation where an ACK response has not been sent yet for an OUT token of the data stage.

(b) The first data packet ”DATAPID = DATA0” is received in the data stage.

(c) An OUT or PING token is received in the status stage.

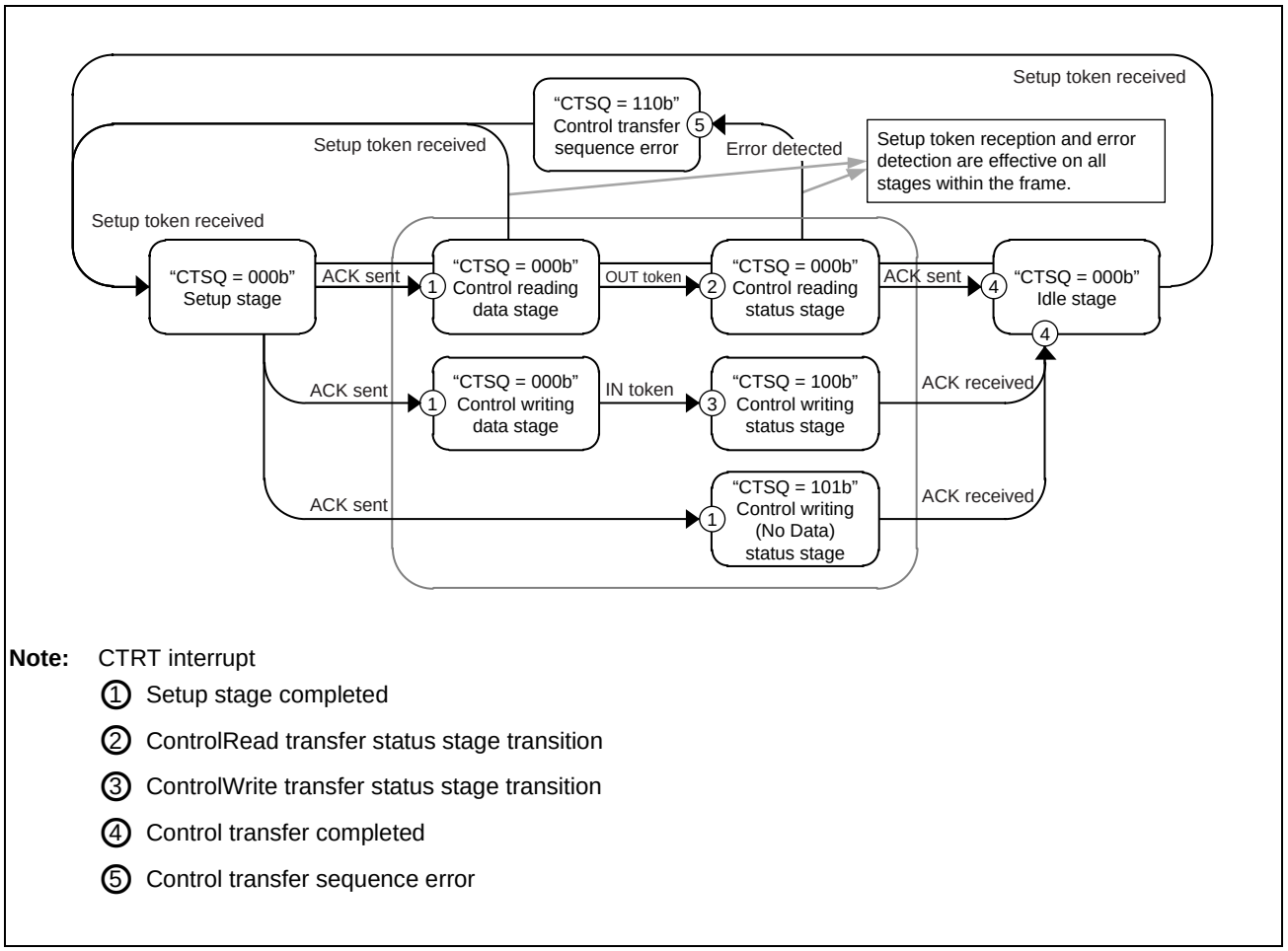
(3) For Nodata Control transfer

(a) An OUT or PING token is received in the status stage.

If the amount of received data exceeds the value of the wLength field in a USB request in the Control Write transfer data stage, this module cannot identify this situation as a control transfer sequence error. If a packet other than a zero-length packet is received in the Control Read transfer status stage, this module sends an ACK response and then terminates processing normally.

If a CTRT interrupt is generated (SERR bit is set to 1) due to a sequence error, the CTSQ bits retain 110 until the system writes 0 to the CTRT bits to clear the interrupt status.

Therefore, as long as the CTSQ bits retain 110, even if a new USB request is received, a CTRT interrupt that reports completion of a setup stage is not generated. (Information on completion of the setup stage is retained by this module, and a CTRT interrupt is generated after the interrupt status is cleared.)



Note: CTRT interrupt

① Setup stage completed

② ControlRead transfer status stage transition

③ ControlWrite transfer status stage transition

④ Control transfer completed

⑤ Control transfer sequence error

Figure 32B.7 Control Transfer Stage Transition Diagram

32B.9.2.4 Interrupts Relating to DMA Master

Table 32B.23 List of Interrupts

Interrupt Name	Interrupt Type
USBFDMAmn (m, n = 0, 1)	A DMA transaction ends.
	An invalid descriptor is read in link mode.
USBFDMAERRm (m = 0, 1)	An error response is returned for a transfer issued by the master interface.

This module has two types of DMA interrupts USBFDMAmn and USBFDMAERRm.

(1) USBFDMAmn interrupt (m, n = 0, 1)
This interrupt occurs when a DMA transaction ends or when an invalid descriptor is read in link mode.
An interrupt is divided for each DMA channel. USBFDMAm0 corresponds to the interrupt for DMA 0ch, and USBFDMAm1 corresponds to the interrupt for DMA 1ch.

(2) USBFDMAERRm interrupt
This interrupt occurs when an error response is returned for a transfer issued by the master interface. This interrupt applies to all DMA channels.

USBFDMAmn interrupt output can be temporarily masked by setting a register.

Interrupt detection can also be masked by setting the register. If interrupt detection is masked, the status register that indicates generation of an interrupt does not change.

On the other hand, a USBFDMAERRm interrupt signal does not have the masking function.

32B.9.3 Pipe Control

Table 32B.24 lists the pipe settings for this module. For USB data transfer, logical pipes called endpoints are used to enable data communication. This module provides 16 pipes for data transfer. Set each pipe according to the specification of the system.

Table 32B.24 PIPE Settings

Register Name	Bit Name	Setting	Comment
DCPCFG	TYPE	Specifies the transfer type.	Pipes 1 to 9: Settable
PIPECFG	BFRE	Selects BRDY interrupt mode.	Pipes 1 to 5: Settable
	DBLB	Selects double-buffer configuration.	Pipes 1 to 5: Settable
	CNTMD	Selects continuous transfer or non-continuous transfer.	DCP: Settable Pipes 1, 2: Settable only in bulk transfers Pipes 3 to 5: Settable
	DIR	Selects transfer direction.	Settable to IN or OUT
	EPNUM	Endpoint number	Pipes 1 to 9: Settable Set to a value other than “0000” when a pipe is in use.
	SHTNAK	Disables pipes when transfer is completed.	DCP: Settable Pipes 1, 2: Settable only in bulk transfers Pipes 3 to 5 Settable
PIPEBUF	BUFSIZE	Buffer memory size	DCP: Unsettable (fixed to 64/256 (CNTMD = 1) bytes) Pipes 1 to 5: Settable (up to 2 Kbytes specifiable) Pipes 6 to 9: Unsettable (fixed to 64 bytes)
	BUFNMB	Buffer memory number	DCP: Unsettable (fixed to area 0 to 3 hex) Pipes 1 to 5 Settable (area 8 to 87 hex specifiable) Pipes 6 to 9: Unsettable (fixed to area 4 to 7 hex)
DCPMAXP PIPEMAXP	MXPS	Maximum packet size	Setting conforming to the USB specification.
PIPEPERI	IFIS	Buffer flush	Pipes 1, 2: Settable only in isochronous transfers Pipes 3 to 5: Unsettable Pipes 6 to 9: Unsettable
	IITV	Interval counter	Pipes 1, 2: Settable only in isochronous transfers Pipes 3 to 5: Unsettable Pipes 6 to 9: Unsettable
DCPCTR PIPExCTR	BSTS	Buffer status	DCP state switched between receive and transmit buffer by ISEL bit
	INBUFM	IN buffer monitor	Available only for pipes 3 to 5
	ACLRM	Auto buffer clear	Pipes 1 to 9: Settable
	SQCLR	Sequence clear	Clears data toggle bit.
	SQSET	Sequence set	Sets data toggle bit.
	SQMON	Sequence monitor	Monitors data toggle bit.
	PBUSY	Pipe busy monitor	
	PID	Response PID	
DCPCTR PIPExCTR	ATREPM	Auto response mode	Pipes 1 to 5: Settable
PIPExTRE	TRENB	Transaction count enable	Pipes 1 to 5: Settable
	TRCLR	Current transaction counter clear	Pipes 1 to 5: Settable
PIPExTRN	TRNCNT	Transaction counter	Pipes 1 to 5: Settable

32B.9.3.1 Maximum packet size setting

Use the MXPS bits in the DCPMAXP and PIPEMAXP registers to specify the maximum packet size for each pipe. The default control pipe (DCP) and pipes 1 to 5 can be set to any of the maximum packet sizes defined by the USB specification. For pipes 6 to 9, 64 bytes are the upper limit of the maximum packet size. Set the maximum packet size before starting transfer (by setting “PID = BUF”).

DCP: Set 64 for high-speed operation.

DCP: Set 8, 16, 32, or 64 for full-speed operation. Pipes 1 to 5: Set 512 for high-speed bulk transfer.

Pipes 1 to 5: Set 8, 16, 32, or 64 for full-speed bulk transfer.

Pipes 1, 2: Set a value from 1 to 1024 for high-speed isochronous transfer.

Pipes 1, 2: Set a value from 1 to 1023 for full-speed isochronous transfer.

Pipes 6 to 9: Set 64.

High-bandwidth transfers used for interrupt transfers and isochronous transfers are not supported.

32B.9.3.2 Response PID

Set the response PID for each pipe with the PID bits in the DCPCTR and PIPExCTR registers.

(1) Response PID setting
The response PID specifies the response to a transaction from the host.

a) NAK: Always sends a NAK response to a generated transaction.

b) BUF: Responds to a transaction in accordance with the buffer memory state.

c) STALL: Always sends a STALL response to a generated transaction.

Regardless of the value set in the PID bits, an ACK is always sent as a response to a setup transaction and a USB request is stored in corresponding registers.

This module might write data to the PID bits depending on the transaction result. This module writes data to the PID bits in the following cases:

a) NAK:

(A) The SETUP token is received normally (for the DCP only).

(B) If 1 is set for the SHTNAK bit in the PIPECFG register during bulk transfer, a short packet is received.

(C) If 1 is set for the SHTNAK bit during bulk transfer, the transaction counter finishes.

b) BUF:

This module does not write "BUF".

c) STALL:

(A) When a maximum packet size over error is detected in a received data packet

(B) When a control transfer sequence error is detected

32B.9.3.3 Pipe control register switching procedure

The following bits in the pipe control registers can be modified only when USB transmission is disabled (PID = NAK).

Figure 32B.8 shows the procedure for changing the pipe control register state from the USB transmission enabled (PID = BUF) state.

The registered that cannot be manipulated in the USB transmission enabled (PID = BUF) state are as follows:

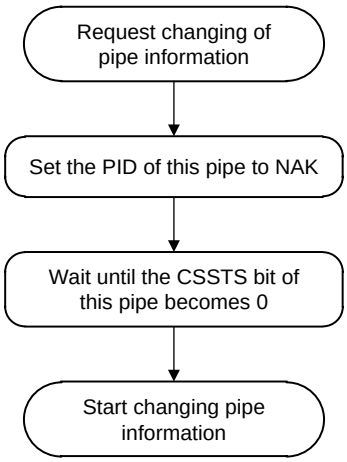
(1) All bits in the DCPCFG and DCPMAXP registers

(2) SQCLR and SQSET bits in the DCPCTR register

(3) All bits in the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers

(4) ATREPM, ACLRM, SQCLR, and SQSET bits in the PIPExCTR register

(5) All bits in the PIPExTRE and PIPExTRN registers



Note: If the pipe is detached during a USB transaction, the PBUSY bit might remain 1.

Figure 32B.8 Procedure for Changing Pipe Information from USB Transmission Enabled (PID = BUF) State

In addition, for the settings of the following bits in pipe control registers, only the pipe information that is not set for the CURPIPE bit in the CPU, DMA0, or DMA1 FIFO port can be changed.

Registers that cannot be set with pipe information that is set for the CURPIPE bit in a FIFO port:

(6) All bits in the DCPCFG and DCPMAXP registers

(7) All bits in the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers

(8) ACLRM bit in the PIPExCTR register

If you change the pipe information, make sure that the setting of CURPIPE is different from the new pipe number. For the default control pipe (DCP), after modifying the pipe information, use the BCLR bit to clear the buffer.

32B.9.3.4 Data PID sequence bit

When a normal data transfer is performed during bulk transfer or interrupt transfer, or in the data stage of control transfer, this module automatically toggles the sequence bit of a data PID. The sequence bit of the next data PID to be transferred can be confirmed with the SQMON bit in the DCPCTR or PIPExCTR registers. For data transmission, the sequence bit is switched when an ACK handshake is received. For data reception, the sequence bit is switched when an ACK handshake is sent. The data PID sequence bit can be changed by using the SQCLR and SQSET bits in the DCPCTR and PIPExCTR registers.

During control transfer, this module automatically sets the sequence bit when the stage changes. When the setup stage finishes, the data PID is set to DATA1. In the status stage, this module responds with “PID = DATA1” without referencing the sequence bit.

Note that, the data PID sequence bit must set, when a clear feature request is sent or received.

Also note that for isochronous transfer setting pipes, you cannot use the SQSET bit to manipulate the sequence bit.

32B.9.4 FIFO Buffer

This section describes the processing related to the FIFO buffer of this module.

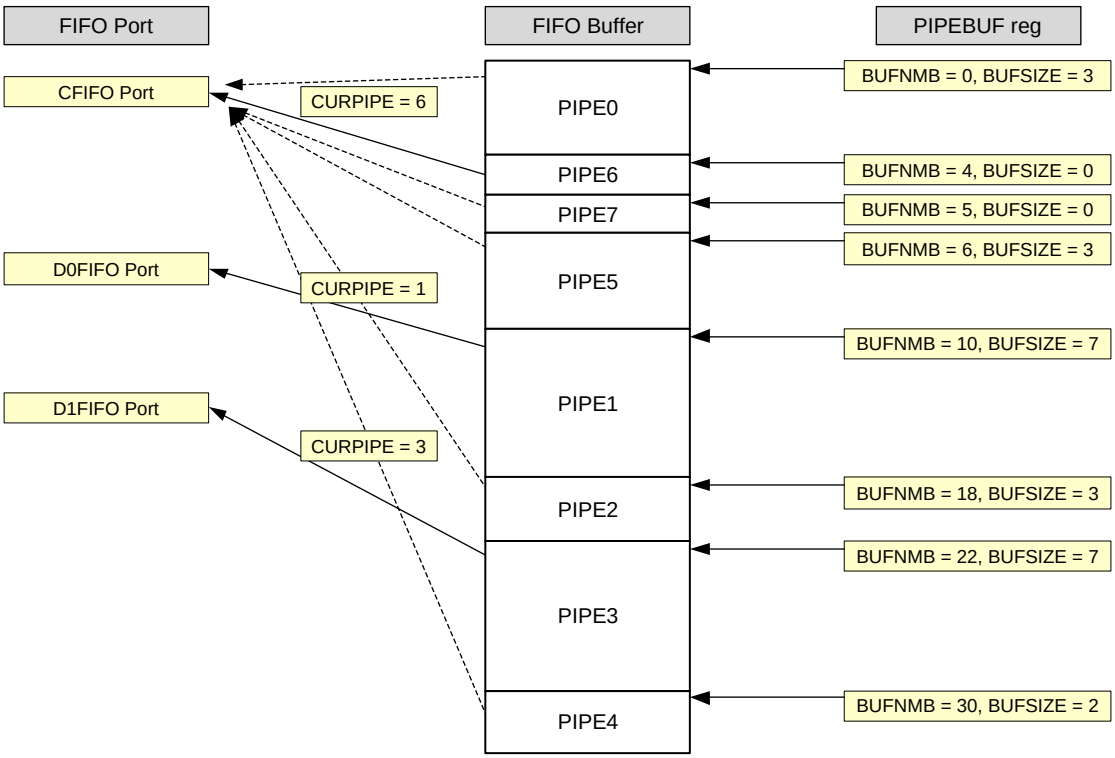
32B.9.4.1 FIFO buffer allocation

Figure 32B.9 shows an example of FIFO buffer memory mapping of this module. The FIFO buffer area is shared by the CPU and this module. The FIFO buffers can be accessed by either the system (CPU) or this module (SIE).

An independent FIFO buffer area is allocated for each pipe. The memory area is made up of memory blocks (1 block = 64 bytes) and is specified by the starting block number and the number of blocks (specified by the BUFNMB and BUFSIZE bits in the PIPEBUF register). When the CNTMD bit in the PIPExCFG register is used to set "continuous transfer mode", the value set with the BUFSIZE bit must be an integral multiple of the maximum packet size. If the double-buffer configuration is selected by the DBLB bit in the PIPExCFG register, two memory areas the size of which is specified by the BUFSIZE bit in the PIPEBUF register are allocated to a single pipe.

FIFO ports are used to access the FIFO buffer (data read/write). The pipe number of a pipe to be assigned to a FIFO port is specified by the CURPIPE bits in the CFIFOSEL/DxFIFOSEL register.

The FIFO buffer status of each pipe can be confirmed by using the BSTS and INBUFM bits in the DCPCTR and PIPExCTR registers. The access right of a FIFO port can be confirmed by using the FRDY bit in the CFIFOCTR/ DxFIFOCTR register.



Note: If pipes 8 and 9 are not in use: Settings such as BUFSIZE are unspecified.

Figure 32B.9 Example of FIFO Buffer Memory Mapping

32B.9.4.2 Clearing FIFO buffers

Table 32B.25 lists the modes in which this module can clear the FIFO buffer. Clearing of the FIFO buffer is controlled by the following bits.

Table 32B.25 List of FIFO Buffer Clearing Modes

Bit Name	BCLR	DCLRM	ACLRM
Register	CFIFOCTR register DxFIFOCTR register	DxFIFOSEL register	PIPExCTR register
Function	The FIFO buffer on the CPU side is cleared.	The FIFO buffer is automatically cleared after the data is read from the specified pipe.	The buffer is automatically cleared to discard all the received packets.
Clearing method	Write “1” to clear.	1: Mode enabled 0: Mode disabled	1: Mode enabled 0: Mode disabled

32B.9.5 FIFO Port Functions

This section describes FIFO port functions. **Table 32B.26** shows the FIFO port function settings for this module.

If data write access is enabled and data is written up to buffer full state (in non-continuous transfer: maximum packet size), the port automatically goes to the USB bus transmittable state. To enable transmission of data smaller than the buffer full (in non-continuous transfer: less than the maximum packet size), the BVAL bit in the CFIFOCTR/ DxFIFOCTR register must be used to set write end.

To send a zero-length packet, the BCLR bit of that register must be used to clear the buffer before the BVAL bit is used to write end.

When all the data is read in a read access, the port automatically enters the state in which new packets can be received. However, when a zero-length packet is received (DTLN = 0), no data can be read, and therefore the buffer must be cleared by using the BCLR bit.

The receive data length is confirmed with the DTLN bits in the CFIFOCTR/DxFIFOCTR register.

Table 32B.26 FIFO Port Function Settings

Register Name	Bit Name	Function	Note
C/DxFIFOSEL	RCNT	Selects DTLN read mode	
	REW	Rewinds buffer memory (re-read, re-write)	
	DCLRM	Reads received data of the specified pipe, and then automatically clears the received data	DxFIFO only
	DREQE	Asserts DREQ signal	DxFIFO only
	MBW	Specifies FIFO port access bit width	
	BIGEND	Selects FIFO port endian	CFIFO only
	ISEL	Specifies FIFO port access direction	DCP only
	CURPIPE	Selects current pipe	
C/DxFIFOCTR	BVAL	Finishes buffer memory write	
	BCLR	Clears CPU-side buffer memory	
	FRDY	Monitors FIFO port ready	
	DTLN	Confirms received data length	

32B.9.5.1 FIFO port selection

Table 32B.27 lists pipes that can be selected for each FIFO port.

Use the CURPIPE bits in the C/DxFIFOSEL register to select the pipe to be accessed. After selecting the pipe, confirm that the value written to the CURPIPE bits can be read correctly (if the previous pipe number is read out, this module is currently changing the pipe), confirm that FRDY = 1, and then access the FIFO port.

Figure 32B.10 shows the procedure for switching the pipe when accessing a FIFO port.

In addition, use the MBW bit to select the bus width with which to access the FIFO port. If the target pipe is the default control pile (DCP), the ISEL bit determines the buffer memory access direction. If the target pipe is not the DCP, the DIR bit in the PIPExCFG register determines the buffer memory access direction.

Table 32B.27 FIFO Port Access for Each Pipe

Pipe	Access Method	Usable Port
DCP	CPU access	CFIFO port register
Pipes 1 to 9	CPU access	CFIFO port register
	DMA access	DxFIFO port register

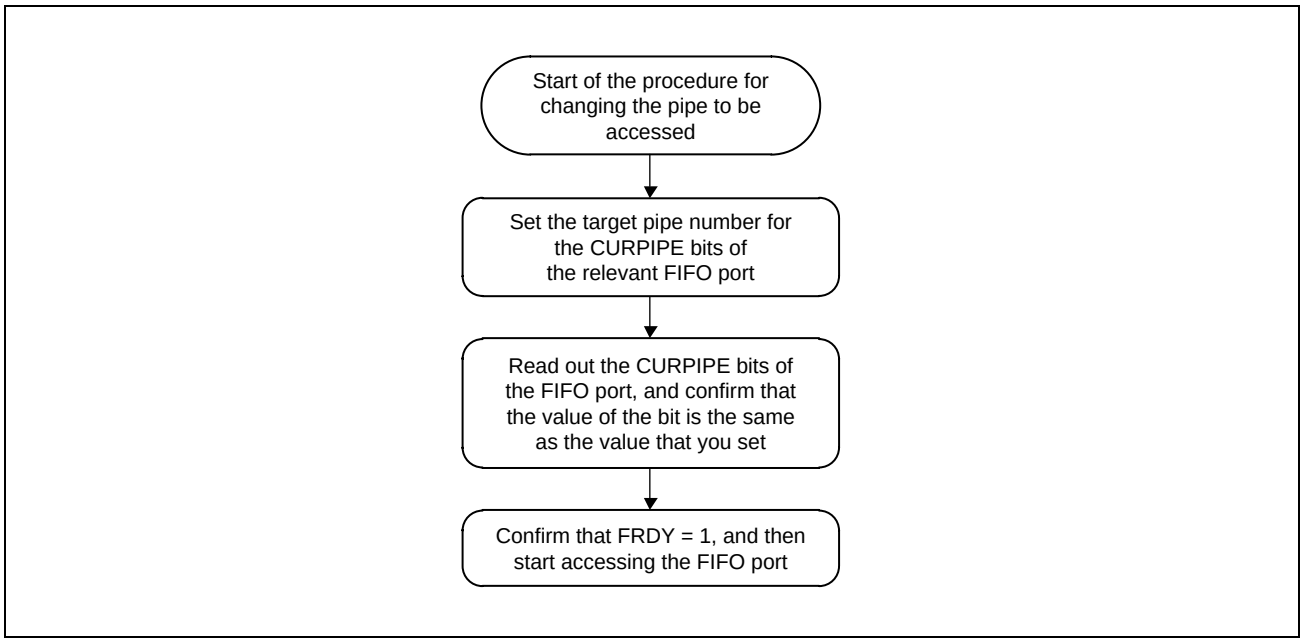


Figure 32B.10 Pipe Switching Procedure for FIFO Port Access

32B.9.5.2 DxFIFO automatic clear mode (DxFIFO port read direction)

This module automatically clears the buffer memory for a pipe when data is read out from the buffer memory if the DCLRM bit in the DxFIFOSEL register is set to 1.

Table 32B.28 shows the correspondence between packet reception and buffer memory clear processing by the software for each setting.

As shown in **Table 32B.28**, the buffer clearing conditions vary with the value that is set for the BFRE bit. However, using the DCLRM bit eliminates the need for buffer clear by the software even in states where clearing is required, which enables DMA transfers without using the software.

Note that for this function, only the buffer memory read direction can be set.

Table 32B.28 Relationship between Packet Reception and Buffer Memory Clear Processing by the Software

	Register Setting			
	DCLRM = 0		DCLRM = 1	
Buffer State during Packet Reception	BFRE = 0	BFRE = 1	BFRE = 0	BFRE = 1
Buffer full	Clearing unnecessary		Clearing unnecessary	
Zero-length packet received	Clearing necessary			
Normal short packet received	Clearing unnecessary	Clearing necessary		
Transaction count end				

32B.9.5.3 BRDY interrupt timing selection function

The BFRE bit in the PIPECFG register can be set so that a BRDY interrupt is not generated when a data packet of maximum packet size is received.

When using a DMA transfer, this function enables an interrupt to be generated only when the last data is received. The “last data” here indicates either a short packet reception or the transaction count end. If the BFRE bit is set to 1, a BRDY interrupt is generated after the received data is read. By reading the DTLN bits in the DxFIFOCTR register, the receive data length of last data packet received just before the BRDY interrupt is generated can be confirmed.

Table 32B.29 shows when this module generates a BRDY interrupt.

Table 32B.29 BRDY Interrupt Generation Timing

	Register Setting	
Buffer State during Packet Reception	BFRE = 0	BFRE = 1
Buffer full (normal packet received)	When packet is received	No interrupt generated
Zero-length packet received	When packet is received	When packet is received
Normal short packet received	When packet is received	When received data has been read from the buffer memory
Transaction count end	When packet is received	When received data has been read from the buffer memory

The BFRE bit function is valid only in the reading direction from the buffer memory. For the writing direction, fix the BFRE bit to 0.

32B.9.6 Control Transfer (DCP)

In the data stage of control transfer, the default control pipe (DCP) is used to transfer data. For the DCP, a single 64-byte buffer is allocated as a fixed area that is used for both control reading and writing (in continuous transfer mode (CNTMD = 1), the size of this area is fixed to 256 bytes). The buffer memory can be accessed through the CFIFO port only.

32B.9.6.1 Control transfer

(1) Setup stage

This module always responds with ACK to any normal setup packet for this module. The following shows the behavior of this module in the setup stage:

1. When this module receives a new setup packet, this module sets the following bits:
   (a) VALID bit in the INTSTS0 register: 1
   (b) PID bit in the DCPCTR register: NAK
   (c) CCPL bit in the DCPCTR register: 0

2. When this module receives a data packet after receiving a setup packet, this module stores USB request parameters in the USBREQ, USBVAL, USBINDX, and USBLENG registers.

Response processing for control transfer must be performed after VALID is set to 0. While VALID is 1, PID cannot be set to BUF, and therefore the data stage cannot end.

By using a function of the VALID bit, when this module receives a new USB request during control transfer, this module can respond to the newest request, canceling the request that is being processed.

This module also automatically recognizes the type of transfer (ControlRead, ControlWrite, or NodataControl) from the direction bit (bit 8 of bmRequestType) and request data length (wLength) of the received USB request to manage stage transition. For an incorrect sequence, a control transfer stage transition interrupt occurs to report a sequence error to software. For details about stage management by this module, see **Figure 32B.7**.

(2) Data stage

Use the DCP to transfer data in response to the received USB request. Before the DCP buffer memory is accessed, use the ISEL bit in the CFIFOSEL register to specify the access direction.

A transaction is executed by setting the PID bit in the DCPCTR register to BUF.

The end of data transfer is detected with a BRDY or BEMP interrupt. For ControlWrite transfer, use a BRDY interrupt. For ControlRead transfer, use a BEMP interrupt.

For ControlWrite transfer in high-speed operation mode, an NYET handshake response is performed in accordance with the buffer memory state.

(3) Status stage

If the PID bit in the DCPCTR register is BUF, setting the CCPL bit to 1 terminates control transfer.

After control transfer is terminated by the above setting, this module automatically executes the status stage according to the data transfer direction determined in the setup stage, as shown below:

(a) For ControlRead transfer

Upon receiving a zero-length packet from the USB Host Controller, this module sends an ACK response.

(b) For ControlWrite or NodataControl transfer

This module sends a zero-length packet, and then receives an ACK response from the USB Host Controller.

(4) Control transfer automatic response function

This module automatically responds to any normal SET_ADDRESS request. However, if a SET_ADDRESS request has any of the following errors, software must respond, instead of this module:

(a) bmRequestType ≠ “H’00”

(b) wIndex ≠ “H’00”

(c) wLength ≠ “H’00”

(d) wValue > “H’7F”

(e) DVSQ = “011 (Configured) ”

Software must respond to all requests other than SET_ADDRESS.

32B.9.7 Bulk Transfer (Pipes 1 to 5)

The user can select the buffer memory usage method (single/double buffer, continuous/non-continuous transfer mode) for bulk transfers. The buffer memory size can be set up to 2 KB. The controller manages the buffer memory state and automatically responds to PING packets and NYET handshakes.

32B.9.7.1 NYET handshake control

Table 32B.30 lists responses to tokens received in a bulk or control transfer.

When an OUT token is received in a bulk or control transfer and there is only an open space for one packet in the buffer memory, this module sends a NYET response. However, when a short packet is received, this module sends an ACK response instead of a NYET response even under these conditions.

Table 32B.30 List of Responses to Received Tokens

PID bit value	Buffer memory state*1	Received token	Response	Note
NAK/STALL	—	SETUP	ACK	—
	—	IN/OUT/PING	NAK/STALL	—
BUF	—	SETUP	ACK	—
	RCV-BRDY	OUT/PING	ACK	Receives data packet when OUT token is received*1
	RCV-BRDY	OUT	NYET	Receives data packet*2
	RCV-BRDY	OUT (Short)	ACK	Receives data packet*2
	RCV-BRDY	PING	ACK	*2
	RCV-NRDY	OUT/PING	NAK	
	TRN-BRDY	IN	DATA0/1	Sends data packet
	TRN-NRDY	IN	NAK	

Note: Details are described below.
RCV-BRDY*1: Buffer memory has a space for 2 packets or more when an OUT or PING token is received.
RCV-BRDY*2: Buffer memory has only a space for one packet when an OUT token is received.
RCV-NRDY: Buffer memory has no space for any packet when a PING token is received.
TRN-BRDY: Buffer memory has sent data when an IN token is received.
TRN-NRDY: Buffer memory has no send data when an IN token is received.

32B.9.8 Interrupt Transfer (Pipes 6 to 9)

This module performs an interrupt transfer in accordance with the period managed by the host controller. This module ignores (no response) PING packets in interrupt transfers. In addition, this module does not send a NYET handshake, but sends an ACK, NAK or STALL response.

Note that this module does not support high-bandwidth interrupt transfers.

32B.9.9 Isochronous Transfer (Pipes 1 and 2)

This module is provided with the following functions for isochronous transfers:

1. Notification of error information about isochronous transfers
2. Interval counter (IITV bit)
3. Data setup control for isochronous IN transfers (IDLY function)
4. Buffer flush function for isochronous IN transfers (IFIS bit)
5. SOF pulse output function

This module does not support high-bandwidth Isochronous transfers.

32B.9.9.1 Isochronous transfer error detection

This module has the following error information detection functions for the software to manage errors that occur during isochronous transfer.

Table 32B.31 and **Table 32B.32** describe the error checking procedure and interrupts that are generated.

1. PID error
   The PID of the received packet is invalid.
2. CRC error and bit stuffing error
   The received packet has a CRC error or invalid bit stuffing.
3. Max packet size over
   The data size of the received packet is larger than the preset maximum packet size.
4. Overrun error and underrun error
   (a) The buffer memory has no data when an IN token is received during IN-direction (send) transfer.
   (b) The buffer memory has no space although an OUT token is received during OUT-direction (receive) transfer.
5. Interval error
   An interval error occurs in the following cases:
   (a) An IN token cannot be received within the interval frame during isochronous IN transfer.
   (b) An OUT token cannot be received within the interval frame during isochronous OUT transfer.

Table 32B.31 Error Detection during Transmission/Reception of Token

Error Detection Priority	Error Type	Interrupt Generated at Error Detection and Status
1	PID error	No interrupt is generated (ignored as a corrupted packet).
2	CRC error, bit stuffing error	No interrupt is generated (ignored as a corrupted packet).
3	Overrun error, underrun error	An NRDY interrupt is generated and the OVRN bit is set. A zero-length packet is sent in response to an IN token. A data packet is not received in response to an OUT token.
4	Interval error	An NRDY interrupt is not generated.

Table 32B.32 Error Detection during Reception of Data Packet

Error Detection Priority	Error Type	Interrupt Generated at Error Detection and Status
1	PID error	No interrupt is generated (ignored as a corrupted packet).
2	CRC error, bit stuffing error	An NRDY interrupt is generated and the CRCE bit is set.
3	Packet size error (too large packet)	A BEMP interrupt is generated and the PID bit is set to STALL.

32B.9.9.2 DATA-PID

This module does not support high-bandwidth transfers.

The following shows actions that can be taken in response to a received PID.

1. IN direction:

(a) DATA0: Used to send packets.

(b) DATA1: Not used to send packets.

(c) DATA2: Not used to send packets.

(d) mDATA: Not used to send packets.

2. OUT direction (in full-speed operation):

(a) DATA0: Packets are received normally.

(b) DATA1: Packets are received normally.

(c) DATA2: Packets are ignored.

(d) mDATA: Packets are ignored.

3. OUT direction (in high-speed operation):

(a) DATA0: Packets are received normally.

(b) DATA1: Packets are received normally.

(c) DATA2: Packets are received normally.

(d) mDATA: Packets are received normally.

32B.9.9.3 Interval counter

(1) Outline of operation

The IITV bit in the PIPEPERI register can be used to set the interval of isochronous transfer. The interval counter enables the functions listed in **Table 32B.33**.

Table 32B.33 Functions of the Interval Counter

Transfer Direction	Function	Detecting Condition
IN	Transmit buffer flush function	An IN token cannot successfully be received within the interval frame during isochronous IN transfer.
OUT	Notification of unreceived token	An OUT token cannot successfully be received within the interval frame during isochronous OUT transfer.

Counting of intervals is based on received SOF packets or interpolated SOFs. Therefore, even if SOF packets are damaged, the isochronism can still be maintained. Frame intervals are set as 2n (micro) frames, where n is the value of the IITV bit.

(2) Interval counter initialization

This module initializes the interval counter under the following conditions:

(a) Power on reset
The IITV bit is initialized.

(b) Clearing of the buffer memory by the ACLRM bit
The IITV bit is not initialized but the counter is initialized.

(c) USB bus reset

After the interval counter is initialized and a packet is successfully transferred, counting of intervals starts under the following conditions:

1) An SOF packet is received after data is sent in response to an IN token when PID = BUF.

2) An SOF packet is received after data is received in response to an OUT token when PID = BUF.

Note that the interval counter is not initialized in the following conditions:

(a) The PID bit is set to NAK or STALL.
The interval timer is not stopped at this interval. The transaction will be attempted at the next interval.

(b) USB bus reset or USB suspension
The IITV bit is not initialized. When an SOF packet is received, counting starts from the value existing before reception.

32B.9.9.4 Send data setup for isochronous transfer

This module becomes able to send data packets by isochronous transfer from the next frame after data is written to the buffer memory and then an SOF packet is detected. This is called “send data setup for isochronous transfer”.

This function can identify the frame with which data sending started.

If the buffer memory is in a double-buffer configuration and writing to both buffers has been completed, only the buffer to which writing finished earlier can transfer data. Therefore, even when several IN tokens are received within the same frame, only one packet of data is sent from the buffer memory.

When an IN token is received, if the buffer memory is ready for sending data, the data is transferred and a normal response is returned. However, if the buffer memory is not ready for sending data, a zero-length packet is sent and an underrun error occurs.

Figure 32B.11 shows examples of sending using the send data setup function for isochronous transfer by setting “IITV = 0” (each frame) in this module.

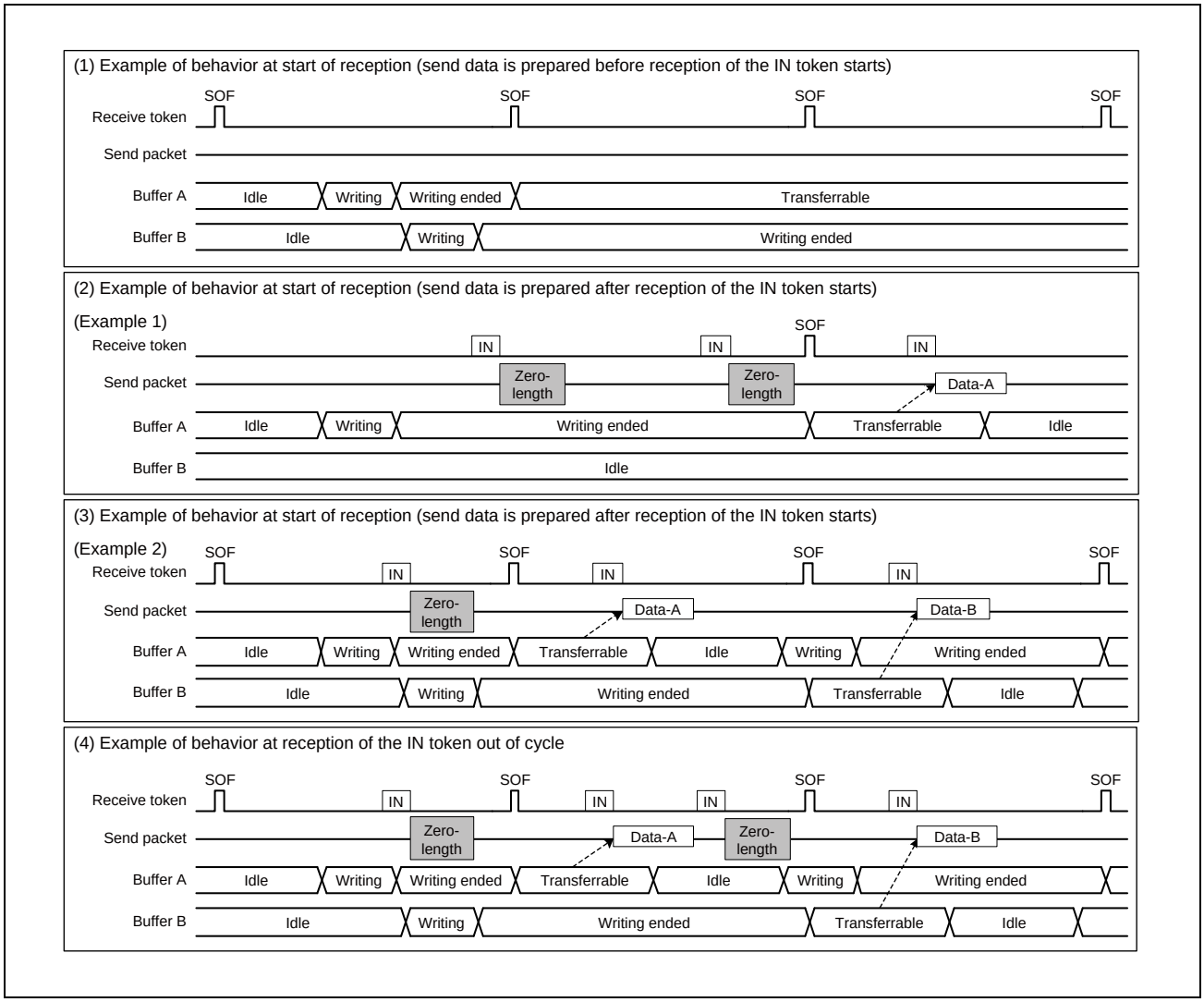


Figure 32B.11 Examples of Data Setup Function Behavior

32B.9.9.5 Transmit buffer flush for isochronous transfer

If this module does not receive an IN token in an interval frame and receives a (micro) SOF packet in the next frame during isochronous data transfer, this module handles the IN token as a corrupted token and clears the buffer that can send data to make the buffer writable.

At this time, if the buffer memory is in a double-buffer configuration and writing to both buffers has been completed, this module assumes the discarded buffer memory to be sent within the same interval frame. As a result, the buffer memory that is not discarded by reception of a (micro) SOF packet becomes to be able to transfer data.

The operation start timing of the buffer flush function varies with the value of the IITV bit.

1. If IITV is 0
   Buffer flush operation is performed from the first frame after the pipe is enabled.
2. If IITV is not 0
   Buffer flush operation is performed after the first successful transaction.

Figure 32B.12 shows an example of how the buffer flush function of this module behaves. For a token outside the set interval (token prior to the interval frame), however, this module sends the written data or a zero-length packet as an underrun error according to the data setup state.

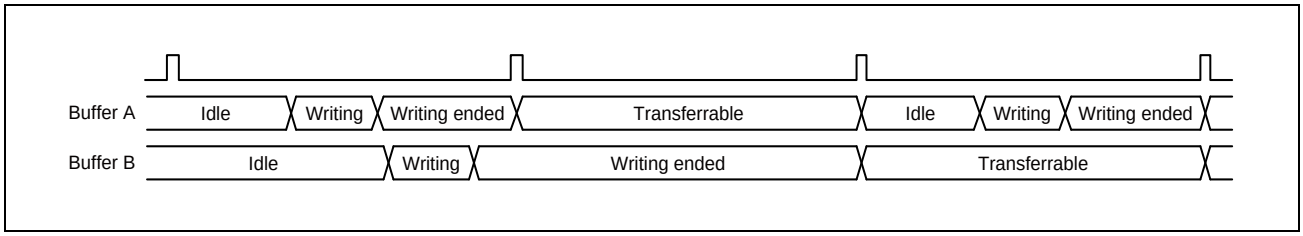


Figure 32B.12 Example of Buffer Flush Function Behavior

Figure 32B.13 shows an example of an interval error that occurs in this module.

There are five types of interval errors, as listed below. At timing (1) in the figure, an interval error occurs and the buffer flush function operates.

If an interval error occurs during IN transfer, the buffer flush function starts. If an interval error occurs during OUT transfer, an NRDY interrupt occurs.

Use the OVRN bit to determine whether an error is an NRDY interrupt (such as a receive packet error) or an overrun error.

In the figure, responses to tokens indicated as shaded boxes are made in accordance with the buffer memory state.

1. IN direction:

(a) If the buffer is ready to transfer data, data is transferred and a normal response is returned.

(b) If the buffer is not ready to transfer data, a zero-length packet is sent and an underrun error occurs.

2. OUT direction:

(a) If the buffer is ready to receive data, data is received and a normal response is returned.

(b) If the buffer is not ready to receive data, data is discarded and an overrun error occurs.

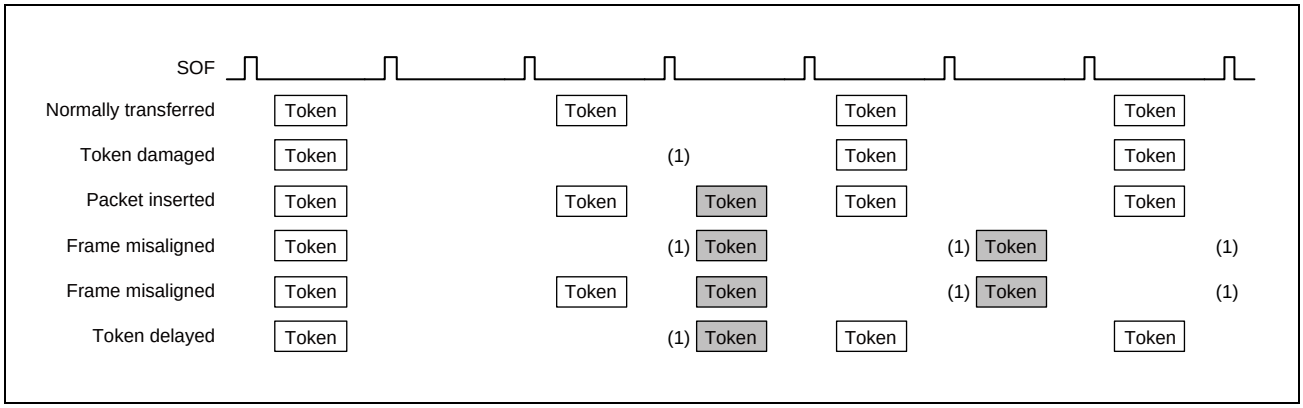


Figure 32B.13 Example of Occurrence of Interval Error (when IITV is 1)

32B.9.10 SOF Interpolation Function

If the controller cannot receive data at intervals of 1 ms (in full-speed operation) or 125 µs (in high-speed operation) due to corruption or missing of an SOF packet, the controller internally interpolates the SOF. The controller starts SOF interpolation upon receiving an SOF packet when both the USBE bit and SUSPM bit are set to 1.

The interpolation function is initialized under the following conditions:

(1) Power on reset

(2) USB bus reset

(3) Suspended state detected

The SOF interpolation operates according to the following specifications:

(1) Frame interval (125 µs or 1 ms) is based on the results of the reset handshake protocol.

(2) The interpolation function does not operate until an SOF packet is received.

(3) After receiving the first SOF packet, this module interpolates the SOF by using the 60-MHz internal clock to measure 125 µs or 1 ms.

(4) After receiving the second or a subsequent SOF packet, this module interpolates the SOF by using the previous reception interval.

(5) Interpolation is not performed in the suspended state or while a USB bus reset is being received.
(If this module enters the suspended state in high-speed operation, interpolation continues for 3 ms after receiving the last packet.)

The SOF interpolation function operates with the following functions:

(1) Updating of frame number or micro-frame number

(2) SOFR interrupt and micro SOF lock

(3) SOF pulse output

(4) Counting of isochronous transfer intervals

If an SOF packet is lost in full-speed operation, the FRNM bit in the FRMNUM register is not updated.

If a micro SOF packet is lost in high-speed operation, the UFRNM bit in the uFRMNUM register is updated.

However, if a micro SOF packet for which “micro-FRNM = 000” is set is lost, the FRNM bit is not updated. In this case, the FRNM bit is not updated even if subsequent micro SOF packets for which “micro-FRNM = 000” is not set are successfully received.

32B.9.11 Link Power Management Processing

According to the Link Power Management specification, the existing suspend state is redefined as the L2 state and a new L1 state is defined as a state where transition and resumption at a lower latency than L2 (suspend) are possible.

The table below compares the features of the L2 (suspend) state and L1 state.

Table 32B.34 Comparison Between Suspend (L2) State and L1 State

Item	L1	Suspend (L2)
Transition	LPM Transaction	3-ms idle period
Host-activated resumption	(Host) Minimum drive period specifiable by the host. Specified between 75 µs to 1.175 ms (Device) 10-µs K drive	(Host) Min. 20-ms K drive (Device) 10-ms K drive
Device-activated resumption	(Device) 50-µs K drive (Host) 60- to 990-µs K drive (Device) 10-µs K drive	(Device) 1-ms to 15-ms K drive (Host) Min. 20-ms K drive (Device) 10-ms K drive
Signaling	Low and Full Speed Idle	Low and Full Speed Idle

The following describes the processing for transition to and resumption from the L1 state.

32B.9.11.1 Descriptor

This module must return its own descriptor when receiving the GetDescriptor command.

Whether the contents of the descriptor to be returned need to be modified is dependent on whether this module responds to the transition to and resumption from the L1 state with an LPM transaction. The details are summarized in the table below.

Table 32B.35 Relationship between LPM Response and Descriptor

LPM Response	bcdUSB	Presence of USB 2.0 Ex.Desc	USB 2.0 Ex.Desc LPM	Response When LPM Is Received	Remarks
Not respond	0200	Not present	—	Not Respond	Standard action in the case where this module does not respond to LPM
	0201	Present	LPM = 0	STALL	This is when rejection of response to LPM is explicitly declared. In this case, it is necessary to send a STALL response instead of making no response.
Respond	0201	Present	LPM = 1	ACK or NYET	Standard action in the case where this module responds to LPM

Whether to respond to transition to and resumption from L1 is declared by the LPM bit of the USB 2.0 extension descriptor. To provide this module with the USB 2.0 extension descriptor, it is necessary to set the bcdUSB field of the device descriptor to 0201 or greater.

When not responding to LPM, set the bcdUSB value to 0200 without providing this module with the USB 2.0 extension descriptor. In this case, it is necessary to ignore any LPM token received.

When not responding to LPM, it is also possible to set bcdUSB to 0201 and set the LPM bit of the USB 2.0 extension descriptor to 0 (noncompliant). In this case, however, it is not allowed to ignore LPM and is necessary to send a STALL response.

When responding to LPM, set bcdUSB to 0201 and the LPM bit of the USB 2.0 extension descriptor to 1 (compliant). This grants this module to send an NYET or ACK in response to an LPM token.

32B.9.11.2 Basic processing

This module needs to execute the following processing.

1. Responds to the LPM token received from the host with “No response”, “ACK”, “NYET”, or “STALL” according to this module's own state.
2. Transitions to the L1 state if it fails to detect the retransmission of an LPM token for 8 μs after making an ACK response.
3. Detects a K drive of the host and performs resume processing to the idle state.
4. Performs resume processing to the idle state based on the Remote Wake signal.

For 1, the software specifies the response method according to the values of the L1RESPEN, L1RESPMD, and L1NEGOMD bits in the PL1CTRL register. The hardware makes the response that is designated by the software upon receiving an LPM token.

For 2, both retransmission control and transition to the L1 state are processed by the hardware. Transition to the L1 state can be identified through a DVST interrupt.

For 3, a RESM interrupt occurs on detection of host K in the L1 state.

For 4, starting the Remote Wake processing can be instructed to the hardware by setting the WKUP bit by the software. The specification stipulates that the software clears this bit on resumption from the L2 state. On the other hand, the hardware clears this bit on resumption from the L1 state.

32B.9.11.3 HIRD value negotiation

The HIRD value contained in the LPM token is the K period of the host on resumption from the L1 state.

This module can respond with ACK if the received HIRD value falls within the desired range as specified by the L1NEGOMD and HIRDTHR bits in the L1CTRL register; otherwise, this module can respond with NYET and request the host to modify the HIRD value.

NOTE

This HIRD value negotiation function must also be supported on the host side.

32B.9.12 DMA Mode

32B.9.12.1 Register mode/link mode

The DMS bit in the CHCFG_n register can be used to switch between register mode and link mode.

Table 32B.36 DMA mode settings

DMS (CHCFG)	Mode	Description
0	Register mode	Performs DMA transfer based on the values set by Next Register Set.
1	Link mode	Accesses the descriptor area, and executes DMA transfer based on the values set by descriptors. This module repeats descriptor reading and DMA transfer unless the descriptor settings are changed or the control register is used to stop the processing.

(1) Register mode

In register mode, this module executes DMA transfer based on the values set in internal registers.

Two sets of the transfer-source address, transfer-destination address, and number of bytes to be transferred can be held (in Next0 Register Set and Next1 Register Set registers). One of these Next registers can be used to execute transfer, and both Next registers can be used to execute continuous transfer.

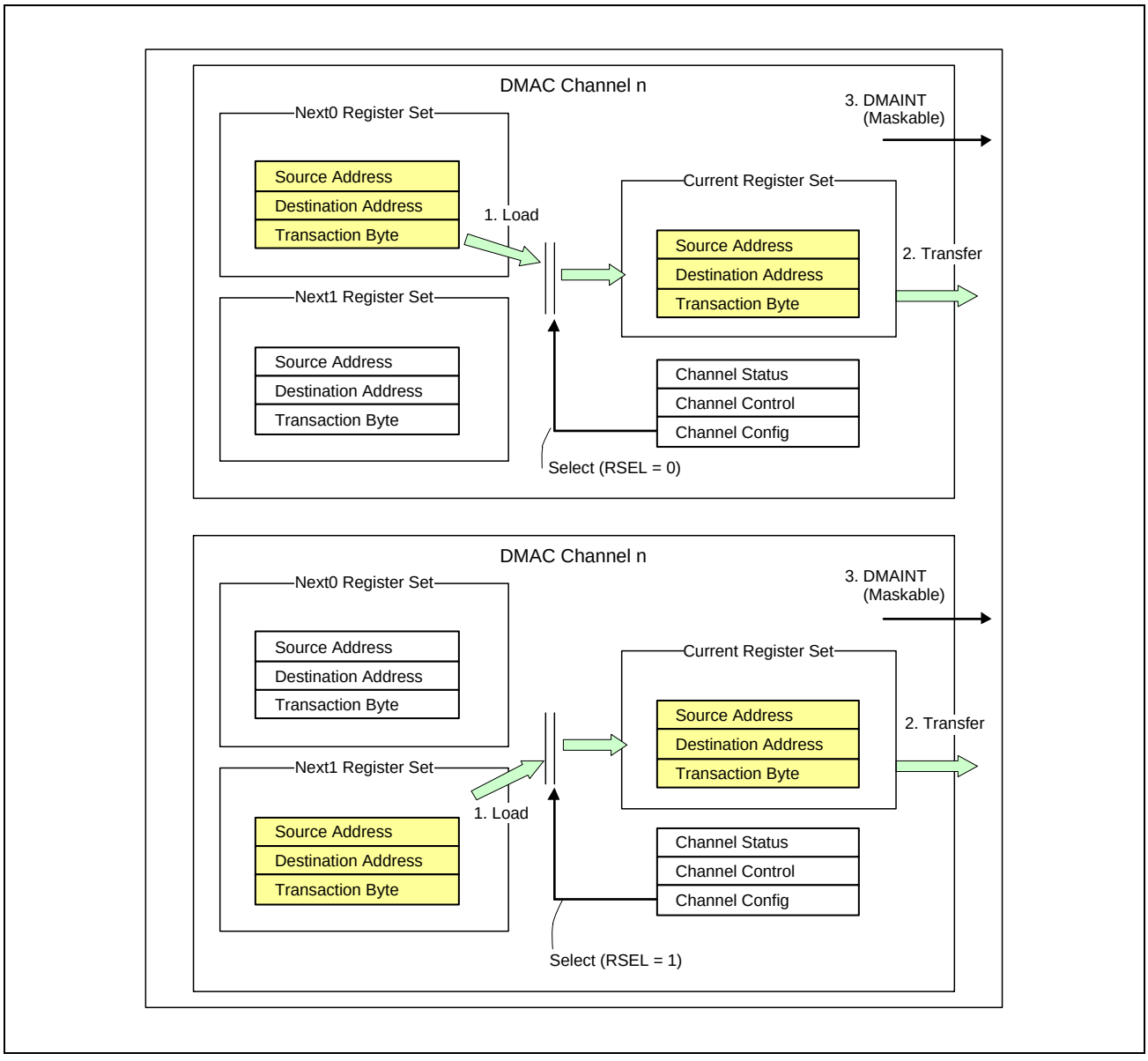


Figure 32B.14 Overview of Normal Behavior of Register Mode

The upper part of the above figure indicates a case when Next0 Register Set is processed. The lower part of the above figure indicates a case when Next1 Register Set is processed.

(a) Operation flow of register mode

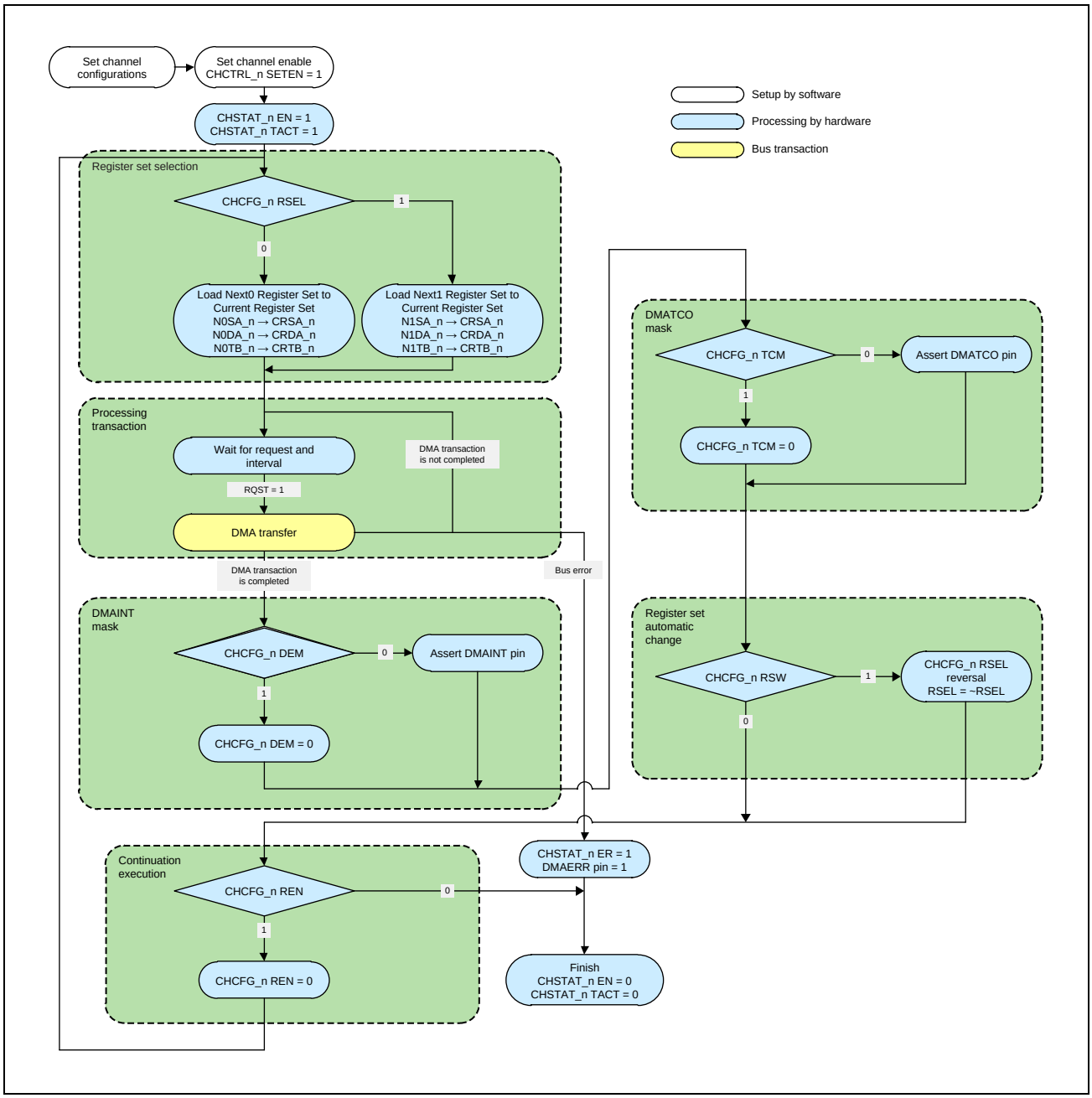


Figure 32B.15 Register Mode Operation Flow

Description of the register mode operation flow:

1. Channel setting
   The Next0 Register Set or Next1 Register Set register is set (transfer-destination address, transfer-source address, and total number of bytes to be transferred). In addition, the FIFO channel, volume of transfer data, and other items are set for USB control that is used for the Channel Register Set register.
   No software settings are required in OUT transfer (reception in peripheral mode) or in IN transfer (transmission in peripheral mode) because the hardware sets the transfer source or destination address according to the setting of the CURPIPE bits in the DxFIFOSEL register.
   (see **Section 32B.9.13, DMA Transfer**).

2. Register set selection
   When 1 is written to the SETEN bit in the CHCTRL_n register, the EN and TACT bits in the CHSTAT_n register are set to 1. As a result, the values set by the Next Register Set register selected by the RSEL bit in the CHCFG_n register are loaded to the Current Register Set register.

3. DMA transaction
   A DMA transaction is executed based on the values that are set. For details about transfer, **Section 32B.9.13, DMA Transfer**.

4. The USBFDMAmn masking
   The USBFDMAmn interrupt is masked depending on the value of the DEM bit in the CHCFG_n register. If DEM = 1, the USBFDMAmn interrupt is masked, and the DEM bit is automatically cleared to 0.

5. DMATC masking
   DMATC from DMAC control to USB control is masked depending on the value of the TCM bit in the CHCFG_n register. If TCM = 1, DMATC is masked, and the TCM bit is automatically cleared to 0.

6. Automatic register set switchover
   Whether the current Next register set is to be switched to the other Next register set is determined by the value of the RSW bit in the CHCFG_n register.

7. Continuation of execution
   Whether to continue DMA transfer is determined by the value of the REN bit in the CHCFG_n register. If REN = 0, the EN and TACT bits in the CHSTAT_n register are cleared to 0, and DMAC operation stops. If REN = 1, DMAC operation continues, and the REN bit is automatically cleared to 0.

(b) Register mode setting

- Register mode setting

  The register set to be processed is selected.

Table 32B.37 Register Mode Setting

DMS (CHCFG_n)	RSEL (CHCFG_n)	Description
0	0	Processes Next0 Register Set.
	1	Processes Next1 Register Set.

- USBFDMAmn masking

  The USBFDMAmn interrupt can be masked.

Table 32B.38 USBFDMAmn Mask Setting

DEM (CHCFG_n)	Description
0	Asserts the USBFDMAmn interrupt when the DMA transaction is completed.
1	Does not assert the USBFDMAmn interrupt even when the DMA transaction is completed. After the DMA transaction is completed, the DEM bit is cleared to 0.

- DMATC mask setting

  DMATC from DMAC control to USB control can be masked.

Table 32B.39 DMATC Mask Setting

DEM (CHCFG_n)	Description
0	Asserts DMATC when the DMA transaction is completed.
1	Does not assert DMATC even when the DMA transaction is completed. After the DMA transaction is completed, the TCM bit is cleared to 0.

- Automatic transaction execution for a register set

  After a DMA transaction finishes, another DMA transaction can be executed.

Table 32B.40 Automatic Execution Setting for a Register Set

REN (CHCFG_n)	Behavior	Remarks
0	The EN bit is cleared and DMA operation is terminated when the DMA transaction for the register set that is set by RSEL finishes.	Use this setting to execute a DMA transaction only once.
1	After a DMA transaction finishes, DMA transfer of the contents of the next register set continues. The REN bit is cleared to 0 when continuous transfer is successful.	Use this setting to continue processing of register set contents.

- Automatic register set switchover setting

  After a DMA transaction finishes, the next register set to be processed can be switched.

Table 32B.41 Automatic Execution Setting for a Register Set

RSW (CHCFG_n)	Behavior	Remarks
0	The register set is not switched when a DMA transaction finishes.	Use this setting to use only one register set.
1	When REN = 1 and a DMA transaction finishes, the RSEL setting is automatically reversed to select the other register set.	Use this setting to switch the register set.

(c) Example of setting the register mode

- Example of setting the register mode when using only the Next0 register set

Table 32B.42 Register Mode Setting Example

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	TCM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0	0	0	0	0	0
(register mode)	(Next0)	(not masked)	(not masked)	(not switched)	(continuous execution disabled)

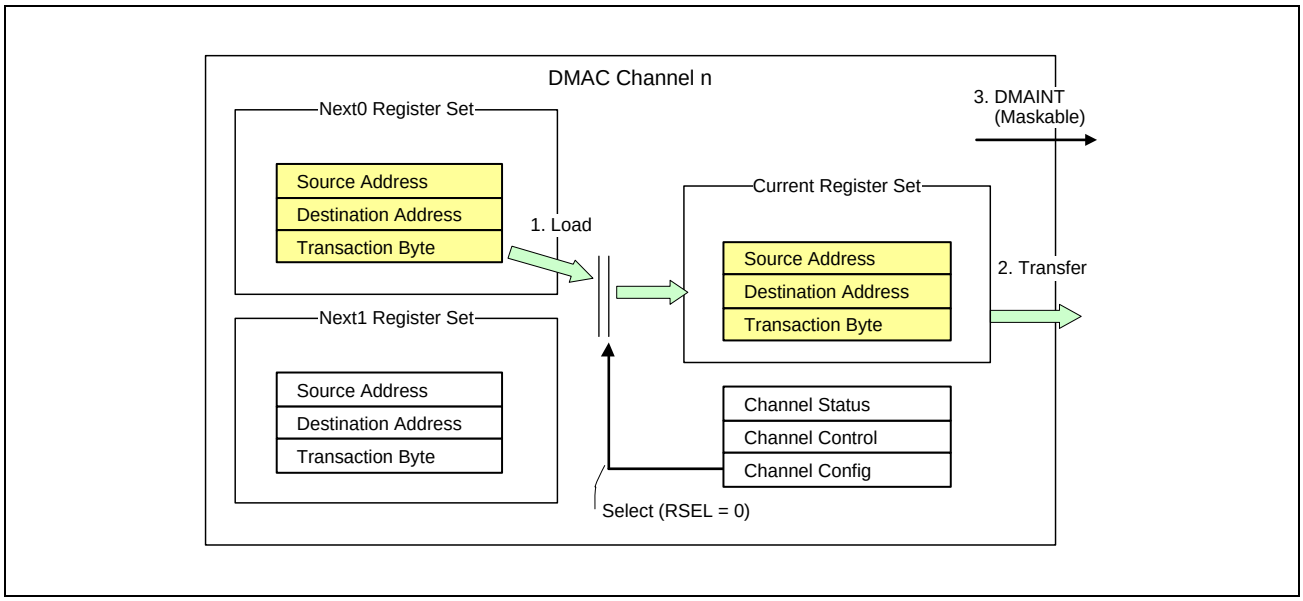


Figure 32B.16 Register Mode Setting Example 1

1. By writing 1 to the SETEN bit in the CHCTRL_n register, the EN bit in the CHSTAT_n register is set to 1. As a result, the contents of Next0 Register Set are loaded to Current Register Set.
2. A DMA transaction is executed based on the values of Current Register Set and Channel Register Set.
3. Because the DEM bit in the CHCFG_n register is 0, the USBFDMAmn interrupt is asserted after the DMA transaction finishes.
4. Because the TCM bit in the CHCFG_n register is 0, DMATC is asserted after the DMA transaction finishes.
5. Because the REN bit in the CHCFG_n register is 0, the EN bit in the CHSTAT_n register is cleared to 0, and the operation ends.

- Example of setting the register mode when using two register sets continuously

Table 32B.43 Automatic Register Set processing Setting

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	TCM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0 (register mode)	0 (Next0)	1 (masked)	0 (not masked)	1 (switched)	1 (continuous execution enabled)

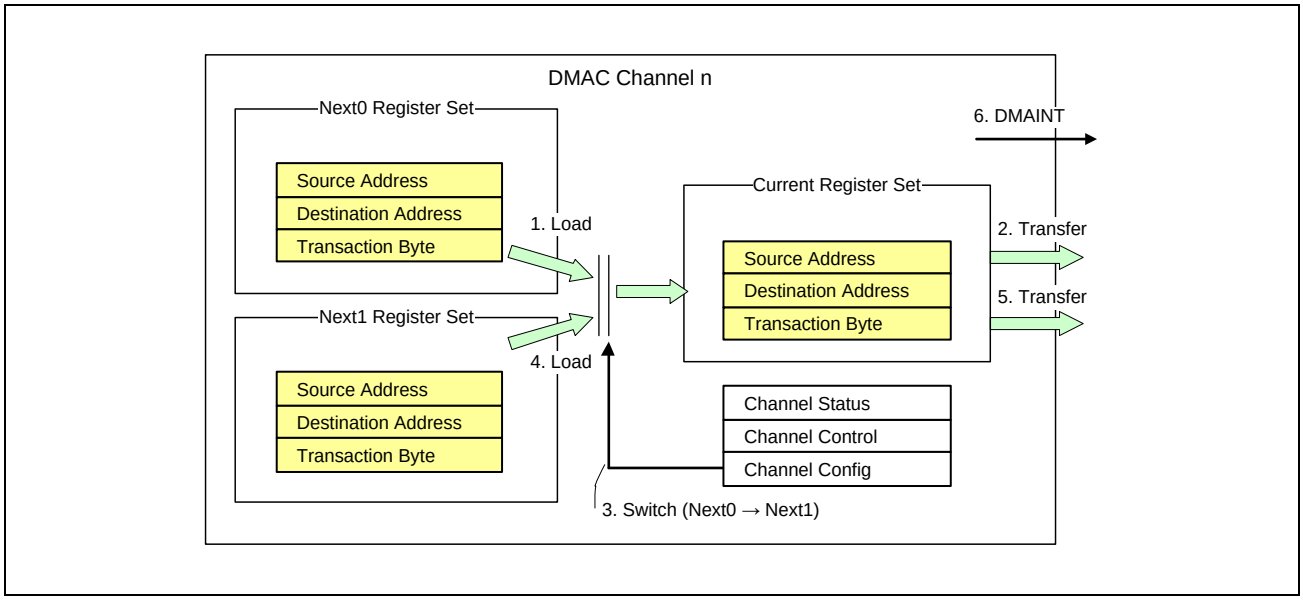


Figure 32B.17 Register Mode Setting Example 2

1. By writing 1 to the SETEN bit in the CHCTRL_n register, the EN bit in the CHSTAT_n register is set to 1. As a result, the contents of Next0 Register Set are loaded to Current Register Set.
2. A DMA transaction is executed based on the values of Current Register Set and Channel Register Set.
3. Because the DEM bit in the CHCFG_n register is 1, the USBFDMAmn interrupt is not asserted after the DMA transaction finishes.
4. Because the REN bit in the CHCFG_n register is 1, operation continues. The REN bit is automatically cleared to 0.
5. Because the RSW bit in the CHCFG_n register is 1, the next register set to be processed is switched (RSEL = 0 to 1).
6. The contents of Next1 Register Set are loaded to Current Register Set.
7. A DMA transaction is executed based on the values of Current Register Set and Channel Register Set.
8. Because the DEM bit in the CHCFG_n register is 0, the USBFDMAmn interrupt is asserted after the DMA transaction finishes.
9. Because the TCM bit in the CHCFG_n register is 0, DMATC is asserted after the DMA transaction finishes.
10. Because the REN bit in the CHCFG_n register is 0, the EN bit in the CHSTAT_n register is cleared to 0, and the operation ends.

(2) Link mode

In link mode, this module reads the value set in a descriptor placed in an external storage area to execute a DMA transaction. In DMAC, there are Next Link Address (NXLA_n) and Current Link Address (CRLA_n) registers for each channel. The Next Link Address (NXLA_n) register is used to set the address of the descriptor to be read the next time. The Current Link Address (CRLA_n) register is used to display the descriptor address for the current DMA transaction.

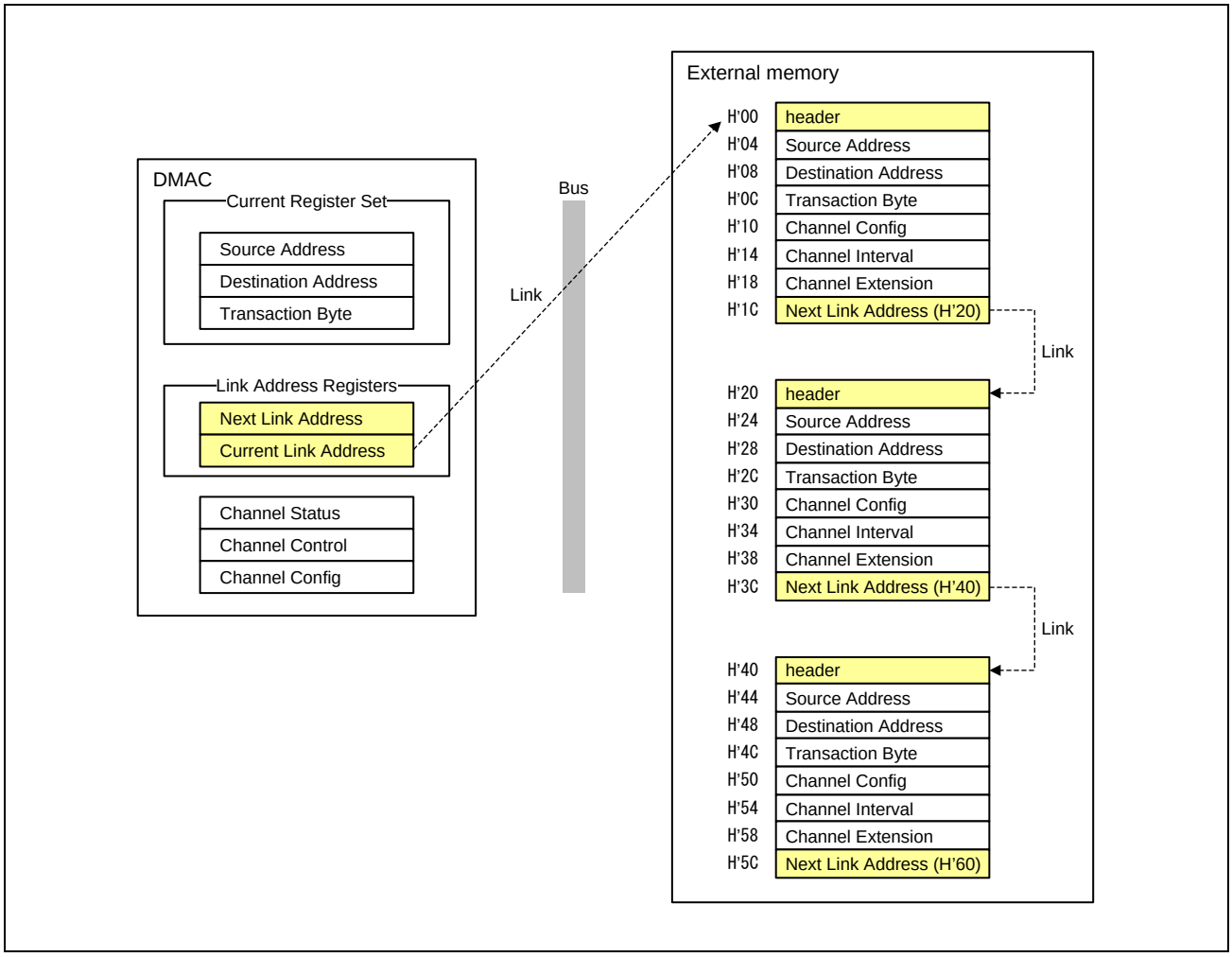


Figure 32B.18 Overview of Link Mode

(a) Operation flow of link mode

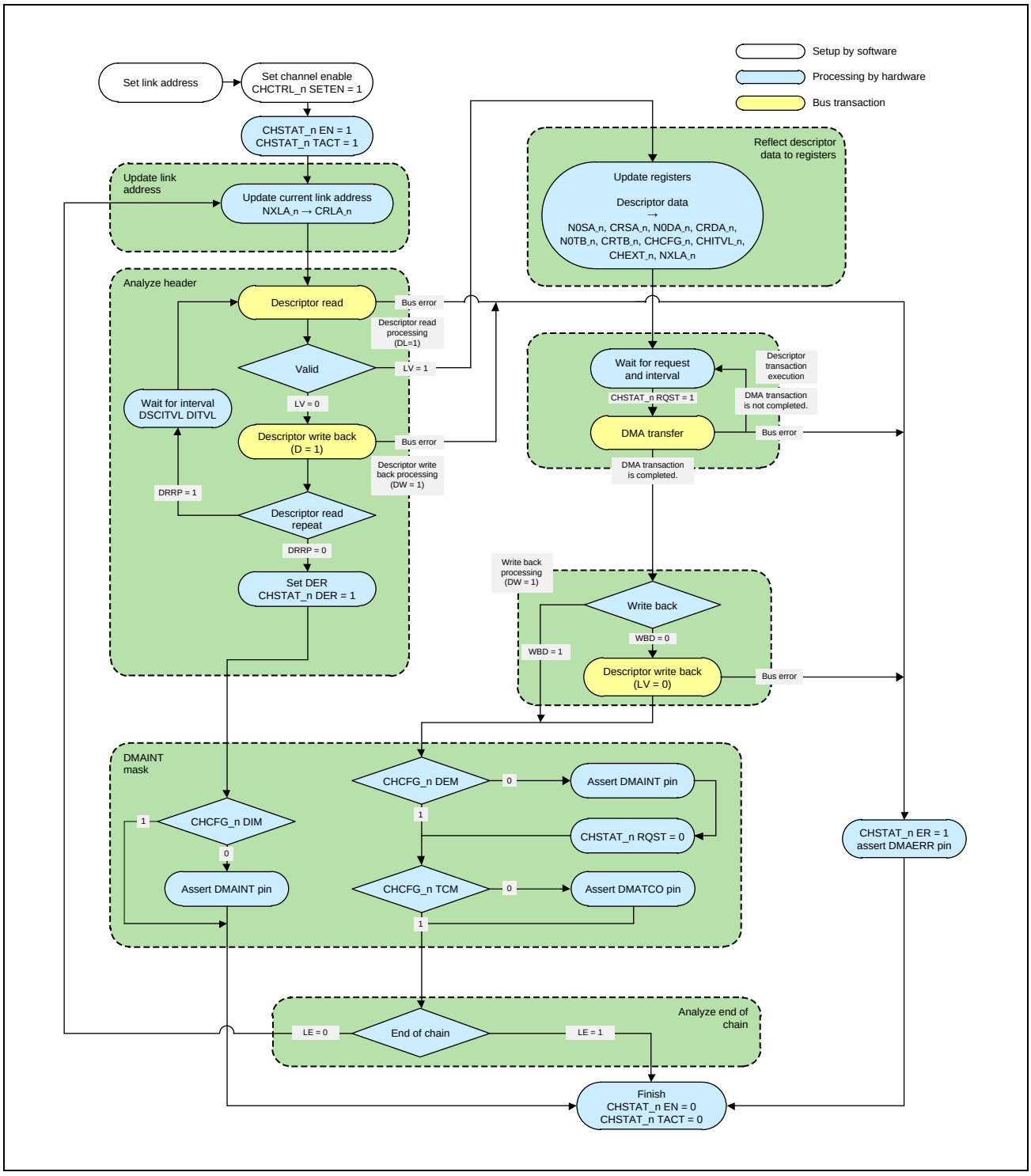


Figure 32B.19 Link Mode Operation Flow

Description of the link mode operation flow:

1. Channel setting
   The beginning address of the link destination is set in the NXLA_n register.

2. Link address updating
   If 1 is written to the SETEN bit in the CHCTRL_n register, the EN and TACT bits of the CHSTAT_n register are set to 1. As a result, the address set in the NXLA_n register is loaded to the CRLA_n register.

3. Descriptor reading and header judgment
   A read of the descriptor starts, and DMAC checks the contents of header. If LV = 0, this module writes 1 back to the D bit of header. After that, if the DRRP bit in the CHCFG_n register is 1, this module waits for the time intervals set by the DSCITVL register, and then reads the same descriptor again. If DRRP = 0, the DER bit in the CHSTAT_n register is set to 1, and this module is placed in the end state (both EN and TACT bits in the CHSTAT_n register are 0). At this time, if the DIM bit in the CHCFG_n register is 0, this module asserts the USBFDMAmn interrupt.

4. Descriptor setting
   If LV = 1, the data read from the descriptor is loaded to Current Register Set and Channel Register Set. In addition, the next link target is loaded to the NXLA_n register.

5. DMA transaction
   A DMA transaction is executed based on the values that are set. For details about transfer, see **Section 32B.9.13, DMA Transfer**.

6. Header write-back
   If WBD of header is 0, DMAC writes LV = 0 to the header area.

7. USBFDMAmn masking
   If the DEM bit in the CHCFG_n register is 0, this module asserts the USBFDMAmn interrupt.

8. DMATC masking
   If the TCM bit in the CHCFG_n register is 0, this module asserts DMATC.

9. Link end judgment
   If LE of header is 1, the EN and TACT bits in the CHSTAT_n register are cleared to 0, and DMAC terminates operation. If LE is 0, this module updates Current Register Set, and then restarts reading the next descriptor.

(b) Register setting

- Link mode setting

  To use link mode, set the DMS bit in the CHCFG_n register to 1.

Table 32B.44 Link Mode Setting

DMS (CHCFG_n)	Description
1	This module operates in link mode. The setting of this bit cannot be changed by using a descriptor.

- Link address setting

  The Next Link Address (NXLA_n) and Current Link Address (CRLA_n) registers are used to indicate a link target. Before starting link mode, set the link target in the NXLA_n register.
  After reading a descriptor, this module updates the NXLA_n register to the next link. Note that the CRLA_n register indicates the address of the link target that is being executed.

Table 32B.45 Link Address Register Set

Register	Description
Next Link Address Register (NXLA_n)	This register is used to set and display the next link target. Before starting link mode, set the address of the link target in this register.
Current Link Address Register (CRLA_n)	This register is used to display the link target that is being executed. This is a read-only register.

(c) Descriptor setting

DMAC supports multiple descriptor formats.

A switchover between formats is specified by using the DSCFM field of bits [31:28] of the 1st word (header) of the descriptor.

The following table shows the relationship between DSCFM values and descriptor formats.

Table 32B.46 Descriptor Formats

DSCFM	Descriptor Size	Next Link Address	Channel Extension	Channel Interval	Channel Config	Transaction Size	Destination Address	Source Address	header
3	4 words	✓	— (reload)	— (reload)	— (reload)	— (header)	✓	✓	✓ (with STS)
1	8 words	✓	✓	✓	✓	✓	✓	✓	✓ (no STS)
Other than the above	If DSCFM is set to a value that is not 1 or 3, operation cannot be guaranteed. Make sure that DSCFM is set to 1 or 3.								

Table 32B.47 Explanation of the Marks in **Table 32B.46**

Field	Mark	Description	Remarks
Header	✓ (with STS)	The STS field of bits [15:0] in the header is valid. The value set in the STS field is used as the total number of transfer bytes (Transaction Size).	—
	✓ (no STS)	The STS field of bits [15:0] in the header is invalid. The value of "Transaction Size" in the descriptor is used as the total number of bytes.	
Source Address	✓	Specify the source address.	—
Destination Address	✓	Specify the destination address.	—
Transaction Size	✓	Specify the transaction size.	—
	— (header)	Omit the transaction size. The value set in the STS field is used as the total number of transfer bytes (Transaction Size)	Because the STS field is of 16 bits, a maximum of 65,535 bytes can be set.
Channel Config Channel Interval Channel Extension	✓	Specify Channel Config, Channel Interval, and Channel Extension.	—
	— (reload)	Omit Channel Config, Channel Interval, and Channel Extension. The previous settings (the values of the CHCFG_n, CHITVL_n, and CHEXT_n registers of the last time) are inherited.	—
Next Link Address	✓	Specify the next descriptor address (Next Link Address) to be read after DMA transfer of the descriptor.	—

DMAC sequentially interprets data read from descriptors. If the number of words specified for DSCFM is less than 8, place the data of descriptors that are indicated by "✓" in **Table 32B.46** on memory.

No software settings are required in OUT transfer (reception in peripheral mode) or in IN transfer (transmission in peripheral mode) because the hardware sets the transfer source or destination address according to the setting of the CURPIPE bits in the DxFIFOSEL register.

Table 32B.48 Example of Placing Descriptors

DSCFM	Address							
	Link Address + H'1C	Link Address + H'18	Link Address + H'14	Link Address + H'10	Link Address + H'0C	Link Address + H'08	Link Address + H'04	Link Address + H'00
H'3	—	—	—	—	Next Link Address	Destination Address	Source Address	header
H'1	Next Link Address	Extension	Interval	Config	Transaction Byte	Destination Address	Source Address	header

- header

The header area provides the descriptor status and other information as shown below.

DMAC reads this area before DMA transfer in link mode starts. After a DMA transaction terminates, DMAC writes the transfer status back to this area.

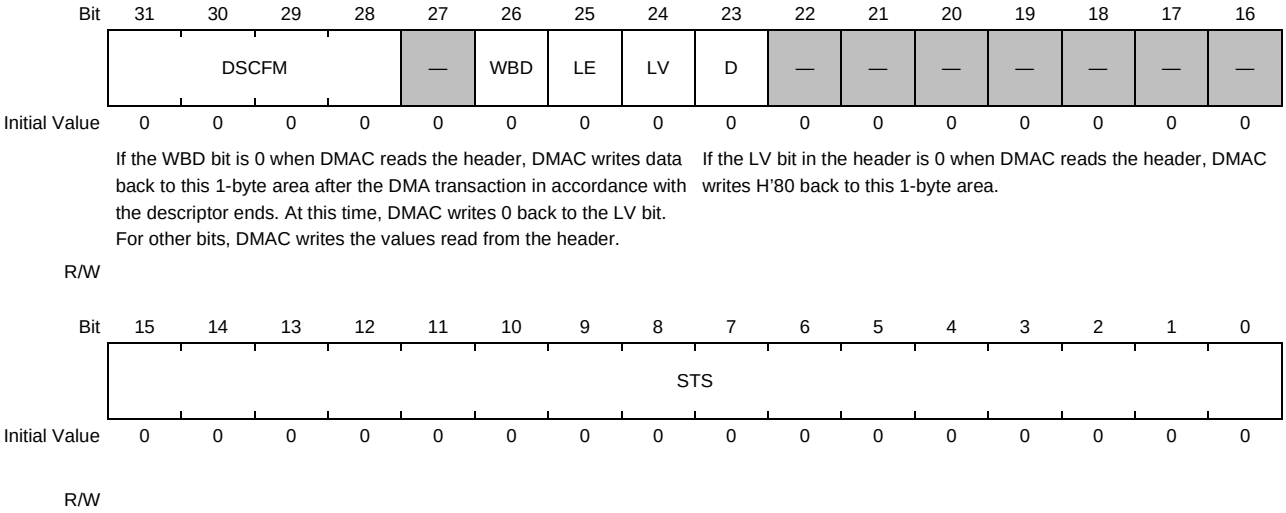


Figure 32B.20 Header Area

Table 32B.49 Header Area (1/2)

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	DSCFM			Descriptor Format Specifies the descriptor format (length and combination of descriptors). For details see **Table 32B.46**.
27	—			Reserved area. Set 0.
26	WBD			Write Back Disable Masks a write-back operation for the LV bit. If this bit is 1, DMAC does not perform a write-back operation. 0: Writes 0 back to the LV bit. 1: Does not perform a write-back operation for the LV bit.
25	LE			Link End Indicates that the link will end with the DMA transaction for this descriptor. Set this bit to 1 to indicate the end of link. 0: The link continues. 1: The link ends.

Table 32B.49 Header Area (2/2)

Bit	Bit Name	Initial Value	R/W	Description
24	LV			Link Valid Indicates that this descriptor is valid. If WBD = 0, after DMAC executes the DMA transaction written in the descriptor, DMAC writes 0 to this bit. When header is set, set 1 to this bit. 0: This descriptor is invalid. 1: This descriptor is enabled.
23	D			Descriptor Error Indicates a descriptor access error. If LV is 0 when the descriptor is read, DMAC writes 1 back to this bit. 0: A descriptor error has not occurred. 1: LV was 0 when the descriptor was read.
22 to 16	—			Reserved area. Set 0.
15 to 0	STS			Short Transaction Size If DSCFM is 3, the transaction size is set (in bytes). The maximum number of transfer bytes that can be set is 65,535. If DSCFM is 3, Do not set 0 for STS. If 0 is set, operation cannot be guaranteed.

If descriptors are added sequentially while DMAC is operating, the access that the CPU sets 0 to the LV bit and the access that DMAC writes 1 back to the D bit might contend with each other. If this contention occurs, prior-written data is overwritten by latter-written data.

To prevent this problem from occurring, DMAC performs a write-back operation for the D bit in a byte-write manner. Therefore, the CPU must also set LV to 1 in a byte-write manner. Because the byte lanes for the D and LV bits are different, by writing data to different areas, occurrence of this problem can be prevented.

- Settings of descriptors other than header

  The specifications of data of the descriptors other than header are the same as the specifications of internal registers.

- Settings specified when descriptors are accessed

  The MHPROT pin output can be set for the LWPR and LDPR fields of the DCTRL register when descriptors are accessed. Set it according to the access target in which descriptors are deployed.

- Descriptor areas and DMA transfer areas

  The following provides an overview of the descriptor areas and DMA transfer areas that are accessed by DMAC.

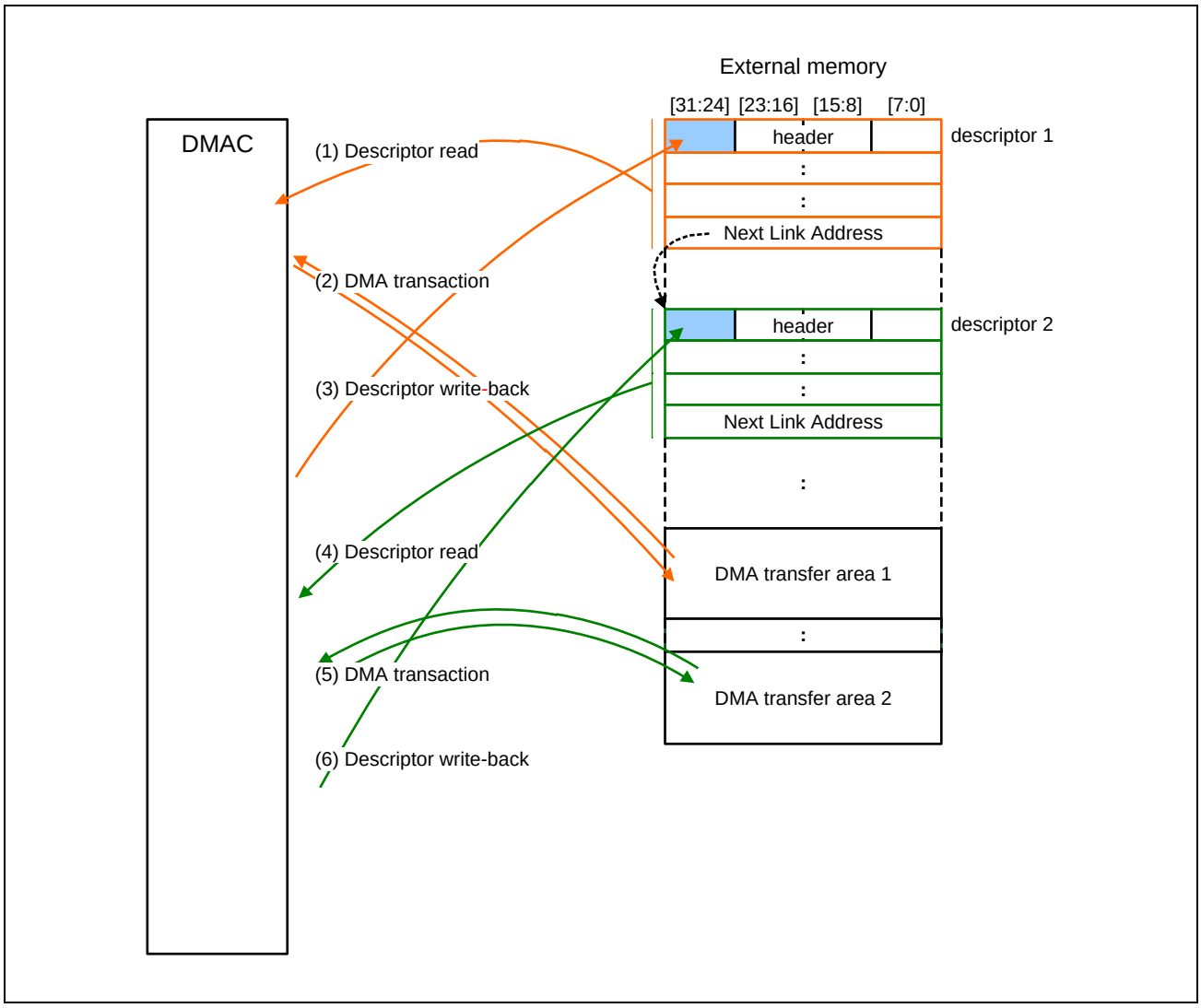


Figure 32B.21 Overview of Descriptor Areas and DMA Transfer Areas

(1) Descriptor read
DMAC loads a value from the internal NXLA_n register to the CRLA_n register, and then reads a descriptor from the external memory space indicated by the CRLA_n register (descriptor 1).

(2) DMA transfer
If the LV bit of the header descriptor is 1, DMAC performs a DMA transfer in accordance with the descriptor information.

(3) Descriptor write-back
After performing a DMA transfer of data by the number of bytes that are set, if the WBD bit in header is 0, DMAC performs write-back for bits [31:24] in header of Descriptor 1. For the LV field, 0 is written back. For other fields, the values read in (1) are written back on a byte size basis.

(4) Descriptor read
If the value of the LE bit in the header descriptor that was read previously (in (1)) is 0, DMAC reads the next descriptor from the address (descriptor2) indicated by Next Link Address in the current descriptor.

(5) DMA transfer
If the LV bit in the header descriptor is 1, DMAC performs a DMA transfer in accordance with the descriptor information.

(6) Descriptor write-back
After performing a DMA transfer of data by the number of bytes that are set, if the WBD bit in header is 0, DMAC performs write-back for bits [31:24] in header of Descriptor 2. For the LV field, 0 is written back. For other fields, the values read in (4) are written back on a byte size basis.

(Steps (4) to (6) are repeated.)

If LE = 1 and WBD = 0 in header, DMAC performs a DMA transfer with the descriptor settings, writes 0 back to the LV bit in header, and then terminates processing.

If LE = 1 and WBD = 1 in header, DMAC performs a DMA transfer with the descriptor settings, and then terminates processing (without performing write-back).

If LV = 0 in header, DMAC writes 1 back to the D bit in header, and then, if the DRRP bit in the CHCFG_n register is 1, DMAC waits for the number of intervals specified by the DITVL field of the DSCITVL_n register, and then reads the descriptor again. If DRRP = 0, DMAC terminates processing.

- Notes on descriptors
  - In link mode, settings can be changed by reading descriptors. However, it is impossible to synchronize the setting change times and hardware requests. Therefore, to use hardware requests, before setting the SETEN bit in the CHCTRL_n register, set the AM, LVL, HIEN, LOEN, and SEL bits in the CHCFG_n register. Note that these setting bits must not be changed in descriptors.
  - Descriptors cannot be used to change the DMS field in the CHCFG_n register (DMAC is always placed in link mode). Although descriptors can be used to change the settings of the REN, RSW, and RSEL fields in the CHCFG_n register, changes of those fields do not affect operation.
  - The descriptor can be initialized by overwriting the memory area corresponding to the descriptor you intend to initialize while the DMAC is not operating. The DMAC determines whether or not the descriptor is valid by referring to the DSCFM field and LV bit in the header. Accordingly, set the areas in memory corresponding to the DSCFM field and LV bit to 1 or 3, and to 1, respectively, before enabling DMAC operation.
  - To set the next descriptor on memory while DMAC is operating, make sure that 1 is written to the LV bit after the descriptors subsequent to header (Source Address, Destination Address, ..., Next Link Address) are set. If this is not performed and descriptor setting by the CPU and descriptor reading by DMAC contend, DMAC performs a DMA transfer using the previous values of those descriptors (Source Address, Destination Address, ..., Next Link Address).
  - To leave the write-back information for the D bit of header, make sure that 1 is written to the LV bit of header in a byte-access manner.

(d) Link configuration example

In link mode, descriptors can be configured as shown below.

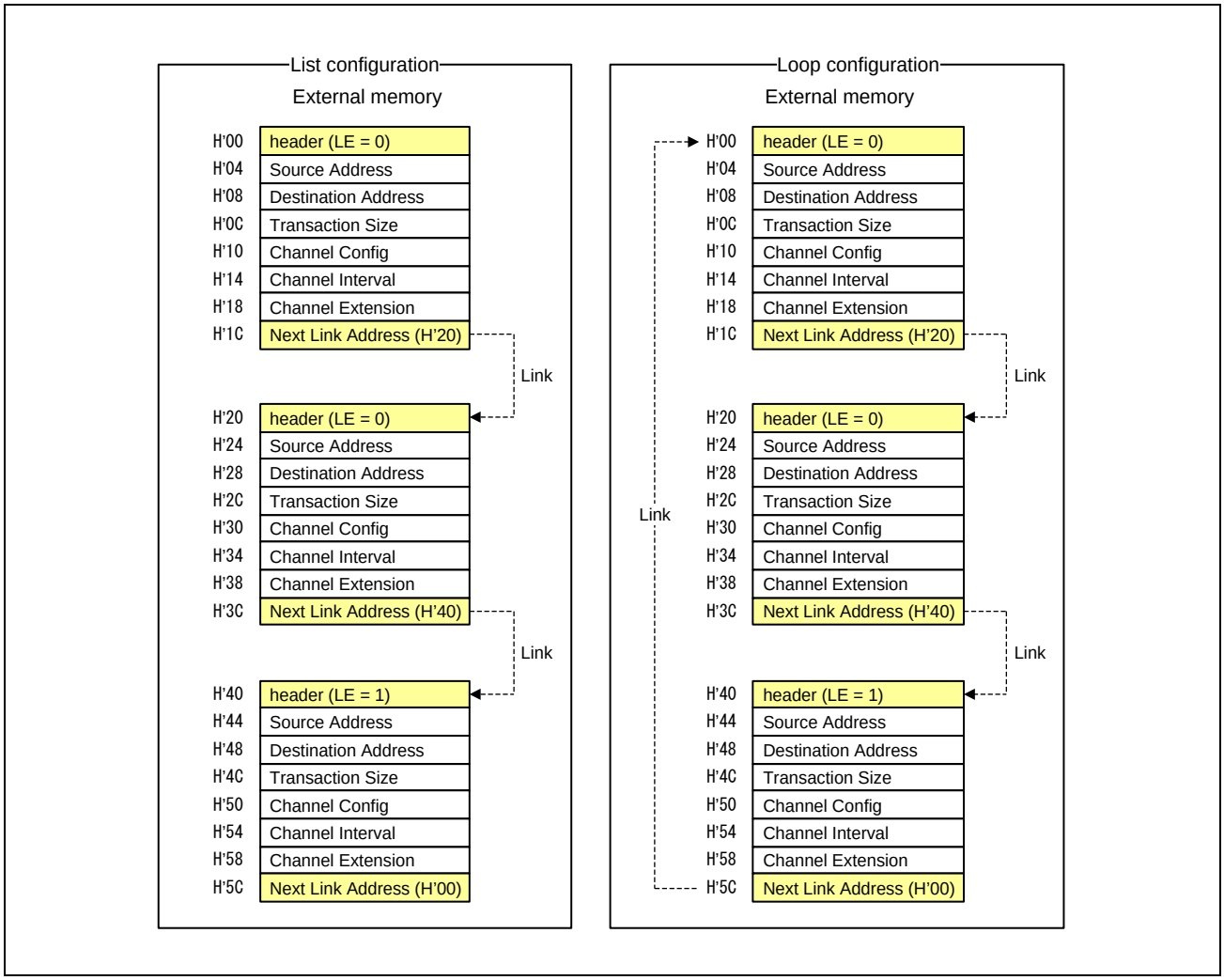


Figure 32B.22 Link Mode Configuration Example

- List configuration

  The link ends by setting 1 for the LE bit in the header of the last descriptor.

- Loop configuration

  A loop of descriptors can be created by setting the link target of the last descriptor to the address of the first descriptor. To end the loop, change the value of the LE bit of header to 1 or use the transfer interrupt procedure.

32B.9.12.2 Write only mode

Write-only mode is enabled by setting 1 for the WONLY bit in the CHCFG_n register.

Table 32B.50 Write-Only Mode Setting

WONLY (CHCFG)	Mode	Description
0	Normal mode	A DMA transfer is performed using the values set in Next Register Set.
1	Write-only mode	A DMA write transfer is performed without performing a DMA read transfer.

In write-only mode, no DMA read transfer is performed (note that descriptors are read in the same way as in normal mode). In register mode, the values set in the NxSA_n register (if RSEL = 0, x = 0; if RSEL = 1, x = 1) are used as write data. In link mode, the values of the SA fields of descriptors are used as write data.

Use this mode to, for example, initialize the memory area.

32B.9.13 DMA Transfer

This chapter describes the basic operation of DMA transfer.

32B.9.13.1 Transfer modes

DMAC supports only single transfer mode.

Upon receiving a DMA transfer request from USB control, DMAC executes a single DMA transfer on the side (source or destination) indicated by the REQD bit in the CHCFG_n register. DMAC then asserts internal DMA permission from internal USB control to DMAC control. DMAC performs a single transfer each time a transfer is received. DMAC continues this operation by the transfer size loaded to the CRTB_n register (arbitration between channels is performed for each DMA transfer).

The timing of internal DMA permission from internal USB control to DMAC control differs depending on the setting of the REQD bit in the CHCFG_n register and the setting of the transfer size (DDS[2:0] and SDS[2:0] in the CHCFG_n register). For details, see **Section 32B.9.13.7, Operational difference depending on the transfer size**.

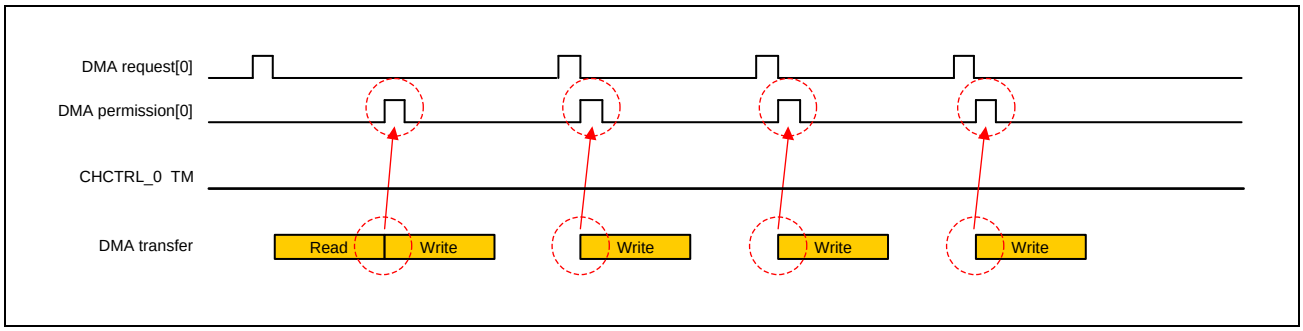


Figure 32B.23 Single Transfer Mode (REQD = 1, SDS > DDS)

32B.9.13.2 DMA channel priority control

DMAC supports fixed-priority mode and round-robin mode as methods of arbitration between channels. The mode is selected by using the PR bit in the DCTRL register. If the PR bit is 0, fixed-priority mode is selected. If the PR bit is 1, round-robin mode is selected.

Table 32B.51 Priority Control Setting

Mode	PR (DCTRL)	Description	Remarks
Fixed-priority	0	Controls requests based on fixed priority (CH0 > CH1 > ...).	Use this mode if channels have priority.
Round-robin	1	Controls requests in a round-robin manner.	Use this mode to execute requests equally.

(1) Fixed-priority mode

In fixed-priority mode, a fixed priority level is assigned to each channel as shown below.

(High) CH0 > CH1 (Low)

If DMA transfer requests simultaneously occur over multiple channels, the DMA transfer over the channel whose number is smallest is performed first.

The transfer over channel 0 is performed first. However, while a transfer switches to another transfer over channel 0, a transfer over the channel with the next highest priority level is performed in order to increase the bus usage rate.

(2) Round-Robin Mode

In round-robin mode, each time a transfer over a channel is received, the priority level of the channel that was used for the previous transfer is changed to the lowest level.

In the status immediately after the mode is reset, channels are assigned priority levels in the same way as fixed-priority mode as shown below.

(High) CH0 > CH1 (Low)

In this status, if a transfer request for DMA channel 0 does not occur and a transfer request for DMA channel 1 occurs, the transfer over DMA channel 1 is performed. When the transfer finishes, the channel priority is changed as follows.

(High) CH1 > CH0 (Low)

The following shows an example of DMA transfer in round-robin mode.

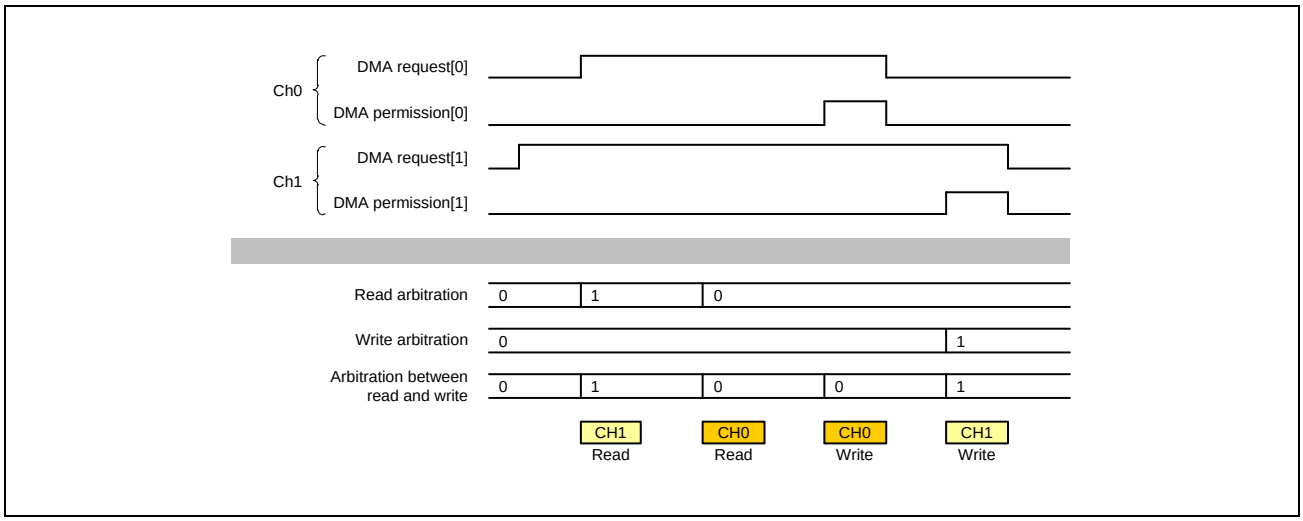


Figure 32B.24 Example of Operation in Round-Robin Mode (with 4 channels, REQD = 1)

DMAC internally performs arbitration between read channels and arbitration between write channels, further performs arbitration between the arbitration results, and then issues bus access.

32B.9.13.3 Forced sweeping request

When a forced sweeping request is entered, DMAC transfers the data that is left untransferred in the buffer to the destination address. After the sweep processing finishes, DMAC continues DMA transfer.

The following provides notes on forced sweeping requests:

(1) If a forced sweeping request and a transfer request from USB control contend, DMAC performs forced sweeping first, and then performs a DMA transfer.

(2) If the USB-control-side system is the destination (REQD bit in the CHCFG_n register is 1) register REQD = 1, buffer overflow or another error might occur on the destination unit because data is transferred although no DMA transfer request is made on the USB control side. Therefore, the specifications physically prohibit DMAC from using forced sweeping if REQD = 1.

(3) The difference from ordinary sweeping mode described in **Section 32B.9.13.8(3)(b), Transfer suspension (buffer sweeping enabled: SBE = 1)** (EN is cleared by setting 1 for the SBE bit in the CHCFG_n register) is as follows: DMAC stops operation after writing data in the buffer in ordinary sweeping mode, whereas DMAC can continue DMA transfer after sweeping the buffer in forced sweeping mode.

(1) Forced software sweeping request

The SETSSWPRQ bit in the CHCTRL_n register determines whether software-based forced sweeping requests can be used. To perform a forced sweeping request, write 1 to the SETSSWPRQ bit. DMAC then outputs the data in the buffer to the destination.

32B.9.13.4 DMA transfer completion interrupt (USBFDMAmn)

USBFDMAmn (m, n = 0, 1) is an interrupt signal that indicates termination of a DMA transaction.

If a transfer for the total number of transfer bytes loaded to the CRTB_n register is completed by an OKAY response, the END bit in the CHSTAT_n register is set to 1. At this time, if the DEM bit in the CHCFG_n register is 0, DMAC generates a USBFDMAmn (m, n = 0, 1) interrupt.
(If a write-back operation is performed in link mode, the interrupt is generated after the write-back operation finishes.)

In link mode, when the DRRP bit in the CHCFG_n register is 0, if the LV bit of the header of the descriptor that is read is 0, the DER bit in the CHSTAT_n register is set to 1. At this time, if the DIM bit in the CHCFG_n register is 0, DMAC generates a USBFDMAmn interrupt.

Use this signal to detect a transfer completion interrupt performed by the interrupt controller.

Table 32B.52 USBFDMAmn Assertion Conditions

Cause	Condition	INT_DMA[n] mask signal
DMA transaction ended	A transfer of data by the number of transfer bytes loaded to the CRTB_n register is completed by an OKAY response (if a write-back operation is performed in link mode, after the operation finishes)	DEM bit in the CHCFG_n register
Descriptor was invalid	LV of header of the descriptor that is read is 0 when DRRP and DIM in the CHCFG_n register are both 0 in link mode	DIM bit in the CHCFG_n register

32B.9.13.5 DMA error interrupt (USBFDMAERRm)

If an error response is received for DMA transfer or descriptor access, this module stops transfer, assuming that an error occurred. When an error response is received, the EN bit in the CHSTAT_n register for channel n that is being used for transfer is cleared to 0, and the ER bit is set to 1 (n = 1, 0). Also, the USBFDMAERRm interrupt is asserted.

The USBFDMAERRm signal cannot be masked.

For a sequence of transfers for which an error occurred, data integrity cannot be guaranteed. Always use the following procedure to restart the transfer sequence from the beginning.

1. Set the SWRST bit in the CHCTRL_n register to 1.
2. Reset each register.

32B.9.13.6 Interval Count Function

The execution interval of a DMA transfer can be adjusted by using the ITVL field in the CHITVL_n register. This function prevents DMAC from continuously occupying the bus. If this function is enabled, DMAC does not perform a DMA transfer for the next request until the counter value becomes 0.

The following shows an operation example.

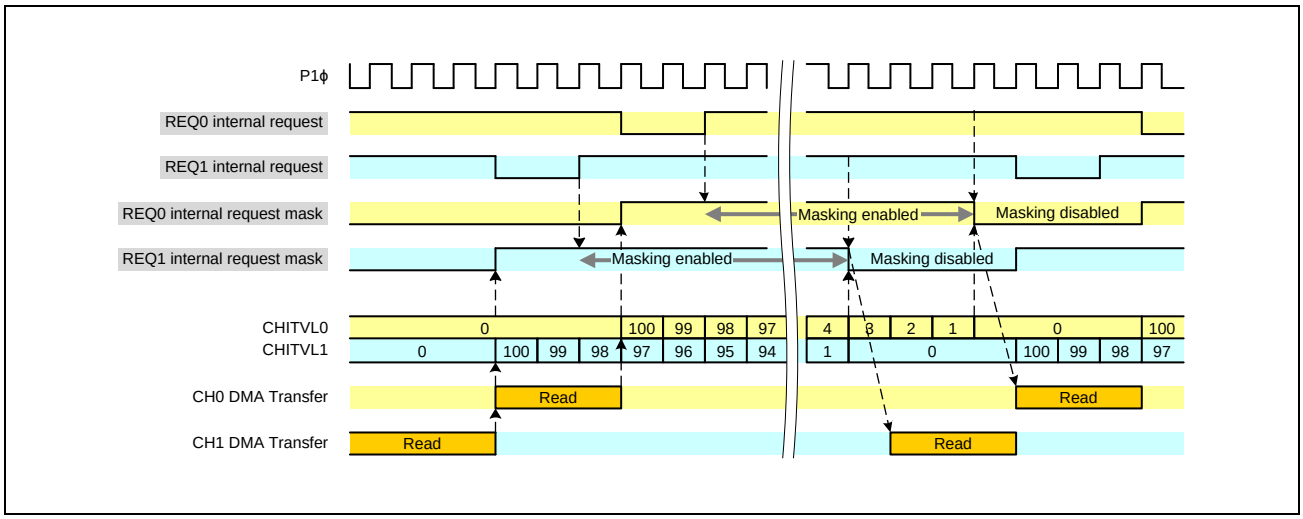


Figure 32B.25 Example of Counting Intervals (REQD = 0, SDS < DDS)

An interval is inserted after a transfer is performed on the side specified by the REQD bit in the CHCFG_n register. The following shows how the REQD, SDS, and DDS values of the CHCFG_n register are related with the interval.

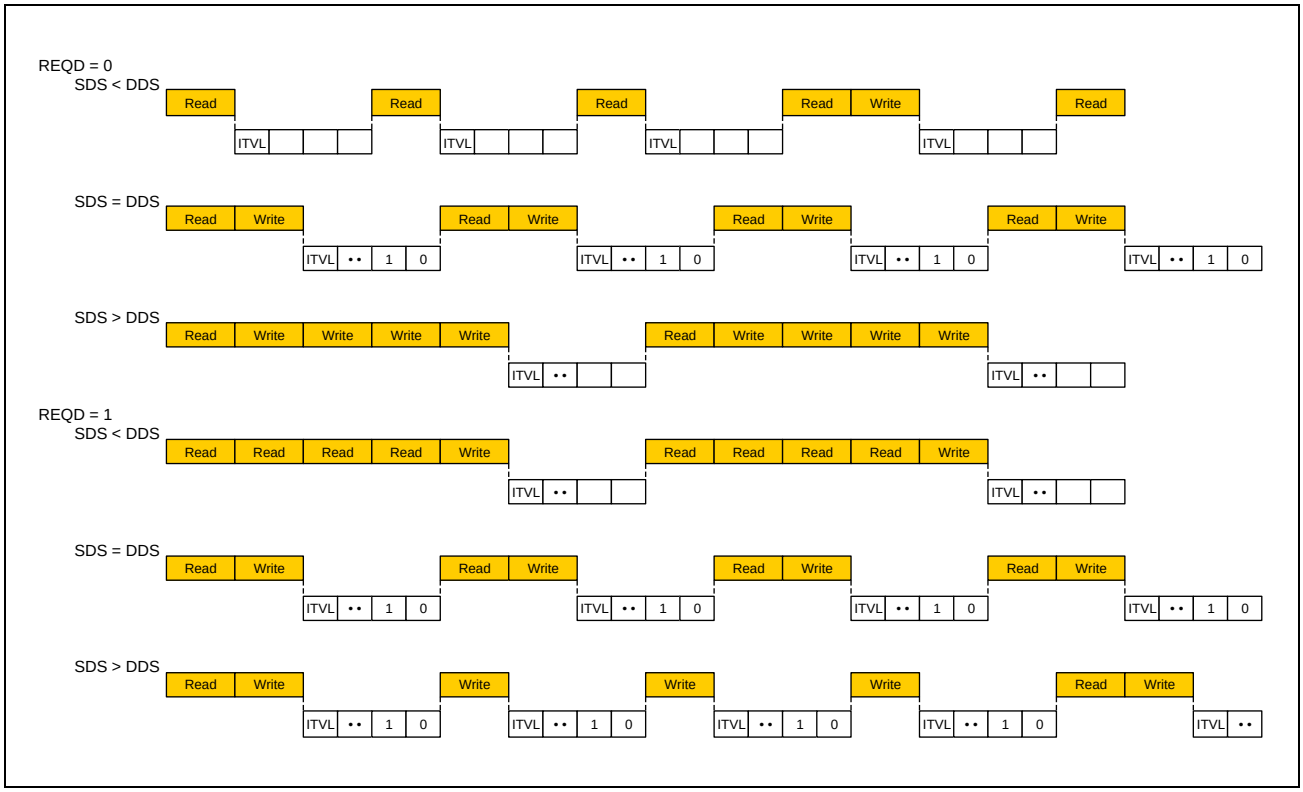


Figure 32B.26 DMA Transfer Settings and Interval Count

32B.9.13.7 Operational difference depending on the transfer size

(1) If the transfer size on the source side is smaller

When reading of as much data as the destination data size finishes, a write to the destination starts.

The following figure is an example of the timing chart in the case where the source is an 8-bit field and the destination is a 32-bit field (SDS = 0 and DDS = 2 in the CHCFG_n register) (when the rising edge is detected).

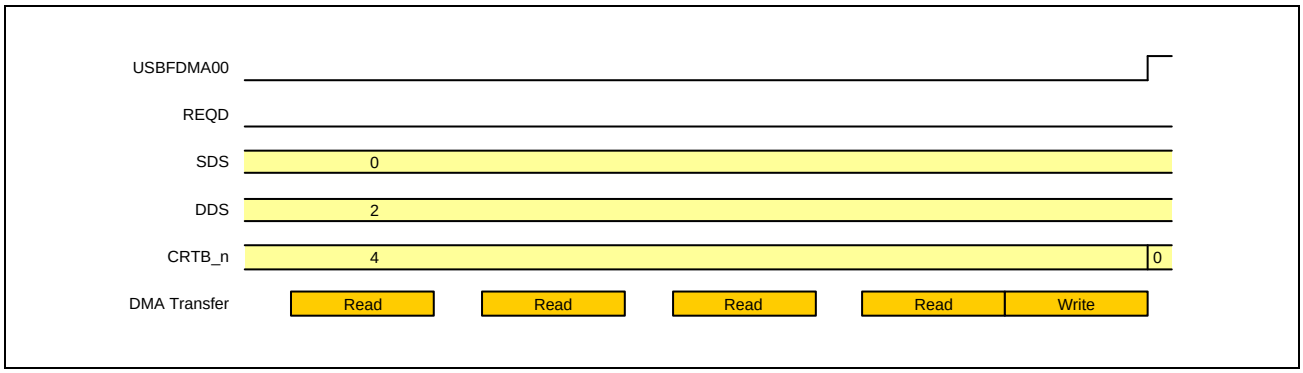


Figure 32B.27 Example of Timing Chart in the Case Where the Source is Smaller
(LVL = 0, HIEN = 1, REQD = 0, and SDS < DDS)

(2) If the transfer size on the destination side is smaller

Because the source side is larger than the destination side, two or more destination write operations occur for one source read operation. The following is an example of the timing chart in the case where the source is a 64-bit field and the destination is a 16-bit field (SDS = 3 and DDS = 1 in the CHCFG_n register) (when the rising edge is detected).

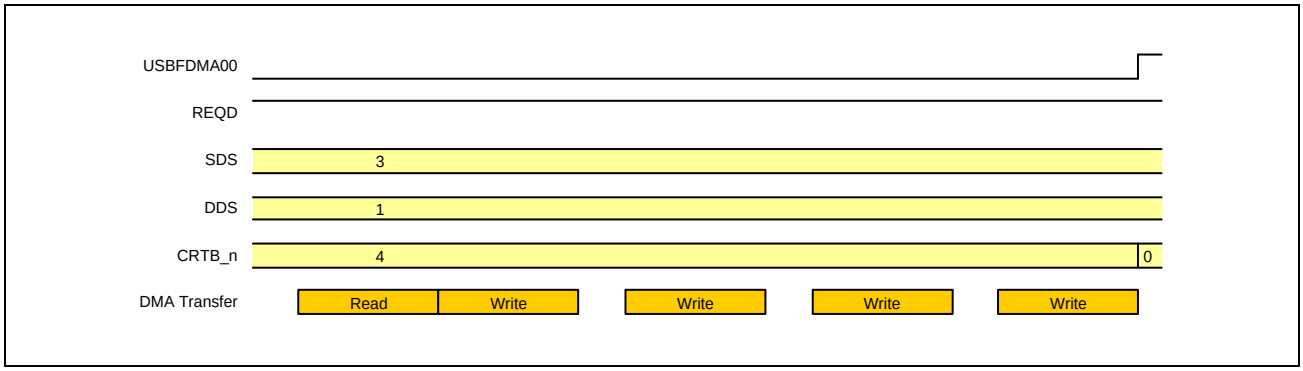


Figure 32B.28 Example of Timing Chart in the Case Where the Destination Is Smaller
(LVL = 0, HIEN = 1, REQD = 1, and SDS > DDS in the CHCFG_n Register)

(3) If the Source and Destination Transfer Sizes Are the Same

Each time a DMA transfer request is detected, a source read operation and a destination write operation occur.

The following is an example of the timing chart in the case where the source and destination are 8-bit fields (SDS = 0 and DDS = 0 in the CHCFG_n register) (when the rising edge is detected).

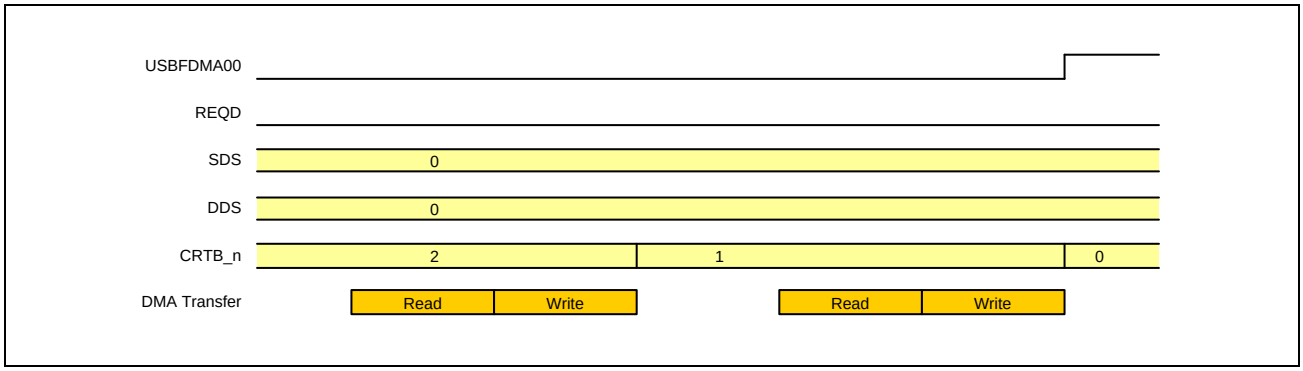


Figure 32B.29 Example of Timing Chart in the Case where the Source and Destination Sizes are the Same (LVL = 0, HIEN = 1, REQD = 0, and SDS = DDS in the CHCFG_n Register)

32B.9.13.8 Transfer state

The CHSTAT_n register indicates the transfer state of a channel.

(1) Transfer state

The TACT bit in the CHSTAT_n register indicates that channel n is operating. When 1 is written to the SETEN bit in the CHCTRL_n register, the TACT bit is set to 1. The TACT bit continues to be 1 while DMAC is accessing a descriptor or waiting for a DMA request.

The TACT bit is cleared when the EN bit in the CHSTAT_n register is cleared (for details about the conditions in which the EN bit is cleared, see **Section 32B.4.1, Current Source Address Register**) and the DMA transfer ends.

If the EN bit is not cleared when the DMA transaction ends (for example, when the REN bit of the CHCFG_n register is 1 in register mode or when DMAC accesses the next descriptor in link mode), the TACT bit is not cleared.

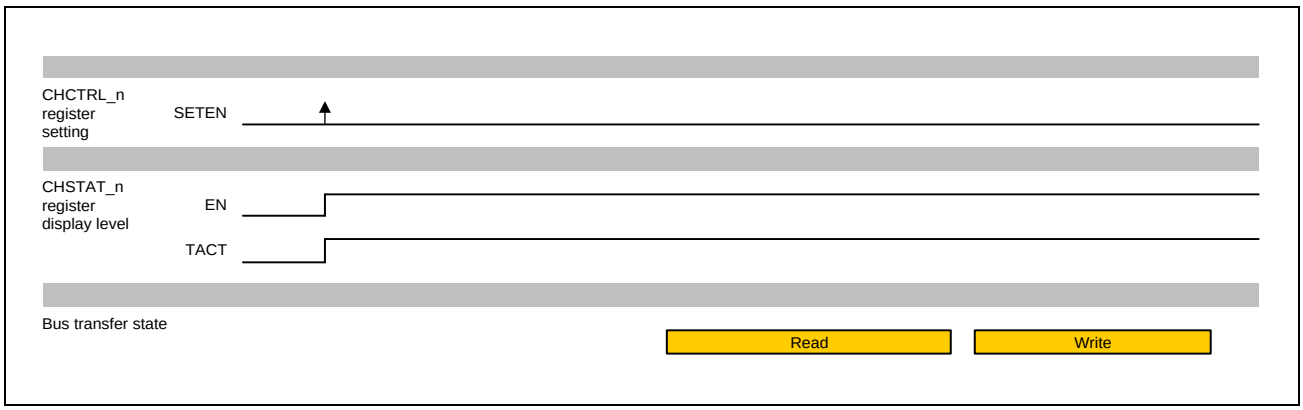


Figure 32B.30 DMAC State Example 1 (Hardware Request)

(2) Suspension

The SETSUS bit in the CHCTRL_n register can be used to suspend a DMA transfer. If suspension of a DMA transfer is attempted when a bus cycle is running, the DMA transfer is suspended after the bus cycle finishes. The suspended transfer can be resumed by writing 1 to the CLRSUS bit in the CHCTRL_n register.

To check whether a DMA transfer is suspended, after setting the SETSUS bit in the CHCTRL_n register, check the SUS bit in the CHSTAT_n register or the SUS bit for the relevant channel in the DSTAT_SUS register. If The SUS bit is 1, the DMA transfer is currently suspended.

(3) Transfer suspension

By writing 1 to the CLREN bit in the CHCTRL_n register during a DMA transaction, the DMA transaction of the channel can be suspended. As the post-processing for suspension, the SBE bit in the CHCFG_n register can be used to select whether to sweep the data remaining in the buffer when the transaction is suspended. The default is SBE = 0 (do not sweep remaining data).

If the sweep mode is enabled and a transfer is suspended by setting 1 for the CLREN bit in the CHCTRL_n register, DMAC sweeps any data remaining in the buffer and stops operation.

(a) Transfer suspension (buffer sweeping disabled: SBE = 0)

If 1 is written to the CLREN bit in the CHCTRL_n register during DMA transfer, DMAC suspends DMA transfer and then stops. The timing of stoppage depends on the value set for the REQD bit. After DMAC stops, write 1 to the SWRST bit in the CHCTRL_n register to clear the internal status of DMAC, and then specify the next transfer settings.

Complete deactivation of the channel can be confirmed when the value of the TACT bit in the CHSTAT_n register changes from 1 to 0.

If DMA transfer is suspended before it is completed, the USBFDMAmn interrupt is not asserted.

If the REQD bit in the CHCFG_n register is 0, DMAC stops when the next read operation is completed. However, if data that can be written exists in the buffer, DMAC writes the data and then stops.

If the REQD bit in the CHCFG_n register is 1, DMAC stops when the next read operation is completed.

(b) Transfer suspension (buffer sweeping enabled: SBE = 1)

If 1 is written to the CLREN bit in the CHCTRL_n register during DMA transfer, DMAC suspends DMA transfer. If the REQD bit in the CHCFG_n register is 0, DMAC sweeps (writes) the already read data, and then stops DMA transfer. If the REQD bit is 1, sweep mode cannot be used physically.

After DMAC stops, set the SWRST bit in the CHCTRL_n register to clear the internal status of DMAC, and then specify the next transfer settings.

Complete deactivation of the channel can be confirmed when the value of the TACT bit in the CHSTAT_n register changes from 1 to 0.

(c) How to confirm deactivation of the channel

Even when the EN bit of the CHSTAT_n register is cleared to 0 by writing 1 to the CLREN bit in the CHCTRL_n register, if a transfer has already been executed over the bus, DMAC cannot immediately stop. To check whether DMAC has stopped completely, check the EN and TACT bits in the CHSTAT_n register. If both bits are 0, DMAC has stopped completely.

(d) Procedure for suspending transfer

To suspend transfer:

1. Write 1 to the CLREN bit in the CHCTRL_n register.
2. If the SBE bit in the CHCFG_n register is 0, DMAC stops according to the value of the REQD bit in the CHCFG_n register. If the SBE bit is 1, DMAC is placed in sweep mode.
3. Read the CHSTAT_n register to check whether the TACT bit is 0. If the TACT bit is 0, DMAC has stopped completely. If the TACT bit is 1, continue polling until the bit changes to 0.
4. To perform the next DMA transfer after it is suspended, make sure that the SWRST (software reset) bit in the CHCTRL_n register is turned on before the next DMA transfer starts.

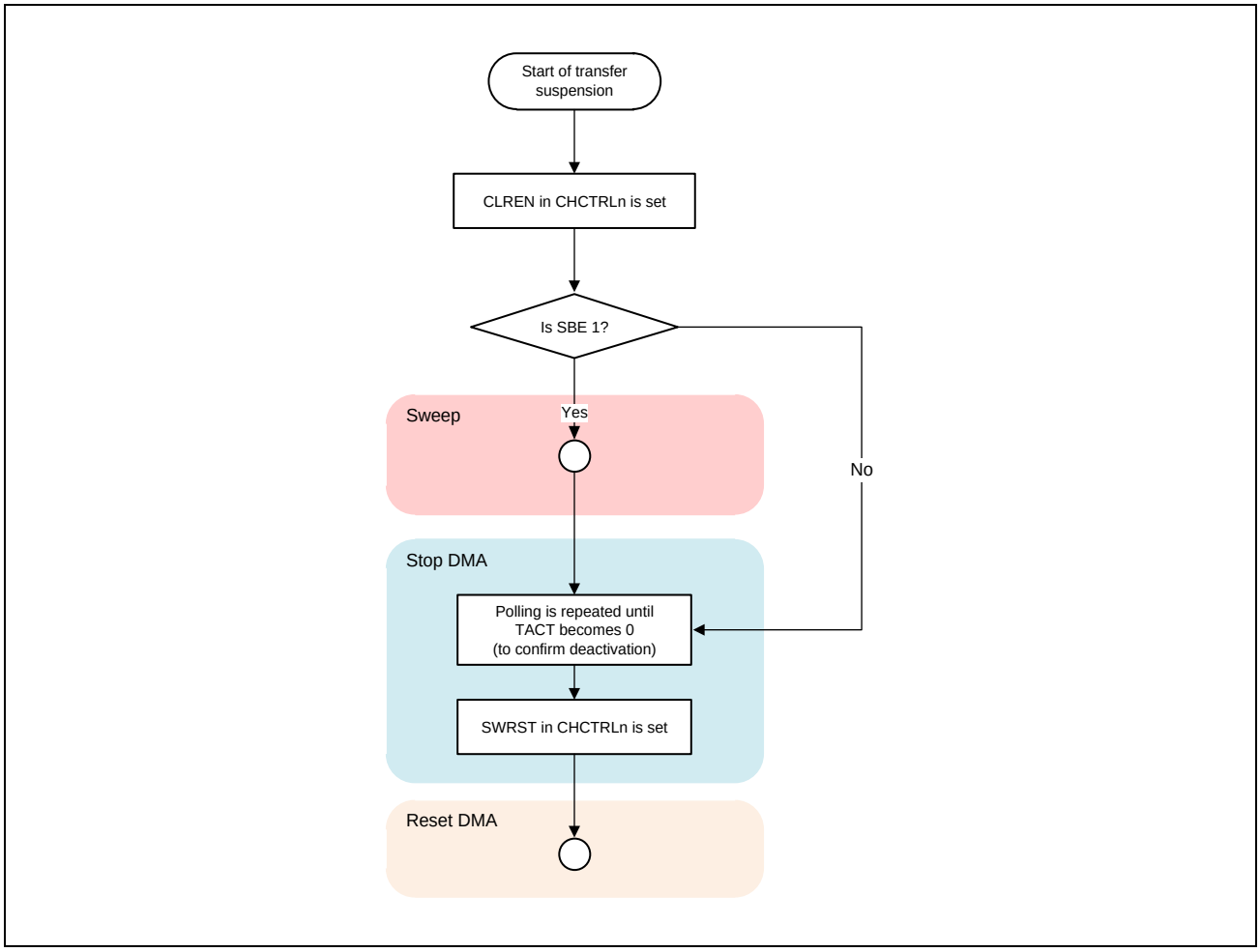


Figure 32B.31 Transfer Suspension Flowchart

32B.9.14 Access Type

32B.9.14.1 DMA master transfer combination list

(1) Read access

The following describes characteristics of the issuance type for DMA read access.

- An access is performed for a beat align space whose size is set by SDS[2:0] in the CHCFG_n register, including the source address indicated in the CRSA_n register. A beat unaligned transfer for the bus is not performed. An excess area is sometimes read depending on CRSA_n or SKIP settings. In this case, the necessary data is imported into the buffer from the read data.
- The size and burst length are determined based on the value set in the SDS[2:0] field.
  If the value set in the SDS[2:0] field is equal to or less than the bus width
  Size: Value set in SDS[2:0]
  Burst type: SINGLE
  If the value set in the SDS[2:0] field is larger than the bus width
  Size: Bus width
  Burst type: Fixed-length burst (burst length = SDS[2:0] value/bus width)

The following indicates the access types for the bus.

Table 32B.53 DMA Read Transfer Combination List

SDS	Source Address	AHB transfer					
		First transfer			Second transfer		
		Address	Data Size	Burst Type	Address	Data Size	Burst Type
0 (8 bits)	-	addr	8	SINGLE			
1 (16 bits)	2 byte align	{addr[31:1], 0b}	16	SINGLE			
	2 byte unalign				{addr[31:1], 0b} + H'2	16	SINGLE
2 (32 bits)	4 byte align	{addr[31:2], 00b}	32	SINGLE			
	4 byte unalign				{addr[31:2], 00b} + H'4	32	SINGLE
4 (128 bits)	16 byte align	{addr[31:4], H'0}	32	INCR4			
	16 byte unalign				{addr[31:4], H'0} + H'10	32	INCR4
5 (256 bits)	32 byte align	{addr[31:5], H'00}	32	INCR8			
	32 byte unalign				{addr[31:5], H'00} + H'20	32	INCR8
6 (512 bits)	64 byte align	{addr[31:6], H'00}	32	INCR16			
	64 byte unalign				{addr[31:6], H'00} + H'40	32	INCR16

Note: If EBT is detected in the middle of burst, 32-bit INCR burst is used to transfer the remaining data.

(2) Write access

The following describes characteristics of the issuance type for DMA write access.

- An access is performed from the destination address indicated by the CRDA_n register to the beat align boundary whose size is set by DDS[2:0] in the CHCFG_n register.
- The size and burst length are determined based on the value set in the DDS[2:0] field.
  If the value set in the DDS[2:0] field is equal to or less than the bus width
  Size: Value set in DDS[2:0]
  Burst type: SINGLE
  If the value set in the DDS[2:0] field is greater than the bus width
  Size: Bus width
  Burst type: Fixed-length burst (burst length = DDS[2:0] value/bus width)
- In write access, only the specified space is accessed. In the following cases, the combination of values smaller than the value set in the DDS[2:0] field is used for access.
  - The destination address is beat-unaligned for the value set in the DDS[2:0] field.
  - An access specified in the DDS field will be across the SKIP boundary.
  - The size specified in the DDS[2:0] field is too large for the number of remaining bytes to be transferred.

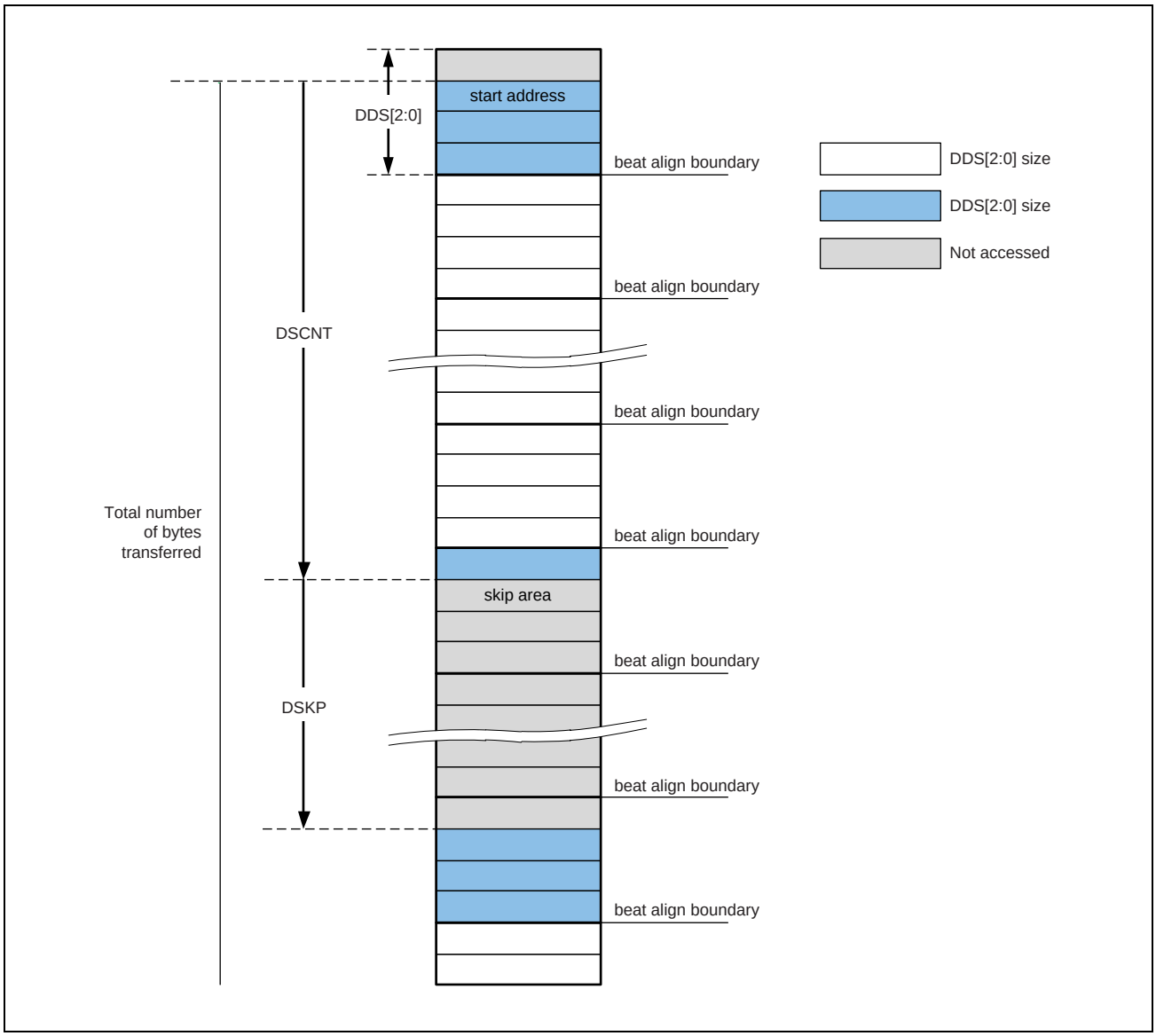


Figure 32B.32 Example of DMA Write Access Space and Access Types

The following shows the access types for the bus in the case of beat-aligned.

Table 32B.54 DMA Write Transfer Combination List

	AHB transfer		
	First transfer		
DDS[2:0]	Address	Data Size	Burst Type
0 (8 bits)	addr	8	SINGLE
1 (16 bits)	{addr[31:1], 0b}	16	SINGLE
2 (32 bits)	{addr[31:2], 00b}	32	SINGLE
4 (128 bits)	{addr[31:4], H'0}	32	INCR4
5 (256 bits)	{addr[31:5], H'00}	32	INCR8
6 (512 bits)	{addr[31:6], H'00}	32	INCR16

32B.9.14.2 DMA master descriptor combination list

(1) Read access

The following describes characteristics of a descriptor access.

- Accesses are performed for 8-word beat align space including the LINK address (the address indicated in the CRLA_n register). A beat unaligned transfer is not performed.
- The size and burst length set by INCR8 are used.
- The descriptor format (DSCFM) of the read header is analyzed, and then the descriptor data is set in an internal register.
- If the descriptor extends over the 8-word boundary, an additional read is performed on the succeeding eight words.

The following indicates the types of access to the bus.

Table 32B.55 Descriptor Read Transfer Combination List

Descriptor Format	Address	AHB Transfer					
		First Transfer			Second Transfer		
	addr[4:0]	Address	Size	Burst	Address	Size	Burst
4 words	H'00	{addr[31:4], 0000b}	32	INCR8			
	H'04						
	H'08						
	H'0C						
	H'10						
	H'14				{addr[31:4], 0000b} + H'20	32	INCR8
	H'18						
	H'1C						
8 words	H'00						
	H'04				{addr[31:4], 0000b} + H'20	32	INCR8
	H'08						
	H'0C						
	H'10						
	H'14						
	H'18						
	H'1C						

Note: If EBT is detected in the middle of burst, 32-bit INCR burst is used to transfer the remaining data.

(2) Write access

Use single transfer as the issuance type for writing data back to a descriptor. The following indicates the types of access to the bus.

Table 32B.56 Descriptor Write Transfer Combination List

Type	AHB transfer		
	Address	Size	Burst
Write-back in normal mode	{addr[31:2], 00b} + H'3	8	SINGLE
Write-back in the case of an error	{addr[31:2], 00b} + H'2	8	SINGLE

32B.9.15 Arbitration between DMACs

Arbitration between internal DMACs is performed in round-robin mode.

In round-robin mode, the highest priority is given to the DMAC whose DMAC number is the DMC number being used for transfer + 1. Immediately after a reset, DMAC0 has the highest priority.

Table 32B.57 Priority of a transfer request for DMACs that are performing transfer

	Next DMAC	
Current DMAC	DMAC0	DMAC1
DMAC0	2	1
DMAC1	1	2

Note: Priority: 1 (high), 2 (low)

32B.9.16 Notes

32B.9.16.1 Access

During read access, a beat align area is accessed by one transfer. Therefore, if beat unaligned is set, the beat align area including the specified area is accessed.

For example, if the Source Address is H’0000_1038 and SDS is 5 (256-bit), read starts from area H’0000_1020, not from address H’0000_1038. At this time, if the area from H’0000_1020 to H’0000_1037 contains a register whose value changes by read, the operation might be disrupted.

To prevent problems, use the beat align setting to access a register whose value changes by read access or access to an area adjacent to FIFO.

32B.9.16.2 Level Interrupt bit

This is a DMA interrupt output enable bit. Always set this bit to 1 irrespective of whether peripheral module interrupt USBFDMAmn (m, n = 0, 1) or USBFDMAERRm (m = 0, 1) is used.

To use peripheral module interrupt USBFDMAmn or USBFDMAERRm, set this bit to 1, and then set up the interrupt controller.

33. SD/MMC Host Interface

NOTE

Development of the SD host--related products needs the conclusion of the following agreement. SD Host/Ancillary Product License Agreement (SD HALA)

33.1 Overview

33.1.1 Features

- 3 channels
- Channel 0 supports SDHI / e-MMC
- Channel 1 and channel 2 support SDHI
- SD memory/IO card interface (1-bit/4-bit SD bus)
- SD, SDHC, and SDXC SD memory card access supported
- Default, high-speed, UHS-I/SDR50, and SDR104 transfer modes supported
- SD clock (SD_CLK) frequency = SDxϕ/4 frequency/2^n (n = 0 to 9) (x = 0, 1, 2)
- Error check function: CRC7 (for command/response), CRC16 (for data)
- Interrupt request: 2
- Card detect function
- Write protect supported
- MMC interface (1-/4-/8-bit MMC bus)
- e-MMC device access supported
- Backward-compatible, high-speed, HS200 transfer modes supported
- High-priority interrupt (HPI) supported

Block Diagram

Figure 33.1 shows a block diagram of the SD/MMC host interfaces.

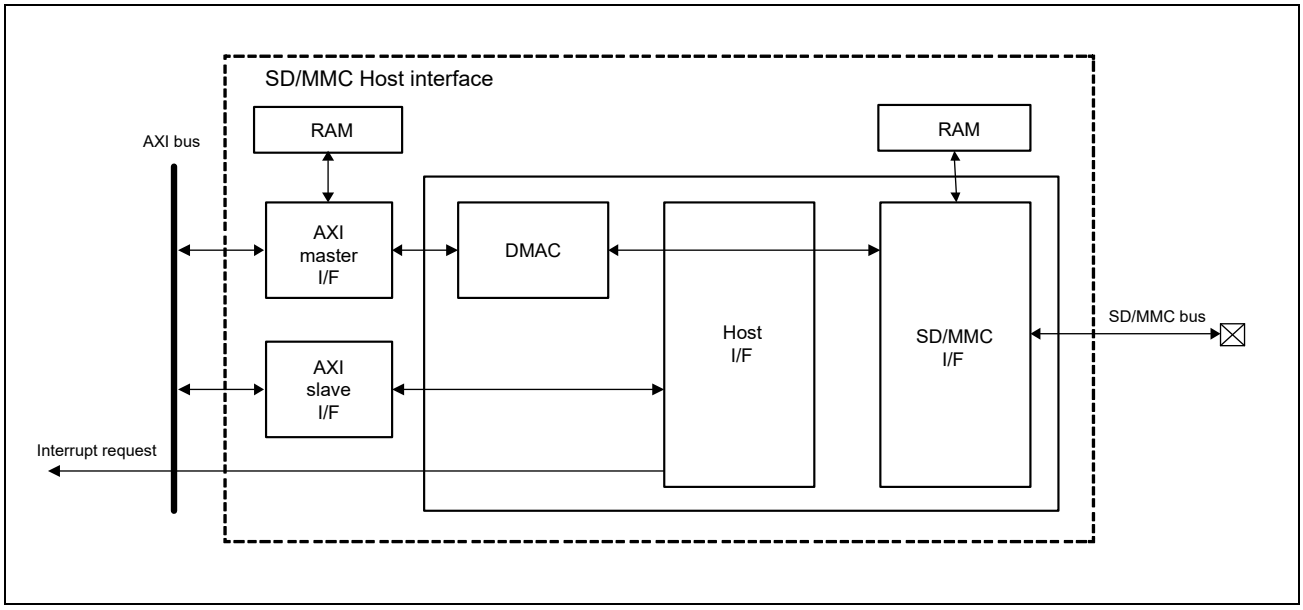


Figure 33.1 Block Diagram of SD/MMC Host Interface

33.1.2 External Pins

Table 33.1 lists the input and output pins used by the interface. The operating voltage on the pins of the SD/MMC host interface is 3.3 V or 1.8 V for ch0 and ch1, and only 3.3 V for ch2. Channels 0, 1 and 2 can operate with different voltages. Before using the pins of the SD/MMC host interface, be sure to set the SD ch0/1 IO Voltage Mode Control Register (SD_ch0 and SD_ch1) and Driving Ability Control Register (IOLH).

For details, see the following sections:

Section 45.3.7, Driving Ability Control Register,
Section 45.3.16, SD Ch0 IO Voltage Mode Control Register,
Section 45.3.17, SD Ch1 IO Voltage Mode Control Register.

Table 33.1 Pin Configuration

PIN Name	I/O	Function
SDx_CLK*1	O	SD/MMC clock output
SDx_CMD*1	I/O	SD/MMC command output, response input
SDx_DATA0*1	I/O	SD/MMC Data 0 [bit 0]
SDx_DATA1*1	I/O	SD/MMC Data 1 [bit 1], SDIO interrupt
SDx_DATA2*1	I/O	SD/MMC Data 2 [bit 2], read wait
SDx_DATA3*1	I/O	SD/MMC Data 3 [bit 3], Card detection
SD0_DATA4*1	I/O	SD/MMC Data 4 [bit 4]
SD0_DATA5*1	I/O	SD/MMC Data 5 [bit 5]
SD0_DATA6*1	I/O	SD/MMC Data 5 [bit 6]
SD0_DATA7*1	I/O	SD/MMC Data 7 [bit 7]
SDx_CD*1	I	SD/MMC card detection*2
SDx_WP*1	I	SD/MMC write protection*2
SD0_RST#	O	SD/MMC reset

Note 1. x (= 0, 1, 2) is the channel number of the SD/MMC host interface. In this manual, those pins are referred as SDDAT0, SDDAT1, …, SDDAT7.

Note 2. Fix to 1 when not in use.

33.1.3 Register Configuration

The base addresses for each channel are as follows. The SD interface and MMC interface are switched by setting the command type register (SD_CMD).

Channel 0:

H'0_11C0_0000 (Cortex-A55 Address Space)
H'41C0_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
H'51C0_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Channel 1:

H'0_11C1_0000 (Cortex-A55 Address Space)
H'41C1_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
H'51C1_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Channel 2:

H'0_11C2_0000 (Cortex-A55 Address Space)
H'41C2_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
H'51C2_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 33.2 Register Configurations (1/2)

Name	Abbreviation	Address	Access Width	Mirror
Command type register	SD_CMD	H'0000	16/32/64	
Command argument registers	SD_ARG	H'0010	16/32/64	
	SD_ARG1	H'0018	16/32/64	SD_ARG[31:16]
Data stop register	SD_STOP	H'0020	16/32/64	
Block count register	SD_SECCNT	H'0028	16/32/64	
Card response registers	SD_RSP10	H'0030	16/32/64	
	SD_RSP1	H'0038	16/32/64	SD_RSP10[31:16]
	SD_RSP32	H'0040	16/32/64	SD_RSP10[63:32]
	SD_RSP3	H'0048	16/32/64	SD_RSP32[31:16]
	SD_RSP54	H'0050	16/32/64	
	SD_RSP5	H'0058	16/32/64	SD_RSP54[31:16]
	SD_RSP76	H'0060	16/32/64	SD_RSP54[63:32]
	SD_RSP7	H'0068	16/32/64	SD_RSP76[31:16]
SD card interrupt flag register 1	SD_INFO1	H'0070	16/32/64	
SD card interrupt flag register 2	SD_INFO2	H'0078	16/32/64	
SD_INFO1 interrupt mask register	SD_INFO1_MASK	H'0080	16/32/64	
SD_INFO2 interrupt mask register	SD_INFO2_MASK	H'0088	16/32/64	
SD clock control register	SD_CLK_CTRL	H'0090	16/32/64	
Transfer data length register	SD_SIZE	H'0098	16/32/64	
Card access control option register	SD_OPTION	H'00A0	16/32/64	
SD error status register 1	SD_ERR_STS1	H'00B0	16/32/64	

Table 33.2 Register Configurations (2/2)

Name	Abbreviation	Address	Access Width	Mirror
SD error status register 2	SD_ERR_STS2	H'00B8	16/32/64	
SD buffer read/write register	SD_BUF0	H'00C0	16/32/64	
SDIO mode control register	SDIO_MODE	H'00D0	16/32/64	
SDIO interrupt flag register	SDIO_INFO1	H'00D8	16/32/64	
SDIO_INFO1 interrupt mask register	SDIO_INFO1_MASK	H'00E0	16/32/64	
DMA mode enable register	CC_EXT_MODE	H'0360	16/32/64	
Software reset register	SOFT_RST	H'0380	16/32/64	
Version register	VERSION	H'0388	16/32/64	
Host interface mode setting register	HOST_MODE	H'0390	16/32/64	
SD interface mode setting register	SDIF_MODE	H'0398	16/32/64	
SD status register*1	SD_STATUS	H'03C8	16/32/64	
DMAC mode register	DM_CM_DTRAN_MODE	H'0820	16/32/64	
DMAC control register	DM_CM_DTRAN_CTRL	H'0828	16/32/64	
DMAC reset register	DM_CM_RST	H'0830	16/32/64	
DMAC interrupt register 1	DM_CM_INFO1	H'0840	16/32/64	
DM_CM_INFO1 interrupt mask register	DM_CM_INFO1_MASK	H'0848	16/32/64	
DMAC interrupt register 2	DM_CM_INFO2	H'0850	16/32/64	
DM_CM_INFO2 interrupt mask register	DM_CM_INFO2_MASK	H'0858	16/32/64	
DMAC address register	DM_DTRAN_ADDR	H'0880	16/32/64	
SCC register area*2	—	—	—	—

Note 1. Only for channel 0.

Note 2. Refer to **Section 33.7, SCC Register Descriptions**.

33.2 Register Description

33.2.1 Command Type Register (SD_CMD)

The command type register (SD_CMD) is used to select the command type and response type. The command sequence is started by writing to SD_CMD.

For details on the SD_CMD setting, refer to **Section 33.4.14, Example of SD_CMD Register Setting**.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	MD7	00	R/W	Multiple Block Transfer Mode (enabled at multiple block transfer)
14	MD6			00: CMD12 is automatically issued at multiple block transfer. 01: CMD12 is not automatically issued at multiple block transfer. 10: Setting prohibited 11: Setting prohibited
13	MD5	0	R/W	Single/Multiple Block Transfer (enabled when the command with data is handled) 0: Single block transfer 1: Multi block transfer
12	MD4	0	R/W	Write/Read Mode (enabled when the command with data is handled) 0: Write (SD/MMC host interfaces -> SD card) 1: Read (SD/MMC host interfaces <- SD card)
11	MD3	0	R/W	Data Mode (Command Type) 0: Command without data transfer (bc, bcr, ac) 1: Command with data transfer (adtc)
10	MD2	000	R/W	Mode/Response Type
9	MD1			000: Normal mode
8	MD0			The response type and the transfer mode are selected by SD_CMD[7:0], and the SD_CMD[15:11] setting is disabled. 001: Setting prohibited 010: Setting prohibited 011: Extended mode/No response 100: Extended mode/R1, R5, R6, or R7 response from the SD card 101: Extended mode/R1b response from the SD card 110: Extended mode/R2 response from the SD card 111: Extended mode/R3 or R4 response from the SD card Some commands cannot be used in normal mode. For details, see **Section 33.4.14, Example of SD_CMD Register Setting** to select mode/response type.
7	C1	00	R/W	00: CMD
6	C0			01: ACMD 10: Setting prohibited 11: Setting prohibited
5	CF45	000000	R/W	Command Index
4	CF44			These bits specify Command Format[45:40] (command index).
3	CF43			[Examples]
2	CF42			CMD6: SD_CMD[7:0] = 0000_0110b CMD18: SD_CMD[7:0] = 0001_0010b
1	CF41			ACMD13: SD_CMD[7:0] = 0100_1101b
0	CF40			

Note: SD_CMD cannot be written to when the CBSY bit in SD_INFO2 is 1.

33.2.2 Command Argument Register (SD_ARG)

Command arguments for SD cards are set in the SD command argument registers (SD_ARG). Set the command arguments before writing to SD_CMD.

Note that the argument of CMD12 within command sequences is H'0000_0000 regardless of the setting of SD_ARG.

SD_ARG

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 0	CF39 to CF8	All 0	R/W	Set command format[39:8] (argument).

SD_ARG1 (Mirror of SD_ARG[31:16])

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	CF39 to CF24	All 0	R/W	Set command format[39:24] (argument).

33.2.3 Data Stop Register (SD_STOP)

The data stop register (SD_STOP) is used to enable or disable block counting at multiple block transfer, and to control the issuing of CMD12 within command sequences.

Bit	Bit Name	Initial Value	R/W	Description
63 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
18	—	0*1	R/W	Reserved The write value should always be 0.
17	HPIMODE	0*1	R/W	HPI Mode Enable 0: Disables HPI mode. 1: Enables HPI mode.
16	HPICMD	0*1	R/W*3	HPI Command Issue When HPICMD is set to 1 while HPIMODE is 1, the HPI command (CMD12) is issued. This bit is cleared to 0 when reception of the response to CMD12 is completed. The timing with which this bit is set to 1 is as follows. • After reception of the response to CMD12 that was issued by setting the STP bit to 1 has been completed during the CMD6/CMD38 or CMD25 sequence. • After reception of the response to CMD24/CMD25 has been completed After HPICMD is set to 1, do not write 0 to this bit while the CBSY bit in SD_INFO2 is 1. Do not set this bit to 1 when the CBSY bit in SD_INFO2 is 0.
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8	SEC	0*1	R/W	Block Count Enable*2 0: Disables SD_SECCNT setting value. 1: Enables SD_SECCNT setting value. Set SEC to 1 at multiple block transfer. When SD_CMD is set as follows to start the command sequence while SEC is set to 1, CMD12 is automatically issued to stop multi-block transfer with the number of blocks which is set to SD_SECCNT. 1. CMD18 or CMD25 in normal mode (SD_CMD[10:8] = 000) 2. SD_CMD[15:13] = 001 in extended mode (CMD12 is automatically issued, multiple block transfer) When the command sequence is halted because of a communications error or timeout, CMD12 is not automatically issued.
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.

Bit	Bit Name	Initial Value	R/W	Description
0	STP	0*1	R/W	Stop • When STP is set to 1 during multiple block transfer, CMD12 is issued to halt the transfer through the SD/MMC host interface. However, if a command sequence is halted because of a communications error or timeout, CMD12 is not issued. Although continued buffer access is possible even after STP has been set to 1, the buffer access error bit (ERR5 or ERR4) in SD_INFO2 will be set accordingly. • When STP has been set to 1 during transfer for single block write, the access end flag is set when SD_BUF becomes empty, and CMD12 is not issued. If SD_BUF does contain data, the access end flag is set on completion of reception of the busy state without CMD12 having been issued. • When STP has been set to 1 during transfer for single block read, the access end flag is set immediately after setting of the STP bit and CMD12 is not issued. • When STP is set to 1 during reception of the busy state after an R1b response, the access end flag is set on completion of reception of the busy state without CMD12 having been issued. • When STP is set to 1 after a command sequence has been completed, CMD12 is not issued and the access end flag is not set. • Set STP to 1 after the response end flag has been set. • Set STP to 0 after the response end flag has been set.

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 2. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1.

Note 3. It is effective only if 1 is written.

33.2.4 Block Count Register (SD_SECCNT)

The block count register (SD_SECCNT) is used to specify the number of transfer blocks at multiple block transfer.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 0	CNT31 to CNT0	All 0	R/W	Number of Transfer Blocks*1 When H'0000_0001 is set, the number of transfer blocks is 1. ⋮ When H'0000_FFFF is set, the number of transfer blocks is 65535. ⋮ When H'FFFF_FFFF is set, the number of transfer blocks is 4294967295. Do not set this register to H'0000_0000 if multiple blocks are to be transferred.

Note 1. Do not change the value of these bits when the CBSY bit in SD_INFO2 is set to 1.

33.2.5 SD Card Response Registers (SD_RSP)

The SD card response registers (SD_RSP) hold the response from the SD card.

SD_RSP10

Bit	Bit Name	Initial Value	R/W	Description
63 to 0	R71 to R8	All 0	R	Hold the response from the SD card

SD_RSP1 (Mirror of SD_RSP10[31:16])

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	R39 to R24	All 0	R	Hold the response from the SD card

SD_RSP32 (Mirror of SD_RSP10[63:32])

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 0	R71 to R40	All 0	R	Hold the response from the SD card

SD_RSP3 (Mirror of SD_RSP32[31:16])

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	R71 to R56	All 0	R	Hold the response from the SD card

SD_RSP54

Bit	Bit Name	Initial Value	R/W	Description
63 to 56	—	All 0	R	These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
55 to 0	R127 to R72	All 0	R	Hold the response from the SD card

SD_RSP5 (Mirror of SD_RSP54[31:16])

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	R103 to R88	All 0	R	Hold the response from the SD card

SD_RSP76 (Mirror of SD_RSP54[63:32])

Bit	Bit Name	Initial Value	R/W	Description
63 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
23 to 0	R127 to R104	All 0	R	Hold the response from the SD card

SD_RSP7 (Mirror of SD_RSP76[31:16])

Bit	Bit Name	Initial Value	R/W	Description
63 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
7 to 0	R127 to R120	All 0	R	Hold the response from the SD card

Table 33.3 lists the response types and corresponding SD_RSP registers.

Table 33.3 Response Types and Corresponding SD_RSP Registers

Response Types	SD_RSP Registers
R1, R1b[39:8]	SD_RSP10 SD_RSP54*1
R2[127:8]	SD_RSP54 and SD_RSP10
R3[39:8]	SD_RSP10
R4[39:8]	SD_RSP10
R5[39:8]	SD_RSP10
R6[39:8]	SD_RSP10
R7[39:8]	SD_RSP10

Note 1. The response to CMD18 and to CMD25 is stored in both R[39:8] and R[103:72].
This makes it possible to confirm the response to CMD18 and CMD25 by reading R[103:72] even if the response to automatic CMD12 is stored in R[39:8].

33.2.6 SD Card Interrupt Flag Register (SD_INFO1)

The SD card interrupt flag register 1 (SD_INFO1) indicates the response end and access end in the command sequence. This register also indicates the card detect/write protect state.

For CMD12 and CMD52 (SDIO abort) at multiple block transfer, INFO0 is not set but only INFO2 is set.

Even if the command sequence is halted because of a communications error or timeout, INFO0 or INFO2 is set.

INFO10, INFO9, and INFO8 change depending on the SDDAT3 state after a reset is released and continue to change in 4-bit transfer mode.

To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	HPIRES	0*2	R/W*1	Response Reception Completion [Setting condition] When reception of the response to CMD12 that was issued by setting the STP bit to 1 is completed during the CMD6/CMD38 or CMD25 sequence in HPI mode. [Clearing condition] When 0 is written to HPIRES
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
10	INFO10	Unknown	R	Indicates the SDDAT3 state. 0: SDDAT3 is set to 0. 1: SDDAT3 is set to 1.
9	INFO9	0	R/W*1	SDDAT3 Card Insertion [Setting condition] After change in SDDAT3 from 0 to 1, two cycles of 2 P1ϕ have elapsed with SDDAT3 held 1. [Clearing condition] When 0 is written to INFO9
8	INFO8	0	R/W*1	SDDAT3 Card Removal [Setting condition] After change in SDDAT3 from 1 to 0, two cycles of 2 P1ϕ have elapsed with SDDAT3 held 0. [Clearing condition] When 0 is written to INFO8
7	INFO7	Unknown	R	Write Protect Indicates the ISDWP state. 0: ISDWP is set to 1. 1: ISDWP is set to 0.
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
5	INFO5	Unknown	R	Indicates the ISDCD state. 0: Indicates that Mcycle has elapsed with ISDCD held 1. 1: Indicates that Mcycle has elapsed with ISDCD held 0. Mcycle is set by bits 3 to 0 in SD_OPTION.

Bit	Bit Name	Initial Value	R/W	Description
4	INFO4	0	R/W*1	ISDCD Card Insertion [Setting condition] After change in ISDCD from 1 to 0, Mcycle has elapsed with ISDCD held 0. [Clearing condition] When 0 is written to INFO4 Mcycle is set by bits 3 to 0 in SD_OPTION.
3	INFO3	0	R/W*1	ISDCD Card Removal [Setting condition] After change in ISDCD from 0 to 1, Mcycle has elapsed with ISDCD held 1. [Clearing condition] When 0 is written to INFO3 Mcycle is set by bits 3 to 0 in SD_OPTION.
2	INFO2	0*2	R/W*1	Access End [Setting conditions] 1. When read access to the buffer is completed in the case of transfer for single block read 2. When read access to the buffer for the last block of data is completed in the case of transfer for multiple block read 3. When read access to the buffer and reception of the response to CMD12 are completed in the case of transfer for multiple block read with automatic issuing of CMD12 4. When reception of the busy state after reception of the CRC status is completed in the case of transfer for single block write 5. When reception of the busy state after reception of the CRC status of the last block of data is completed in the case of transfer for multiple block write 6. When reception of the response busy state for CMD12 is completed in the case of transfer for multiple block write with automatic issuing of CMD12 7. When reception of the response to CMD12 that was issued by setting the STP bit to1 is completed in the case of transfer for multiple block read 8. When reception of the response busy state for CMD12 that was issued by setting the STP bit to1 is completed in the case of transfer for multiple block write 9. When reception of the response to CMD52 that was issued by setting the IOABT bit to1 is completed in the case of transfer for multiple block read 10. When reception of the response to CMD52 that was issued by setting the IOABT bit to1 is completed in the case of transfer for multiple block write In addition to the above conditions, this bit is set when a command sequence is halted because of a communications error or timeout. [Clearing condition] When 0 is written to INFO2 When the access end bit is set to 1, the command sequence is terminated.
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.

Bit	Bit Name	Initial Value	R/W	Description
0	INFO0	0*2	R/W*1	Response End [Setting conditions] 1. When the reception of the response is completed 2. When the transmission of the command not requiring a response is completed 3. When receiving busy reception after R1b response 4. When reception of the response to CMD52 that was issued by setting the C52PUB bit to1 is completed in the case of transfer for multiple block read 5. When reception of the response to CMD52 that was issued by setting the C52PUB bit to1 is completed in the case of transfer for multiple block write In addition to the above conditions, this bit is set when a command sequence is halted because of a communications error or timeout. [Clearing condition] When 0 is written to INFO0 When issuing a command without data, the command sequence ends when the response end is set to 1.

Note 1. It is effective only if 0 is written.

Note 2. The value is initialized by a reset and also in the case of a reset by the SDRST bit in SOFT_RST.

33.2.7 SD Card Interrupt Flag Register (SD_INFO2)

The SD card interrupt flag register 2 (SD_INFO2) indicates the access status of the SD buffer (SD_BUF) and SD card. To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	ILA	0*2	R/W*1	Illegal Access Error [Setting conditions] 1. Writing of data to SD_CMD within a command sequence (CBSY=1) 2. When SD_CMD[11] = 1 (command with data transfer) and SD_CMD[7:0] = 0000_1100b (CMD12) are set in SD_CMD [Clearing condition] When 0 is written to ILA
14	CBSY	0*2	R	Command Type Register Busy 0: A command sequence has been completed. 1: A command sequence is being executed.
13	SCLKDIVEN	1*2	R	0: The SD bus (CMD, DAT) is busy. Do not attempt to write to the SD_CLK_CTRL register. 1: The SD bus (CMD, DAT) is not busy. When a command sequence is started by writing to SD_CMD, the CBSY bit is set to 1 and, at the same time, the SCLKDIVEN bit is set to 0. The SCLKDIVEN bit is set to 1 after 8 cycles of SDCLK have elapsed after setting of the CBSY bit to 0 due to completion of the command sequence.
12	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
11	—	0	R/W	Reserved The write value should always be 1.
10	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
9	BWE	0*2	R/W*1	SD_BUF Write Enable 0: Data cannot be written in SD_BUF0. 1: Data can be written in SD_BUF0. [Setting conditions] 1. When SD_BUF is empty at single block transfer 2. When either bank 1 or bank 2 of SD_BUF is empty at multiple block transfer [Clearing conditions] 1. When 0 is written to BWE 2. Writing of a block of data to SD_BUF by DMA transfer When data is written to SD_BUF0 by the CPU, clear BWE and then write amount of data specified by SD_SIZE*3

Bit	Bit Name	Initial Value	R/W	Description
8	BRE	0*2	R/W*1	SD_BUF Read Enable 0: Data cannot be read from SD_BUF0. 1: Data can be read from SD_BUF0. [Setting conditions] 1. When data set in SD_SIZE is stored in SD_BUF at single block transfer 2. When data set in SD_SIZE is stored in either bank 1 or bank 2 of SD_BUF at multiple block transfer [Clearing conditions] 1. When 0 is written to BRE 2. Reading of a block of data from SD_BUF by DMA transfer When data is read from SD_BUF0 by the CPU, clear BRE and then read amount of data specified by SD_SIZE*3. Even if a CRC error or an END error occurs while block data is read, data is stored in SD_BUF and BRE is set.
7	DAT0	Unknown	R	SDDAT0 Indicates the SDDAT0 state. 0: SDDAT0 is set to 0. 1: SDDAT0 is set to 1. If the data timeout (ERR3) is set but the response timeout (ERR6) is not set after the Erase command has been issued, the end of the Erase sequence (DAT0 = 1) is confirmed by polling DAT0. If a communications error or timeout occurs during a write sequence, the DAT0 bit may retain the value 0. While the SD clock (SDCLK) is stopped, the DAT0 bit retains the value before the clock is stopped.
6	ERR6	0*2	R/W*1	Response Timeout [Setting condition] When a response is not received even after 640 cycles of SDCLK have elapsed (including a response to a command issued within a command sequence*5) [Clearing condition] When 0 is written to ERR6 The command sequence is halted by a response timeout.*4
5	ERR5	0*2	R/W*1	SD_BUF Illegal Read Access [Setting conditions] 1. When SD_BUF is empty while SD_BUF0 is read 2. When data with a CRC error or END error is read from SD_BUF0 [Clearing condition] When 0 is written to ERR5
4	ERR4	0*2	R/W*1	SD_BUF Illegal Write Access [Setting conditions] 1. When data is written to SD_BUF0 while it is not in the data read/write command state 2. When data is written to SD_BUF0 while SD_BUF is full 3. When data is written to SD_BUF0 while an error occurs in the CRC status or CRC status length 4. When data is written to SD_BUF0 while the interface remains in a busy state for at least Ncycle after the CRC status [Clearing condition] When 0 is written to ERR4 Ncycle is set by bits 7 to 4 in SD_OPTION.

Bit	Bit Name	Initial Value	R/W	Description
3	ERR3	0*2	R/W*1	Data Timeout (except response timeout) [Setting conditions] 1. After R1b response, the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle. 2. After CRC status, the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle. 3. After write data, the CRC status is not received even after Ncycle has elapsed. 4. After read command, read data is not received even after Ncycle has elapsed. 5. After CMD12 has been issued within a command sequence, the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle. 6. After the reception of read data, read data for the next block are not received even after Ncycle has elapsed. 7. After release of the read wait state, read data for the next block are not received even after Ncycle has elapsed. [Clearing condition] When 0 is written to ERR3 Ncycle is set by bits 7 to 4 in SD_OPTION. The command sequence is halted by the data timeout.
2	ERR2	0*2	R/W*1	END Error [Setting conditions] 1. When an error occurs in the response length (and the end bit has not been detected) 2. When an error occurs in the read data length (and the end bit has not been detected among the valid bits) 3. When an error occurs in the CRC status length (and the end bit has not been detected) 4. An error in the length of a response to a command issued within a command sequence*5 (i.e. the end bit has not been detected) [Clearing condition] When 0 is written to ERR2 The command sequence is halted by the End error.*4
1	ERR1	0*2	R/W*1	CRC Error [Setting conditions] 1. When an error occurs in the CRC status (i.e. the received CRC status was not ‘010’) 2. When a CRC error occurs in the read data 3. When a CRC error occurs in the response 4. A CRC error in the response to a command issued within a command sequence*5 [Clearing condition] When 0 is written to ERR1 The command sequence is halted by the CRC error.*4
0	ERR0	0*2	R/W*1	CMD Error [Setting conditions] 1. The command index of the transmitted command differing from the command index of the received response 2. The command index of a command issued within a command sequence*5 differing from the command index of the received response [Clearing condition] When 0 is written to ERR0 The command sequence is halted by the CMD error.*4

Note 1. It is effective only if 0 is written.

Note 2. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 3. When the WMODE bit in HOST_MODE is 0, the single byte from the fraction of a full 16-bit unit is regarded as excess data due to an odd value for the number of bytes setting in SD_SIZE. When the WMODE bit in HOST_MODE is 1, the single byte or three bytes from the fraction of a full 64-bit unit are regarded as excess data due to an odd value for the number of bytes

setting in SD_SIZE, or the two bytes from the fraction of a full 64-bit unit are regarded as excess data due if the value for the number of bytes setting in SD_SIZE is even but is not on a four-byte boundary.

Note 4. After the C52PUB bit in SDIO_MODE has been set to 1, if a communications error or timeout for response occurs in response to the CMD52 that is issued, since the command sequence has not been completed, complete the sequence with error processing as in usage examples in **Figure 33.16** under **Section 33.4.8, IO_RW_EXTENDED (CMD53/Multiple Block Read)** or in **Figure 33.19** under **Section 33.4.9, IO_RW_EXTENDED (CMD53/Multiple Block Write)**.

Note 5. “Command issued within a command sequence” refers to CMD12 when automatic issuing is enabled for multiple block transfer by the setting in SD_CMD, CMD12 when the STP bit in SD_STOP is set to 1, or CMD52 when the C52PUB or IOABT bit in SDIO_MODE is set to 1.

33.2.8 SD_INFO1 Interrupt Mask Register (SD_INFO1_MASK)

The SD_INFO1 interrupt mask register (SD_INFO1_MASK) is used to enable or disable the SD_INFO1 interrupt.

When 0 is set in SD_INFO1_MASK while the corresponding flag in SD_INFO1 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	IMASK16	1	R/W	HPIRES interrupt masked
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9	IMASK9	1	R/W	INFO9 interrupt masked
8	IMASK8	1	R/W	INFO8 interrupt masked
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
4	IMASK4	1	R/W	INFO4 interrupt masked
3	IMASK3	1	R/W	INFO3 interrupt masked
2	IMASK2	1	R/W	INFO2 interrupt masked
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
0	IMASK0	1	R/W	INFO0 interrupt masked

33.2.9 SD_INFO2 Interrupt Mask Register (SD_INFO2_MASK)

The SD_INFO2 interrupt mask register (SD_INFO2_MASK) is used to enable or disable the SD_INFO2 interrupt.

When 0 is set in SD_INFO2_MASK while the corresponding flag in SD_INFO2 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	IMASK	1	R/W	ILA interrupt masked
14 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
11	—	1	R/W	Reserved The write value should always be 1.
10	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
9	BMASK1	1	R/W	BWE interrupt masked
8	BMASK0	1	R/W	BRE interrupt masked
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
6	EMASK6	1	R/W	ERR6 interrupt masked
5	EMASK5	1	R/W	ERR5 interrupt masked
4	EMASK4	1	R/W	ERR4 interrupt masked
3	EMASK3	1	R/W	ERR3 interrupt masked
2	EMASK2	1	R/W	ERR2 interrupt masked
1	EMASK1	1	R/W	ERR1 interrupt masked
0	EMASK0	1	R/W	ERR0 interrupt masked

33.2.10 SD Clock Control Register (SD_CLK_CTRL)

The SD clock control register (SD_CLK_CTRL) is used to control the SD clock (SDCLK) output and to set the frequency. Set SCLKEN to 1 before writing to SD_CMD to issue a command. Do not write to SD_CLK_CTRL while the SCLKDIVEN bit in SD_INFO2 is set to 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	—	0	R/W	Reserved The write value should always be 0.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
10	—	0*1	R/W	Reserved The write value should always be 0.
9	SDCLKOFFEN	0	R/W	SD Clock (SDCLK) Output Automatic Control Enable 0: Automatic control for SD clock (SDCLK) output is disabled. 1: Automatic control for SD clock (SDCLK) output is enabled. This function of automatic control for SD clock (SDCLK) output causes SDCLK output only within a command sequence. The timing with which SDCLK output starts and stops is as follows. SDCLK output starts after writing to SD_CMD. SDCLK output stops when 8 cycles of SDCLK have elapsed after the end of the command sequence. In addition, SDCLK is fixed to 0 while SCLKEN of SD_CLK_CTRL is 0, regardless of the value of this bit.
8	SCLKEN	0*1	R/W*2	SD Clock (SDCLK) Output Control Enable 0: SD clock (SDCLK) output is disabled. The SDCLK signal is fixed 0. 1: SD clock (SDCLK) output is enabled.
7	DIV7	0	R/W*2	SD Clock (SDCLK) *3 1000_0000b: (SDxϕ/4)/512 0100_0000b: (SDxϕ/4)/256 0010_0000b: (SDxϕ/4)/128 0001_0000b: (SDxϕ/4)/64 0000_1000b: (SDxϕ/4)/32 0000_0100b: (SDxϕ/4)/16 0000_0010b: (SDxϕ/4)/8 0000_0001b: (SDxϕ/4)/4 0000_0000b: (SDxϕ/4)/2 1111_1111b: SDxϕ/4 (x = 0, 1) Other settings are prohibited.
6	DIV6	0	R/W*2	
5	DIV5	1	R/W*2	
4	DIV4	0	R/W*2	
3	DIV3	0	R/W*2	
2	DIV2	0	R/W*2	
1	DIV1	0	R/W*2	
0	DIV0	0	R/W*2	

Note 1. This initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 2. Writing to SD_CLK_CTRL is impossible when the CBSY bit in SD_INFO2 is 1.

Note 3. SD Clock on ch2 must be set to 50 MHz or less.

Notes on when setting the SD clock (SDCLK) to P1ϕ (DIV[7:0] = 1111_1111b)

When changing the setting of bits DIV[7:0] to 1111_1111b, or from 1111_1111b to other setting, perform the following processing before writing to SD_CMD.

(1) Set the SCLKEN bit to 0 by writing to SD_CLK_CTRL. (Do not change the setting of bits other than SCLKEN at this time.)

(2) Change the setting of bits DIV[7:0] by writing to SD_CLK_CTRL. (Do not change the setting of bits other than DIV[7:0] at this time. The SCLKEN bit should retain the value 0.)

(3) Set the SCLKEN bit to 1 by writing to SD_CLK_CTRL. (Do not change the setting of bits other than SCLKEN at this time.)

Also when changing the setting of bits DIV[7:0] to 1111_1111b after having set the SDRST bit in SOFT_RST to 0 and then changed it to 1, perform this processing before writing to SD_CMD.

33.2.11 Transfer Data Length Register (SD_SIZE)

The transfer data length register (SD_SIZE) is used to specify the transfer data size.

Bit	Bit Name	Initial Value	R/W	Description
63 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
11, 10	—	All 0	R	Reserved
9 to 0	LEN9 to LEN0	1000000000	R/W	Transfer Data Size*1 These bits specify a size between 1 and 512 bytes for single block transfer. In cases of multiple block transfer with automatic issuing of CMD12 (CMD18 and CMD25), the only specifiable transfer data size is 512 bytes. Furthermore, in cases of multiple block transfer without automatic issuing of CMD12, as well as 512 bytes, 32, 64, 128, and 256 bytes are specifiable. However, in the reading of 32, 64, 128, and 256 bytes for the transfer of multiple blocks, this is restricted to multiple block transfer by CMD53. Additionally, if a command accompanies data transfer, do not set these bits to 0. Do not specify a data size larger than 512 bytes.

Note 1. Do not change the values of these bits when the CBSY bit in SD_INFO2 is 1.

33.2.12 SD Card Access Control Option Register (SD_OPTION)

The SD card access control option register (SD_OPTION) is used to set the bus width and timeout counter.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	WIDTH	0*1	R/W	Bus width*2 {WIDTH,WIDTH8} = 01: 8 bits width {WIDTH,WIDTH8} = 00: 4 bits width {WIDTH,WIDTH8} = 10,11: 1 bit width
14	—	1	R	Reserved This bit is always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to this bit.
13	WIDTH8	0*1	R/W	Bus width*2 See the description of the WIDTH bit.
12 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9	EXTOP	0*1	R/W	Timeout Mode Select 0: Bits TOP27 to TOP24 specify the timeout count from SDCLK*2^{13} to SDCLK*2^{27}. 1: Bits TOP27 to TOP24 specify the timeout count from SDCLK*2^{14} to SDCLK*2^{28}.
8	TOUTMASK	0*1	R/W	Timeout Mask 0: Enables timeout. 1: Disables timeout. (The ERR6 and ERR3 bits in SD_INFO2 and the E6 to E0 bits in SD_ERR_STS2 are not set.) If a timeout occurs while it is disabled, perform a software reset to terminate a command sequence.
7	TOP27	1*1	R/W	Timeout Counter*2 0000: SDCLK*2^{13} 0001: SDCLK*2^{14} ⋮ 1101: SDCLK*2^{26} 1110: SDCLK*2^{27} 1111: Setting prohibited
6	TOP26	1*1	R/W	
5	TOP25	1*1	R/W	
4	TOP24	0*1	R/W	
3	CTOP24	1*1	R/W	Card Detect Time Counter 0000: SDCLK*2^{10} 0001: SDCLK*2^{11} ⋮ 1101: SDCLK*2^{23} 1110: SDCLK*2^{24} 1111: Setting prohibited
2	CTOP23	1*1	R/W	
1	CTOP22	1*1	R/W	
0	CTOP21	0*1	R/W	

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 2. Do not change the values of these bits when the CBSY bit in SD_INFO2 is 1.

33.2.13 SD Error Status Register 1 (SD_ERR_STS1)

The SD error status register 1 (SD_ERR_STS1) indicates the CRC status, CRC error, End error, and CMD error.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 15	—	Unknown	R	Reserved These bits are always read as unknown. The write value should always be 0.
14	E14	0*1	R	These bits hold the CRC status. (normal: 010)
13	E13	1*1		
12	E12	0*1		
11	E11	0*1	R	Set to 1 when an error occurs in the CRC status.
10	E10	0*1	R	Set to 1 when a CRC error occurs in the read data.
9	E9	0*1	R	Set to 1 when a CRC error occurs in the response to a command issued within a command sequence*2. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E8.
8	E8	0*1	R	Set to 1 when a CRC error occurs in a response (other than a response to a command issued within a command sequence*2).
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5	E5	0*1	R	Set to 1 when an error occurs in the CRC status length (and the end bit has not been detected).
4	E4	0*1	R	Set to 1 when an error occurs in the read data length (and the end bit has not been detected among the valid bits).
3	E3	0*1	R	Set to 1 when an error occurs in the response length to a command issued within a command sequence*2. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E2.
2	E2	0*1	R	Set to 1 when an error occurs in the response length (other than a response to a command issued within a command sequence*2).
1	E1	0*1	R	Set to 1 when an error occurs in the command index of the response to a command issued within a command sequence*2. In cases where CMD12 is issued by setting a command index in SD_CMD, this is Indicated in E0.
0	E0	0*1	R	Set to 1 when an error occurs in the command index of a response (other than a response to a command issued within a command sequence*2).

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 2. “Command issued within a command sequence” refers to CMD12 when automatic issuing is enabled for multiple block transfer by the setting in SD_CMD, CMD12 when the STP bit in SD_STOP is set to 1, or CMD52 when the C52PUB or IOABT bit in SDIO_MODE is set to 1.

33.2.14 SD Error Status Register 2 (SD_ERR_STS2)

The SD error status register 2 (SD_ERR_STS2) indicates the timeout state. Ncycle is set by bits 7 to 4 in SD_OPTION.

Bit	Bit Name	Initial Value	R/W	Description
63 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
6	E6	0*1	R	Set to 1 when the interface remains in a busy state for at least Ncycle after the CRC status.
5	E5	0*1	R	Set to 1 when the CRC status is not received even after Ncycle has elapsed after data writing.
4	E4	0*1	R	Set to 1 when read data is not received even after Ncycle has elapsed after the command has been read. Set to 1 when read data for the next block are not received even after Ncycle has elapsed after the reception of read data. Set to 1 when read data for the next block are not received even after Ncycle has elapsed after release of the read wait state.
3	E3	0*1	R	Set to 1 when the interface remains in a busy state for at least Ncycle after CMD12 has been issued within a command sequence. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E2.
2	E2	0*1	R	Set to 1 when the interface remains in a busy state for at least Ncycle after R1b response.
1	E1	0*1	R	Set to 1 when the response to a command issued within a command sequence*2 is not received even after 640 cycles of SDCLK have elapsed. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E0.
0	E0	0*1	R	Set to 1 when the response (other than a response to a command issued within a command sequence*2) is not received even after 640 cycles of SDCLK have elapsed.

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 2. “Command issued within a command sequence” refers to CMD12 when automatic issuing is enabled for multiple block transfer by the setting in SD_CMD, CMD12 when the STP bit in SD_STOP is set to 1, or CMD52 when the C52PUB or IOABT bit in SDIO_MODE is set to 1.

33.2.15 SD Buffer Read/Write Register (SD_BUF0)

Bit	Bit Name	Initial Value	R/W	Description
63 to 0	BUF63 to BUF0	Unknown	R/W	When writing to the SD card, the write data is written to this register. When reading from the SD card, the read data is read from this register. This register is internally connected to two 512-byte buffers (SD_BUF). When both buffers are not empty at multiple block read, suspend data reception by stopping the SD clock. When either buffer becomes empty, restart data reception by starting supply of the SD clock.

Note: When using the DMAC, the bus width should be fixed at 64 bits.

33.2.16 SDIO Mode Control Register (SDIO_MODE)

The SDIO mode control register (SDIO_MODE) controls the CMD52 issuance and the read wait state at multiple block transfer, and the reception of SDIO interrupt. C52PUB and IOABT should not be set to 1 simultaneously.

Bit	Bit Name	Initial Value	R/W	Description
63 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9	C52PUB	0	R/W	SDIO None Abort • When C52PUB is set to 1 in the CMD53 (multiple block) write sequence, CMD52 is automatically issued between blocks if SD_BUF becomes empty.C52PUB is automatically cleared to 0 after reception of the response to CMD52 is completed. Additionally, if C52PUB is set to 1 while the last block is being transferred, CMD52 is not issued. In this case, C52PUB is automatically cleared to 0 after the access end flag has been set to 1. • When C52PUB and RWREQ are set to 1 in the CMD53 (multiple block) read sequence, the block transfer enters the read wait state between blocks and CMD52 is automatically issued. C52PUB is automatically cleared to 0 after reception of the response to CMD52 is completed. Additionally, if C52PUB is set to 1 while the last block is being transferred, CMD52 is not issued. In this case, C52PUB is automatically cleared to 0 after the access end flag has been set to 1. • If C52PUB is set to 1 in the CMD53 (multiple block) read sequence, be sure to set RWREQ to 1 as well as C52PUB. • Set SD_ARG before setting C52PUB to 1. • Set C52PUB to 1 after the response end flag has been set.
8	IOABT	0	R/W	SDIO Abort • When IOABT is set to 1 in the CMD53 (multiple block) sequence, the CMD53 sequence is halted and CMD52 is issued. However, if a command sequence is halted because of a communications error or timeout, CMD52 is not issued. Although continued buffer access is possible even after IOABT has been set to 1, the buffer access error bit (ERR5 or ERR4) in SD_INFO2 will be set accordingly. Set SD_ARG before setting IOABT to 1. • When IOABT has been set to 1 during transfer for single block write, the access end flag is set when SD_BUF becomes empty, and CMD52 is not issued. If SD_BUF does contain data, the access end flag is set on completion of reception of the busy state without CMD52 having been issued. • When IOABT has been set to 1 during transfer for single block read, the access end flag is set immediately after setting of IOABT and CMD52 is not issued. • When IOABT is set to 1 during reception of the busy state after an R1b response, the access end flag is set on completion of reception of the busy state without CMD52 having been issued. • When IOABT is set to 1 after a command sequence has been completed, CMD52 is not issued and the access end flag is not set. • Set IOABT to 1 after the response end flag has been set. Set IOABT to 0 after the access end flag has been set.
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.

Bit	Bit Name	Initial Value	R/W	Description
2	RWREQ	0	R/W	Read Wait Request When RWREQ is set to 1 in the CMD53 (multiple block) read sequence, the block transfer enters the read wait state between blocks. [Read wait state releasing] (1) The read wait state is released, when RWREQ is cleared to 0 in the read wait state. (2) When IOABT is set to 1 in the read wait state, RWREQ is automatically cleared to 0 after CMD52 has been issued, and then the read wait state is released. (3) When C52PUB and RWREQ are set to 1 simultaneously in the CMD53 (multiple block) read sequence, the read wait state is not automatically released. Therefore, after the CMD52 response is received, clear RWREQ. (Be sure to set RWREQ and C52PUB simultaneously.) When RWREQ is set to 1 while the last block in the CMD53 (multiple block) read sequence is transferred, the read wait state is not entered and RWREQ is automatically cleared to 0 by setting access end. Set RWREQ to 1 after the response end flag has been set.
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
0	IOMOD	0	R/W	SDIO Mode*1 0: Disables the SD/MMC host interfaces to receive SDIO interrupt from the SDIO card 1: Enables the SD/MMC host interfaces to receive SDIO interrupt from the SDIO card

Note 1. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1.

33.2.17 SDIO Interrupt Flag Register (SDIO_INFO1)

The SDIO interrupt flag register (SDIO_INFO1) indicates the status regarding to the SDIO card access. To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	EXWT	0*2	R/W*1	[Setting condition] While the last block in the CMD53 (multiple block) read sequence is transferred, RWREQ in SDIO_MODE is set to 1. [Clearing condition] When 0 is written to EXWT
14	EXPUB52	0*2	R/W*1	[Setting conditions] 1. While the last block in the CMD53 (multiple block) sequence is transferred, C52PUB in SDIO_MODE is set to 1. 2. While C52PUB is set to 1 in the CMD53 (multiple block) write sequence, the last block is transferred. [Clearing condition] When 0 is written to EXPUB52
13 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2, 1	—	All 0*2	R/W	Reserved The write value should always be 1. The value may change during operation.
0	IOIRQ	0*2	R/W*1	[Setting condition] When SDIO interrupt from an SDIO card is received while IOMOD in SDIO_MODE is set to 1. [Clearing condition] When 0 is written to IOIRQ*3

Note 1. It is effective only if 0 is written.

Note 2. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 3. Before clearing this bit, access the SDIO card to negate the SDIO interrupt signal from the SDIO card. If the interrupt signal is not negated, this bit may be set again.

33.2.18 SDIO_INFO1 Interrupt Mask Register (SDIO_INFO1_MASK)

The SDIO_INFO1 interrupt mask register (SDIO_INFO1_MASK) enables or disables the SD_INFO1 interrupt. When 0 is set in SDIO_INFO1_MASK while the corresponding flag in SD_INFO1 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	MEXWT	1	R/W	EXWT interrupt masked
14	MEXPUB52	1	R/W	EXPUB52 interrupt masked
13 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2	—	1	R/W	Reserved The write value should always be 1.
1	—	1	R/W	Reserved The write value should always be 1.
0	IOMSK	1	R/W	IOIRQ interrupt masked

33.2.19 DMA Mode Enable Register (CC_EXT_MODE)

The DMA mode enable register (CC_EXT_MODE) enables the DMA transfer.

Bit	Bit Name	Initial Value	R/W	Description
63 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
12	—	1	R	Reserved This bit is always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to this bit.
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9, 8	—	All 0	R/W	Reserved The write value should always be 0.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5	—	0	R/W	Reserved The write value should always be 0.
4	—	1	R	Reserved This bit is always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to this bit.
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
1	DMASDRW	0	R/W	SD_BUF Read/Write DMA Transfer*1 0: The SD_BUF read/write DMA transfer is disabled. 1: The SD_BUF read/write DMA transfer is enabled.
0	—	0	R/W	Reserved The write value should always be 0.

Note 1. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1.

33.2.20 Software Reset Register (SOFT_RST)

The software reset register (SOFT_RST) sets a software reset. Also use this register to check that release from the reset state has been completed before attempting to use the SD/MMC host interfaces and before attempting access to the other registers.

Bit	Bit Name	Initial Value	R/W	Description
63 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2, 1	—	11	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
0	SDRST	1	R/W	Software Reset of SD Interface Unit 0: Reset 1: Reset released

33.2.21 Version Register (VERSION)

The version register (VERSION) indicates the version of the SD/MMC host interfaces.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15, 14	UR7, UR6	11	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
13, 12	UR5, UR4	00	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
11 to 8	UR3 to UR0	H'C	R	Version of Renesas' IP
7 to 0	IP7 to IP0	H'10	R	Version of introductory IP

33.2.22 Host Interface Mode Setting Register (HOST_MODE)

The host interface mode setting register (HOST_MODE) selects the width for access to the data bus.

Bit	Bit Name	Initial Value	R/W	Description
63 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8	BUSWIDTH	0	R/W	Width for Access to SD_BUF*1*2 Read or write access to SD_BUF0 can be performed with the specified width for access. 0: 16-bit access 1: 32-bit access This bit is enabled while the WMODE bit is set to 1.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
1	ENDIAN	0	R/W	SD_BUF0 data swap
0	WMODE	0	R/W	Width for Access to SD_BUF*1*2 Read or write access to SD_BUF0 can be performed with the specified width for access. 0: 64-bit access 1: 16-bit or 32-bit access

Note 1. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1.

Note 2. When using the built-in DMAC of this module, fix the bus width to 64 bits.

33.2.23 SD Interface Mode Setting Register (SDIF_MODE)

The SD interface mode setting register specifies transfer mode.

Bit	Bit Name	Initial Value	R/W	Description
63 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9	—	0*1	R/W	Reserved The write value should always be 0.
8	NOCHKCR	0*1	R/W	CRC Check Mask (test command for MMC supported) Enables or disables checking of the CRC16 and CRC status. 0: Enables the CRC check. 1: Disables the CRC check (the CRC16 value is ignored at read, and the CRC status is not detected at write)
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	—	0*1	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

33.2.24 SD Status Register (SD_STATUS)

The effective bit of the SD status register controls the output value on the SD0_RST# pin.

Bit	Bit Name	Initial Value	R/W	Description
63 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
1	SD_RST	0	R/W	Controls the output value on the SD0_RST# pin. 0: The output value on the SD0_RST# pin is 0. 1: The output value on the SD0_RST# pin is 1.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.

33.2.25 DMAC Transfer Mode Register (DM_CM_DTRAN_MODE)

The DMAC Transfer Mode Register (DM_CM_DTRAN_MODE) sets the operation mode of the module built-in DMAC.

High 32 bits (bit 63 to 32) are read only 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
17, 16	CH_NUM	00	R/W	DMAC channel selector 00: SD down stream 01: SD up stream Other settings are prohibited.
15 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5, 4	BUS_WIDTH [1:0]	11	R/W	Bus width selector 11: 64-bit Other settings are prohibited.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.

33.2.26 DMAC Transfer Control Register (DM_CM_DTRAN_CTRL)

The DMAC Transfer Control Register (DM_CM_DTRAN_CTRL) controls the module built-in DMAC operation. High 32 bits (bit 63-32) are read only 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8	—	0	R/W	Reserved The write value should always be 0.
7 to 1	—	All 0	R	Reserved
0	DM_START	0	R/W	DMAC Start Writing 1 to this bit starts DMAC operation. This bit is automatically cleared when DMA transfer is started.

33.2.27 DMAC Reset Register (DM_CM_RST)

High 32 bits (bit 63 to 32) are read only 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 10	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
9	DTRANRST1	1	R/W	Soft resets of the module built-in DMAC channel 1 0: Reset 1: Reset released
8	DTRANRST0	1	R/W	Soft resets of the module built-in DMAC channel 0 0: Reset 1: Reset released
7 to 1	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
0	SEQRST	1	R/W	Soft resets of the sequencer 0: Reset 1: Reset released

NOTE

Make sure there is no communication to the SD/MMC device before applying a software reset using this register.

33.2.28 DMAC Interrupt Register 1 (DM_CM_INFO1)

The DMAC interrupt register 1 (DM_CM_INFO1) indicates the status of the module built-in DMAC and a sequencer. To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
20	DTRANEND1	0*1	R/W	Module built-in DMAC Channel 1 Transfer End [Setting conditions] 1. When transfer of DMAC channel 1 is completed 2. When an error occurs on DMAC channel 1 [Clearing condition] When 0 is written to DTRANEND1
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	DTRANEND0	0*2	R/W	Module built-in DMAC Channel 0 Transfer End [Setting conditions] 1. When transfer of DMAC channel 0 is completed 2. When an error occurs on DMAC channel 0 [Clearing condition] When 0 is written to DTRANEND0
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	SEQEND	0*3	R/W	Sequencer Operation End [Setting conditions] 1. When operation of a sequencer is completed 2. When a sequencer error occurs [Clearing condition] When 0 is written to SEQEND

Note 1. The initial value is applied at a reset and when the DTRANRST1 bit in DM_CM_RST is 0.

Note 2. The initial value is applied at a reset and when the DTRANRST0 bit in DM_CM_RST is 0.

Note 3. The initial value is applied at a reset and when the SEQRST bit in DM_CM_RST is 0.

33.2.29 DM_CM_INFO1 Interrupt Mask Register (DM_CM_INFO1_MASK)

The DM_CM_INFO1 interrupt mask register (DM_CM_INFO1_MASK) enables or disables the DM_CM_INFO1 interrupt. When 0 is set in DM_CM_INFO1_MASK while the corresponding flag in DM_CM_INFO1 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 21	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
20	DTRANEND1_MASK	1	R/W	DTRANEND1 interrupt masked
19 to 17	—	All 1	R/W	Reserved The write value should always be 1.
16	DTRANEND0_MASK	1	R/W	DTRANEND0 interrupt masked
15 to 1	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
0	SEQEND_MASK	1	R/W	SEQEND interrupt masked

33.2.30 DMAC Interrupt Register 2 (DM_CM_INFO2)

The DMAC interrupt register 2 (DM_CM_INFO2) indicates the status of the module built-in DMAC and a sequencer. To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
19, 18	—	00	R	Reserved The write value should always be 0.
17	DTRANERR1	0*1	R/W	Module built-in DMAC Channel 1 Error [Setting condition] When an error occurs on DMAC channel 1 [Clearing condition] When 0 is written to DTRANERR1
16	DTRANERR0	0*2	R/W	Module built-in DMAC Channel 0 Error [Setting condition] When an error occurs on the DMAC channel 0 [Clearing condition] When 0 is written to DTRANERR0
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	SEQERR	0*3	R/W	Sequencer error [Setting condition] When a sequencer error occurs [Clearing condition] When 0 is written to SEQERR

Note 1. The initial value is applied at a reset and when the DTRANRST1 bit in DM_CM_RST is 0.

Note 2. The initial value is applied at a reset and when the DTRANRST0 bit in DM_CM_RST is 0.

Note 3. The initial value is applied at a reset and when the SEQRST bit in DM_CM_RST is 0.

33.2.31 DM_CM_INFO2 Interrupt Mask Register (DM_CM_INFO2_MASK)

The DM_CM_INFO2 interrupt mask register (DM_CM_INFO2_MASK) enables or disables the DM_CM_INFO2 interrupt. When 0 is set in DM_CM_INFO2_MASK while the corresponding flag in DM_CM_INFO2 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 20	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
19, 18	—	11	R	Reserved The write value should always be 1.
17	DTRANERR1_MASK	1	R/W	DTRANERR1 interrupt masked
16	DTRANERR0_MASK	1	R/W	DTRANERR0 interrupt masked
15 to 1	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
0	SEQERR_MASK	1	R/W	SEQERR interrupt masked

33.2.32 DMAC Transfer Address Register (DM_DTRAN_ADDR)

The DMAC Transfer Address Register (DM_DTRAN_ADDR) sets the transfer destination and source address of the module built-in DMAC.

Higher-order 32 bits (bit 63 to 32) are read only 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 3	DADDR	All 0	R/W	Destination address / Source address (8 bytes unit) Note that the value of DM_DTRAN_ADDR + transfer data length is less than or equal to 2^{32}.
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.

33.3 Operation

33.3.1 SD Interface

(1) SD Data Format

When data is read from the SD card, the procedure is as follows.

1. The SD/MMC host interface receives data from the SD card via the SDDAT signal. (SDDAT signal: see **Figure 33.2**, **Figure 33.3** and **Figure 33.4**.)
2. The receive data is stored in SD_BUF of the SD/MMC host interfaces. (SD_BUF store data: see **Figure 33.5**)
3. The data stored in SD_BUF is read from SD_BUF0. (Reading from SD_BUF0: see **Table 33.4**)

When data is written to the SD card, the above procedure will be reversed.

When accessing SD_BUF0, caution should be taken for the transfer order in SDDAT and the store order in SD_BUF. In addition, data stored in SD_BUF0 can be replaced in bytes with the EXT_SWAP. (See **Figure 33.5**)

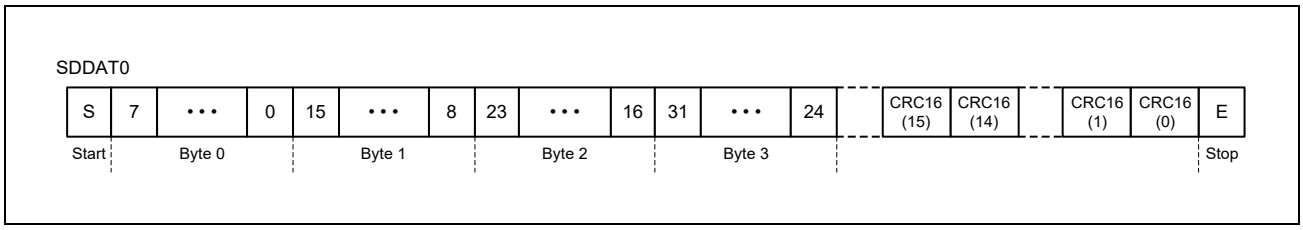


Figure 33.2 SDDAT in 1-Bit Width Mode

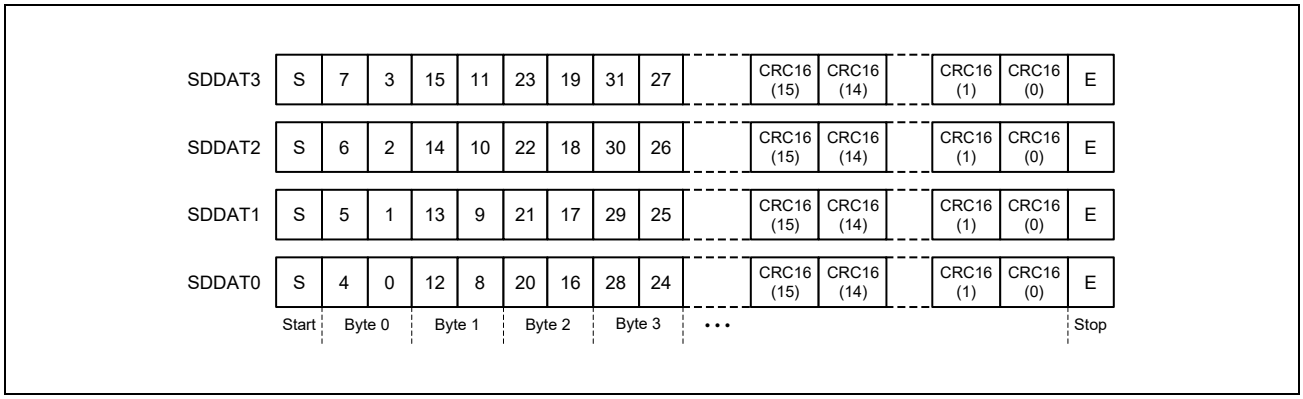


Figure 33.3 SDDAT in 4-Bit Width Mode

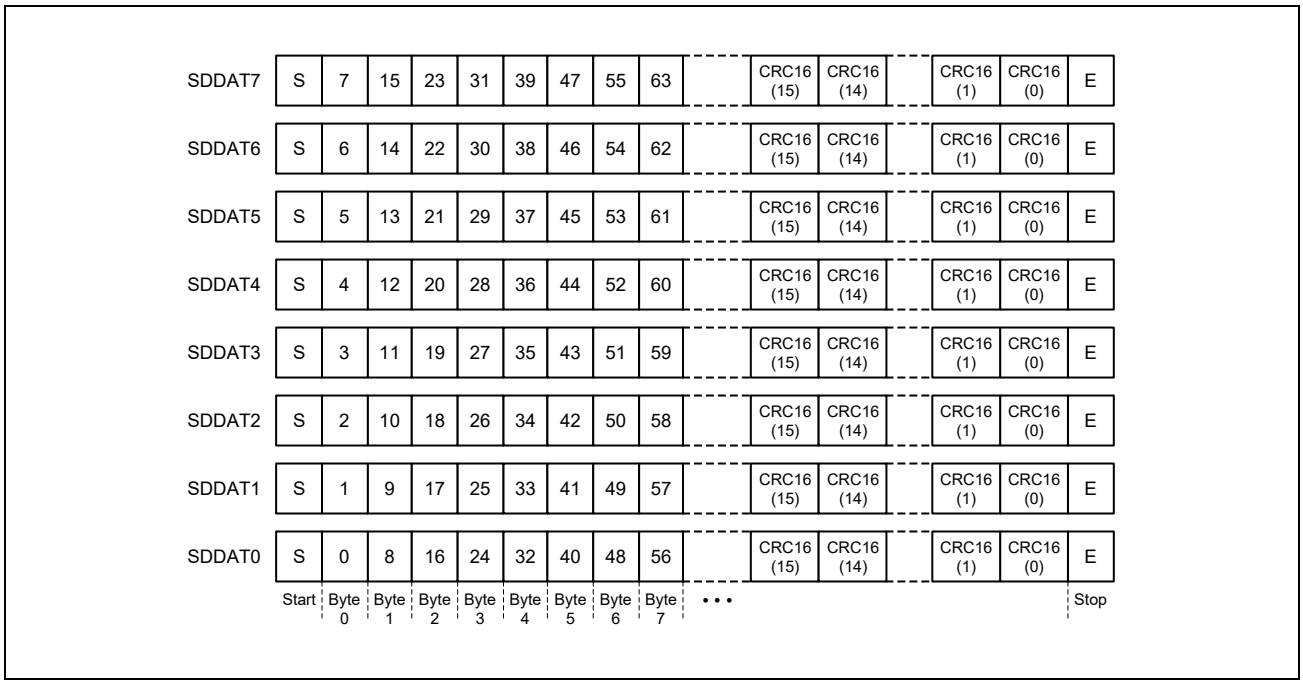


Figure 33.4 SDDAT in 8-Bit Width Mode

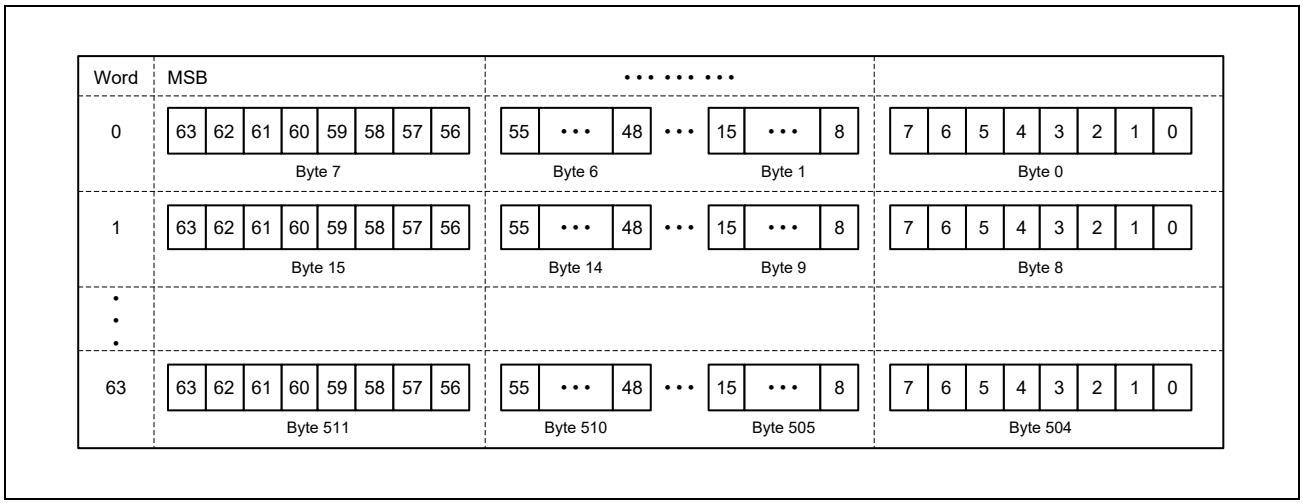


Figure 33.5 SD_BUF Store Data

Table 33.4 Reading from SD_BUF0

WMODE*1	BUSWIDTH*1	ENDIAN*1	Read Data*2
0	0	0	H'0123_4567_89AB_CDEF
0	0	1	H'EFCD_AB89_6745_2301
1	1	0	H'89AB_CDEF (1st) H'0123_4567 (2nd)
1	1	1	H'EFCD_AB89 (1st) H'6745_2301 (2nd)
1	0	0	H'CDEF (1st) H'89AB (2nd) H'4567 (3rd) H'0123 (4th)
1	0	1	H'EFCD (1st) H'AB89 (2nd) H'6745 (3rd) H'2301 (4th)

Note 1. The name of a bit in HOST_MODE.

Note 2. When the data stored in SD_BUF is H'0123_4567_89AB_CDEF

(2) Bus Signal Voltage Switch

Change the electric potential of the bus signal in the following procedure after checking that the SD card supports 1.8 V.

(1) Issuing CMD11
Perform command sequence processing of CMD11.

(2) Stopping the SD clock (a)
Set the SCLKEN bit in the SD_CLK_CTRL to 0 to stop*1 the output of the SD clock. When the SDCLKOFFEN bit in the SD_CLK_CTRL register is 1, the SDCLKOFFEN bit is also set to 0.

Note 1. When the SDCLKOFFEN bit in the SD_CLK_CTRL register is 1, the SD clock has automatically been stopped.

(3) Checking the value of SDDAT
Check that the DAT0 bit in the SD_INFO2 register is 0.

(4) Changing the supply voltage of the host device
Change the voltage which is supplied through the power supply pin of the given channel (SD0_PVDD for channel 0 or SD1_PVDD for channel 1) from 3.3 V to 1.8 V.

For details, see the following sections:

Section 45.3.4, Port Function Control Register,
Section 45.3.7, Driving Ability Control Register,
Section 45.3.16, SD Ch0 IO Voltage Mode Control Register,
Section 45.3.17, SD Ch1 IO Voltage Mode Control Register.

(5) Starting supply of the SD clock (b)
After the SD clock has been stopped ((a) above) and 5 ms or more has elapsed, set the SCLKEN bit in the SD_CLK_CTRL register to 1 and allow the output of the SD clock. The SDCLKOFFEN bit must be 0.

(6) Checking the value of SDDAT
After supplying the SD clock has been started ((b) above) and 1 ms or more has elapsed, check that the DAT0 bit in the SD_INFO2 register is 1. It is possible to set the SDCLKOFFEN bit in the SD_CLK_CTRL register to 1 and allow SD Clock (SDCLK) Output Automatic Control Enable.

33.3.2 Card Detect/Write Protect

(1) Card Detect

The SD/MMC host interface has two types of card detect functions as described in the following.

- Card detect with ISDCD

  Figure 33.6 shows the timing chart of card detect using ISDCD. ISDCD is connected to the card socket and pulled up on the host device. The resistance of the pull-up resistor is decided by the specification of the SD host device.

  [Card insertion]

  ISDCD is pulled down when a card is inserted. At this time, if ISDCD has been pulled down for the Mcycle period (set in SD_OPTION), INFO4 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

  [Card removal]

  ISDCD is pulled up when a card is removed. At this time, if ISDCD has been pulled up for the Mcycle period (set in SD_OPTION), INFO3 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

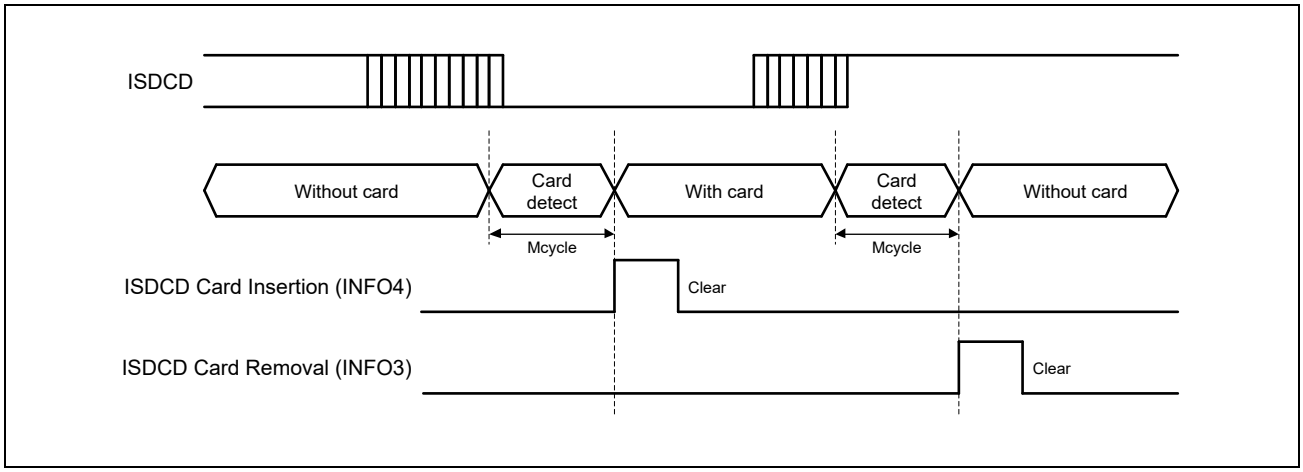


Figure 33.6 Example of Card Detect with ISDCD

- SD card detect with SDDAT3

Figure 33.7 shows the timing chart when the SD card is detected using SDDAT3. In addition, SDDAT3 is pulled down on the host device. The resistance of the pull-down resistor is decided by the specification of the SD host device.

[Card insertion]

When an SD card is inserted, SDDAT3 is pulled up. Accordingly, INFO9 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

[Card removal]

When an SD card is removed, SDDAT3 is pulled down. Accordingly, INFO8 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

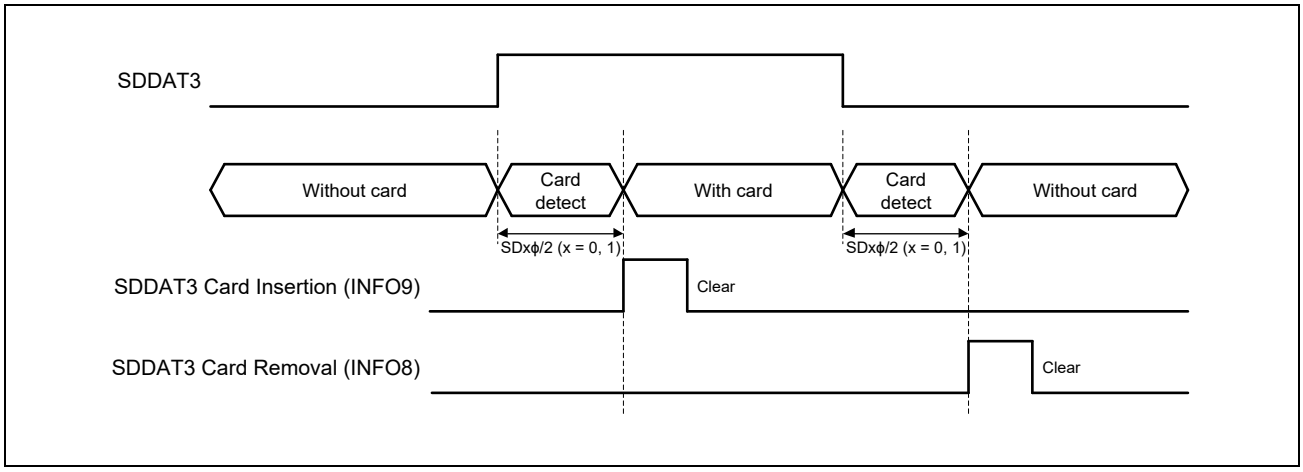


Figure 33.7 SD Card Detect with SDDAT3

(2) Write Protect

The SD/MMC host interface has two types of write protect functions.

- Write protect with ISDWP

ISDWP is connected to the card socket, and pulled up or pulled down by the card insertion. The selection of pulling up or pulling down and the resistance value is decided by the specifications of the SD host device. As the ISDWP state is reflected to INFO7 in SD_INFO1, the write protect is decided after the SD card is inserted.

- Write protect with command

The card's internal write protection and the card lock/unlock operation are realized by the command.

33.3.3 Interrupt Request

(1) Interrupt Request

The SD/MMC host interface has the interrupt requests shown in **Table 33.5** shows the relationship between the interrupt flag registers and the interrupt mask registers. When a bit in an interrupt mask register is set to 0, an interrupt occurs by setting the corresponding bit in the interrupt flag register to 1.

To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Table 33.5 Interrupt Request

	Interrupt Flag Register		Interrupt Mask Register	
Interrupt Request	Register Name	Bit Name	Register Name	Bit Name
Card access interrupt*1	SD_INFO1	INFO2	SD_INFO1_MASK	IMASK2
		INFO0		IMASK0
	SD_INFO2	ILA	SD_INFO2_MASK	IMASK
		BWE		BMASK1
		BRE		BMASK0
		ERR6		EMASK6
		ERR5		EMASK5
		ERR4		EMASK4
		ERR3		EMASK3
		ERR2		EMASK2
		ERR1		EMASK1
		ERR0		EMASK0
SDIO access interrupt*2	SDIO_INFO1	EXWT	SDIO_INFO1_MASK	MEXWT
		EXPUB52		MEXPUB52
		IOIRQ		IOMSK
Card detect interrupt*1	SD_INFO1	INFO9	SD_INFO1_MASK	IMASK9
		INFO8		IMASK8
		INFO4		IMASK4
		INFO3		IMASK3
DMAC interrupt*1	DM_CM_INFO1	DTRANEND1	DM_CM_INFO1_MASK	DTRANEND1_MASK
		DTRANEND0		DTRANEND0_MASK
		SEQEND		SEQEND_MASK
	DM_CM_INFO2	DTRANERR1	DM_CM_INFO2_MASK	DTARERR1_MASK
		DTRANERR0		DTARERR0_MASK
		SEQERR		SEQERR_MASK

Note 1. Interrupt signal “OXMNIRQ” is asserted.

Note 2. Interrupt signal both “OXMNIRQ” and “OXASIOIRQ” are asserted.

33.3.4 Communications Errors and Timeouts

- Communications Errors and Timeouts

Table 33.6 and **Table 33.7** show the relationships between the SD card interrupt flag register and SD error status register for communications errors and timeouts, respectively. When a bit in the SD card interrupt flag register is set to 1, the corresponding bit in the SD error status register is set to 1. The values of the SD error status register are cleared by writing to SD_CMD or writing 0 to the SDRST bit in SOFT_RST.

Table 33.6 Communications Errors

Communication Error	SD Interrupt Flag Register		SD Error Status Register		Description
	Register Name	Bit Name	Register Name	Bit Name	
END error	SD_INFO2	ERR2	SD_ERR_STS1	E5	When an error occurs in the CRC status length
				E4	When an error occurs in read data length
				E3	When an error occurs in the response length to a command issued within a command sequence
				E2	When an error occurs in the response length (other than a response to a command issued within a command sequence)
CRC error		ERR1		E11	When an error occurs in the CRC status
				E10	When a CRC error occurs in the read data
				E9	When a CRC error occurs in the response to a command issued within a command sequence
				E8	When a CRC error occurs in the response (other than a response to a command issued within a command sequence)
CMD error		ERR0		E1	The command index of the transmitted command differed from the command index of the received response (for a command issued within a command sequence)
				E0	The command index of the transmitted command differed from the command index of the received response (for a command issued other than within a command sequence)

Table 33.7 Timeouts

Timeout	SD Interrupt Flag Register		SD Error Status Register		Description
	Register Name	Bit Name	Register Name	Bit Name	
Response timeout	SD_INFO2	ERR6	SD_ERR_STS2	E1	When the response to a command issued within a command sequence is not received even after 640 cycles of SDCLK have elapsed
				E0	When the response (other than a response to a command issued within a command sequence) is not received even after 640 cycles of SDCLK have elapsed
Data timeout (other than response timeout)		ERR3		E6	When the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle*1 after the CRC status
				E5	When the CRC status is not received t even after Ncycle*1 has elapsed after data writing
				E4	When read data is not received even after Ncycle*1 has elapsed after read command
					When read data for the next block are not received even after Ncycle*1 has elapsed after the reception of read data
					When read data for the next block are not received even after Ncycle*1 has elapsed after release of the read wait state
				E3	When the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle*1 after CMD12 has been issued within a command sequence
				E2	When the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle*1 after R1b response

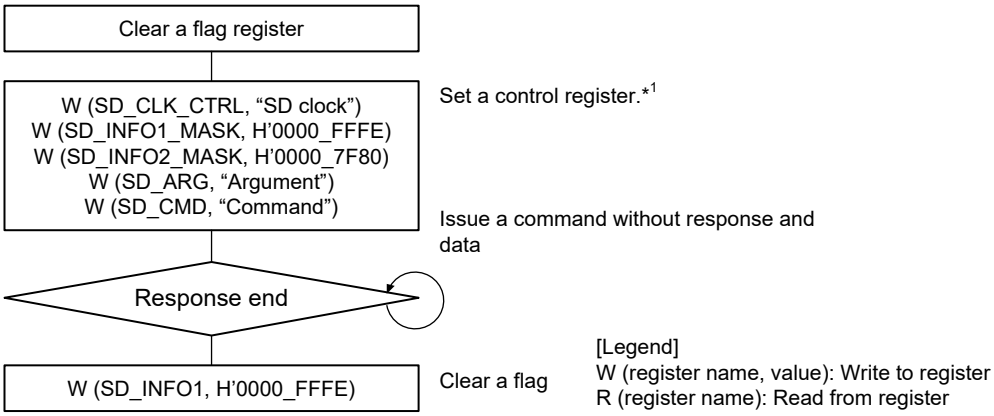
Note 1. Ncycle is set by bit 7 to bit 4 in SD_OPTION.

33.4 Usage Example

33.4.1 Command without Data Transfer

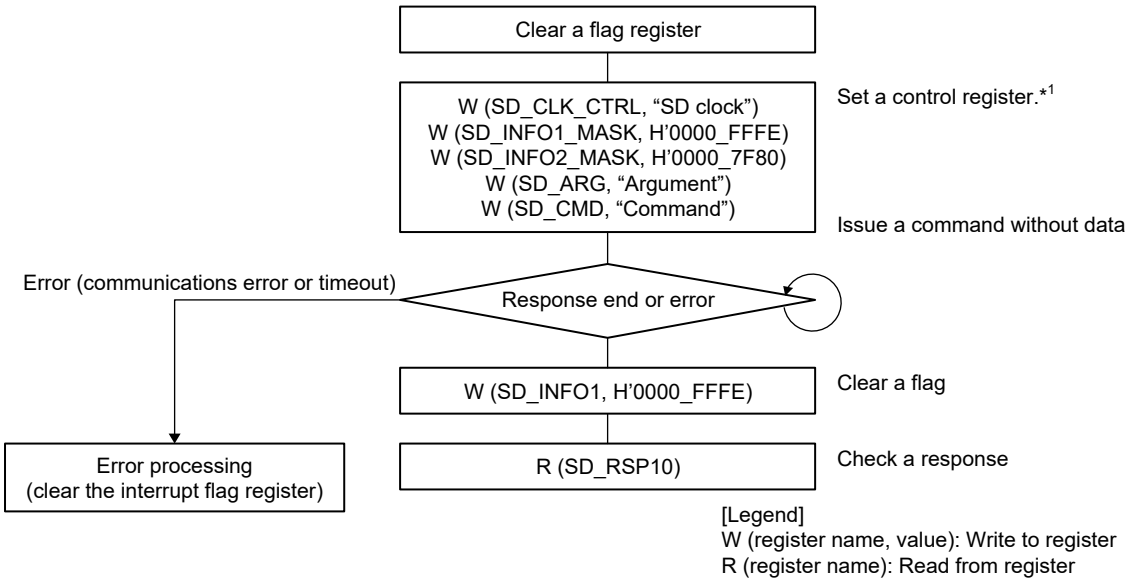
(1) Flowchart

Figure 33.8 and **Figure 33.9** show flowchart examples.



Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.8 Flow Example of Command without Response and Data



Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.9 Flow Example of Command without Data

(2) Operation for Command without Data Transfer

The legend below is used for description of register read/write.

W (register name, value):Write to register
R (register name):Read from register

The operation is described as below.

(a) Command without response and data

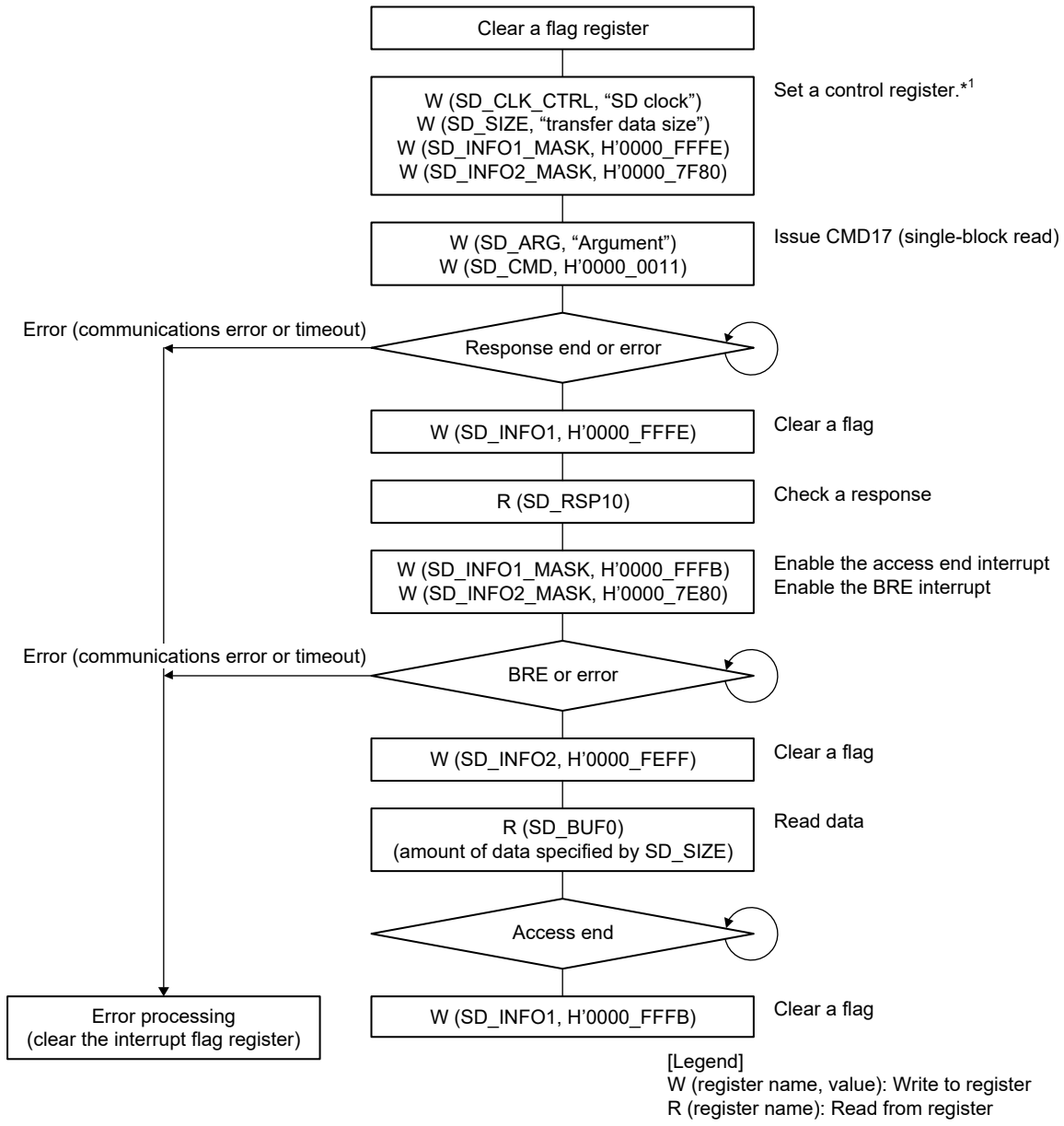
1. Flag register clear.
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set.
   Set the SD clock (SDCLK), interrupt mask, and so on. (SD_CLK_CTRL, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue.
   Set CMD Argument in SD_ARG and write to SD_CMD.
   Accordingly, CMD is issued, and the operation is started.
4. Flag clear.
   When transmission of a command is completed, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0.

(b) Command without data

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set.
   Set the SD clock (SDCLK), interrupt mask, and so on. (SD_CLK_CTRL, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue.
   Set CMD Argument in SD_ARG and write to SD_CMD.
   Accordingly, CMD is issued, and the operation is started.
4. Flag clear.
   When a response is received, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0.
5. Read a response from SD_RSP10.
   Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

33.4.2 Single Block Read

(1) Flowchart



Note: The error processing of SDR104/HS200 requires a re-tuning.

Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.10 Single Block Read Flowchart Example

(2) Operation for Single Block Read

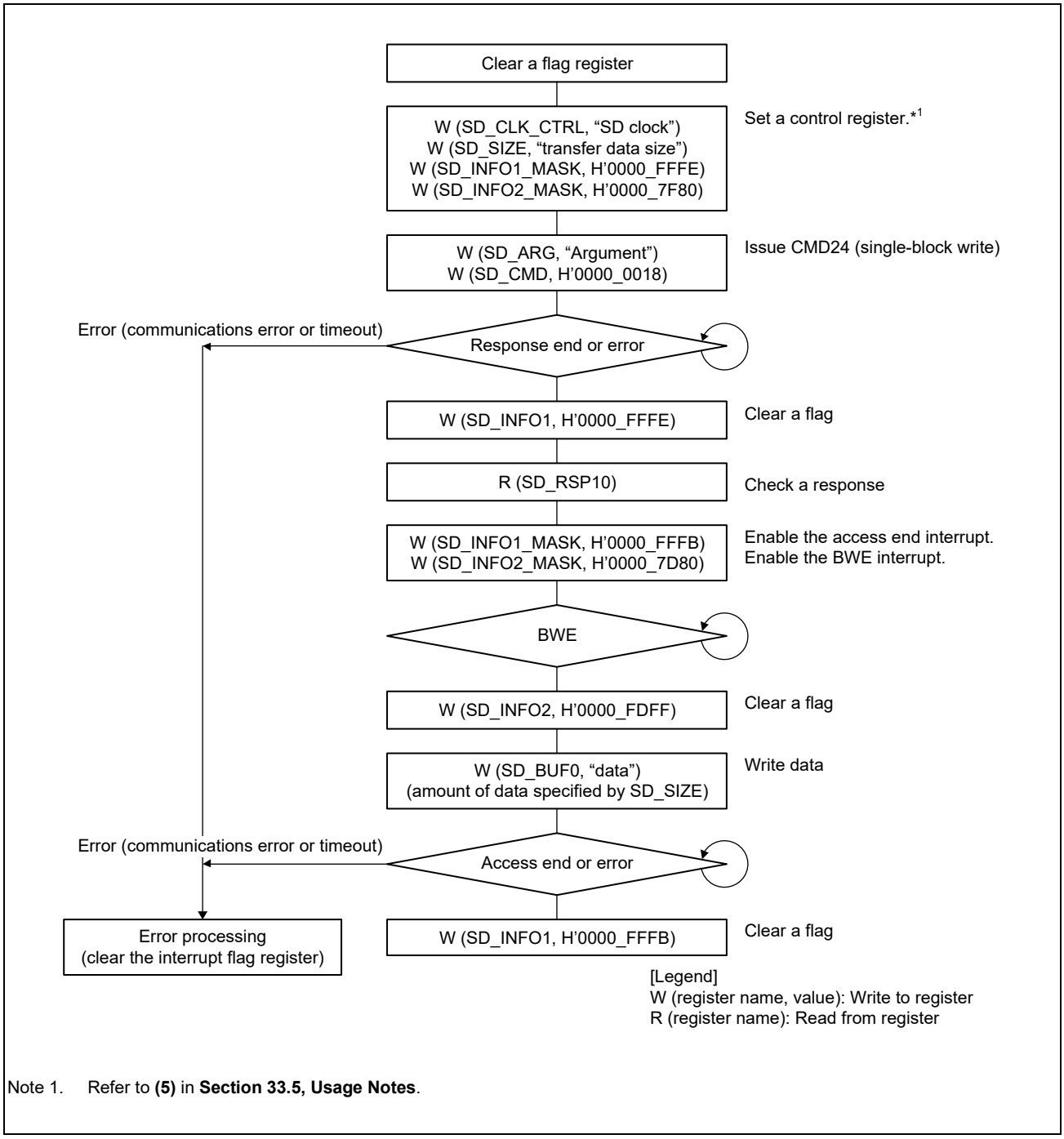
The operation of the single block read is described below.

1. Flag register clear.
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set.
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue (CMD17)
   Set CMD17 Argument in SD_ARG and write H'0000_0011 to SD_CMD.
   Accordingly, CMD17 is issued, and the single block read operation is started.
4. Response check.
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP10.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP or the IOABT bit in SDIO_MODE to 1. Furthermore, this causes CMD12 and CMD52 to not be issued.
   If the command sequence is halted, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt.
5. Data receive from SD card and data read.
   Write H'0000_FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000_7E80 to SD_INFO2_MASK to enable the BRE interrupt. When the data receive from the SD card is completed, the BRE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BRE bit to 0 and read the amount of data specified by SD_SIZE from SD_BUF0.
   However, a communications error or timeout may be generated if data are being received while reading of SD_BUF0 is in progress.
6. Operation complete.
   When the data read from SD_BUF0 is completed, INFO2 (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO2 to 0 to end the single block read operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

33.4.3 Single Block Write

(1) Flowchart



Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.11 Single Block Write Flowchart Example

(2) Operation for Single Block Write

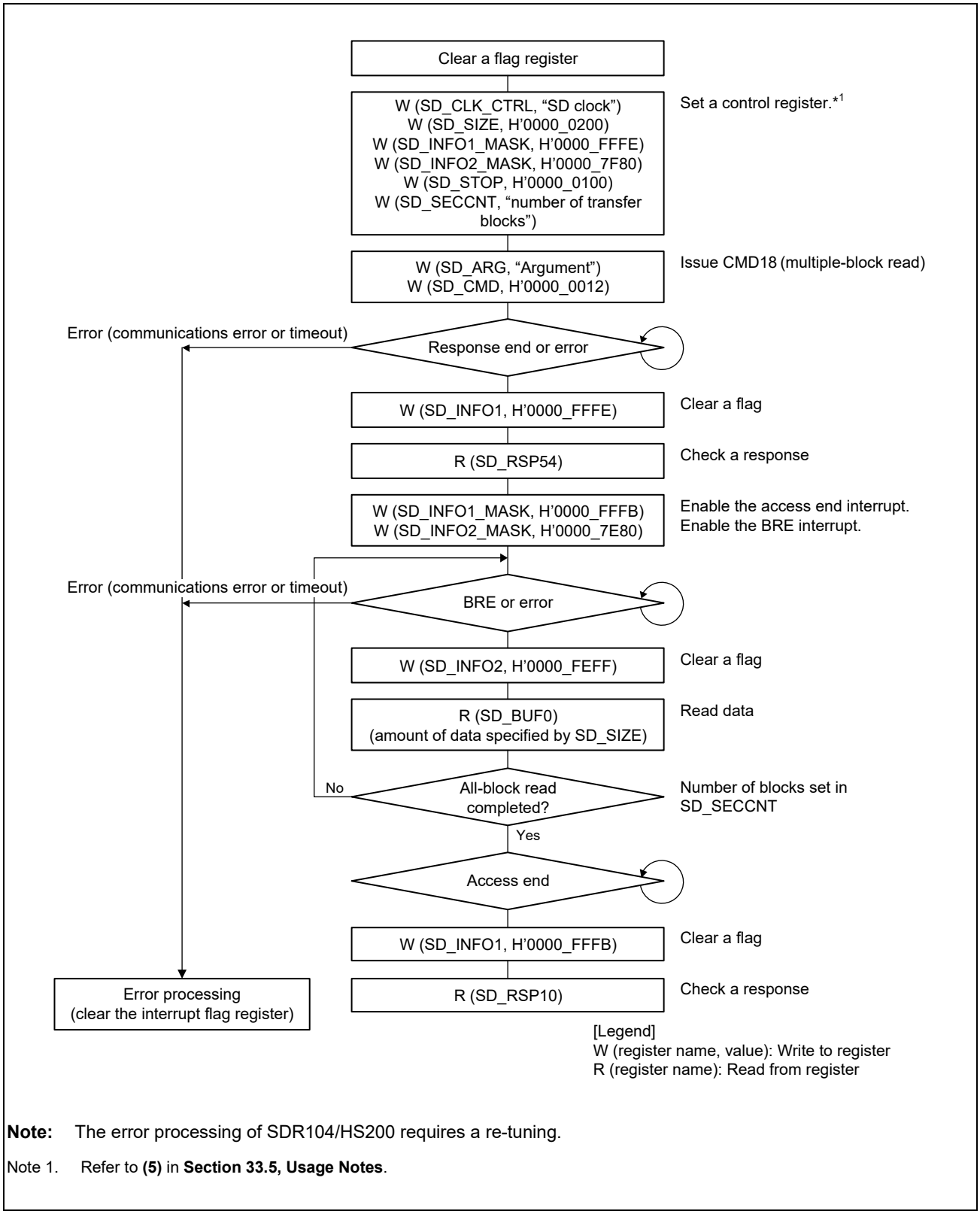
The operation of the single block write is described below.

1. Flag register clear.
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set.
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue (CMD24).
   Set CMD24 Argument in SD_ARG and write H'0000_0018 to SD_CMD.
   Accordingly, CMD24 is issued, and the single block write operation is started.
4. Response check.
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP10.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP or the IOABT bit in SDIO_MODE to 1. Furthermore, this causes CMD12 and CMD52 to not be issued.
   If the command sequence is halted, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt.
5. Data write and data transmit to SD card.
   Write H'0000_FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000_7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card. Then, the CRC status and busy state are received from the SD card.
   However, a communications error or timeout may be generated if data are being transmitted after writing to SD_BUF0.
6. Operation complete.
   When the CRC status and busy state are received from the SD card, INFO2 (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO2 bit to 0 to end the single block write operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

33.4.4 Multiple Block Read

(1) Flowchart



Note: The error processing of SDR104/HS200 requires a re-tuning.

Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.12 Multiple Block Read Flowchart Example

(2) Operation for Multiple Block Read

The operation of the multiple block read is described below.

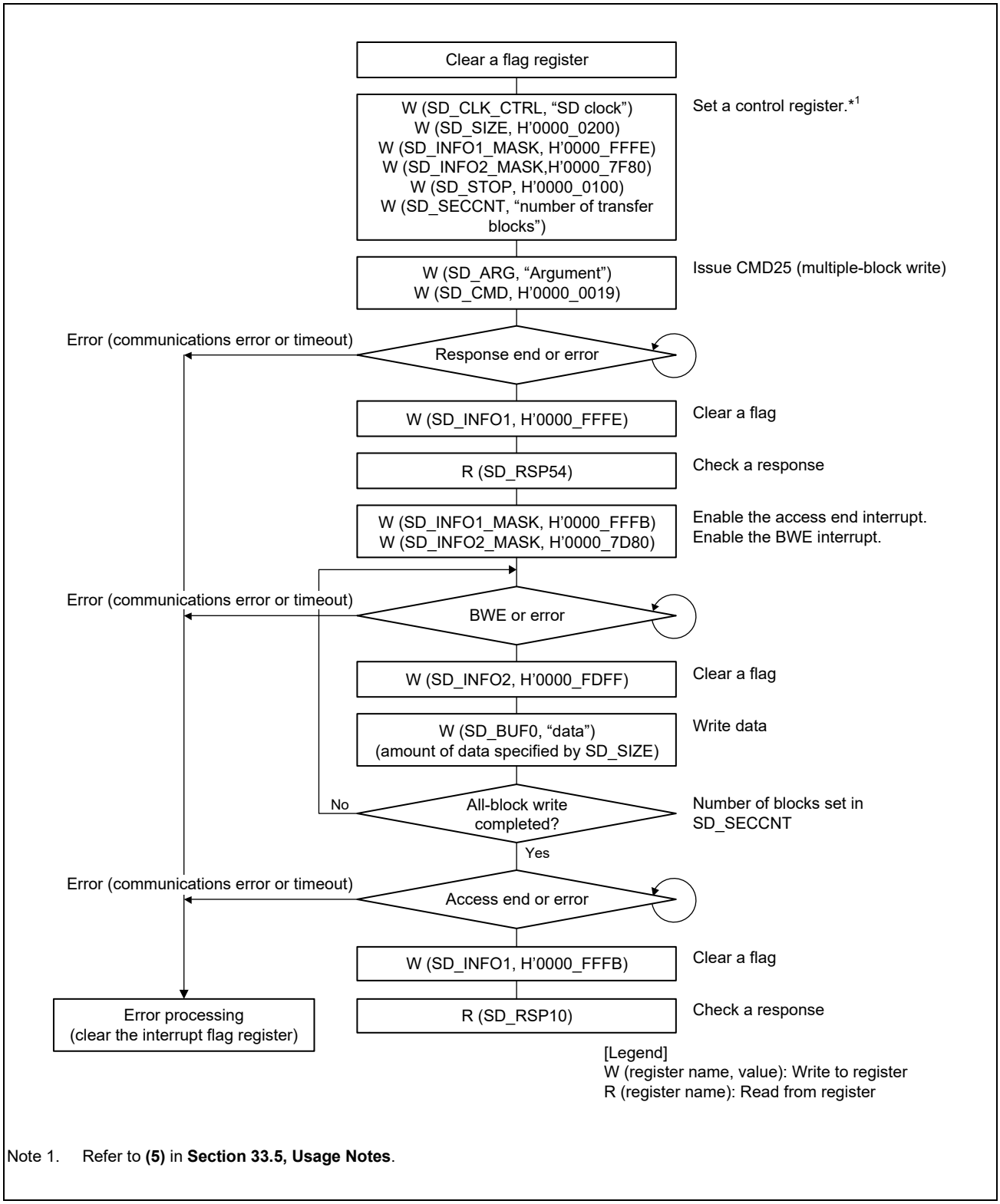
1. Flag register clear.
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set.
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD18)
   Set CMD18 Argument in SD_ARG and write H'0000_0012 to SD_CMD.
   Accordingly, CMD18 is issued, and the multiple block read operation is started.
4. Response check.
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data receive from SD card and data read.
   Write H'0000_FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000_7E80 to SD_INFO2_MASK to enable the BRE interrupt. When one-block data receive from the SD card is completed, the BRE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BRE bit to 0 and read the amount of data specified by SD_SIZE from SD_BUF0. Doing this repeats transfer of the number of blocks set in SD_SECCNT. However, a communications error or timeout may be generated if data are being received while reading of SD_BUF0 is in progress. CMD12 is automatically issued to stop multi-block transfer with the number of blocks which is set to SD_SECCNT and the response is received. At this point, CMD12 Argument is automatically set to H'0000_0000.
6. Operation complete.
   When all-block data read and the CMD12 response receive are completed, INFO2 (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO2 to 0 to read the response. This is the end of multiple block read operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

33.4.5 Multiple Block Write (when Using Internal Timer)

(1) Flowchart

Figure 33.13 shows the flowchart when using an internal timer.



Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.13 Multiple Block Write Flowchart Example (when Using Internal Timer)

(2) Operation for Multiple Block Write

The operation of the multiple block write is described below.

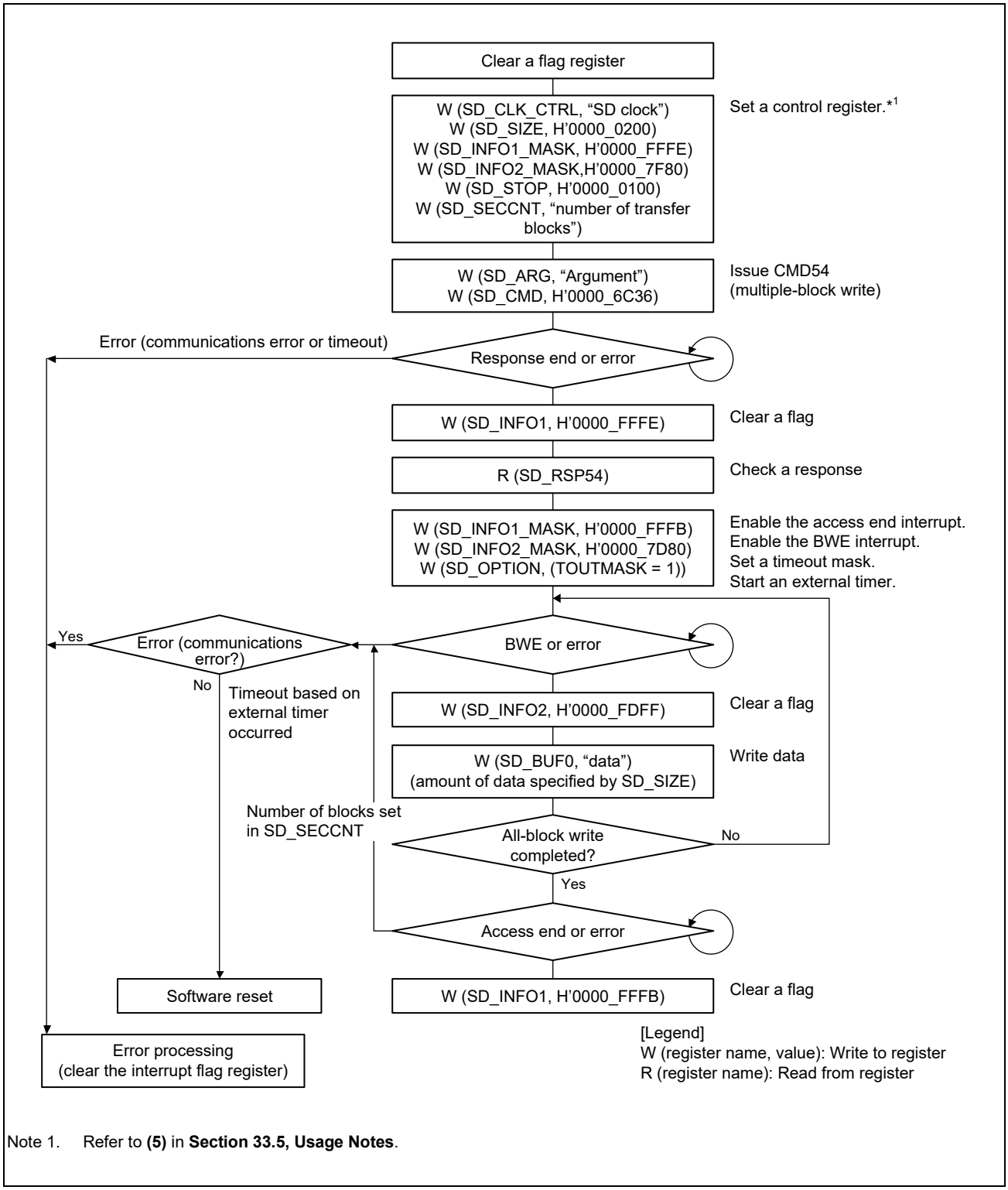
1. Flag register clear.
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set.
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD25).
   Set CMD25 Argument in SD_ARG and write H'0000 0019 to SD_CMD.
   Accordingly, CMD25 is issued, and the multiple block write operation is started.
4. Response check.
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) bit in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card.
   Write H'0000_FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000_7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card. Then, the CRC status and busy state are received from the SD card. Doing this repeats transfer of the number of blocks set in SD_SECCNT.
   However, a communications error or timeout may be generated if data are being transmitted while writing to SD_BUF0 is in progress. CMD12 is automatically issued to stop multi-block transfer with the number of blocks which is set to SD_SECCNT and the response is received. At this point, CMD12 Argument is automatically set to H'0000_0000.
6. Operation complete
   When all-block data transmit and the CRC status receive are completed, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO2 bit to 0 to read the response. This is the end of multiple block write operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

33.4.6 Multiple Block Write (when Using External Timer)

(1) Flowchart

The flowchart when using an external timer instead of an internal timer of this module is shown below.



Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.14 Multiple Block Write Flowchart Example (when Using External Timer)

(2) Operation for Multiple Block Write

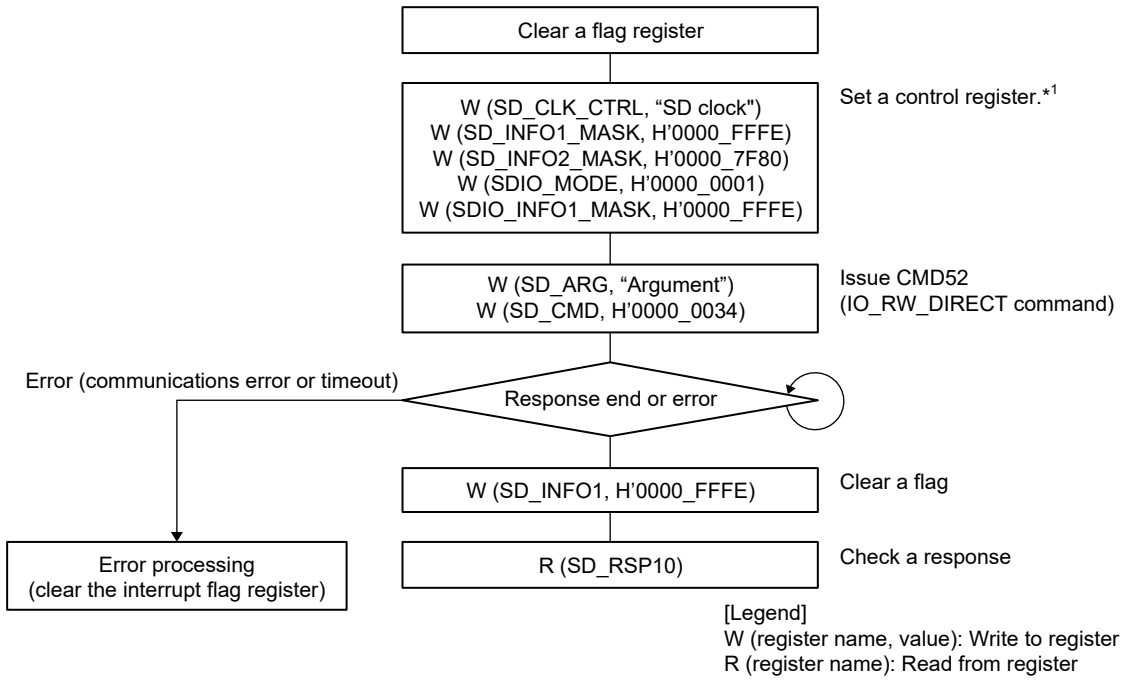
The operation of the multiple block write is described below.

1. Flag register clear.
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set.
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD54)
   Set CMD54 Argument in SD_ARG and write H'0000_6C36 to SD_CMD.
   Accordingly, CMD54 is issued, and the multiple block write operation is started.
4. Response check.
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) bit in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card.
   Write H'0000_FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000_7D80 to SD_INFO2_MASK to enable the BWE interrupt. Set the TOUTMASK bit in SD_OPTION to disable timeout and start an external timer.
   When SD_BUF0 is ready for the data to be written, the BWE bit in the SD_INFO2 resister is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card. Then, the CRC status and busy state are received from the SD card. Doing this repeats transfer of the number of blocks set in SD_SECCNT. However, a communications error may be generated if data are being transmitted while writing to SD_BUF0 is in progress.
6. Operation complete.
   When all-block data transmit and the CRC status receive are completed, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO2 bit to 0 to read the response. This is the end of multiple block write operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs at response reception (a communications error or timeout) or at data transmission. Perform a software reset if a timeout occurs at data transmission based on an external timer.

33.4.7 IO_RW_DIRECT Commend (CMD52)

(1) Flowchart



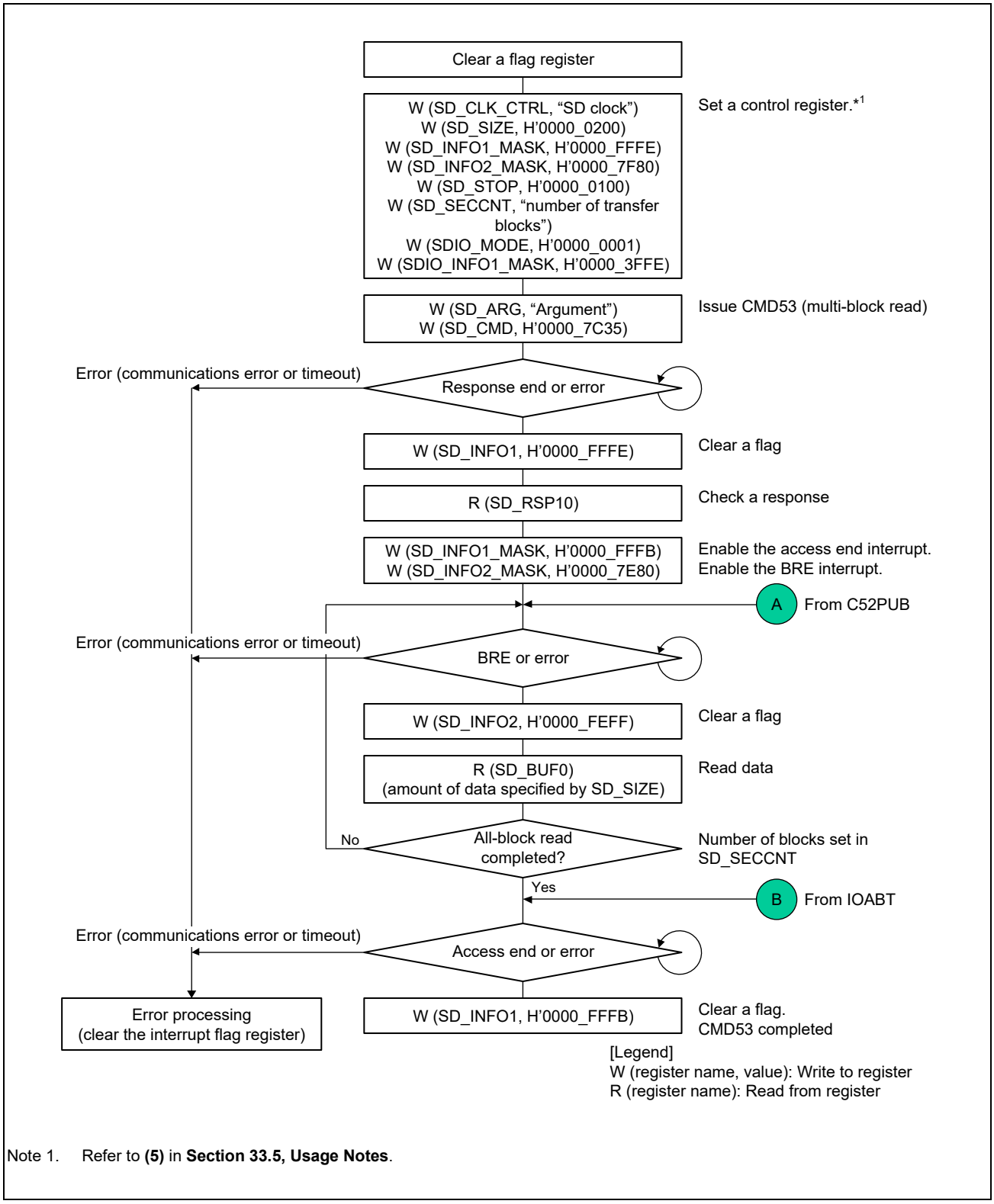
Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.15 IO_RW_DIRECT Command (CMD52) Flowchart Example

33.4.8 IO_RW_EXTENDED (CMD53/Multiple Block Read)

(1) Flowchart

Figure 33.16 shows a flowchart example for CMD53 (multiple block read).



Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.16 CMD53 (Multiple Block Read) Flowchart Example

Figure 33.17 shows a flowchart example when CMD52 (SDIO abort) is issued at CMD53 (multiple block read).

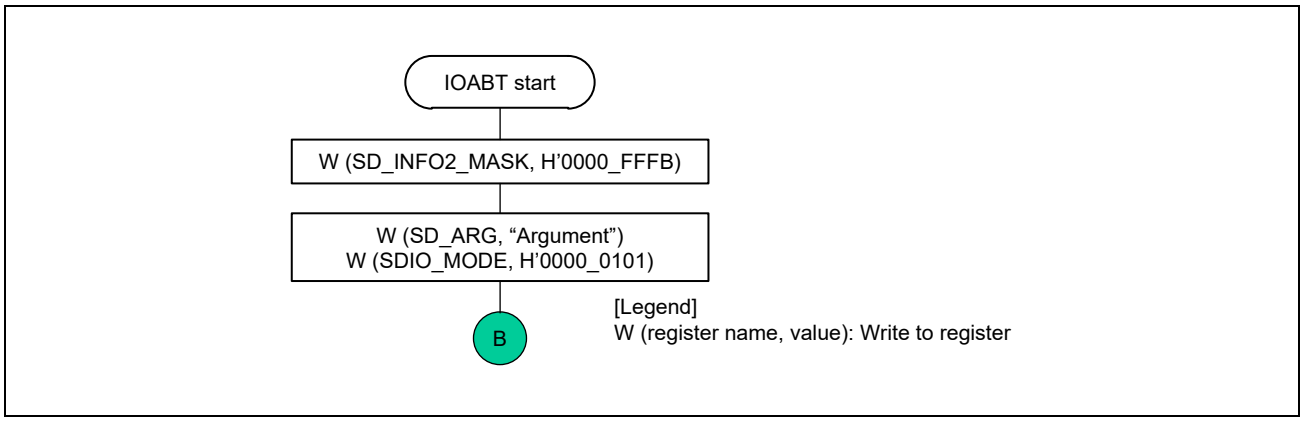


Figure 33.17 Flowchart Example when CMD52 (SDIO Abort) is Issued at CMD53 (Multiple Block Read)

Figure 33.18 shows a flowchart example when CMD52 (SDIO none abort) is issued at CMD53 (multiple block read) while the SD/MMC host interface is in the read wait state.

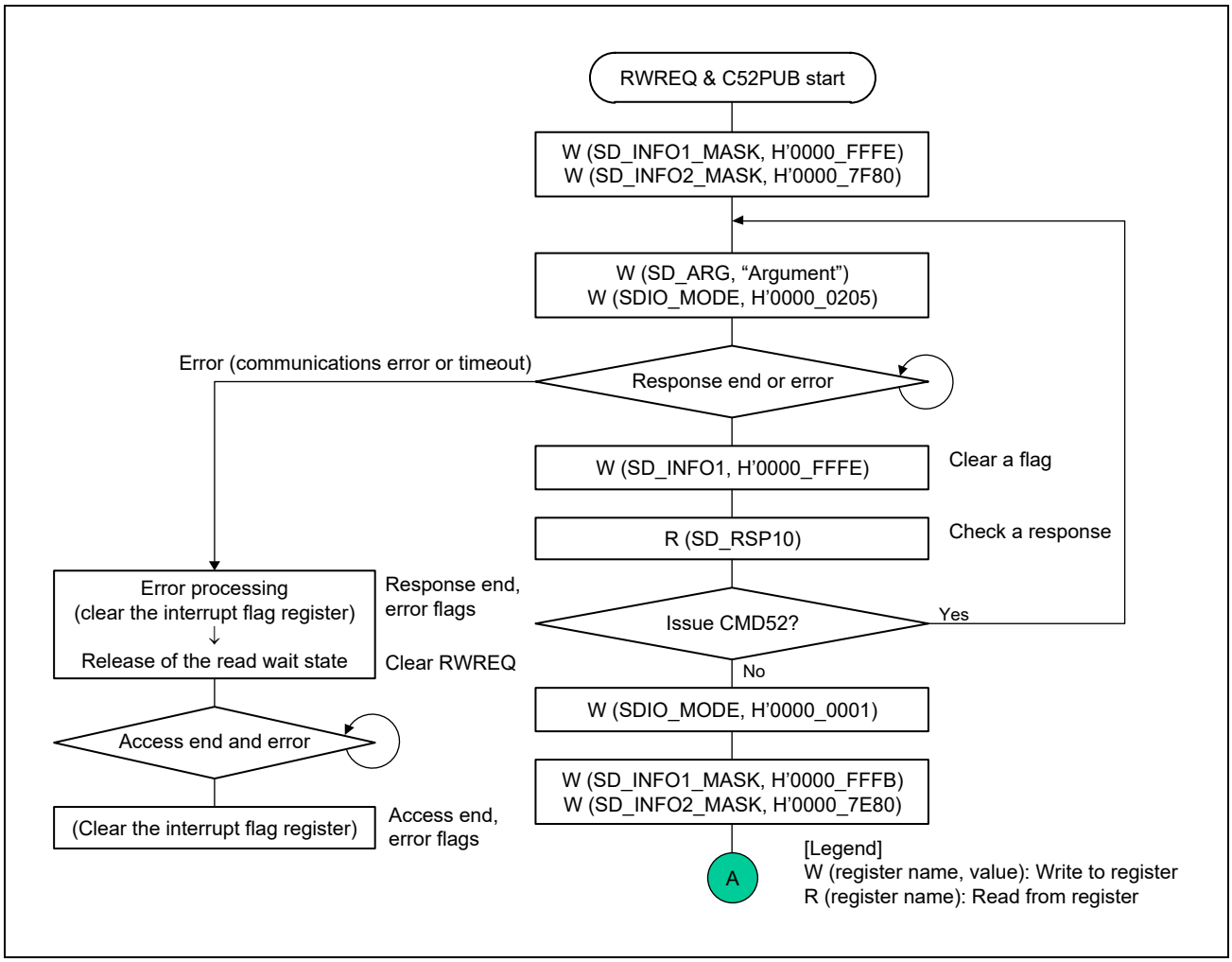
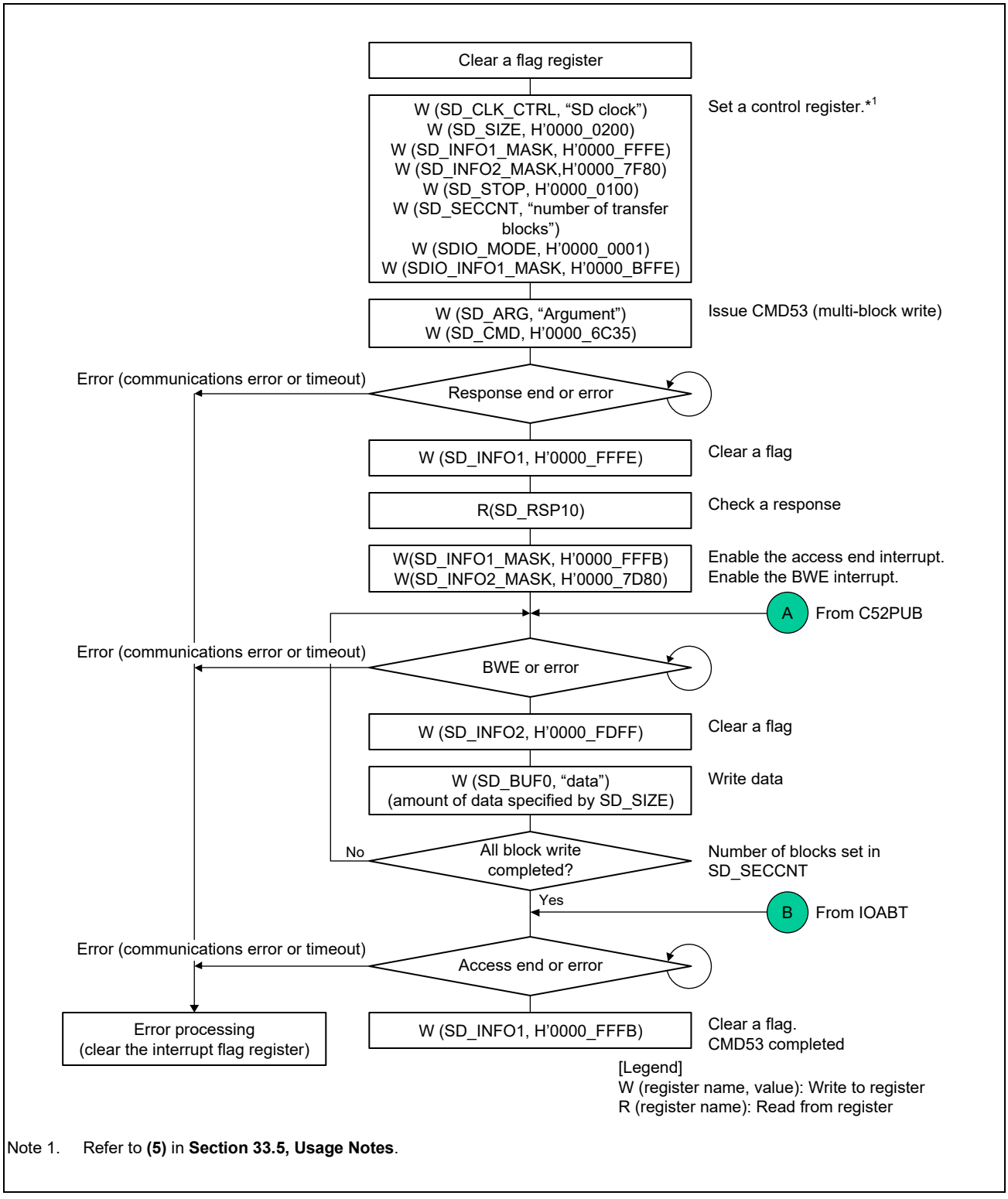


Figure 33.18 Flowchart Example when CMD52 (SDIO None Abort) is Issued after Read Wait State is Entered at CMD53 (Multi Block Read)

33.4.9 IO_RW_EXTENDED (CMD53/Multiple Block Write)

(1) Flowchart

Figure 33.19 shows a flowchart example for CMD53 (multiple block write).



Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.19 CMD53 (Multiple Block Write) Flowchart Example

Figure 33.20 shows a flowchart example when CMD52 (SDIO abort) is issued at CMD53 (multiple block write).

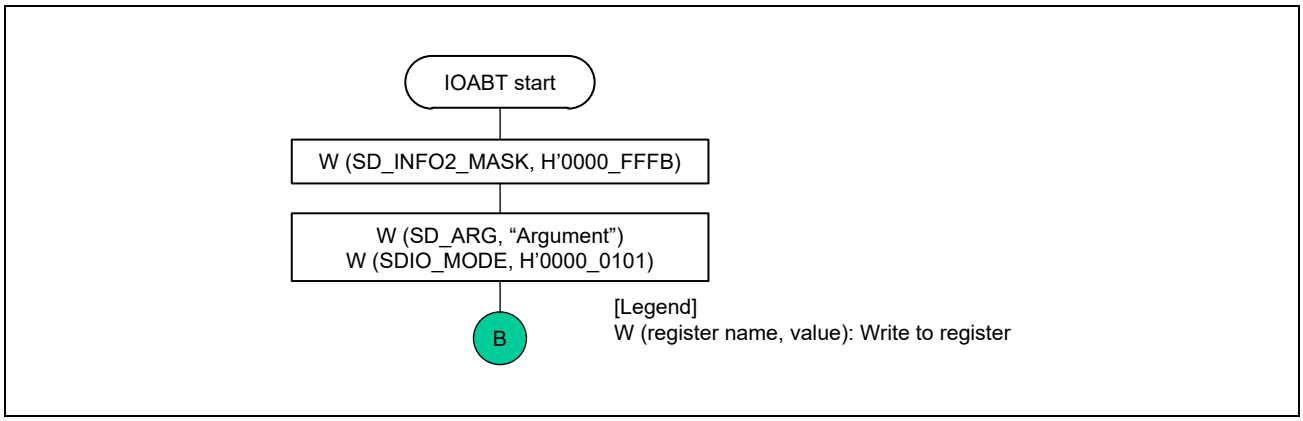


Figure 33.20 Flowchart Example when CMD52 (SDIO Abort) is Issued at CMD53 (Multiple Block Write)

Figure 33.21 shows a flowchart example when CMD52 (SDIO none abort) is issued at CMD53 (multiple block write).

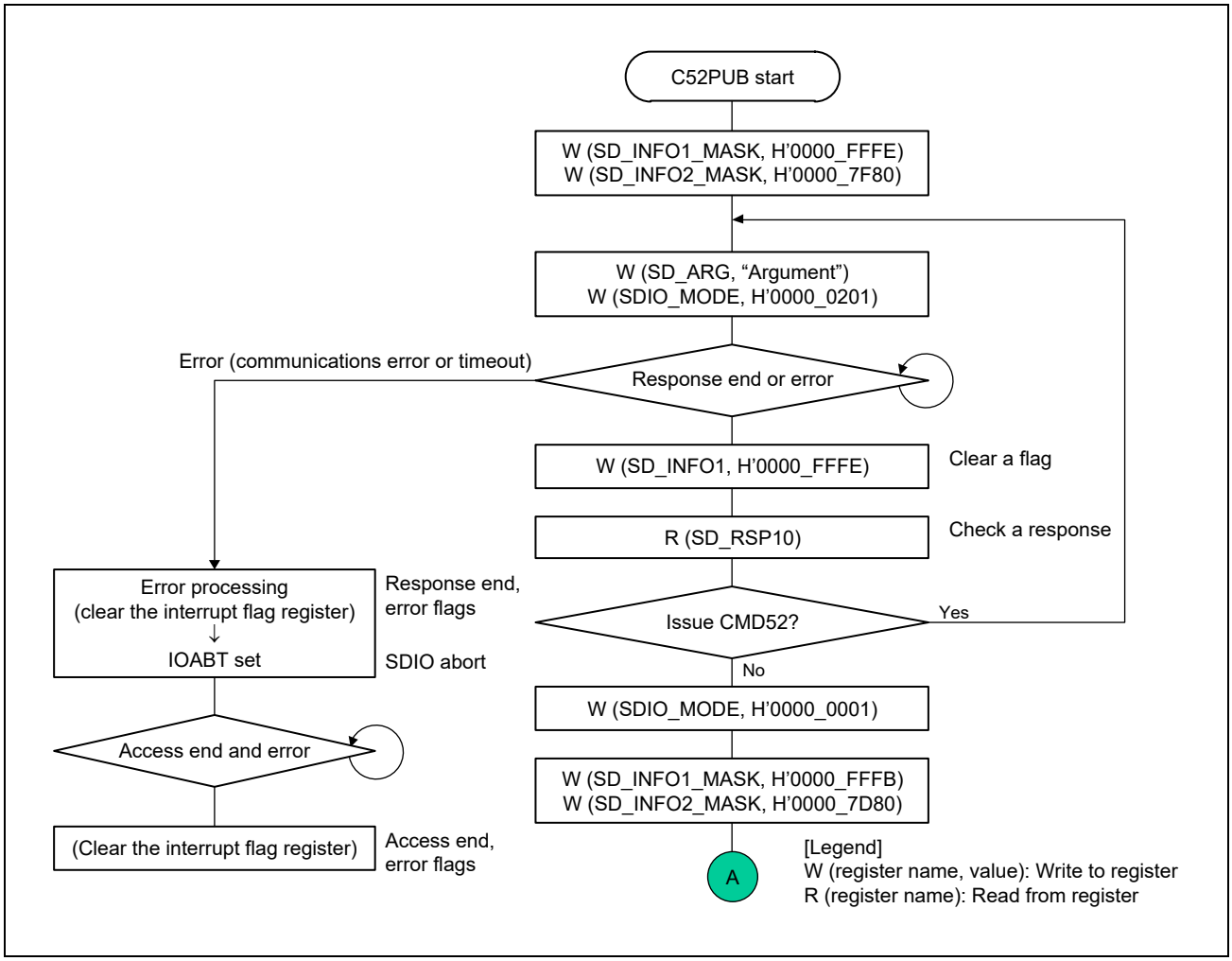
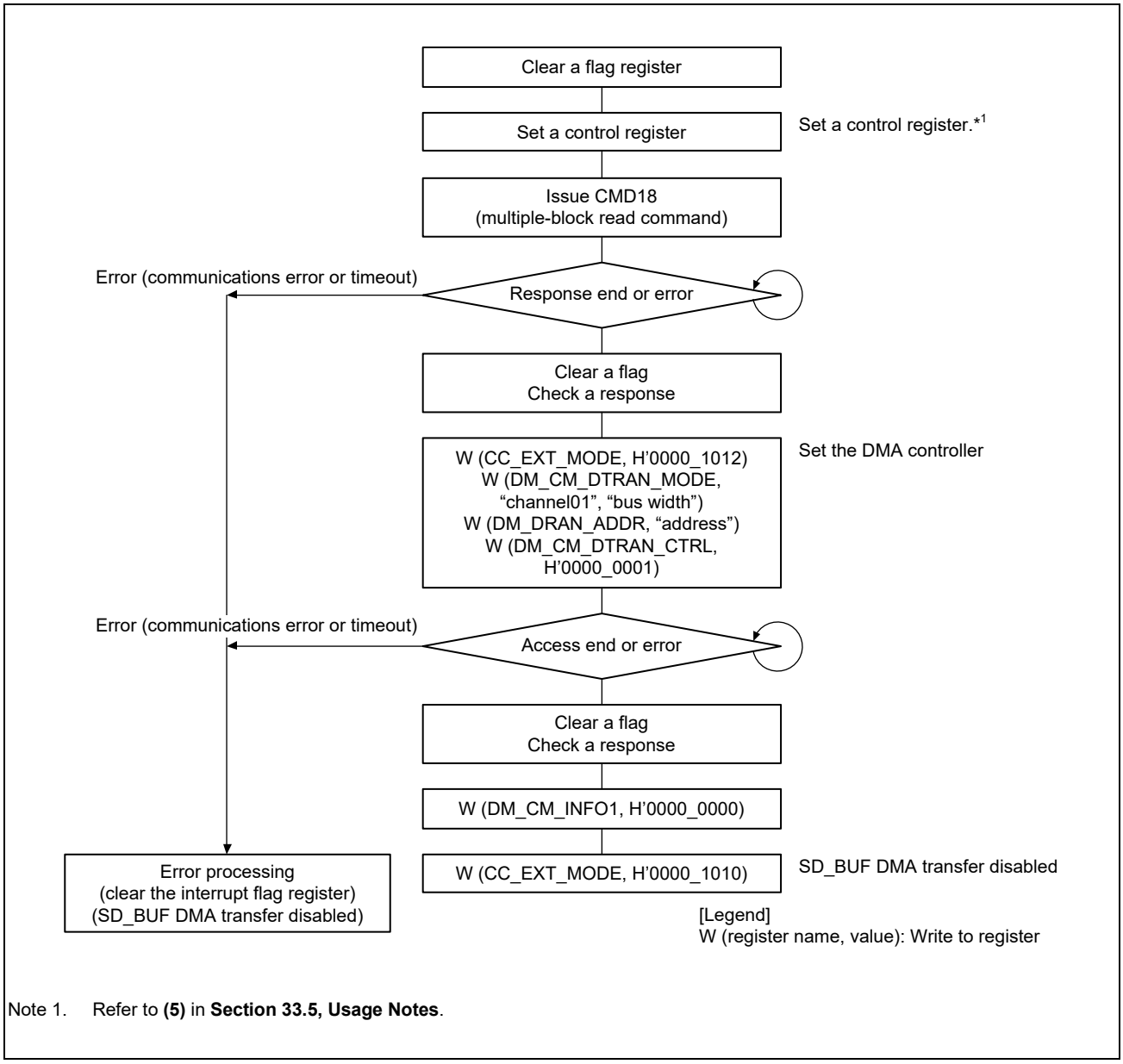


Figure 33.21 Flowchart Example when CMD52 (SDIO None Abort) is Issued at CMD53 (Multiple Block Write)

33.4.10 DMA Transfer

(1) SD_BUF DMA Transfer

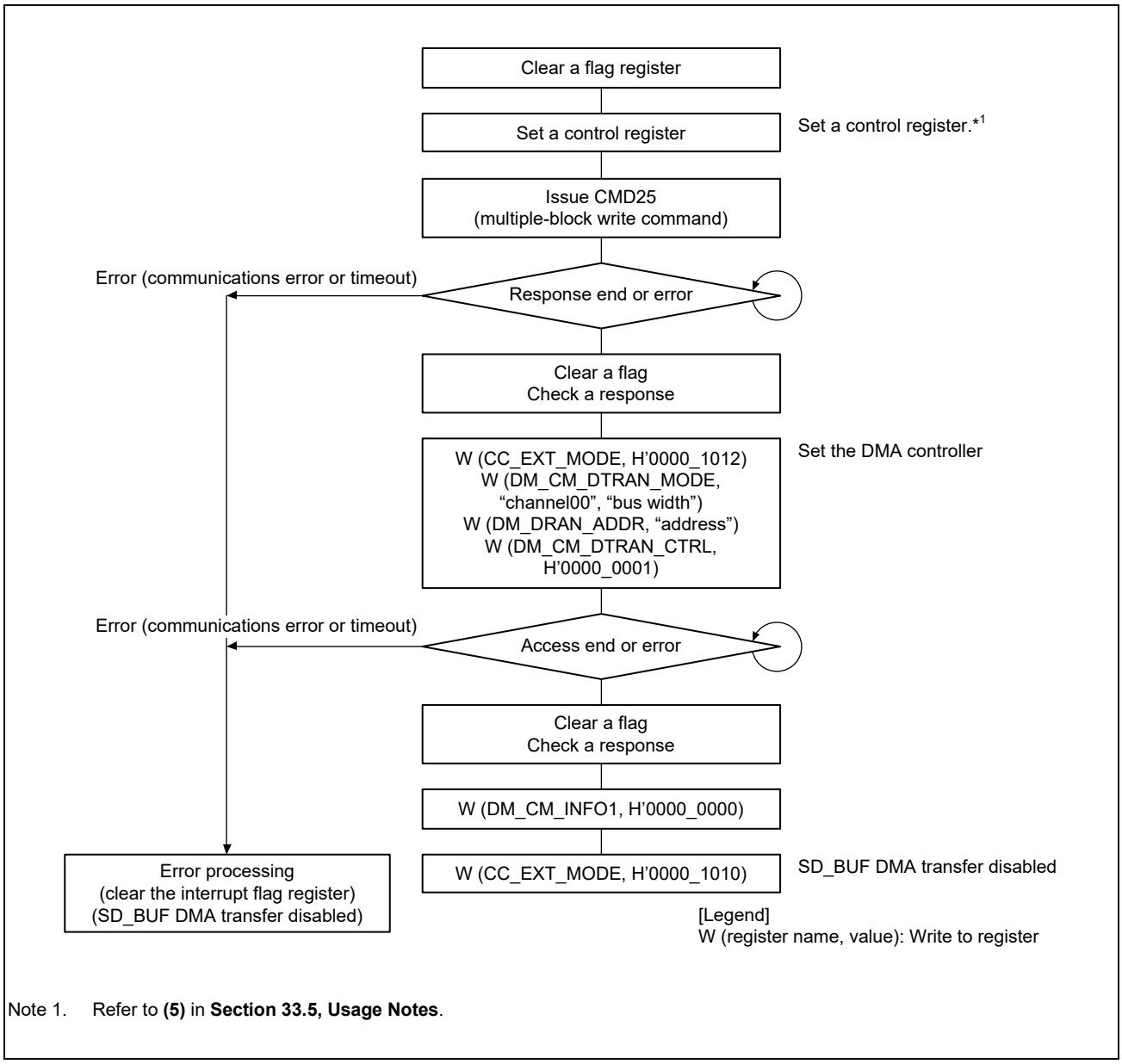
Figure 33.22 shows a flowchart example for SD_BUF DMA read when CMD18 (multiple block read) is issued.



Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.22 SD_BUF DMA Read Flowchart Example

Figure 33.23 shows a flowchart example for SD_BUF DMA write when CMD25 (multiple block write) is issued.

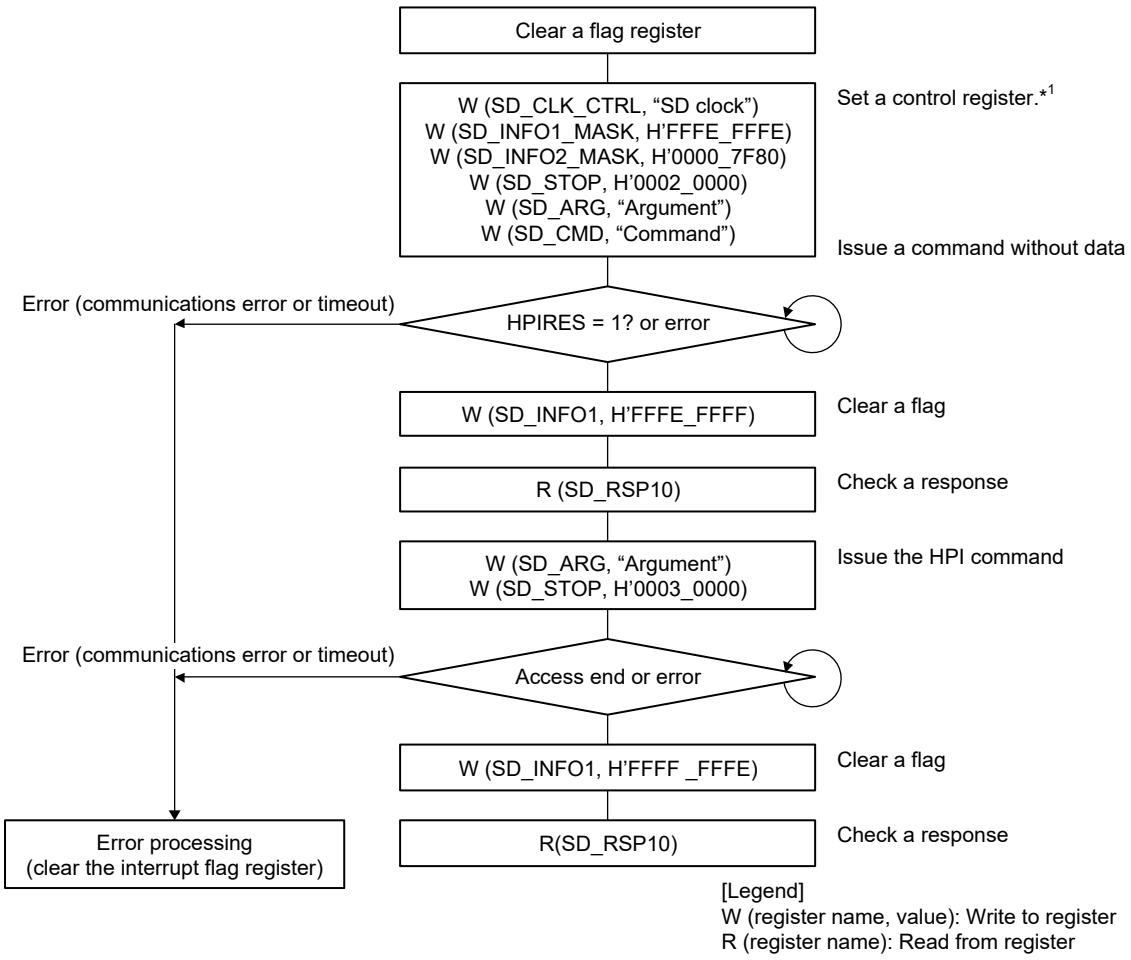


Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.23 SD_BUF DMA Write Flowchart Example

33.4.11 High-Priority Interrupt (without Data Transfer)

(1) Flowchart



Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.24 Example of the High-Priority Interrupt (without Data Transfer) Flowchart

(2) Operation for High-Priority Interrupt without Data Transfer

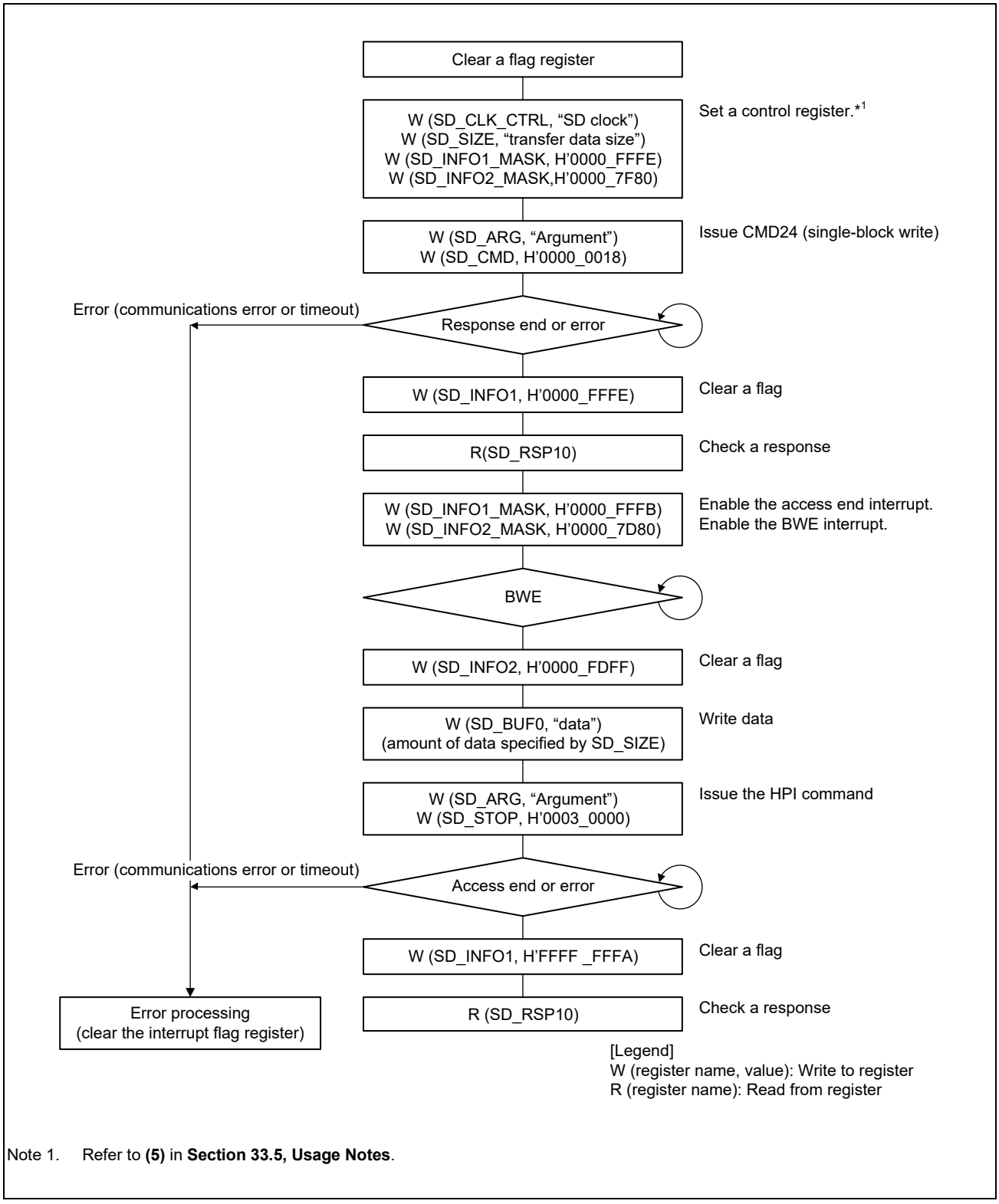
The operation of the high-priority interrupt (HPI) without data transfer is described below.

1. Flag register clear.
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set.
   Set the SD clock (SDCLK), HPI enable, interrupt mask, and so on. (SD_CLK_CTRL, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue.
   Set the CMD Argument in SD_ARG and write to SD_CMD.
   Accordingly, CMD is issued, and the operation is started.
4. Flag clear.
   On receiving the response, the HPIRES bit in SD_INFO1 is set to 1 to generate an interrupt. Clear the HPIRES bit to 0.
5. Read the response from SD_RSP10.
6. HPI command issue.
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
7. Operation complete.
   When reception of the response to the HPI command is completed and the busy state is released, the INFO0 bit in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO0 bit to 0 to read the response from SD_RSP10.
   This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

33.4.12 High-Priority Interrupt (at Single Block Write)

(1) Flowchart



Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.25 Example of the High-Priority Interrupt (at Single Block Write) Flowchart

(2) Operation for HPI at Single Block Write

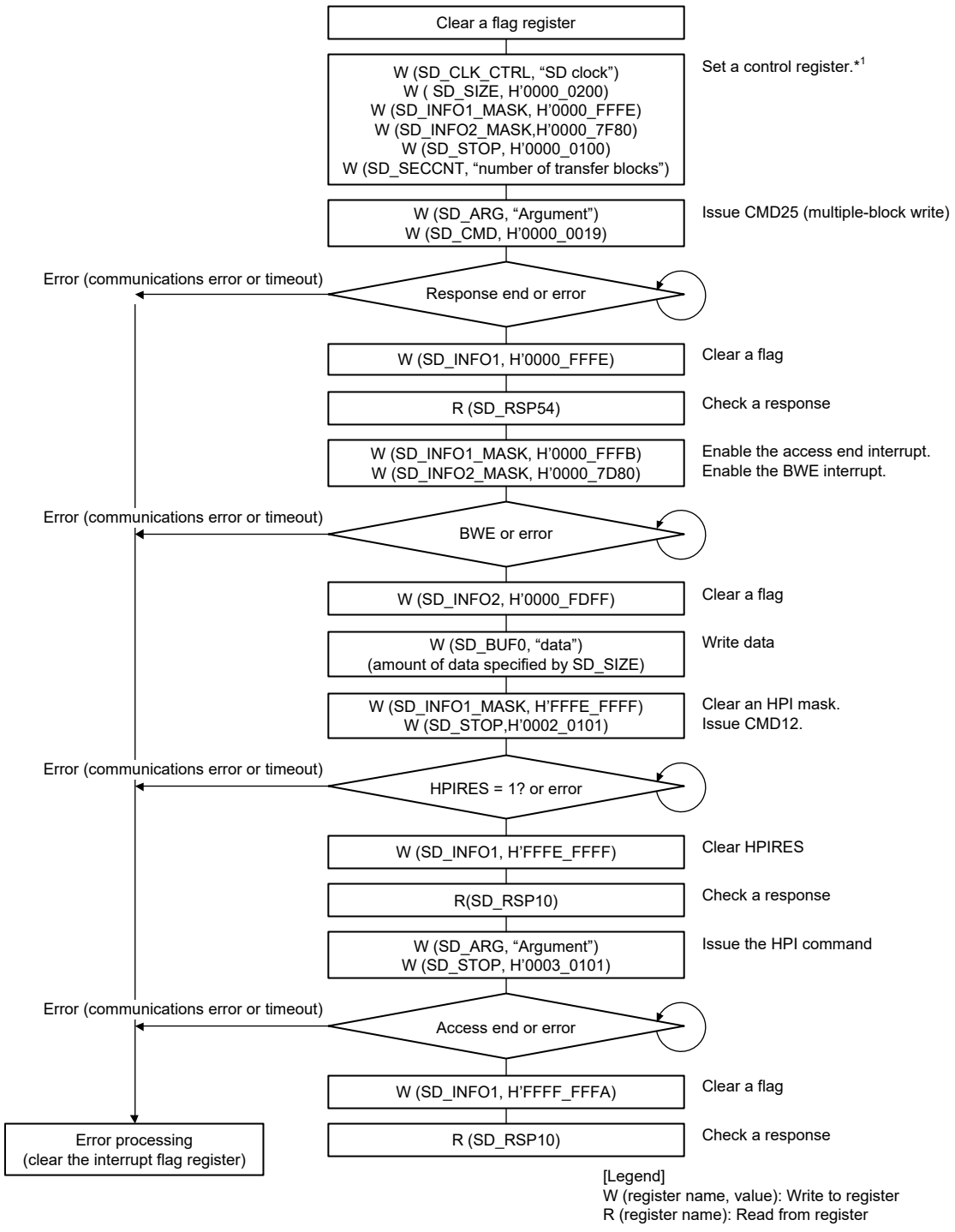
The operation of HPI at single block write is described below.

1. Flag register clear.
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set.
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue (CMD24)
   Set CMD24 Argument in SD_ARG and write H'0000_0018 to SD_CMD.
   Accordingly, CMD24 is issued, and the single block write operation is started.
4. Response check.
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP10.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP or the IOABT bit in SDIO_MODE to 1. Furthermore, this causes CMD12 and CMD52 to not be issued.
   If the command sequence is halted, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt.
5. Data write and data transmit to SD card.
   Write H'0000_FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000_7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card.
6. HPI command issue.
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
7. Operation complete.
   When reception of the response to the HPI command is completed and the busy state is released, the INFO2 and INFO0 bits in SD_INFO1 are set to 1 to generate an interrupt. Clear the INFO2 and INFO0 bits to 0 to read the response from SD_RSP10. This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

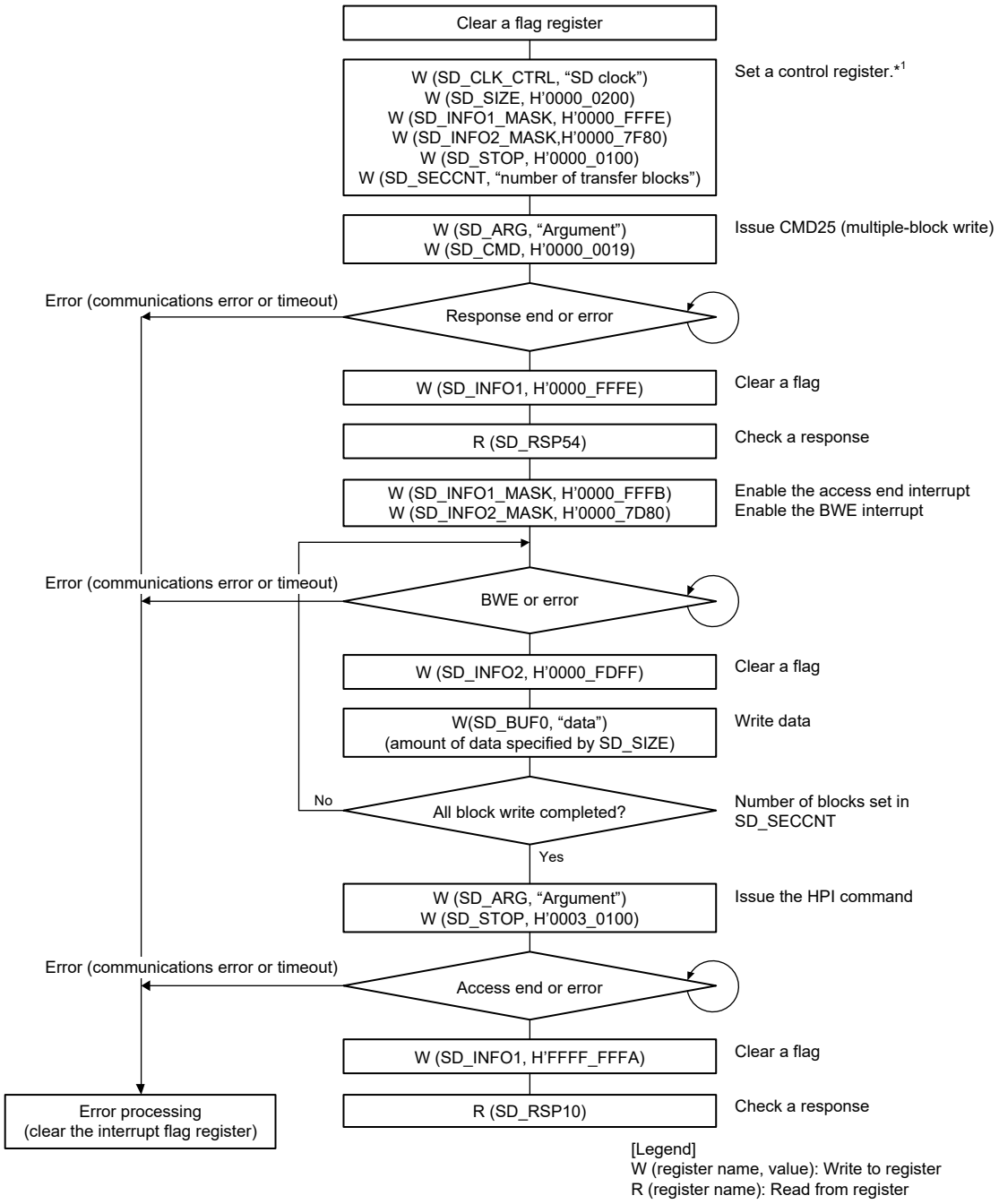
33.4.13 High-Priority Interrupt (at Multiple Block Write)

(1) Flowchart



Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.26 Example of the High-Priority Interrupt (at Multiple Block Write) Flowchart (a)



Note 1. Refer to **(5)** in **Section 33.5, Usage Notes**.

Figure 33.27 Example of the High-Priority Interrupt (at Multiple Block Write) Flowchart (b)

(2) Operation for HPI at Multiple Block Transfer

The operation of HPI at the multiple block write is described below.

(a) When not all the data has been written to SD_BUF

1. Flag register clear.
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set.
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD25).
   Set CMD25 Argument in SD_ARG and write H'0000_0019 to SD_CMD.
   Accordingly, CMD25 is issued, and the multiple block write operation is started.
4. Response check.
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) bit in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card.
   Write H'0000_FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000_7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card. Then, the CRC status and busy state are received from the SD card.
   However, a communications error or timeout may be generated if data are being transmitted while writing to SD_BUF0 is in progress.
6. Command issue (CMD12).
   Write H'FFFE_FFFF to SD_INFO1_MASK to enable the HPIRES interrupt. Write H'0002_0101 to SD_STOP, which causes CMD12 to be issued.
7. Response check.
   When the response is received, the HPIRES bit in SD_INFO1 is set to 1 to generate an interrupt. Clear the HPIRES bit to 0 to read the response from SD_RSP10.
8. HPI command issue (CMD25).
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
9. Operation complete.
   When reception of the response to the HPI command is completed and the busy state is released, the INFO2 and INFO0 bits in SD_INFO1 are set to 1 to generate an interrupt. Clear the INFO2 and INFO0 bits to 0 to read the response from SD_RSP10. This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

(b) When all the data has been written to SD_BUF

1. Flag register clear.
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set.
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD25).
   Set CMD25 Argument in SD_ARG and write H'0000_0019 to SD_CMD.
   Accordingly, CMD25 is issued, and the multiple block write operation is started.
4. Response check.
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) bit in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card.
   Write H'0000_FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000_7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card.
   Then, the CRC status and busy state are received from the SD card. Doing this repeats transfer of the number of blocks set in SD_SECCNT.
   However, a communications error or timeout may be generated if data are being transmitted while writing to SD_BUF0 is in progress.
6. HPI command issue.
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
7. Operation complete.
   When reception of the response to the HPI command is completed and the busy state is released, the INFO2 and INFO0 bits in SD_INFO1 are set to 1 to generate an interrupt. Clear the INFO2 and INFO0 bits to 0 to read the response from SD_RSP10. This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

33.4.14 Example of SD_CMD Register Setting

Table 33.8 lists the example of SD_CMD (SD interface) register setting.

Table 33.8 Example of SD_CMD Register Setting (SD) (1/2)

Type	Command	Example of SD_CMD Register Setting	Remark
CMD	CMD0	H'0000_0000	
	CMD2	H'0000_0002	
	CMD3	H'0000_0003	
	CMD4	H'0000_0004	
	CMD5	H'0000_0705 or H'0000_0005	
	CMD6	H'0000_1C06 or H'0000_0006	
	CMD7	H'0000_0007	When the card is placed in the deselected state, the response timeout flag will be set since there is no response.
	CMD8	H'0000_0408 or H'0000_0008	
	CMD9	H'0000_0009	
	CMD10	H'0000_000A	
	CMD11	H'0000_040B or H'0000_000B	
	CMD12	H'0000_000C	
	CMD13	H'0000_000D	
	CMD15	H'0000_000F	
	CMD16	H'0000_0010	
	CMD17	H'0000_0011	
	CMD18	H'0000_0012	With auto CMD12 enabled (other than in SDR104 mode)
		H'0000_7C12	With auto CMD12 disabled (only in SDR104 mode)
	CMD19	H'0000_1C13 or H'0000_0013	
	CMD20	H'0000_0514 or H'0000_0014	
	CMD23	H'0000_0417 or H'0000_0017	
	CMD24	H'0000_0018	
	CMD25	H'0000_0019	With auto CMD12 enabled (other than in SDR104 mode)
		H'0000_6C19	With auto CMD12 disabled (only in SDR104 mode)
	CMD27	H'0000_001B	
	CMD28	H'0000_001C	
	CMD29	H'0000_001D	
	CMD30	H'0000_001E	
	CMD32	H'0000_0020	
	CMD33	H'0000_0021	
	CMD38	H'0000_0026	
	CMD42	H'0000_002A	
	CMD48	H'0000_1C30	
	CMD49	H'0000_0C31	
	CMD52	H'0000_0434 or H'0000_0034	

Table 33.8 Example of SD_CMD Register Setting (SD) (2/2)

Type	Command	Example of SD_CMD Register Setting	Remark
CMD	CMD53	H'0000_1C35	Single read
		H'0000_0C35	Single write
		H'0000_7C35	Multiple read
		H'0000_6C35	Multiple write
		H'0000_0035	The value on the left can be set irrespective of whether single or multi. However, the CF39 bit in SD_ARG must be set as follows. Read: 0 Write: 1
	CMD55	H'0000_0037	
	CMD56	H'0000_0038	
	CMD58	H'0000_7C3A	
	CMD59	H'0000_6C3B	
ACMD	ACMD6	H'0000_0046	
	ACMD13	H'0000_004D	
	ACMD22	H'0000_0056	
	ACMD23	H'0000_0057	
	ACMD41	H'0000_0069	
	ACMD42	H'0000_006A	
	ACMD51	H'0000_0073	

Table 33.9 lists the example of SD_CMD (MMC interface) register setting.

Table 33.9 Example of SD_CMD Register Setting (MMC)

Type	Command	Example of SD_CMD Register Setting	Remark
CMD	CMD0	H'0000_0000	
	CMD1	H'0000_0701	
	CMD2	H'0000_0002	
	CMD3	H'0000_0003	
	CMD4	H'0000_0004	
	CMD5	H'0000_0505	
	CMD6	H'0000_0506	In the response busy state
		H'0000_0406	Not in the response busy state
	CMD7	H'0000_0007	When the card is placed in the deselected state, the response timeout flag will be set since there is no response.
	CMD8	H'0000_1C08	
	CMD9	H'0000_0009	
	CMD10	H'0000_000A	
	CMD12	H'0000_000C	
	CMD13	H'0000_000D	
	CMD14	H'0000_1C0E	SDIF_MODE must be set to H'0100 (with CRC16 disabled).
	CMD15	H'0000_000F	
	CMD16	H'0000_0010	
	CMD17	H'0000_0011	
	CMD18	H'0000_7C12	Pre-defined
	CMD19	H'0000_0C13	SDIF_MODE must be set to H'0100 (with CRC16 disabled).
	CMD21	H'0000_1C15	
	CMD23	H'0000_0017	
	CMD24	H'0000_0018	
	CMD25	H'0000_6C19	Pre-defined
	CMD26	H'0000_0C1A	
	CMD27	H'0000_001B	
	CMD28	H'0000_001C	
	CMD29	H'0000_001D	
	CMD30	H'0000_001E	
	CMD31	H'0000_1C1F	
	CMD35	H'0000_0423	
	CMD36	H'0000_0424	
	CMD38	H'0000_0026	
	CMD39	H'0000_0427	
	CMD40	H'0000_0428	
	CMD42	H'0000_002A	
	CMD49	H'0000_0C31	
	CMD53	H'0000_7C35	
	CMD54	H'0000_6C36	
	CMD55	H'0000_0037	
	CMD56	H'0000_0038	

33.5 Usage Notes

(1) SD_BUF Illegal Write Access

When writing data to SD_BUF0 after the single block write or multi block write command is issued, the data of the size specified by SD_SIZE must be written to.

If the data of the size which exceeds the size specified by SD_SIZE is written to, the ERR4 bit in SD_INFO2 is set to 1. In addition, the data written to SD_BUF0 may not be transmitted and it causes the SCLKDIVEN bit in SD_INFO2 to hold the value of 0. In such cases, clearing the SDRST bit in SOFT_RST to 0 and then restoring its value to 1 clears the SCLKDIVEN bit to 1.

However, for the single byte (in the case of 16- or 32-bit access) or three bytes (in the case of 32-bit access) when the number of bytes setting in SD_SIZE is odd, or the fraction of bytes when the number of bytes setting in SD_SIZE is even (in the case of 32-bit access), since the portion of dummy data writing is regarded as excess data and ignored, it is not within the scope of the above description (the fraction of bytes: the two bytes that are not in a four-byte unit).

(2) Block Number Limitation for Multiple Block Read

When performing a multiple block read of one or two blocks, depending on the timing with which the response register is read, the response value may not be read properly. This must be avoided by either of the following countermeasures.

1) When receiving one or two blocks of data, use single block reading.
2) Read the response to CMD18 from SD_RSP54.

<Mechanism of incorrect reading>

Figure 33.28 shows the processing flows of SD/MMC host interface (hardware) operation and software operation when a multiple block read is performed on two blocks. As shown in the incorrect operation of **Figure 33.28**, when an interrupt is generated on reception of the CMD18 response and the timing with which the SD card response register (SD_RSP10) is read by the interrupt is delayed, the data during the CMD12 response reception or the CMD12 response may be read. In the case of a multiple block read of three or more blocks, CMD12 is not issued until the block of data has been read, so this problem does not arise. Furthermore, in the case of a multiple block write, since the CMD25 response is read before the block of data is sent, the problem does not arise.

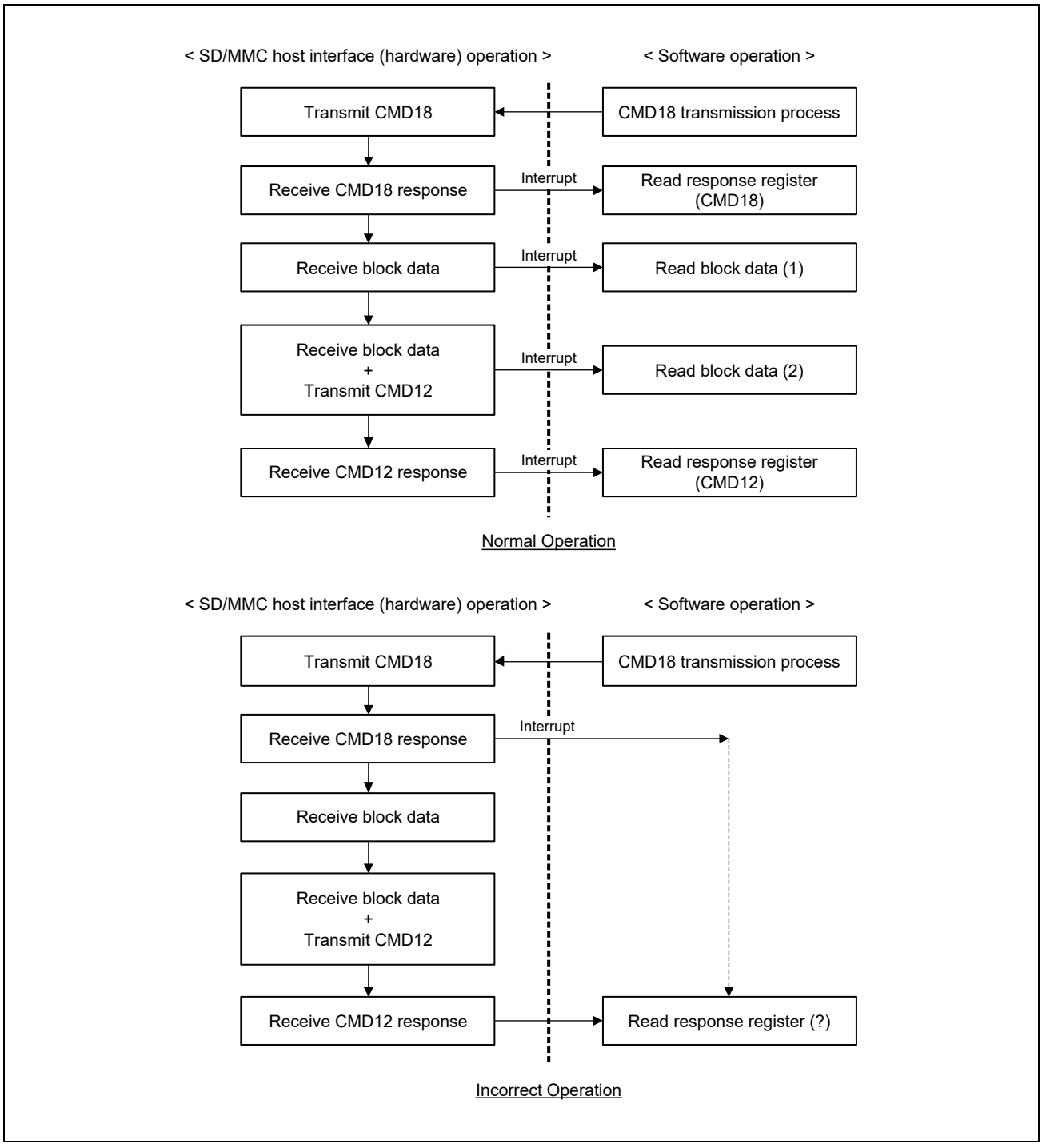


Figure 33.28 Flowcharts for Multiple Block Read Operation (Two Blocks)

(3) Automatic Control of SDCLK Output

In the SD Card standard, 74 cycles of SDCLK must be output before initialization of the card. For this reason, use automatic control of SDCLK output after 74 cycles of SDCLK have been output. Furthermore, if automatic control of SDCLK output was in use, SDCLK output is stopped on completion of the sequence for a communications error or timeout. Thus, in cases where state transitions within the SD card are necessary and so on after completion of the sequence, release automatic control of SDCLK output and restart supply of SDCLK to the SD card.

(4) Control of the C52PUB Setting for Multiple Block Write

If the C52PUB bit in SDIO_MODE is set to 1 during a sequence of multiple block write due to CMD53, CMD52 is not issued until SD_BUF becomes empty. For this reason, set the C52PUB bit after suspending writing to SD_BUF by following the appropriate procedure below.

- When DMA transfer is not in use

1. Before setting the C52PUB bit, suspend writing to SD_BUF by making the setting in SD_INFO2 to disable BWE interrupts.
2. Set the C52PUB bit in SDIO_MODE to 1 (so that CMD52 is issued when SD_BUF becomes empty).
3. After the INFO0 interrupt processing in SD_INFO1 due to the issuing of CMD52 has been completed, restart writing to SD_BUF by making the setting in SD_INFO2 to enable BWE interrupts.

- When DMA transfer is in use

1. Every time DMA transfer of the value set in SD_SIZE × n blocks (where n = 1, 2, ...) proceeds, suspend writing to SD_BUF by DMA transfer before the C52PUB bit is set.
2. Set the C52PUB bit in SDIO_MODE to 1 (so that CMD52 is issued when SD_BUF becomes empty).
3. After the INFO0 interrupt processing in SD_INFO1 due to the issuing of CMD52 has been completed, restart writing to SD_BUF by DMA transfer.

(5) Notes on SD_CLK_CTRL Register Settings

When the SCLKDIVEN bit in SD_INFO2 is 0, SD_CLK_CTRL cannot be written to. Before writing to SD_CLK_CTRL, be sure to check that the SCLKDIVEN bit in SD_INFO2 is 1.

(6) Restrictions on specifications

1. The SDIO suspend/resume is not supported.
2. The SPI bus is not supported.
3. The shared bus and 8-bit SD bus for embedded SDIO are not supported.
4. The stream transfer for MMC cards is not supported.

(7) STP bit setting during multiple block read

During execution of multiple block read with automatic CMD12 execution by setting the SEC bit in SD_STOP to 1, even if the STP bit in SD_STOP is set to 1 to forcibly stop the execution, the command sequence may not stop depending on the timing of setting the STP bit.

To avoid this, when setting the STP bit in SD_STOP to 1 during multiple block transfer, clear the SEC bit in SD_STOP to 0 at the same time. (Even when the SCLKDIVEN bit in SD_INFO2 is 0, change the SEC bit from 1 to 0.)

When the command sequence has not stopped because the SEC bit was not cleared to 0, the command sequence can be stopped by clearing the SDRST bit in SOFT_RST to 0.

When forcibly terminating the CMD53 multiple block transfer through the IOABT bit in SDIO_MODE, be sure to leave the SEC bit in SD_STOP as 1.

33.6 Sampling Clock Controller (SCC)

33.6.1 Features

This module controls a sampling clock (hereafter referred to as the SCC sampling clock) that is used for SD UHS-I/SDR104 and MMC HS200. When this module is used with the LSI, SD UHS-I/SDR104 and MMC HS200 can be supported.

33.6.2 SCC Block Diagram

Figure 33.29 shows a block diagram of the sampling clock controller.

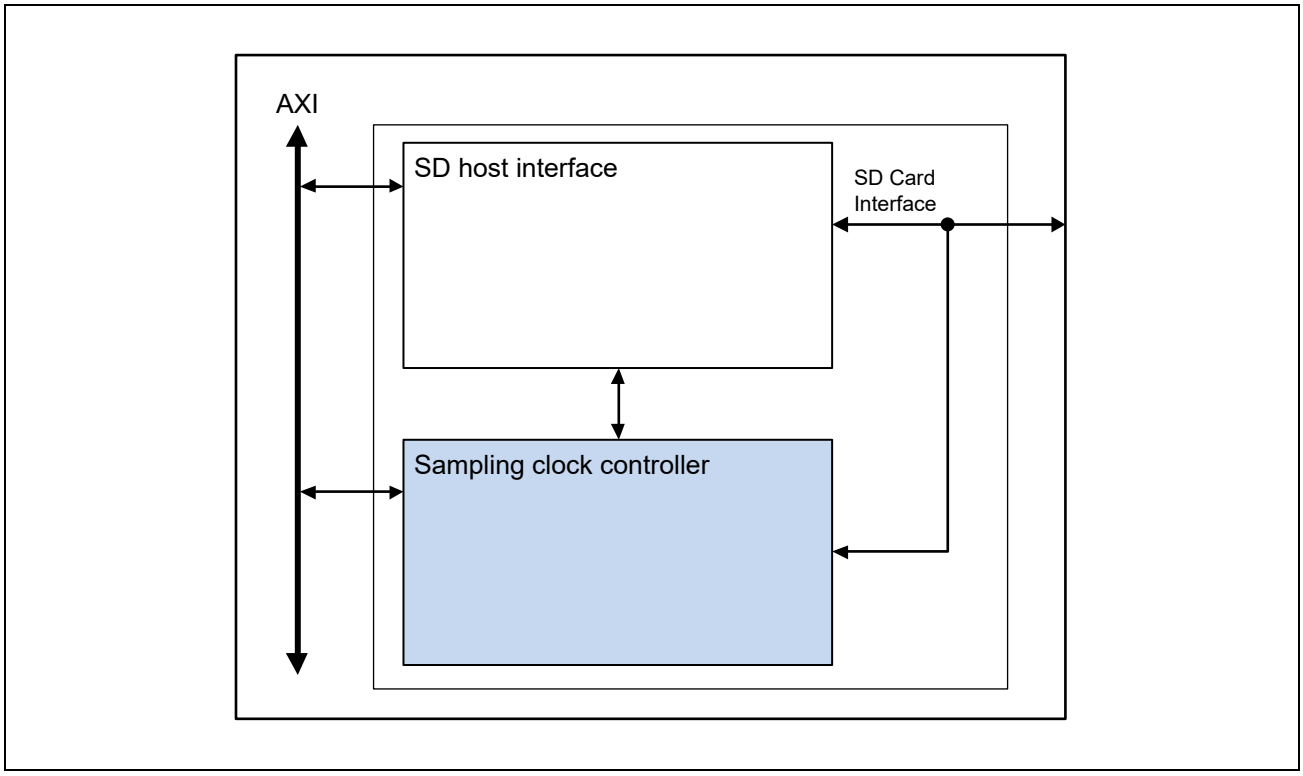


Figure 33.29 Block Diagram of the Sampling Clock Controller

33.6.3 SCC Register Configuration

Table 33.10 shows the SCC registers. Regarding the base address of registers as follows, refer to **Section 33.1.3, Register Configuration**.

Channel 0:

H'0_11C0_1000 (Cortex-A55 Address Space)
H'41C0_1000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
H'51C0_1000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Channel 1:

H'0_11C1_1000 (Cortex-A55 Address Space)
H'41C1_1000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
H'51C1_1000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 33.10 Register Configuration

Name	Abbreviation	Address [9:0]	Access Size
Initial setting register	SCC_DTCNTL	H'1000	32/64
Sampling clock position setting register	SCC_TAPSET	H'1008	32/64
Hardware adjustment register 1	SCC_DT2FF	H'1010	32/64
Sampling clock selection register	SCC_CKSEL	H'1018	32/64
Sampling clock position correction register	SCC_RVSCNTL	H'1020	32/64
Sampling clock position correction request register	SCC_RVSREQ	H'1028	32/64
Sampling data comparison register	SCC_SMPCMP	H'1030	32/64
Hardware adjustment register 2	SCC_TMPPORT	H'1038	32/64

33.7 SCC Register Descriptions

33.7.1 Initial Setting Register (SCC_DTCNTL)

Bit	Bit Name	Initial Value	R/W	Description
63 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
23 to 16	TAPNUM7 to TAPNUM0	H'08	R/W	When bits DIV7 to DIV0 in the SD_CLK_CTRL register are H'FF (1: 1 mode), set these bits to H'08.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	TAPEN	0	R/W	SCC Sampling Clock Operation Enable 0: SCC sampling clock operation is disabled. 1: SCC sampling clock operation is enabled.

33.7.2 Sampling Clock Position Setting Register (SCC_TAPSET)

Bit	Bit Name	Initial Value	R/W	Description
63 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
7 to 0	TAPSET7 to TAPSET0	H'00	R/W	SCC Sampling Clock Position • Set the tuning result in the range from 0 to TAPNUM-1.

33.7.3 Sampling Clock Selection Register (SCC_CKSEL)

Bit	Bit Name	Initial Value	R/W	Description
63 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	DTSEL	0	R/W	Sampling Clock Selection 0: An SCC sampling clock is not used (for other than SDR104 and HS200). 1: An SCC sampling clock is used (for SDR104 or HS200). • For SDR104 or HS200, set DTSEL to 1. DIV[7:0] in the SD_CLK_CTRL register to H'FF (1:1 mode). • When this bit is switched, stop the SD clock output from the SD/MMC host interface (set SCLKEN in SD_CLK_CTRL to 0).

33.7.4 Sampling Clock Position Correction Register (SCC_RVSCNTL)

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 8	TAPSEL7 to TAPSEL0	H'00	R	SCC Sampling Clock Position Display • Displays the SCC sampling clock position selected by hardware. • After RVSEN has been set to 1, the value may differ from that of TAPSET.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
1	RVSW	0	R/W	This bit is read as 0. The write value should be 1.
0	RVSEN	0	R/W	SCC Sampling Clock Position Correction Enable 0: SCC sampling clock position correction is disabled. 1: SCC sampling clock position correction is enabled. • When RVSEN is set to 1 after tuning has been performed, this module corrects the SCC sampling clock position each time of a command sequence of the SD/MMC host interface. • However, when RVSERR is 1, this module does not correct the SCC sampling clock position. • While tuning is being performed, set RVSEN to 0.

33.7.5 Sampling Clock Position Correction Request Register (SCC_RVSREQ)

Bit	Bit Name	Initial Value	R/W	Description
63 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2	RVSERR	0	R/W	SCC Sampling Clock Position Correction Error 0: There is no correction error. 1: There is a correction error. • If this bit is set to 1 after a command sequence, write 0 to this bit and perform tuning again. • Ignore this bit while tuning is being performed. • Writing 1 to this bit is disabled and writing 0 to this bit is only enabled.
1	REQTAPUP	0	R/W	SCC Sampling Clock Position Positive Direction Correction Request 0: There is no correction request. 1: There is a correction request. • If this bit is set to 1 after a command sequence, write 0 to this bit and rewrite TAPSET in the positive direction (when TAPSEL = TAPNUM-1, set 0 to TAPSET). • When RVSEN is 1, this bit is disabled (this bit is not set to 1). • Writing 1 to this bit is disabled and writing 0 to this bit is only enabled.
0	REQTAPDWN	0	R/W	SCC Sampling Clock Position Negative Direction Correction Request 0: There is no correction request. 1: There is a correction request. • If this bit is set to 1 after a command sequence, write 0 to this bit and rewrite TAPSET in the negative direction (when TAPSEL = 0, set TAPNUM-1 to TAPSET). • When RVSEN is 1, this bit is disabled (this bit is not set to 1). • Writing 1 to this bit is disabled and writing 0 to this bit is only enabled.

33.7.6 Hardware Adjustment Register 1 (SCC_DT2FF)

This register makes a setting that SD_DATA, which has been fetched by the sampling clock at each TAP position, is used in the appropriate timing.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 8	DT2NE	H'03	R/W	Hardware Adjustment 2 This is a setting register for adjusting the timing inside the IP when using Tuning. When using Tuning, set H'02 to this bit, make the setting before setting 1 in SCC_CKSEL [0] .DTSEL.
7 to 0	DT2NS	H'00	R/W	Hardware Adjustment 1 This is a setting register for adjusting the timing inside the IP when using Tuning. When using Tuning, set H'07 to this bit., make the setting before setting 1 in SCC_CKSEL [0] .DTSEL.

33.7.7 Sampling data comparison register (SCC_SMPCMP)

Data comparison register indicates the result of the comparison of the sampling data. The subject of the comparison is the before and the behind TAP.

Bit	Bit Name	Initial Value	R/W	Description
63 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
24 to 16	CMPNGU	All 0	R/W	Comparison of sampling data with the previous TAP Clock. Bit 16-23 is the comparison result of data 0-7. Bit-24 is the comparison result of CMD. 0: Match 1: Mismatch < Clear conditions > The start of the command sequence Write to SCC_TAPSET register
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8 to 0	CMPNGD	All 0	R/W	Comparison of sampling data with the after TAP Clock. Bit 0-7 is the comparison result of data 0-7. Bit-8 is the comparison result of CMD. 0: Match 1: Mismatch < Clear conditions > The start of the command sequence Write to SCC_TAPSET register

33.7.8 Hardware adjustment register 2 (SCC_TMPPORT)

This register makes a setting that SD_DATA, which has been fetched by the sampling clock at each TAP position, is used in the appropriate timing.

Bit	Bit Name	Initial Value	R/W	Description
63 to16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	TMPOUT	H'0000	R/W	Hardware adjustment 3 When using the delay tuning mechanism, set the following values. When TAPNUM = 8, set H'0000 to these bits. When operating with other than SDR104 / HS200, this register adjusts the clock delay of the Flip Flop that latches the received data from the outside of the chip. For this product, set H'0000 when transferring at 3.3 V, and set H'0001 when transferring at 1.8 V.

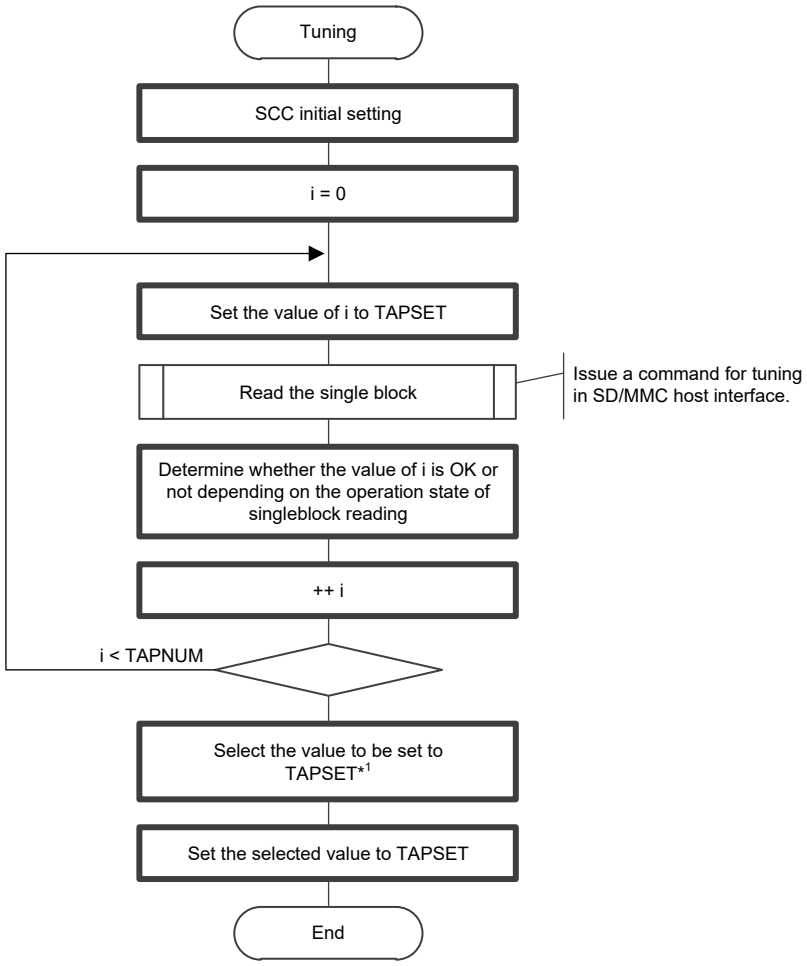
33.8 Usage Example of SCC

33.8.1 Tuning

SCC is tuned by using operation of single-block reading.

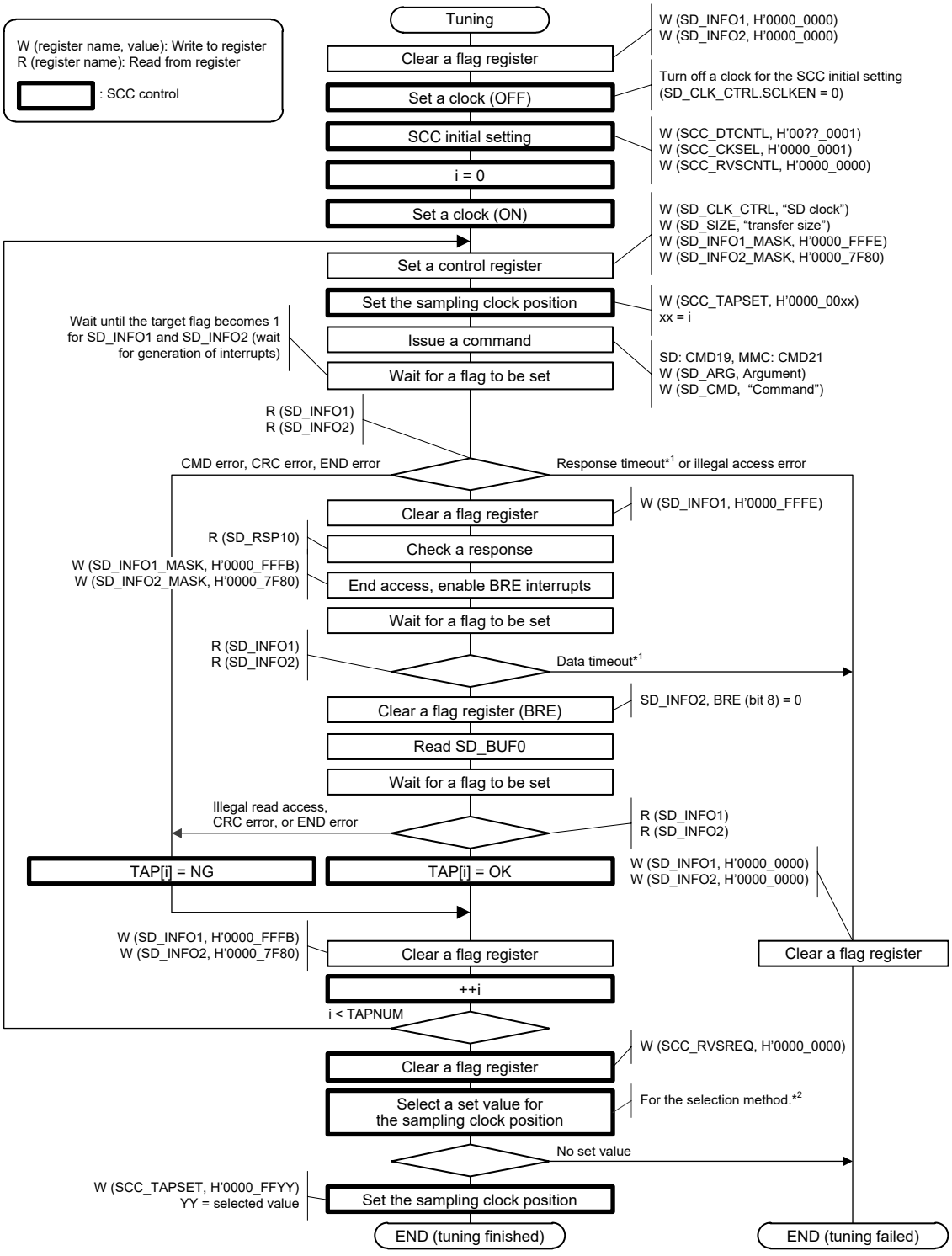
As shown in **Figure 33.30**, check whether the single-block read command normally ends when the sampling clock position is changed from 0 to TAPNUM-1 and save the result. After checking, confirm that there exists the range which has three or more continuous normal ends (OK). Then, the median value within the continuous range is determined as the final adjustment value.

Figure 33.31 and **Table 33.11** show the detailed tuning flow and the method how to select the sampling clock position (example when TAPNUM = 8).



Note 1. See the example shown in **Table 33.11**.

Figure 33.30 Example of Tuning Flow (Outline)



Note 1. If all of the TAP[i] has become "NG".

Note 2. See **Table 33.11**.

Figure 33.31 Example of Tuning Flow (Detailed)

Table 33.11 Example of the Method How to Select the Sampling Clock Position (when TAPNUM = 8)

Item	i	Case 1	Case 2	Case 3	Case 4	Case 5*1
TAP[i]	0	NG	OK	NG	OK	OK
	1	OK	OK	NG	NG	OK
	2	OK	NG	OK	NG	OK
	3	OK (←)	NG	OK	NG	OK
	4	OK	NG	NG	NG	OK
	5	OK	OK	NG	OK	OK
	6	NG	OK	NG	OK (←)	OK
Max. value→	7	NG	OK (←)	NG	OK	OK
	(0)	NG	OK	NG	OK	OK
	(1)	OK	OK	NG	NG	OK
	(2)	OK	NG	OK	NG	OK
	(3)	OK	NG	OK	NG	OK
	(4)	OK	NG	NG	NG	OK
	(5)	OK	OK	NG	OK	OK
	(6)	NG	OK	NG	OK	OK
	(7)	NG	OK	NG	OK	OK
Selected value		i = 3	i = 7	Fail	i = 6 or 7	i = 0 to 7

Remarks: (←): Example of selection, (x): repeated display of index x of TAP[x]

(a) The sampling clock position is selected by considering a margin in the range which has three or more continuous 'TAP[i] = OK'.

(b) The sampling clock position is repeated from 0 after the maximum value (TAPNUM-1). In case 2 above, that position is continued in the order of 5→6→7→0→1.

Note 1. If all of the TAP [i] is OK, the sampling clock position is selected by identifying the change point of data. Change point of the data can be found in the value of SCC_SMPCMP register. Usage example is **Section 33.8.3, Change point of the input data**.

33.8.2 Sampling Clock Position Correction after Tuning

After tuning, correction of the sampling clock position may be required when a command is issued.

There are manual and automatic correction methods. After a command sequence, if the CMD, CRC, END error or time out occurs or the correction error occurs, tuning will be performed again. The following shows examples of manual and automatic correction methods.

(1) Manual correction of the sampling clock position

Figure 33.32 shows the flow of manual correction of the sampling clock position. **Table 33.12** shows set values determined when correction is required (when TAPNUM = 8).

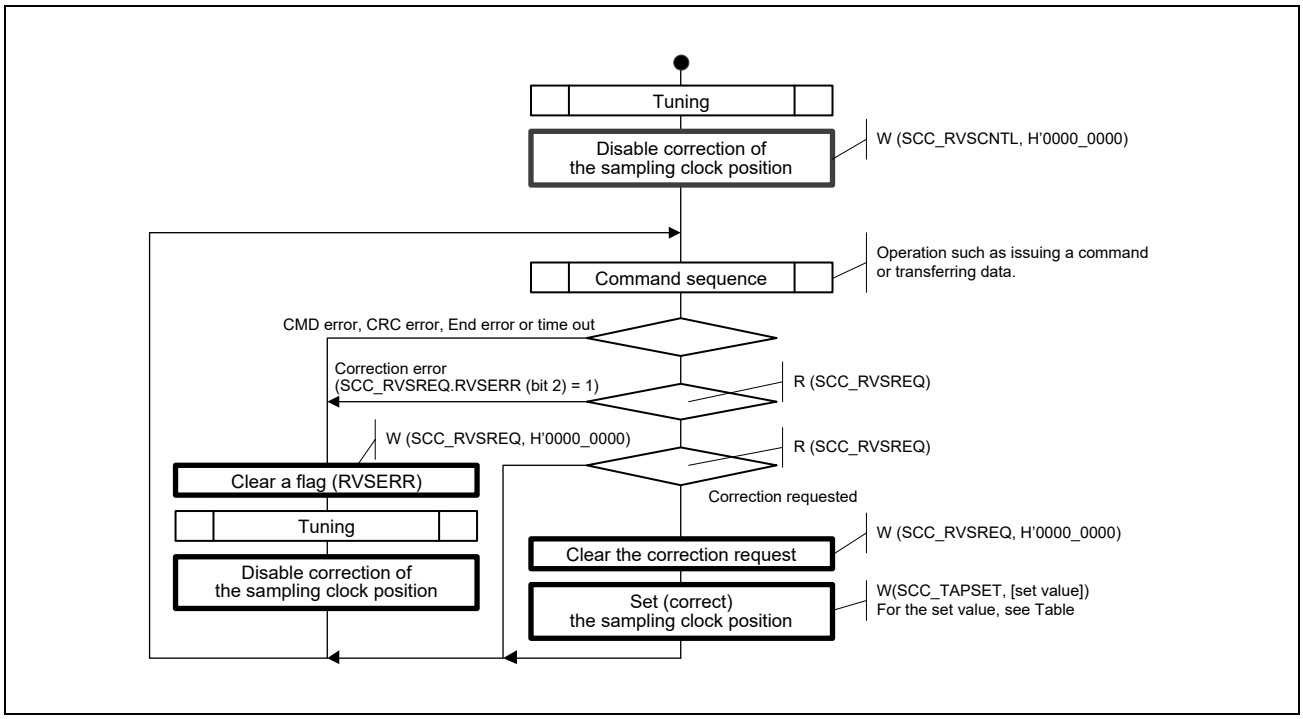


Figure 33.32 Flow of Manual Correction of the Sampling Clock Position (Example)

Table 33.12 Set Values for TAPSET when Correction is Required (when TAPNUM = 8)

No.	Current value of TAPSET	Value set to TAPSET when REQTAPUP = 1	Value set to TAPSET when REQTAPDWN = 1
1	0	1	7
2	1	2	0
3	2	3	1
4	3	4	2
5	4	5	3
6	5	6	4
7	6	7	5
8	7	0	6

Note: As is the case in the tuning selection method, the sampling clock position is 0 after the maximum value (TAPNUM-1).

(2) Automatic correction of the sampling clock position

Figure 33.33 shows the flow of automatic correction of the sampling clock position.

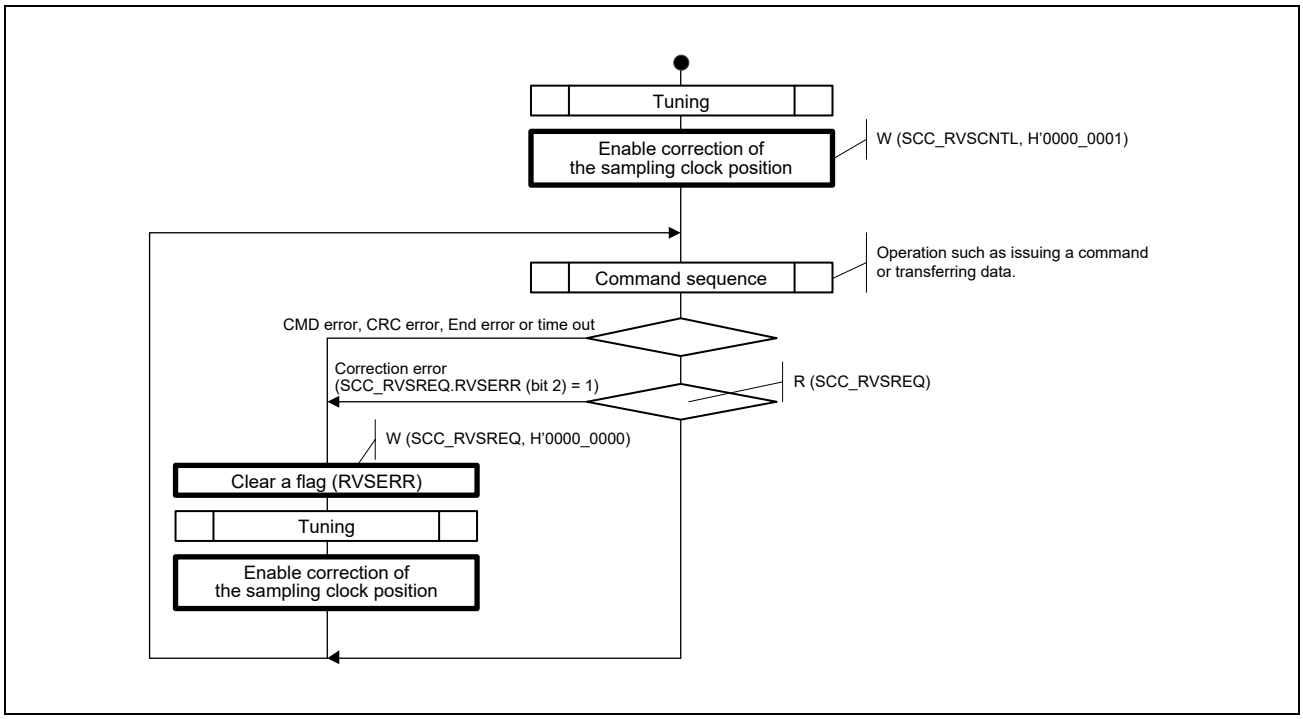
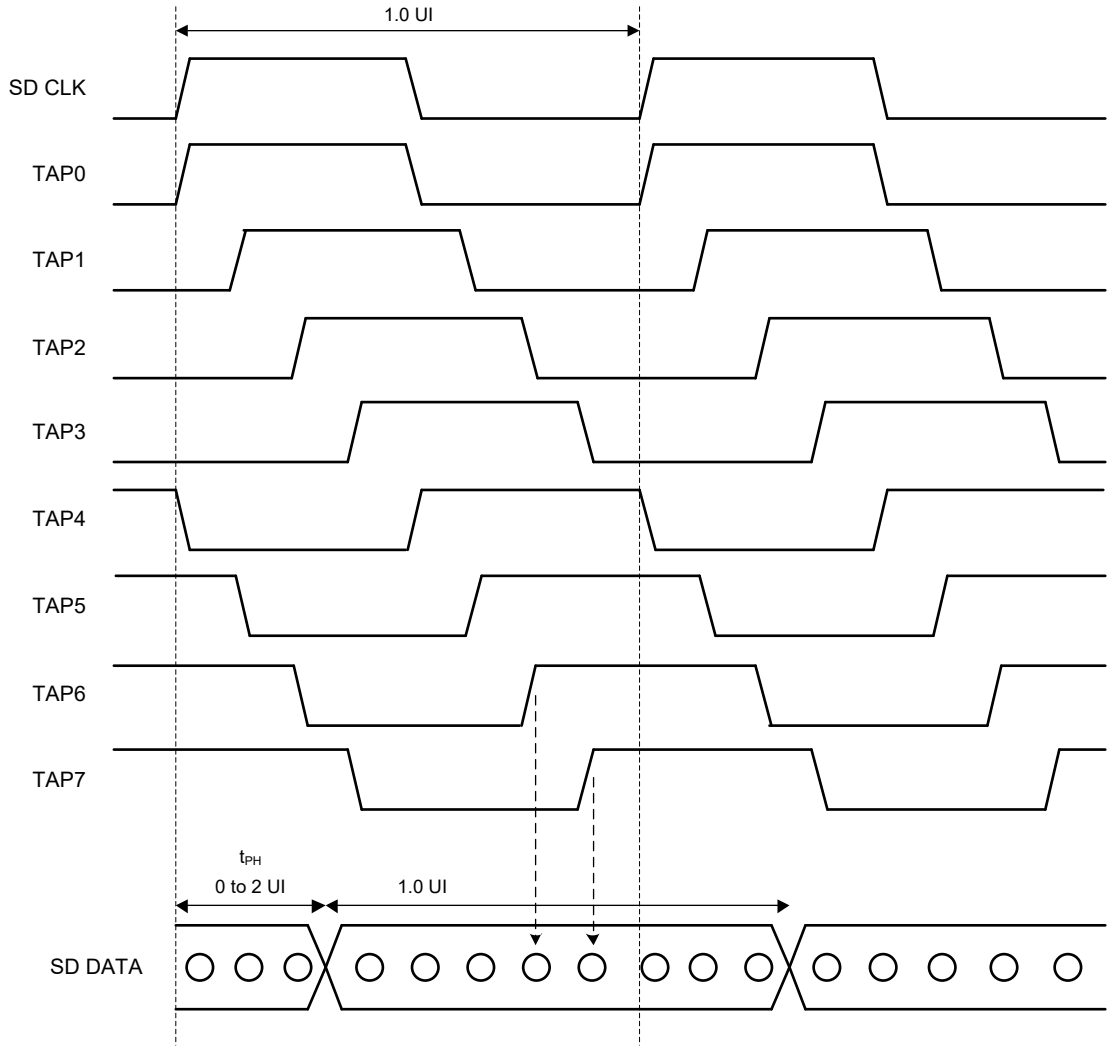


Figure 33.33 Flow of Automatic Correction of the Sampling Clock Position (Example)

33.8.3 Change point of the input data

Tuning is capturing the data by the TAP clock selected. However, also captures the data by the previous TAP clock and the behind the TAP clock at the same time. This result is reflected in the sampling data comparison register (SCC_SMPCMP). Point of mismatch before and after the selected TAP clock is the changing point of the data. In this example, it is desirable to set as TAP6 or TAP7.



Note: After recognizing the changing point of the input data, need to set the TAP clock.

Figure 33.34 Example, All taps is OK.

Figure 33.34 is shown change point of data between TAP2 and TAP3. Change point of the data can be confirmed by sampling data comparison register (SCC_SMPCMP). When the tuning of TAP2 or TAP3, CMPNGU bit of sampling data comparison register indicates a mismatch. As the width of the input data is 1 (UI), select TAP6 or TAP7 which is the median of next TAP3 from TAP3.

34. PCI Express Interface (PCIe)

This section describes the functions of the PCI Express interface (PCIe).

34.1 Overview

This module complies with PCI Express Base Specification 4.0 and supports up to 5.0 GT/s (Gen2).

It works as only a root complex and includes Type1 Configuration Register. It also has a built-in DMA controller function.

34.1.1 Features

■ AXI Interface Specification

Little endian is only supported.

- Master interface
  - 1 port
  - Bus width:128 bits
  - ID width: 4 bits (fixed value)
  - Interleave not supported
  - Burst type: Incremental (INCR 1 beat (MAxSIZE=1/2/4 bytes), INCR 2 to 16 beats (MAxSIZE = 16 bytes))
  - Byte-lane transfer and narrow transfer (8 bits/16 bits/32 bits MAxLEN = only in 1beat) are supported. Unaligned transfer is not supported.
  - Allowable number of read/write requests: Variable (1 to 16)
  - Write interleave depth: 1
  - Protection (data/user/non-secure) can be set by using registers.
  - Access type: Normal only (exclusive or locked access not supported)

- Slave interface
  - 1 port
  - Bus width:128 bits
  - ID width: 20 bits
  - Burst type: Incremental (1 to 16 beats (SAxSIZE = 1/2/4/8/16 bytes))
  - Support for byte-lane transfer (only when valid bytes are consecutive), unaligned transfer, and narrow transfer (1, 2, 4, 8 bytes)
  - Number of read transactions which can be accepted: 1 to 8
  - Number of write transactions which can be accepted: 1
  - Read data reordering depth: 1 to 8
  - Write interleave depth: 1
  - Required memory area: 16 Kbytes or larger (variable)

- Protection is not distinguished (reception is possible)
- Access type: Normal only (exclusive or locked access not supported)
- Cache signals (ARCACHE/AWCACHE) are only supported for bufferable bits for writing.

- DMAC
  - DMA method: Register control, descriptor control
  - Number of channels: 8
  - Allowable number of requests to be issued (the maximum total number: 8. When DMAC-ch1 issues eight PCIe MRd/MW requests, other channels cannot issue requests)
  - Outstanding: 8

    PCIe MRd: Up to 8/ch PCIe MWr: 1/ch

    AXI Read: 1/ch AXI Write: 1/ch

■ PCI Express Specification (Compliant with the PCI Express Base Specification 4.0)

- PCI Express Gen1(2.5[GT/s])/Gen2(5.0[GT/s])
- Root Complex Applications, Type1 Configuration Register
- Lane implementation x1
- Support Polarity inversion
- Maximum data payload of 256 bytes, Maximum read request size 512 bytes
- Not support for Virtual channels (support VC0 only)
- Number of outstanding 1-8
- Dynamic control of speed/width up/down configuration
- Not support for Clock Power Management (not support P1.CPM, P2.CPM)
- Power Management (ASPM L1-Substate Support (Support PowerDown Sequence only))
- Error handling/logging (AER Support)
- Replay FIFO with ECC
- Internal Memory without Parity
- Number of Support Functions 1

34.1.2 External Pins

The following table shows the external pins of PCIe.

Table 34.1 External Pin List

Pin Name	I/O	Function
PCIE_REFCLKP0	I	Reference clock input (Positive) (Differential: 100 MHz ±300 ppm)*1
PCIE_REFCLKN0	I	Reference clock input (Negative) (Differential: 100 MHz ±300 ppm)*1
PCIE_RX_DP	I	Serial data input (Positive)
PCIE_RX_DN	I	Serial data input (Negative)
PCIE_TX_DP	O	Serial data output (Positive)
PCIE_TX_DN	O	Serial data output (Negative)
PCI_RST_OUT#	O	Reset output (Low active)
PCI_CLKREQ#	I/O	Clock request (Low active) 1: Reference clock can be stopped 0: Reference clock cannot be stopped

Note 1. A jitter of the reference clock input is defined by the PCI Express Card Electromechanical (CEM) Specification.
Tccjitter < 150 ps (Cycle to Cycle jitter)

34.1.3 Block Diagram

The following figure shows the block diagram of PCIe.

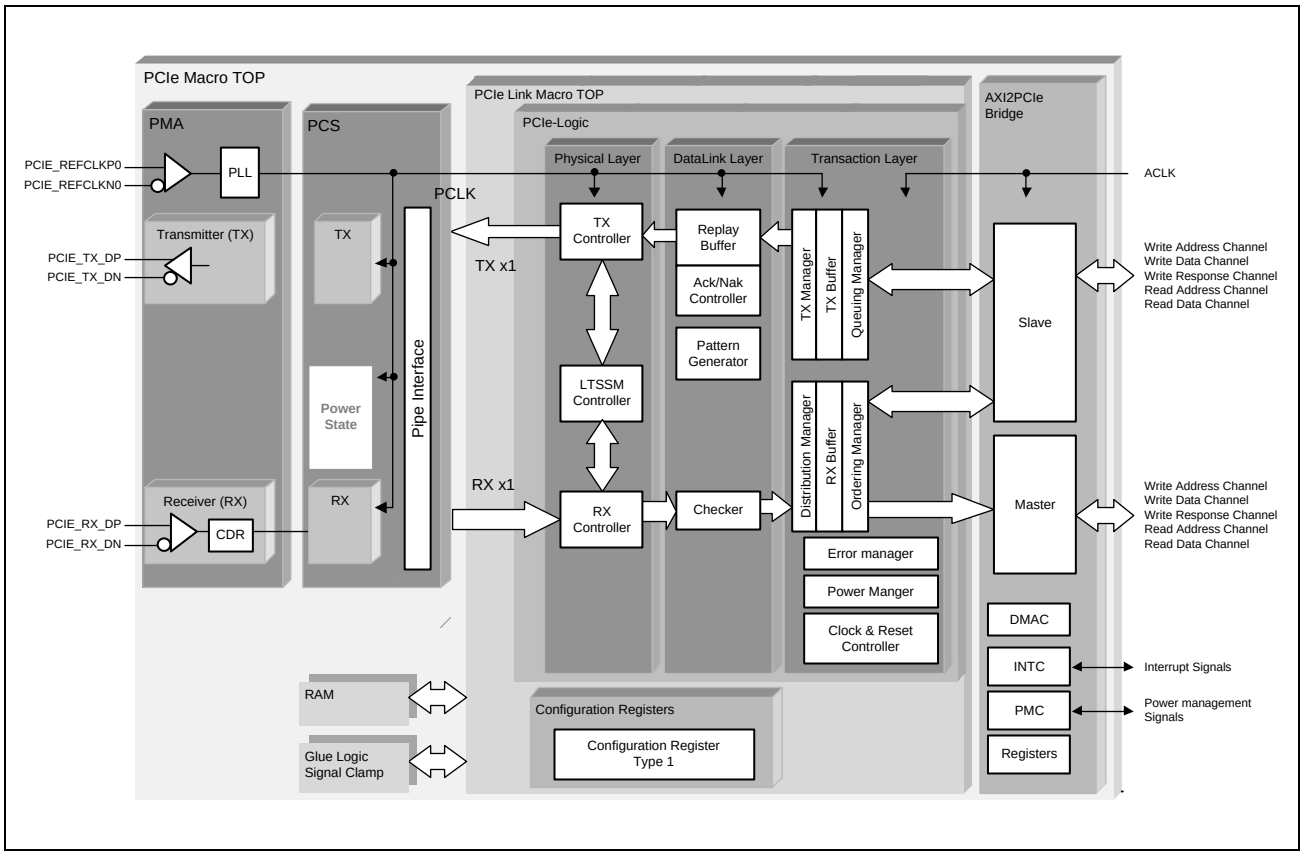


Figure 34.1 Block Diagram of PCIe

34.1.4 Reset

This module has the following reset signals.

These reset signals are controlled by the CPG_RST_PCI register. Refer to **Section 7.2.4, Register Descriptions**.

Table 34.2 Reset Signals

Reset Signal	Description	CPG_RST_PCI register
ARESETn	Reset for AXI2PCIe Bridge	Bit [0]: PCI_ARESETN
RST_B	Reset for Data Link layer/Transaction layer	Bit [1]: PCI_RST_B
PCI_RST_OUT#	Reset for Endpoint device	
RST_GP_B	Reset for Transaction layer (ACLK domain)	Bit [2]: PCI_RST_GP_B
RST_PS_B	Reset for Transaction layer (PCLK domain)	Bit [3]: PCI_RST_PS_B
RST_RSM_B	Reset for Physical layer	Bit [4]: PCI_RST_RSM_B
RST_CFG_B	Reset for Configuration register	Bit [5]: PCI_RST_CFG_B
RST_LOAD_B	Reset for Configuration register	Bit [6]: PCI_RST_LOAD_B

34.1.4.1 Reset Sequence for Power-On Reset

The following figure shows the reset sequence for power on reset.

ARESETn, RST_CFG_B and RST_LOAD_B signals should be de-asserted more than 1[ms] later after PRST# is de-asserted.

RST_B, RDT_PS_B, RST_GP_B and RST_RSM_B signals should be de-asserted after loading values into the Configuration Register.

If loading values into the Configuration Register are not needed, all reset signals (ARESETn, RST_CFG_B, RST_LOAD_B RST_B, RDT_PS_B, RST_GP_B and RST_RSM_B) can be de-asserted at the same timing.

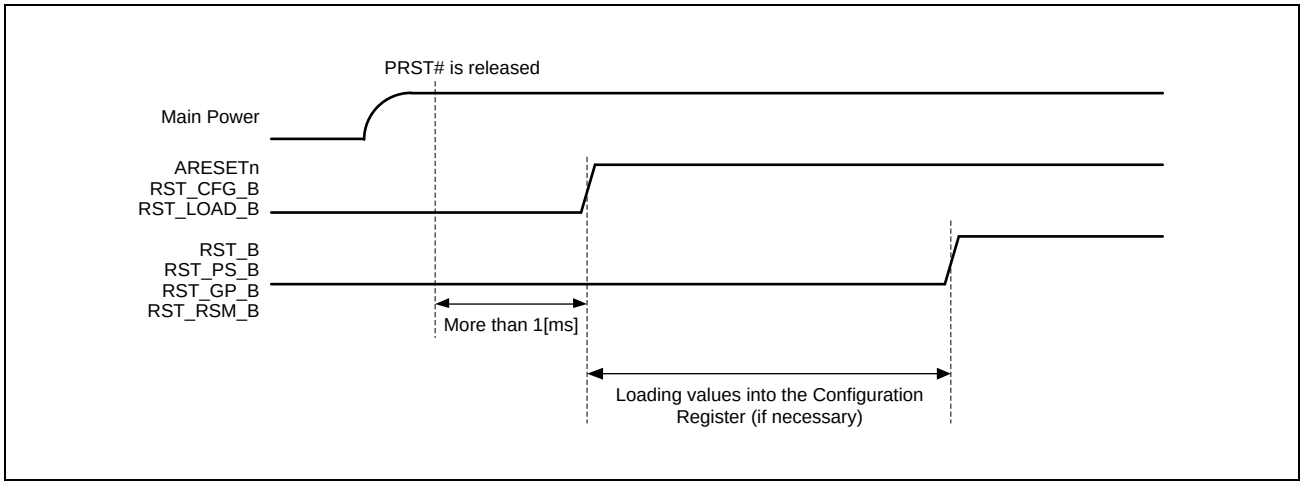


Figure 34.2 Reset Sequence for Power-On Reset

34.2 Register Configuration

34.2.1 Register Type

Table 34.3 Register Type

Register Attribute	Description
HwInit	Hardware Initialized*1
R	Read-only This register can only be read.
R/W	Read-Write This register can be read/written.
R/W1C	Write-1-to-clear status It can be cleared to 0b by writing 1b with a readable register. Writing 0b does not change anything.
RS	Sticky - Read-only Read-only registers that are reset with Resume Reset.*2
R/WS	Sticky - Read-Write A read-write register that is reset with Resume Reset.*2
R/W1CS	Sticky - Write-1-to-clear status Readable registers can be cleared to 0b by writing 1b. In addition, Resume reset indicates that this is a register that will be reset.*2

Note 1. HwInit attribute register.
Some registers with the RO attribute stated in the PCI Express Base Specification are writable at the time of initialization. When writing to these registers, CFG_HWINIT_EN (Permission Register (offset: H'300) bit[2]) must be set to 1b.

Note 2. The results of access to reserved bit areas, debug bit areas, and undefined areas are not guaranteed. The combination of reserved, debug, and undefined areas includes cases where the initial values will be non-zero, and the results of changes to such values are not guaranteed.

34.2.2 Register List

The base address of PCIe is as follows.

Base Address: H'0_11E4_0000 (Cortex-A55 Address Space)
Base Address: H'51E4_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
Base Address: H'41E4_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above is in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

The following is a summary classification of the register space in PCIe.

Table 34.4 Register Space

Register	Address Range
AXI Bridge Registers	H'0000 to H'1FFC
Reserved	H'2000 to H'5FFC
PCI Express Configuration Registers (Type1)	H'6000 to H'6FFC
Reserved	H'7000 to H'7FFC

Note: The reserved space that corresponds to the above is not accessible.

34.2.2.1 AXI Bridge Registers

Table 34.5 AXI Bridge Registers (1/6)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size (bits)
Reserved H'00 to H'7C					
Reserved	—	—	—	H'00 to H'7C	—
Request Issuing H'80 to H'FC					
Request Data Register 0	PCI_REQDATA0	R/W	H'xxxx_xxxx	H'80	32, 16, 8
Request Data Register 1	PCI_REQDATA1	R/W	H'xxxx_xxxx	H'84	32, 16, 8
Request Data Register 2	PCI_REQDATA2	R/W	H'xxxx_xxxx	H'88	32, 16, 8
Request Receive Data Registers	PCI_REQRCVDAT	R	H'xxxx_xxxx	H'8C	32, 16, 8
Request Address Registers 1	PCI_REQADR1	R/W	H'xxxx_xxxx	H'90	32, 16, 8
Request Address Registers 2	PCI_REQADR2	R/W	H'xxxx_xxxx	H'94	32, 16, 8
Request Byte Enable Registers	PCI_REQBE	R/W*1	H'0000_000F	H'98	32, 16, 8
Request Issue Registers	PCI_REQISS	R/W*1	H'0000_0000	H'9C	32, 16, 8
Reserved	—	—	—	H'A0 to FC	—
PCI Interruption H'100 to H'11C					
MSI Receive Window Address (Lower) Registers	PCI_MSIRCVWADRL	R/W*1	H'0000_0000	H'100	32, 16, 8
MSI Receive Window Address (Upper) Registers	PCI_MSIRCVWADRU	R/W	H'0000_0000	H'104	32, 16, 8
MSI Receive Window Mask (Lower) Registers	PCI_MSIRCVWMSKL	R/W*1	H'0000_0003	H'108	32, 16, 8
MSI Receive Window Mask (Upper) Registers	PCI_MSIRCVWMSKU	R/W	H'0000_0000	H'10C	32, 16, 8
PCI INTx Receive Interrupt Enable Registers	PCI_PINTRCVIE	R/W*1	H'0000_0000	H'110	32, 16, 8
PCI INTx Receive Interrupt Status Registers	PCI_PINTRCVIS	R/W1C*1	H'0000_0000	H'114	32, 16, 8
Reserved	—	—	—	H'118 to 11C	—
Message Interruption H'120 to H'13C					
Message Receive Interrupt Enable Registers	PCI_MSGRCVIE	R/W*1	H'0000_0000	H'120	32, 16, 8
Message Receive Interrupt Status Registers	PCI_MSGRCVIS	R/W1C*1	H'0000_0000	H'124	32, 16, 8
Reserved	—	—	—	H'128 to 12C	—
Message Code Registers	PCI_MSGCODE	R	H'0000_0000	H'130	32, 16, 8
Message Data Registers	PCI_MSGDATA	R	H'0000_0000	H'134	32, 16, 8
Message Header 3rdDW Registers	PCI_MSGH3DW	R	H'0000_0000	H'138	32, 16, 8
Message Header 4thDW Registers	PCI_MSGH4DW	R	H'0000_0000	H'13C	32, 16, 8
Interrupt Table H'140 to H'1FC					
Interrupt Table Registers	PCI_INTTABLE	R	H'00xx_0000	H'140	32, 16, 8
Reserved	—	—	—	H'144 to 1FC	—
Error Event H'200 to H'2FC					
PCIe Event Interrupt Enable 0 Registers	PCI_PEIE0	R/W*1	H'0000_0000	H'200	32, 16, 8
PCIe Event Interrupt Status 0 Registers	PCI_PEIS0	R/W1C*1	H'0000_0000	H'204	32, 16, 8
PCIe Event Interrupt Enable 1 Registers	PCI_PEIE1	R/W*1	H'0000_0000	H'208	32, 16, 8
PCIe Event Interrupt Status 1 Registers	PCI_PEIS1	R/W1C*1	H'0000_0000	H'20C	32, 16, 8
AXI Master Error Interrupt Enable Registers	PCI_AMEIE	R/W*1	H'0000_0000	H'210	32, 16, 8
AXI Master Error Interrupt Status Registers	PCI_AMEIS	R/W1C*1	H'0000_0000	H'214	32, 16, 8
Reserved	—	—	—	H'218 to 21C	—
AXI Slave Error Interrupt Enable 1 Registers	PCI_ASEIE1	R/W*1	H'0000_0000	H'220	32, 16, 8
AXI Slave Error Interrupt Status 1 Registers	PCI_ASEIS1	R/W1C*1	H'0000_0000	H'224	32, 16, 8
Reserved	—	—	—	H'228 to 22C	—
AXI Slave Error Interrupt Status 3 Registers	PCI_ASEIS3	R	H'0000_0000	H'230	32, 16, 8

Table 34.5 AXI Bridge Registers (2/6)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size (bits)
Reserved	—	—	—	H'234 to 2FC	—
Macro Control H'300 to H'7FC					
Permission Registers	PCI_PERM	R/W*1	H'0000_0000	H'300	32, 16, 8
Reserved	—	—	—	H'304 to 310	—
Mode Set 0 Registers	PCI_MSET0	R/W*1	H'2001_2000	H'314	32, 16, 8
Mode Set 1 Registers	PCI_MSET1	R/W*1	H'0000_33F2	H'318	32, 16, 8
Reserved	—	—	—	H'31C to 37C	—
Mode Set 3 Registers	PCI_MSET3	R/W*1	H'0000_0000	H'380	32, 16, 8
Reserved	—	—	—	H'384 to 3FC	—
PCIe Core Mode Set 1 Registers	PCI_PCMSET1	R/W*1	H'07D0_00F2	H'400	32, 16, 8
PCIe Core Control 1 Registers	PCI_PCCTRL1	R/W*1	H'0000_0000	H'404	32, 16, 8
PCIe Core Status 1 Registers	PCI_PCSTAT1	R	H'000x_xxxx	H'408	32, 16, 8
Reserved	—	—	—	H'40C	—
PCIe Core Control 2 Registers	PCI_PCCTRL2	R/W*1	H'003E_0000	H'410	32, 16, 8
PCIe Core Status 2 Registers	PCI_PCSTAT2	R	H'xxxx_xxxx	H'414	32, 16, 8
Reserved	—	—	—	H'418 to 428	—
PCIe Core Status 5 Registers	PCI_PCSTAT5	R	H'0000_0x00	H'42C	32, 16, 8
Reserved	—	—	—	H'430 to 46C	—
LTR Reported value Registers	PCI_LTRRV	R	H'0000_0000	H'470	32, 16, 8
Reserved	—	—	—	H'474 to 4CC	—
DMA Interrupt Vector 0 Registers	PCI_DMAINTVEC0	R/W*1	H'0000_0000	H'4D0	32, 16, 8
DMA Interrupt Vector 1 Registers	PCI_DMAINTVEC1	R/W*1	H'0000_0000	H'4D4	32, 16, 8
Reserved	—	—	—	H'4D8 to 5FC	—
Message Interruption2 H'600 to H'61C MSI receive register group (H'6x0 to H'61C) is equipped with 2 sets.					
MSI receive Enable 0 Registers	PCI_MSIRE0	R/W*1	H'0000_0000	H'600	32, 16, 8
MSI receive Message Data 0 Registers	PCI_MSIRMD0	R/W*1	H'0000_0000	H'604	32, 16, 8
MSI receive Mask 0 Registers	PCI_MSIRM0	R/W	H'FFFF_FFFF	H'608	32, 16, 8
MSI receive Status 0 Registers	PCI_MSIRS0	R/W1C	H'0000_0000	H'60C	32, 16, 8
MSI receive Enable 1 Registers	PCI_MSIRE1	R/W*1	H'0000_0000	H'610	32, 16, 8
MSI receive Message Data 1 Registers	PCI_MSIRMD1	R/W*1	H'0000_0000	H'614	32, 16, 8
MSI receive Mask 1 Registers	PCI_MSIRM1	R/W	H'FFFF_FFFF	H'618	32, 16, 8
MSI receive Status 1 Registers	PCI_MSIRS1	R/W1C	H'0000_0000	H'61C	32, 16, 8
Reserved	—	—	—	H'620 to 7FC	—
DMAC Registers H'800 to H'FFC					
Common Control H'800 to H'8FC					
DMA Control Registers	PCI_DMACTRL	R/W*1	H'0000_0000	H'800	32, 16, 8
Reserved	—	—	—	H'804	—
DMA Interrupt Enable Registers	PCI_DMAINTE	R/W	H'0000_0000	H'808	32, 16, 8
DMA Interrupt Status Registers	PCI_DMAINTS	R/W1C	H'0000_0000	H'80C	32, 16, 8
Reserved	—	—	—	H'810 to 8FC	—
Channel Control H'900 to H'91C*2					
DMA Channel Control Registers	PCI_DMACHCTL	R/W*1	H'0000_0000	H'900	32, 16, 8
Reserved	—	—	—	H'904	—
Descriptor Start Address (Lower) Registers	PCI_DESSAL	R/W*1	H'0000_0000	H'908	32, 16, 8

Table 34.5 AXI Bridge Registers (3/6)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size (bits)
Descriptor Start Address (Upper) Registers	PCI_DESSAU	R/W	H'0000_0000	H'90C	32, 16, 8
QUE Entry Registers	PCI_QUEE	R/W*1	H'0000_0000	H'910	32, 16, 8
Reserved	—	—	—	H'914 to 91C	—
DMA Setting H'920 to H'94C*2					
DMA Descriptor Control (Descriptor H'00)	PCI_DMADPCTL	R	H'0000_0000	H'920	32, 16, 8
DMA Transaction Control (Descriptor H'04)	PCI_DMATRCTL	R/W*1	H'0000_0000	H'924	32, 16, 8
DMA Size (Descriptor H'08)	PCI_DMASIZE	R/W*1	H'0000_0000	H'928	32, 16, 8
Reserved	—	—	—	H'92C	—
DMA Source Lower Address (Descriptor H'10)	PCI_DMAPCIEUA	R/W*1	H'0000_0000	H'930	32, 16, 8
DMA Source Upper Address (Descriptor H'14)	PCI_DMATCTL	R/W	H'0000_0000	H'934	32, 16, 8
DMA Destination Lower Address (Descriptor H'18)	PCI_DMADLA	R/W*1	H'0000_0000	H'938	32, 16, 8
DMA Destination Upper Address (Descriptor H'1C)	PCI_DMADUA	R/W	H'0000_0000	H'93C	32, 16, 8
DMA Descriptor Lower Link Pointer (Descriptor H'20)	PCI_DMADLLP	R	H'0000_0000	H'940	32, 16, 8
DMA Descriptor Upper Link Pointer (Descriptor H'24)	PCI_DMADULP	R	H'0000_0000	H'944	32, 16, 8
Reserved	—	—	—	H'948 to 94C	—
DMA Status H'950 to H'97C*2					
DMA Rest Size Registers	PCI_DMARESTSIZ	R	H'0000_0000	H'950	32, 16, 8
Reserved	—	—	—	H'954 to 95C	—
AXI Request Address (Lower) Registers	PCI_AXIREQAL	R	H'0000_0000	H'960	32, 16, 8
AXI Request Address (Upper) Registers	PCI_AXIREQAU	R	H'0000_0000	H'964	32, 16, 8
PCIe Request Address (Lower) Registers	PCI_PCIREQAL	R	H'0000_0000	H'968	32, 16, 8
PCIe Request Address (Upper) Registers	PCI_PCIREQAU	R	H'0000_0000	H'96C	32, 16, 8
QUE Status Registers	PCI_QUESTA	R	H'0000_0000	H'970	32, 16, 8
Reserved	—	—	—	H'974	—
DMAC Error Status Registers	PCI_DMACESTA	R	H'0000_0000	H'978	32, 16, 8
Reserved	—	—	—	H'97C	—
PCI Express to AXI Access H'1000 to H'10FC					
AXI Window Base 0 (Lower) Registers	PCI_AWBASE0L	R/W*1	H'0000_0000	H'1000	32, 16, 8
AXI Window Base 0 (Upper) Registers	PCI_AWBASE0U	R/W	H'0000_0000	H'1004	32, 16, 8
AXI Window Mask 0 (Lower) Registers	PCI_AWMASK0L	R/W*1	H'0000_0FFF	H'1008	32, 16, 8
AXI Window Mask 0 (Upper) Registers	PCI_AWMASK0U	R/W*1	H'0000_0000	H'100C	32, 16, 8
AXI Destination 0 (Lower) Registers	PCI_ADEST0L	R/W*1	H'0000_0000	H'1010	32, 16, 8
AXI Destination 0 (Upper) Registers	PCI_ADEST0U	R/W	H'0000_0000	H'1014	32, 16, 8
Reserved	—	—	—	H'1018 to 101C	—
AXI Window Base 1 (Lower) Registers	PCI_AWBASE1L	R/W*1	H'0000_0000	H'1020	32, 16, 8
AXI Window Base 1 (Upper) Registers	PCI_AWBASE1U	R/W	H'0000_0000	H'1024	32, 16, 8
AXI Window Mask 1 (Lower) Registers	PCI_AWMASK1L	R/W*1	H'0000_0FFF	H'1028	32, 16, 8
AXI Window Mask 1 (Upper) Registers	PCI_AWMASK1U	R/W*1	H'0000_0000	H'102C	32, 16, 8
AXI Destination 1 (Lower) Registers	PCI_ADEST1L	R/W*1	H'0000_0000	H'1030	32, 16, 8
AXI Destination 1 (Upper) Registers	PCI_ADEST1U	R/W	H'0000_0000	H'1034	32, 16, 8
Reserved	—	—	—	H'1038 to 103C	—
AXI Window Base 2 (Lower) Registers	PCI_AWBASE2L	R/W*1	H'0000_0000	H'1040	32, 16, 8
AXI Window Base 2 (Upper) Registers	PCI_AWBASE2U	R/W	H'0000_0000	H'1044	32, 16, 8
AXI Window Mask 2 (Lower) Registers	PCI_AWMASK2L	R/W*1	H'0000_0FFF	H'1048	32, 16, 8
AXI Window Mask 2 (Upper) Registers	PCI_AWMASK2U	R/W*1	H'0000_0000	H'104C	32, 16, 8

Table 34.5 AXI Bridge Registers (4/6)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size (bits)
AXI Destination 2 (Lower) Registers	PCI_ADEST2L	R/W*1	H'0000_0000	H'1050	32, 16, 8
AXI Destination 2 (Upper) Registers	PCI_ADEST2U	R/W	H'0000_0000	H'1054	32, 16, 8
Reserved	—	—	—	H'1058 to 105C	—
AXI Window Base 3 (Lower) Registers	PCI_AWBASE3L	R/W*1	H'0000_0000	H'1060	32, 16, 8
AXI Window Base 3 (Upper) Registers	PCI_AWBASE3U	R/W	H'0000_0000	H'1064	32, 16, 8
AXI Window Mask 3 (Lower) Registers	PCI_AWMASK3L	R/W*1	H'0000_0FFF	H'1068	32, 16, 8
AXI Window Mask 3 (Upper) Registers	PCI_AWMASK3U	R/W*1	H'0000_0000	H'106C	32, 16, 8
AXI Destination 3 (Lower) Registers	PCI_ADEST3L	R/W*1	H'0000_0000	H'1070	32, 16, 8
AXI Destination 3 (Upper) Registers	PCI_ADEST3U	R/W	H'0000_0000	H'1074	32, 16, 8
Reserved	—	—	—	H'1078 to 107C	—
AXI Window Base 4 (Lower) Registers	PCI_AWBASE4L	R/W*1	H'0000_0000	H'1080	32, 16, 8
AXI Window Base 4 (Upper) Registers	PCI_AWBASE4U	R/W	H'0000_0000	H'1084	32, 16, 8
AXI Window Mask 4 (Lower) Registers	PCI_AWMASK4L	R/W*1	H'0000_0FFF	H'1088	32, 16, 8
AXI Window Mask 4 (Upper) Registers	PCI_AWMASK4U	R/W*1	H'0000_0000	H'108C	32, 16, 8
AXI Destination 4 (Lower) Registers	PCI_ADEST4L	R/W*1	H'0000_0000	H'1090	32, 16, 8
AXI Destination 4 (Upper) Registers	PCI_ADEST4U	R/W	H'0000_0000	H'1094	32, 16, 8
Reserved	—	—	—	H'1098 to 109C	—
AXI Window Base 5 (Lower) Registers	PCI_AWBASE5L	R/W*1	H'0000_0000	H'10A0	32, 16, 8
AXI Window Base 5 (Upper) Registers	PCI_AWBASE5U	R/W	H'0000_0000	H'10A4	32, 16, 8
AXI Window Mask 5 (Lower) Registers	PCI_AWMASK5L	R/W*1	H'0000_0FFF	H'10A8	32, 16, 8
AXI Window Mask 5 (Upper) Registers	PCI_AWMASK5U	R/W*1	H'0000_0000	H'10AC	32, 16, 8
AXI Destination 5 (Lower) Registers	PCI_ADEST5L	R/W*1	H'0000_0000	H'10B0	32, 16, 8
AXI Destination 5 (Upper) Registers	PCI_ADEST5U	R/W	H'0000_0000	H'10B4	32, 16, 8
Reserved	—	—	—	H'10B8 to 10BC	—
AXI Window Base 6 (Lower) Registers	PCI_AWBASE6L	R/W*1	H'0000_0000	H'10C0	32, 16, 8
AXI Window Base 6 (Upper) Registers	PCI_AWBASE6U	R/W	H'0000_0000	H'10C4	32, 16, 8
AXI Window Mask 6 (Lower) Registers	PCI_AWMASK6L	R/W*1	H'0000_0FFF	H'10C8	32, 16, 8
AXI Window Mask 6 (Upper) Registers	PCI_AWMASK6U	R/W*1	H'0000_0000	H'10CC	32, 16, 8
AXI Destination 6 (Lower) Registers	PCI_ADEST6L	R/W*1	H'0000_0000	H'10D0	32, 16, 8
AXI Destination 6 (Upper) Registers	PCI_ADEST6U	R/W	H'0000_0000	H'10D4	32, 16, 8
Reserved	—	—	—	H'10D8 to 10DC	—
AXI Window Base 7 (Lower) Registers	PCI_AWBASE7L	R/W*1	H'0000_0000	H'10E0	32, 16, 8
AXI Window Base 7 (Upper) Registers	PCI_AWBASE7U	R/W	H'0000_0000	H'10E4	32, 16, 8
AXI Window Mask 7 (Lower) Registers	PCI_AWMASK7L	R/W*1	H'0000_0FFF	H'10E8	32, 16, 8
AXI Window Mask 7 (Upper) Registers	PCI_AWMASK7U	R/W*1	H'0000_0000	H'10EC	32, 16, 8
AXI Destination 7 (Lower) Registers	PCI_ADEST7L	R/W*1	H'0000_0000	H'10F0	32, 16, 8
AXI Destination 7 (Upper) Registers	PCI_ADEST7U	R/W	H'0000_0000	H'10F4	32, 16, 8
Reserved	—	—	—	H'10F8 to 10FC	—
AXI to PCI Express Access H'1100 to H'11FC					
PCIe Window Base 0 (Lower) Registers	PCI_PWBASE0L	R/W*1	H'0000_0000	H'1100	32, 16, 8
PCIe Window Base 0 (Upper) Registers	PCI_PWBASE0U	R/W	H'0000_0000	H'1104	32, 16, 8
PCIe Window Mask 0 (Lower) Registers	PCI_PWMASK0L	R/W*1	H'0000_0FFF	H'1108	32, 16, 8
PCIe Window Mask 0 (Upper) Registers	PCI_PWMASK0U	R/W*1	H'0000_0000	H'110C	32, 16, 8
PCIe Destination 0 (Lower) Registers	PCI_PDEST0L	R/W*1	H'0000_0000	H'1110	32, 16, 8
PCIe Destination 0 (Upper) Registers	PCI_PDEST0U	R/W	H'0000_0000	H'1114	32, 16, 8

Table 34.5 AXI Bridge Registers (5/6)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size (bits)
Reserved	—	—	—	H'1118 to 111C	—
PCIe Window Base 1 (Lower) Registers	PCI_PWBASE1L	R/W*1	H'0000_0000	H'1120	32, 16, 8
PCIe Window Base 1 (Upper) Registers	PCI_PWBASE1U	R/W	H'0000_0000	H'1124	32, 16, 8
PCIe Window Mask 1 (Lower) Registers	PCI_PWMASK1L	R/W*1	H'0000_0FFF	H'1128	32, 16, 8
PCIe Window Mask 1 (Upper) Registers	PCI_PWMASK1U	R/W*1	H'0000_0000	H'112C	32, 16, 8
PCIe Destination 1 (Lower) Registers	PCI_PDEST1L	R/W*1	H'0000_0000	H'1130	32, 16, 8
PCIe Destination 1 (Upper) Registers	PCI_PDEST1U	R/W	H'0000_0000	H'1134	32, 16, 8
Reserved	—	—	—	H'1138 to 113C	—
PCIe Window Base 2 (Lower) Registers	PCI_PWBASE2L	R/W*1	H'0000_0000	H'1140	32, 16, 8
PCIe Window Base 2 (Upper) Registers	PCI_PWBASE2U	R/W	H'0000_0000	H'1144	32, 16, 8
PCIe Window Mask 2 (Lower) Registers	PCI_PWMASK2L	R/W*1	H'0000_0FFF	H'1148	32, 16, 8
PCIe Window Mask 2 (Upper) Registers	PCI_PWMASK2U	R/W*1	H'0000_0000	H'114C	32, 16, 8
PCIe Destination 2 (Lower) Registers	PCI_PDEST2L	R/W*1	H'0000_0000	H'1150	32, 16, 8
PCIe Destination 2 (Upper) Registers	PCI_PDEST2U	R/W	H'0000_0000	H'1154	32, 16, 8
Reserved	—	—	—	H'1158 to 115C	—
PCIe Window Base 3 (Lower) Registers	PCI_PWBASE3L	R/W*1	H'0000_0000	H'1160	32, 16, 8
PCIe Window Base 3 (Upper) Registers	PCI_PWBASE3U	R/W	H'0000_0000	H'1164	32, 16, 8
PCIe Window Mask 3 (Lower) Registers	PCI_PWMASK3L	R/W*1	H'0000_0FFF	H'1168	32, 16, 8
PCIe Window Mask 3 (Upper) Registers	PCI_PWMASK3U	R/W*1	H'0000_0000	H'116C	32, 16, 8
PCIe Destination 3 (Lower) Registers	PCI_PDEST3L	R/W*1	H'0000_0000	H'1170	32, 16, 8
PCIe Destination 3 (Upper) Registers	PCI_PDEST3U	R/W	H'0000_0000	H'1174	32, 16, 8
Reserved	—	—	—	H'1178 to 117C	—
PCIe Window Base 4 (Lower) Registers	PCI_PWBASE4L	R/W*1	H'0000_0000	H'1180	32, 16, 8
PCIe Window Base 4 (Upper) Registers	PCI_PWBASE4U	R/W	H'0000_0000	H'1184	32, 16, 8
PCIe Window Mask 4 (Lower) Registers	PCI_PWMASK4L	R/W*1	H'0000_0FFF	H'1188	32, 16, 8
PCIe Window Mask 4 (Upper) Registers	PCI_PWMASK4U	R/W*1	H'0000_0000	H'118C	32, 16, 8
PCIe Destination 4 (Lower) Registers	PCI_PDEST4L	R/W*1	H'0000_0000	H'1190	32, 16, 8
PCIe Destination 4 (Upper) Registers	PCI_PDEST4U	R/W	H'0000_0000	H'1194	32, 16, 8
Reserved	—	—	—	H'1198 to 119C	—
PCIe Window Base 5 (Lower) Registers	PCI_PWBASE5L	R/W*1	H'0000_0000	H'11A0	32, 16, 8
PCIe Window Base 5 (Upper) Registers	PCI_PWBASE5U	R/W	H'0000_0000	H'11A4	32, 16, 8
PCIe Window Mask 5 (Lower) Registers	PCI_PWMASK5L	R/W*1	H'0000_0FFF	H'11A8	32, 16, 8
PCIe Window Mask 5 (Upper) Registers	PCI_PWMASK5U	R/W*1	H'0000_0000	H'11AC	32, 16, 8
PCIe Destination 5 (Lower) Registers	PCI_PDEST5L	R/W*1	H'0000_0000	H'11B0	32, 16, 8
PCIe Destination 5 (Upper) Registers	PCI_PDEST5U	R/W	H'0000_0000	H'11B4	32, 16, 8
Reserved	—	—	—	H'11B8 to 11BC	—
PCIe Window Base 6 (Lower) Registers	PCI_PWBASE6L	R/W*1	H'0000_0000	H'11C0	32, 16, 8
PCIe Window Base 6 (Upper) Registers	PCI_PWBASE6U	R/W	H'0000_0000	H'11C4	32, 16, 8
PCIe Window Mask 6 (Lower) Registers	PCI_PWMASK6L	R/W*1	H'0000_0FFF	H'11C8	32, 16, 8
PCIe Window Mask 6 (Upper) Registers	PCI_PWMASK6U	R/W*1	H'0000_0000	H'11CC	32, 16, 8
PCIe Destination 6 (Lower) Registers	PCI_PDEST6L	R/W*1	H'0000_0000	H'11D0	32, 16, 8
PCIe Destination 6 (Upper) Registers	PCI_PDEST6U	R/W	H'0000_0000	H'11D4	32, 16, 8
Reserved	—	—	—	H'11D8 to 11DC	—
PCIe Window Base 7 (Lower) Registers	PCI_PWBASE7L	R/W*1	H'0000_0000	H'11E0	32, 16, 8
PCIe Window Base 7 (Upper) Registers	PCI_PWBASE7U	R/W	H'0000_0000	H'11E4	32, 16, 8

Table 34.5 AXI Bridge Registers (6/6)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size (bits)
PCIe Window Mask 7 (Lower) Registers	PCI_PWMASK7L	R/W*1	H'0000_0FFF	H'11E8	32, 16, 8
PCIe Window Mask 7 (Upper) Registers	PCI_PWMASK7U	R/W*1	H'0000_0000	H'11EC	32, 16, 8
PCIe Destination 7 (Lower) Registers	PCI_PDEST7L	R/W*1	H'0000_0000	H'11F0	32, 16, 8
PCIe Destination 7 (Upper) Registers	PCI_PDEST7U	R/W	H'0000_0000	H'11F4	32, 16, 8
Reserved	—	—	—	H'11F8 to 1FFC	—

Note: Reserved: Reserved registers. The results of access to these registers and operation of the core through such access are not guaranteed.

Note 1. Depending on the field, Read Only bit exists.

Note 2. The addresses listed in the above table are those for channel 0. These addresses should be read as the addresses for the other DMAC channels according to the table below.

Channel Offset Address	
offset + H'000	Channel 0
offset + H'080	Channel 1
offset + H'100	Channel 2
offset + H'180	Channel 3
offset + H'200	Channel 4
offset + H'280	Channel 5
offset + H'300	Channel 6
offset + H'380	Channel 7

[Example]

Channel 0: DMA Descriptor Control Register (offset: H'920 = H'920 + H'000)
Channel 1: DMA Descriptor Control Register (offset: H'9A0 = H'920 + H'080)
Channel 2: DMA Descriptor Control Register (offset: H'A20 = H'920 + H'100)
Channel 3: DMA Descriptor Control Register (offset: H'AA0 = H'920 + H'180)
Channel 4: DMA Descriptor Control Register (offset: H'B20 = H'920 + H'200)
Channel 5: DMA Descriptor Control Register (offset: H'BA0 = H'920 + H'280)
Channel 6: DMA Descriptor Control Register (offset: H'C20 = H'920 + H'300)
Channel 7: DMA Descriptor Control Register (offset: H'CA0 = H'920 + H'380)

34.2.2.2 PCI Express Configuration Registers (Type1)

Table 34.6 PCI Express Configuration Registers (1/2)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size (bits)
Common Configuration Space					
Vendor and Device ID	PCI_CFG_VID	HwInit	H'0033_1912	H'6000	32, 16, 8
Command and Status	PCI_CFG_COM_STA	R/W*2	H'0010_0000	H'6004	32, 16, 8
Revision ID and Class Code	PCI_CFG_RID_CC	HwInit	H'0604_0000	H'6008	32, 16, 8
Cache Line and Header Type	PCI_CFG_CL_HT	R/W*1	H'0001_0000	H'600C	32, 16, 8
Configuration Space					
Base Address Register 0	PCI_CFG_BAR0	R/W*3	H'0000_0004	H'6010	32, 16, 8
Base Address Register 1	PCI_CFG_BAR1	R/W	H'0000_0000	H'6014	32, 16, 8
Bus Number Register	PCI_CFG_BNR	R/W*1	H'0000_0000	H'6018	32, 16, 8
I/O Base/Limit and Secondary Status	PCI_CFG_IOBL_SS	R/W*2	H'0000_0000	H'601C	32, 16, 8
Memory Base/Limit Register	PCI_CFG_MEMBL	R/W*1	H'FFF0_FFF0	H'6020	32, 16, 8
Prefetchable Memory Base/Limit Register	PCI_CFG_PMBL	R/W*1	H'0001_0001	H'6024	32, 16, 8
Prefetchable Base Upper 32bits Register	PCI_CFG_PBUP32	R/W	H'0000_0000	H'6028	32, 16, 8
Prefetchable Limit Upper 32bits Register	PCI_CFG_PLUP32	R/W	H'0000_0000	H'602C	32, 16, 8
Reserved	—	—	—	H'6030	—
Capability Pointer	PCI_CFG_CP	HwInit*1	H'0000_0040	H'6034	32, 16, 8
Reserved	—	—	—	H'6038	—
Bridge Control and Interrupt	PCI_CFG_BC_INT	R/W*3	H'0000_0000	H'603C	32, 16, 8
PCI Power Management Capability Structure					
PM Capabilities	PCI_CFG_PMC	HwInit*1	H'4803_6001	H'6040	32, 16, 8
PM Status/Control	PCI_CFG_PMSC	R/W*2,*8	H'0000_0008	H'6044	32, 16, 8
Reserved	—	—	—	H'6048 to H'605C	—
PCI Express Capability Structure					
PCI Express Capability	PCI_CFG_PCIEC	HwInit*1	H'0042_0010	H'6060	32, 16, 8
Device Capabilities	PCI_CFG_DEVC	HwInit*1	H'0000_8001	H'6064	32, 16, 8
Device Control/Status	PCI_CFG_DEVCS	R/W*2	H'0000_2010	H'6068	32, 16, 8
Link Capabilities	PCI_CFG_LINKC	HwInit*1	H'0073_4C12	H'606C	32, 16, 8
Link Control/Status	PCI_CFG_LINKCS	R/W*4	H'1000_0008	H'6070	32, 16, 8
Slot Capabilities	PCI_CFG_SLOTC	HwInit	H'0000_0000	H'6074	32, 16, 8
Slot Control/Status	PCI_CFG_SLOTCS	R/W*2	H'0040_0000	H'6078	32, 16, 8
Root Control/Capabilities	PCI_CFG_ROOTCC	R/W*1	H'0000_0000	H'607C	32, 16, 8
Root Status	PCI_CFG_ROOTS	R*5	H'0000_0000	H'6080	32, 16, 8
Device Capabilities 2	PCI_CFG_DEVC2	HwInit*1	H'0000_0812	H'6084	32, 16, 8
Device Control 2/Status 2	PCI_CFG_DEVCS2	R/W*1	H'0000_0000	H'6088	32, 16, 8
Link Capabilities 2	PCI_CFG_LINKC2	HwInit*1	H'0000_0006	H'608C	32, 16, 8
Link Control 2/Status 2	PCI_CFG_LINCS2	R/W1CS *3,*7,*8	H'0000_0002	H'6090	32, 16, 8
Reserved	—	—	—	H'6094 to H'609C	H'609C
Specific Registers*9					
Base Address Register Mask 00 (Lower)	PCI_CFG_BARMSK00L	HwInit	H'FFFF_FFFF*6	H'60A0	32, 16, 8
Base Address Register Mask 00 (Upper)	PCI_CFG_BARMSK00U	HwInit	H'FFFF_FFFF*6	H'60A4	32, 16, 8
Reserved	—	—	—	H'60A8 to H'60C4	—

Table 34.6 PCI Express Configuration Registers (2/2)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size (bits)
Base Size 00/01	PCI_CFG_BSIZE00_01	HwInit*1	H'0000_0000	H'60C8	32, 16, 8
Reserved	—	—	—	H'60CC to H'60D4	
Type Supported 00/01/02	PCI_CFG_TSUPPORT00_01_02	HwInit	H'0033_3333	H'60D8	32, 16, 8
Reserved	—	—	—	H'60DC to H'60FC	—
Advanced Error Reporting (AER) Capability					
Advanced Error Reporting Capability	PCI_CFG_ADVERC	HwInit*1	H'1501_0001	H'6100	32, 16, 8
Uncorrectable Error Status Register	PCI_CFG_UNCESTS	R/W1CS*1	H'0000_0000	H'6104	32, 16, 8
Uncorrectable Error Mask Register	PCI_CFG_UNCEMASK	R/WS*1	H'0000_0000	H'6108	32, 16, 8
Uncorrectable Error Severity Register	PCI_CFG_UNCESVY	R/WS*3	H'0046_2030	H'610C	32, 16, 8
Correctable Error Status Register	PCI_CFG_CESTS	R/W1CS*1	H'0000_0000	H'6110	32, 16, 8
Correctable Error Mask Register	PCI_CFG_CEMASK	R/WS*1	H'0000_2000	H'6114	32, 16, 8
Advanced Error Capabilities and Control Register	PCI_CFG_ADVECC	R/WS*1,*7	H'0000_00A0	H'6118	32, 16, 8
Header Log Register 0	PCI_CFG_HLOG0	RS	H'0000_0000	H'611C	32, 16, 8
Header Log Register 1	PCI_CFG_HLOG1	RS	H'0000_0000	H'6120	32, 16, 8
Header Log Register 2	PCI_CFG_HLOG2	RS	H'0000_0000	H'6124	32, 16, 8
Header Log Register 3	PCI_CFG_HLOG3	RS	H'0000_0000	H'6128	32, 16, 8
Root Error Command	PCI_CFG_ROOTEC	R/W*1	H'0000_0000	H'612C	32, 16, 8
Root Error Status	PCI_CFG_ROOTES	R/W1C*1	H'0000_0000	H'6130	32, 16, 8
Error Source Identification Register	PCI_CFG_ERRSI	R	H'0000_0000	H'6134	32, 16, 8
Reserved	—	—	—	H'6138 to H'614C	—
Device Serial Number Capability					
Device Serial Number Extended Capability	PCI_CFG_DEVSNEXTC	HwInit*1	H'1601_0003	H'6150	32, 16, 8
Serial Number Register (Lower DW)	PCI_CFG_SNL	HwInit	H'0000_0000	H'6154	32, 16, 8
Serial Number Register (Upper DW)	PCI_CFG_SNU	HwInit	H'0000_0000	H'6158	32, 16, 8
Reserved	—	—	—	H'615C	—
L1 PM Substates Capability					
L1 PM Substates Extended Capability Header	PCI_CFG_L1PMSECH	HwInit*1	H'0001_001E	H'6160	32, 16, 8
L1 PM Substates Capabilities Register	PCI_CFG_L1PMSC	HwInit*1	H'0028_001F	H'6164	32, 16, 8
L1 PM Substates Control 1 Register	PCI_CFG_L1PMSCTRL1	R/W*1	H'0000_0000	H'6168	32, 16, 8
L1 PM Substates Control 2 Register	PCI_CFG_L1PMSCTRL2	R/W*1	H'0000_0028	H'616C	32, 16, 8
Reserved	—	—	—	H'6170 to H'6FFC	—

Note 1. Depending on the field, Read Only bit exists.

Note 2. Depending on the field, R, R/W1C bit exists.

Note 3. Depending on the field, R, HwInit bit exists.

Note 4. Depending on the field, R, R/W1C, HwInit bit exists.

Note 5. Depending on the field, R/W1C bit exists.

Note 6. Recommended to write All “1b” at initial setting.

Note 7. Depending on the field, RS (Sticky - Read - Only) bit exists.

Note 8. Depending on the field, R/WS (Sticky - Read - Write) bit exists.

Note 9. “Specific Registers” refers to registers specific to this LSI.

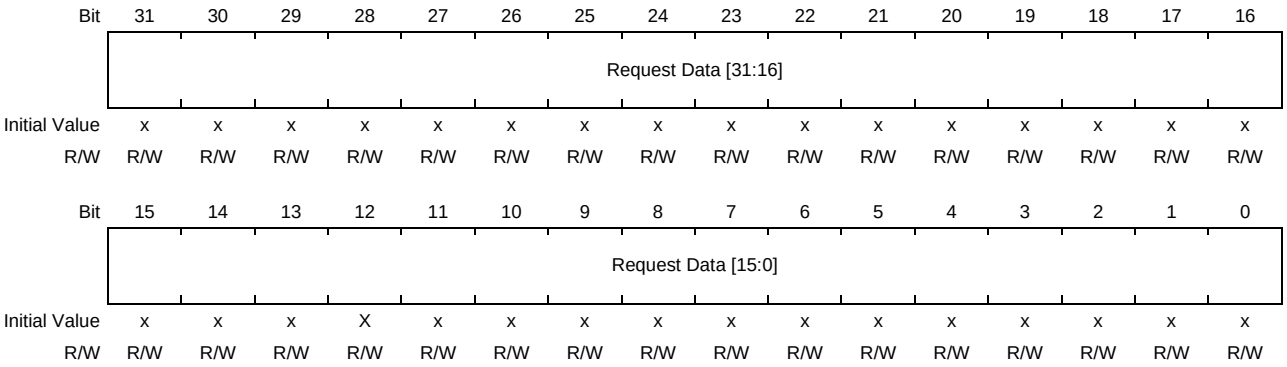
34.3 Register Description

34.3.1 AXI Bridge Register

The function of each register is described below. Register can only be accessed on the AXI bus.

34.3.1.1 Request Data Registers (Offset: H'80/ H'84/ H'88)

This register issues various Requests to PCIe.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Request Data	All x	R/W	Set Write Data when issuing a Request.

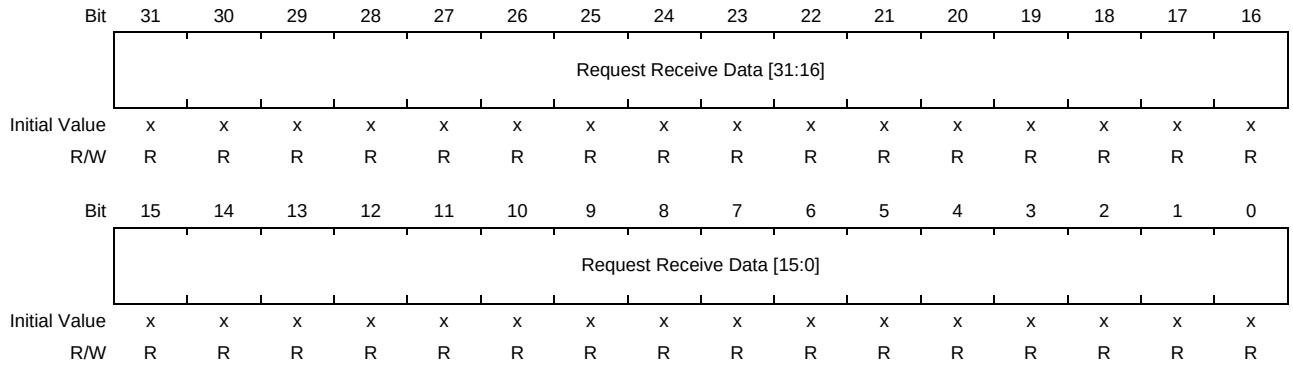
Table 34.7 Request Data Register 0, 1, 2

	Request Data Register 0 (Offset: H'80)	Request Data Register 1 (Offset: H'84)	Request Data Register 2 (Offset: H'88)
Zero-Length Read Request	Invalid	Invalid	Invalid
Configuration Write Type 0/1	Invalid	Invalid	Write Data
Configuration Read Type 0/1	Invalid	Invalid	Invalid
Message Request	3rd Header	4th Header	Invalid
Message Request with data payload	3rd Header	4th Header	Message Data

Note: The bits should be set to 0 for the requests indicated as "Invalid".

34.3.1.2 Request Receive Data Registers (Offset: H'8C)

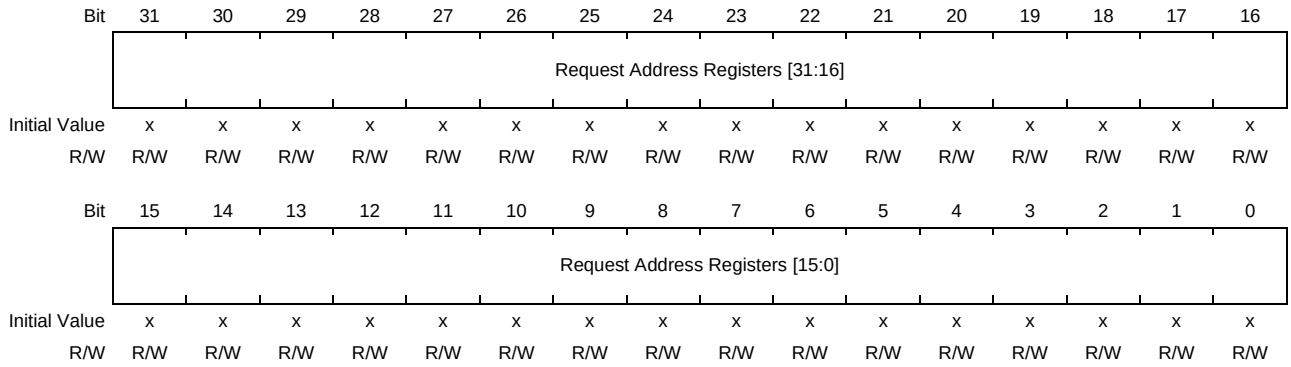
This register is set to read data when Completion is received after a Read Request is issued.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Request Receive Data	All x	R	Read data at Completion reception after Read Request issuance is set. However, it is invalid for Zero-Length Read and various write requests.

34.3.1.3 Request Address Registers 1 (Offset: H'90)

This register issues a Request to PCIe.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Request Address Registers	All x	R/W	Set Address when issuing a Request.

Table 34.8 Bit 31 - Bit 16 of Request Address Registers 1

Request Address Registers 1	[31:27]	[26:24]	[23:19]	[18:16]
Zero-Length Read Request	Address			
Configuration Read Type 0/1	Bus Number		Device Number	Function Number
Configuration Write Type 0/1	Bus Number		Device Number	Function Number
Message Request	Reserved	Routing Type	Reserved	Reserved
Message Request with data payload	Reserved	Routing Type	Reserved	Reserved

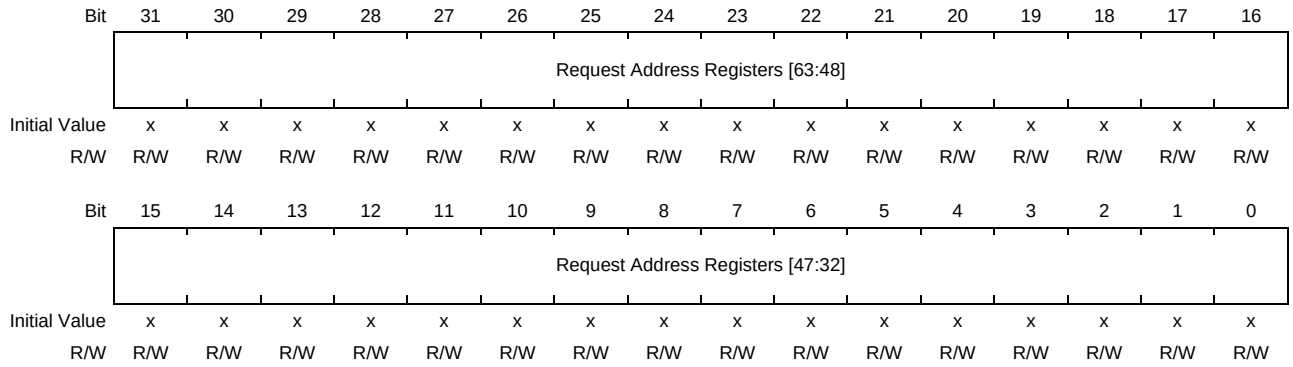
Table 34.9 Bit 15 - Bit 0 of Request Address Registers 1

Request Address Registers 1	[15:12]	[11:8]	[7:2]	[1:0]
Zero-Length Read Request	Address			Reserved
Configuration Read Type 0/1	Reserved	Ext. Reg. Number	Register Number	Reserved
Configuration Write Type 0/1	Reserved	Ext. Reg. Number	Register Number	Reserved
Message Request	Reserved	Reserved	Message Code	
Message Request with data payload	Reserved	Reserved	Message Code	

Note: The bits should be set to 0 for the requests indicated as "Reserved".

34.3.1.4 Request Address Registers 2 (Offset: H'94)

This register issues various Requests to PCIe.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Request Address Registers	All x	R/W	Set Address when issuing a Request.

Table 34.10 Request Address Registers 2

Request Address Registers 2	[31:0]
Zero-Length Read Request	Address
Configuration Read Type 0/1	Invalid
Configuration Write Type 0/1	Invalid
Message Request	Invalid
Message Request with data payload	invalid

Note: The bits should be set to 0 for the requests indicated as "Invalid".

34.3.1.5 Request Byte Enable Registers (Offset: H'98)

This register specifies First Byte (1st DW Byte) Enable in the TLP Header when issuing a Request to PCIe.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	Request Byte Enable			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

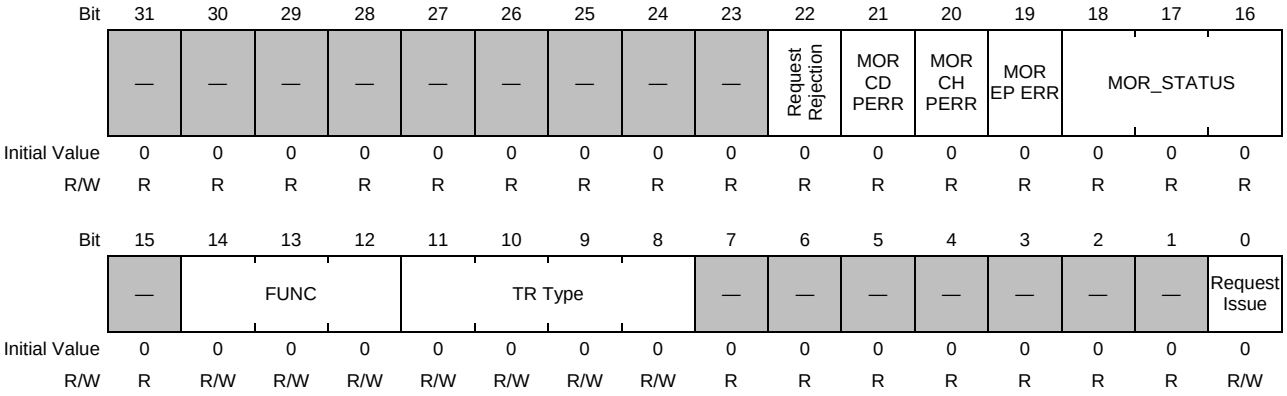
Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	Request Byte Enable	1111b	R/W	If necessary, specify Byte Enable when issuing a Cfg Request. Normally, use 1111b. 0: Byte Enable invalid 1: Byte Enable valid

Table 34.11 Bit 3 - Bit 0 of Request Byte Enable Registers

Request Byte Enable Registers	[3:0]
Zero-Length Read Request	0000b
Configuration Read Type 0/1	Optional (Usually 1111b)
Configuration Write Type 0/1	Optional (Usually 1111b)
Message Request	Invalid (1111b)
Message Request with data payload	Invalid (1111b)

34.3.1.6 Request Issue Registers (Offset: H'9C)

This register issues a Request to PCIe.



Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	Request Rejection	0b	R	PCI Express indicates that a stop or hibernation was detected on the sending (TX side) and processing was killed. 0: Normal state (Issue Request) 1: Forced termination (Rejection)
21	MOR CD PERR	0b	R	1 is set when a data error occurs in Completion TLP for a Non-Posted Request issued in this register. Usually do not use it. It is not updated during Posted Requests.
20	MOR CH PERR	0b	R	This is set if a header error occurs in the Completion TLP for a Non-Posted Request issued in this register. Usually do not use it. It is not updated during Posted Requests.
19	MOR EP ERR	0b	R	1 is set when the Poisoned Completion TLP for the Non-Posted Request issued in this register is received. Usually do not use it. It is not updated during Posted Requests.
18 to 16	MOR_STATUS	000b	R	Holds the MOR Status of the Completion TLP for Non-Posted Requests issued in this register. It is not updated during Posted Requests. 000b: Successful Completion (SC) 001b: Unsupported Request (UR) 010b: Configuration Request Retry Status (CRS) (Not Support) 011b: Completion Timeout 100b: Completer Abort (CA) 101b: Unexpected Completion and mismatched type (LockCompletion respond to non-Lock Request) 110b: Reserved 111b: Overrun Completion length
15	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14 to 12	FUNC	000b	R/W	Set the Function of Request.
11 to 8	TR Type	0000b	R/W	Set the Type of Request.
7 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

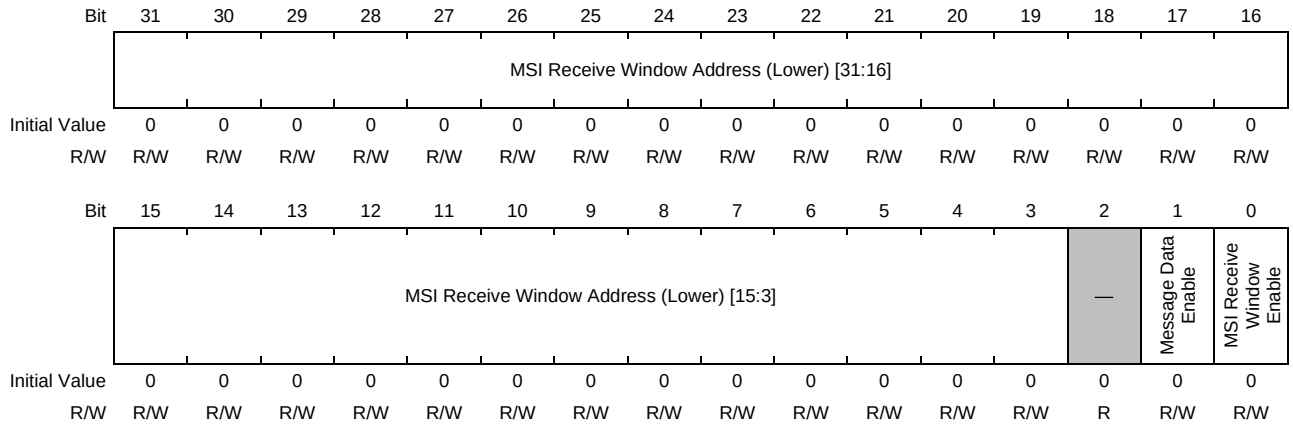
Bit	Bit Name	Initial Value	R/W	Description
0	Request Issue	0b	R/W	[Write] 1: Request 0: No Operation [Read] 1: Processing Request (Indicates that the issued Request is being processed.) 0: Request can be issued and accepted (This indicates that the issued Request has been processed.)

Table 34.12 TR Type of Request Byte Enable Registers

Request Byte Enable Registers	TR Type [11:8]	Posted/Non-Posted	Root Complex
Zero-Length Read Request	0000b (H'0)	Non-Posted	Issuable
Configuration Read Type 0	0100b (H'4)	Non-Posted	Issuable
Configuration Write Type 0	0101b (H'5)	Non-Posted	Issuable
Configuration Read Type 1	0110b (H'6)	Non-Posted	Issuable
Configuration Write Type 1	0111b (H'7)	Non-Posted	Issuable
Message Request	1000b (H'8)	Posted	Issuable
Message Request with data payload	1001b (H'9)	Posted	Issuable
—	Others	—	Issuing prohibited

34.3.1.7 MSI Receive Window Address (Lower) Registers (Offset: H'100)

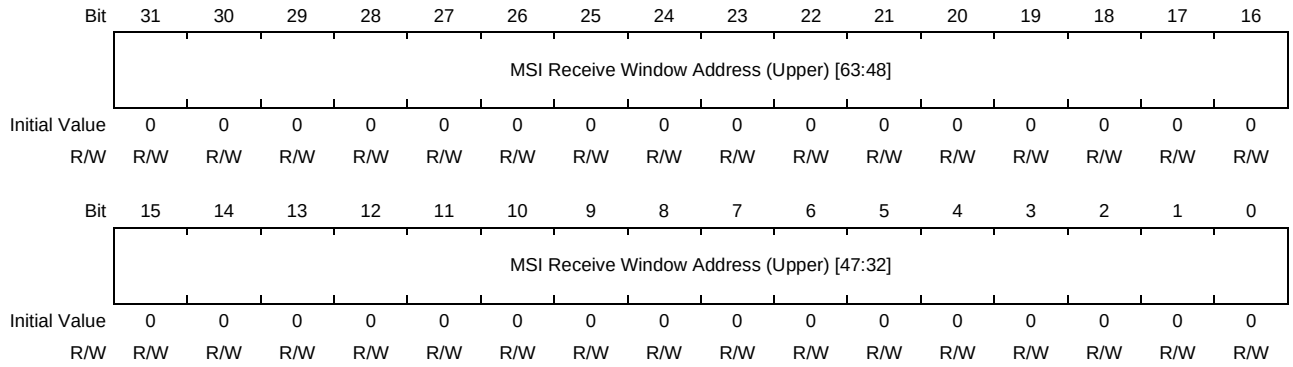
This register sets the start address in the Memory space of the receiving MSI.



Bit	Bit Name	Initial Value	R/W	Description
31 to 3	MSI Receive Window Address (Lower) [31:3]	All 0	R/W	Set the MSI receiving window's Start Address [31:3]. However, they must be aligned to the size set by the MSI Receive Window Mask. Even if an unaligned Address is set, the Address Bit set with the MSI Receive Window Mask will be 0. *Note*: To change this register, make sure that MSI Receive Window Enable is set to 0b.
2	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	Message Data Enable	0b	R/W	Controls whether Message Data is used in MSI reception decisions. 0: Unused (conventional mode) 1: Used (multiple interrupts) The MSI reception judgment result selected in this register is reflected in the INTMSI_RC terminal.
0	MSI Receive Window Enable	0b	R/W	Enable setting for the MSI Receive Window. 0: Window is disabled 1: Window is enabled

34.3.1.8 MSI Receive Window Address (Upper) Registers (Offset: H'104)

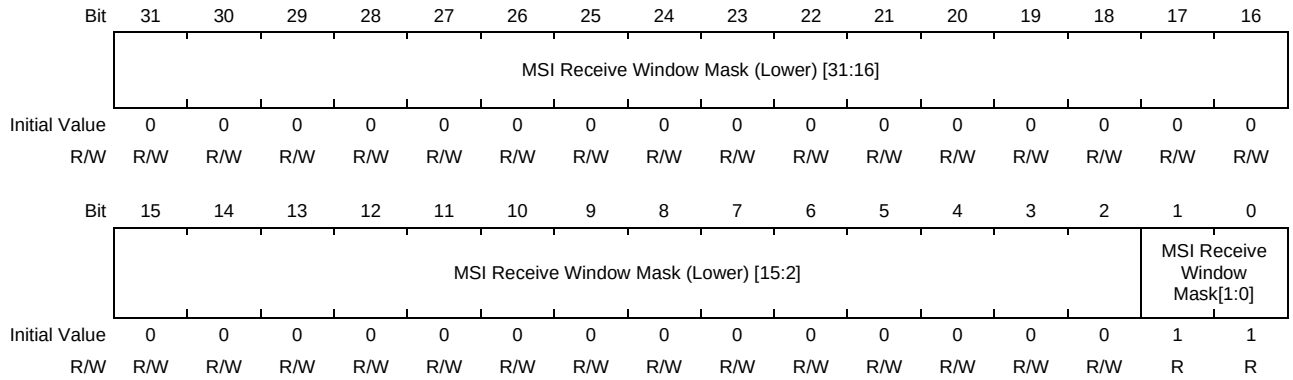
This register sets the start address in the Memory space of the receiving MSI.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MSI Receive Window Address (Upper) [63:32]	All 0	R/W	Set the MSI receiving window's Start Address [31:3]. However, they must be aligned to the size set by the MSI Receive Window Mask. Even if an unaligned Address is set, the Address Bit set with the MSI Receive Window Mask will be 0. *Note*: To change this register, make sure that MSI Receive Window Enable is set to 0b.

34.3.1.9 MSI Receive Window Mask Registers (Lower) (Offset: H'108)

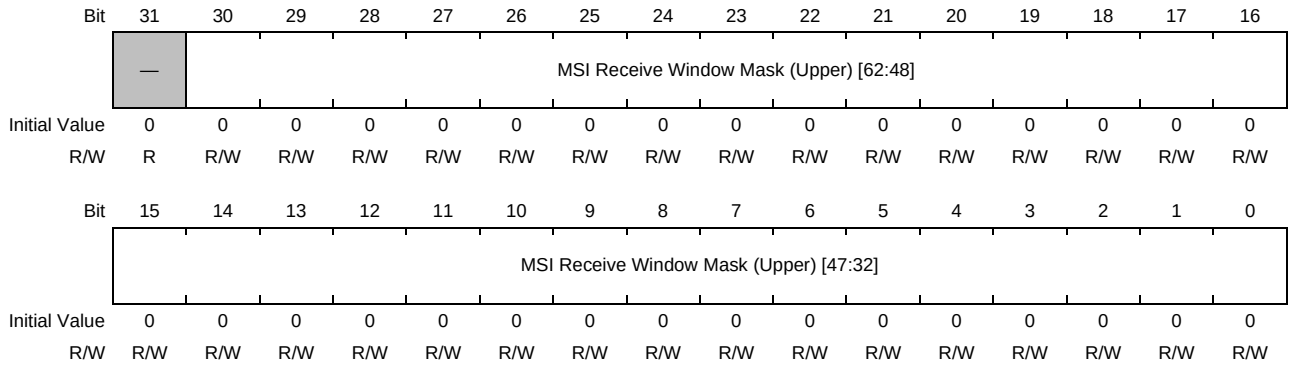
This register shows the size of the area from the address set by the MSI Receive Window Address.



Bit	Bit Name	Initial Value	R/W	Description
31 to 2	MSI Receive Window Mask (Lower) [31:2]	All 0	R/W	Set the lower bit of the reserved area to "1". (Min4byte, Max4Gbyte free space) The following is a sample configuration. H'0000_0000: 4 bytes free space H'0000_0001: 8 bytes free space H'0000_007F: 512 bytes free space *Note*: To change this register, make sure that MSI Receive Window Enable is set to 0b.
1 ,0	MSI Receive Window Mask [1:0]	11b	R	11b fixed. *Note*: MSI Receive Window Mask [1:0] assumes that Mask is always set.

34.3.1.10 MSI Receive Window Mask Registers (Upper) (Offset: H'10C)

This register shows the size of the area from the address set by the MSI Receive Window Address.



Bit	Bit Name	Initial Value	R/W	Description
31	—	0b	R	0b fixed. *Note*: MSI Receive Window Mask of this register [63] is fixed at 0
30 to 0	MSI Receive Window Mask (Upper) [62:32]	All 0	R/W	(For setting up more than 4 Gbytes space) Set the MSI Receive Window Mask to [62:32]

34.3.1.11 PCI INTx Receive Interrupt Enable Registers (Offset: H'110)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	MSI Receive Interrupt Enable	INTD Receive Interrupt Enable	INTC Receive Interrupt Enable	INTB Receive Interrupt Enable	INTA Receive Interrupt Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	MSI Receive Interrupt Enable	0b	R/W	This register permit INTMSI_RC with MSI reception. 0: Do not allow 1: Allow
3	INTD Receive Interrupt Enable	0b	R/W	This register permit INTD_RC. 0: Do not allow 1: Allow
2	INTC Receive Interrupt Enable	0b	R/W	This register permit INTC_RC. 0: Do not allow 1: Allow
1	INTB Receive Interrupt Enable	0b	R/W	This register permit INTB_RC. 0: Do not allow 1: Allow
0	INTA Receive Interrupt Enable	0b	R/W	This register permit INTA_RC. 0: Do not allow 1: Allow

34.3.1.12 PCI INTx Receive Interrupt Status Registers (Offset: H'114)

In case of Assert_INTx is received from PCIe to Msg_Req, set the corresponding interrupt bit in this register and assert interrupt INTx_RC. Also, in case of Deassert_INTx is received to Meg Req, clear the corresponding interrupt register (bit) and de-assert INTx_RC. This register can do RW1C by S/W, but PCIe does not recommend clearing this interrupt bit by S/W itself during normal operation for interrupts on the bus. In case of each factor is detected on this field of register, it is always set regardless of the PCI_INTx_Receive_Interrupt_Enable register for RC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	MSI Receive Interrupt Enable	INTD Receive Interrupt Enable	INTC Receive Interrupt Enable	INTB Receive Interrupt Enable	INTA Receive Interrupt Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C

Bit	Bit Name	Initial Value	R/W	Description
31:5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	MSI Receive Interrupt Status	0b	R/W1C	Set when a Memory Write Req is received from the PCI bus side in the MSI receive window set area. This bit is set when an MSI is issued to AXI as a Write Transaction and a Response is returned.
3	INTD Receive Interrupt Status	0b	R/W1C	Set on receipt of Assert INTD Message. It is cleared upon receipt of a Deassert INTD Message. Or 1 write to clear it.
2	INTC Receive Interrupt Status	0b	R/W1C	Set on receipt of Assert INTD Message. It is cleared upon receipt of a Deassert INTC Message. Or 1 write to clear it.
1	INTB Receive Interrupt Status	0b	R/W1C	Set on receipt of Assert INTD Message. It is cleared upon receipt of a Deassert INTB Message. Or 1 write to clear it.
0	INTA Receive Interrupt Status	0b	R/W1C	Set on receipt of Assert INTD Message. It is cleared upon receipt of a Deassert INTA Message. Or 1 write to clear it.

34.3.1.13 Message Receive Interrupt Enable Registers (Offset: H'120)

Controls the enabling of MSG_INT upon receipt of a non-INTx, Error-related Message Request.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	Message Receive Interrupt Enable	—	—	—	—	PM_Active_State_Nak Receive Interrupt Enable	PM_PME Receive Interrupt Enable	PME Turn_Off Receive Interrupt Enable	PME TO_Ack Receive Interrupt Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	Message Receive Interrupt Enable	0b	R/W	Enabling the MSG_INT Assert by Receiving a Message/ 0: Assert disallowed 1: Assert allowed
23 to 20	—	0000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	PM_Active_State_Nak Receive Interrupt Enable	0b	R/W	Enabling the MSG_INT Assert by Receiving a PM_Active_State_Nak. 0: Disable 1: Enable
18	PM_PME Receive Interrupt Enable	0b	R/W	Enabling the MSG_INT Assert by Receiving a PM_PME.
17	PME_Turn_Off Receive Interrupt Enable	0b	R/W	Enabling the MSG_INT Assert by Receiving a PME_Turn_Off. 0: Disable 1: Enable
16	PME_TO_Ack Receive Interrupt Enable	0b	R/W	Enabling the MSG_INT Assert by Receiving a PME_TO_Ack.
15 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

34.3.1.14 Message Receive Interrupt Status Registers (Offset: H'124)

This register is a status register indicating Message Request reception other than INTx, Error related. It is reflected in MSG_INT. Only the Message Code is used to determine the type of message, the validity of the Routing and the validity of the choice of Msg/MsgD are not verified, and the Message in question is considered to have been received.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	Message Receive Interrupt	—	—	—	—	PM_Active_State_Nak Receive Interrupt	PM_PME Receive Interrupt	PME_Turn_Off Receive Interrupt	PME_TO_Ack Receive Interrupt
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W1C	R	R	R	R	R/W1C	R/W1C	R/W1C	R/W1C

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	Message Receive Interrupt	0b	R/W1C	Set for receiving Message (Independent of Message type)
23 to 20	—	0000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	PM_Active_State_Nak Receive Interrupt	0b	R/W1C	Invalid due to RC
18	PM_PME Receive Interrupt	0b	R/W1C	Set when PM_PME Message is received.
17	PME_Turn_Off Receive Interrupt	0b	R/W1C	Invalid due to RC
16	PME_TO_Ack Receive Interrupt	0b	R/W1C	Set when PM_TO_Ack Message is received.
15 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

34.3.1.15 Message Code Registers (Offset: H'130)

Stores the Code, Routing of the last received Message.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Message Code								Routing			—	—	—	—	Message Payload
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

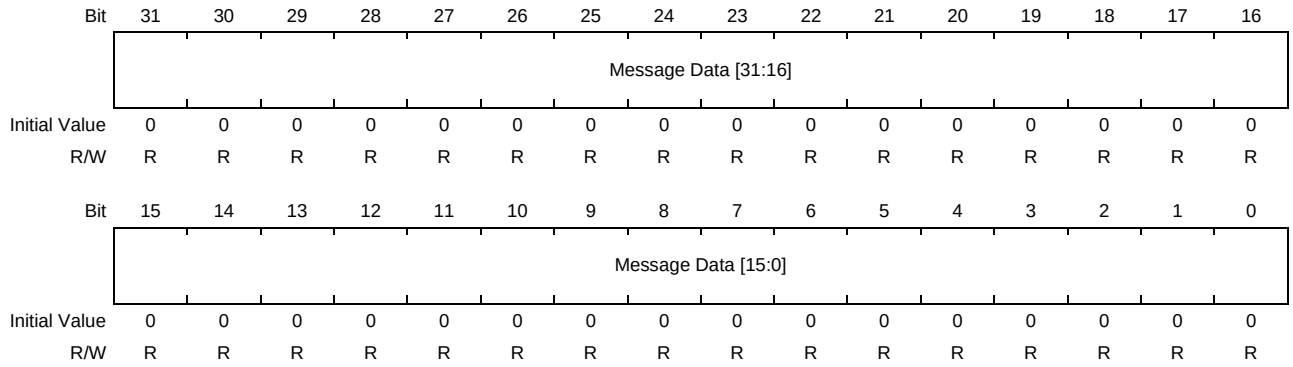
Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 8	Message Code	All 0	R	Stores the Code of the last Message received.
7 to 5	Routing	000b	R	Stores the Routing of the last Message received.
4 to1	—	0000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	Message Payload	0b	R	Stores whether there is a data payload for the last Message received. 0: Msg (Without Payload) 1: MsgD (with Payload)

NOTE

This register is not written when Power Management Message (PME_TO_Ack Message, PME_Turn_Off Message, PM_PME Message, PM_Active_State_Nak Message) is received.

34.3.1.16 Message Data Registers (Offset: H'134)

Stores the Data of the last received Message.



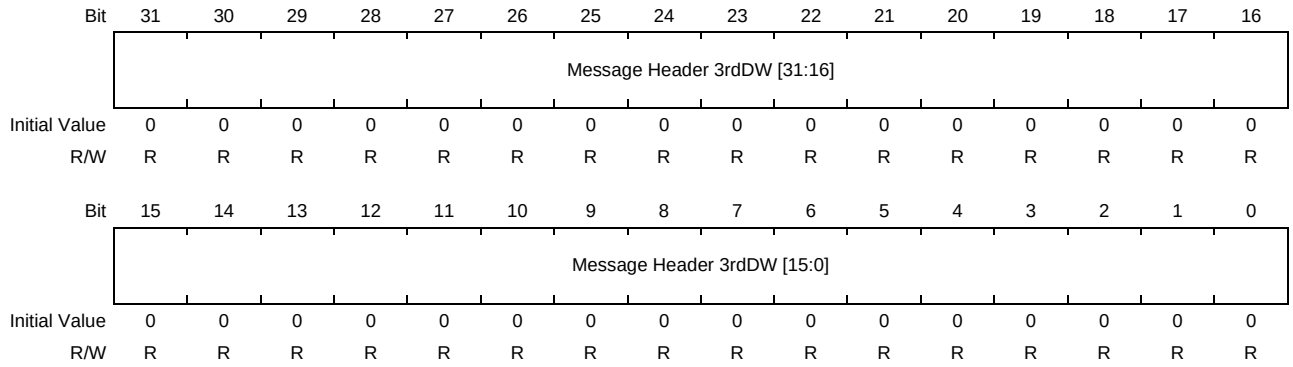
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Message Data	All 0	R	Stores the first DW Data of the last received Message. It is only updated when MsgD (with Data) is received, and the previous value is retained when Msg (without Data) is received.

NOTE

This register is not written when Power Management Message (PME_TO_Ack Message, PME_Turn_Off Message, PM_PME Message, PM_Active_State_Nak Message) is received.

34.3.1.17 Message Header 3rdDW Registers (Offset: H'138)

Stores the Header (3rdDW) of the last received Message.



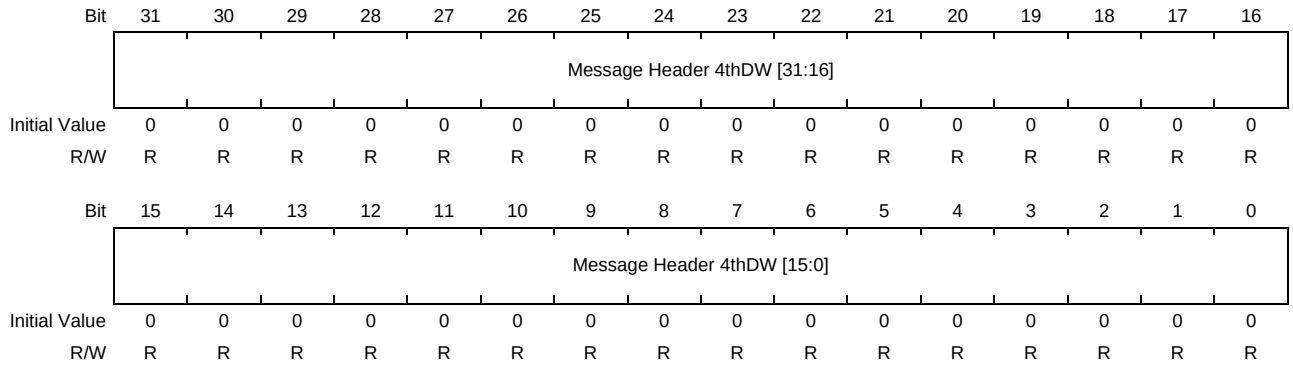
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Message Header 3rdDW	All 0	R	Stores the Header (3rdDW) of the last received Message.

NOTE

This register is not written when Power Management Message (PME_TO_Ack Message, PME_Turn_Off Message, PM_PME Message, PM_Active_State_Nak Message) is received.

34.3.1.18 Message Header 4thDW Registers (Offset: H'13C)

Stores the Header (4thDW) of the last received Message.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Message Header 4thDW	All 0	R	Stores the Header (4thDW) of the last received Message.

NOTE

This register is not written when Power Management Message (PME_TO_Ack Message, PME_Turn_Off Message, PM_PME Message, PM_Active_State_Nak Message) is received.

34.3.1.19 Interrupt Table Registers (Offset: H'140)

This register is the Index of the interrupt factor.

Interrupt signal (Active: High) status can be monitored by category in a list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	AXI_ERR_INT	PCIE_EVT_INT	MSG_INT	—	—	—	INTMSI_RC	INTD_RC	INTC_RC	INTB_RC	INTA_RC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
14 to 12	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
10	AXI_ERR_INT	0b	R	Error Interrupt signal Monitor･Register
9	PCIE_EVT_INT	0b	R	Event Interrupt signal Monitor･Register
8	MSG_INT	0b	R	Message Interrupt signal Monito･Register
7 to 5	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	INTMSI_RC	0b	R	INT_MSI receive interrupt signal Monitor Register
3	INTD_RC	0b	R	INTD_RC receive interrupt signal Monitor Register
2	INTC_RC	0b	R	INTC_RC receive interrupt signal Monitor Register
1	INTB_RC	0b	R	INTB_RC receive interrupt signal Monitor Register
0	INTA_RC	0b	R	INTA_RC receive interrupt signal Monitor Register

NOTE

For details on the register, refer to the description of the terminal of the same name.

34.3.1.20 PCIe Event Interrupt Enable 0 Registers (Offset: H'200)

This register is for interrupt permission due to various event factors in PCI Express. Write permission to PCIe Event Interrupt Status 0 Register (Offset: H'204). For more information about the factors, see the Status Register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	UI_LINK_WIDTH_CHANGE_DONE EN	UI_LINK_SPEED_CHANGE_DONE EN	Request Done EN	—	—	—	CA EN	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R

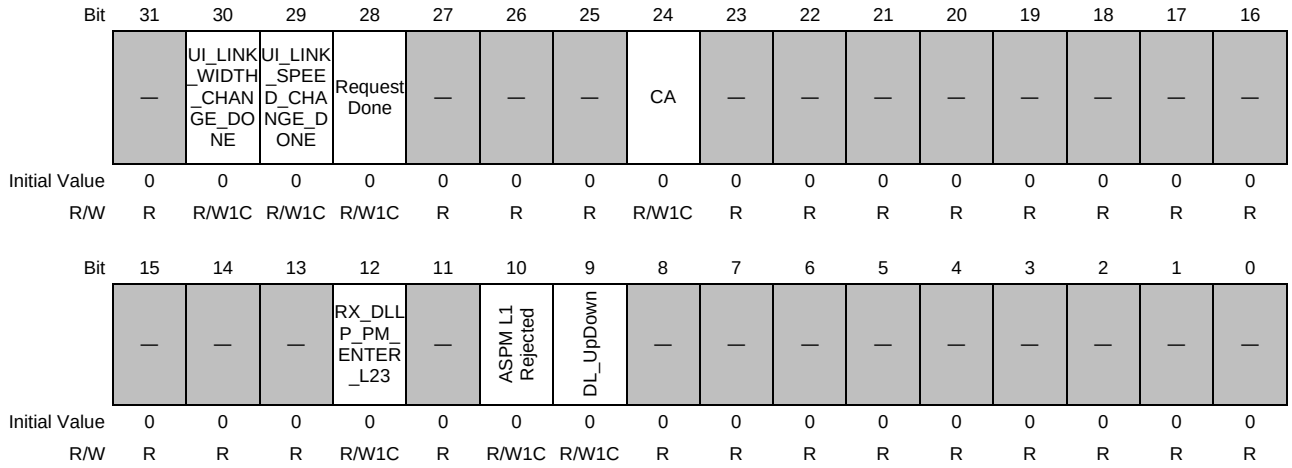
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	RX_DLLP_PM_ENTER_L23 EN	—	ASPM L1 Rejected EN	DL_UpDown EN	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R/W	R/W	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30	UI_LINK_WIDTH_CHANGE_DONE EN	0b	R/W	Up/Down Configure Complete Interrupt Enabled. 0: Disable 1: Enable
29	UI_LINK_SPEED_CHANGE_DONE EN	0b	R/W	Speed Change Complete Interrupt Enabled. 0: Disable 1: Enable
28	Request Done EN	0b	R/W	Request Complete interrupt enable. 0: Disable 1: Enable
27, 26	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
24	CA EN	0b	R/W	CA (Completer Abort) Interrupt Enabled. 0: Disable 1: Enable
23, 22	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written
21	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20 to 14	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
13	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	RX_DLLP_PM_ENTER_L23 EN	0b	R/W	RX_DLLP_PM_ENTER_L23 Interrupt Enabled. 0: Disable 1: Enable

Bit	Bit Name	Initial Value	R/W	Description
11	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
10	ASPM L1 Rejected EN	0b	R/W	ASPM L1 Rejected Interrupt Enabled. 0: Disable 1: Enable
9	DL_UpDown EN	0b	R/W	Interrupt enabled due to DL state change. 0: Disable 1: Enable
8	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 4	—	0000b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3 to 1	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

34.3.1.21 PCIe Event Interrupt Status 0 Registers (Offset: H'204)

This register is a status register used to indicate various events in PCI Express. It is set to 1b by the occurrence of factors in the table. After checking the factors, please do 1 write clear.



Bit	Bit Name	Initial Value	R/W	Description
31	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30	UI_LINK_WIDTH_CHANGE_DONE	0b	R/W1C	Indicates Up/Down Configure operation complete.
29	UI_LINK_SPEED_CHANGE_DONE	0b	R/W1C	Indicates Speed Change operation complete.
28	Request Done	0b	R/W1C	In response to the request issued in Request Issue Registers (Offset: 9Ch): Non-Posted: Indicates completion received. Posted: Indicates that the request has been issued.
27, 26	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
24	CA	0b	R/W1C	Indicates that the opposing device responded with a CA (Completer Abort).
23, 22	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
21	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20 to 14	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
13	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	RX_DLLP_PM_ENTER_L23	0b	R/W1C	Indicates transition to L2/L3 State in Power Management control.

Bit	Bit Name	Initial Value	R/W	Description
11	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
10	ASPM L1 Rejected	0b	R/W1C	Indicates that the ASPM L1 transition was rejected.
9	DL_UpDown	0b	R/W1C	Set to 1b when the state changes from DL_Down to DL_Up or from DL_Up to DL_Down.. Check the DL_Down/DL_Up status with the PCIe Core Status 1 Registers (Offset: H'408).
8	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

34.3.1.22 PCIe Event Interrupt Enable 1 Registers (Offset: H'208)

This register is the Parity Error/ECC Error interrupt grant register. When each bit is set to allow, each corresponding status bit value in the PCIe Event Interrupt Status 1 Register (Offset: 20 Ch) takes effect.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TXB_PARITY_ERR EN	ERR_RPC_REPLAYFIFO_PERR EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ERR_REPLAY_UPPER_CORRECTABLE_ERROR EN	ERR_REPLAY_LOWER_CORRECTABLE_ERROR EN	—	—	—	—	—	—	ERR_REPLAY_UPPER_UNCORRECTABLE_ERROR EN	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	TXB_PARITY_ERR EN	0b	R/W	TXB_PARITY_ERR allow to interrupt. Parity error interrupt notification permission setting for TX Buffer included in Transaction Layer 0: Do not allow 1: Allow
16	ERR_RPC_REPLAYFIFO_PERR EN	0b	R/W	ERR_RPC_REPLAYFIFO_PERR allow to interrupt. Parity error interrupt notification permission setting for Replay FIFO included in Data Link Layer 0: Do not allow 1: Allow
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	ERR_REPLAY_UPPER_CORRECTABLE_ERROR EN	0b	R/W	ERR_REPLAY_UPPER_CORRECTABLE_ERROR allow to interrupt. Interrupt notification permission settings when an ECC 1 bit error (Correctable Error) occurs in the upper 64bit data bus of the Replay FIFO installed in the Data Link Layer 0: Do not allow 1: Allow
8	ERR_REPLAY_LOWER_CORRECTABLE_ERROR EN	0b	R/W	ERR_REPLAY_LOWER_CORRECTABLE_ERROR allow to interrupt. Interrupt notification permission settings when an ECC 1 bit error (Correctable Error) occurs in the lower 64bit data bus of the Replay FIFO installed in the Data Link Layer 0: Do not allow 1: Allow
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	ERR_REPLAY_UPPER_UNCORRECTABLE_ERROR EN	0b	R/W	ERR_REPLAY_UPPER_UNCORRECTABLE_ERROR allow to interrupt. Interrupt notification permission setting when error (correctable error) of ECC 2 bit or more occurs in upper 64bit data bus of replay FIFO installed in data link layer r 0: Do not allow 1: Allow

Bit	Bit Name	Initial Value	R/W	Description
0	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROREN	0b	R/W	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR allow to interrupt. Interrupt notification permission setting when error (correctable error) of ECC 2 bit or more occurs in lower 64bit data bus of replay FIFO installed in data link layer 0: Do not allow 1: Allow

34.3.1.23 PCIe Event Interrupt Status 1 Registers (Offset: H'20C)

This register is a status register that indicates parity error and ECC error interrupts. After checking the factor, write 1b to the corresponding bit to clear it.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TXB_PARITY_ERR	ERR_RPC_REPLAYFIFO_PERR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W1C	R/W1C

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ERR_REPLAY_UPPER_CORRECTABLE_ERROR	ERR_REPLAY_LOWER_CORRECTABLE_ERROR	—	—	—	—	—	—	ERR_REPLAY_UPPER_UNCORRECTABLE_ERROR	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W1C	R/W1C	R	R	R	R	R	R	R/W1C	R/W1C

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	TXB_PARITY_ERR	0b	R/W1C	TXB_PARITY_ERR interrupt Parity error interrupt notification for TX Buffer, which is included in the Transaction Layer
16	ERR_RPC_REPLAYFIFO_PERR	0b	R/W1C	ERR_RPC_REPLAYFIFO_PERR interrupt Parity error interrupt notification for Replay FIFO, which is included in the Data Link Layer
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	ERR_REPLAY_UPPER_CORRECTABLE_ERROR	0b	R/W1C	ERR_REPLAY_UPPER_CORRECTABLE_ERROR interrupt Interrupt notification permission settings when an ECC 1 bit error (Correctable Error) occurs in the upper 64bit data bus of the Replay FIFO installed in the Data Link Layer
8	ERR_REPLAY_LOWER_CORRECTABLE_ERROR	0b	R/W1C	ERR_REPLAY_LOWER_CORRECTABLE_ERROR interrupt Interrupt notification permission settings when an ECC 1 bit error (Correctable Error) occurs in the lower 64bit data bus of the Replay FIFO installed in the Data Link Layer
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	ERR_REPLAY_UPPER_UNCORRECTABLE_ERROR	0b	R/W1C	ERR_REPLAY_UPPER_UNCORRECTABLE_ERROR interrupt Interrupt notification permission setting when error (correctable error) of ECC 2 bit or more occurs in upper 64bit data bus of replay FIFO installed in data link layer
0	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR	0b	R/W1C	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR interrupt Interrupt notification permission setting when error (correctable error) of ECC 2 bit or more occurs in lower 64 bit data bus of replay FIFO installed in data link layer

34.3.1.24 AXI Master Error Interrupt Enable Registers (Offset: H'210)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	Write MSTERR INT EN				—	—	—	—	Read MSTERR INT EN			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	Write MSTERR INT EN	All 0	R/W	Write MSTERR INT Permit Each corresponding bit can be turned on or off individually. 0: Disable 1: Enable
7 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	Read MSTERR INT EN	All 0	R/W	Read MSTERR INT Permit Each corresponding bit can be turned on or off individually. 0: Disable 1: Enable

34.3.1.25 AXI Master Error Interrupt Status Registers (Offset: H'214)

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 24	Write ERR ID	All 0	R	Saves the ID of the first DECERR/SLVERR received. When bit [11:8] is cleared, a new error ID can be saved. H'0: Normal Access
23 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19 to 16	Read ERR ID	All 0	R	Saves the ID of the first DECERR/SLVERR received. When bit [3:0] is cleared, a new error ID can be saved. H'0: Normal Access
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	Write MSTERR INT	All 0	R/W1C	The AXI Master Port indicates that an error has been detected. Only the first error detected will be saved, and a new error will be ready to save if bit [11:8] is cleared Bit 11: Length Error In case of discrepancy in length of data sent by TEF and data channel. Bit 10: ID Inconsistency In case of the MBID value received in the MAWID response channel is different. Bit 9: In case of receiving DECERR Bit 8: In case of receiving SLVERR
7 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	Read MSTERR INT	All 0	R/W1C	The AXI Master Port indicates that an error has been detected. Only the first error detected will be saved, and a new error will be ready to save if bit [3:0] is cleared Bit 3: Length Error In case of discrepancy in length of data sent by TER and data channel. Bit 2: ID Inconsistency In case of the MRID value received in the MARID response channel is different. Bit 1: In case of receiving DECERR Bit 0: In case of receiving SLVERR

34.3.1.26 AXI Slave Error Interrupt Enable 1 Registers (Offset: H'220)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	Write SLVERR INT EN				—	—	—	—	—	—	Read SLVERR INT EN	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	Write SLVERR INT EN	All 0	R/W	Write SLVERR INT permit Each corresponding bit can be turned on or off individually. 0: Disable 1: Enable
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1 to 0	Read SLVERR INT EN	All 0	R/W	Read SLVERR INT permit Each corresponding bit can be turned on or off individually. 0: Disable 1: Enable

34.3.1.27 AXI Slave Error Interrupt Status 1 Registers (Offset: H'224)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	Write SLVERR INT				—	—	—	—	—	—	Read SLVERR INT	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W1C	R/W1C	R/W1C	R/W1C	R	R	R	R	R	R	R/W1C	R/W1C

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	Write SLVERR INT	All 0	R/W1C	An AXI Slave Port has detected an unrecoverable error. (The response to the Transaction is SLVERR.) Bit 11: Burst length error In case of the burst length of data received by SAWLEN and the data channel does not match. Bit 10: ID Mismatch In case of the value of the SWID received in the SAWID and data channel is different. Bit 9: Burst type Invalid In case of SAWBURST is b 11 (undefined). In case of SAWBURST is b 10 (wrapping) and burst length is other than 2, 4, 8 or 16. Bit 8: Data size Invalid In case of SAWSIZE is between b 100 and b 111 (over AXI Bus width/not supported). <Each bit means> 0: Error Not Found 1: Error detection
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	Read SLVERR INT	00b	R/W1C	An AXI Slave Port has detected an unrecoverable error. (The response to the Transaction is SLVERR.) Bit 1: Burst type Invalid In case of SARBURST is b11(Undefined) In case of SARBURST is b 10 (wrapping) and burst length is other than 2, 4, 8 or 16. Bit 0: Burst type Invalid In case of SARSIZE is (over AXI Bus width/not supported). <Each bit means> 0: Error Not Found 1: Error detection

34.3.1.28 AXI Slave Error Interrupt Status 3 Registers (Offset: H'230)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ERR ID [31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERR ID [15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ERR ID	All 0	R	Stores the ID of the first AXI Slave Error Interrupt Status 1 register error (Offset H'224). Only the first error detected is saved, and a new error ID can be saved when bits 11:8 and 1:0 of the AXI Slave Error Interrupt Status 1 register are all cleared.

34.3.1.29 Permission Registers (Offset: H'300)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CFG_HWINIT_EN	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	CFG_HWINIT_EN	0b	R/W	It is output to the CFG_HWINIT_EN terminal. It is sed for register access control in the CFGU.
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

34.3.1.30 Mode Set 0 Registers (Offset: H'314)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	AWPROT			AWCACHE_L				—	—	AWLOCK		AWCACHE_D			
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ARPROT			—	—	—	—	—	—	ARLOCK		ARCACHE			
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 28	AWPROT	010b	R/W	Set the protection type from PCIe to AXI transactions. These bits indicate whether the protection level of the transaction is normal, privileged, or secure, and whether the transaction is for data access or instruction access. [2] 1b: Instruction Access 0b: Data Access [1] 1b: Non-secure access 0b: Secure access [0] 1b: Privileged Access 0b: Normal Access
27 to 24	AWCACHE_L	All 0	R/W	Indicate the value of MAWCACHE [3: 0] to be issued to AXI. This setting is printed when issuing an AXI request that contains the last byte. *Note*: The recommended value is 0000b. [3] 1b: Write assignment possible 0b: Write assignment possible [2] 1b: Read assignment possible 0b: Read assignment impossible [1] 1b: Cacheable 0b: Not Cacheable [0] 1b: Buffable 0b: Not buffered
23, 22	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21, 20	AWLOCK	00b	R/W	Lock type in PCIe to AXI transactions. This signal provides information about the atomic nature of the transfer. 00b: Normal access 01b: Exclusive access 10b: Lock access 11b: Reserved
19 to 16	AWCACHE_D	0001b	R/W	Indicates the value of MAWCACHE [3: 0] to be issued to AXI. This setting is output when an AXI request other than the AWCACHE_L output condition is issued. *Note*: The recommended value is 0000b. [3] 1b: Write assignment possible 0b: Write assignment possible [2] 1b: Read assignment possible 0b: Read assignment impossible [1] 1b: Cacheable 0b: Not Cacheable [0] 1b: Buffable 0b: Not buffered
15	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
14 to 12	ARPROT	010b	R/W	Sets the protection type from PCIe to AXI transactions. This bit indicates whether the protection level of the transaction is normal, privileged, or secure, and whether the transaction is for data access or instruction access. [2] 1b: Instruction Access 0b: Data Access [1] 1b: Non-secure access 0b: Secure access [0] 1b: Privileged Access 0b: Normal Access
11 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5, 4	ARLOCK	00b	R/W	Lock type in PCIe to AXI transactions. This bit provides information about the atomic nature of the transfer. 00b: Normal access 01b: Exclusive access 10b: Lock access 11b: Reserved
3 to 0	ARCACHE	All 0	R/W	Cache type in PCIe to AXI transactions. This bit indicates whether the transaction is “buffered”, “cacheable”, “write-through”, “write-back”, or “assign” as an attribute.

34.3.1.31 Mode Set 1 Registers (Offset: H'318)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AXI Max Issue Write				AXI Max Issue Read				AXI Master Max Burst				—	—	RAM Parity Enable	PCIe Request Order
Initial Value	0	0	1	1	0	0	1	1	1	1	1	1	0	0	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 12	AXI Max Issue Write	0011b	R/W	Set the number of write issuable AXI Master. Set as much as Interconnect allows. 0000b: Number 1 0001b: Number 2 ⋮ 1111b: Number 16
11 to 8	AXI Max Issue Read	0011b	R/W	Set the number of read issuable AXI Master. Set as much as Interconnect allows. 0000b: Number 1 0001b: Number 2 ⋮ 1111b: Number 16
7 to 4	AXI Master Max Burst	1111b	R/W	Set the maximum burst length as an AXI Master operation.
3, 2	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RAM Parity Enable	1b	R/W	Set whether to check the internal SRAM parity. The Default value is ignored for macros that are Enable but do not have Parity. 0: RAM Parity Check disabled 1: RAM Parity Check enabled
0	PCIe Request Order	0b	R/W	Issue read requests to PCIe from the same AXI master without waiting for completion. Set it to 1 if the order of Request to Completer should be strictly followed. 0: Do not wait for completion. 1: Wait for completion.

34.3.1.32 Mode Set 3 Registers (Offset: H'380)

Outputs the value set as the ASPM L1 Idle Time bit.

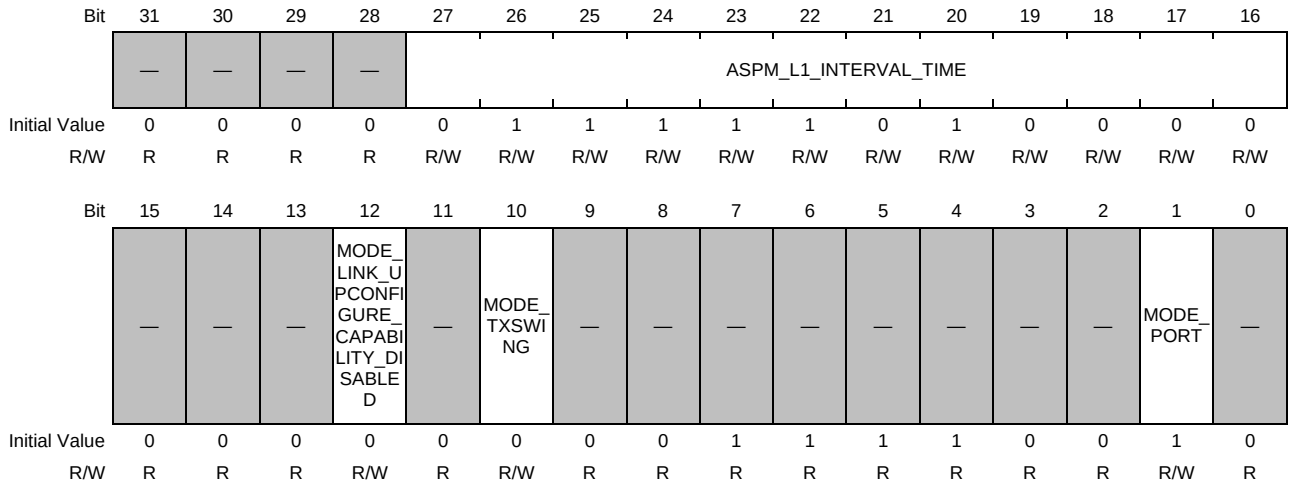
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ASPM L1 Idle Time							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7 to 0	ASPM L1 Idle Time	All 0	R/W	Sets the idle period for AXI transactions that the macro checks during the ASPM L1 transition. One condition of the ASPM L1 transition is that the idle period is confirmed by adding H'FF to the lower 8 bits to the 8 bits set by this bit for several cycles. H'00: 256 [ACLK] H'01: 512 [ACLK] ⋮ H'FF: 65536 [ACLK]

34.3.1.33 PCIe Core Mode Set 1 Registers (Offset: H'400)

This register is used to set the operating mode of the PCI Express core.



Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29, 28	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
27 to 16	ASPM_L1_INTERVAL_TIME	H'7D0	R/W	Setting the interval for ASPM L1 requests. The PCIe Base Spec specifies that ASPM L1 transition requests must not be received consecutively within 10 µs, and this field is used to set a timer value to guard against them. Set the ACLK period x set value to 10 µs or more. Setting of this product (Default): In case of ACLK = 200 MHz (5ns): D'2000(d) = H'7D0 = 0111_1101_0000b
15	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14, 13	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
12	MODE_LINK_UPCONFIGURE_CAPABILITY_DISABLED	0b	R/W	Setting the Link Upconfigure Capability bit in the Training Sequence Ordered-set (TS-OS) 0: Link Upconfigure Capability bit = 1b set 1: Link Upconfigure Capability bit = 0b setting (Gen1 × 1) LinkUp may not occur when connecting to Gen1 PCIe devices unless this bit is 0b. In that case, use F/W to change it to 0b.
11	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	MODE_TXSWING	0b	R/W	Amplitude control of SerDes serial output 0: Full Swing Mode (Default) 1: Half Swing Mode
9 to 4	—	H'0F	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3, 2	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	MODE_PORT	1b	R/W	Register for Device Type setting on our company. Changes from the initial values can cause malfunction. Please be careful.

Bit	Bit Name	Initial Value	R/W	Description
0	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

34.3.1.34 PCIe Core Control 1 Registers (Offset: H'404)

This register is a control register for Power Management control and LTSSM (Link Training Sequence State Machine) state transitions in the PCI Express core.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	BLB_RELAX_ORDERING_EN	—	—	—	—	—	UI_ENTER_L1S	—	RETURN_TO_L0	UI_RC_REJECT_ASPML1	Auto PM_Active_State_Nak	UI_ENTER_L2	UI_ENTER_TXL0S
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W

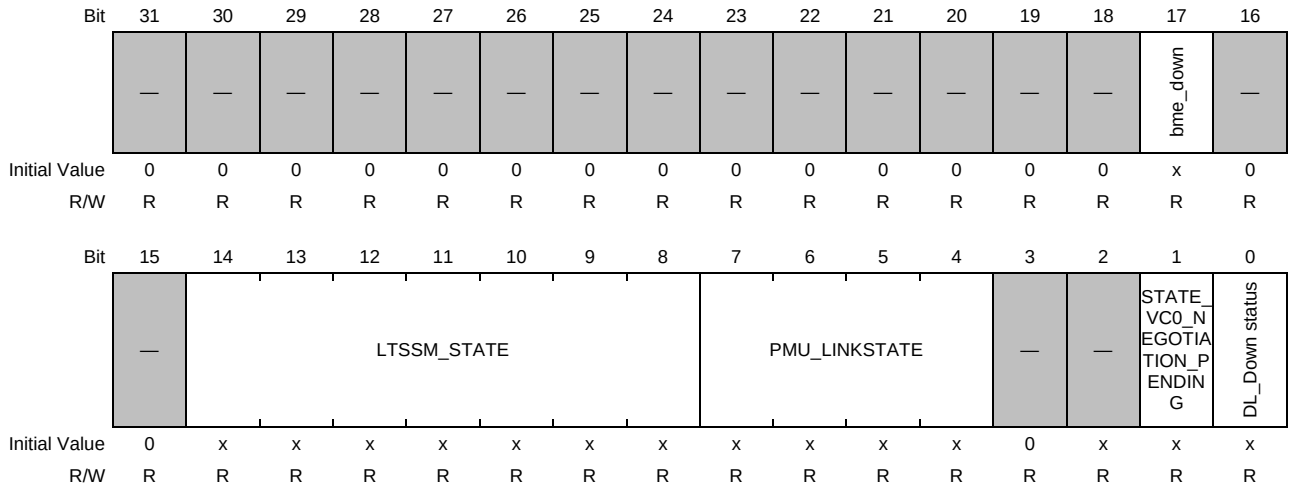
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	UI_ENTER_TX MODE_SRIS	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	BLB_RELAX_ORDERING_EN	0b	R/W	Controlling the RO bit in send requests 0: RO bit of Request TLP sent is always 0b (Default) 1: TLP can be sent with RO bit of Request TLP to be sent set to 1b.
27 to 25	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	UI_ENTER_L1S	0b	R/W	L1 SubState transition permission setting 0: L1 SubState not transitionable (Default) 1: L1 SubState transition allowed
21	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
20	RETURN_TO_L0	0b	R/W	Control of L1, L2 to L0 states in RC mode (Not normally used) 0: Normal operation (Default) 1: Start the return operation to the L0 state when in the L1 and L2 states in the RC mode *Note*: It is automatically cleared after checking PMU_LINKSTATE [0] = 1.
19	UI_RC_REJECT_ASPML1	0b	R/W	ASPM L1 transition rejection control 0: Accepts ASPM L1 transition requests from opposing EP devices (Default) 1: Reject ASPM L1 transition requests from opposing EP devices
18	Auto PM_Active_State_Nak	0b	R/W	PM_Active_State_Nak Message Send Mode for ASPM L1 Rejection Set 1b, if the RC wants to reject ASPM L1. When PM_Active_State_Nak is automatically sent, this Bit is automatically cleared. *Note*: Auto send only once.

Bit	Bit Name	Initial Value	R/W	Description
17	UI_ENTER_L2	0b	R/W	L2 transition control in RC mode Set to 1b when shifting to L2 in RC mode. The PCIe core must enter a reset state under the control of the Reset register during the L2 state transition. When returning, this bit must be cleared to 0b after canceling the reset.
16	UI_ENTER_TXL0S	0b	R/W	TxL0s transition control 0: Do not perform ASPM L0s transition (Default) 1: Perform ASPM L0s transition when internal conditions are satisfied
15 to 13	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
11	UI_ENTER_TXMODE_SRIS	0b	R/W	Setting of Clock Tolerance Compensation 0: SRNS (Default) 1: SRIS (Not support)
10 to 8	—	000b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7 to 4	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	—	H'0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

34.3.1.35 PCIe Core Status 1 Registers (Offset: H'408)

This register is the Power Management status register in the PCI Express core.

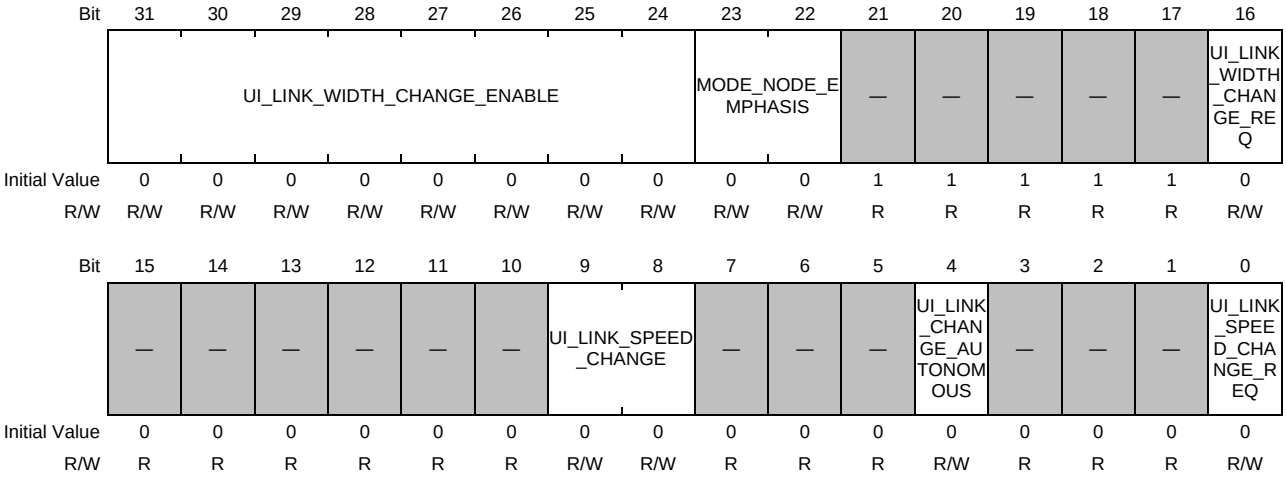


Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 24	—	H'0	R	Reserved When read, the initial value is read. The written value will be ignored.
23	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22 to 20	—	000b	R	Reserved When read, the initial value is read. The written value will be ignored.
19, 18	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	bme_down	x	R	The PCIe core transmitter is in an unusable state.
16	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
15	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14 to 8	LTSSM_STATE	All x	R	Shows the state of the Link Training & Status State Machine in the PCIe Core Link. The high-order 5 bits (14:10) indicate the following conditions: 000xxb: Detect 001xxb: Polling 010xxb: Config 01100b: L0 01101b: L1 0111xb: L2 100xxb: Recovery 101xxb: Disable 110xxb: Loopback
7 to 4	PMU_LINKSTATE	All x	R	L-state monitor in power management control 0100b: L1 state 1000b: L2 state

Bit	Bit Name	Initial Value	R/W	Description
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	x	R	Reserved When read, the initial value is read. The written value will be ignored.
1	STATE_VC0_NEGOTIATION_PENDING	x	R	Flow Control Initialization Action Monitor If this bit is 1b, the transaction must not be initiated from the AXI side. Check the status of this bit as 0b and DL_Down Status (bit [0]) as 0b. 0: Indicates that Flow Control has been initialized 1: Indicates that Flow Control has not been initialized
0	DL_Down status	x	R	Indicates whether PCIe Core is in DL_Down or DL_Up state 0: DL_Up Status 1: DL_Down Status

34.3.1.36 PCIe Core Control 2 Registers (Offset: H'410)

This register is for Link Speed/Width Change control in the PCI Express core.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	UI_LINK_WIDTH_CHANGE_ENABLE	All 0	R/W	Link Width settings to be changed Set the Lane to 1b that should be activated when asserting UI_LINK_WIDTH_CHANGE_REQ to issue a Link Width change request. The low-order bit (bit [24]) drives Lane 0, and the high-order bit (bit [31]) drives Lane 7.
23, 22	MODE_NODE_EMPHASIS	00b	R/W	No de-emphasis mode setting terminal for Gen 1/Gen 2 operation [0] Gen1 operating 0: Normal operating mode (Default) 1: No de-emphasis mode [1] Gen2 operating 0: Normal operating mode (Default) 1: No de-emphasis mode
21 to 17	—	All 1	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
16	UI_LINK_WIDTH_CHANGE_REQ	0b	R/W	Link Width change request control Setting this bit to 1b issues a request to change the link width to the configuration set in the bit [31:24] UI_LINK_WIDTHCHANGE_ENABLE field. Asserting while in the L0 state moves from the recovered state to the configured state and negotiates with the other device. PCIe Core Status 2 Register (Offset: H'414) bit [29] Set it to 0b after seeing UI_LINK_WIDTH_CHANGE_DONE asserted.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9, 8	UI_LINK_SPEED_CHANGE	00b	R/W	Link Speed Settings Set the Link Speed to be changed. 00b: 2.5 GT/s 01b: 5.0 GT/s 10b: 8.0 GT/s (not allowed) 11b: 16.0 GT/s (not allowed)
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	UI_LINK_CHANGE_AUTONOMOUS	0b	R/W	Setting a reason for changing Link Width/Speed 0: reliability reason (Changes for reliability. Direction of band reduction) 1: autonomous reason

Bit	Bit Name	Initial Value	R/W	Description
3 to 1	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UI_LINK_SPEED_CHANGE_REQ	0b	R/W	Setting a reason for changing Link Speed Setting this bit to 1b issues a request to change the Link Speed to the Speed set in the bit [8] UI_LINK_SPEED_CHANGE field. Asserting while in the L0 state moves to the recovery state and negotiates with the other device. PCIe Core Status 2 Register (Offset: H'414) bit [28] Set it to 0b after seeing UI_LINK_SPEED_CHANGE_DONE asserted.

34.3.1.37 PCIe Core Status 2 Registers (Offset: H'414)

This register is the Link Speed/Width Change status register in the PCI Express core.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	UI_LINK_WIDTH_CHANGE_DONE	UI_LINK_SPEED_CHANGE_DONE	—	—	—	STATE_UPCONFIGURE_CAPABLE	—	STATE_NEGOTIATED_LANE_END			—	STATE_NEGOTIATED_LANE_START		
Initial Value	0	0	x	x	0	0	0	x	0	x	x	x	0	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STATE_RECEIVER_DETECTED								STATE_DATA_RATE_IDENTIFIER_RECEIVED							
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29	UI_LINK_WIDTH_CHANGE_DONE	x	R	Link Width Change operation completed display PCIe Core Status 2 Register (Offset: H'410) bit[16] The setting of UI_LINK_WIDTH_CHANGE_REQ notifies (1b) that the Width Change has been completed. The setting of UI_LINK_WIDTH_CHANGE_REQ to 0b sets it to 0b.
28	UI_LINK_SPEED_CHANGE_DONE	x	R	Link Width Change operation completed display PCIe Core Status 2 Register (Offset: H'410) bit[0] The setting of UI_ LINK_WIDTH_CHANGE_REQ notifies (1b) that the Width Change has been completed. The setting of UI_LINK_WIDTH_CHANGE_REQ to 0b sets it to 0b.
27 to 25	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	STATE_UPCONFIGURE_CAPABLE	x	R	Upconfigure Capable bit display for opposing devices Indicates whether the opposing device can change the direction of widening the Link Width. If this bit is 0b, changing the Link Width will not restore the original Link Width.
23	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22 to 20	STATE_NEGOTIATED_LANE_END	All x	R	n Displays the location of Lane Number (n − 1) (n = 2 means Lane 1) after Link Negotiation with an oncoming device during lane operation. Used to check the state of the current behavior Lane before changing Link Width. 000b: Lane0 is Lane Number (n − 1) 001b: Lane1 is Lane Number (n − 1) Others: Reserved
19	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18 to 16	STATE_NEGOTIATED_LANE_START	All x	R	n When the lane is operating, the position of Lane Number 0 after Link Negotiation with the opposing device is displayed. Used to check the state of the current behavior Lane before changing Link Width. 000b: Lane0 is Lane Number 0 001b: Lane1 is Lane Number 0 Others: Reserved

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	STATE_RECEIVER_DETECTED	All x	R	Display connection status with other devices The Receiver Detection results are displayed. [0]: Opposing device detected on Lane 0 [1]: Opposing device detected on Lane 1 (for x2 only) [2] to [7]: Reserved
7 to 0	STATE_DATA_RATE_IDENTIFIER_RECEIVED	All x	R	Viewing Link Speeds Supported by Opposing Devices Displays the TS-OS Data Rate Identifier feed received from the other device. Bit 0: Reserved Bit 1: 2.5 GT/s Data Rate Supported. Must be set to 1b. Bit 2: 5.0 GT/s Data Rate Supported. Must be set to 1b if Bit 3 is 1b. Bit 3-7: Reserved

34.3.1.38 PCIe Core Status 5 Registers (Offset: H'42C)

This register is a status register in the PCI Express core.

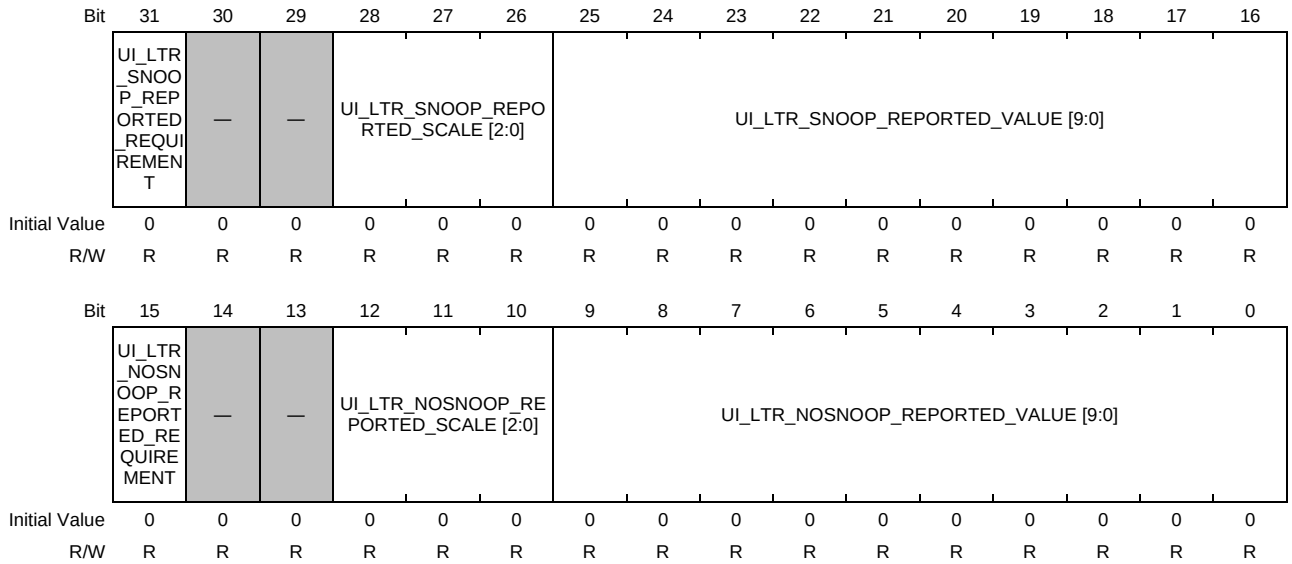
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ORT_TRANSACTION_PENDING
Initial Value	0	0	0	0	0	0	x	x	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
9, 8	—	xx		
7 to 1	—	All 0		
0	ORT_TRANSACTION_PENDING	0b	R	Indicates the presence or absence of an Outstanding Request (All Completions corresponding to Non-Posted Requests sent from the AXI side have not been received.). Check this bit to make sure that there is no Outstanding Request before making/granting transition requests to TxL0s/L1/L2. 0: No Outstanding Request 1: With an Outstanding Request

34.3.1.39 LTR Reported value Registers (Offset: H'470)

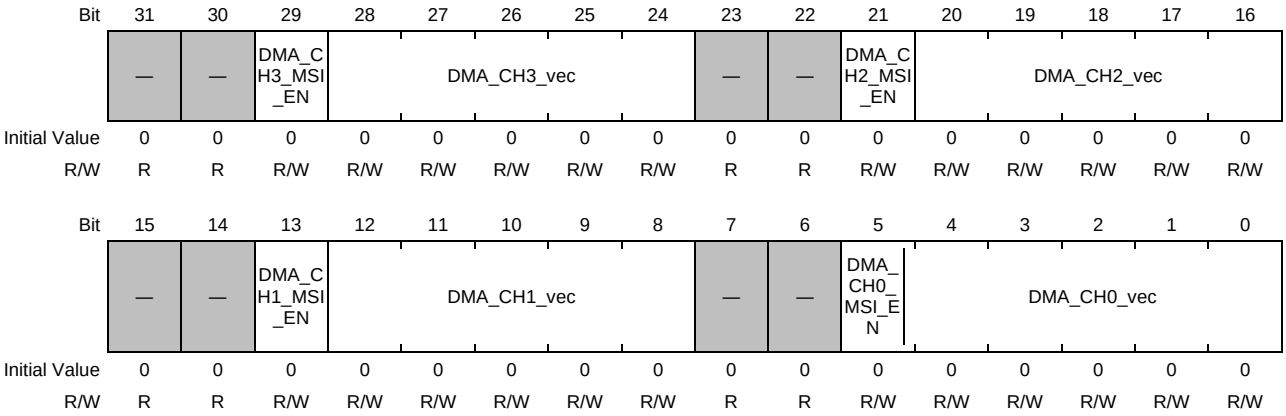
EP only, RC reserved.



Bit	Bit Name	Initial Value	R/W	Description
31	UI_LTR_SNOOP_REPORTED_REQUIREMENT	0b	R	Snoop Latency Requirement field value for issued LTR Msg
30 to 29	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 26	UI_LTR_SNOOP_REPORTED_SCALE [2:0]	000b	R	Snoop Latency Scale field value for issued LTR Msg
25 to 16	UI_LTR_SNOOP_REPORTED_VALUE [9:0]	All 0	R	Snoop Latency Value field value for issued LTR Msg
15	UI_LTR_NOSNOOP_REPORTED_REQUIREMENT	0b	R	NoSnoop Latency Requirement field value for issued LTR Msg
14 to 13	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 10	UI_LTR_NOSNOOP_REPORTED_SCALE [2:0]	000b	R	NoSnoop Latency Scale field value for issued LTR Msg
9 to 0	UI_LTR_NOSNOOP_REPORTED_VALUE [9:0]	All 0	R	NoSnoop Latency Value field value for issued LTR Msg

34.3.1.40 DMA Interrupt Vector 0 Registers (Offset: H'4D0)

Interrupt vector specification function when notifying (MSI) an interrupt from the AXI to the PCIe direction during DMAC transfer.

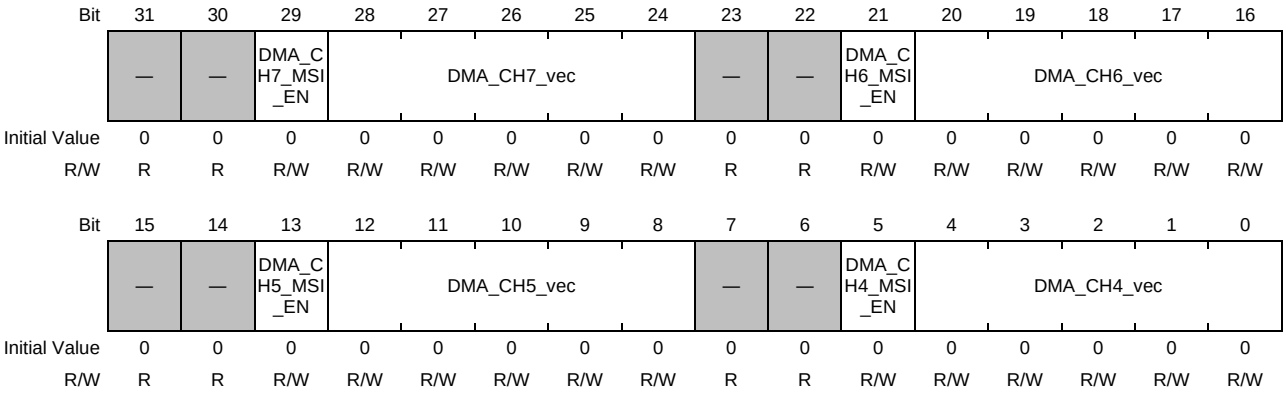


Bit	Bit Name	Initial Value	R/W	Description
31 to 30	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29	DMA_CH3_MSI_EN	0b	R/W	DMA Ch3 MSI Enable 0: Do not use MSI (DMA interrupt notification via DMA_INT terminal) 1: Use MSI For RC function, set it to 0b fixed because MSI transmission is prohibited.
28 to 24	DMA_CH3_vec	All 0	R/W	Vector specification of MSI interrupts sent by DMAC Ch3
23 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	DMA_CH2_MSI_EN	0b	R/W	DMA Ch2 MSI Enable 0: Do not use MSI (DMA interrupt notification via DMA_INT terminal) 1: Use MSI For RC function, set it to 0b fixed because MSI transmission is prohibited.
20 to 16	DMA_CH2_vec	All 0	R/W	Vector specification of MSI interrupts sent by DMAC Ch2
15 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	DMA_CH1_MSI_EN	0b	R/W	DMA Ch1 MSI Enable 0: Do not use MSI (DMA interrupt notification via DMA_INT terminal) 1: Use MSI For RC function, set it to 0b fixed because MSI transmission is prohibited.
12 to 8	DMA_CH1_vec	All 0	R/W	Vector specification of MSI interrupts sent by DMAC Ch1
7 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	DMA_CH0_MSI_EN	0b	R/W	DMA Ch0 MSI Enable 0: Do not use MSI (DMA interrupt notification via DMA_INT terminal) 1: Use MSI For RC function, set it to 0b fixed because MSI transmission is prohibited.
4 to 0	DMA_CH0_vec	All 0	R/W	Vector specification of MSI interrupts sent by DMAC Ch0

Note: DMA_CHx_vec should be fixed at H'00 for RC functions.

34.3.1.41 DMA Interrupt Vector 1 Registers (Offset: H'4D4)

Interrupt vector specification function when notifying (MSI) an interrupt from the AXI to the PCIe direction during DMAC transfer.



Bit	Bit Name	Initial Value	R/W	Description
31 to 30	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29	DMA_CH7_MSI_EN	0b	R/W	DMA Ch7 MSI Enable 0: Do not use MSI (DMA interrupt notification via DMA_INT terminal) 1: Use MSI For RC function, set it to 0b fixed because MSI transmission is prohibited.
28 to 24	DMA_CH7_vec	All 0	R/W	Vector specification of MSI interrupts sent by DMAC Ch7
23 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	DMA_CH6_MSI_EN	0b	R/W	DMA Ch6 MSI Enable 0: Do not use MSI (DMA interrupt notification via DMA_INT terminal) 1: Use MSI For RC function, set it to 0b fixed because MSI transmission is prohibited.
20 to 16	DMA_CH6_vec	All 0	R/W	Vector specification of MSI interrupts sent by DMAC Ch6
15 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	DMA_CH5_MSI_EN	0b	R/W	DMA Ch5 MSI Enable 0: Do not use MSI (DMA interrupt notification via DMA_INT terminal) 1: Use MSI For RC function, set it to 0b fixed because MSI transmission is prohibited.
12 to 8	DMA_CH5_vec	All 0	R/W	Vector specification of MSI interrupts sent by DMAC Ch5
7 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	DMA_CH4_MSI_EN	0b	R/W	DMA Ch4 MSI Enable 0: Do not use MSI (DMA interrupt notification via DMA_INT terminal) 1: Use MSI For RC function, set it to 0b fixed because MSI transmission is prohibited.
4 to 0	DMA_CH4_vec	All 0	R/W	Vector specification of MSI interrupts sent by DMAC Ch4

Note: DMA_CHx_vec should be fixed at H'00 for RC functions.

34.3.1.42 MSI Receive Enable n Registers (Offset: H'6x0)

Registers to enable control of a group of one-set MSI receiving (Message Data Judgment) registers:

- MSI reception enabled n register (offset H'6x0)
- MSI received message data n register (offset H'6x4)
- MSI receive mask n register (offset H'6x8)
- MSI reception status n register (offset H'6xC)

Remark n is 0 ~ 1 and x is the 16 decimal number of n.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

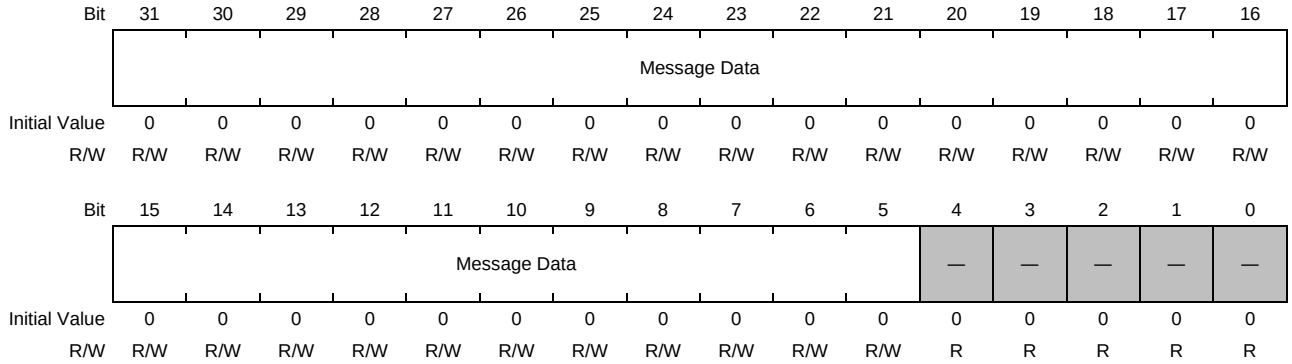
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	Enable	0b	R/W	Enable/disable the MSI Receive (Message Data Judgment) register group consisting of 1 set of + 16 bytes registers from this register's Offset. 0: Disabled 1: Valid

34.3.1.43 MSI Receive Message Data n Registers (Offset: H'6x4)

Message Data setting register for MSI reception judgment of the register group to be enabled in the MSI Receive Enable n register.

Remark n is 0 ~ 1 and x is the 16 decimal number of n.

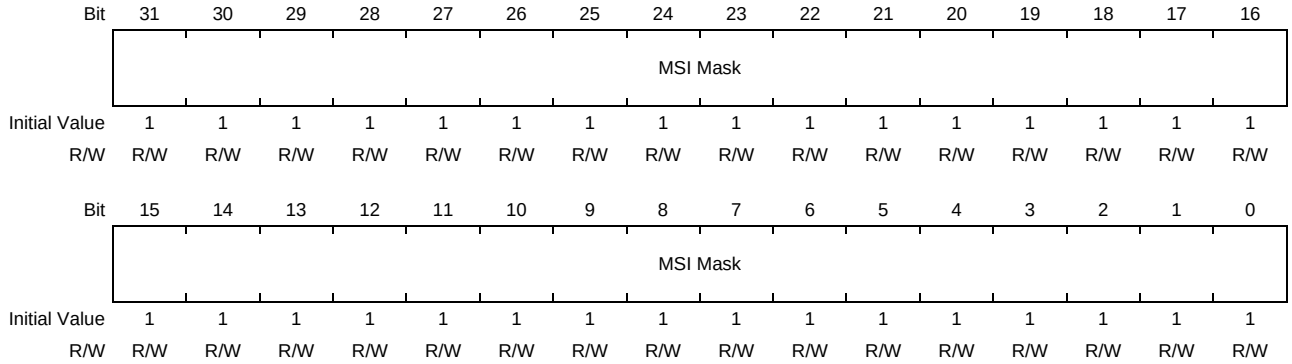


Bit	Bit Name	Initial Value	R/W	Description
31 to 5	Message Data	All 0	R/W	Setting Message Data for MSI reception judgment
4 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

34.3.1.44 MSI Receive Mask n Registers (Offset: H'6x8)

In the MSI Receive Enable n register, the Mask control register for interrupts by the MSI receive status of the register group to enable.

Remark n is 0 ~ 1 and x is the 16 decimal number of n.

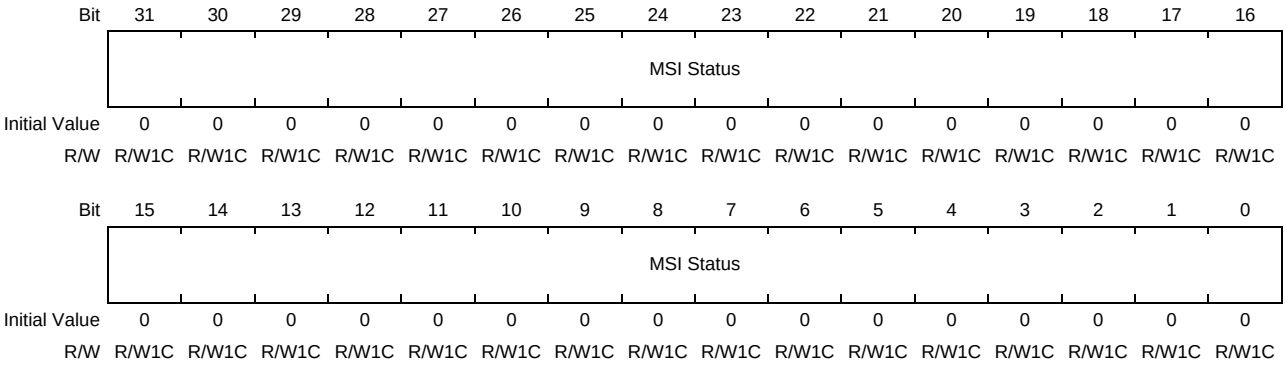


Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MSI Mask	All 1	R/W	Set the interrupt mask with each bit of MSI Status. 0: Allow interrupt 1: Interrupt Mask

34.3.1.45 MSI Receive Status n Registers (Offset: H'6xC)

A register indicating the MSI receive status of the group of registers to be enabled in the MSI Receive Enable n register.

Remark n is 0 ~ 1 and x is the 16 decimal number of n.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MSI Status	All 0	R/W1C	MSI receive status. When Message Data [31:5] and MSI receive Data [31:5] match, the bit corresponding to Data [4:0] is set. (When Data (4:0) = H'00, MSI Status n (1) is set when MSI Status n (0) is Data (4:0) = H'01.)

34.3.1.46 DMA Control Registers (Offset: H'800)

As a function of DMAC, this is a setting related to the upper limit of the Read Request Size that can be issued to PCIe Core. Use with the default setting (128 bytes). Common settings for DMA Channel

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	D_PMRS		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	D_PMRS	000b	R/W	DMAC PCIe Max Read Request Size Sets an upper limit on Read requests issued from DMAC to PCIe. 000b: 128 bytes (Default) 001b: 256 bytes 010b: 512 bytes (Not supported) 011b: 1024 bytes (Not supported) 100b: 2048 bytes (Not supported) 101b: 4096 bytes (Not supported) Others: Reserved (Not set)

34.3.1.47 DMA Interrupt Enable Registers (Offset: H'808)

This register is for interrupt permission on each DMA Channel.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CH7_ERR_EN	CH7_QUE_EMP_EN	—	CH7_END_EN	CH6_ERR_EN	CH6_QUE_EMP_EN	—	CH6_END_EN	CH5_ERR_EN	CH5_QUE_EMP_EN	—	CH5_END_EN	CH4_ERR_EN	CH4_QUE_EMP_EN	—	CH4_END_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH3_ERR_EN	CH3_QUE_EMP_EN	—	CH3_END_EN	CH2_ERR_EN	CH2_QUE_EMP_EN	—	CH2_END_EN	CH1_ERR_EN	CH1_QUE_EMP_EN	—	CH1_END_EN	CH0_ERR_EN	CH0_QUE_EMP_EN	—	CH0_END_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	CH7_ERR_EN	0b	R/W	CH7 Error interrupt Enable 0: Disable 1: Enable
30	CH7_QUE_EMP_EN	0b	R/W	CH7 Que Empty interrupt Enable 0: Disable 1: Enable
29	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
28	CH7_END_EN	0b	R/W	CH7 Completion Interrupt Enable 0: Disable 1: Enable
27	CH6_ERR_EN	0b	R/W	CH6 Error interrupt Enable 0: Disable 1: Enable
26	CH6_QUE_EMP_EN	0b	R/W	CH6 Que Empty interrupt Enable 0: Disable 1: Enable
25	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
24	CH6_END_EN	0b	R/W	CH6 Completion Interrupt Enable 0: Disable 1: Enable
23	CH5_ERR_EN	0b	R/W	CH5 Error interrupt Enable 0: Disable 1: Enable
22	CH5_QUE_EMP_EN	0b	R/W	CH5 Que Empty interrupt Enable 0: Disable 1: Enable
21	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

Bit	Bit Name	Initial Value	R/W	Description
20	CH5_END_EN	0b	R/W	CH5 Completion Interrupt Enable 0: Disable 1: Enable
19	CH4_ERR_EN	0b	R/W	CH4 Error interrupt Enable 0: Disable 1: Enable
18	CH4_QUE_EMP_EN	0b	R/W	CH4 Que Empty interrupt Enable 0: Disable 1: Enable
17	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
16	CH4_END_EN	0b	R/W	CH4 Completion Interrupt Enable 0: Disable 1: Enable
15	CH3_ERR_EN	0b	R/W	CH3 Error interrupt Enable 0: Disable 1: Enable
14	CH3_QUE_EMP_EN	0b	R/W	CH3 Que Empty interrupt Enable 0: Disable 1: Enable
13	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
12	CH3_END_EN	0b	R/W	CH3 Completion Interrupt Enable 0: Disable 1: Enable
11	CH2_ERR_EN	0b	R/W	CH2 Error interrupt Enable 0: Disable 1: Enable
10	CH2_QUE_EMP_EN	0b	R/W	CH2 Que Empty interrupt Enable 0: Disable 1: Enable
9	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
8	CH2_END_EN	0b	R/W	CH2 Completion Interrupt Enable 0: Disable 1: Enable
7	CH1_ERR_EN	0b	R/W	CH1 Error interrupt Enable 0: Disable 1: Enable
6	CH1_QUE_EMP_EN	0b	R/W	CH1 Que Empty interrupt Enable 0: Disable 1: Enable
5	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

Bit	Bit Name	Initial Value	R/W	Description
4	CH1_END_EN	0b	R/W	CH1 Completion Interrupt Enable 0: Disable 1: Enable
3	CH0_ERR_EN	0b	R/W	CH0 Error interrupt Enable 0: Disable 1: Enable
2	CH0_QUE_EMP_EN	0b	R/W	CH0 Que Empty interrupt Enable 0: Disable 1: Enable
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	CH0_END_EN	0b	R/W	CH0 Completion Interrupt Enable 0: Disable 1: Enable

34.3.1.48 DMA Interrupt Status Registers (Offset: H'80C)

This register is the interrupt status register on each DMA Channel.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CH7_ERR	CH7_QUE_EMP	—	CH7_END	CH6_ERR	CH6_QUE_EMP	—	CH6_END	CH5_ERR	CH5_QUE_EMP	—	CH5_END	CH4_ERR	CH4_QUE_EMP	—	CH4_END
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W1C	R/W1C	R	R/W1C	R/W1C	R/W1C	R	R/W1C	R/W1C	R/W1C	R	R/W1C	R/W1C	R/W1C	R	R/W1C

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH3_ERR	CH3_QUE_EMP	—	CH3_END	CH2_ERR	CH2_QUE_EMP	—	CH2_END	CH1_ERR	CH1_QUE_EMP	—	CH1_END	CH0_ERR	CH0_QUE_EMP	—	CH0_END
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W1C	R/W1C	R	R/W1C	R/W1C	R/W1C	R	R/W1C	R/W1C	R/W1C	R	R/W1C	R/W1C	R/W1C	R	R/W1C

Bit	Bit Name	Initial Value	R/W	Description
31	CH7_ERR	0b	R/W1C	Set when an Error occurs during a DMA transfer.
30	CH7_QUE_EMP	0b	R/W1C	It is set when the QUE is empty by pulling the list from the descriptor queue (moving it to the execution descriptor list).
29	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
28	CH7_END	0b	R/W1C	Set when DMAC completes successfully. Successful completion refers to the following conditions: • Transfer finished as indicated by DMA_SIZE • End of descriptor list with EI field equal to 1
27	CH6_ERR	0b	R/W1C	Set when an Error occurs during a DMA transfer.
26	CH6_QUE_EMP	0b	R/W1C	It is set when the QUE is empty by pulling the list from the descriptor queue (moving it to the execution descriptor list).
25	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
24	CH6_END	0b	R/W1C	Set when DMAC completes successfully. Successful completion refers to the following conditions: • Transfer finished as indicated by DMA_SIZE • End of descriptor list with EI field equal to 1
23	CH5_ERR	0b	R/W1C	Set when an Error occurs during a DMA transfer.
22	CH5_QUE_EMP	0b	R/W1C	It is set when the QUE is empty by pulling the list from the descriptor queue (moving it to the execution descriptor list).
21	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
20	CH5_END	0b	R/W1C	Set when DMAC completes successfully. Successful completion refers to the following conditions: • Transfer finished as indicated by DMA_SIZE • End of descriptor list with EI field equal to 1
19	CH4_ERR	0b	R/W1C	Set when an Error occurs during a DMA transfer.
18	CH4_QUE_EMP	0b	R/W1C	It is set when the QUE is empty by pulling the list from the descriptor queue (moving it to the execution descriptor list).
17	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

Bit	Bit Name	Initial Value	R/W	Description
16	CH4_END	0b	R/W1C	Set when DMAC completes successfully. Successful completion refers to the following conditions: • Transfer finished as indicated by DMA_SIZE • End of descriptor list with EI field equal to 1
15	CH3_ERR	0b	R/W1C	Set when an Error occurs during a DMA transfer.
14	CH3_QUE_EMP	0b	R/W1C	It is set when the QUE is empty by pulling the list from the descriptor queue (moving it to the execution descriptor list).
13	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
12	CH3_END	0b	R/W1C	Set when DMAC completes successfully. Successful completion refers to the following conditions: • Transfer finished as indicated by DMA_SIZE • End of descriptor list with EI field equal to 1
11	CH2_ERR	0b	R/W1C	Set when an Error occurs during a DMA transfer.
10	CH2_QUE_EMP	0b	R/W1C	It is set when the QUE is empty by pulling the list from the descriptor queue (moving it to the execution descriptor list).
9	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
8	CH2_END	0b	R/W1C	Set when DMAC completes successfully. Successful completion refers to the following conditions: • Transfer finished as indicated by DMA_SIZE • End of descriptor list with EI field equal to 1
7	CH1_ERR	0b	R/W1C	Set when an Error occurs during a DMA transfer.
6	CH1_QUE_EMP	0b	R/W1C	It is set when the QUE is empty by pulling the list from the descriptor queue (moving it to the execution descriptor list).
5	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
4	CH1_END	0b	R/W1C	Set when DMAC completes successfully. Successful completion refers to the following conditions: • Transfer finished as indicated by DMA_SIZE • End of descriptor list with EI field equal to 1
3	CH0_ERR	0b	R/W1C	Set when an Error occurs during a DMA transfer.
2	CH0_QUE_EMP	0b	R/W1C	It is set when the QUE is empty by pulling the list from the descriptor queue (moving it to the execution descriptor list).
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	CH0_END	0b	R/W1C	Set when DMAC completes successfully. Successful completion refers to the following conditions: • Transfer finished as indicated by DMA_SIZE • End of descriptor list with EI field equal to 1

34.3.1.49 DMA Channel Control Registers (Offset: H'900/+Channel Offset)

This register is used to set the control method for each DMA Channel.

Please set register/descriptor method exclusively.

- Setting QUE_EN = 1b and QUE_CLR = 1b is prohibited during register-type DMA transfer (TDMA_EN = 1b).
- Setting RDMA_EN = 1b is prohibited during descriptor-type DMA transfer (QUE_EN = 1b).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

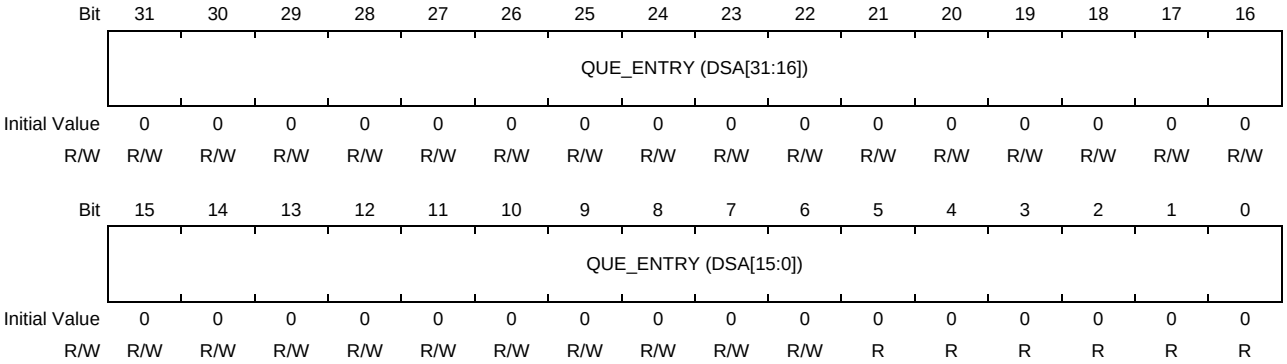
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	QUE_CLR	—	—	—	—	—	—	QUE_EN	RDMA_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	QUE_CLR	0b	R/W	QUE Clear Write 1 to clear QUE. All descriptor lists (waiting and running lists) registered in QUE are cleared. Do not clear during DMA execution. The lead value is always 0. It is prohibited to 1write at the same time as Set of QUE_EN.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	QUE_EN	0b	R/W	QUE Enable Set to 1 to enable the descriptor list registered in the descriptor queue and start DMA transfer (descriptor method). When the DMA stops (normal or abnormal), it is automatically cleared to 0.
0	RDMA_EN	0b	R/W	Register-based DMA transfer Enable Set to 1 to start a DMA transfer (register method) and perform the transfer set by RDMA_SIZE. It is automatically cleared to 0 when the DMA transfer is complete, or an error is detected, and the DMA is terminated.

34.3.1.50 Descriptor Start Address (Lower) Registers (Offset: H'908/+Channel Offset)

This registry is a register for descriptor queue list settings.

This setting is registered as the lower 32 bits of the DSA (DMA Start Address) of the queue list.



Bit	Bit Name	Initial Value	R/W	Description
31 to 6	QUE_ENTRY (DSA[31:6])	All 0	R/W	Descriptor list queue register. This area becomes the DSA. Sets the low 32 bits of the address where the first descriptor is stored.
5 to 0	QUE_ENTRY (DSA[5:0])	All 0	R	When the Data bus width is 128 bits, 16 bytes alignment: the lower 6 bits are fixed at 0.

NOTE

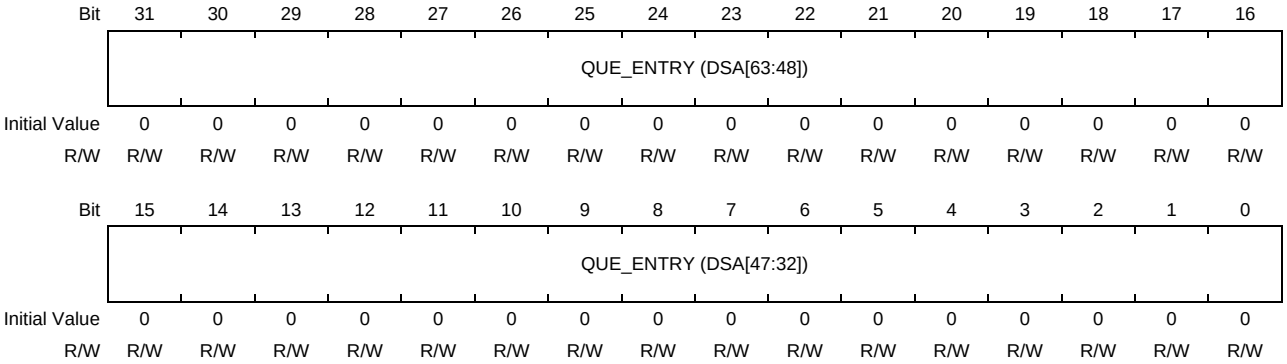
At the time of a lead, the following content will be the lead value depending on the DMA situation.

- DMA transfer in progress: running descriptor list
- DMA stopped (when QUE_EN auto-clear): Last run list

34.3.1.51 Descriptor Start Address (Upper) Registers (Offset: H'90C/+Channel Offset)

This registry is a register for descriptor queue list settings.

This setting is registered as the upper 32 bits of the DSA (DMA Start Address) of the queue list.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	QUE_ENTRY (DSA[63:32])	All 0	R/W	Descriptor list queue register. This area becomes the DSA. Sets the up 32 bits of the address where the first descriptor is stored.

NOTE

At the time of a lead, the following content will be the lead value depending on the DMA situation.

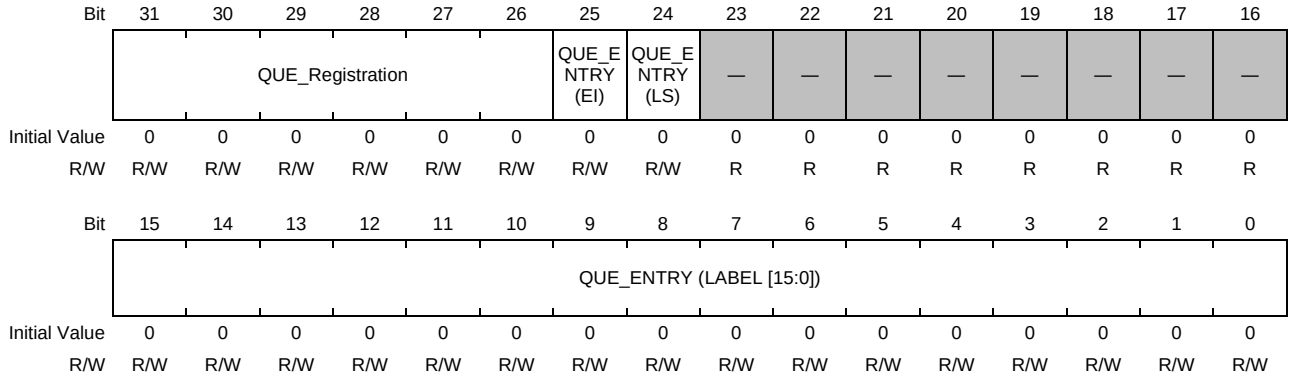
- DMA transfer in progress: running descriptor list
- DMA stopped (when QUE_EN auto-clear): Last run list

34.3.1.52 Que Entry Registers (Offset: H'910/+Channel Offset)

This register is for setting the descriptor queue list.

This setting is registered as EI (End Interrupt), LS (List Stop) and LABEL in the queue list.

Writing to 31:24 puts it in the queue.



Bit	Bit Name	Initial Value	R/W	Description
31 to 26	QUE_Registration	All 0	R/W	Write to 31:24 to queue. The read value is always 0.
25	QUE_ENTRY(EI)	0b	R/W	It indicates whether to signal an interrupt (Interrupt Status CHx_END) when the descriptor list is processing. 0: Do not signal interrupts 1: Notify interrupt
24	QUE_ENTRY(LS)	0b	R/W	It indicates whether to stop DMA upon completion of descriptor list processing. 0: Do not stop 1: Stop
23 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	QUE_ENTRY (LABEL [15:0])	All 0	R/W	In the sense of a list label, there are no rules for how to set it up. Feel free to set values.

NOTE

At the time of a lead, the following content will be the lead value depending on the DMA situation.

- DMA transfer in progress: running descriptor list
- DMA stopped (when QUE_EN auto-clear): Last run list

34.3.1.53 DMA Descriptor Control (Descriptor H’00) Registers (Offset: H’920/+Channel Offset)

This register shows the field value of Offset H’00 in the descriptor table. Valid only when the descriptor method DMA is selected (read values have no meaning when using the register method).

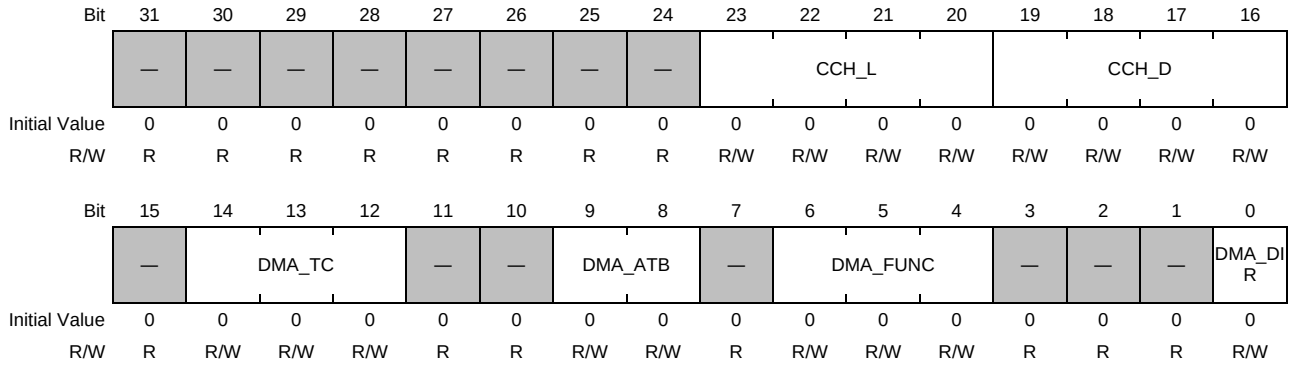
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSCFM				—	WBD	LE	LV	D	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STS															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	DSCFM	H’0	R	It indicates the value of the DSCFM field in the running descriptor table.
27	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	WBD	0b	R	It indicates the value of the WBD field in the running descriptor table.
25	LE	0b	R	It indicates the value of the LE field in the running descriptor table.
24	LV	0b	R	It indicates the value of the LV field in the running descriptor table.
23	D	0b	R	It indicates the value of the D field in the running descriptor table.
22 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	STS	All 0	R	It indicates the value of the STS field in the running descriptor table.

34.3.1.54 DMA Transaction Control (Descriptor H'04) Registers (Offset: H'924/+Channel Offset))

This register is used to set transfer control to the AXI and PCIe sides when performing DMA transfers.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 20	CCH_L	All 0	R/W	It indicates the value of A * CACHE [3:0] to be issued to AXI. CCH_L is printed when issuing an AXI request with the last byte in the transfer indicated by SIZE. The recommended value is 0000b. Bit [3] 1b: Write assignment possible 0b: Write assignment possible Bit [2] 1b: Read assignment possible 0b: Read assignment impossible Bit [1] 1b: Cacheable 0b: Not Cacheable Bit [0] 1b: Buffable 0b: Not buffered
19 to 16	CCH_D	All 0	R/W	It indicates the value of A * CACHE [3:0] to be issued to AXI. CCH_D is output when an AXI request other than the output condition of CCH_L is issued. The recommended value is 0001b when DIR = 0 (PCIe AXI) and 0000b when DIR = 1 (AXI/PCIe). Bit [3] 1b: Write assignment possible 0b: Write assignment possible Bit [2] 1b: Read assignment possible 0b: Read assignment impossible Bit [1] 1b: Cacheable 0b: Not Cacheable Bit [0] 1b: Buffable 0b: Not buffered
15	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14 to 12	DMA_TC	000b	R/W	The traffic class to publish to PCIe. Specifies the traffic class value for requests issued to PCIe. *Note*: This IP does not support Virtual Channel, so use it fixed at 000b.
11, 10	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9, 8	DMA_ATB	00b	R/W	Attribute to publish to PCIe. Bit [1]: Relaxed Ordering (unsupported features: 0b fixed) Bit [0]: No Snoop (0b recommended)
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6 to 4	DMA_FUNC	000b	R/W	Specifies the function number of the request to be issued to PCIe.
3 to 1	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

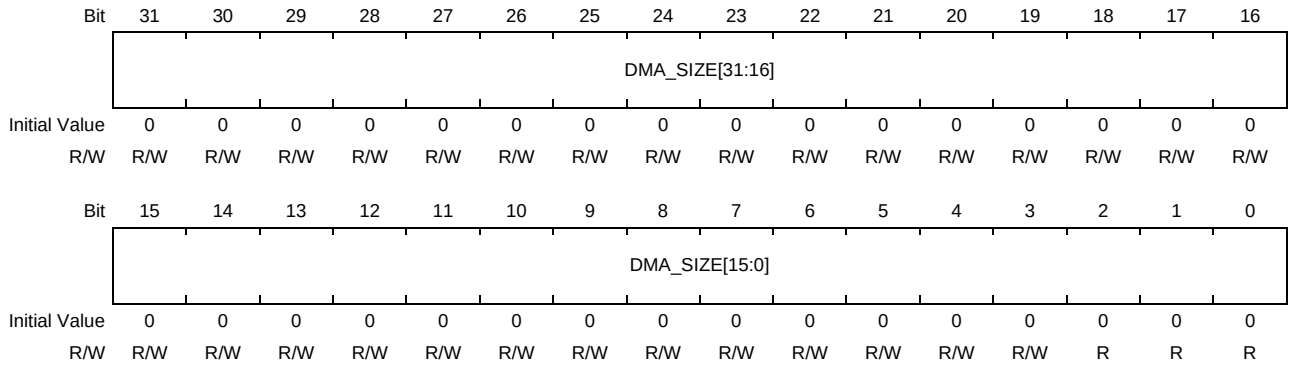
Bit	Bit Name	Initial Value	R/W	Description
0	DMA_DIR	0b	R/W	Sets the DMA transfer direction. 0: PCIe → AXI 1: AXI → PCIe

NOTE

During descriptor method DMA execution, write is prohibited and the read value indicates the value of the CCH_L, CCH_D, TC, ATB, DIR fields in the running descriptor table.

34.3.1.55 DMA Size (Descriptor H’08) Registers (Offset: H’928/+Channel Offset)

This register sets the number of DMA transfer bytes. It is reflected in the field OffsetH’08 in the descriptor table.



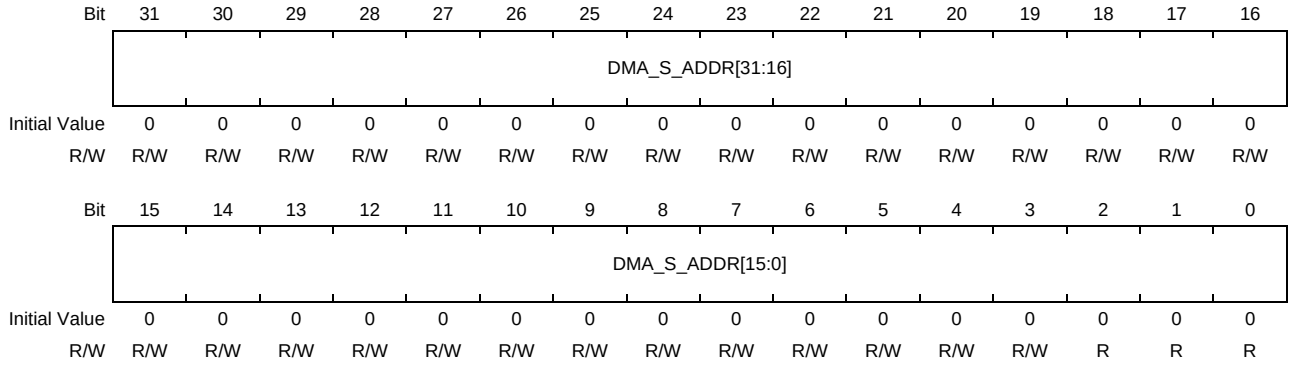
Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DMA_SIZE [31:3]	All 0	R/W	Sets the number of DMA transfer bytes.
2 to 0	DMA_SIZE [2:0]	000b	R	Since the setting for 8-byte alignment, the lower 3 bits are fixed to 000b.

NOTE

During descriptor method DMA execution, write is prohibited and the read value is the value of the SIZE field in the running descriptor table.
The number of Bytes transferred when H’0000_0000 is set is H’1_0000_0000.

34.3.1.56 DMA Source Lower Address (Descriptor H'10) Registers (Offset: H'930/+Channel Offset)

This register sets the low-order 32 bits of the start address of the transfer source for DMA transfers. It is reflected in the Offset H'10 field of the descriptor table.



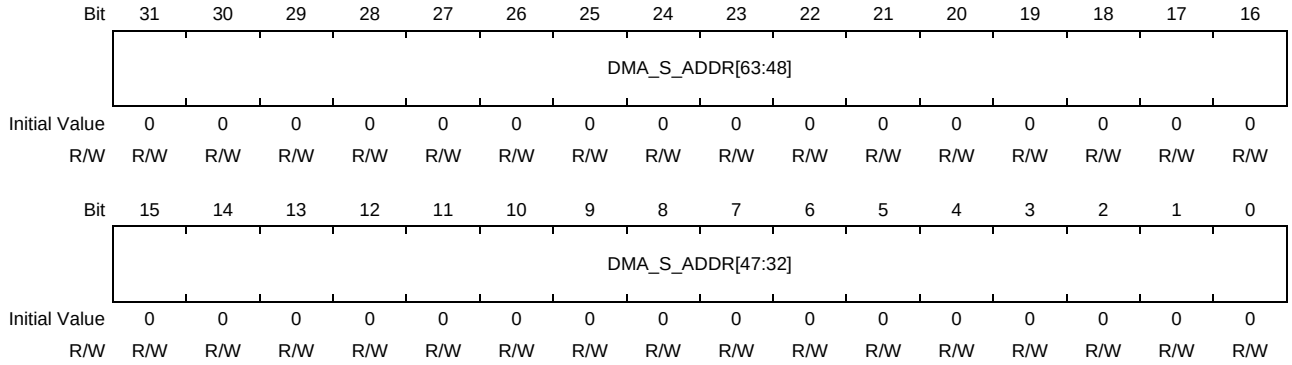
Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DMA_S_ADDR [31:3]	All 0	R/W	Sets the low 32 bits of the source start address for DMA transfers.
2 to 0	DMA_S_ADDR [2:0]	000b	R	Since the setting for 8-byte alignment, the lower 3 bits are fixed to 000b.

NOTE

When Source Address indicates the PCIe space (DIR = 0), it is combined with the DMA PCIe Upper Address (Descriptor H'10) to form the PCIe request address. During descriptor method DMA execution, write is prohibited and the read value indicates the value of the SA field in the running descriptor table.

34.3.1.57 DMA Source Upper Address (Descriptor H'14) Registers (Offset: H'934/+Channel Offset)

This register sets the up-order 32 bits of the start address of the transfer source for DMA transfers. It is reflected in the Offset H'14 field of the descriptor table.



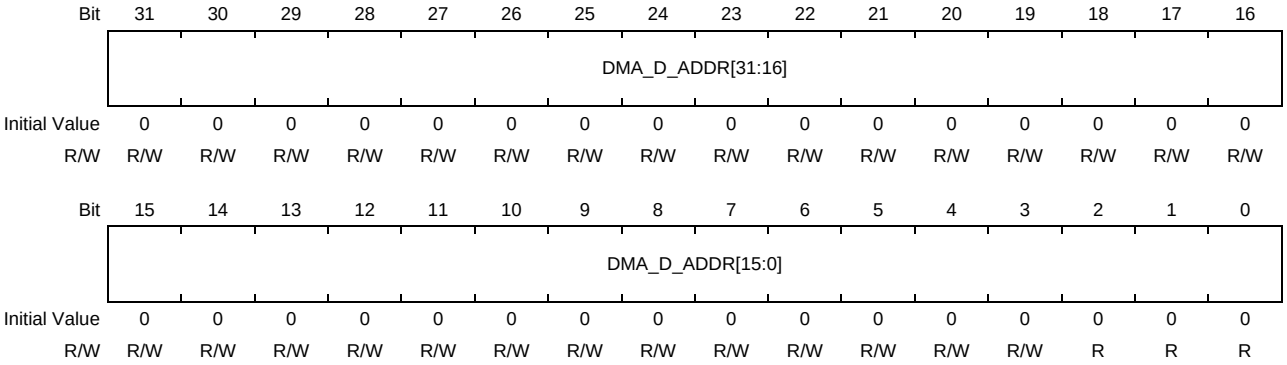
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_S_ADDR	All 0	R/W	Sets the up 32 bits of the source start address for DMA transfers.

NOTE

When Source Address indicates the PCIe space (DIR = 0), it is combined with the DMA PCIe Upper Address (Descriptor H'10) to form the PCIe request address. During descriptor method DMA execution, write is prohibited and the read value indicates the value of the SA field in the running descriptor table.

34.3.1.58 DMA Destination Lower Address (Descriptor H’18) Registers (Offset: H’938/+Channel Offset)

This register sets the low-order 32 bits of the start address of the transfer source for DMA transfers. It is reflected in the Offset H’18 field of the descriptor table.



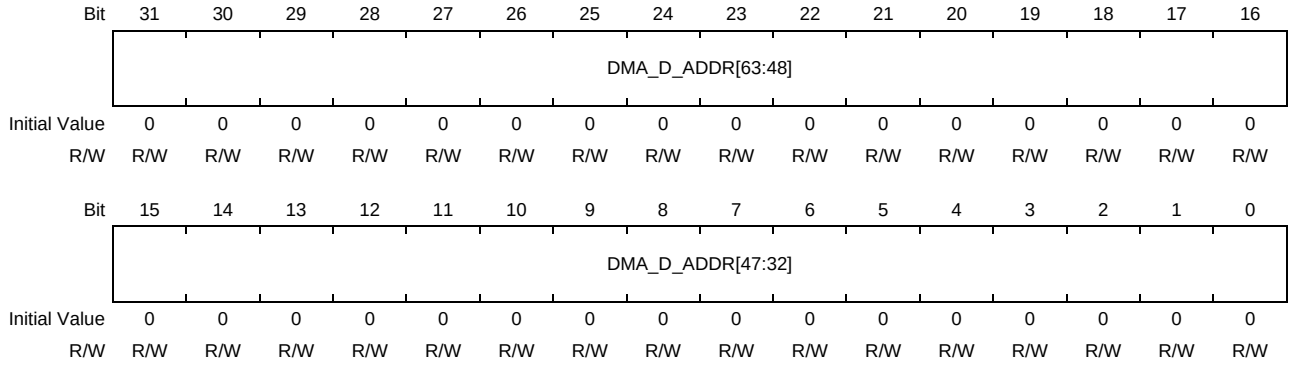
Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DMA_D_ADDR [31:3]	All 0	R/W	Sets the low 32 bits of the source start address for DMA transfers.
2 to 0	DMA_D_ADDR [2:0]	000b	R	Since the setting for 8-byte alignment, the lower 3 bits are fixed to 000b.

NOTE

When Destination Address indicates the PCIe space (DIR = 1), it is combined with the DMA PCIe Upper Address (Descriptor H’10) to form the PCIe request address. During descriptor method DMA execution, write is prohibited and the read value indicates the value of the DA field in the running descriptor table.

34.3.1.59 DMA Destination Upper Address (Descriptor H'1C) Registers (Offset: H'93C/+Channel Offset)

This register sets the up-order 32 bits of the start address of the transfer source for DMA transfers. It is reflected in the Offset H'32 field of the descriptor table.

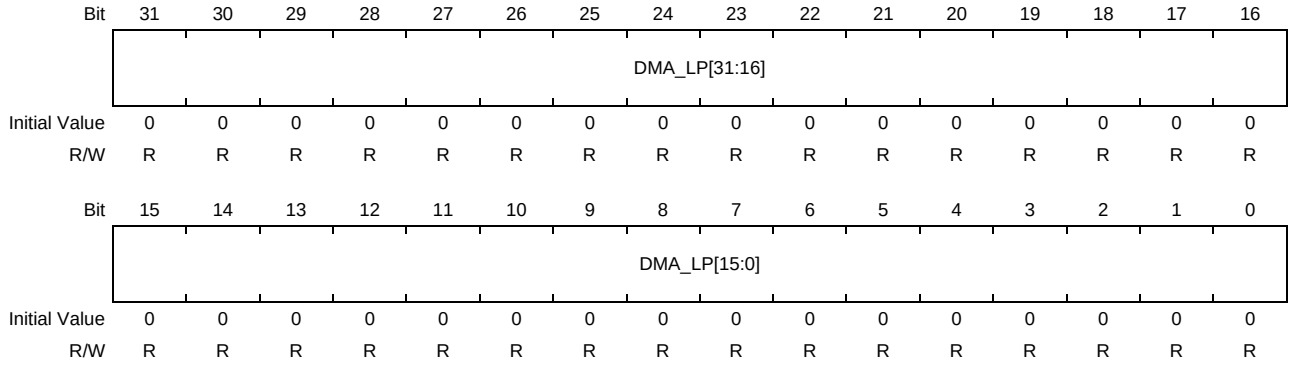


Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_D_ADDR	All 0	R/W	Sets the up 32 bits of the source start address for DMA transfers.

Note: When Destination Address indicates PCIe space (DIR = 1), it becomes PCIe request address together with DMA PCIe Upper Address (Descriptor H'10). When executing descriptor method DMA, writing is prohibited, and the read value indicates the value of the DA field of the descriptor table being executed.

34.3.1.60 DMA Descriptor Lower Link Pointer (Descriptor H'20) Registers (Offset: H'940/+Channel Offset)

This register shows the Offset H'20 field value in the descriptor table. Valid only when descriptor method DMA is selected (read value has no meaning when using register method)



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_LP	All 0	R	It indicates the value of the LP field in the running descriptor table.

34.3.1.61 DMA Descriptor Upper Link Pointer (Descriptor H'24) Registers (Offset: H'944/+Channel Offset)

This register shows the Offset H'24 field value in the descriptor table. Valid only when descriptor method DMA is selected (read value has no meaning when using register method)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_LP[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_LP[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_LP	All 0	R	Indicates the value of the LP field in the running descriptor table.

34.3.1.62 DMA Rest Size Registers (Offset: H'950/+Channel Offset)

This register shows the number of Bytes for which DMA transfers have not been completed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_REST_SIZE[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_REST_SIZE[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_REST_SIZE	All 0	R	It shows the number of Bytes for which DMA transfers have not been completed. (Register/descriptor method common)

34.3.1.63 AXI Request Address (Lower) Registers (Offset: H'960/+Channel Offset)

This register displays the low 32 bits of the address of a transfer that is running an AXI transfer or has just completed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AXI_REQ_ADDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AXI_REQ_ADDR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	AXI_REQ_ADDR	All 0	R	It shows the low 32 bits of the address of a transfer that is running an AXI transfer or has just completed. (Register/descriptor method common)

34.3.1.64 AXI Request Address (Upper) Registers (Offset: H'964/+Channel Offset)

This register displays the up 32 bits of the address of a transfer that is running an AXI transfer or has just completed.

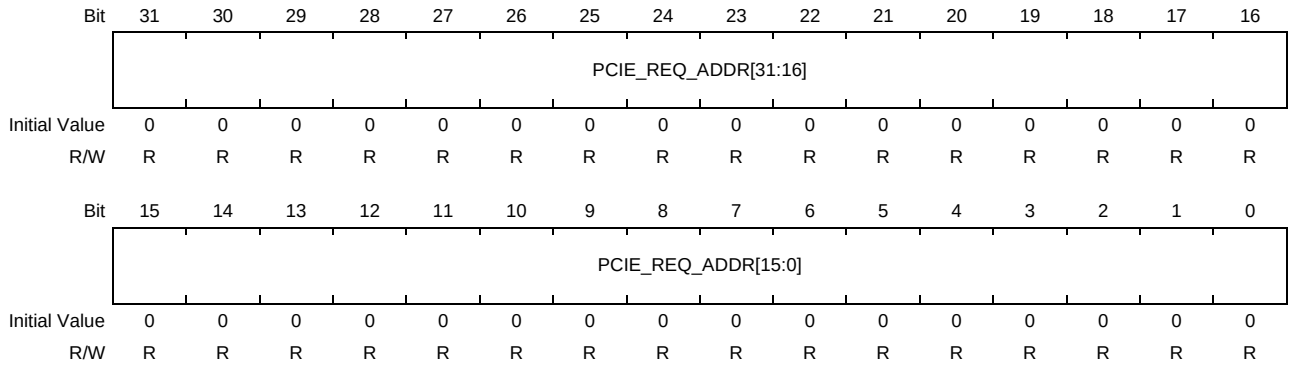
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AXI_REQ_ADDR[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AXI_REQ_ADDR[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	AXI_REQ_ADDR	All 0	R	It shows the up 32 bits of the address of a transfer that is running an AXI transfer or has just completed. (Register/descriptor method common)

34.3.1.65 PCIe Request Address (Lower) Registers (Offset: H'968/+Channel Offset)

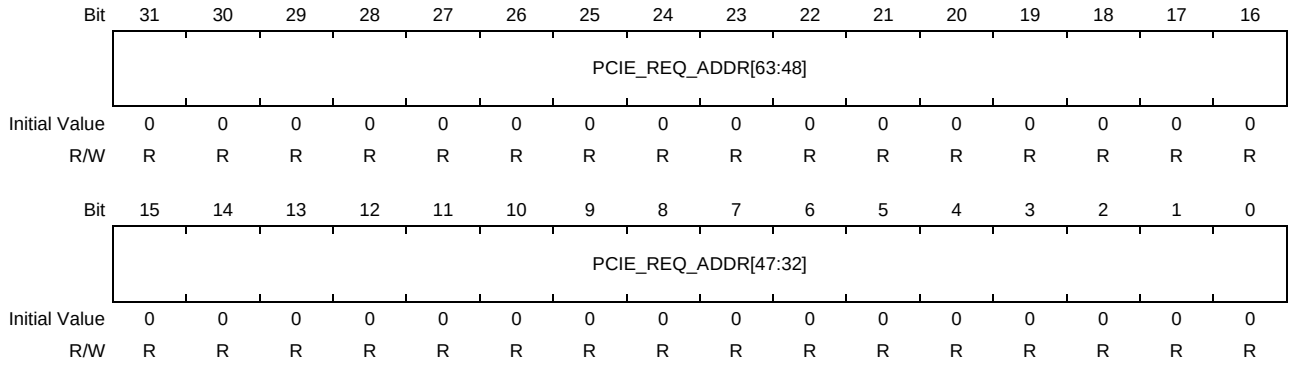
This register displays the low 32 bits of the address of a transfer that is running an PCIe transfer or has just completed.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PCIE_REQ_ADDR	All 0	R	It shows the low 32 bits of the address of a transfer that is running an PCIe transfer or has just completed. (Register/descriptor method common)

34.3.1.66 PCIe Request Address (Upper) Registers (Offset: H'96C/+Channel Offset)

This register displays the up 32 bits of the address of a transfer that is running an PCIe transfer or has just completed.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PCIE_REQ_ADDR	All 0	R	It shows the up 32 bits of the address of a transfer that is running an PCIe transfer or has just completed. (Register/descriptor method common)

34.3.1.67 QUE Status Registers (Offset: H'970/+Channel Offset)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	GO_LIST	LIST_NUM			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	GO_LIST	0b	R	It shows with or without a running descriptor list. 0: None 1: Yes
3 to 0	LIST_NUM	H'0	R	It shows the number of descriptor lists stacked on the QUE (not including the currently running list). When this register indicates H'8, new registrations to the QUE (write access to the QUE Entry) are disabled (discarded).

34.3.1.68 DMAC Error Status Registers (Offset: H'978/+Channel Offset)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BME_DOWN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	MOR_CD_PERR	MOR_CH_PERR	MOR_EP_ERR	MOR_STATUS			—	—	—	—	—	—	AXI_RESP	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

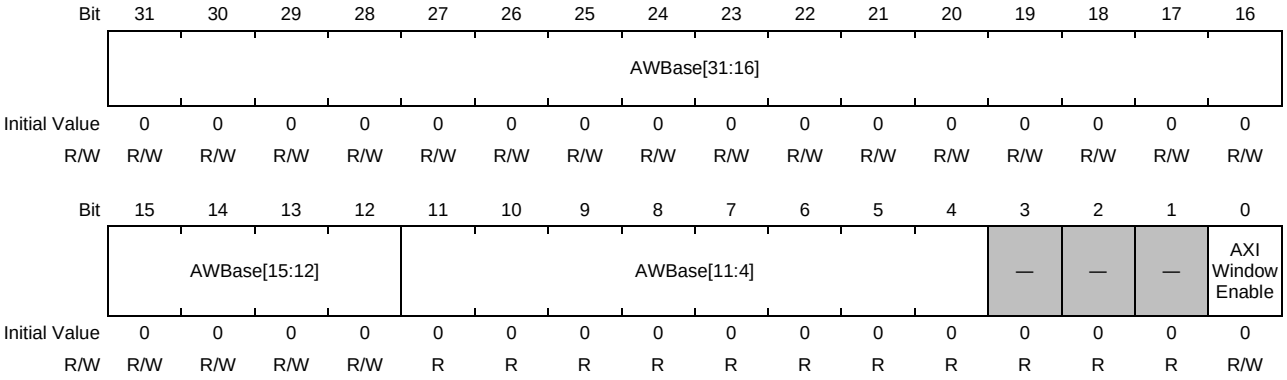
Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	BME_DOWN	0b	R	This bit is set, when a stop signal from the PCIe core by CHx_ERR error detection. Holds this bit until CHx_ERR error is cleared.
15, 14	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	MOR_CD_PERR	0b	R	This bit is set, detecting MOR_CD_PERR error by CHx_ERR error detection. Holds this bit until CHx_ERR error is cleared.
12	MOR_CH_PERR	0b	R	This bit is set, detecting MOR_CH_PERR error by CHx_ERR error detection. Holds this bit until CHx_ERR error is cleared.
11	MOR_EP_ERR	0b	R	This bit is set, detecting Poisoned Completion status by CHx_ERR error detection. Holds this bit until CHx_ERR error is cleared.
10 to 8	MOR_STATUS	All 0	R	These bits set below value, detecting MOR_STATUS is not 000b (Success) by CHx_ERR error detection. Holds these bits until CHx_ERR error is cleared. 000b: Initial value 001b: Unsuspended Request 010b: CRS 011b: Completion Timeout 100b: Completer Abort 101b: Unexpected Completion 110b: Reserved 111b: Mismatched Length (Length Overrun)
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	AXI_RESP	00b	R	These bits represent the slave response status, in AXI Master transaction. These bits are updated, in causing CHx_ERR error. Holds these bits until CHx_ERR error is cleared. 00b: Initial value 01b: Unsuspended Request 10b: SLVERR 11b: DECERR

Note: CHx_ERR (x = 7 to 0).

34.3.1.69 AXI Window Base (Lower) Registers (H'1000 + Offset: H'00/H'20/H'40/…/H'E0)

This register is Window setting register for Lower address translation when accessing AXI from PCIe.

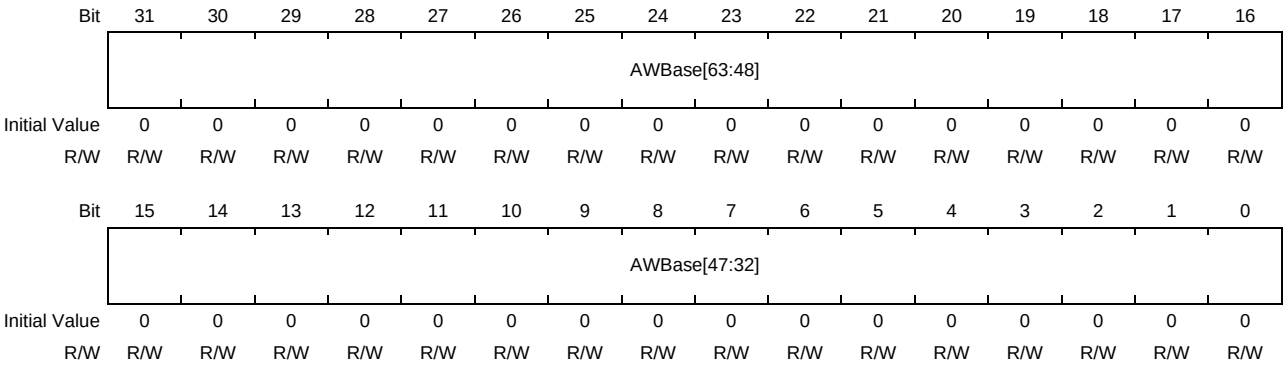
Sets the base point for PCIe side addresses. The configurable area is a 4K boundary.



Bit	Bit Name	Initial Value	R/W	Description
31 to 12	AWBase	All 0	R/W	Window setting register for address translation when accessing AXI from PCIe. The configurable area is a 4K boundary.
11 to 4	AWBase	All 0	R	H'00 fixed
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	AXI Window Enable	0b	R/W	Enable setting for the AXI Window. 0: Window disabled 1: Window enabled

34.3.1.70 AXI Window Base (Upper) Registers (H'1000 + Offset: H'04/H'24/H'44/…/H'E4)

This register is Window setting register for Upper address translation when accessing AXI from PCIe.

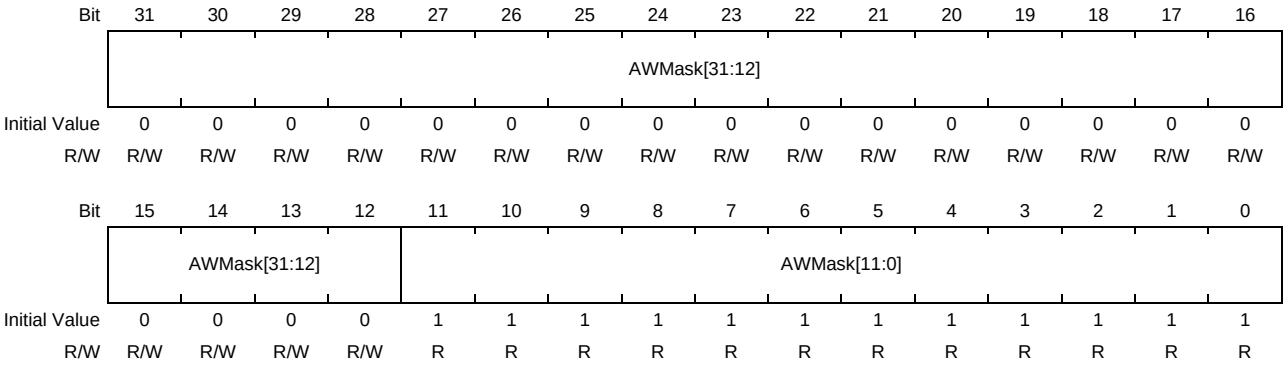


Bit	Bit Name	Initial Value	R/W	Description
31 to 0	AWBase	All 0	R/W	Window setting register for address translation when accessing AXI from PCIe.

34.3.1.71 AXI Window Mask (Lower) Registers (H'1000 + Offset: H'08/H'28/H'48/…/H'E8)

This register is Window setting register for Lower address translation when accessing AXI from PCIe.

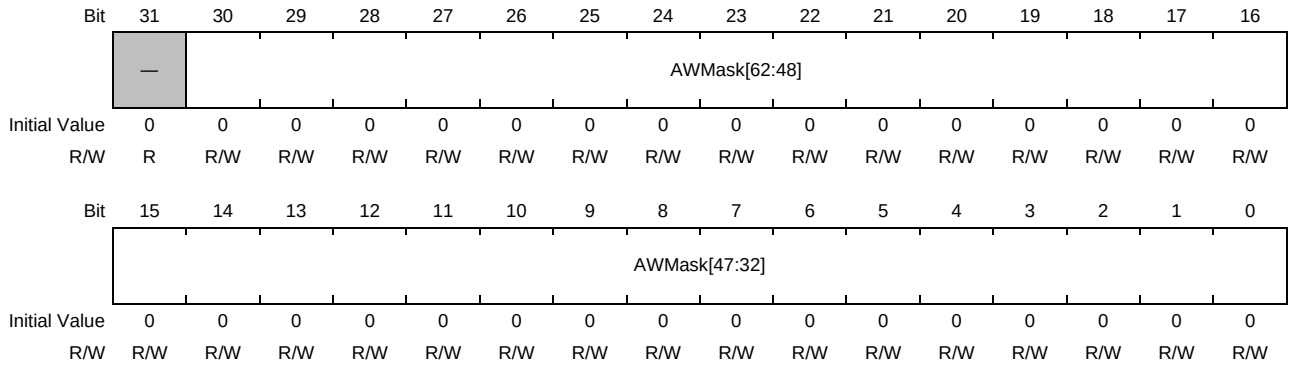
Set the Window from the address set in the AWBase register to the area of the set number of bits. The configurable area is 4K x 2N bytes.



Bit	Bit Name	Initial Value	R/W	Description
31 to 12	AWMask	All 0	R/W	Set the Window from the address set in the AWBase register to the area of the set number of bits. The setting is set from the low order bit to 1. Therefore, the configurable area is 4K × 2N bytes.
11 to 0	AWMask	All 1	R	H'FFF fixed

34.3.1.72 AXI Window Mask (Upper) Registers (H'1000 + Offset: H'0C/H'2C/H'4C/…/H'EC)

This register is Window setting register for Upper address translation when accessing AXI from PCIe.

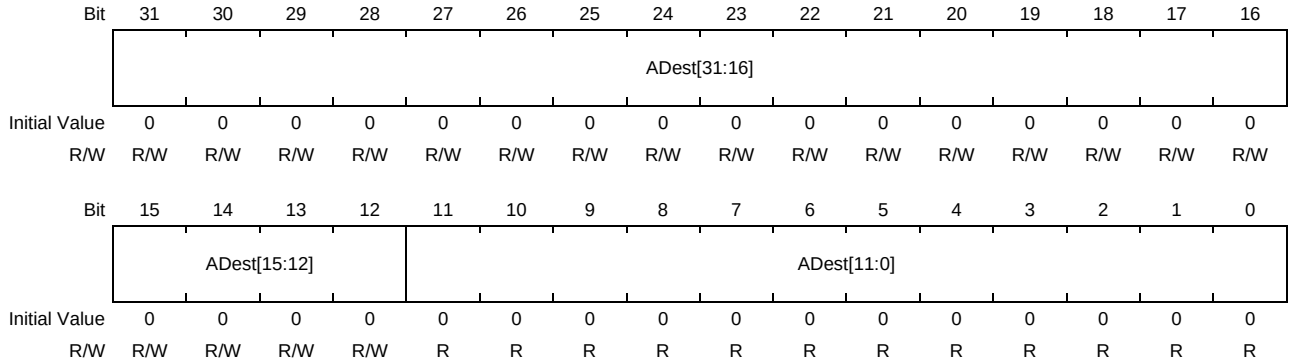


Bit	Bit Name	Initial Value	R/W	Description
31	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 0	AWMask	All 0	R/W	Set the Window from the address set in the AWBase register to the area of the set number of bits. The setting is set from the low order bit to 1.

34.3.1.73 AXI Destination (Lower) Registers (H'1000 + Offset: H'10/H'30/H'50/H'70/…/H'F0)

This register is Window setting register for Lower address translation when accessing AXI from PCIe.

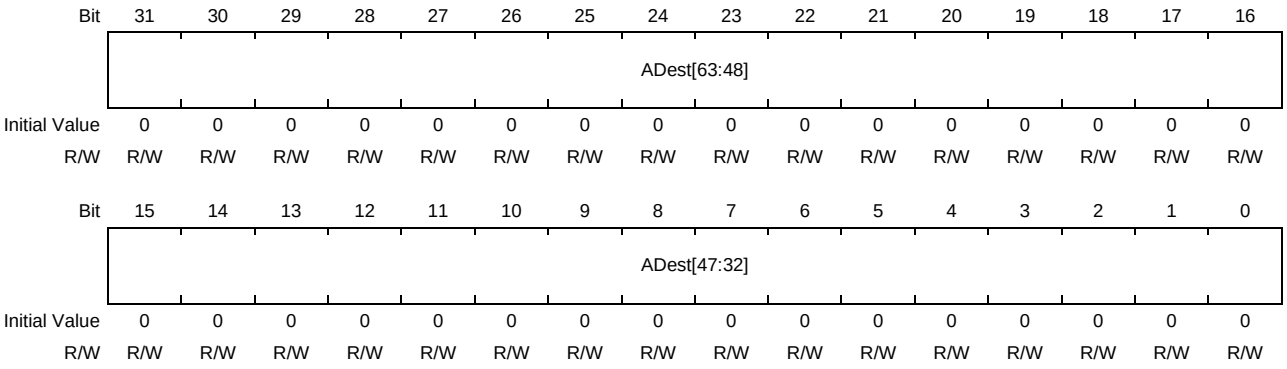
From the AXI side, set the Window point in the address space. The configurable area is a 4K boundary.



Bit	Bit Name	Initial Value	R/W	Description
31 to 12	ADest	All 0	R/W	From the AXI side, set the Window point in the address space. The configurable area is a 4K boundary.
11 to 0	ADest	All 0	R	H'000 fixed

34.3.1.74 AXI Destination (Upper) Registers (H'1000 + Offset: H'14/H'34/H'54/H'74/…/H'F4)

This register is Window setting register for Upper address translation when accessing AXI from PCIe.

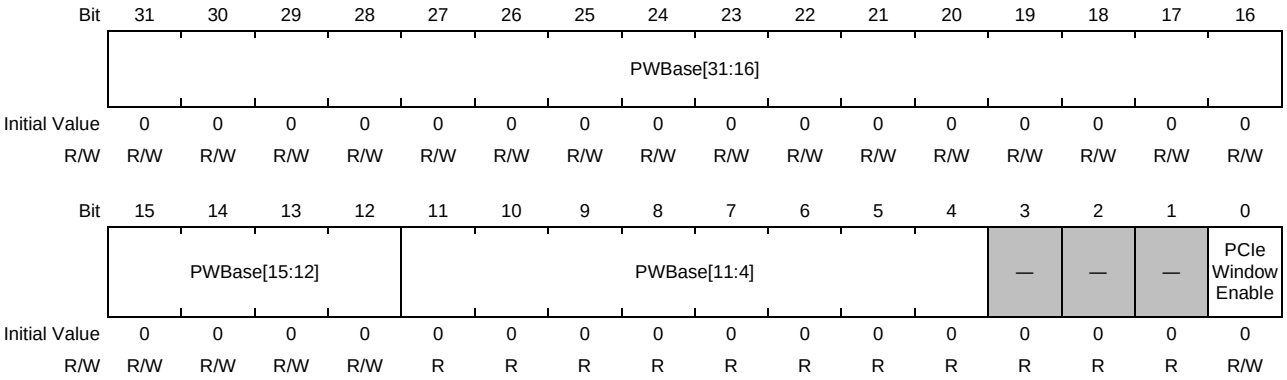


Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ADest	All 0	R/W	From the AXI side, set the Window point in the address space.

34.3.1.75 PCIe Window Base (Lower) Registers (H'1000 + Offset: H'100/H'120/H'140/…/H'1E0)

This register is Window setting register for Lower address translation when accessing PCIe from AXI.

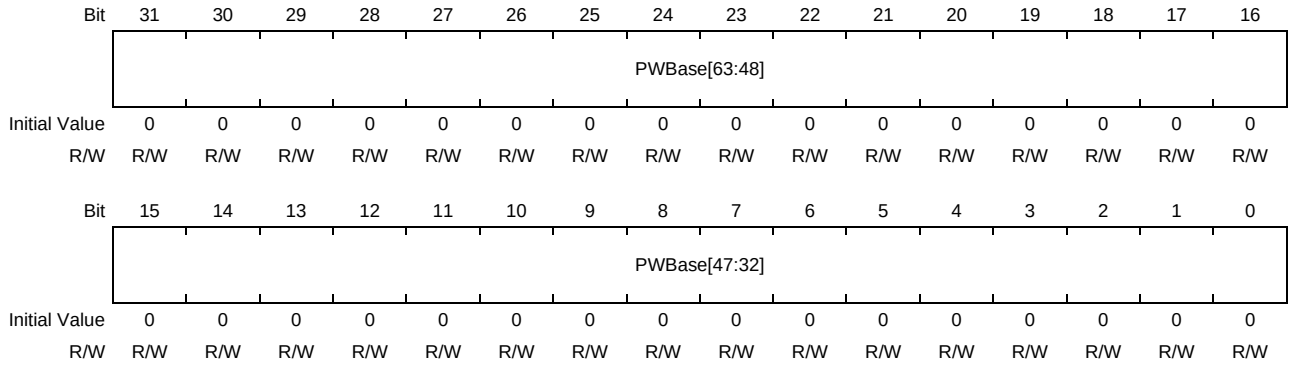
Sets the base point for AXI side addresses. The configurable area is a 4K boundary.



Bit	Bit Name	Initial Value	R/W	Description
31 to 12	PWBase	All 0	R/W	Sets the base point for AXI side addresses. The configurable area is a 4K boundary.
11 to 4	PWBase	All 0	R	H'00 fixed
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PCIe Window Enable	0b	R/W	PCIe Window Enable setting. 0: Window disabled 1: Window enabled

34.3.1.76 PCIe Window Base (Upper) Registers (H'1000 + Offset: H'104/H'124/H'144/…/H'1E4)

This register is Window setting register for Upper address translation when accessing PCIe from AXI.

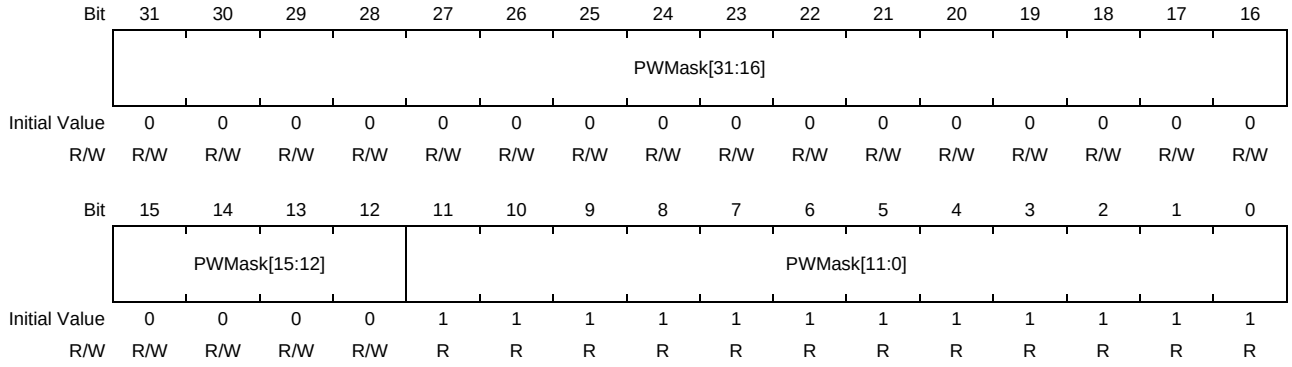


Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PWBase	All 0	R/W	Sets the base point for AXI side addresses. The configurable area is a 4K boundary.

34.3.1.77 PCIe Window Mask (Lower) Registers (H'1000 + Offset: H'108/H'128/H'148/…/H'1E8)

This register is Window setting register for Lower address translation when accessing PCIe from AXI.

Set the Window from the address set in the PWBase register to the area of the set number of bits.

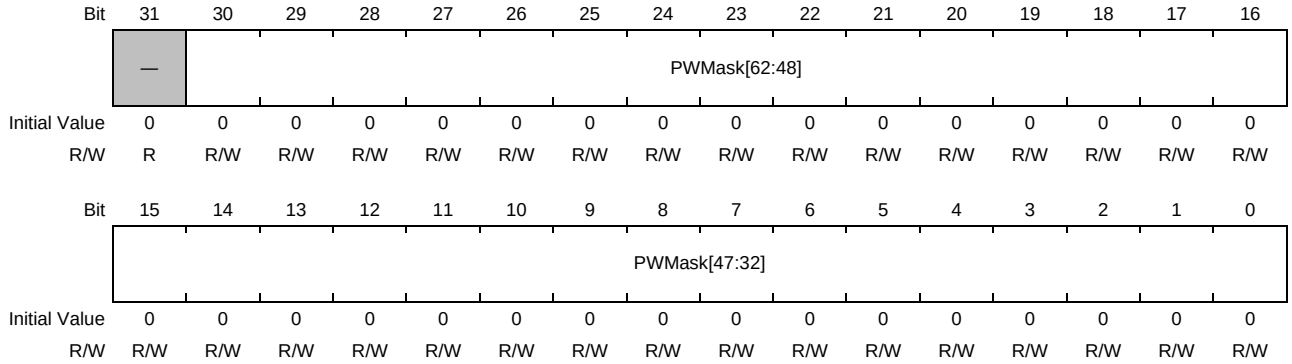


Bit	Bit Name	Initial Value	R/W	Description
31 to 12	PWMask	All 0	R/W	Set the Window from the address set in the PWBase register to the area of the set number of bits. The setting is set from the low order bit to 1.
11 to 0	PWMask	All 1	R	H'FFF fixed

34.3.1.78 PCIe Window Mask (Upper) Registers (H'1000 + Offset: H'10C/H'12C/H'14C/…/H'1EC)

This register is Window setting register for Upper address translation when accessing PCIe from AXI.

Set the Window from the address set in the PWBase register to the area of the set number of bits.

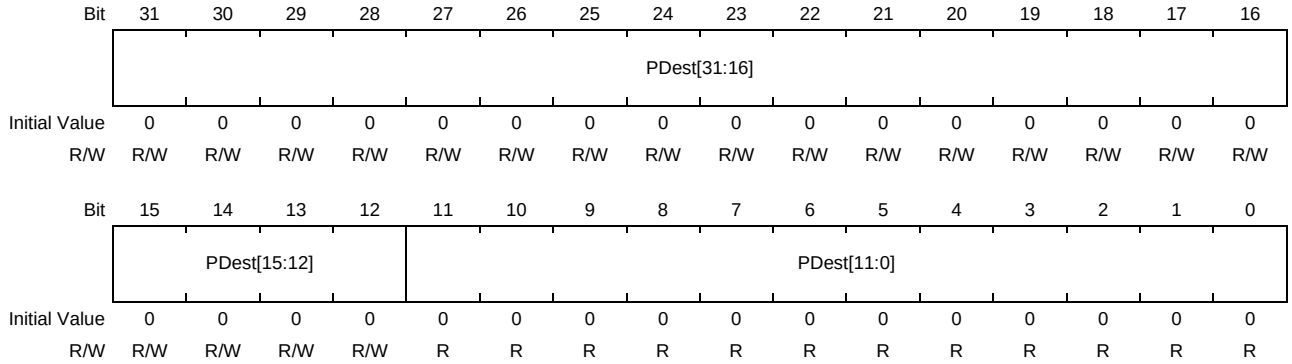


Bit	Bit Name	Initial Value	R/W	Description
31	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 0	PWMask	All 0	R/W	Set the Window from the address set in the PWBase register to the area of the set number of bits. The setting is set from the low order bit to 1.

34.3.1.79 PCIe Destination (Lower) Registers (H'1000 + Offset: H'110/H'130/H'150/H'170/…/H'1F0)

This register is Window setting register for Upper address translation when accessing PCIe from AXI.

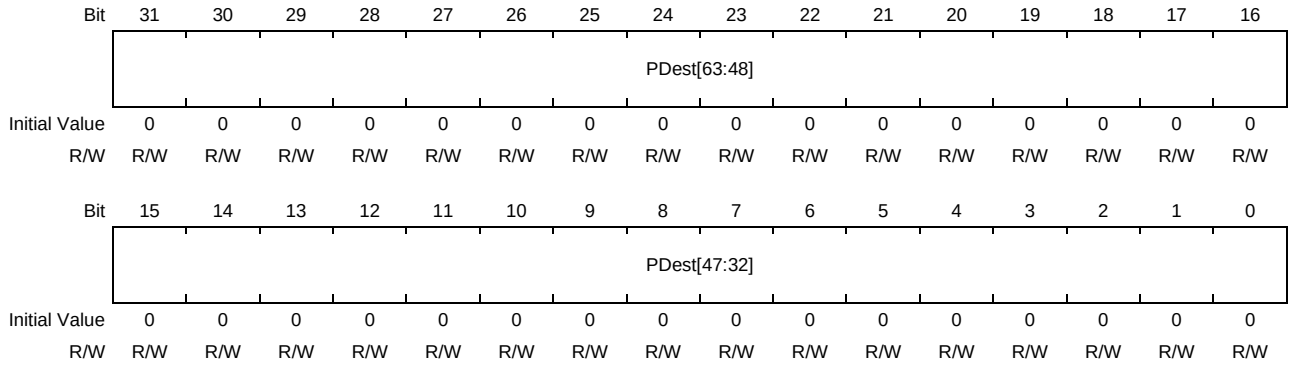
Set the Window point in the PCIe side address space. The configurable area is a 4K boundary.



Bit	Bit Name	Initial Value	R/W	Description
31 to 12	PDest	All 0	R/W	Set the Window point in the PCIe side address space. The configurable area is a 4K boundary.
11 to 0	PDest	All 0	R	H'000 fixed

34.3.1.80 PCIe Destination (Upper) Register (H'1000 + Offset: H'114/H'134/H'154/H'174/…/H'1F4)

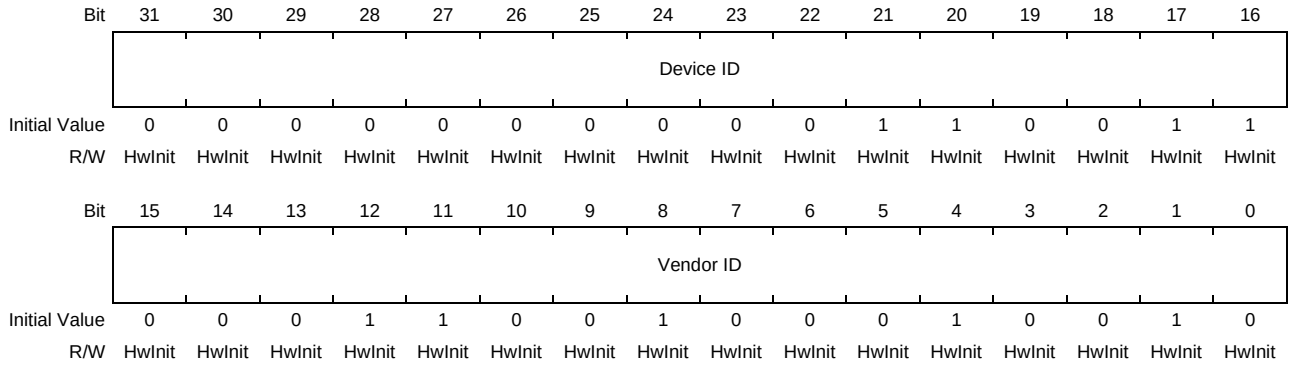
This register is Window setting register for Upper address translation when accessing PCIe from AXI.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PDest	All 0	R/W	Set the Window point in the PCIe side address space.

34.3.2 PCI Express Configuration Registers (Type1)

34.3.2.1 Vendor and Device ID (Offset: H'6000)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 16	Device ID	H'0033	HwInit	RST_LOAD_B	Indicates the manufacturer of the device. Set a fixed value.
15 to 0	Vendor ID	H'1912	HwInit	RST_LOAD_B	Used to identify the device manufactured by the manufacturer specified by Vendor ID. Set a fixed value.

34.3.2.2 Command and Status (Offset: H'6004)

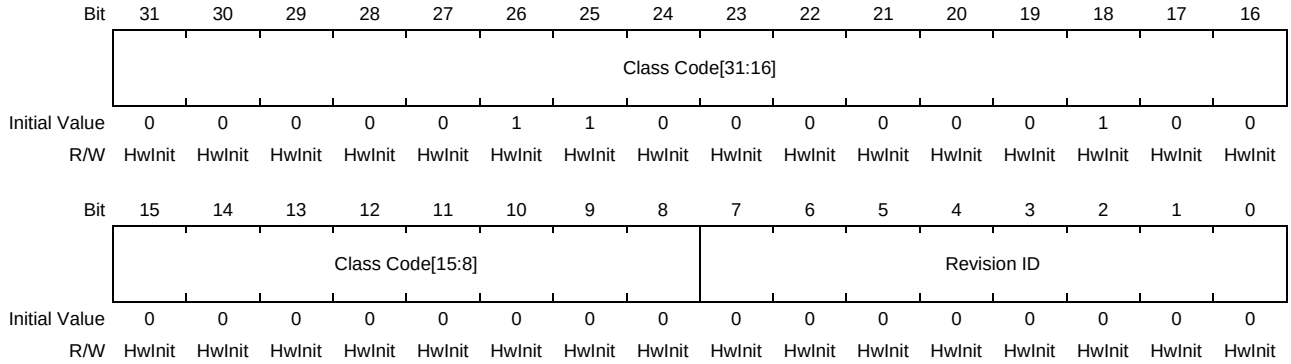
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Detected Parity Error	Signaled System Error	Received Master Abort	Received Target Abort	Signaled Target Abort	DEVSEL Timing		Master Data Parity Error	—	—	—	Capabilities List	Interrupt Status	—	—	Immediate Readiness
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R/W1C	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	Interrupt Disable	—	SERR# Enable	—	Parity Error Response	—	—	—	Bus Master Enable	Memory Space Enable	I/O Space Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R	R/W	R	R/W	R	R	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31	Detected Parity Error	0b	R	—	Not applicable for Root Complex devices. 0b fixed.
30	Signaled System Error	0b	R/W1C	RST_CFG_B	The SERR Enable bit is set to 1b when this macro sends an ERR_FATAL or ERR_NONFATAL Message.
29	Received Master Abort	0b	R	—	Not applicable for Root Complex devices. 0b fixed.
28	Received Target Abort	0b	R	—	Not applicable for Root Complex devices. 0b fixed.
27	Signaled Target Abort	0b	R	—	Not applicable for Root Complex devices. 0b fixed.
26, 25	DEVSEL Timing	00b	R	—	Reserved Does not apply to PCI Express.
24	Master Data Parity Error	0b	R	—	Not applicable for Root Complex devices. 0b fixed.
23 to 21	—	000b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	Capabilities List	1b	R	RST_CFG_B	All PCI Express devices must implement PCI Express Capability, so this bit is fixed at 1b.
19	Interrupt Status	0b	R	RST_CFG_B	Indicates the interrupt status of the device.
18, 17	—	00b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	Immediate Readiness	0b	R	—	Fixed to 0b.
15 to 11	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	Interrupt Disable	0b	R/W	RST_CFG_B	Suppresses the sending of Assert_INTx Messages. Not used for Root Complex devices.
9	—	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.

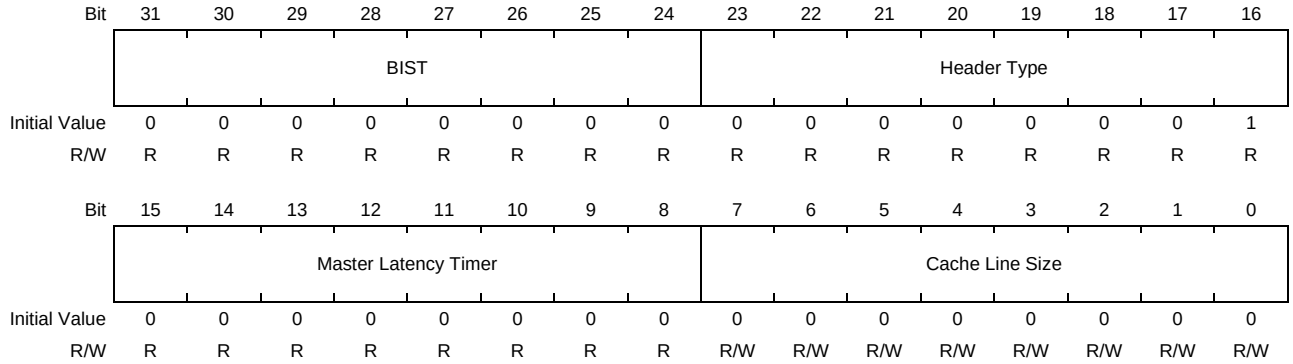
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
8	SERR# Enable	0b	R/W	RST_CFG_B	When this bit is set to “1b”, Non-Fatal Errors and Fatal Errors are notified to the Root Complex by Message Transaction. Note: Even if this bit is not set, if the Error Reporting-related bit in the Device Control register of PCI Express Capability is set to “1b”, Message If this bit is not set, the Root Complex is notified of the error by means of a Message Transaction.
7	—	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	Parity Error Response	0b	R/W	RST_CFG_B	Controls the behavior when a Poisoned TLP is sent or received. Note: The Detected Parity Error field in the Status register, the Device Status register in the PCI Express Capability, and the Uncorrectable Error in the Advanced Error Reporting Capability Error Logging to the Status register is performed regardless of the setting of this bit.
5 to 3	—	000b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	Bus Master Enable	0b	R/W	RST_CFG_B	Controls whether the device acts as a bus master.
1	Memory Space Enable	0b	R/W	RST_CFG_B	Controls whether to respond to accesses to memory space.
0	I/O Space Enable	0b	R	—	Does not respond to I/O space accesses. 0b fixed

34.3.2.3 Revision ID and Class Code (Offset: H’6008)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 8	Class Code	H’06_0400	HwInit	RST_LOAD_B	Information indicating the type and function of the device, the definition of the value is divided into the following two bytes by the PCI SIG: Bit [31:24]: Base class (H’06) Bit [23:16]: Sub-class (H’04) Bit [15:8]: Programming interface (H’00) Set a fixed value. The initial value is parameter settable.
7 to 0	Revision ID	All 0	HwInit	RST_LOAD_B	An 8-bit ID used to represent the revision of a particular device identified by its Vendor ID and Device ID. Set a fixed value. The initial value is parameter settable.

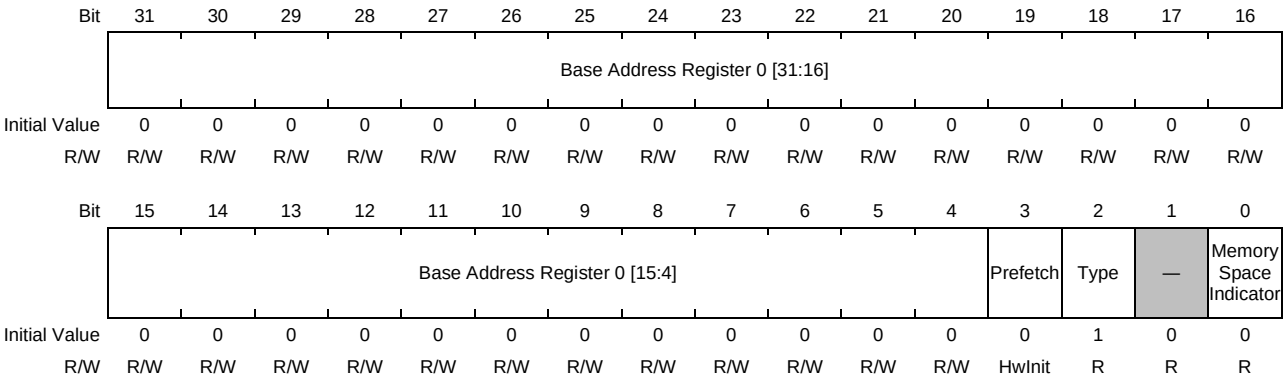
34.3.2.4 Cache Line and Header Type (Offset: H'600C)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 24	BIST	H'00	R	—	It does not support the BIST register function, so it is fixed at H'00.
23 to 16	Header Type	H'01	R	—	Root Complex: H'01 fixed.
15 to 8	Master Latency Timer	H'00	R	—	Master delay timer register not used in PCI Express, so it is fixed at H'00.
7 to 0	Cache Line Size	H'00	R/W	RST_CFG_B	Implemented as a read-write field for legacy compatibility, but the value set has no effect on this device.

34.3.2.5 Base Address Register 0 (Offset: H'6010)

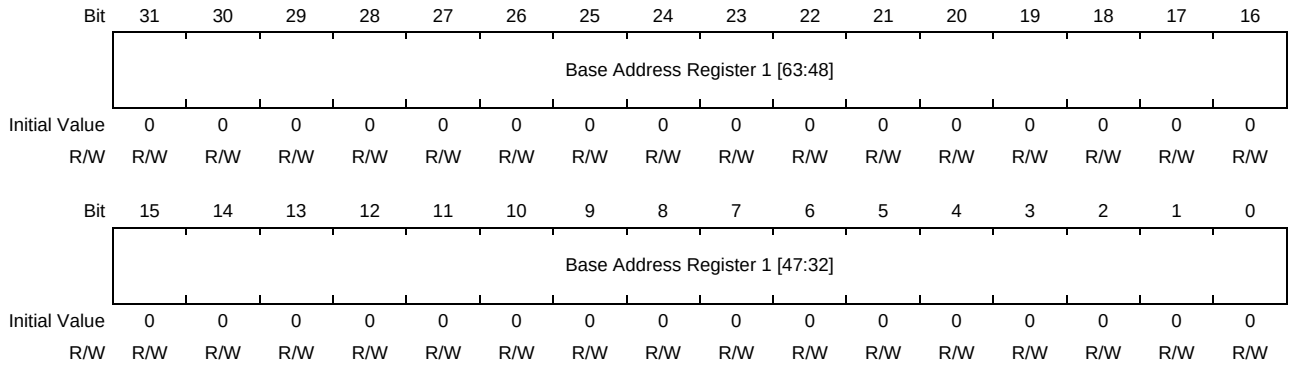
The combination with Base Address Register 1 (BAR1) constitutes a 64 bits Memory Space.



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 4	Base Address Register 0	All 0	R/W	RST_CFG_B	It indicates the base address. Depending on the size of the address block required, some low-order bits of this field should be implemented as 0b fixed Read Only bits. In this macro, the Read Only bit can be set in the Base Address 0 Mask register (Offset: H'0A0).
3	Prefetch	0b	HwInit	RST_LOAD_B	The Prefetchable Memory Base and Prefetchable Memory Limit registers must indicate that 64-bit addresses are supported, as defined in PCI-to-PCI Bridge Architecture Specification. 0: Disable 1: Enable
2	Type	1b	R	—	The 64-bit addresses are supported, as defined in PCI-to-PCI Bridge Architecture Specification. 0b: 32bit address 1b: 64bit address 1b: Fixed to use 64bit Address
1	—	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	Memory Space Indicator	0b	R	—	0b fixed to indicate memory space.

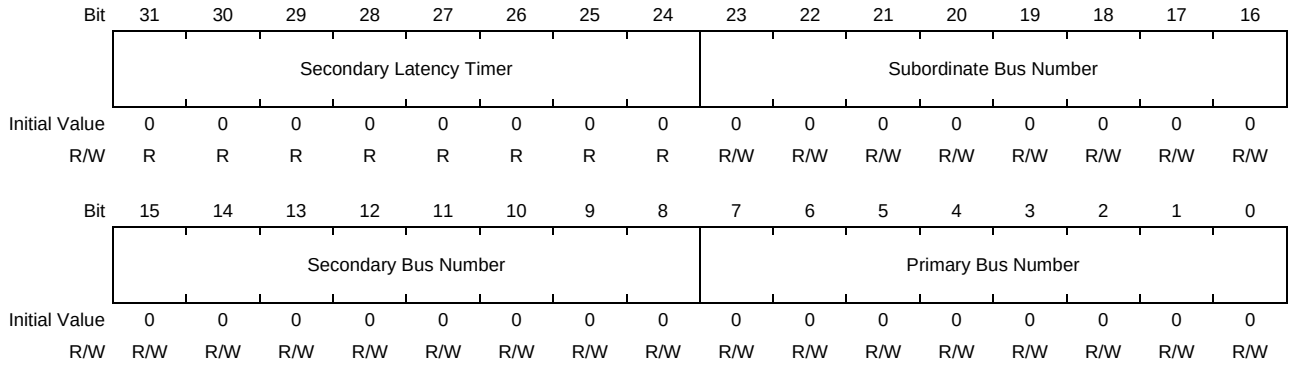
34.3.2.6 Base Address Register 1 (Offset: H'6014)

The combination with Base Address Register 0 (BAR0) constitutes a 64 bits Memory Space.



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 0	Base Address Register 1	All 0	R/W	RST_CFG_B	Base Address Register 1 (64bit Upper Address) It indicates the high-order 32 bits of the base address.

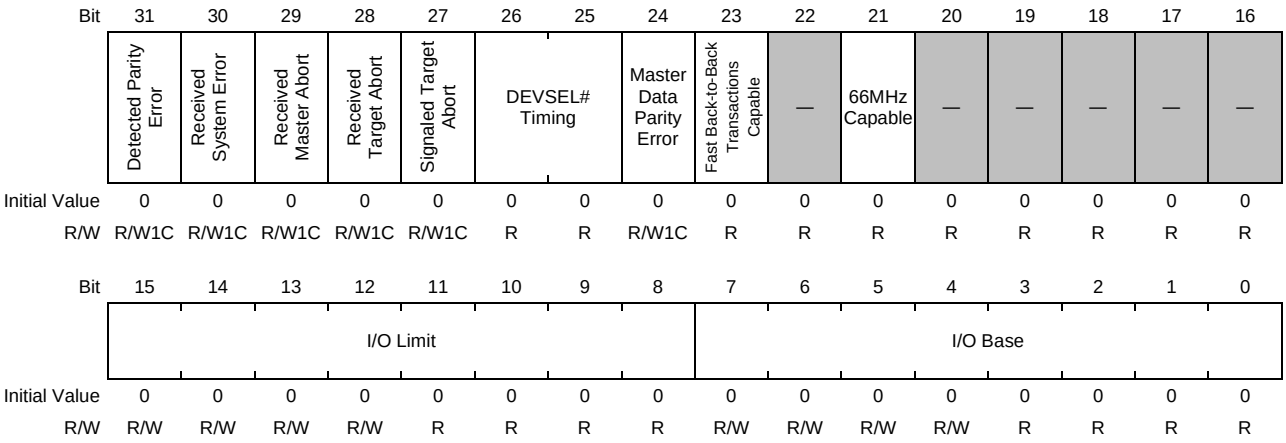
34.3.2.7 Bus Number Register (Offset: H'6018)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 24	Secondary Latency Timer	H'00	R	—	PCI Express does not use H'00 fixed
23 to 16	Subordinate Bus Number	H'00	R/W	RST_CFG_B	Set the Subordinate Bus Number. Default: H'00
15 to 8	Secondary Bus Number	H'00	R/W	RST_CFG_B	Set the Subordinate Bus Number. Default: H'00
7 to 0	Primary Bus Number	H'00	R/W	RST_CFG_B	Set the Primary Bus Number. Default: H'00

34.3.2.8 I/O Base/Limit and Secondary Status (Offset: H'601C)

Do not change the initial value.

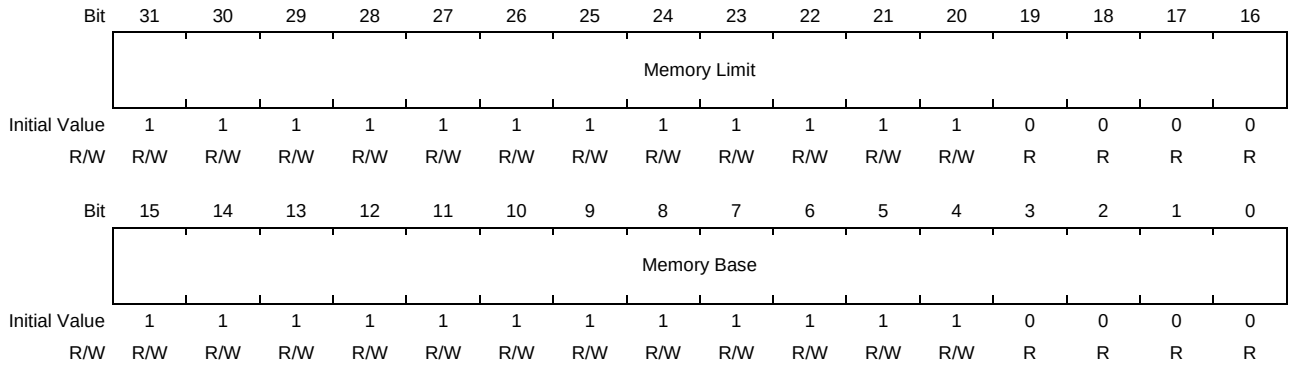


Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31	Detected Parity Error	0b	R/W1C	RST_CFG_B	It is set to 1b when a Poisoned TLP is received, regardless of the setting of the Parity Error Response Enable bit in the Bridge Control and Interrupt Register (Offset: H'03C).
30	Received System Error	0b	R/W1C	RST_CFG_B	It is set to 1b when ERR_FATAL or ERR_NONFATAL Message is received.
29	Received Master Abort	0b	R/W1C	RST_CFG_B	The Completion Status field is set to 1b when Completion of an Unsupported Request is received.
28	Received Target Abort	0b	R/W1C	RST_CFG_B	The Completion Status field is set to 1b when Completion of Completer Abort is received.
27	Signaled Target Abort	0b	R/W1C	RST_CFG_B	The Completion Status field is set to 1b when a Completion of Completer Abort (Posted or Non-Posted Request) is sent.
26, 25	DEVSEL# Timing	00b	R	—	PCI Express does not use it, so it is fixed at 00b.
24	Master Data Parity Error	0b	R/W1C	RST_CFG_B	The Parity Error Response bit is set to 1b and is set to 1b when the following two conditions occur: 1) The Requester (BME) received Completion TLP of Poisoned. 2) The Requester (BME) sent the Write Request TLP of Poisoned. If the Parity Error Response bit is 0b, this bit is not set to 1b.
23	Fast Back-to-Back Transactions Capable	0b	R	—	PCI Express does not use it 0b fixed
22	—	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	66MHz Capable	0b	R	—	PCI Express does not use it 0b fixed
20 to 16	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 12	I/O Limit	All 0	R/W	—	This field is not used. When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
11 to 8		All 0	R	—	This field is not used. When read, the initial value is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
7 to 4	I/O Base	All 0	R/W	—	This field is not used. When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3 to 0		All 0	R	—	This field is not used. When read, the initial value is read. The written value will be ignored.

34.3.2.9 Memory Base/Limit Register (Offset: H’6020)

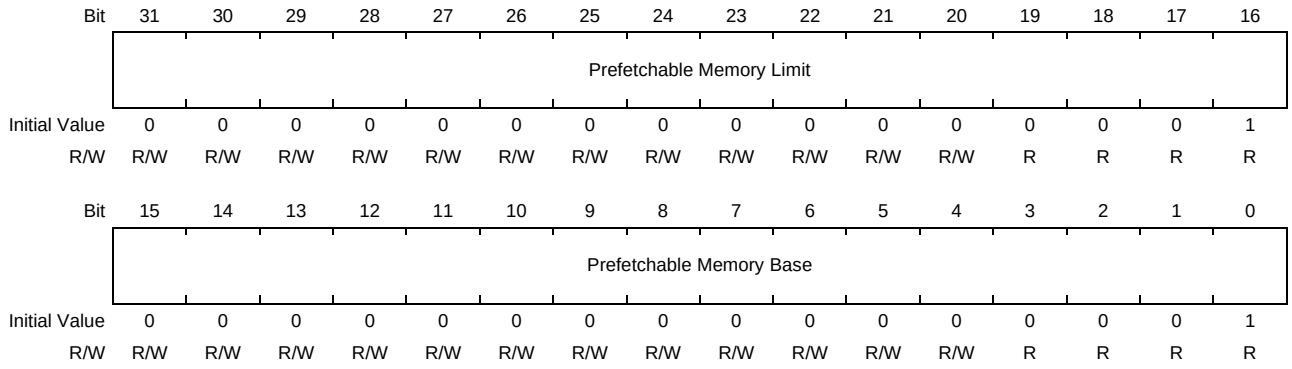
Do not change the initial value.



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 20	Memory Limit	H’FFF0	R/W	RST_CFG_B	The Memory Limit register, define a memory mapped address range which is used by the bridge to determine when to forward memory transactions from one interface to the other. Initial value: H’FFF0
19 to 16			R		
15 to 4	Memory Base	H’FFF0	R/W	RST_CFG_B	The Memory Base register, define a memory mapped address range which is used by the bridge to determine when to forward memory transactions from one interface to the other. Initial value: H’FFF0
3 to 0			R		

34.3.2.10 Prefetchable Memory Base/ Limit Register (Offset: H’6024)

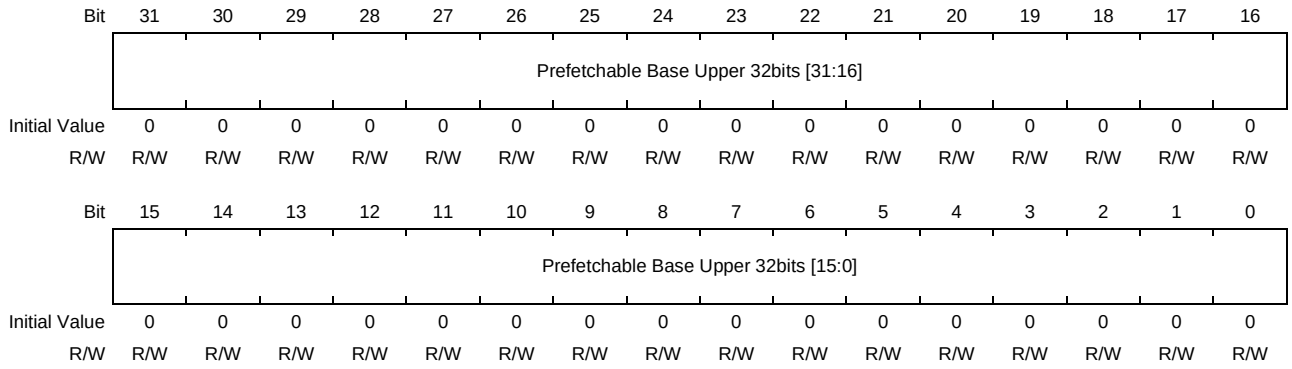
Do not change the initial value.



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 20	Prefetchable Memory Limit	H’0001	R/W	RST_CFG_B	The Prefetchable Memory Limit register, define a prefetchable memory address range which is used by the bridge to determine when to forward memory transactions from one interface to the other. These are optional. Initial value: H’0001
19 to 16			R		
15 to 4	Prefetchable Memory Base	H’0001	R/W	RST_CFG_B	The Prefetchable Memory Base register, define a prefetchable memory address range which is used by the bridge to determine when to forward memory transactions from one interface to the other. These are optional. Initial value: H’0001
3 to 0			R		

34.3.2.11 Prefetchable Base Upper 32Bits Registers (Offset: H’6028)

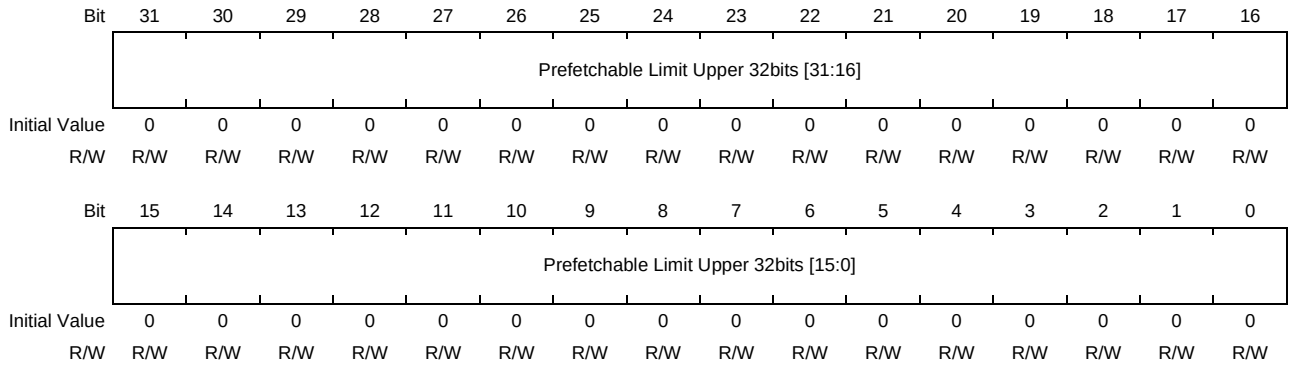
Do not change the initial value.



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 0	Prefetchable Base Upper 32bits	All 0	R/W	RST_CFG_B	These registers are optional extensions to the Prefetchable Base Upper 32 bits registers. Initial value: H’0000_0000

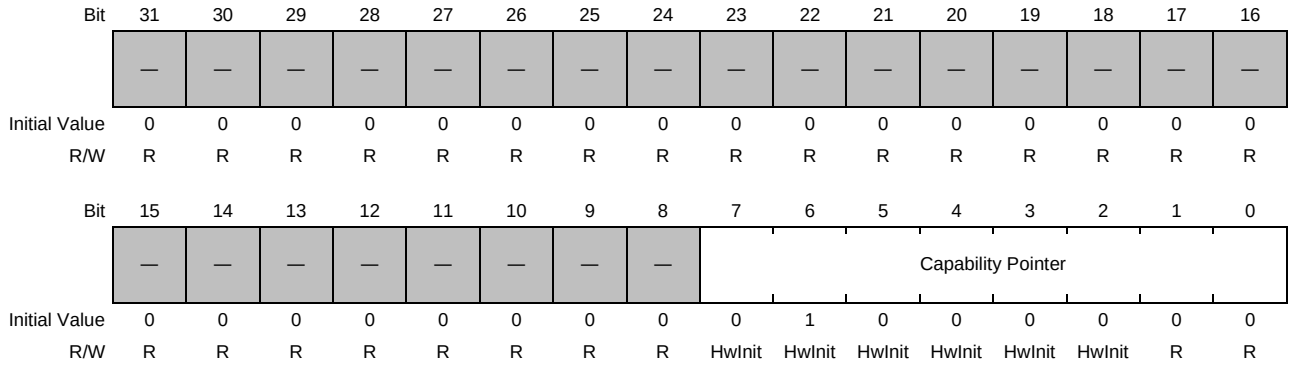
34.3.2.12 Prefetchable Limit Upper 32Bits (Offset: H'602C)

Do not change the initial value.



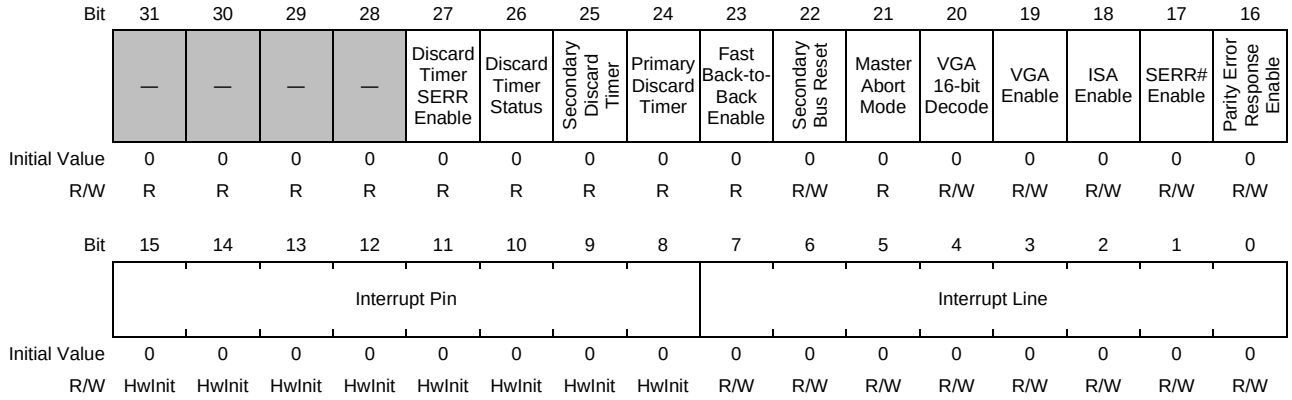
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 0	Prefetchable Limit Upper 32bits	All 0	R/W	RST_CFG_B	These registers are optional extensions to the Prefetchable Limit Upper 32 bits registers. Initial value: H'0000_0000

34.3.2.13 Capability Pointer (Offset: H'6034)



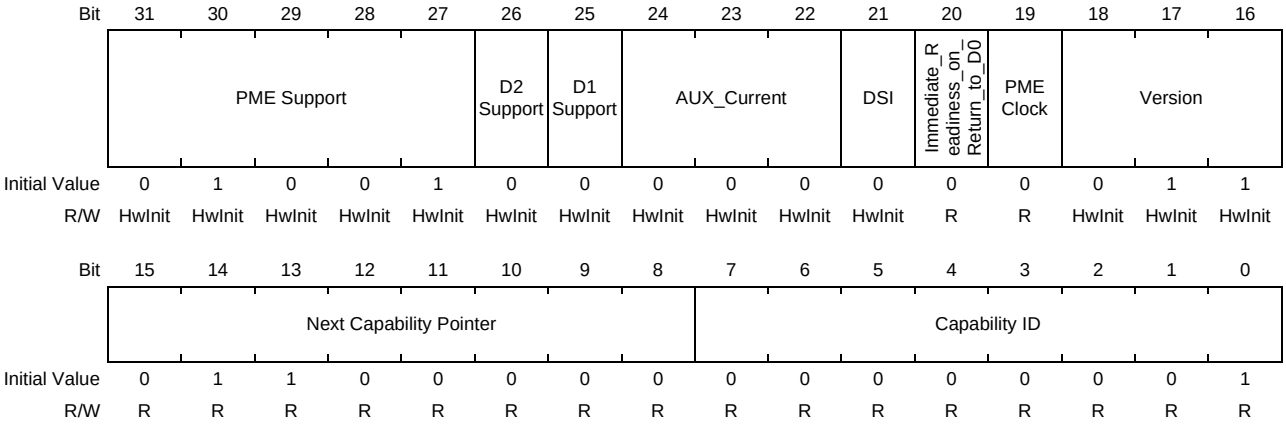
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 8	—	All 0	R	RST_CFG_B	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 2	Capability Pointer	H'40	HwInit	RST_CFG_B	Capability Implementation Start Address H'40
1, 0			R		PCI Power Management Capability has been implemented since H'40. The lower 2 bits are 00b fixed (reserved) and cannot be written even from the UDL side.

34.3.2.14 Bridge Control and Interrupt (Offset: H’603C)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 28	—	H’0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27	Discard Timer SERR Enable	0b	R	—	Reserved: Does not apply to PCI Express. 0b fixed
26	Discard Timer Status	0b	R	—	Reserved: Does not apply to PCI Express. 0b fixed
25	Secondary Discard Timer	0b	R	—	Reserved: Does not apply to PCI Express. 0b fixed
24	Primary Discard Timer	0b	R	—	Reserved: Does not apply to PCI Express. 0b fixed
23	Fast Back-to-Back Enable	0b	R	—	Reserved: Does not apply to PCI Express. 0b fixed
22	Secondary Bus Reset	0b	R/W	RST_CFG_B	Transition to Hot Reset state with 1b write
21	Master Abort Mode	0b	R	—	Reserved: Does not apply to PCI Express. 0b fixed
20	VGA 16-bit Decode	0b	R/W	RST_CFG_B	Initial value 0b It is not used in this macro.
19	VGA Enable	0b	R/W	RST_CFG_B	Initial value 0b It is not used in this macro.
18	ISA Enable	0b	R/W	RST_CFG_B	Initial value 0b It is not used in this macro.
17	SERR# Enable	0b	R/W	RST_CFG_B	1b Write enables INT_SERR * interrupt terminal notification
16	Parity Error Response Enable	0b	R/W	RST_CFG_B	1b Set the Master Data Parity Error bit in the Secondary Status register when Poisoned TLP is received by writing.
15 to 8	Interrupt Pin	H’00	HwInit	RST_LOAD_B	The Interrupt Pin register is a read-only register that identifies the legacy interrupt Message(s) the Function uses H’00 fixed.
7 to 0	Interrupt Line	H’00	R/W	RST_CFG_B	The Interrupt Line register communicates interrupt line routing information. H’00 fixed.

34.3.2.15 PM Capabilities (Offset: H'6040)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 27	PME Support	01001b	HwInit	RST_LOAD_B	Indicates with or without PME support of each Device State. xxxx1: Supports D0 xxx1x: Supports D1 xx1xx: Supports D2 x1xxx: Supports D3hot 1xxxx: Supports D3cold (Not supported)
26	D2 Support	0b	HwInit	RST_LOAD_B	Indicates with or without D2 Power Management State support. 0: Not Support 1: Support
25	D1 Support	0b	HwInit	RST_LOAD_B	Indicates with or without D1 Power Management State support. 0: Not Support 1: Support
24 to 22	AUX_Current	000b	HwInit	RST_LOAD_B	Indicates 3.3 Vaux auxiliary current (Maximum current value, supplied by the auxiliary power supply). 111b: 375 mA 110b: 320 mA 101b: 250 mA 100b: 220mA 011b: 160 mA 010b: 100 mA 001b: 55 mA 000b: 0 (self-powered) (Read value returns 000b. AUX not supported)
21	DSI	0b	HwInit	RST_LOAD_B	Indicates with or without to use DSI (Device Specific Initialization). 0: Not Support 1: Support
20	Immediate_Readiness_on_Return_to_D0	0b	R	—	This Function is guaranteed to be ready to successfully complete valid accesses immediately after being set to D0. 0b fixed
19	PME Clock	0b	R	—	Reserved: Does not apply to PCI Express. 0b fixed.
18 to 16	Version	011b	HwInit	RST_LOAD_B	PCI Power Management Interface Specification Rev.1.2 011b fixed

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
15 to 10	Next Capability Pointer	H'60	R	RST_LOAD_B	Indicates the PCI Express Capability starting address. This field provides an offset into the Function's Configuration Space pointing to the location of next item in the capabilities list. If there are no additional items in the capabilities list, this field is set to H'00.
9, 8				—	
7 to 0	Capability ID	H'01	R	—	Indicates PCI Power Management Capability. H'01 fixed

34.3.2.16 PM Status/Control (Offset: H'6044)

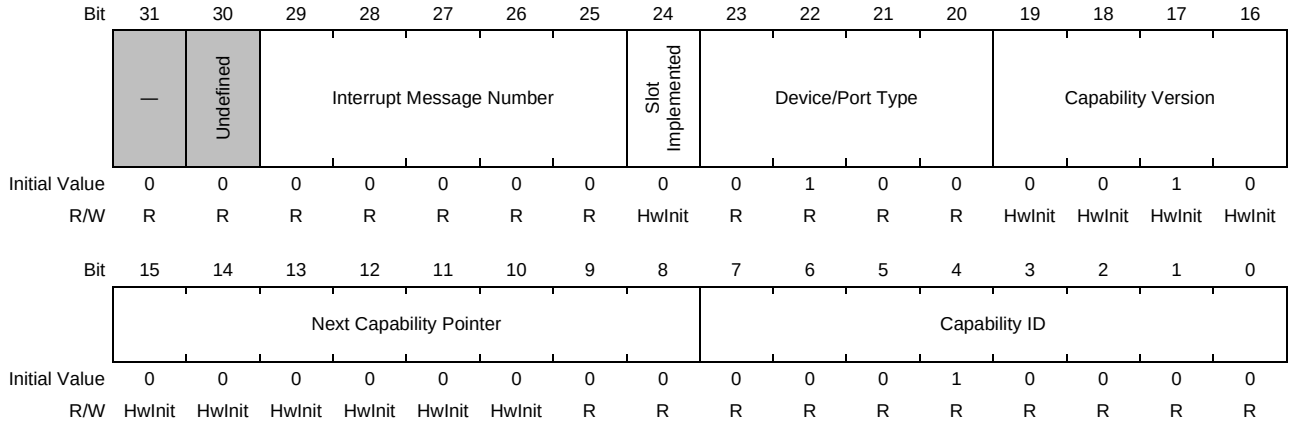
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Data								Bus Power/ Clock Control Enable	B2/B3 Support	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PME Status	Data Scale		Data Select				PME Enable	—	—	—	—	No_Soft _Reset	—	PowerState	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R/W1C	R	R	R	R	R	R	R/WS	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 24	Data	H'00	R	—	Does not support. Fixed H'00.
23	Bus Power/Clock Control Enable	0b	R	—	Does not support. Fixed 0b.
22	B2/B3 Support	0b	R	—	Does not support. Fixed 0b.
21 to 16	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15	PME Status	0b	R/W1C	—	Indicates that a PME assert factor is occurring. When this bit is 1b, indicates that there is a PME assert factor.
14,13	Data Scale	00b	R	—	Does not support. Fixed 00b.
12 to 9	Data Select	H'0	R	—	Does not support. Fixed H'0.
8	PME Enable	0b	R/WS	RST_RSM_B	Controls the assertion of the PME. When this bit is 1b, PME assertions are enabled. Assert PME if PME_Status is set. PCI Express performs Link wake-up processing and then PME assert processing by sending a PM_PME Message. *Note*: According to the value of PM Capabilities register PME Support [4] (PME Support in D3cold), the specifications are as follows. – PME Support [4] =1b Reset: RST_RSM_B PME – Support [4] = 0b Reset: RST_CFG_B
7 to 4	—	H'0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	No_Soft_Reset	1b	R	—	Indicates that no internal reset occurs inside the device during the power state transition from D3hot to D0.
2	—	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.

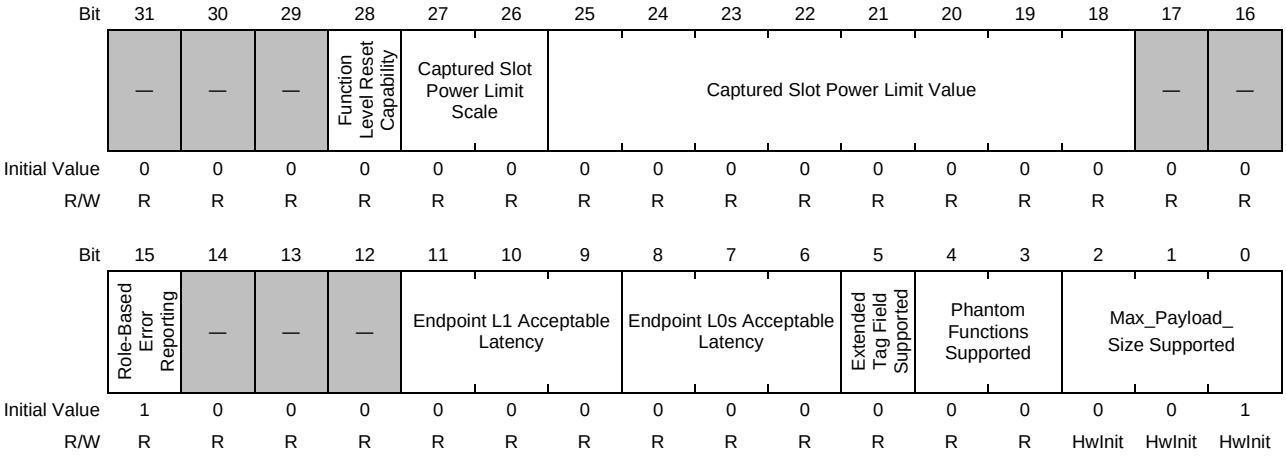
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
1 to 0	PowerState	00b	R/W	RST_CFG_B	Set the PCI Device State. 00b: D0 (Default) 01b: D1 (Setting prohibited) 10b: D2 (Setting prohibited) 11b: D3hot

34.3.2.17 PCI Express Capability (Offset: H'6060)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31	—	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30	Undefined	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29 to 25	Interrupt Message Number	All 0	R	—	This field indicates which MSI/MSI-X vector is used for the interrupt message generated in association with any of the status bits of this Capability structure. 00000b: fixed
24	Slot Implemented	0b	HwInit	RST_LOAD_B	Setting 1b to this bit, indicates that the PCI Express Link (Down Port) is connected to the Slot.
23 to 20	Device/Port Type	0100b	R	—	Indicates the PCI Express Root Complex device. 0000b: PCI Express Endpoint device 0001b: Legacy PCI Express Endpoint device 0100b: Root Port of PCI Express Root Complex (Default) 0101b: Upstream Port of PCI Express Switch 0110b: Downstream Port of PCI Express Switch 0111b: PCI Express-to-PCI/PCI-X Bridge 1000b: PCI/PCI-X-to-PCI Express Bridge 1001b: Root Complex Integrated Endpoint Device 1010b: Root Complex Event Collector All other encodings are reserved.
19 to 16	Capability Version	0010b	HwInit	—	Indicates the version of the PCI Express Capability Structure. 0010b: fixed
15 to 10	Next Capability Pointer	H'00	HwInit	RST_LOAD_B	Indicates that this Capability List is a final List (H'00 fixed) The lower two bits [9: 8] are Reserved and fixed at 00b.
9, 8			R	—	
7 to 0	Capability ID	H'10	R	—	Indicates the PCI Express Capability. H'10: fixed.

34.3.2.18 Device Capabilities (Offset: H'6064)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 29	—	000b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	Function Level Reset Capability	0b	R	—	It does not apply to Root Complex devices. 0b fixed.
27, 26	Captured Slot Power Limit Scale	00b	R	—	It does not apply to Root Complex devices. 00b fixed.
25 to 18	Captured Slot Power Limit Value	All 0	R	—	It does not apply to Root Complex devices. H'00 fixed.
17, 16	—	00b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15	Role-Based Error Reporting	1b	R	—	1b set if implementing the Error Reporting feature (compliant since Rev 1.1). 1b fixed in this core.
14 to 12	—	000b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 9	Endpoint L1 Acceptable Latency	000b	R	—	It does not apply to Root Complex devices. 000b fixed.
8 to 6	Endpoint L0s Acceptable Latency	000b	R	—	It does not apply to Root Complex devices. 000b fixed.
5	Extended Tag Field Supported	0b	R	—	Extended Tag support. 0b: 5-bit Tag field supported 1b: 8-bit Tag field supported 0b fixed in this core.
4, 3	Phantom Functions Supported	00b	R	—	Phantom Function is not supported. 00b fixed.

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
2 to 0	Max_Payload_Size Supported	001b	HwInit	RST_LOAD_B	This field indicates the maximum payload size that the Function can support for TLPs. 000b: 128 bytes max payload size 001b: 256 bytes max payload size (Default) 010b: 512 bytes max payload size 011b: 1024 bytes max payload size 100b: 2048 bytes max payload size 101b: 4096 bytes max payload size 110b: Reserved 111b: Reserved

34.3.2.19 Device Control/Status (Offset: H'6068)

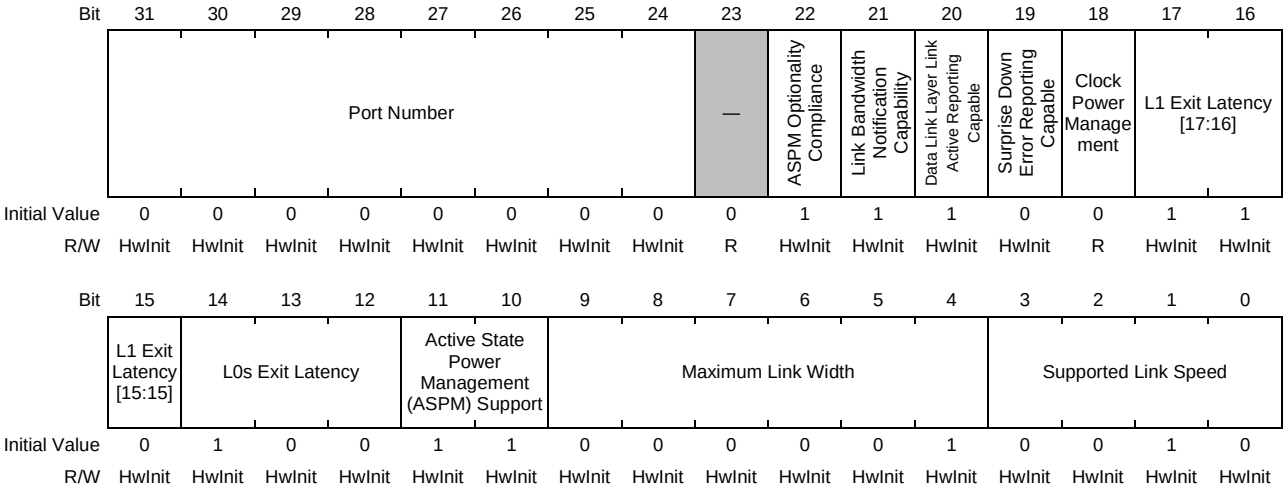
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	Emergency Power Reduction Detected	Transaction Pending	AUX Power Detected	Unsupported Request Detected	Fatal Error Detected	Non-Fatal Error Detected	Correctable Error Detected
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W1C	R/W1C	R/W1C	R/W1C

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Initiate Function Level Reset	Max_Read_Request_Size			Enable No Snoop	Auxiliary (AUX) Power PM Enable	Phantom Functions Enable	Extended Tag Field Enable	Max_Payload_Size			Enable Relaxed Ordering	Unsupported Request Reporting Enable	Fatal Error Reporting Enable	Non-Fatal Error Reporting Enable	Correctable Error Reporting Enable
Initial Value	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 23	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	Emergency Power Reduction Detected	0b	R	—	Not supported 0b fixed.
21	Transaction Pending	0b	R	—	It does not apply to Root Complex devices. 0b fixed.
20	AUX Power Detected	0b	R	—	Indicates AUX Power is detected. Not supported 0b fixed
19	Unsupported Request Detected	0b	R/W1C	RST_CFG_B	Indicates that an Unsupported Request Error was detected. 1b has detected an error.
18	Fatal Error Detected	0b	R/W1C	RST_CFG_B	Indicates that a Fatal Error has been detected. 1b has detected an error.
17	Non-Fatal Error Detected	0b	R/W1C	RST_CFG_B	Indicates that a Non-Fatal error has been detected. 1b has detected an error.
16	Correctable Error Detected	0b	R/W1C	RST_CFG_B	Indicates that a Correctable Error has been detected. 1b has detected an error.
15	Initiate Function Level Reset	0b	R	—	A write of 1b initiates Function Level Reset to the Function. The value read by software from this bit is always 0b.
14 to 12	Max_Read_Request_Size	010b	R/W	RST_CFG_B	Set Max_Read_request_Size. 000b: 128 bytes max read request size 001b: 256 bytes max read request size 010b: 512 bytes max read request size (Default) 011b: 1024 bytes max read request size 100b: 2048 bytes max read request size 101b: 4096 bytes max read request size 110b: Reserved 111b: Reserved

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
11	Enable No Snoop	0b	R	—	This macro does not use No Snoop Attribute as a Requester.
10	Auxiliary (AUX) Power PM Enable	0b	R/W	RST_RSM_B	Set to with or without use Auxiliary (AUX) power. Not supported 0b fixed. 0: Not support (Default) 1: Support If this bit is set to 1b (Enable), AUX_Current (Power Management Capabilities register) must be set.
9	Phantom Functions Enable	0b	R	—	Phantom Function is not supported. 0b fixed.
8	Extended Tag Field Enable	0b	R/W	RST_CFG_B	Extended Tag is not supported. It uses only 5 Bit-Tag.
7 to 5	Max_Payload _Size	000b	R/W	RST_CFG_B	Set Max_Payload_Size. 000b: 128 bytes max payload size (Default) 001b: 256 bytes max payload size 010b: 512 bytes max payload size 011b: 1024 bytes max payload size 100b: 2048 bytes max payload size 101b: 4096 bytes max payload size 110b: Reserved 111b: Reserved
4	Enable Relaxed Ordering	1b	R/W	RST_CFG_B	Set to with or without Relaxed Ordering use as a Requester. 0: Not support 1: Support
3	Unsupported Request Reporting Enable	0b	R/W	RST_CFG_B	ERR_NONFATAL with Unsupported Request detection or control the generation of the ERR_FATAL Message. Message generation is enabled on 1b.
2	Fatal Error Reporting Enable	0b	R/W	RST_CFG_B	Controls the generation of the ERR_FATAL Message. Message generation is enabled on 1b.
1	Non-Fatal Error Reporting Enable	0b	R/W	RST_CFG_B	Controls the generation of the ERR_NONFATAL Message. Message generation is enabled on 1b.
0	Correctable Error Reporting Enable	0b	R/W	RST_CFG_B	Controls the generation of the ERR_COR Message. Message generation is enabled on 1b.

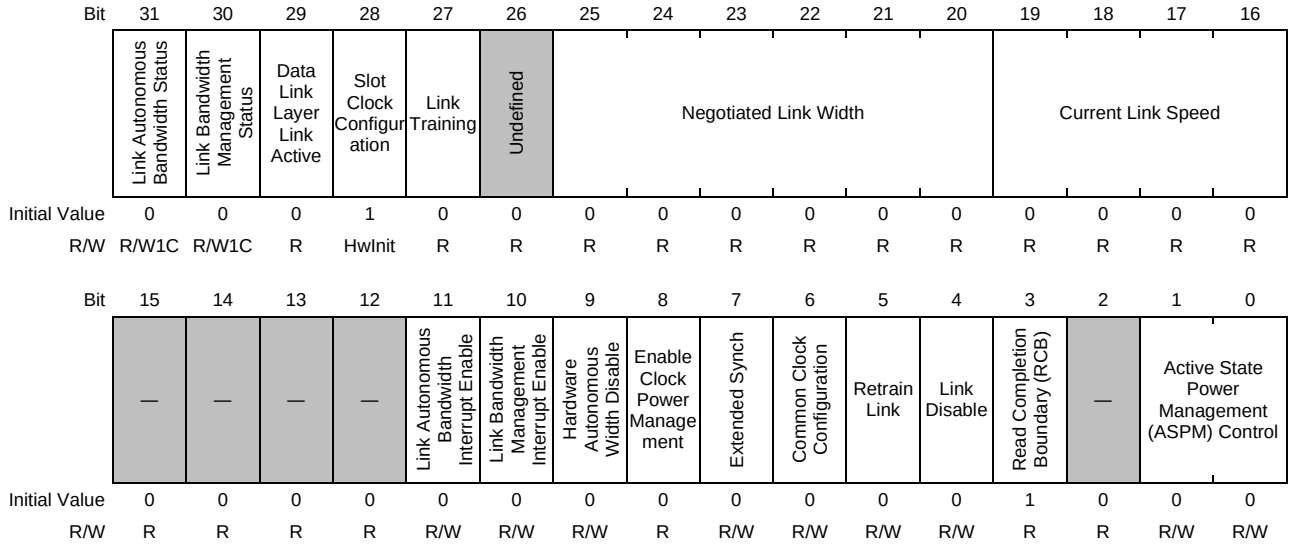
34.3.2.20 Link Capabilities (Offset: H'606C)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 24	Port Number	All 0	HwInit	—	Indicate PCI Express Link Port number.
23	—	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	ASPM Optionality Compliance	1b	HwInit	RST_LOAD_B	This bit must be set to 1b in all Functions. Components implemented against certain earlier versions of this specification will have this bit set to 0b. 1b fixed.
21	Link Bandwidth Notification Capability	1b	HwInit	RST_LOAD_B	Indicates support for Link Bandwidth Notification status and Interrupt features.
20	Data Link Layer Link Active Reporting Capable	1b	HwInit	RST_LOAD_B	Indicates support for the DL_Active state reporting feature in the Data Link Control and Management State Machine.
19	Surprise Down Error Reporting Capable	0b	HwInit	RST_LOAD_B	0b fixed. It does not support detection and reporting of Surprise Down error.
18	Clock Power Management	0b	R	—	It does not apply to Root Complex devices. 0b fixed.
17 to 15	L1 Exit Latency	110b	HwInit	RST_LOAD_B	This field indicates the L1 exit latency for the given PCI Express Link. The value reported indicates the length of time this Port requires to complete transition from ASPM L1 to L0. If ASPM L1 is not supported, the value is undefined. 000b: Less than 1µs 001b: 1 µs to less than 2 µs 010b: 2 µs to less than 4 µs 011b: 4 µs to less than 8 µs 100b: 8 µs to less than 16 µs 101b: 16 µs to less than 32 µs (Default) 110b: 32 µs to 64 µs 111b: More than 64 µs

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
14 to 12	L0s Exit Latency	100b	HwInit	RST_LOAD_B	This field indicates the L0s exit latency for the given PCI Express Link. The value reported indicates the length of time this Port requires to complete transition from L0s to L0. 000b: Less than 64 ns 001b: 64 ns to less than 128 ns 010b: 128 ns to less than 256 ns 011b: 256 ns to less than 512 ns 100b: 512 ns to less than 1 µs (Default) 101b: 1 µs to less than 2 µs 110b: 2 µs-4 µs 111b: More than 4 µs
11, 10	Active State Power Management (ASPM) Support	11b	HwInit	RST_LOAD_B	This field indicates the level of ASPM supported on the given PCI Express Link. 00b: No ASPM support 01b: L0s Entry Supported 10b: L1 Entry Supported 11b: L0s and L1 Entry Supported (Default)
9 to 4	Maximum Link Width	000001b	HwInit	RST_LOAD_B	This field indicates the maximum Link width Lanes, implemented by the component. This value is permitted to exceed the number of Lanes routed to the slot (Downstream Port), adapter connector (Upstream Port), or in the case of component-to-component connections, the actual wired connection width. 000000b: Reserved 000001b: x1 (Default) 000010b: x2 (Set prohibited) 000100b: x4 (Set prohibited) 001000b: x8 (Set prohibited) 001100b: x12 (Set prohibited) 010000b: x16 (Set prohibited) 100000b: x32 (Set prohibited)
3 to 0	Supported Link Speed	0010b	HwInit	RST_LOAD_B	This field indicates the maximum Link speed of the associated Port. 0001b: 2.5 GT/s Link speed supported 0010b: 5.0 GT/s and 2.5 GT/s Link speeds supported (Default) 0011b: 8.0 GT/s Link speed supported (Set prohibited) 0100b: 16.0 GT/s Link speed supported (Set prohibited) All other encodings are reserved.

34.3.2.21 Link Control/Status (Offset: H'6070)



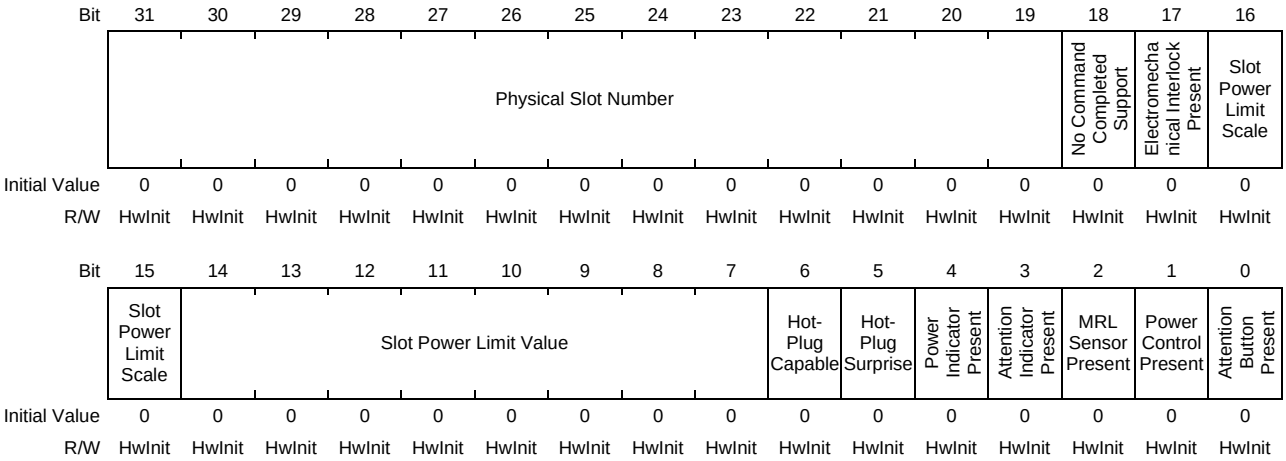
Bit	31	30	29	28	27	26	25 to 16	15 to 12	11	10	9	8	7	6	5	4	3	2	1	0
	Link Autonomous Bandwidth Status	Link Bandwidth Management Status	Data Link Layer Link Active	Slot Clock Configuration	Link Training	Undefined	Negotiated Link Width / Current Link Speed	—	Link Autonomous Bandwidth Interrupt Enable	Link Bandwidth Management Interrupt Enable	Hardware Autonomous Width Disable	Enable Clock Power Management	Extended Synch	Common Clock Configuration	Retrain Link	Link Disable	Read Completion Boundary (RCB)	—	Active State Power Management (ASPM) Control	
Initial Value	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R/W1C	R/W1C	R	HwInit	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31	Link Autonomous Bandwidth Status	0b	R/W1C	RST_CFG_B	This bit is output when Link Autonomous Bandwidth Interrupt Enable is asserted.
30	Link Bandwidth Management Status	0b	R/W1C	RST_CFG_B	This bit is output when Link Bandwidth Management Interrupt Enable is asserted.
29	Data Link Layer Link Active	0b	R	—	1b indicates that the Data Link Layer is in the Link Active state.
28	Slot Clock Configuration	1b	HwInit	RST_LOAD_B	It indicates that it uses a common reference clock with EP. 0b: Do not use Connector Reference Clock 1b: Use Connector Reference Clock (Default)
27	Link Training	0b	R	—	1b indicates that the Physical layer LTSSM is in the Configuration state or Recovery state. This bit is cleared when exiting the Configuration/Recovery state.
26	Undefined	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25 to 20	Negotiated Link Width	All 0	R	—	Indicates the link width established because of negotiation. 000001b: x1 000010b: x2 000100b: x4 001000b: x8 001100b: x12 010000b: x16 100000b: x32 The other encoding is Reserved.
19 to 16	Current Link Speed	All 0	R	—	This field indicates the negotiated Link speed of the given PCI Express Link. 0001b: 2.5 GT/s PCI Express Link 0010b: 5.0 GT/s PCI Express Link 0011b: 8.0 GT/s PCI Express Link 0100b: 16.0 GT/s PCI Express Link 0000b during reset period

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
15 to 12	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	Link Autonomous Bandwidth Interrupt Enable	0b	R/W	RST_CFG_B	Controls interrupt generation via the Link Autonomous Bandwidth Management Status bit (bit 31). 0b: Interrupt Disable (Default) 1b: Interrupt Enable
10	Link Bandwidth Management Interrupt Enable	0b	R/W	RST_CFG_B	Controls interrupt generation via the Link Bandwidth Status bit (bit 30). 0b: Interrupt Disable (Default) 1b: Interrupt Enable
9	Hardware Autonomous Width Disable	0b	R/W	RST_CFG_B	Set the Link Width Change feature to Disable. 0b: Link Width Change Enable (Default) 1b: Link Width Change Disable
8	Enable Clock Power Management	0b	R	RST_CFG_B	It does not apply to Root Complex devices. 0b fixed
7	Extended Synch	0b	R/W	RST_CFG_B	If set to 1 b, 4096 FTS Ordered-sets are transmitted when transitioning from L0s to L0. It also sends 1024 TS1 Ordered-sets at the beginning of the Recovery state during the transition from L1 to L0. The default value is 0b.
6	Common Clock Configuration	0b	R/W	RST_CFG_B	Sets whether the Common Reference Clock is used. 0b: Supplied by Non-Common Reference Clock (Default) 1b: Supplied by Common Reference Clock
5	Retrain Link	0b	R/W	RST_CFG_B	Setting it to 1b will transition LTSSM into the Recovery state and start Link Retraining. Readout is always 0b.
4	Link Disable	0b	R/W	RST_CFG_B	Setting it to 1b to transition LTSSM to the disabled state.
3	Read Completion Boundary (RCB)	1b	R	—	Read Completion Boundary (RCB) – Indicates the RCB value for the Root Port. 0: 64 bytes 1: 128 bytes (Default)
2	—	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1 to 0	Active State Power Management (ASPM) Control	00b	R/W	RST_CFG_B	Sets the permission level for Active State Power Management. 00b: Disabled (Default) 01b: L0s Entry Supported 10b: Reserved 11b: L0s and L1 Entry Supported

34.3.2.22 Slot Capabilities (Offset: H'6074)

PCI Express Capability Structure: Configuration Space Bit [24]: If Slot Implemented is 0 b, then this register is all '0' because it does not implement Slot.



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 19	Physical Slot Number	All 0	HwInit	RST_LOAD_B	Indicates Slot Number connected to Port.
18	No Command Completed Support	0b	HwInit	RST_LOAD_B	If the issued command is completed by the Hot-Plug Controller, it indicates that the Slot does not make a software notification.
17	Electromechanical Interlock Present	0b	HwInit	RST_LOAD_B	Indicates that Electromechanical Interlock is implemented.
16, 15	Slot Power Limit Scale	00b	HwInit	RST_LOAD_B	Set the Scale for the Slot Power Limit Value. This register is effective when the Slot Implemented bit (Express Capability Register (offset: H'060), bit 24) is set.00b = 1.0x 01b: 0.1x 10b: 0.01x 11b: 0.001x
14 to 7	Slot Power Limit Value	All 0	HwInit	RST_LOAD_B	In combination with the Slot Power Limit Scale Register value, sets the upper power limit in the Slot. This register is effective when the Slot Implemented bit (Express Capability Register (offset: H'060), bit 24) is set. H'F0: 250 W Slot Power Limit H'F1: 275 W Slot Power Limit H'F2: 300 W Slot Power Limit H'F3 to H'FF: Reserved
6	Hot-Plug Capable	0b	HwInit	RST_LOAD_B	Indicates that the Slot can support Hot-plug operation.
5	Hot-Plug Surprise	0b	HwInit	RST_LOAD_B	Indicates that an adapter in a Slot may be removed without notice.
4	Power Indicator Present	0b	HwInit	RST_LOAD_B	Indicates that the Power Indicator is electrically controlled by a Slot (enclosure).
3	Attention Indicator Present	0b	HwInit	RST_LOAD_B	Indicates that the Attention Indicator is electrically controlled by a Slot (enclosure).
2	MRL Sensor Present	0b	HwInit	RST_LOAD_B	Indicates that MRL Sensor is implemented.

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
1	Power Control Present	0b	HwInit	RST_LOAD_B	Indicates that a software programmable Power Controller is implemented.
0	Attention Button Present	0b	HwInit	RST_LOAD_B	Indicates that the Attention Button is electrically controlled by a Slot (enclosure).

34.3.2.23 Slot Control/Status (Offset: H'6078)

PCI Express Capability Structure: Configuration Space Bit [24]: If Slot Implemented is 0 b, it means that it has not implemented Slot, so all registers except bit [22]: Presence Detect State are all '0'.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	Data Link Layer State Changed	Electromechanical Interlock Status	Presence Detect State	MRL Sensor State	Command Completed	Presence Detect Changed	MRL Sensor Changed	Power Fault Detected	Attention Button Pressed
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W1C	R	R	R	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	Data Link Layer State Changed Enable	Electromechanical Interlock Control	Power Controller Control	Power Indicator Control		Attention Indicator Control		Hot-Plug Interrupt Enable	Command Completed Interrupt Enable	Presence Detect Changed Enable	MRL Sensor Changed Enable	Power Fault Detected Enable	Attention Button Pressed Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 25	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	Data Link Layer State Changed	0b	R/W1C	RST_CFG_B	This bit is Set when the value reported in the Data Link Layer Link Active bit of the Link Status Register is changed. Not used in this macro. 0b fixed.
23	Electromechanical Interlock Status	0b	R	—	If an Electromechanical Interlock is implemented, this bit indicates the status of the Electromechanical Interlock. Not used in this macro. 0b fixed.
22	Presence Detect State	1b	R	—	This bit indicates the presence of an adapter in the slot. When the In-Band PD Disable bit is Clear, this is reflected by the logical "OR" of the Physical Layer in-band presence detect mechanism and, if present, any out-of-band presence detect mechanism defined for the slot's corresponding form factor. The Slot Implemented bit in the PCI Express Capabilities register is 0 and then this bit is 1b fixed.
21	MRL Sensor State	0b	R	—	This bit reports the status of the MRL sensor if implemented. Not used in this macro. 0b fixed.
20	Command Completed	0b	R/W1C	RST_CFG_B	If Command Completed notification is supported (if the No Command Completed Support bit in the Slot Capabilities Register is 0b), this bit is Set when a hot-plug command has completed and the Hot-Plug Controller is ready to accept a subsequent command. Not used in this macro. 0b fixed.
19	Presence Detect Changed	0b	R/W1C	RST_CFG_B	This bit is Set when the value reported in the Presence Detect State bit is changed. Not used in this macro. 0b fixed.
18	MRL Sensor Changed	0b	R/W1C	RST_CFG_B	If an MRL sensor is implemented, this bit is Set when a MRL Sensor State change is detected. If an MRL sensor is not implemented, this bit must not be Set. Not used in this macro. 0b fixed.

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
17	Power Fault Detected	0b	R/W1C	RST_CFG_B	If a Power Controller that supports power fault detection is implemented, this bit is Set when the Power Controller detects a power fault at this slot. Not used in this macro. 0b fixed.
16	Attention Button Pressed	0b	R/W1C	RST_CFG_B	If an Attention Button is implemented, this bit is Set when the attention button is pressed. Not used in this macro. 0b fixed.
15 to 13	—	000b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	Data Link Layer State Changed Enable	0b	R/W	RST_CFG_B	If the Data Link Layer Link Active Reporting Capable is 1b, this bit enables software notification when Data Link Layer Link Active bit is changed. Not used in this macro. Please use 0b fixed.
11	Electromechanical Interlock Control	0b	R/W	RST_CFG_B	If an Electromechanical Interlock is implemented, a write of 1b to this bit causes the state of the interlock to toggle. A write of 0b to this bit has no effect. A read of this bit always returns a 0b. Not used in this macro. Please use 0b fixed.
10	Power Controller Control	0b	R/W	RST_CFG_B	If a Power Controller is implemented, this bit when written sets the power state of the slot per the defined encodings. Not used in this macro. Please use 0b fixed.
9, 8	Power Indicator Control	00b	R/W	RST_CFG_B	If a Power Indicator is implemented, writes to this field set the Power Indicator to the written state Not used in this macro. Please use 00b fixed.
7, 6	Attention Indicator Control	00b	R/W	RST_CFG_B	If an Attention Indicator is implemented, writes to this field set the Attention Indicator to the written state. Not used in this macro. Please use 0b fixed.
5	Hot-Plug Interrupt Enable	0b	R/W	RST_CFG_B	When Set, this bit enables generation of an interrupt on enabled hot-plug events. Not used in this macro. Please use 0b fixed.
4	Command Completed Interrupt Enable	0b	R/W	RST_CFG_B	If Command Completed notification is supported when Set, this bit enables software notification when a hot-plug command is completed by the Hot-Plug Controller. Not used in this macro. Please use 0b fixed.
3	Presence Detect Changed Enable	0b	R/W	RST_CFG_B	When Set, this bit enables software notification on a presence detect changed event. Not used in this macro. Please use 0b fixed.
2	MRL Sensor Changed Enable	0b	R/W	RST_CFG_B	When Set, this bit enables software notification on a MRL sensor changed event. Not used in this macro. Please use 0b fixed.
1	Power Fault Detected Enable	0b	R/W	RST_CFG_B	When Set, this bit enables software notification on a power fault event. Not used in this macro. Please use 0b fixed.
0	Attention Button Pressed Enable	0b	R/W	RST_CFG_B	When Set to 1b, this bit enables software notification on an attention button pressed event. Not used in this macro. Please use 0b fixed.

34.3.2.24 Root Control/Capabilities (Offset: H'607C)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CRS Software Visibility
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	CRS Software Visibility Enable	PME Interrupt Enable	System Error on Fatal Error Enable	System Error on Non-Fatal Error Enable	System Error on Correctable Error Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 17	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CRS Software Visibility	0b	R	—	Indicates whether the Root Port can notify the software of the CRS status. Not used in this macro. Please use 0b fixed.
15 to 5	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	CRS Software Visibility Enable	0b	R/W	RST_CFG_B	When implemented, allows the Root Port to inform the software of the CRS status. Not used in this macro. Please use 0b fixed.
3	PME Interrupt Enable	0b	R/W	RST_CFG_B	When a PME Message is received, it can be combined with a set of PME status bits to generate a PME interrupt. If the PME status is already up and this bit is enabled later, an interrupt will also occur.
2	System Error on Fatal Error Enable	0b	R/W	RST_CFG_B	This bit allows a system error to be generated when ERR_FATAL occurs in any of the hierarchies below the root port.
1	System Error on Non-Fatal Error Enable	0b	R/W	RST_CFG_B	This bit allows a system error to be generated when NON_ERR_FATAL occurs in any of the hierarchies below the root port.
0	System Error on Correctable Error Enable	0b	R/W	RST_CFG_B	This bit allows a system error to be generated when ERR_ERR_COR occurs in any of the hierarchies below the root port.

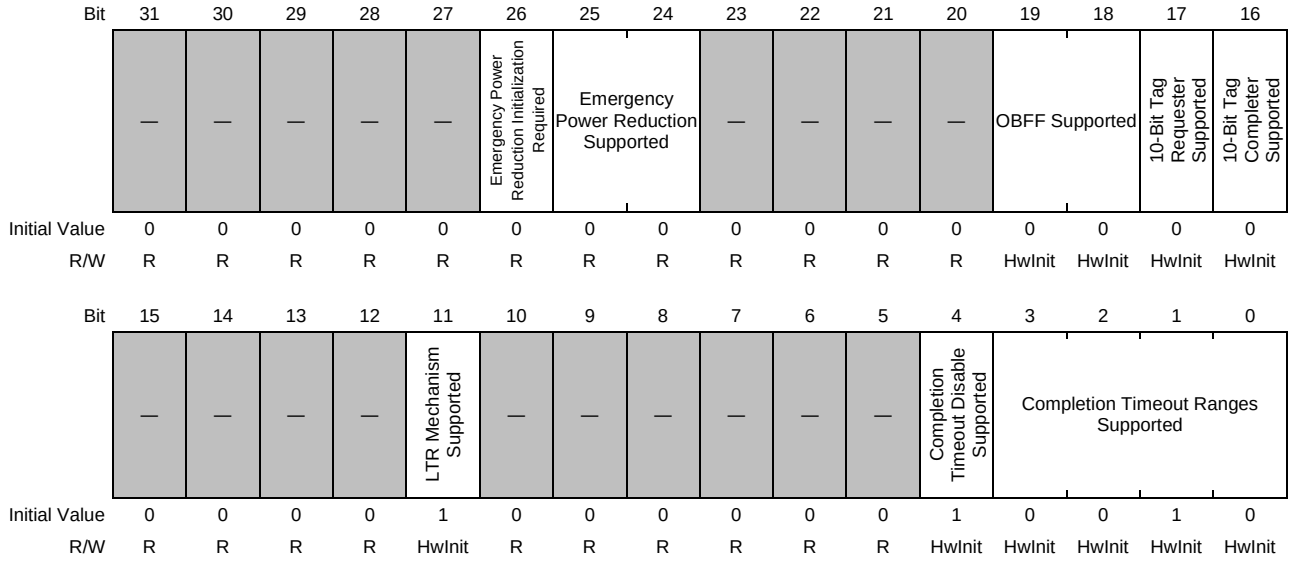
34.3.2.25 Root Status (Offset: H'6080)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PME Pending	PME Status
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W1C

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PME Requester ID															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 18	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	PME Pending	0b	R	RST_CFG_B	This bit indicates a status where the PME Status bit is 1 and further PME is pending. When the software clears the PME status bit, the pending PME is resent in hardware by resetting the PME status bit and properly updating the Requester ID field and PME status. The PME pending bit is cleared in hardware when there are no more PME's pending.
16	PME Status	0b	R/W1C	RST_CFG_B	This bit indicates that a PME was received by the requester indicated by the PME Requester ID field. Subsequent PMEs are held until this bit is cleared by the software.
15 to 0	PME Requester ID	All 0	R	RST_CFG_B	This field shows the PCI Requester ID of the most recent PME requester. This field is valid only when the PME status bit is 1.

34.3.2.26 Device Capabilities 2 (Offset: H'6084)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 27	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	Emergency Power Reduction Initialization Required	0b	R	RST_LOAD_B	Emergency Power Reduction Initialization Required Not supported 0b fixed
25, 24	Emergency Power Reduction Supported	00b	R	RST_LOAD_B	Emergency bower Reduction Supported Not supported 00b fixed
23 to 20	—	H'0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19, 18	OBFF Supported	00b	HwInit	RST_LOAD_B	This field indicates if OBFF is supported and, if so, what signaling mechanism is used. 00b (Not Supported). 00b: OBFF Not Supported (Default) 01b: OBFF supported using Message signaling only 10b: OBFF supported using WAKE# signaling only 11b: OBFF supported using WAKE# and Message signaling
17	10-Bit Tag Requester Supported	0b	HwInit	RST_LOAD_B	10-Bit Tag (Requester) supported. 0b fixed. 0b: Not supported (Default) 1b: Supported
16	10-Bit Tag Completer Supported	0b	HwInit	RST_LOAD_B	10-Bit Tag (Completer) supported. 0b fixed. 0b: Not supported (Default) 1b: Supported
15 to 12	—	H'0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	LTR Mechanism Supported	1b	HwInit	RST_LOAD_B	Set with or without to support Latency Tolerance Reporting (LTR). 0b: Not Supported 1b: Supported (Default)
10 to 5	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
4	Completion Timeout Disable Supported	1b	HwInit	RST_LOAD_B	Set with or without to support Completion Timeout Disable. 0b: Not Supported 1b: Supported (Default)
3 to 0	Completion Timeout Ranges Supported	0010b	HwInit	RST_LOAD_B	Set Completion Timeout Range. Range A: 50 µs to 10 ms Range B: 10 ms to 250 ms Range C: 250 ms to 4 s Range D: 4 s to 64 s The four patterns described above are determined, and settings can be made in the following combinations. 0000b: Program settings not supported 0001b: Range A 0010b: Range B 0011b: Ranges A and B 0110b: Ranges B and C 0111b: Ranges A, B, and C 1110b: Ranges B, C and D 1111b: Ranges A, B, C, and D The other encoding is Reserved. *Note*: The initial value in the UM description is “0010b”, but please change the initial value according to the installed system/device performance or specify the initial value as a product requirement.

34.3.2.27 Device Control 2/Status 2 (Offset: H'6088)

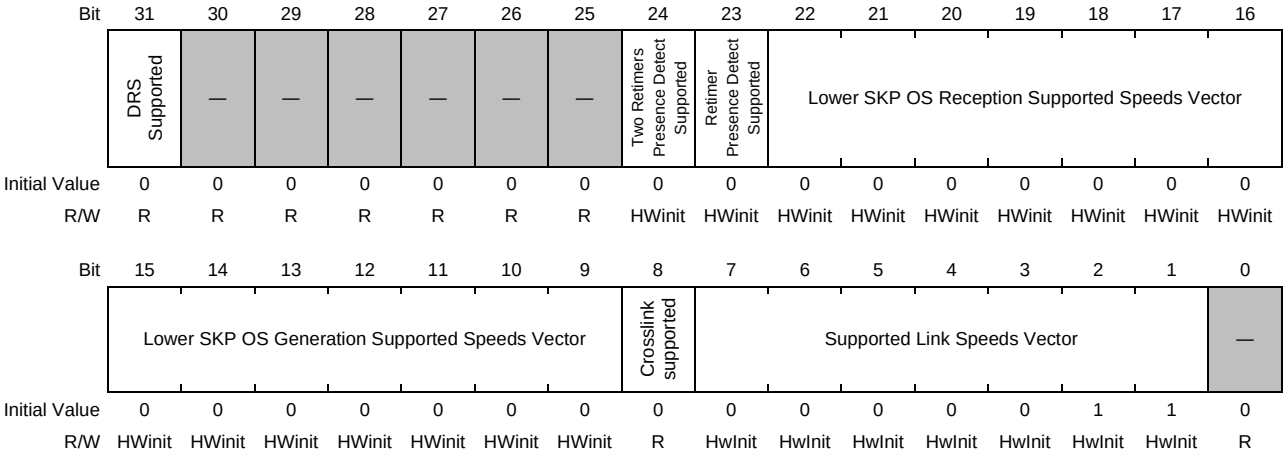
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	OBFF Enable		10-Bit Tag Requester Enable	Emergency Power Reduction Request	LTR Mechanism Enable	—	—	—	—	—	Completion Timeout Disable	Completion Timeout Value			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 15	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14, 13	OBFF Enable	00b	R/W	RST_CFG_B	This field enables the OBFF mechanism and selects the signaling method. CHANGE prohibited 00b (Disabled) 01b Enabled using Message signaling [Variation A] 10b Enabled using Message signaling [Variation B] 11b Enabled using WAKE# signaling *Note*: OBFF Supported = 00b equals Reserved.
12	10-Bit Tag Requester Enable	0b	R/W	RST_CFG_B	10-Bit Tag (Requester) enabled. 0b fixed. 0b: Disabled (Default) 1b: Valid
11	Emergency Power Reduction Request	0b	R	—	Emergency Power Reduction Request. 0b fixed due to unsupported
10	LTR Mechanism Enable	0b	R/W	RST_CFG_B	If this bit is set, the LTR function is enabled.
9 to 5	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	Completion Timeout Disable	0b	R/W	RST_CFG_B	If this bit is set, the Completion Timeout Disable function is enabled.

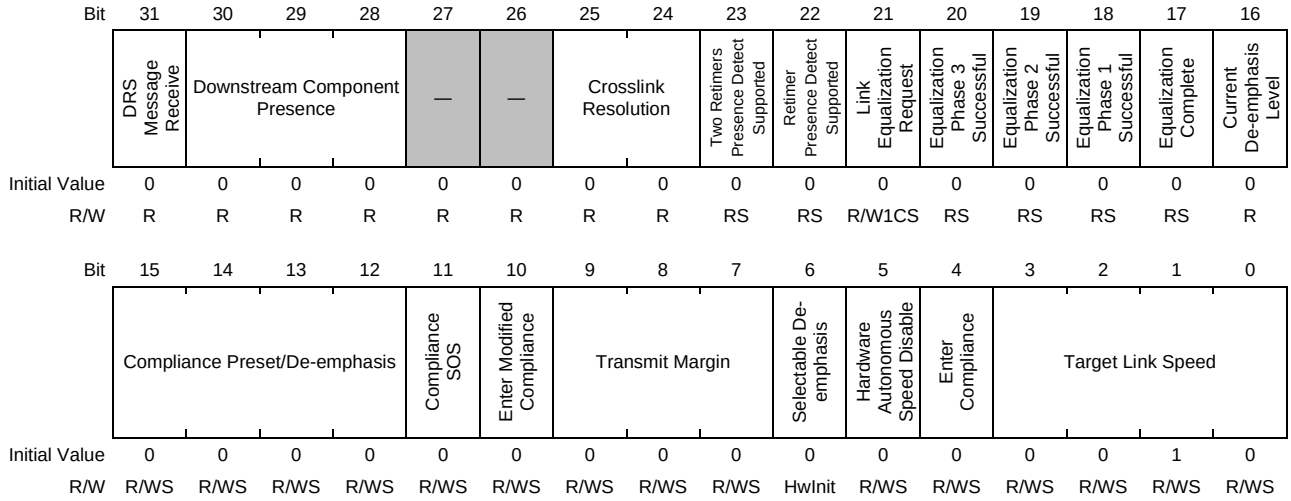
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
3 to 0	Completion Timeout Value	H'0	R/W	RST_CFG_B	Set Completion Timeout Range. 0000b: 10 ms to 50 ms (Default) 0001b: 50 µs to 100 µs 0010b: 1 ms to 10 ms 0101b: 16 ms to 55 ms 0110b: 65 ms to 210 ms 1001b: 260 ms to 900 ms 1010b: 1 s to 3.5 s 1101b: 4 s to 13 s 1110b: 17 s to 64 s Others: reserved (set prohibited) *Note*: The default, 0000b, sets Base Spec to a time longer than the default 50usec lower limit. This considers the 10 msec lower bound recommended for Base Spec.

34.3.2.28 Link Capabilities 2 (Offset: H'608C)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31	DRS Supported	0b	R	—	When Set, indicates support for the optional Device Readiness Status (DRS) capability. This function is not supported by this product. 0b fixed
30 to 25	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	Two Retimers Presence Detect Supported	0b	HWinit	RST_LOAD_B	When set to 1b, this bit indicates that the associated Port supports detection and reporting of two Retimers presence. This function is not supported by this product. 0b fixed
23	Retimer Presence Detect Supported	0b	HWInit	RST_LOAD_B	When set to 1b, this bit indicates that the associated Port supports detection and reporting of Retimer presence. This function is not supported by this product. 0b fixed
22 to 16	Lower SKP OS Reception Supported Speeds Vector	All 0	HWinit	RST_LOAD_B	If this field is non-zero, it indicates that the Port, when operating at the indicated speeds supports SRIS and also supports receiving SKP OS at the rate defined for SRNS while running in SRIS. This function is not supported by this product. 7'h00 fixed
15 to 9	Lower SKP OS Generation Supported Speeds Vector	All 0	HWinit	RST_LOAD_B	If this field is non-zero, it indicates that the Port, when operating at the indicated speeds supports SRIS and also supports software control of the SKP Ordered Set transmission scheduling rate. This function is not supported by this product. 7'h00 fixed
8	Crosslink supported	0b	R	—	When set to 1b, this bit indicates that the associated Port supports crosslinks. This function is not supported by this product. 0b fixed
7 to 1	Supported Link Speeds Vector	0000011b	HwInit	RST_LOAD_B	Indicate Support Link Speed. This field indicates the supported Link speed(s) of the associated Port. For each bit, a value of 1b indicates that the corresponding Link speed is supported; otherwise, the Link speed is not supported. Bit 0: 2.5 GT/s Bit 1: 5.0 GT/s Bit 2: 8.0 GT/s (set prohibited) Bit 3: 16.0 GT/s (set prohibited) Bits 6 to 4: Reserved
0	—	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.

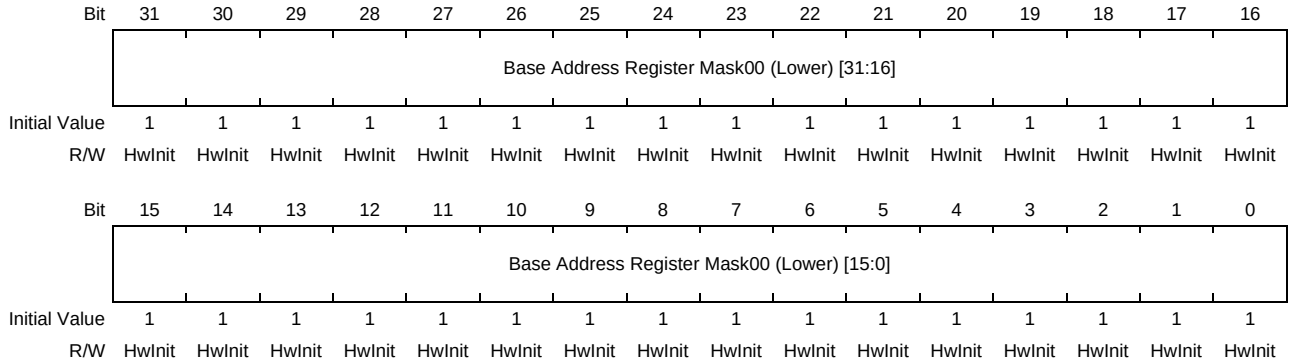
34.3.2.29 Link Control 2/Status 2 (Offset: H'6090)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31	DRS Message Received	0b	R	—	This bit must be Set whenever the Port receives a DRS Message. This function is not supported by this product. 0b fixed
30 to 28	Downstream Component Presence	000b	R	—	This field indicates the presence and DRS status for the Downstream Component, if any, connected to the Link. This function is not supported by this product. 000b fixed
27, 26	—	00b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25, 24	Crosslink Resolution	00b	R	—	This field indicates the state of the Crosslink negotiation. This function is not supported by this product. 00b fixed
23	Two Retimers Presence Detect Supported	0b	RS	—	When set to 1b, this bit indicates that two Retimers were present during the most recent Link negotiation. This function is not supported by this product. 0b fixed
22	Retimer Presence Detect Supported	0b	RS	—	When set to 1b, this bit indicates that a Retimer was present during the most recent Link negotiation. This function is not supported by this product. 0b fixed
21	Link Equalization Request	0b	R/W1CS	RST_RSM_B	This bit is Set by hardware to request the Link equalization process to be performed on the Link. This function is not supported by this product. 0b fixed
20	Equalization Phase 3 Successful	0b	RS	—	When set to 1b, this bit indicates that Phase 3 of the Transmitter Equalization procedure has successfully completed. This function is not supported by this product. 0b fixed
19	Equalization Phase 2 Successful	0b	RS	—	When set to 1b, this bit indicates that Phase 2 of the Transmitter Equalization procedure has successfully completed. This function is not supported by this product. 0b fixed
18	Equalization Phase 1 Successful	0b	RS	—	When set to 1b, this bit indicates that Phase 1 of the Transmitter Equalization procedure has successfully completed. This function is not supported by this product. 0b fixed
17	Equalization Complete	0b	RS	—	When set to 1b, this bit indicates that the Transmitter Equalization procedure has completed. This function is not supported by this product. 0b fixed

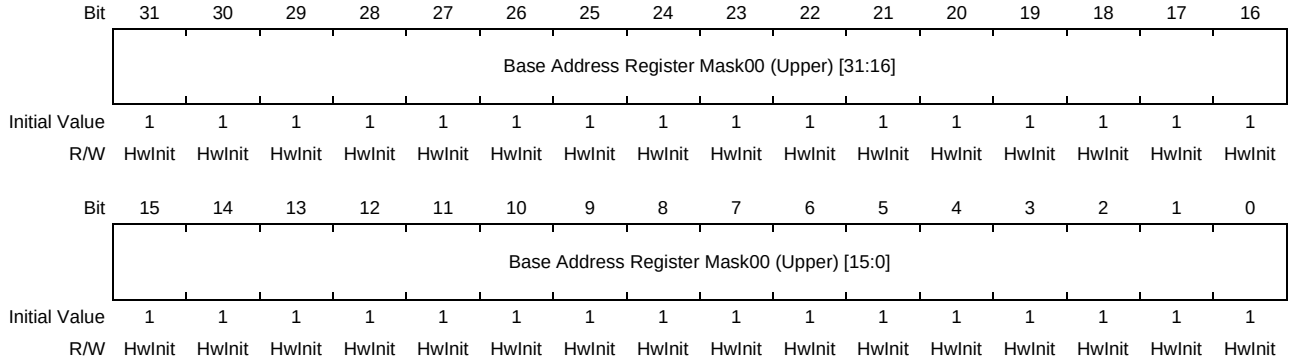
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
16	Current De-emphasis Level	0b	R	—	Status register showing De-emphasis Level during Gen2 operation. 0: −6 dB (Default) 1: −3.5 dB *Note*: The initial value of this bit is 0b at reset and after Gen2-Linkup. However, Gen1-Linkup indicates 1b, but this bit has no meaning in Gen1, so ignore it.
15 to 12	Compliance Preset/De-emphasis	0000b	R/WS	RST_RSM_B	Polling. Set the De-emphasis Level in the Compliance State. 0000b: −6 dB (Default) 0001b: −3.5 dB
11	Compliance SOS	0b	R/WS	RST_RSM_B	If this bit set to 1b, the SKP Ordered-set is inserted periodically during Compliance pattern transmission.
10	Enter Modified Compliance	0b	R/WS	RST_RSM_B	Set bit for Modified Compliance pattern transmission. 0: Compliance Pattern (Default) 1: Modified Compliance Pattern
9 to 7	Transmit Margin	000b	R/WS	RST_RSM_B	Adjust the voltage level of the Transmitter. 000b: Normal operating range 001b to 111b: See *Base Spec Section 8.3.4*.
6	Selectable De-emphasis	0b	HwInit	—	This register is for De-emphasis setting during RC Gen 2 operation. 0: −6 dB 1: −3.5 dB
5	Hardware Autonomous Speed Disable	0b	R/WS	RST_RSM_B	Controls the Link Speed Change function. 0: Support of Link Speed Change (Enable) 1: not Support of Link Speed Change (Disable)
4	Enter Compliance	0b	R/WS	RST_RSM_B	By setting it to 1b, Allows a transition to Compliance mode. At this time, the Link Speed will be the value set in the Target Link Speed field.
3 to 0	Target Link Speed	0010b	R/WS	RST_RSM_B	Set Link Speed value to notify opposing devices during training. 0001b: 2.5 GT/s Target Link Speed 0010b: 5.0 GT/s Target Link Speed (Default) 0011b: 8.0 GT/s Target Link Speed (set prohibited) 0100b: 16.0 GT/s Target Link Speed (set prohibited) All other encodings are reserved.

34.3.2.30 Base Address Register Mask00(Lower) (Offset: H'60A0)



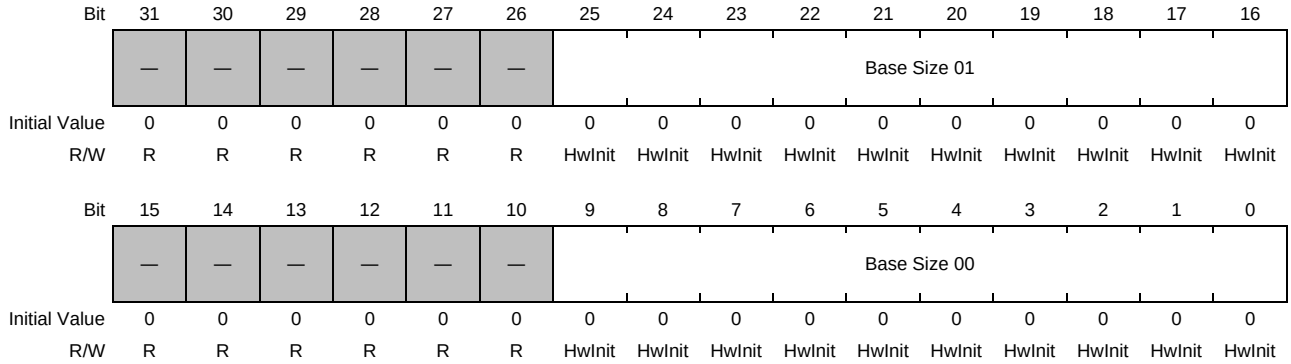
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 0	Base Address Register Mask00 (Lower)	All 1	HwInit	RST_LOAD_B	This register is a mask register for Base Address Register 0 (BAR 0).

34.3.2.31 Base Address Register Mask00(Upper) (Offset: H'60A4)



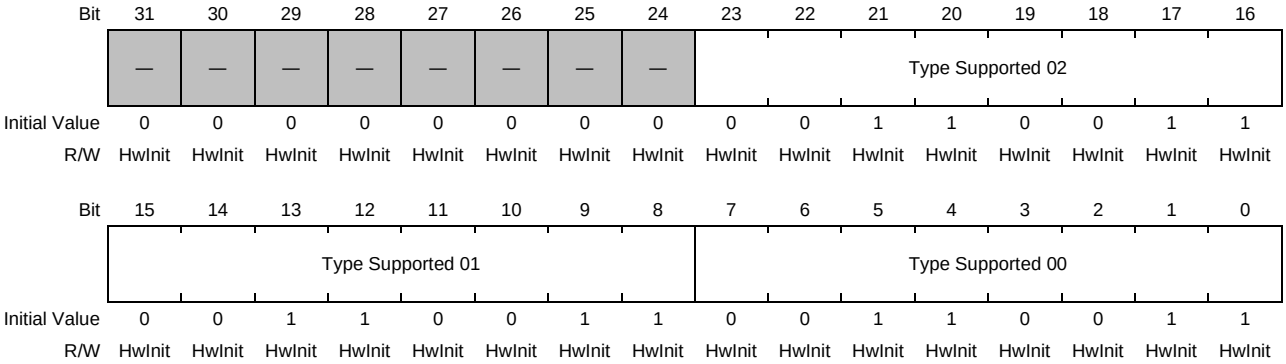
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 0	Base Address Register Mask00 (Upper)	All 1	HwInit	RST_LOAD_B	This register is a mask register for Base Address Register 1 (BAR1).

34.3.2.32 Base Size 00/01 (Offset: H’60C8)



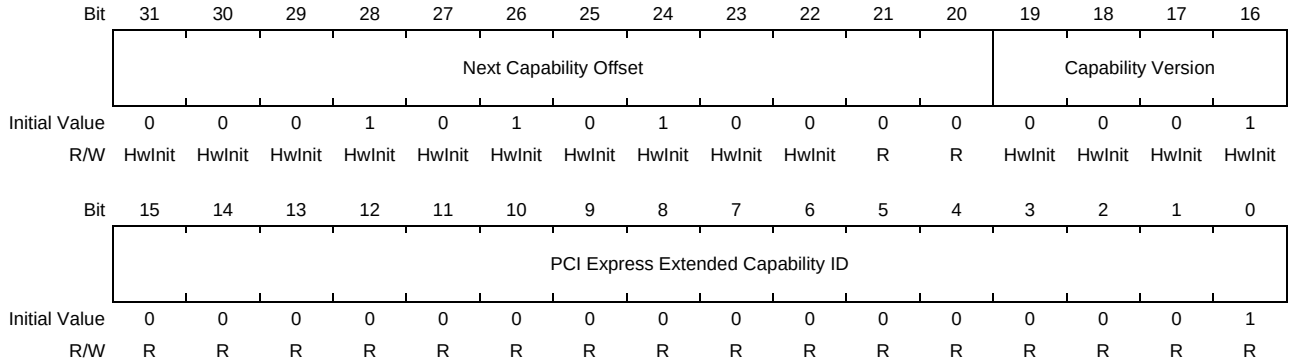
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 26	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25 to 16	Base Size 01	All 0	HwInit	RST_LOAD_B	(H’000 fixed) Not used.
15 to 10	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9 to 0	Base Size 00	All 0	HwInit	RST_LOAD_B	Sets the size of TLP (DW Size) accepted by Address Space 00 set by Base Address Register and Base Address Mask Register. Detects a Completer Abort Error (CA) when a TLP with a packet length greater than the size set here is received (even if it is smaller than the Max Payload Size). Note that the default value is H’000, which disables this feature.

34.3.2.33 Type Supported 00/01/02 (Offset: H’60D8)



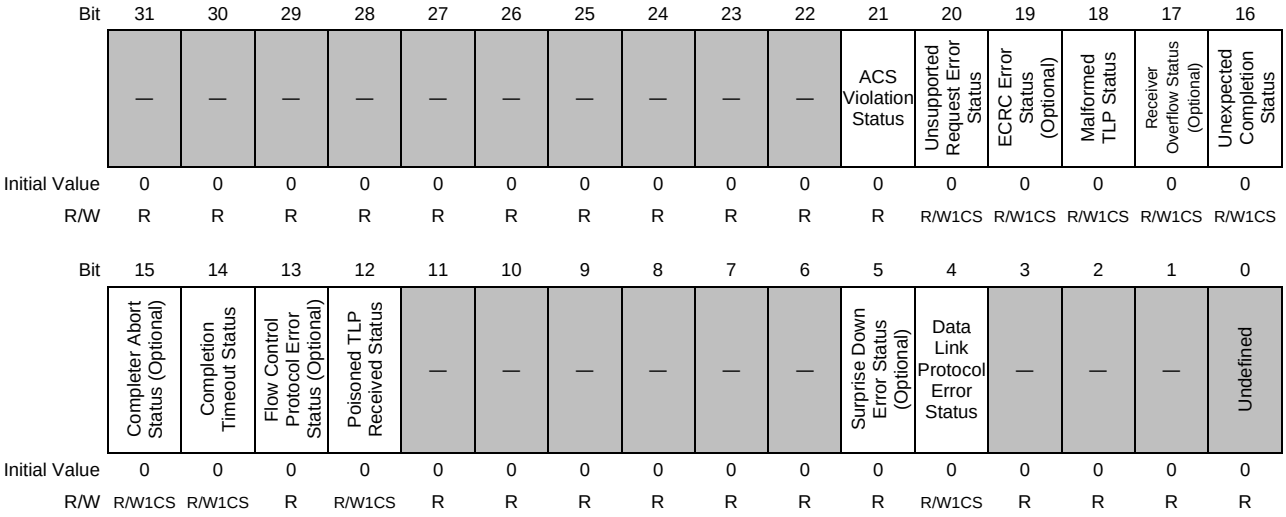
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 24	—	H’00	HwInit	RST_LOAD_B	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23 to 16	Type Supported 02	H’33	HwInit	RST_LOAD_B	*Note*: Do not use.
15 to 8	Type Supported 01	H’33	HwInit	RST_LOAD_B	*Note*: Do not use.
7 to 0	Type Supported 00	H’33	HwInit	RST_LOAD_B	Set the Transaction Type that can be supported by Space00 (CFG_SPACE00_BASE). See the below for the meaning of each bit. Bit 0: Memory read 32 bits Bit 1: Memory read 64 bits Bit 2: Memory read lock 32 bits Bit 3: Memory read lock 64 bits Bit 4: Memory write 32 bits Bit 5: Memory write 64 bits Bit 6: IO Read Bit 7: IO Write

34.3.2.34 Advanced Error Reporting Capability (Offset: H’6100)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 22	Next Capability Offset	H’150	HwInit	RST_LOAD_B	Indicates the starting address of the Device Serial Number Capability.
21, 20			R	—	
19 to 16	Capability Version	0001b	HwInit	RST_LOAD_B	Indicates the Capability Structure version. Default: 0001b
15 to 0	PCI Express Extended Capability ID	H’0001	R	—	Indicates the Advanced Error Reporting Capability. Default: H’0001

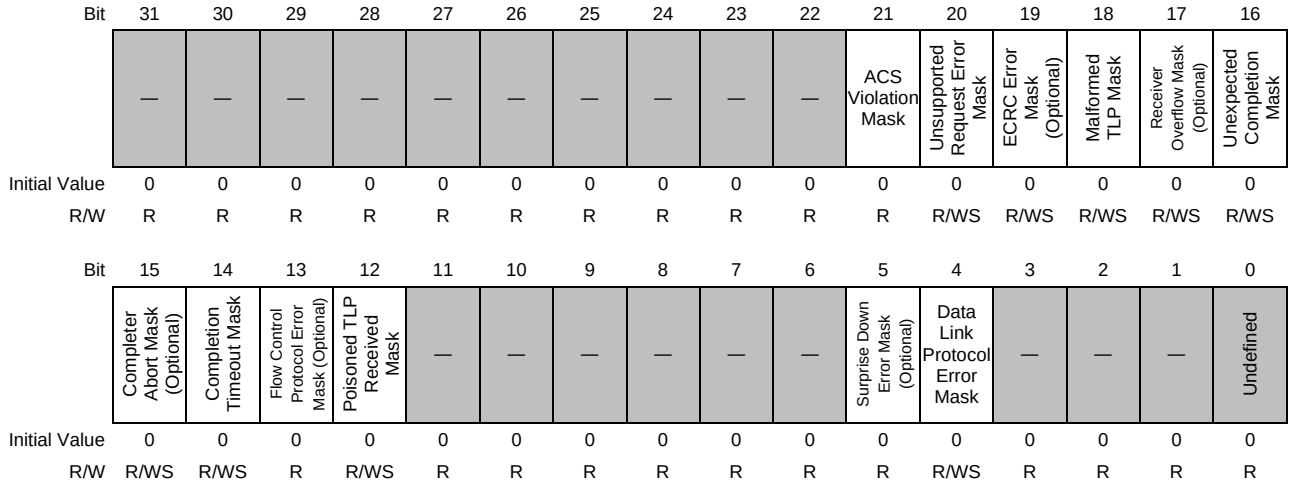
34.3.2.35 Uncorrectable Error Status Register (Offset: H'6104)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 22	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	ACS Violation Status	0b	R	—	ACS is not supported 0b fixed
20	Unsupported Request Error Status	0b	R/W1CS	—	Indicate an unsupported TLP was received. 0: No error detected 1: Error detected
19	ECRC Error Status (Optional)	0b	R/W1CS	—	ECRC Error was received. 0: No error detected 1: Error detected
18	Malformed TLP Status	0b	R/W1CS	—	Indicate Malformed TLP received. 0: No error detected 1: Error detected
17	Receiver Overflow Status (Optional)	0b	R/W1CS	—	Indicates A receipt of a TLP larger in size than the free credits in the receive buffer. 0: No error detected 1: Error detected
16	Unexpected Completion Status	0b	R/W1CS	—	Indicates that Completion was received, but there is no record of a corresponding Non-Posted Request issue (Transaction Descriptor did not match). 0: No error detected 1: Error detected
15	Completer Abort Status (Optional)	0b	R/W1CS	—	Indicates that Completion has been returned with a Completion Status of Completer Abort (CA) after receiving a Non-Posted Request. 0: No error detected 1: Error detected
14	Completion Timeout Status	0b	R/W1CS	—	Indicates that after sending a Non-Posted Request, the corresponding Completion was not received within the specified time. 0: No error detected 1: Error detected

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
13	Flow Control Protocol Error Status (Optional)	0b	R	—	Not Implemented (0b fixed)
12	Poisoned TLP Received Status	0b	R/W1CS	—	Indicates that TLP of Poisoned TLP (with payload and EP field of header is 1b) has been received. 0: No error detected 1: Error detected
11 to 6	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	Surprise Down Error Status (Optional)	0b	R	—	Not Implemented (0b fixed)
4	Data Link Protocol Error Status	0b	R/W1CS	—	Indicates that Sequence Number error has been detected in the Data Link Layer. 0: No error detected 1: Error detected
3 to 1	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	Undefined	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.

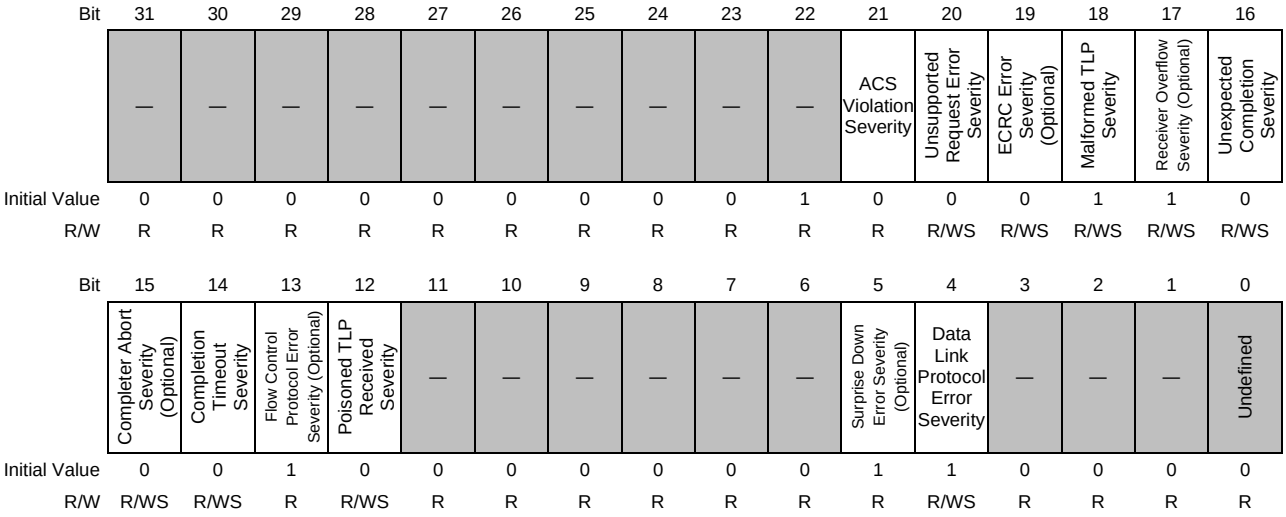
34.3.2.36 Uncorrectable Error Mask Register (Offset: H'6108)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 22	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	ACS Violation Mask	0b	R	—	ACS is not supported 0b fixed
20	Unsupported Request Error Mask	0b	R/WS	RST_RSM_B	Mask error notifications when the Unsupported Request Error is detected. 0b: No mask 1b: Send Error Message, recording headers to the Header Log register, Mask the First Error Pointer Update.
19	ECRC Error Mask (Optional)	0b	R/WS	RST_RSM_B	Mask error notifications when the ECRC Error is detected. 0b: No mask 1b: Mask
18	Malformed TLP Mask	0b	R/WS	RST_RSM_B	Mask error notifications when the Malformed TLP Error is detected. 0b: No mask 1b: Send Error Message, recording headers to the Header Log register, Mask the First Error Pointer Update.
17	Receiver Overflow Mask (Optional)	0b	R/WS	RST_RSM_B	Mask error notifications when the Receiver Overflow Error is detected. 0b: No mask 1b: Send Error Message, Mask the First Error Pointer Update.
16	Unexpected Completion Mask	0b	R/WS	RST_RSM_B	Mask error notifications when the Unexpected Completion Error is detected. 0b: No mask 1b: Send Error Message, recording headers to the Header Log register, Mask the First Error Pointer Update.
15	Completer Abort Mask (Optional)	0b	R/WS	RST_RSM_B	Mask error notifications when the Completer Abort Error is detected. 0b: No mask 1b: Send Error Message, recording headers to the Header Log register, Mask the First Error Pointer Update.
14	Completion Timeout Mask	0b	R/WS	RST_RSM_B	Mask error notifications when the Completion Timeout Error is detected. 0b: No mask 1b: Send Error Message, Mask the First Error Pointer Update.

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
13	Flow Control Protocol Error Mask (Optional)	0b	R	—	Not Implemented (0b fixed)
12	Poisoned TLP Received Mask	0b	R/WS	RST_RSM_B	Mask error notifications when the Poisoned TLP Error is detected. 0b: No mask 1b: Send Error Message, recording headers to the Header Log register, Mask the First Error Pointer Update.
11 to 6	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	Surprise Down Error Mask (Optional)	0b	R	—	Not Implemented (0b fixed)
4	Data Link Protocol Error Mask	0b	R/WS	RST_RSM_B	Mask error notifications when the Data Link Protocol Error is detected. 0b: No mask 1b: Send Error Message, Mask the First Error Pointer Update.
3 to 1	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	Undefined	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.

34.3.2.37 Uncorrectable Error Severity Register (Offset: H'610C)



Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	ACS Violation Severity	Unsupported Request Error Severity	ECRC Error Severity (Optional)	Malformed TLP Severity	Receiver Overflow Severity (Optional)	Unexpected Completion Severity
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/WS	R/WS	R/WS	R/WS	R/WS

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Completer Abort Severity (Optional)	Completion Timeout Severity	Flow Control Protocol Error Severity (Optional)	Poisoned TLP Received Severity	—	—	—	—	—	—	Surprise Down Error Severity (Optional)	Data Link Protocol Error Severity	—	—	—	Undefined
Initial Value	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0
R/W	R/WS	R/WS	R	R/WS	R	R	R	R	R	R	R	R/WS	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 23	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	—	1b	R	RST_LOAD_B	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
21	ACS Violation Severity	0b	R	—	ACS is not supported. 0b fixed
20	Unsupported Request Error Severity	0b	R/WS	RST_RSM_B	Set the Error Severity when the Unsupported Request Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
19	ECRC Error Severity (Optional)	0b	R/WS	RST_RSM_B	Set the Error Severity when the ECRC Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
18	Malformed TLP Severity	1b	R/WS	RST_RSM_B	Set the Error Severity when the Malformed TLP Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
17	Receiver Overflow Severity (Optional)	1b	R/WS	RST_RSM_B	Set the Error Severity when the Receiver Overflow Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
16	Unexpected Completion Severity	0b	R/WS	RST_RSM_B	Set the Error Severity when the Unexpected Completion Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
15	Completer Abort Severity (Optional)	0b	R/WS	RST_RSM_B	Set the Error Severity when the Completer Abort Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
14	Completion Timeout Severity	0b	R/WS	RST_RSM_B	Set the Error Severity when the Completion Timeout Error is detected. 0b: Non-Fatal Error 1b: Fatal Error

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
13	Flow Control Protocol Error Severity (Optional)	1b	R	—	Not Implemented (1b fixed)
12	Poisoned TLP Received Severity	0b	R/WS	RST_RSM_B	Set the Error Severity when the Poisoned TLP Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
11 to 6	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	Surprise Down Error Severity (Optional)	1b	R	—	Not Implemented (1b fixed)
4	Data Link Protocol Error Severity	1b	R/WS	RST_RSM_B	Set the Error Severity when the Data Link Protocol Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
3 to 1	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	Undefined	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.

34.3.2.38 Correctable Error Status Register (Offset: H'6110)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	Advisory Non-Fatal Error Status	Replay Timer Timeout Status	—	—	—	REPLAY_NUM Rollover Status	Bad DLLP Status	Bad TLP Status	—	—	—	—	—	Receiver Error Status
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W1CS	R/W1CS	R	R	R	R/W1CS	R/W1CS	R/W1CS	R	R	R	R	R	R/W1CS

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 14	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	Advisory Non-Fatal Error Status	0b	R/W1CS	RST_RSM_B	Indicates that an Advisory Non-Fatal Error has been detected. 0b: No error detected 1b: Error detected
12	Replay Timer Timeout Status	0b	R/W1CS	RST_RSM_B	Indicates that a timeout error has occurred when an Ack or Nak DLLP was not received within the specified time after the transmission of a TLP. 0b: No error detected 1b: Error detected
11 to 9	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	REPLAY_NUM Rollover Status	0b	R/W1CS	RST_RSM_B	Replay occurs 4 times in a row, indicating that REPLAY_NUM has rolled over from 11b to 00b. 0b: No error detected 1b: Error detected
7	Bad DLLP Status	0b	R/W1CS	RST_RSM_B	Indicates that the CRC error in the DLLP has been detected. 0b: No error detected 1b: Error detected
6	Bad TLP Status	0b	R/W1CS	RST_RSM_B	Indicates that the CRC or Sequence Number error in the TLP has been detected. 0b: No error detected 1b: Error detected
5 to 1	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	Receiver Error Status (optional)	0b	R/W1CS	RST_RSM_B	0b: No error detected 1b: Error detected

34.3.2.39 Correctable Error Mask Register (Offset: H'6114)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	Advisory Non-Fatal Error Mask	Replay Timer Timeout Mask	—	—	—	REPLAY_NUM Rollover Mask	Bad DLLP Mask	Bad TLP Mask	—	—	—	—	—	Receiver Error Mask
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/WS	R/WS	R	R	R	R/WS	R/WS	R/WS	R	R	R	R	R	R/WS

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 14	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	Advisory Non-Fatal Error Mask	1b	R/WS	RST_RSM_B	Mask error notifications when the Advisory Non-Fatal Error is detected. 0b: No mask 1b: Mask the Advisory Non-Fatal Error handling. (Mask First Error Pointer and Header Logging updates and sending Error Messages)
12	Replay Timer Timeout Mask	0b	R/WS	RST_RSM_B	Mask error notifications when the Replay Timer Timeout Error is detected. 0b: No mask 1b: Mask the Error Message transmission
11 to 9	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	REPLAY_NUM Rollover Mask	0b	R/WS	RST_RSM_B	Mask error notifications when the REPLAY_NUM Roll Over Error is detected. 0b: No mask 1b: Mask the Error Message transmission
7	Bad DLLP Mask	0b	R/WS	RST_RSM_B	Mask error notifications when the Bad DLLP Error is detected. 0b: No mask 1b: Mask the Error Message transmission
6	Bad TLP Mask	0b	R/WS	RST_RSM_B	Mask error notifications when the Bad TLP Error is detected. 0b: No mask 1b: Mask the Error Message transmission
5 to 1	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	Receiver Error Mask (optional)	0b	R/WS	RST_RSM_B	0b: No mask 1b: Mask the Error Message transmission

34.3.2.40 Advanced Error Capabilities and Control Register (Offset: H'6118)

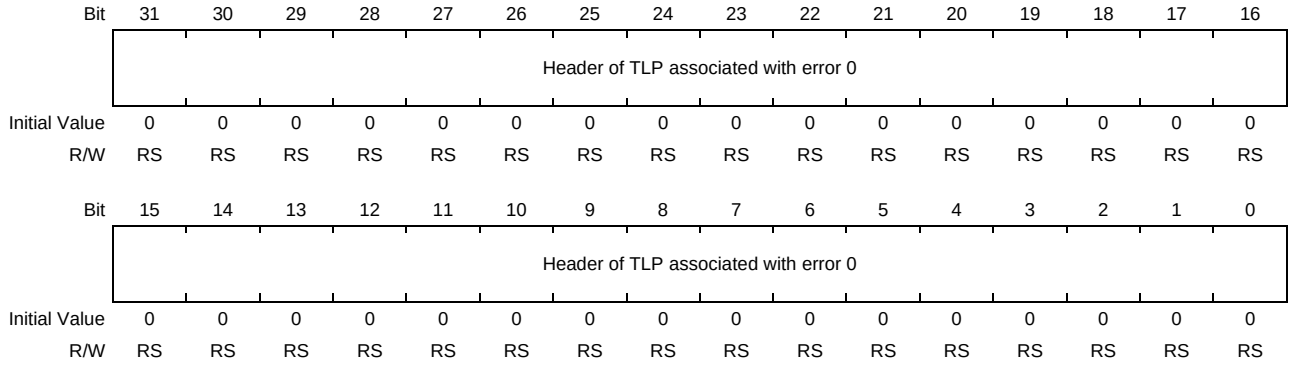
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ECRC Check Enable	ECRC Check Capable	ECRC Generation Enable	ECRC Generation Capable	First Error Pointer				
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/WS	R	R/WS	R	RS	RS	RS	RS	RS

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 9	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	ECRC Check Enable	0b	R/WS	RST_RSM_B	Set for Enable ECRC Check. 0b: Disable (Default) 1b: Enable
7	ECRC Check Capable	1b	R	—	Whether the ECRC Check function is installed. 0b: No mounting 1b: Equipped (Default)
6	ECRC Generation Enable	0b	R/WS	RST_RSM_B	Set for Enable ECRC Check. 0b: Disable (Default) 1b: Enable
5	ECRC Generation Capable	1b	R	—	Whether ECRC Generation function is installed. 0b: No mounting 1b: Equipped (Default)
4 to 0	First Error Pointer	All 0	RS	—	Indicates the field value in the Uncorrectable Error Status register for the first Uncorrectable Error encountered.

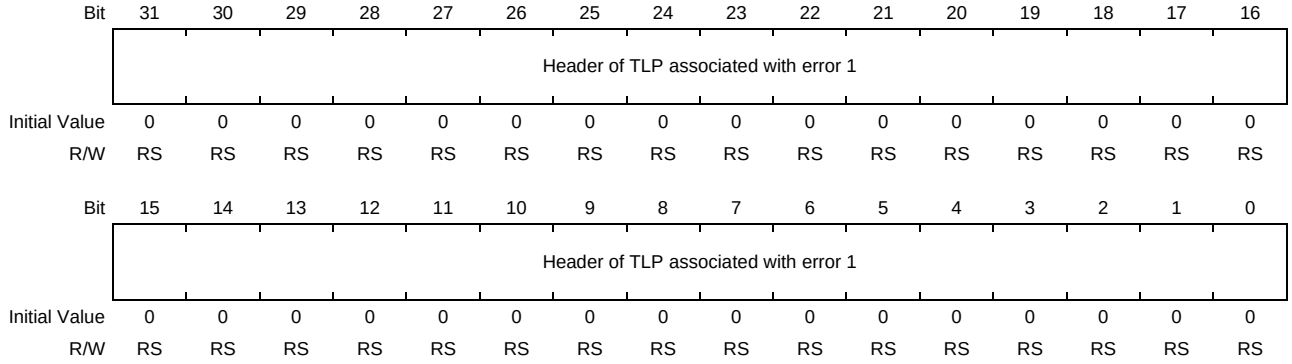
Note: Bit[4:0] The First Error Pointer is reset with PCI_RST_OUT#.

34.3.2.41 Header Log Register 0 (Offset: H'611C)



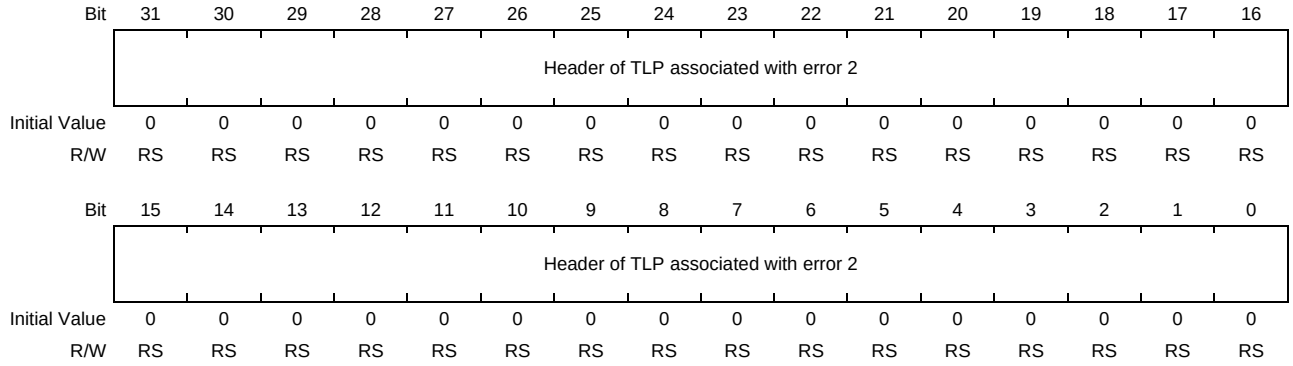
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 0	Header of TLP associated with error 0	All 0	RS	—	For the first Uncorrectable Error detected: Shows the 1st DW of the Header.

34.3.2.42 Header Log Register 1 (Offset: H'6120)



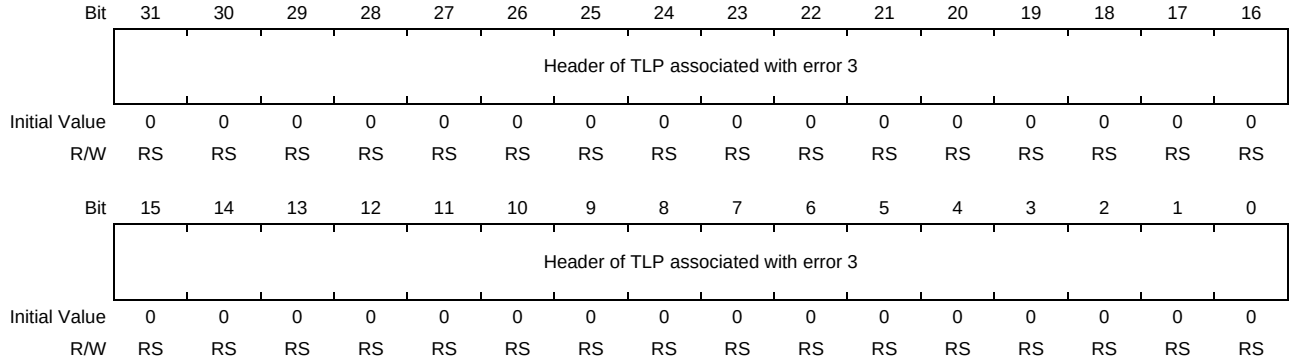
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 0	Header of TLP associated with error 1	All 0	RS	—	For the first Uncorrectable Error detected: Shows the 2nd DW of the Header.

34.3.2.43 Header Log Register 2 (Offset: H'6124)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 0	Header of TLP associated with error 2	All 0	RS	—	For the first Uncorrectable Error detected: Shows the 3rd DW of the Header.

34.3.2.44 Header Log Register 3 (Offset: H'6128)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 0	Header of TLP associated with error 3	All 0	RS	—	For the first Uncorrectable Error detected: Shows the 4th DW of the Header.

34.3.2.45 Root Error Command (Offset: H'612C)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	Fatal Error Reporting Enable	Non-Fatal Error Reporting Enable	Correctable Error Reporting Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 3	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	Fatal Error Reporting Enable	All 0	R/W	RST_CFG_B	This bit allows an interrupt to be generated when ERR_FATAL occurs in any of the hierarchies below the root port.
1	Non-Fatal Error Reporting Enable	All 0	R/W	RST_CFG_B	This bit allows an interrupt to be generated when NON_ERR_FATAL occurs in any of the hierarchies below the root port.
0	Correctable Error Reporting Enable	All 0	R/W	RST_CFG_B	This bit allows an interrupt to be generated when ERR_COR occurs in any of the hierarchies below the root port.

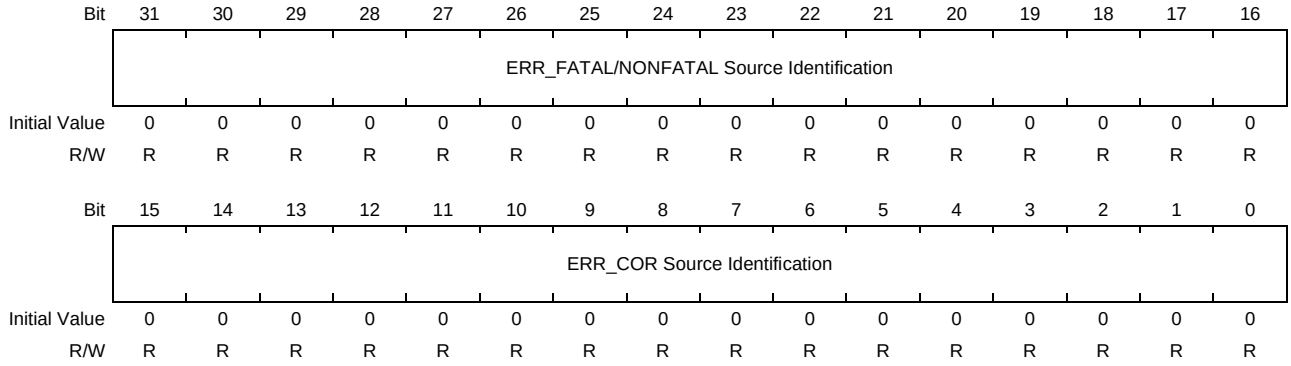
34.3.2.46 Root Error Status (Offset: H'6130)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Advanced Error Interrupt Message Number					—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	Fatal Error Message Received	Non-Fatal Error Message Received	First Uncorrectable Fatal	Multiple ERR_FATAL/NONFATAL Received	ERR_FATAL/NONFATAL Received	Multiple ERR_COR Received	ERR_COR Received
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C

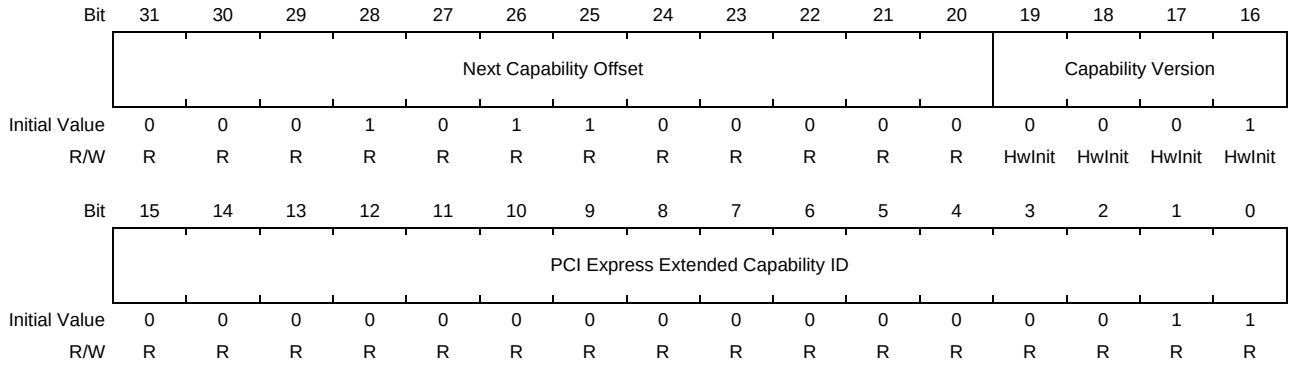
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 27	Advanced Error Interrupt Message Number	All 0	R	—	Not Implemented (H'00 fixed)
26 to 7	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	Fatal Error Message Received	0b	R/W1C	RST_RSM_B	Set when one or more ERR_FATALs are received.
5	Non-Fatal Error Message Received	0b	R/W1C	RST_RSM_B	Set when one or more ERR_NON_FATALs are received.
4	First Uncorrectable Fatal	0b	R/W1C	RST_RSM_B	Set when an uncorrectable error is first received, and the error is ERR_FATAL.
3	Multiple ERR_FATAL/ NONFATAL Received	0b	R/W1C	RST_RSM_B	Indicates that more ERR_FATAL or ERR_NON_FATAL has been received with the ERR_FATAL/NONFATAL Received bit already set.
2	ERR_FATAL/ NONFATAL Received	0b	R/W1C	RST_RSM_B	Indicates that ERR_FATAL or ERR_NON_FATAL has been received with the ERR_FATAL/NONFATAL Received bit cleared.
1	Multiple ERR_COR Received	0b	R/W1C	RST_RSM_B	Indicates that more ERR_COR has been received with the ERR_COR Received bit already set.
0	ERR_COR Received	0b	R/W1C	RST_RSM_B	Indicates that ERR_COR has been received with the ERR_COR Received bit cleared.

34.3.2.47 Error source Identification Register (Offset: H'6134)



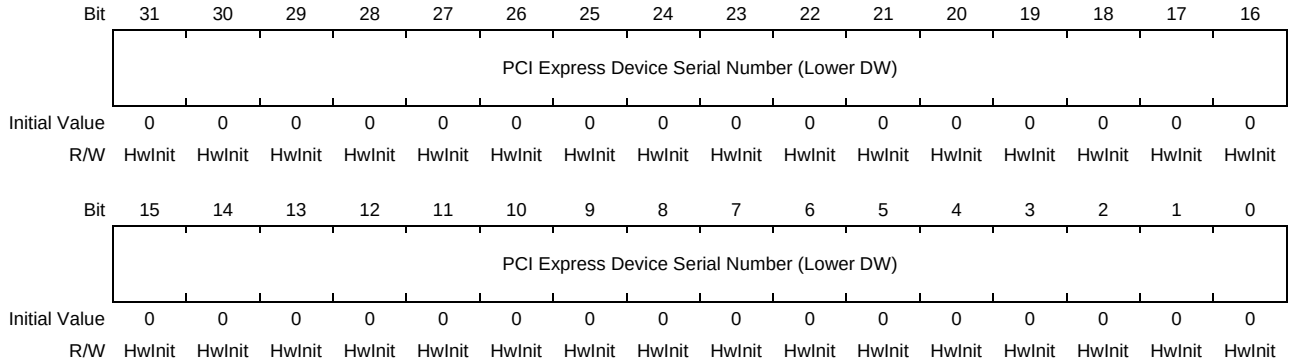
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 16	ERR_FATAL/ NONFATAL Source Identification	All 0	R	RST_RSM_B	When ERR_FATAL or ERR_NONFATAL is received without the ERR_FATAL/NONFATAL Received bit of the Root Error Status register being set, capture the requestor ID for that error.
15 to 0	ERR_COR Source Identification	All 0	R	RST_RSM_B	When ERR_COR is received without the ERR_COR Received bit in the Root Error Status register being set, the requestor ID for that error is captured.

34.3.2.48 Device Serial Number Extended Capability (Offset: H'6150)



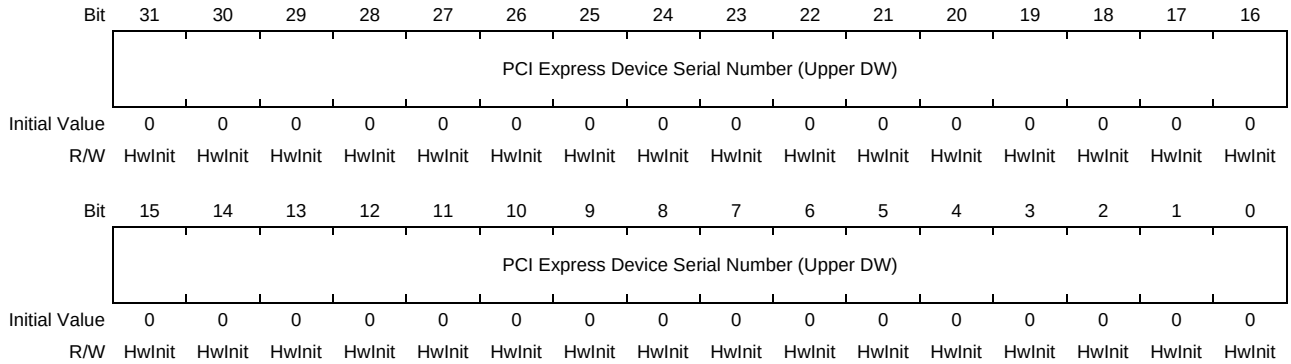
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 22	Next Capability Offset	H'160	R	RST_LOAD_B	Indicates the starting address of the L1 PM Substates Extended Capability Header. H'160 fixed.
21, 20				—	
19 to 16	Capability Version	H'1	HwInit	RST_LOAD_B	Indicates the Capability Structure version.
15 to 0	PCI Express Extended Capability ID	H'0003	R	—	Indicates the Device Serial Number Extended Capability.

34.3.2.49 Serial Number Register (Lower DW) (Offset: H'6154)



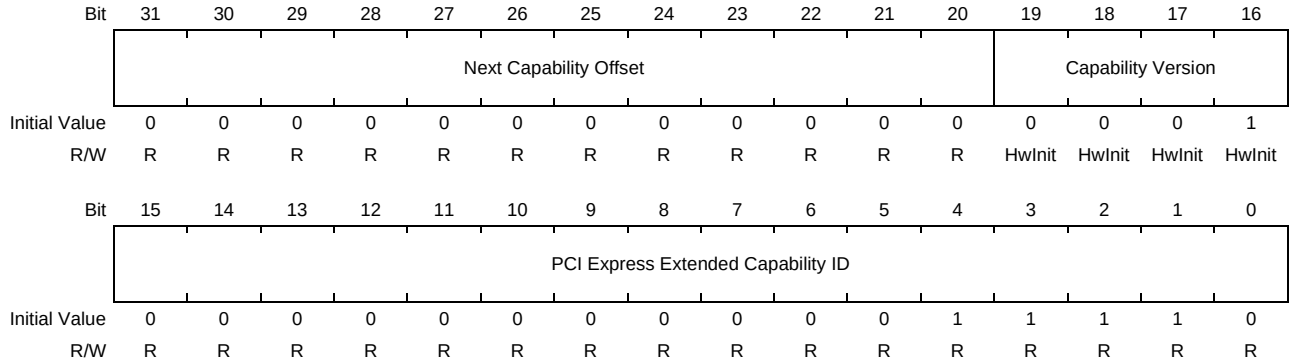
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 0	PCI Express Device Serial Number (Lower DW)	All 0	HwInit	RST_LOAD_B	Lower 32 bits of the IEEE standard's 64-bit unique ID (EUI-64). EUI-64 consists of a 24-bit company ID and 40-bit vendor-defined extensions. If a unique ID is used, write it from the UDL side during the timing indicated by the "Load period into the Configuration Register" in the **Figure 34.2, Reset Sequence for Power-On Reset**.

34.3.2.50 Serial Number Register (Upper DW) (Offset: H'6158)



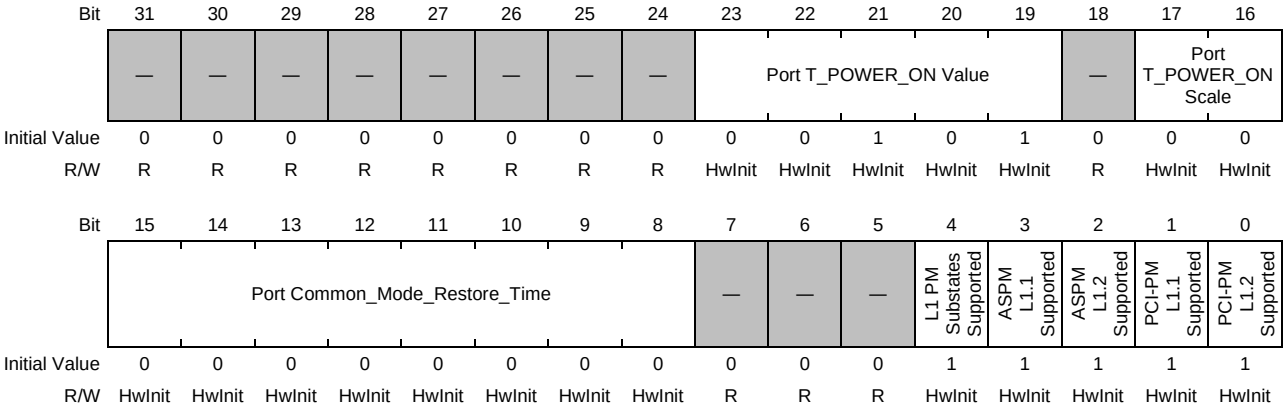
Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 0	PCI Express Device Serial Number (Upper DW)	All 0	HwInit	RST_LOAD_B	Lower 32 bits of the IEEE standard's 64-bit unique ID (EUI-64). EUI-64 consists of a 24-bit company ID and 40-bit vendor-defined extensions. If a unique ID is used, write it from the UDL side during the timing indicated by the "Load period into the Configuration Register" in the **Figure 34.2, Reset Sequence for Power-On Reset**.

34.3.2.51 L1 PM Substates Extended Capability Header (Offset: H'6160)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 20	Next Capability Offset	H'000	R	—	Indicates that this Capability List is the final List.
19 to 16	Capability Version	H'1	HwInit	RST_LOAD_B	Indicates the Capability Structure version.
15 to 0	PCI Express Extended Capability ID	H'001E	R	—	Indicate L1 PM Substates Extended Capability Header.

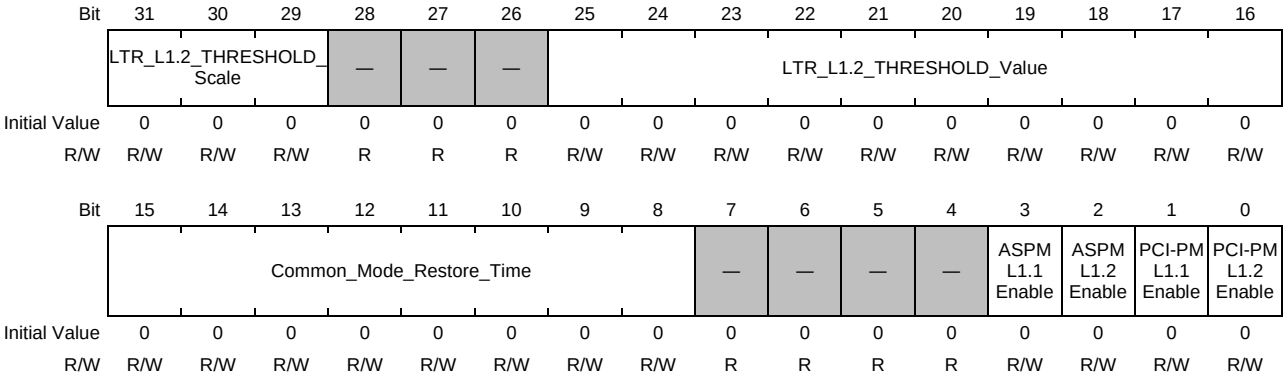
34.3.2.52 L1 PM Substates Capabilities Register (Offset: H'6164)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 24	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 19	Port T_POWER_ON Value	H'05	HwInit	RST_LOAD_B	Along with the Port T_POWER_ON Scale field in the L1 PM Substates Capabilities register sets the time (in µs) that this Port requires the port on the opposite side of Link to wait in L1.2. Exit after sampling CLKREQ# asserted before actively driving the interface. The value of Port T_POWER_ON is calculated by multiplying the value in this field by the scale value in the Port T_POWER_ON Scale field in the L1 PM Substates Capabilities register. Default value is 00101b Required for all Ports for which either the PCI-PM L1.2 Supported bit is Set, ASPM L1.2 Supported bit is Set, or both are Set, otherwise this field is of type RsvdP.
18	—	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17, 16	Port T_POWER_ON Scale	00b	HwInit	RST_LOAD_B	Specifies the scale used for the Port T_POWER_ON Value field in the L1 PM Substates Capabilities register. Range of Values 00b: 2 µs 01b: 10 µs 10b: 100 µs 11b: Reserved Required for all Ports for which either the PCI-PM L1.2 Supported bit is Set, ASPM L1.2 Supported bit is Set, or both are Set, otherwise this field is of type RsvdP.
15 to 8	Port Common_Mode_Restore_Time	All 0	HwInit	RST_LOAD_B	Time (in µs) required for this Port to re-establish common mode as described in table (reference "PCI Express™ Base Specification" Table.5-11 L1.2 Timing Parameters "TCOMMONMODE"). Required for all Ports for which either the PCI-PM L1.2 Supported bit is Set, ASPM L1.2 Supported bit is Set, or both are Set, otherwise this field is of type RsvdP.
7 to 5	—	000b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	L1 PM Substates Supported	1b	HwInit	RST_LOAD_B	L1 PM Substates support
3	ASPM L1.1 Supported	1b	HwInit	RST_LOAD_B	ASPM L1.1 support

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
2	ASPM L1.2 Supported	1b	HwInit	RST_LOAD_B	ASPM L1.2 support
1	PCI-PM L1.1 Supported	1b	HwInit	RST_LOAD_B	PCI-PM L1.1 support
0	PCI-PM L1.2 Supported	1b	HwInit	RST_LOAD_B	PCI-PM L1.2 support

34.3.2.53 L1 PM Substates Control 1 Register (Offset: H'6168)



Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 29	LTR_L1.2_THRESHOLD_Scale	000b	R/W	RST_CFG_B	This field provides a scale for the value contained within the LTR_L1.2_THRESHOLD_Value. Encoding is the same as the LatencyScale fields in the LTR Message. The default value for this field is 000b. Hardware operation is undefined if software writes a Not-Permitted value to this field. This field must only be modified when the ASPM L1.2 Enable bit is Clear. The Port behavior is undefined if this field is modified when the ASPM L1.2 Enable bit is Set. Required for all Ports for which the ASPM L1.2 Supported bit is Set, otherwise this field is of type RsvdP.
28 to 26	—	000b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25 to 16	LTR_L1.2_THRESHOLD_Value	All 0	R/W	RST_CFG_B	Along with the LTR_L1.2_THRESHOLD_Scale, this field indicates the LTR threshold used to determine if entry into L1 results in L1.1 (if enabled) or L1.2 (if enabled). The default value for this field is 00_0000_0000b. This field must only be modified when the ASPM L1.2 Enable bit is Clear. The Port behavior is undefined if this field is modified when the ASPM L1.2 Enable bit is Set. Required for all Ports for which the ASPM L1.2 Supported bit is Set, otherwise this field is of type RsvdP.
15 to 8	Common_Mode_Restore_Time	H'00	R/W	RST_CFG_B	Sets value of TCOMMONMODE (in µs), which must be used by the Downstream Port for timing the re-establishment of common mode, as described in . This field must only be modified when the ASPM L1.2 Enable and PCI-PM L1.2 Enable bits are both Clear. The Port behavior is undefined if this field is modified when either the ASPM L1.2 Enable and/or PCI-PM L1.2 Enable bit(s) are Set. Required for Downstream Ports for which either the PCI-PM L1.2 Supported bit is Set, ASPM L1.2 Supported bit is Set, or both are Set, otherwise this field is of type RsvdP. This field is of type RsvdP for Upstream Ports.
7 to 4	—	H'0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	ASPM L1.1 Enable	0b	R/W	RST_CFG_B	ASPM L1.1 support
2	ASPM L1.2 Enable	0b	R/W	RST_CFG_B	ASPM L1.2 support
1	PCI-PM L1.1 Enable	0b	R/W	RST_CFG_B	PCI-PM L1.1 support

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
0	PCI-PM L1.2 Enable	0b	R/W	RST_CFG_B	PCI-PM L1.2 support

34.3.2.54 L1 PM Substates Control 2 Register (Offset: H'616C)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	T_POWER_ON Value					—	T_POWER_ON Scale	
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Valid Reset	Description
31 to 8	—	All 0	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 3	T_POWER_ON Value	H'05	R/W	RST_CFG_B	Along with the T_POWER_ON Scale sets the minimum amount of time (in µs) that the Port must wait in L1.2. Exit after sampling CLKREQ# asserted before actively driving the interface. T_POWER_ON is calculated by multiplying the value in this field by the value in the T_POWER_ON Scale field. This field must only be modified when the ASPM L1.2 Enable and PCI-PM L1.2 Enable bits are both Clear. The Port behavior is undefined if this field is modified when either the ASPM L1.2 Enable and/or PCI-PM L1.2 Enable bit(s) are Set. Default value is 0_0101b Required for all Ports that support L1.2, otherwise this field is of type RsvdP
2	—	0b	R	—	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	T_POWER_ON Scale	00b	R/W	RST_CFG_B	Specifies the scale used for T_POWER_ON Value. Range of Values 00b: 2 µs 01b: 10 µs 10b: 100 µs 11b: Reserved Required for all Ports that support L1.2, otherwise this field is of type RsvdP. This field must only be modified when the ASPM L1.2 Enable and PCI-PM L1.2 Enable bits are both Clear. The Port behavior is undefined if this field is modified when either the ASPM L1.2 Enable and/or PCI-PM L1.2 Enable bit(s) are Set. Default value is 00b

34.4 Function Description

34.4.1 PCIe Core Functionality

This module is configured based on the Base Spec 4.0. For the detailed specification, refer to the Base Spec 4.0

34.4.2 Issuing of PCIe Request and Register Access (From AXI)

The following gives functional description on access to the PCI Express core by the AXI bus.

34.4.2.1 PCIe Requests which can be Issued (Supported Commands for TX)

Requests to be issued via PCIe windows

MRd: Memory Read Request

MWr: Memory Write Request

Requests to be issued by the registers

MRd: Zero-Length Memory Read Request

CfgRd0: Configuration Read Type 0

CfgWr0: Configuration Write Type 0

CfgRd1: Configuration Read Type 1

CfgWr1: Configuration Write Type 1

Msg: Message Request

MsgD: Message Request with data payload

Unsupported requests (issuance prohibited)

IORd: I/O Read Request

IOWr: I/O Write Request

MRdLk: Memory Read Request-Locked

CplDLk: Completion for Locked Memory Read

34.4.2.2 Register access

- Internal register read/write
- Configuration register read/write

The following restrictions apply to access to the configuration registers.

- Byte-lane transfer is only acceptable if the transfer consists of consecutive valid bytes.
- If the valid bytes are not consecutive, only values which are no greater than 32 bits (1 dword) and do not span the dword alignment are acceptable.
  [Example] A value which can represent a dword TLP for PCIe

34.4.2.3 Issuing Memory Requests

Access from the AXI side is converted into a PCIe request and then issued. Up to eight windows can be allocated. Note that only incremental bursts and fixed bursts (of 1 beat) are acceptable. Wrapping bursts and fixed bursts (of 2 or more beats) are prohibited. If these prohibited bursts are received, the module operates as follows.

- Although the response will in general be “OKAY” (except in cases that involved a protocol error of the AXI), this is
- beyond the scope of guaranteed operation.
- Unexpected requests may be issued to the PCIe bus.
- Unexpected registers may be modified.

To maintain the order between memory writing and memory reading, only issue next transactions after the reception of responses to writing. Buffering is not available in the issuing of memory write requests. (Refer to the description of AWCACHE* of Mode Set 0 Register (offset: H’314).)

Note that when accessing outside the set Window area, it responds “OKAY”, but it is discarded inside this module.

(1) Memory Write Transaction from the AXI

A write transaction from the AXI via a window is converted into a MWr command and then issued.

- Number of write transactions which can be accepted at a time: 1
- Write data are held in an internal buffer.
- The order between memory write transactions is preserved (with the exception of transactions for messages and configuration).
- The order of memory write and other transactions is not preserved (a preceding memory read may be overtaken).
- To preserve the order between memory writing and memory reading or of Msg and MsgD, make sure that a next transaction is only issued after a response is returned. Buffering is not available in the issuing of memory write requests.
- Writing does not proceed if the PCI power state is not D0.
- Transactions where all bits of WSTRB are 0 during a burst return an “OKAY” response, but the written data are not reflected.

(2) Memory Read Transaction from the AXI

A read transaction from the AXI via a window is converted into an MRd command and then issued.

- Number of read transactions which can be accepted at a time: 1 to 8.
- Read data are held in an internal buffer to preserve the order of transactions with the same ID.
- To preserve the order of memory read transactions from the same master ID, the system can also wait until the indicator of completion of a preceding read transaction is returned. The method of waiting is selectable as either of the above by the setting of an internal register as listed in the table below.

PCIe Request Order*1	Method of Waiting	Performance	Severity of Order
0 (Default)	Make Read Data wait in internal buffer	✓	—
1	Waiting to issue a Read Request	—	✓

Note 1. The order can be set by using the PCIe Request Order bit (bit [0]) of Mode Set 1 Register (offset: H'318).

- The order of memory read transactions from a different master ID is not preserved.
- The order of memory read and other transactions is not preserved.
- To preserve the order between memory reading and memory writing, make sure that a next transaction is only issued after a response is returned. Buffering is not available in the issuing of memory write requests.
- MRdLk requests are not supported and therefore cannot be issued.
- Reading does not proceed if the PCI power state is not D0.

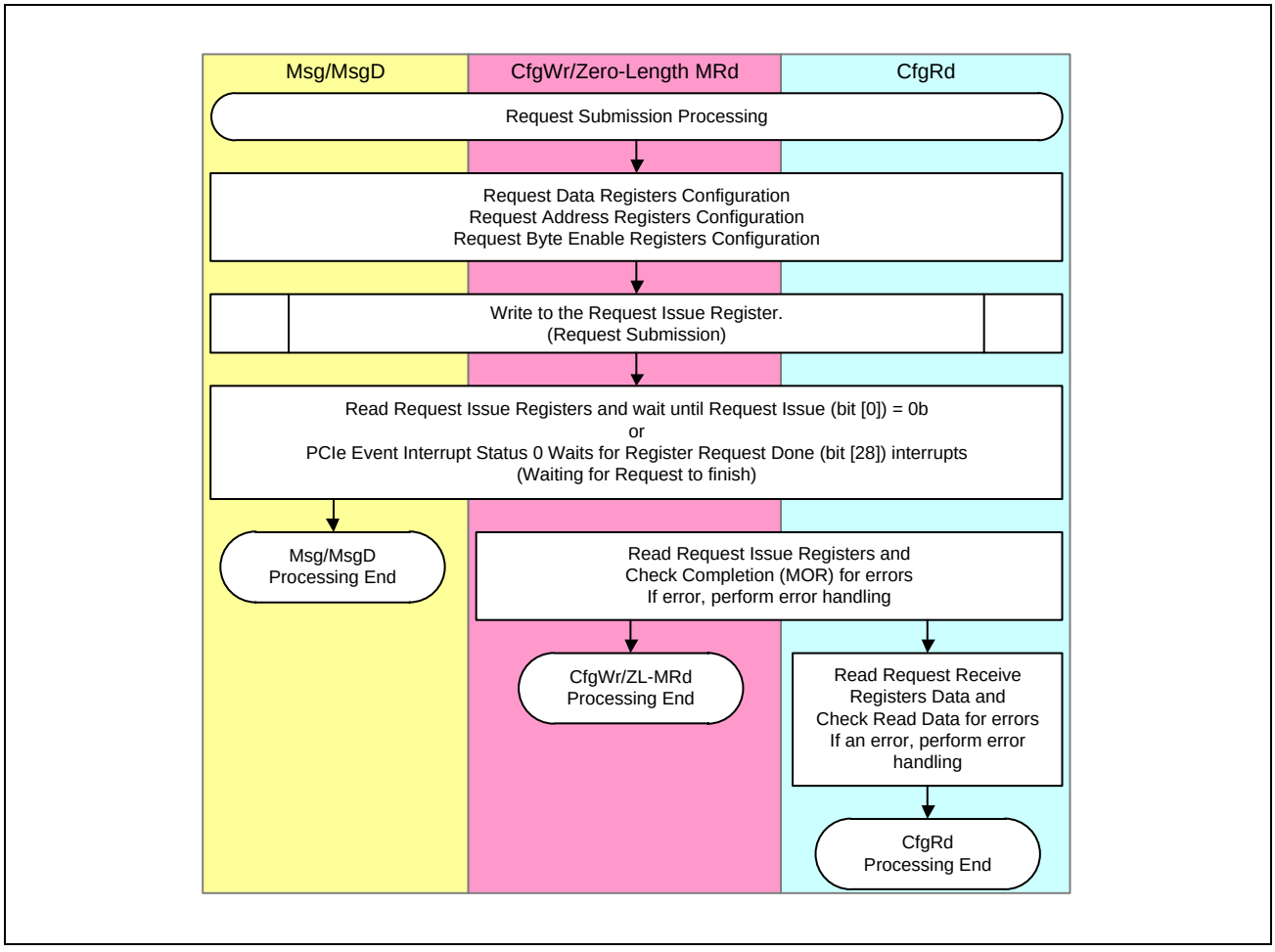
34.4.2.4 Issuing Special Requests

The following requests can be issued by controlling the internal registers.

- Configuration Read Type0/1
- Configuration Write Type0/1
- Zero-Length Memory Read Request
- Message Request
- Message Request with data payload

The internal registers are only writable from the AXI side. Attempted writing from the PCIe side is ignored and the write operation returns an “OKAY” response, but the written data are not reflected.

The figure below shows the flow of issuing requests by the registers.



(1) Issuing a Configuration Request

AXI Slave I/F accesses internal registers, sets destination in Request Address Registers 1 (Offset: H’90), then sets Request Issue (bit [0]) in Request Issue Registers (Offset: H’9C), and automatically issues a Configuration Request.

(a) Flow of issuing Configuration Requests

An example of the flow for issuing a Configuration Requests is described below.

1. Set a destination in Request Address Registers 1 (Offset: H’90)

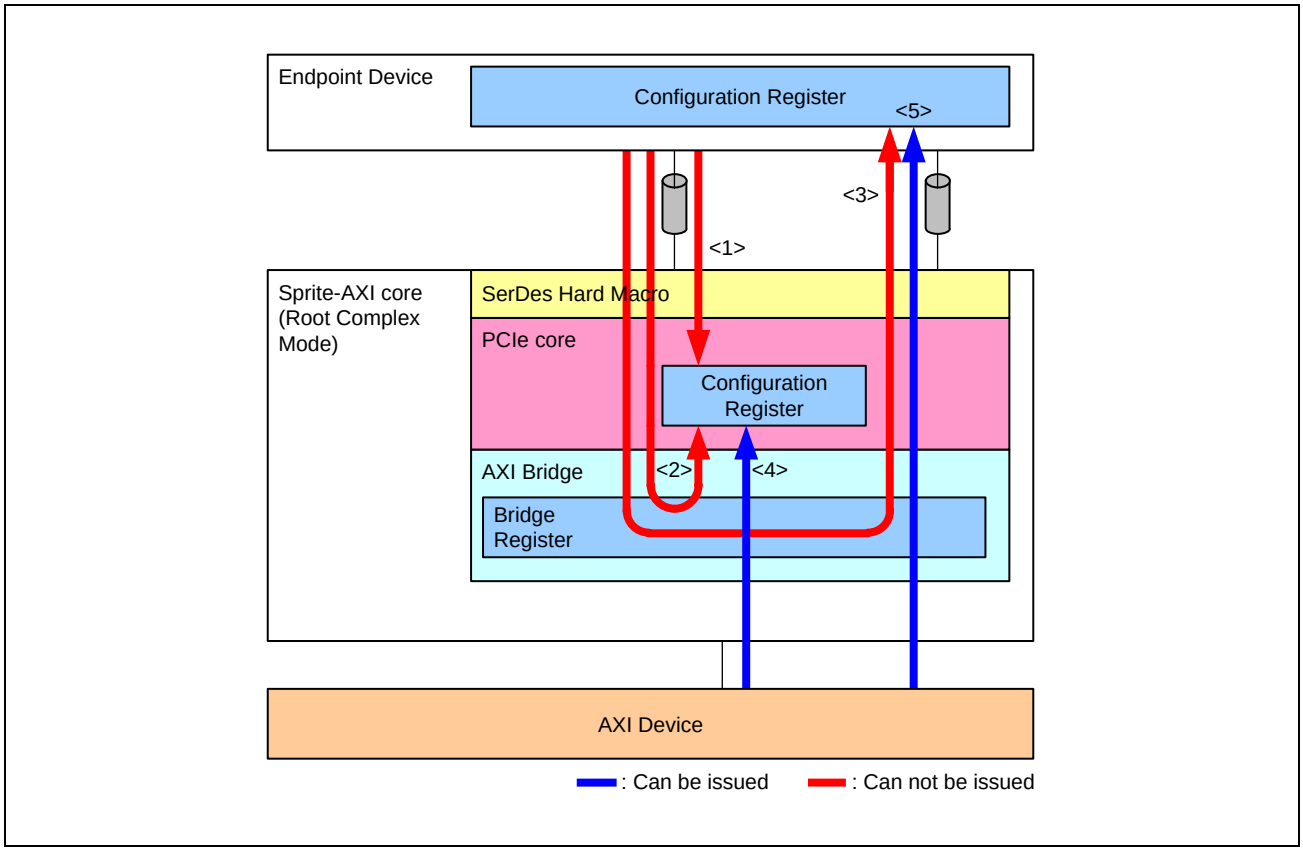
Bits [31: 24]:	Bus Number [7:0]	Set 0 to 255
Bits [23:19]:	Device Number [4:0]	Set 0 to 31
Bits [18:16]:	Function Number [2:0]	Set 0 to 7
Bits [15:12]:	Reserved	Set 0000b (Fixed value)
Bits [11:8]:	Extend Register Number [3:0]	
Bits [7:2]:	Register Number [7:0]	
Bits [1:0]:	Reserved	Set 00b (Fixed value)

2. Set to Byte Enable in Request Byte Enable Registers (Offset: H’98).
   Set Enable/Disable corresponding to each Byte position if Read/Write is desired for each Byte (Default: 1111b)
3. For CfgWr, set Write Data in Request Data Registers (Offset: H’80/H’84/H’88). (Not needed for CfgRd)
4. A Configuration Request is automatically issued by setting the Request Issue (bit [0]) in the Request Issue Registers (Offset: H’9C) to 1b. (Set CfgWr/CfgRd in TR type (bit [11:8]) of Request Issue Registers (Offset: H’9C).)
5. Wait for the Request to finish by reading the Request Issue Registers (Offset: H’9C) and waiting for the Request Done (bit [28]) interrupt of wait (Polling) or PCIe Event Interrupt Status 0 Registers (Offset: H’204) until the Request Issue (bit [0]) reaches 0b.
6. For CfgRd, the CplD from the opposing Endpoint device is stored in Request Receive Data Registers (Offset: H’8C). It is possible to check this register by reading it.

(b) Configuration Request Issue Patterns

The following table shows whether the Configuration Request can be issued in Root complex mode.

No.	Request Direction	Access
1	EP to PCIe Core	Can not be issued
2	EP to Register → PCIe Core	Can not be issued
3	EP to Register → EP	Can not be issued
4	AXI to PCIe Core	Can be issued
5	AXI to Register → EP	Can be issued



(2) Issuing Message Requests

A message request is automatically issued by setting the Request Issue bit (bit [0]) of the Request Issue Register after access to an internal register through the AXI slave interface and setting a destination in Request Address Register 1.

(a) Flow of Issuing Message Requests

An example of the flow of issuing message requests is described below.

1. Set a destination in Request Address Register 1.

Bits [31:27]:	Reserved	Set 0_0000b (Fixed value)
Bits [26:24]:	Routing Type	Specify the message routing
Bits [23:8]:	Reserved	Set H'0000 (Fixed value)
Bits [7:0]:	Message Code	Specify the message code

2. Set the 3rd header in Request Data Register 1 as required.
3. Set the 4th header in Request Data Register 2 as required.
4. For MsgD, set write data in Request Data Register 3 (not required in the case of Msg)
5. Message requests are automatically issued by setting the Request Issue bit (bit [0]) of the Request Issue Register to 1b (the TR type bits (bits [11:8]) of the Request Issue Register are used to set Msg/MsgD).
6. Read the Request Issue Register and wait (polling) or wait for an interrupt of Request Done (bit [28]) of PCIe Event Interrupt Status 0 Register until the Request Issue bit (bit [0]) is set to 0b, wait for the request to end.

(b) Notes when Issuing Message Requests

Points to note when issuing message requests are described below.

- INTx and error type messages are automatically issued. They do not require issuing as message requests by the user (prohibited).
- Set message headers appropriately.
- A value to be set in a message header must be 0 except when sending a vendor defined message.

(3) Issuing Zero-Length Read

A zero-length read request is automatically issued by setting a destination in Request Address Register 1 after access to an internal register through the AXI slave interface and setting the Request Issue bit (bit [0]) of the Request Issue Register.

(a) Flow of Issuing Zero-Length Read Requests

An example of the flow of issuing zero-length read requests is described below.

1. Set a destination in Request Address Register 1.

Bits [31:2]:	Address	Set the address
Bits [1:0]:	Reserved	Set 00b (Fixed value)

2. Set a destination in Request Address Register 1.

Bits [64:32]:	Address	Set the address

3. Set Byte Enable (0000b) in Request Byte Enable Register.
4. Zero-length read requests are automatically issued by setting the Request Issue bit (bit[0]) of the Request Issue Register to 1b (the TR type bits (bits [11:8]) of the Request Issue Register are used to set zero-length read).
5. Read the Request Issue Register and wait (polling) or wait for an interrupt of Request Done (bit [28]) of PCIe Event Interrupt Status 0 Register until the Request Issue bit (bit [0]) is set to 0b, wait for the request to end.

An example of using this request is to issue a zero-length read request to the same address where the write request was issued for a Posted Request that did not receive a response, such as a memory write, and use the response as a completion notification for the write request.

34.4.3 Issuing of AXI Transaction and Register Access (From PCIe)

The following gives functional description on access to the AXI bus by the PCI Express core.

34.4.3.1 PCIe Requests which can be Received (Supported Commands for RX)

MRd: Memory Read Request

MWr: Memory Write Request

CfgRd0: Configuration Read Type 0

CfgWr0: Configuration Write Type 0

Msg: Message Request

MsgD: Message Request with data payload

Cpl: Completion

CplD: Completion with Data

The following requests are not supported.

MRdLk: Memory Read Request-Locked

IORd: I/O Read Request

IOWr: I/O Write Request

CplLK: Completion for Locked Memory Read without Data

CplDLk: Completion for Locked Memory Read

CfgRd1: Configuration Read Type 1

CfgWr1: Configuration Write Type 1

AtomicOP is not supported.

34.4.3.2 Issuing AXI Transaction

Access from the PCIe side is converted into an AXI transaction and then issued. Up to eight windows can be allocated.

To avoid a read-after-write (RAW) hazard, only issue a next memory write request following reception of the completion TLP for the previous one.

Table 34.13 AXI transaction from PCIe Request

PCIe Request Order	Burst Type	Bath Size	Split
MWr/MRd	Increment	64/128	Yes
		32	No
IOWr/IORd	Unsupported request		
MRdLk	Unsupported request		

(1) Memory Write Transaction from the PCIe

- Allowable number of write requests to be issued: Variable (the value is set by the AXI Max Issue Write bits (bits [15:12]) of Mode Set 1 Register).
- Write data are not held in an internal buffer (buffering within the PCIe core).
- The order between memory write transactions is preserved.
- The order of memory writing and memory reading is not preserved.
- To preserve the order between memory writing and memory reading, make sure that a next request is only issued after a completion TLP is returned.
- AWCACHE and AWPROT can be set by using Mode Set 0 Register.
- In cases of contention between DMA transfer (from PCIe to AXI) and memory write transaction, requests are accepted in turn in round-robin fashion.

(2) Memory Read Transaction from the PCIe

- Allowable number of read requests to be issued: Variable (the value is set by the AXI Max Issue Read bits (bits[11:8]) of Mode Set 1 Register).
- Since the PCI specification requires dword-aligned memory addresses in transfer, invalid byte lanes may be read. If an AXI slave as a read destination has a register or FIFO buffer which has been cleared by being read, the transaction may be non-compliant.
- Data read are held in an internal buffer.
- The order between memory read operations is preserved.
- A preceding memory write is not overtaken.
  Data read are held in an internal buffer until a preceding memory write is completed.
- When a zero-length read request is received, an “OKAY” response is returned, but the written data are not reflected and a completion TLP is transmitted after waiting for the completion of the preceding memory write.
  Data read are held in an internal buffer until a preceding memory write is completed.
- ARCACHE can be set by using Mode Set 0 Register.
- In cases of contention between DMA transfer (from AXI to PCIe) and memory read transaction, requests are accepted in turn in round-robin fashion.

34.4.3.3 Narrow Transfer

Narrow transfer to be initiated by the AXI transfer only supports the following AXI transactions.

- MAxBURST = INCR
- MAxSIZE = H’0 to H’2 (8 bits / 16 bits / 32 bits)
- MAxLEN = H’0 (1 beat)

The following restrictions apply to TLPs from the PCIe module in narrow transfer.

- The length is 1 dword.
- The First Byte Enable bit only supports the following.
  - MAxSIZE = H’0 (8 bits): 1000b / 0100b / 0010b / 0001b
  - MAxSIZE = H’1 (16 bits): 1100b / 0011b
  - MAxSIZE = H’2 (32 bits): 1111b

34.4.4 DMA Functions

This section explains functions of the DMAC within the module. Control by registers and by descriptors are both supported as methods of DMA control. The method of control is independently selected per channel.

34.4.4.1 Register-Type Transfer

DMA transfer from AXI to PCIe or vice versa is handled by software making register settings. The table below lists the DMAC registers for execution of register-type DMA transfer.

Table 34.14 Registers Related to Register-Type DMA Transfer

Common Control	
H'800	DMAC Control
H'808	DMAC Interrupt Enable
H'80C	DMAC Interrupt Status
Channel Control	
H'900 + channel offset	DMA Channel Control
DMA Setting	
H'924 + channel offset	DMA Transaction Control (Descriptor H'04)
H'928 + channel offset	DMA Size (Descriptor H'08)
H'930 + channel offset	DMA Source Lower Address (Descriptor H'10)
H'934 + channel offset	DMA Source Upper Address (Descriptor H'14)
H'938 + channel offset	DMA Destination Lower Address (Descriptor H'18)
H'93C + channel offset	DMA Destination Upper Address (Descriptor H'1C)
DMA Status	
H'950 + channel offset	DMA Reset Size
H'960 + channel offset	AXI Request Address (Lower)
H'964 + channel offset	AXI Request Address (Upper)
H'968 + channel offset	PCIe Request Address (Lower)
H'96C + channel offset	PCIe Request Address (Upper)
H'978 + channel offset	DMAC Error Status

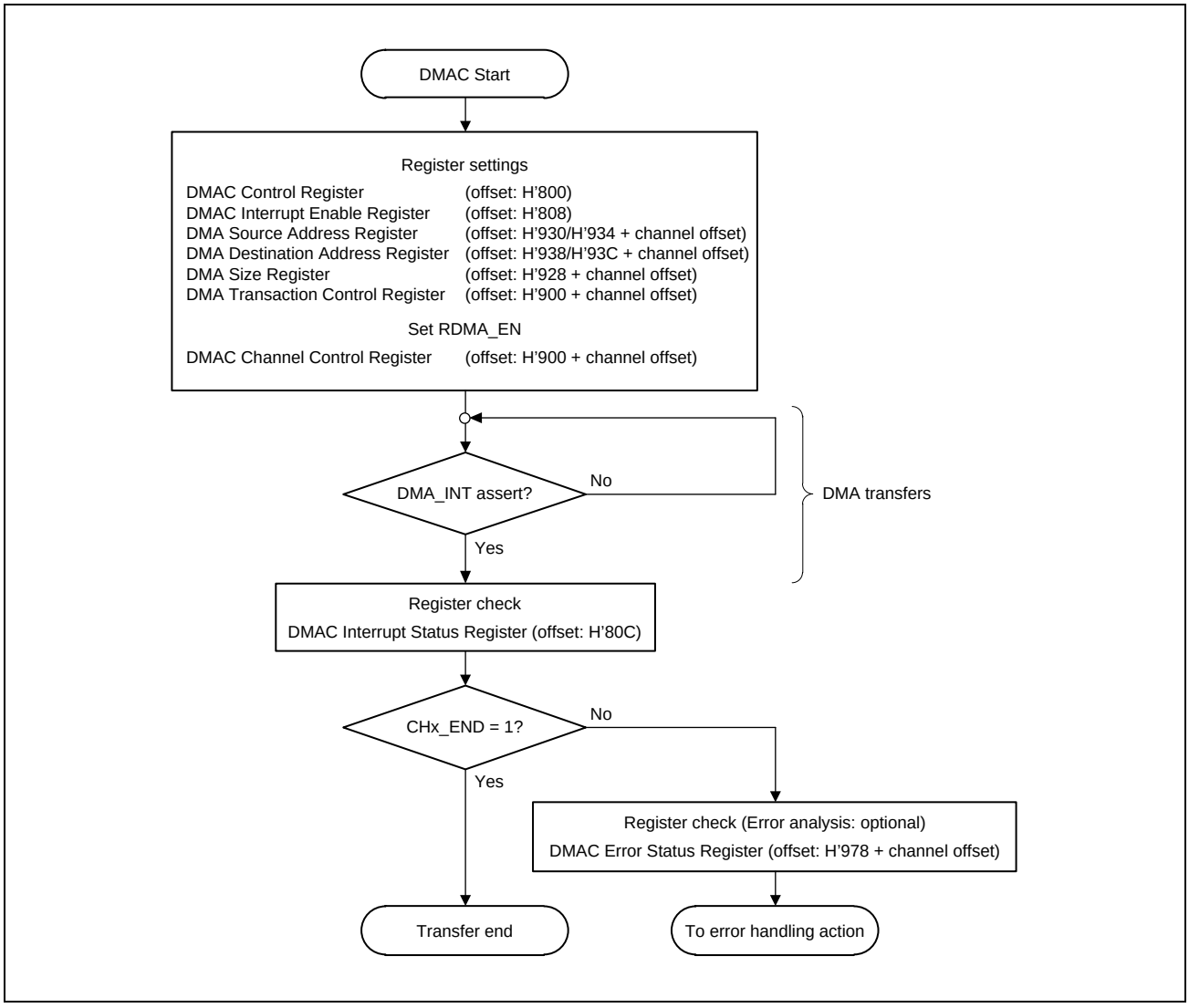
Note: During Register-type DMA, the maximum number of bytes transferred at one time is 4 Gbytes.

(1) Flow of Operations

The following describes the procedures for settings to activate and stop the DMAC in the case of register-type transfer.

(a) Activation and Normal Operation

Before activating the DMAC, set the PCIe and AXI windows. After that, set the registers of the DMAC to start the DMAC.

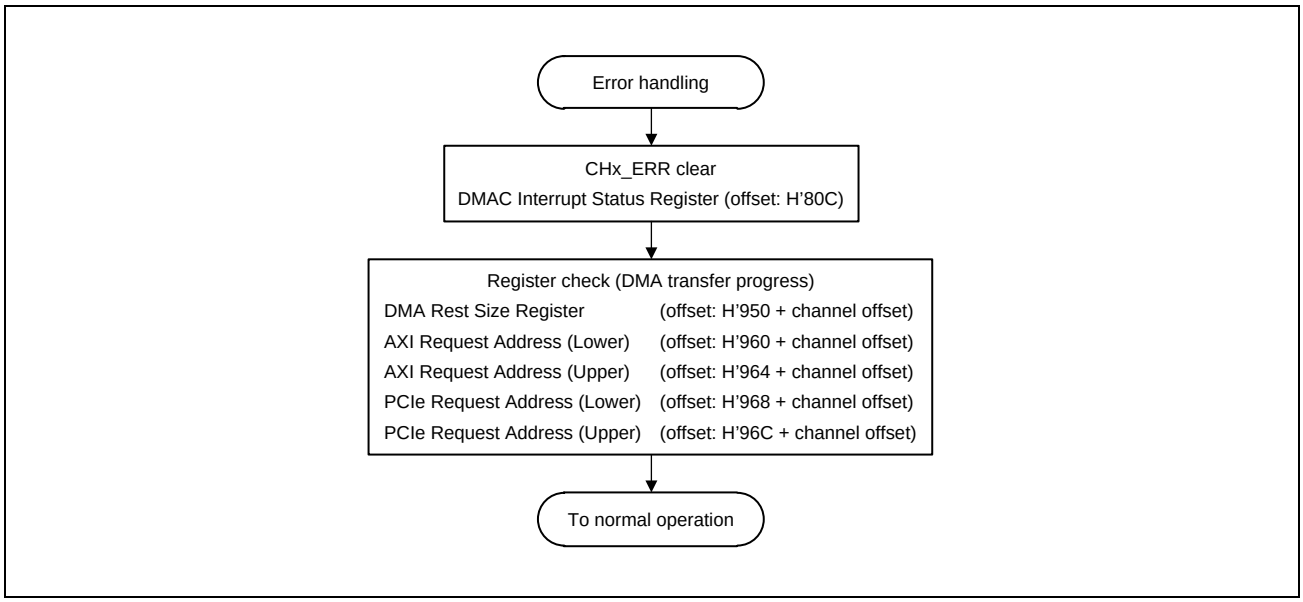


(b) Error Processing

If an error is detected in the AXI bus or a PCIe link while DMA transfer is in progress, the DMA_INT interrupt signal is asserted.

Check the channel where the error was found (CHx_ERR = 1b) by using the DMAC Interrupt Status Register (offset: H’80C) and also the type of error by using the DMAC Error Status Register (offset: H’968 + channel offset) as required.

Clear the CHx_ERR bit that is currently set in the DMAC interrupt status register (offset: H’80C) by writing 0b to it and check the address to which transfer was most recently completed before it was stopped and the remaining number of bytes for transfer by reading the DMA Status registers (offset: H’950 + channel offset to H’96C + channel offset) for the given channel.



34.4.4.2 Descriptor-Type Transfer

Consecutive DMA transfer is achieved by sequentially reading descriptors which indicate parameters of DMA transfer. Descriptors are allocated to the AXI memory space, and the descriptor lists indicate the addresses where the descriptors start. This DMAC has a queue for storing multiple descriptors lists and these descriptor lists are written by software. The first list loaded in the queue is executed after being moved to become the descriptor list for execution following the detection of the condition for starting DMA transfer (the DMAC retains up to nine lists, including the one currently being executed).

Descriptors can have a chained configuration; execution of a descriptor list ends on detection of the last descriptor, and if the queue has a next list, execution of the next descriptor follows.

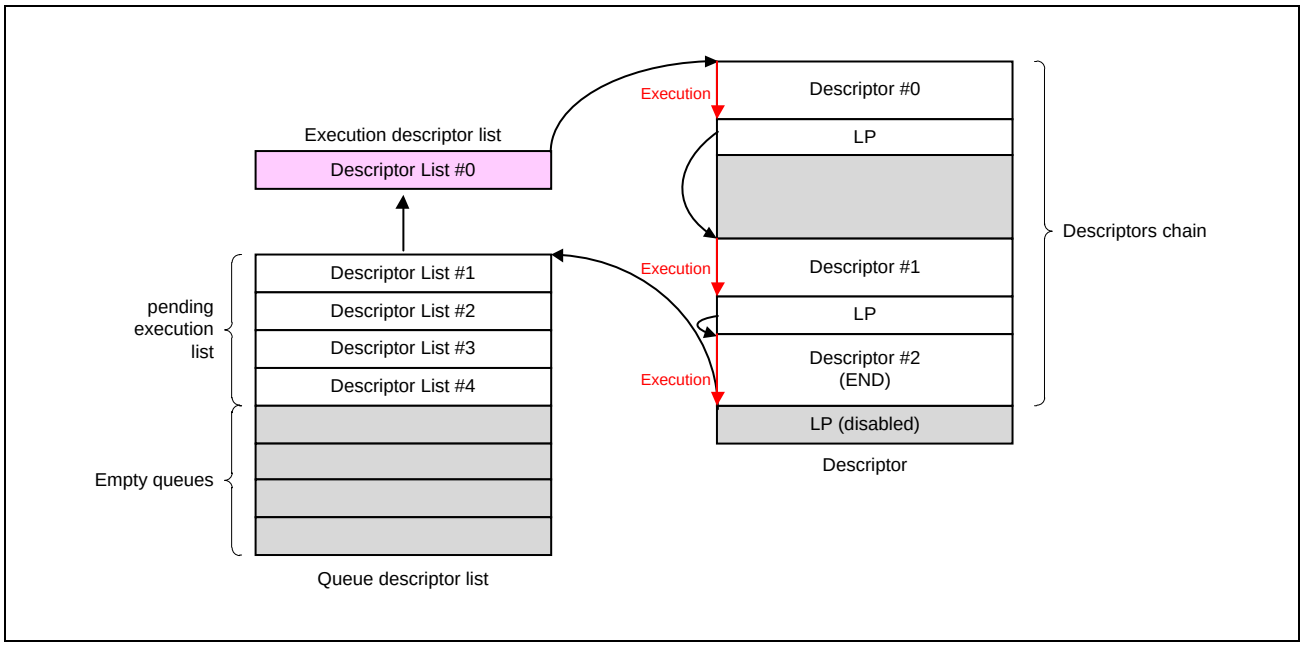


Figure 34.3 Descriptor Chain

The table below lists the DMAC registers for execution of descriptor-type DMA transfer.

Table 34.15 Registers Related to Descriptor-Type DMA Transfer

Common Control	
H'800	DMAC Control
H'808	DMAC Interrupt Enable
H'80C	DMAC Interrupt Status
Channel Control	
H'900 + channel offset	DMA Channel Control
H'908 + channel offset	Descriptor Start Address (Lower)
H'90C + channel offset	Descriptor Start Address (Upper)
H'910 + channel offset	QUE Entry
DMA Setting	
H'920 + channel offset	DMA Descriptor Control (Descriptor H'00)
H'924 + channel offset	DMA Transaction Control (Descriptor H'04)
H'928 + channel offset	DMA Size (Descriptor H'08)
H'930 + channel offset	DMA Source Lower Address (Descriptor H'10)
H'934 + channel offset	DMA Source Upper Address (Descriptor H'14)
H'938 + channel offset	DMA Destination Lower Address (Descriptor H'18)
H'93C + channel offset	DMA Destination Upper Address (Descriptor H'1C)
H'940 + channel offset	DMA Descriptor Lower Link Pointer (Descriptor H'20)
H'944 + channel offset	DMA Descriptor Upper Link Pointer (Descriptor H'24)
DMA Status	
H'950 + channel offset	DMA Rest Size
H'960 + channel offset	AXI Request Address (Lower)
H'964 + channel offset	AXI Request Address (Upper)
H'968 + channel offset	PCIe Request Address (Lower)
H'96C + channel offset	PCIe Request Address (Upper)
H'970 + channel offset	QUE Status
H'978 + channel offset	DMAC Error Status

(1) Descriptor List Queues

The DMAC has queues for storing descriptor lists for each of the channels. The number of FIFO queue stages is eight, so it can hold up to eight descriptor list entries (not including the one currently being executed).

Software places descriptor list entries in the queue. The descriptor lists to be entered are set by the combination of the QUE_ENTRY (upper) register and the QUE_ENTRY (lower) register in 64-bit (or 8-byte) units.

Since the target registers are also accessible in 8-bit units, making the settings in byte units through multiple accesses is also possible. A descriptor list entry is placed in the queue at the time when the QUE_ENTRY (upper) bits [31:24] are written. When making the settings and entering lists in the queue, take care with the order of register access (values written to the QUE_ENTRY (upper) bits [31:26] will not affect the register values and settings because they are read-only bits).

When the DMAC detects the presence of a list which has been entered in the queue, the first list entry is moved to become the pointer to the next descriptor for execution and DMA transfer starts.

(2) Descriptor List Format

The table below is the format of a descriptor list.

Table 34.16 Descriptor List Format

Offset	Byte 3			Byte 2	Byte 1	Byte 0
H'00	DSA[31:0] (Descriptor Start Address)					
H'04	DSA[63:32] (Descriptor Start Address)					
H'08	Reserved	EI	LS	Reserved	LABEL [15:0]	

Field Name	Description
DSA[63:0]	Descriptor start address for DAM. Sets the address where the first descriptor is stored. 16-byte alignment and 1 descriptor (H'00 to H'24) beyond the 4K boundary is prohibited. Set the lower 6 bits [5:0] to be 00_0000b.
EI	End Interrupt bit. This bit indicates whether an interrupt (Interrupt Status CHx_END of the DMAC Interrupt Status Register (offset: H'80C)) is or is not conveyed when processing of this descriptor list is completed. 1b: The interrupt is conveyed. 0b: The interrupt is not conveyed.
LS	List Stop bit. This bit indicates whether DMA transfer is or is not to be stopped on completion of processing for this descriptor list entry. 1b: Stopped 0b: Not stopped On completion of the list in which the setting of this bit is 1b, the QUE_EN bit of the DMA control register is cleared.
LABEL[15:0]	This field represents the label of the list. There are no special rules for the setting procedures. The value can be set as desired.

Operations by the settings of the EI and LS bits following the completion of the descriptor list are as follows.

Table 34.17 Details of the EI/LS Bits

EI	LS	
0	0	No interrupt. DMA transfer is not stopped. If the queue has a next list, execution of the next descriptor follows.
0	1	No interrupt. DMA transfer is stopped (QUE_EN is cleared).
1	0	Interrupt (DMAC Interrupt Status Register (Offset: H'80C), Interrupt Status CHx_END) is present. DMA transfer is not stopped. If the queue has a next list, execution of the next descriptor follows.
1	1	Interrupt (DMAC Interrupt Status Register (Offset: H'80C), Interrupt Status CHx_END) is present. DMA transfer is stopped (QUE_EN is cleared).

(3) Descriptor Format

The table below is the format of descriptors.

When reading descriptors, this module reads data up to Offset H'00 to H'2C, but data up to Offset H'28 to H'2C is not used in the macro.

Table 34.18 Descriptor List Format

Offset	Byte 3								Byte 2								Byte 1								Byte 0							
H'00	DSCFM = 0001b			—	WBD	LE	LV	D	—	—	—	—	—	—	—	STS (not used)																
H'04	—	—	—	—	—	—	—	—	CCH_L								CCH_D								—	TC			—	—	ATB		
H'08	—	DMA_FUNC							—	—	—	DIR					SIZE [31:0]																
H'0C	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'10	Source Address [31:0]																																
H'14	Source Address [63:32]																																
H'18	Destination Address [31:0]																																
H'1C	Destination Address [63:32]																																
H'20	Link Pointer [31:0]																																
H'24	Link Pointer [63:32]																																

Field Name	Description
DSCFM[3:0]	Descriptor Format, which specifies the format of the descriptor. This DMAC only supports 0001b. Do not set any other values.
WBD	Write Back Disable bit Indicates whether the DMAC writes 0b back to the LV bit when DMA transfer specified by a single descriptor is completed. 1b: The LV bit is not written back. 0b: The LV bit is written back.
LE	List End bit Indicates the end of the current descriptor chain. 1b: The current descriptor is the last of the chain. 0b: The current descriptor is not the last of the chain.
LV	Link Valid bit Indicates that the descriptor is valid. When DMA transfer is completed, the DMAC writes 0b back to this bit. When DMA transfer ends due to an error, write-back does not proceed. 1b: The descriptor is valid (DMA transfer specified by the descriptor is not completed). 0b: The descriptor is not valid (DMA transfer specified by the descriptor is completed).
D	Descriptor error bit Indicates a descriptor access error. If LV = 0 when the descriptor is read, the DMAC writes 1b back to the LV bit. When a descriptor error occurs, if the setting of the LE bit is 0 (indicating that the current descriptor is not the last of the chain), the DMAC continues transfer in accord with the descriptor the LP bits indicate. 1b: A descriptor access error occurred. 0b: No error
STS[15:0]	This field has no effect when DSCFM = 0001b. The DMAC does not use this field.

Field Name	Description
CCH_L[3:0]	This field indicates the value of A*CACHE [3:0] to be issued through the AXI. The value of CCH_L [3:0] is output when an AXI request including the last byte is issued in transfer specified in the SIZE bits. Bit [3]: Write allocation Bit [2]: Read allocation Bit [1]: Cache enabled Bit [0]: Buffer enabled The recommend value is 0000b regardless of whether DIR is 0 or 1.
CCH_D[3:0]	This field indicates the value of A*CACHE [3:0] to be issued through the AXI. The value of CCH_D[3:0] is output when an AXI request other than the condition for the output of CCH_L[3:0] is issued. Bit [3]: Write allocation Bit [2]: Read allocation Bit [1]: Cache enabled Bit [0]: Buffer enabled The recommended value is 0001b when DIR = 0b (PCIe to AXI) and 0000b when DIR = 1b (AXI to PCIe).
TC[2:0]	This field specifies the value of the traffic class of the request to be issued through the PCIe interface. The value must be fixed to 000b.
ATB[1:0]	This field indicates the value of the attribute to be issued through the PCIe interface. Bit [1]: Relaxed ordering Bit [0]: No-snoop If neither relaxed ordering nor no-snoop is used, this field should be set to 00b (recommended).
DMA_FUNC[2:0]	This field specifies the function number of the request to be issued through the PCIe interface.
DIR	This bit indicates the direction of data transfer. 1b: AXI to PCIe 0b: PCIe to AXI
SIZE[31:0]	This field indicates the number of bytes for transfer. Since the setting is 16-byte alignment, do not set anything other than 0000b for the lower 4 bits ([3:0]).
SA[63:0]	The Source Address field indicates the address of the source data for transfer. Since the setting is 16-byte alignment, do not set anything other than 0000b for the lower 4 bits ([3:0]).
DA[63:0]	The Destination Address field indicates the destination address. Since the setting is 16-byte alignment, do not set anything other than 0000b for the lower 4 bits ([3:0]).
LP[63:0]	This field indicates the address where a next descriptor is stored. Since 16-byte alignment and 1 descriptor (H'00 to H'24) beyond the 4K boundary is prohibited, set the lower 6 bits ([5:0]) to be 00_0000b.

The conditions where the descriptor is written back after the descriptor is read, the timings, and the corresponding bits are listed below.

Condition	Timing	Corresponding Bit and Value
WBD = 0 && LV = 1	After DMA transfer specified by the descriptor is completed	LV = 0
WBD = (1 or 0) && LV = 0	After the descriptor is read (DMA transfer does not proceed)	D = 1
WBD = 1 && LV = 1	Write-back does not proceed.	—

(4) Flow of Operations

The following describes the procedures for settings to activate and stop the DMAC in the case of descriptor-type transfer.

(a) Activation and Normal Operation

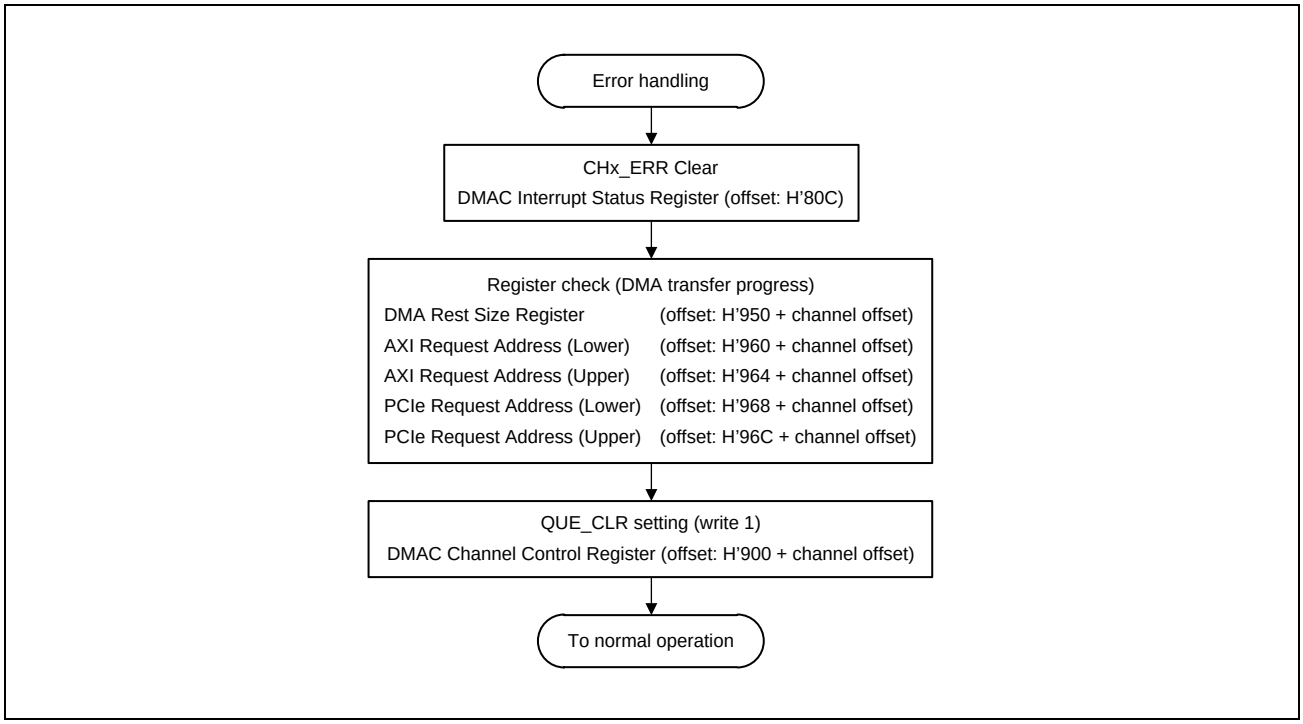
Before activating the DMAC, set the PCIe and AXI windows. After that, set the registers of the DMAC to start the DMAC.

(b) Error Processing

If an error is detected in the AXI bus or a PCIe link while DMA transfer is in progress, the DMA_INT interrupt signal is asserted. Check the channel where the error was found (CHx_ERR = 1b) by using the DMAC interrupt status register and also the type of error by using the DMAC error status register (offset: H'978 + channel offset) as required.

Clear the CHx_ERR bit that is currently set in the DMAC interrupt status register (offset: H'80C) to 0b by writing to it. The address to which transfer was most recently completed before it was stopped and the remaining number of bytes for transfer can be checked by reading the DMA status registers (offset: H'950 + channel offset to H'96C + channel offset) for the given channel.

Based on that information, setting the registers as described in **Section 34.4.4.2(4)(a), Activation and Normal Operation** and create a descriptor table again. At this time, write 1b to the QUE_CLR bit (bit [8]) of the DMAC channel control register (offset: H'900 + channel offset). Writing 1b to it leads to clearing of the queue. The descriptor lists registered in the queue (those waiting for execution and the one currently being executed) are all cleared.



34.4.4.3 Method of Transfer

This section describes the issuing of requests to the AXI or PCIe and data transfer in the case of DMA transfer. In either direction of transfer (from AXI to PCIe or vice versa), a request is issued to the reading side. After confirming the completion of preparation (reception) of the data to be read, a request is issued to the writing side for data transfer.

(1) DMA Transfer from PCIe to AXI

An MRd is issued to the PCIe module and an AXI write request is transferred following the reception of CplD from the PCIe. MRd requests corresponding to the number of outstanding transfers can be issued regardless of the reception of CplD. The received CplD (read data) is stored in the data buffer (RAM) once and then transferred by the DMAC to the AXI.

(2) DMA Transfer from AXI to PCIe

After the DMAC is activated, a read request (address channel) is issued to the AXI. The reception of data for reading from the given AXI slave starts and an MWr is issued to the PCIe. Data are repeatedly transferred until the completion of DMA data transfer corresponding to the amount of data set in the DMA Size bits.

34.4.4.4 Inter-Channel Arbitration

The DMAC arbitrates the following three types of requests between channels.

1) PCIe Read Request
2) AXI Read Request
3) AXI Write Request

PCIe write requests do not require arbitration because data read over the AXI are returned one by one.

In the case of arbitration on the AXI side, not only requests for DMA data, but also reading and writing of descriptor data are subject to arbitration (each channel has a single source of requests for arbitration since a channel will not output a descriptor and request for data at the same time).

(1) Method of Arbitration

An MRd is issued to the PCIe module and an AXI write request is transferred following the reception of CplD from the PCIe. MRd requests corresponding to the number of outstanding transfers (eight) can be issued regardless of the reception of CplD. The received CplD (read data) is stored in the data buffer (RAM) once and then transferred by the DMAC to the AXI.

Arbitration for each type of request 1) to 3) above proceeds in round-robin fashion. A channel for which execution has just been completed is given the lowest priority and each channel which had a lower priority than that channel is raised by one rank in the order of priority.

The initial order of priority is ch0 > ch1 > ch2 > ch3 > ch4 > ch5 > ch6 > ch7.

The order of priority is changed every time a single request is completed.

There are two types of request for the AXI, requests for DMA data and requests for descriptors, but they are not distinguished for purposes of arbitration.

34.4.4.5 DMA Completed Interrupt

The DMAC has interrupt functions. It supports two types of interrupt source: one set is for normal operation and the other is for errors. For details, see **Section 34.4.6.4, DMA Interrupt**.

34.4.5 Reception of PCIe Commands

34.4.5.1 Reception of MSIs (Root Complex)

When an MSI is received (see Judgment conditions), it is forwarded to AXI as a Write Transaction and at the same time, it is reflected in the MSI reception status (see Status). MSI receive Asserts the interrupt signal (INTMSI_RC) when the interrupt generation condition (see Interrupt conditions) is satisfied. S/W identifies MSI by interrupt factor. When Message Data Enable = 0, it is possible to determine the factors by reading the data in memory.

Related Registers

- MSI Receive Window Address (Lower) Registers (Offset: H’100) Message Data Enable
- PCI INTx Receive Interrupt Enable Registers (Offset: H’110) MSI Receive Interrupt Enable
- PCI INTx Receive Interrupt Status Registers (Offset: H’114) MSI Receive Interrupt Status
- MSI Receive Mast n Registers (Offset: H’6x8) MSI Mask
- MSI receive Status n Registers (Offset: H’6xC) MSI Status

Judgment conditions

When all of the following conditions are satisfied, a request is judged to be an MSI.

- The request is a memory write request.
- The request is entered for any area within an AXI window.
- The address of an MWr from the PCIe interface must be within the area set by the MSI Reception Window Address register and the MSI Reception Window Mask register. If a memory read request is received in the MSI reception area, the request is not judged to be an MSI so an interrupt is not generated.
- The length of memory write requests is 1 dword.
- When Message Data matches (only when Message Data Enable is 1)

Status

- When Message Data Enable is 0, MSI Receive Interrupt Status (bit[4]) is set.
- When Message Data Enable is 1, the corresponding vector bit of MSI Status is set.

Interrupt conditions

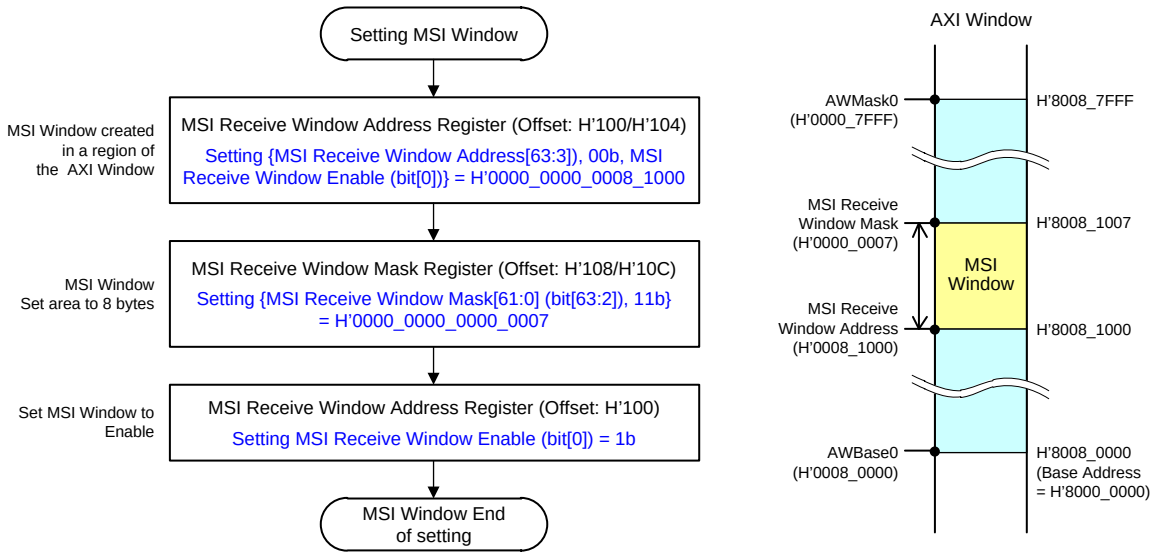
- When Message Data Enable is 0, when MSI Receive Interrupt Enable (bit[4]) is set
- When Message Data Enable is 1, when the corresponding vector bit of MSI Mask is 0.

Notes on the MSI (When Message Data Enable is 0)

- When a response to the MSI write transaction is returned, this module judges execution of the MSI to have been completed and asserts an interrupt signal. Depending on the system, however, the MSI memory write transaction may not be completed due to the latency over the AXI to the actual memory even if the interrupt signal is asserted.
  To avoid this problem, buffering of AXI write transactions must be disabled (MAWCACHE[0] = 0b) at the time an MSI is issued.
  For MAWCACHE[3:1], the setting of AWCACHE_L (bits [27:24]) of Mode Set 0 Register is used.
- If an MSI is received, the MSI write transaction is executed when MAWID[3:0] = 0001b. This is required for the module to recognize the reception of the response to the MSI and for assertion of INTMSI_RC.

34.4.5.2 Setting the MSI Window

To enable MSI interrupts, the MSI window must be set. **Figure 34.4, Settings for the MSI Window** shows example settings. The MSI window can be allocated within any AXI window.



Note: The values assigned in the figure (values in blue) are examples

Figure 34.4 Settings for the MSI Window

34.4.5.3 Reception of an Interrupt in Response to a Message Request (Root Complex)

When Assert_INTx is received in response to an Msg request from the PCIe interface, the corresponding interrupt bit (bits [3] to [0]) is set in the PCI INTx Receive Interrupt Status Register (offset: H'114) and an interrupt (INTx_RC) is asserted. When Deassert_INTx is received in response to a message request, the corresponding interrupt register (bit) is cleared and INTx_RC is de-asserted. The PCI INTx Receive Interrupt Status Register (offset: H'114) can be cleared by writing to it (RW1C) by software. In the PCIe, however, we do not recommend using software to clear an interrupt bit which has been set in response to an Msg in normal operation.

34.4.5.4 Reception of Message Requests

When a message is received through the PCIe interface, the relevant information is stored in the following registers.

Table 34.19 Message Related Registers

	Related Register
Message Code/Routing	Message Code Registers (Offset: H'130)
Message Data	Message Data Registers (Offset: H'134)
Message 3rd Header	Message Header 3rdDW Registers (Offset: H'138)
Message 4th Header	Message Header 4thDW Registers (Offset: H'13C)

The corresponding bit in the Message Receive Interrupt Status Register is set at the same time as the reception of the following messages, and this is indicated by an interrupt.

Received Message		Corresponding Bit in Message Receive Interrupt Status Register
Type of Message	Message Code	
PM_Active_State_Nak	0001_0100	Bit [19]
PM_PME	0001_1000	Bit [18]
PME_Turn_Off	0001_1001	Bit [17]
PME_TO_Ack	0001_1011	Bit [16]

In the reception of PM messages other than the above, the relevant information is not written to the registers listed in **Table 34.19, Message Related Registers**.

- Unsupported messages:
  - Unlock
  - Vendor Defined Type 0
  - Vendor Defined Type 1

34.4.6 Interrupt

This module prepares the output of interrupt signals in the ways stated below. The states of all interrupt signals are indicated at a glance in the Interrupt Table Register.

34.4.6.1 Error and Event Interrupt Notification

Table 34.20 Error/Event Interrupt Outputs

Signal Name	Active	Attribute	Description
DMA_INT	High	Level	DMA-related events
PCIE_EVT_INT	High	Level	PCIe related events
MSG_INT	High	Level	Reception of messages
AXI_ERR_INT	High	Level	AXI-related errors

Interrupt Output Related Status Registers

DMA_INT: DMAC Interrupt Status Registers (Offset: H’80C)
DMAC Error Status Registers (Offset: H’968/+Channel Offset)

PCIE_EVT_INT: PCIe Event Interrupt Status 0 Registers (Offset: H’204)
PCIe Event Interrupt Status 1 Registers (Offset: H’20C)
PCIe Event Interrupt Status 2 Registers (Offset: H’244)

MSG_INT: Message Receive Interrupt Status Registers (Offset: H’124)

AXI_ERR_INT: AXI Master Error Interrupt Status Registers (Offset: H:214)
AXI Slave Error Interrupt Status 1 Registers (Offset: H’224)

34.4.6.2 INTx/MSI Interrupt Notification (Root Complex)

Table 34.21 INTx/MSI Interrupt Outputs

Signal Name	Active	Attribute	Description
INTA_RC	High	Level	Set in response to the reception of an assert INTA message and cleared in response to the reception of a deassert INTA message.
INTB_RC	High	Level	Set in response to the reception of an assert INTB message and cleared in response to the reception of a deassert INTB message.
INTC_RC	High	Level	Set in response to the reception of an assert INTC message and cleared in response to the reception of a deassert INTC message.
INTD_RC	High	Level	Set in response to the reception of an assert INTD message and cleared in response to the reception of a deassert INTD message.
INTMSI_RC	High	Level	Set when a memory write request comes from the PCIe to the area set in the MSI reception window.
INT_LINK_BANDWIDTH	High	Level	Set when the link width has been changed.
INT_PM_PME	High	Level	Set in response to the reception of PME event notification (PM_PME message).
INT_SERR_COR/ INT_SERR_NONFATAL/ INT_SERR_FATAL	High	Level	Set in response to the reception of a correctable error message, non-fatal error message, or fatal error message.

34.4.6.3 Root Complex Interrupt Notification

This module has the following interrupt sources.

- INT_LINK_BAND_WIDTH
- INT_PM_PME
- INT_SERR
- INT_SERR_COR
- INT_SERR_FATAL
- INT_SERR_NONFATAL

(1) Interrupt due to the Change to the Link Bandwidth (INT_LINK_BAND_WIDTH)

This interrupt source is conveyed in response to the change to the bandwidth in link negotiation of PCI Express.

This interrupt source is only enabled when the Link Bandwidth Notification Capability bit (bit 21) of the Link Capabilities register (offset: H’60C) is set to 1b. This interrupt signal can also be disabled by setting the Link Bandwidth Notification Capability bit to 0b even in operation as the root complex.

If the Link Bandwidth Notification Capability bit (bit 21) is set to 1b, the Link Bandwidth Management Status bit (bit 30) and the Link Autonomous Bandwidth Status bit (bit 31) in the Link Status register (offset H’70) serve as a source of the interrupt. These bits are set to 1b when the bandwidth is changed.

Whether to enable this interrupt source corresponds to the respective states of the Link Bandwidth Management Interrupt Enable bit (bit 10) and the Link Autonomous Bandwidth Interrupt Enable (bit 11) of the Link Control register (offset: H’70).

The following is the Related Registers.

Offset	Bit	Description
PCI Express Capability Structure: Link Control / Status		
H’06C	21	Link Bandwidth Notification Capability
H’070	10	Link Bandwidth Management Interrupt Enable
	11	Link Autonomous Bandwidth Interrupt Enable
	30	Link Bandwidth Management Status
	31	Link Autonomous Bandwidth Status

(2) PM-PME Reception Interrupt (INT_PM_PME)

This interrupt source is conveyed when the notification of a PME event (PM_PME message) is received from the other party device.

Note that this interrupt source is enabled when the root complex in use and it is fixed to “0: Low” when the endpoint is in use.

This interrupt source is only conveyed when the PME Interrupt Enable bit (bit 3) of the Root Control/Capabilities Register (offset: H’7C) is set to 1b. This interrupt source is not conveyed when the PME Interrupt Enable bit (bit 3) is set to 0b.

Note that the reception state of the received PM_PME message and the ID information, etc. are stored in the Root Status Register (offset: H’80) regardless of the setting of the PME Interrupt Enable bit mentioned above.

Offset	Bit	Description
PCI Express Capability Structure: Root Control Capabilities		
H’07C	3	PME Interrupt Enable
PCI Express Capability Structure: Root Status		
H’080	15:0	PME Requester ID
	16	PME Status
	17	PME Pending

(3) System Error Interrupt (INT_SERR_xxx)

This interrupt is conveyed when a correctable error message, non-fatal error message, or fatal error message is received.

- INT_SERR: The system error notification mentioned in the Base Spec. is obtained as the logical OR of the following three signals.
- INT_SERR_COR: System Error on Correctable Error
- INT_SERR_FATAL: System Error on Fatal Error
- INT_SERR_NONFATAL: System Error on Non-Fatal Error

(a) Correctable Error Interrupt (INT_SERR_COR)

To proceed with interrupt control due to a correctable error, set the interrupt enable bits of the following registers to 1b.

No.	Offset	Bit	Description
1	PCI Compatible Configuration: Bridge Control and Interrupt		
	H'03C	17	SERR# Enable
2	PCI Express Capability Structure: Device Control		
	H'068	0	Correctable Error Reporting Enable
3	Advanced Error Reporting (AER) Capability: Root Error Command		
	H'12C	0	Correctable Error Reporting Enable

Interrupt Source

(1) ERR_COR Message Reception

If an ERR_COR message is received from the other-party device (endpoint), the INT_SERR_COR interrupt will be asserted. The following status register will be set at the same time.

Offset	Bit	Description
Advanced Error Reporting (AER) Capability: Root Error Status		
H'130	0	ERR_COR Received

Note: If the Enable bits in no. 1, 2, and 3 are not set, the interrupt source will not be generated.
If the Enable bits in no. 1 and 2 are not set, writing will be masked.

(2) Correctable Error Detection

Any of the following correctable errors being detected within the unit will be written to the corresponding status register.

Error	Offset	Bit	Description
	Advanced Error Reporting (AER) Capability: Correctable Error Status		
8B10B Decode Error 8B10B RD Error 1	H'110	0	Receiver Error Status
Bad TLP		6	Bad TLP Status
Bad DLLP		7	Bad DLLP Status
REPLAY_NUM Roll over		8	REPLAY_NUM Rollover Status
Replay Timer Timeout		12	Replay Timer Timeout Status
Advisory Non-Fatal Error		13	Advisory Non-Fatal Error Status

Note: If the Mask bit (Correctable Error Mask Register: Offset H'114) corresponding to each bit is set (masked), the corresponding error will not be detected, and the interrupt source will not be generated.

At the same time, the INT_SERR_COR interrupt will be generated, and the following status registers will be set to 1b.

Offset	Bit	Description
PCI Express Capability Structure: Device Status		
H'068	16	Correctable Error Detect
Advanced Error Reporting (AER) Capability : Root Error Status		
H'130	0	ERR_COR Received

Note: If the Enable bit in no. 2 is not set, the interrupt source will not be generated.
If the Enable bit in no. 2 is not set, writing to the Root Error Status Register will be masked.
If the Correctable Error Reporting Enable bit listed in no. 3 is not set, the source due to (1) and (2) stated above will be generated and INT_SERR_COR will not be asserted even if writing to the Root Error Register has proceeded.

(b) Fatal Error Interrupt (INT_SERR_FATAL)

To proceed with interrupt control due to a fatal error, set the interrupt enable bits of the following registers to 1b.

No.	Offset	Bit	Description
1	PCI Compatible Configuration: Bridge Control and Interrupt		
	H'03C	17	SERR# Enable
2	PCI Express Capability Structure: Device Control		
	H'068	2	Fatal Error Reporting Enable
3	Common Configuration Space: Command and Status		
	H'004	8	SERR# Enable
4	Advanced Error Reporting (AER) Capability: Root Error Command		
	H'12C	2	Fatal Error Reporting Enable

Interrupt Source

(1) ERR_FATAL Message Reception

If an ERR_FATAL message is received from the other-party device (endpoint), the INT_SERR_FATAL interrupt will be asserted (if the Enable bits in no.1 and 2 or no.3 and 4 are not set, the interrupt source will not be generated).

The following status registers will be set at the same time.

Offset	Bit	Description	Remarks
Type1 Configuration Space: Secondary Status			*1
H'01C	30	Received System Error	
Common Configuration Space: Status Register			*2
H'004	30	Signaled System Error	
Advanced Error Reporting (AER) Capability: Root Error Status			*3
H'130	6	Fatal Error Message Received	

Note 1. Writing to this status register proceeds at the point a message is received regardless of the settings of the Enable bits.

Note 2. If the Enable bits in no. 1 and 3 are not set, writing will be masked.

Note 3. If the Enable bit in no. 1, 2, or 3 is not set, writing will be masked.

(2) Fatal Error Detection

Any of the following fatal errors being detected within the module will be written to the corresponding status register.

Error	Offset	Bit	Description
	Advanced Error Reporting (AER) Capability: Uncorrectable Error Status		
Data Link Protocol Error	H'104	4	Data Link Protocol Error Status
Poisoned TLP		12	Poisoned TLP Status
Completion Timeout		14	Completion Timeout Status
Completer Abort		15	Completer Abort Status
Unexpected Completion		16	Unexpected Completion Status
Receiver Overflow		17	Receiver Overflow Status
Malformed TLP		18	Malformed TLP Status
ECRC Error		19	ECRC Error Status
Unsupported Request		20	Unsupported Request Error Status

Note: If the Mask bit (Uncorrectable Error Mask Register: Offset H'108) corresponding to each bit is set (masked), the corresponding error will not be detected and the interrupt source will not be generated.
Also, the setting of the corresponding Severity bit (Uncorrectable Error Severity: Offset H'10C) must be 1b (Fatal).

The INT_SERR_FATAL interrupt will be generated at the same time (if the Enable bits in no. 2 or no.3 and 4 are not set, the interrupt source will not be generated). The following status registers will be set at the same time.

Offset	Bit	Description	Remarks
PCI Express Capability Structure: Device Status			
H'068	18	Fatal Error Detect	
Common Configuration Space: Status Register			*1
H'004	30	Signaled System Error	
Advanced Error Reporting (AER) Capability: Root Error Status			*2
H'130	6	Fatal Error Message Received	

Note 1. If the Enable bit in no. 3 is not set, writing will be masked.

Note 2. If the Enable bit in no. 2 or 3 is not set, writing will be masked.

If the Fatal Error Reporting Enable bit listed in no. 4 is not set, the source due to (1) and (2) stated above will be generated and INT_SERR_FATAL will not be generated even if writing to the Root Error Status Register has proceeded.

(c) Non-Fatal Error Interrupt (INT_SERR_NONFATAL)

To proceed with interrupt control due to a non-fatal error, set the interrupt enable bits of the following registers to 1b.

No.	Offset	Bit	Description
1	PCI Compatible Configuration: Bridge Control and Interrupt		
	H'03C	17	SERR# Enable
2	PCI Express Capability Structure: Device Control		
	H'068	1	Non-Fatal Error Reporting Enable
3	Common Configuration Space: Command and Status		
	H'004	8	SERR# Enable
4	Advanced Error Reporting (AER) Capability: Root Error Command		
	H'12C	1	Non-Fatal Error Reporting Enable

Interrupt Source

(1) ERR_NONFATAL Message Reception

If an ERR_NONFATAL message is received from the other-party device (endpoint), the INT_SERR_NONFATAL interrupt will be generated (if the Enable bits in no.1 and 2 or no.3 and 4 are not set, the interrupt source will not be generated). The following status registers will be set at the same time.

Offset	Bit	Description	Remarks
Type1 Configuration Space: Secondary Status			*1
H'01C	30	Received System Error	
Common Configuration Space: Status Register			*2
H'004	30	Signaled System Error	
Advanced Error Reporting (AER) Capability: Root Error Status			*3
H'130	5	Non-Fatal Error Message Received	

Note 1. Writing to this status register proceeds at the point a message is received regardless of the settings of the Enable bits.

Note 2. If the Enable bits in no. 1 and 3 are not set, writing will be masked.

Note 3. If the Enable bit in no. 1, 2, or 3 is not set, writing will be masked.

(2) Non-Fatal Error Detection

Any of the following non-fatal errors being detected within the module will be written to the corresponding status register.

Error	Offset	Bit	Description
	Advanced Error Reporting (AER) Capability: Uncorrectable Error Status		
Data Link Protocol Error	H'104	4	Data Link Protocol Error Status
Poisoned TLP		12	Poisoned TLP Status
Completion Timeout		14	Completion Timeout Status
Completer Abort		15	Completer Abort Status
Unexpected Completion		16	Unexpected Completion Status
Receiver Overflow		17	Receiver Overflow Status
Malformed TLP		18	Malformed TLP Status
ECRC Error		19	ECRC Error Status
Unsupported Request		20	Unsupported Request Error Status

Note: If the Mask bit (Uncorrectable Error Mask Register: Offset H'108) corresponding to each bit is set (masked), the corresponding error will not be detected, and the interrupt source will not be generated.
Also, the setting of the corresponding Severity bit (Uncorrectable Error Severity: Offset H'10C) must be 0b (Non-fatal).

The INT_SERR_NONFATAL interrupt will be generated at the same time (if the Enable bits in no. 2 or no.3 and 4 are not set, the interrupt source will not be generated). The following status registers will be set at the same time.

Offset	Bit	Description	Remarks
PCI Express Capability Structure: Device Status			
H'068	17	Non-Fatal Error Detect	
Common Configuration Space: Status Register			*1
H'004	30	Signaled System Error	
Advanced Error Reporting (AER) Capability: Root Error Status			*2
H'130	5	Non-Fatal Error Message Received	

Note 1. If the Enable bit in no. 3 is not set, writing will be masked.

Note 2. If the Enable bit in no. 2 or 3 is not set, writing will be masked.

If the Non-Fatal Error Reporting Enable bit listed in no. 4 is not set, the source due to (1) and (2) stated above will be generated and INT_SERR_NONFATAL will not be generated even if writing to the Root Error Status Register has proceeded.

34.4.6.4 DMA Interrupt

The DMAC has interrupt functions. It supports two types of interrupt source: one set is for normal operation and the other is for errors. Use the DMAC Interrupt Enable Register to select the interrupt sources and control masking. DMA interrupt notification includes notification by DMA_INT in the direction of the AXI bus (local bus direction) and notification by issuing an MSI to an other-party RC.

On completion of DMA transfer, either conveying a DMA_INT interrupt or transmitting an MSI to the RC, but not both, is selected per-channel by a register setting (transmitting an MSI is prohibited in RC mode).

The DMA_CH*_MSI_EN bits of the DMA Interrupt Vector 0 Register and the DMA Interrupt Vector 1 Register are used to switch the setting ("*" corresponds to the channel number of a given DMA and per-channel control is possible). If an MSI is to be issued, the value of its traffic class (TC) bits is fixed to 0. The vector value can be specified by using the DMA_CH*_vec bits of these registers.

CAUTION

Take care that the specified vector value does not impose a related restriction on the Multiple Message Enable register.

For example, if the number set for Multiple Message Enable is 4, the two lower-order bits within the value set for the Message Data are overwritten by the value specified for the DMA_CH*_vec bits, and at that time, specify the value 0b for the higher-order bits that are not overwritten by DMA_CH*_vec bits.

34.4.7 Power Management (Root Complex Mode)

34.4.7.1 PCI Power Management (PCI-PM)

An endpoint is placed in non-D0 (the D3 hot state) by changing the setting of the Power State field register of the endpoint by the root complex., which causes a transition of the link state of both RC and EP to L1.

This module does not support the AUX power supply function. D3 cold is not supported. Implementing the L2 function goes to implementation of L2/L3 ready.

(1) Flow of Transition to the L2/L3 Ready State (RC)

< For transition via PCIPM L1 >

1. A root complex (RC) issues a config write to change the D-state of an endpoint (EP) to D3 (D3 hot).
2. The RC is automatically placed in L1 by automatic response of the other-party EP.
3. The RC transmits a PME_Turn_Off Message.
4. Confirm the reception of a PME_TO_Ack Message from the EP by reading the value of the PME_TO_Ack Receive Interrupt bit (bit 16) in the Message Receive Interrupt Status Register of the RC.
5. Set PCIe Core Control 1 Register (bit [17]: UI_ENTER_L2 = 1b).
6. Confirm the transition to the L2/L3 ready state by reading the value of the LTSSM_STATE bits (bits [14:8]) in the PCIe Core Status 1 Register (5 higher-order bits [14:10] = 0111xb: L2).

< For direct transition to the L2/L3 ready state >

Steps 3 to 6 above apply.

(2) Flow of Return from the L2/L3 Ready State (RC)

< Return from the L2/L3 ready state by the RC >

1. Assert the following reset signals.
   RST_B, RST_GP_B, RST_PS_B, RST_CFG_B
2. Deassert the reset signals at a desired time and wait for return to L0 (wait for linkup).
3. Clear PCIe Core Control 1 Register (bit [17]: UI_ENTER_L2 = 0b).
4. If the link is not up, repeat steps 1 and 2.

< Return from the L2/L3 ready state by the EP >

1. Wait for the reception of a beacon by the EP. Read the LINKDN bit (H’0204, bit 9) in the PCIe Event Interrupt Status register regularly to check whether ELECTRICAL_IDLE_BROKEN has been generated.
2. Assert the following reset signals.
   RST_B, RST_GP_B, RST_PS_B, RST_CFG_B
3. Deassert the reset signals at a desired time, clear UI_ENTER_L2 of PCIe Core Control 1 Registers to 0b, and wait for it to return to L0 (wait for linkup).
4. Clear PCIe Core Control 1 Register (bit [17]: UI_ENTER_L2 = 0b).
5. If the link is not up, repeat steps 2 and 3.

34.4.7.2 Active State Power Management (ASPM)

The ASPM L0s and L1 states can be used by the setting of the Active State PM Control bits (bits [1:0]) in the Link Control/Status Register (PCIe Configuration Register: H’6070).

ASPM L0s:

The module is automatically placed in the ASPM L0s state following setting of the Active State PM Control bits to 01b or 11b. When to initiate a transition to this state is determined by judgment of the idle state by the PCIe core.

ASPM L1:

An RC is placed in the ASPM L1 state in response a request for transition to the ASPM L1 state from an EP following setting of the Active State PM Control bit to 11b. When to initiate a transition to this state is determined by judgment of the idle state by the PCIe core.

A transition to the ASPM L1 state is enabled by setting the UI_RC_REJECT_ASPM_L1 bit (bit [19]) of the PCIe Core Control 1 Register (offset: H’404) to 0b. To reject a transition to the ASPM L1 state on the RC side, set the UI_RC_REJECT_ASPM_L1 bit to 1b.

Return to the L0 state from the L1 state:

- Resuming access from the AXI master side:

  Return to the L0 state is initiated in response to the other-party PCIe device resuming access.

- Resuming access from the AXI slave side:

  The AXI can be returned to the L0 state in response to the AXI starting access to a PCIe device. Access to an AXI configuration register also returns the AXI to the L0 state.

34.4.8 Power Management (Common)

34.4.8.1 L1 PM Substate

This module supports the L1-Substate transition function. The procedure for transitioning to the L1 state and the basic procedure are the same, but the main difference information is provided below as supplementary information to the procedure for transitioning to each state and the procedure for returning to the L0 state.

(1) PCI-PM “L0 to L1.1”

1. Set the following register bits with register write by RC and CfgWr to instruct L1.1 transition.

   PCI_PM L1.2 Enable = 0b

   PCI_PM L1.1 Enable = 1b

2. Perform procedures as with PCI-PM control.

(2) PCI-PM “L0 to L1.2”

1. Check the register bit value below and change the setting value with register write by RC and CfgWr if necessary.

   T_POWER_ON Scale

   T_POWER_ON Value

2. Set the following register bits with register write by RC and CfgWr to indicate L1.2 transition.

   PCI_PM L1.2 Enable = 1b

3. Perform procedures as with PCI-PM control.

(3) PCI-PM “L1.1/L1.2 to L0”

< Resume from RC side >

1. Assert PCIE_CLKREQ if PCI_CLKREQ# is de-asserted on RC side.
2. The LTSSM state on the RC side transitions to the Recovery state.
3. RC sets POWERSTATE bit to 00b (D0) by CfgWr.

(4) ASPM “L0 to L1.1”

Transition to L1 is allowed by setting the ALLOW_ENTER_L1 bit of the SYS_PCIE_CFG register to 1. Refer to **Section 7.2.4, Register Descriptions**.

Set the ASPM L1 Idle Time field of the Mode Set 3 Registers in advance according to the AXI bus specifications. (Setting range: 256 (ACLK) to 65536 (ACLK))

1. Set the following register bits with register write by RC and CfgWr to instruct L1.1 transition.

   ASPM L1.2 Enable = 0b

   ASPM L1.1 Enable = 1b
2. RC sets the Active State Power Management (ASPM) Control bit to 1b by CfgWr.
3. Set the Active State Power Management (ASPM) Control bit of RC side to 1b.

After the above settings, if transmission and reception are not performed for the specified time, it will automatically transit to the ASPM L1.1 state.

(5) ASPM “L0 to L1.2

Transition to L1 is allowed by setting the ALLOW_ENTER_L1 bit of the SYS_PCIE_CFG register to 1. Refer to **Section 7.2.4, Register Descriptions**.

Set the ASPM L1 Idle Time field of the Mode Set 3 Registers in advance according to the AXI bus specifications. (Setting range: 256 (ACLK) to 65536 (ACLK))

1. Check the register bit value below and change the setting value with register write by RC and CfgWr if necessary.

   LTR_L1.2_THRESHOLD_Value

   LTR_L1.2_THRESHOLD_Scale

   T_POWER_ON Scale

   T_POWER_ON Value
2. Set the following register bits with register write by RC and CfgWr to instruct L1.2transition.

   ASPM L 1.2 Enable = 1b
3. RC sets the Active State Power Management (ASPM) Control bit to 1b by CfgWr.
4. RC sets the LTR Mechanism Enable bit to 1b by CfgWr.
5. Set the Active State Power Management (ASPM) Control bit of RC side to 1b.

After the above settings, if transmission and reception are not performed for the specified time, it will automatically transit to the ASPM L1.2 state.

Remark Transition to L1.2 is required LTR Message transmission. Refer to **Section 34.4.8.2, LTR Function**.

(6) ASPM “L1.1/L1.2 to L0”

If a cause occurs that returns to the ASPM L0 state, it automatically returns to ASPM L0 as appropriate.

<Resume from own>

1. The following controls automatically transition from L1.x to L0.

   Issued CfgWr/Rd, MemWr/Rd to EP

   Write/Read access to the Configuration Registers of RC.

Remark Write/Read access to AXI Bridge Registers with RC does not return to L0.

<Resume form opposite device>

1. When PCI_CLKREQ# is asserted from EP.
2. When MemWr/Rd is received from EP.

34.4.8.2 LTR Function

This module has the following functions related to the LTR function.

1. ASPM L1.2 transition permission judgment function
2. The transition permission judgement function of ASPM L1.2 determines the state transition based on the L1.2 threshold set in CFGU.

34.4.9 Error Processing

Two error reporting paradigms are defined for PCI Express: baseline error reporting capability, which is the minimal requirement, and advanced error reporting (AER) capability, which can provide greater stability, as an optional error reporting facility. This module supports both error reporting functions.

34.4.9.1 Error Classification

PCI Express errors are of two types, correctable and uncorrectable. Uncorrectable errors are further classified into two types, non-fatal and fatal.

Error Type	Description
Correctable Error	An error which cannot be recovered by hardware
Uncorrectable Error (Non-Fatal)	An error which causes a particular transaction in PCI Express to be unreliable (but the PCI Express link itself is functional).
Uncorrectable Error (Fatal)	An error which causes the PCI Express link itself to be unreliable

In operation as a root complex, the above types of error are indicated by the individual interrupt names.

- INT_SERR_COR
- INT_SERR_NONFATAL
- INT_SERR_FATAL

34.4.9.2 Error Checking Mechanism

(1) Physical Layer Error List

The following type of error is to be detected in the physical layer.

- Receiver Error

(2) Data Link Layer Error List

The following type of error is to be detected in the link layer.

- Bad TLP Error
- Bad DLLP Error
- REPLAY_NUM Rollover
- Replay Timer Timeout
- Receiver Overflow Error

(3) Transaction Layer Error List

The following type of error is to be detected in the transaction layer.

- Completion Timeout
- Completer Abort
- Unsupported Request
- Unexpected Completion
- Malformed TLP
- Poisoned TLP

NOTE

The specifications of the receiver error detection function in each LTSSM state are as listed below.

LTSSM State	Module Specifications	Base Spec Specifications
Configuration	Support	Must
Recovery	Non-Support	Option
L0	Support	Must
Disabled	Support	Option
HotReset	Support	Option

34.4.9.3 Error Message

The PCI Express Base Specification defines error messages as one of the mechanisms for notifying the system or another device of an error when it is detected by a PCI Express agent.

Error Message	Description
ERR_COR	Used when Correctable Error is detected
ERR_NONFATAL	Used when a non-fatal, uncorrectable error is detected ERR_FATAL
ERR_FATAL	Used when a fatal, uncorrectable error is detected

(1) Advisory Non-Fatal Error

The PCI Express Base Specification states that when a PCI Express agent as the detecting agent detects a non-fatal error, it handles the error as an advisory non-fatal error whether it does or does not support AER. In handling a non-fatal error as an advisory non-fatal error, the agent sends an ERR_COR message instead of an ERR_NONFATAL message and sends an advisory notification to the software. At this time, the advisory non-fatal error status bit of the correctable error status register is set to indicate the error state. Note that subsequent setting of the first error pointer register, logging of the header, and message transmission only proceed if the Advisory Non-Fatal Error Mask bit of the Correctable Error Mask Register is clear (no masking). They do not proceed if the bit is set.

The error cases which are handled as advisory non-fatal errors are as follows.

- Reception of unsupported Non-Posted Requests
- Reception of a Non-Posted Request with a completer abort completion response
- Reception of an unexpected completion
- Reception of a poisoned TLP (this is not handled as an advisory non-fatal error in this core)
- Detection of a completion timeout (this is not handled as an advisory non-fatal error in this core)

This module may merge multiple error messages with the same ID. This would occur when multiple errors are detected during the wait for the module to be ready for the transmission of messages after an error condition is detected. However, messages will not always be merged in such cases.

34.5 Operation

34.5.1 Setting Up

Set the internal registers of the module including the configuration registers.

This section explains the procedure for setting up until the PCI Express link is up (the module is ready for data transfer).

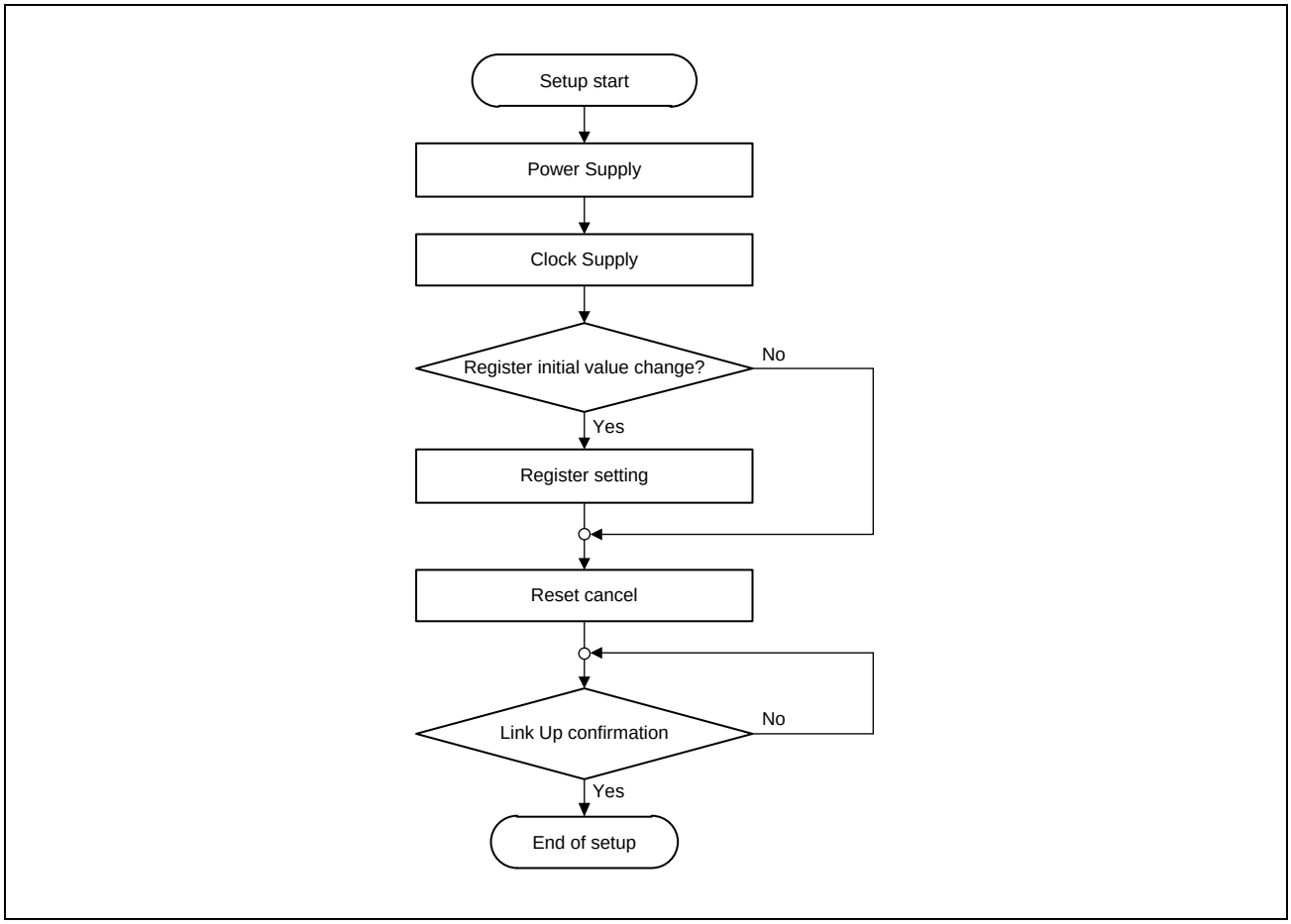


Figure 34.5 Setting Up

34.5.1.1 Changing the Initial Values of the Registers

Of the internal registers, the initial values of the configuration registers can be changed via the AXI slave interface.

AXI Bridge Registers

De-asserting ARESETn following supply of CLK allows access to these registers. In the case of resetting of the PCIe core, write to the registers while the reset signal is being asserted.

PCIe Configuration Register

De-asserting ARESETn following supply of CLK allows access to these registers. Of the reset signals of the PCIe core, those related to the configuration register must be de-asserted.

(1) Setting the Initial Values of the Registers

The initial values of the registers listed below are 0. Set appropriate values in the registers before the start of link up.

- Device ID
- Vendor ID
- Class Code (base class/sub-class/programming interface)
- Revision ID
- Subsystem ID
- Subsystem Vendor ID

34.5.1.2 De-asserting the Reset

De-asserting of the reset of the core automatically starts operations such as receiver detection and the training sequence with the other-party device.

Check that link negotiation with the other party has been completed and the link with the module is up.

Refer to **Section 34.1.4.1, Reset Sequence for Power-On Reset**.

Note that this module requires a wait of at least 5[ms] from power-on to de-assertion of the reset.

34.5.1.3 Checking if the Link is Up

Checking if the link is up can be done through either of the following methods. This is usually done by the root complex.

(1) Polling

Have the CPU of the chip poll the DL_Down Status bit (bit [0]) of the PCIe core Status 1 Register in AXI bridge registers until the value of the bit is 0 (indicating the DL_Up state).

(2) Interrupt

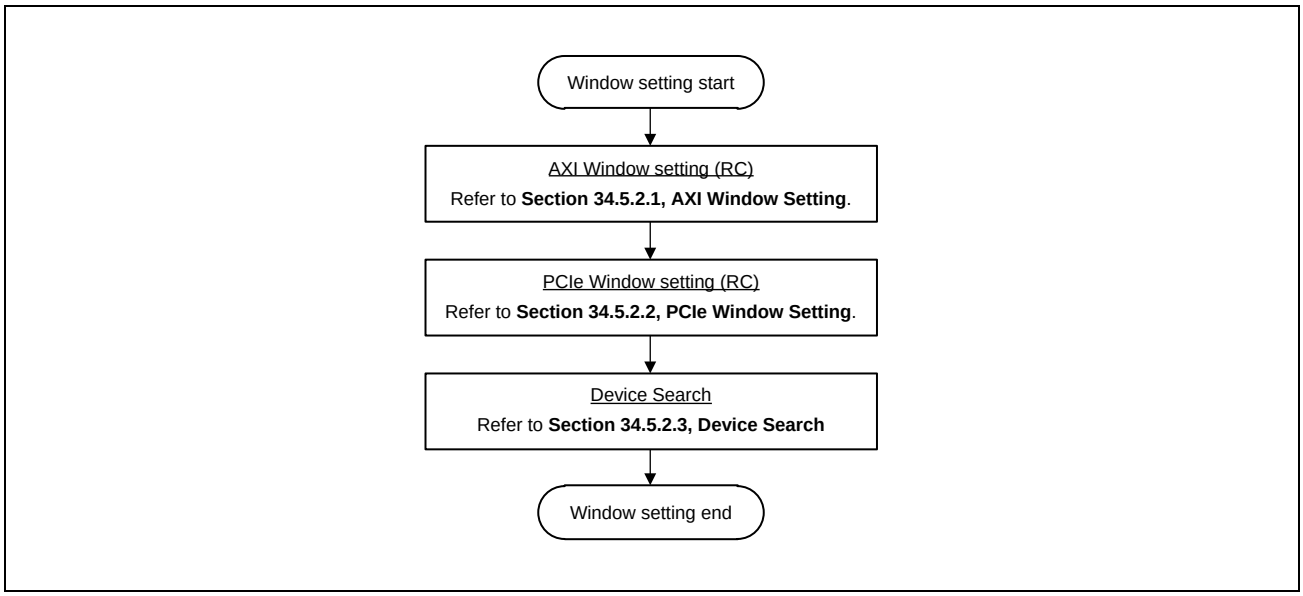
After assertion of the PCIE_EVT_INT pin of the interrupt signal, check the interrupt source by reading the DL_UpDown bit (bit [9]) in the PCIe Event Interrupt Status 0 register in AXI Bridge Registers. After that, check that the DL_Down Status bit (bit [0]) of the PCIe Core Status 1 Register is 0 (indicating the DL_Up state)

The settings above make this module ready for transfer with the other party device.

However, data transfer such as reading or writing memory is not yet possible in this state. This requires subsequent window settings.

34.5.2 Setting the Windows (Root Complex Mode)

Set Window to Root Complex. All PCI Express commands are issued through this memory or register space. After configuring Window settings and Configuration Register, data such as PCIe MRd/MWr can be sent and received.



When using the window settings in the root complex configuration, the recommended settings in the BAR and BAR Mask registers are “BAR: All 0s” and “BAR Mask: All 1s”.

Table 34.22 Example BAR Settings (64 Bits, Function = 1)

Function	BAR (Memory Space)	AXI Window	Note
Function #0	64-bit Memory space {BAR1, BAR0}	AXI #0	
		AXI #1	
		AXI #2	
		AXI #3	
		AXI #4	
		AXI #5	
		AXI #6	
		AXI #7	

34.5.2.1 AXI Window Setting

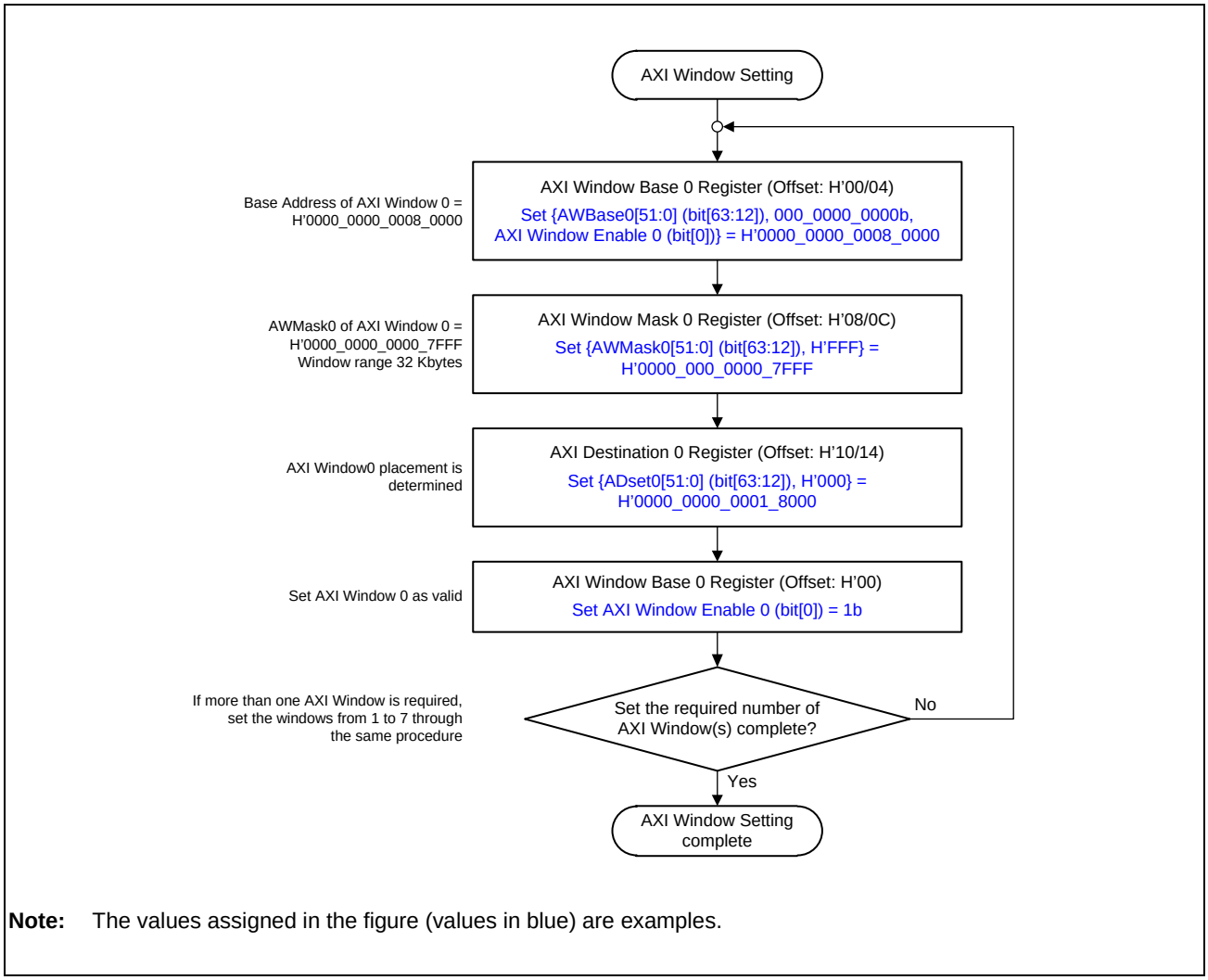
AXI Window is used to transfer MWR/MRd data from PCIe (RX side) to AXI bus.

It can be set by write access to the following registers from the AXI Slave I/F.

Up to 8 windows can be set. The following registers need to be set for the required number of windows.

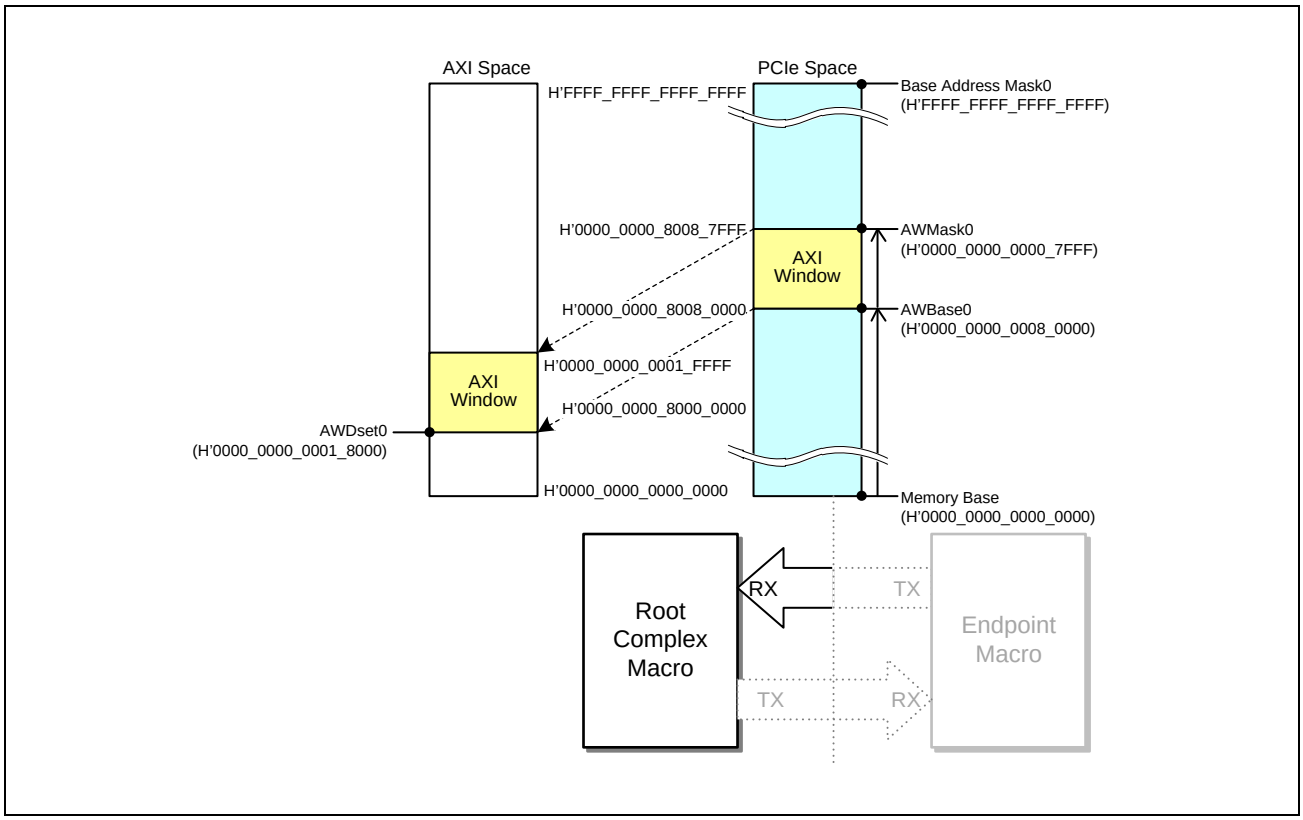
AXI Window Base x Lower Register	(PCI_AWBasexL)	(Offset: H'00/20/40/60/80/A0/C0/E0)
AXI Window Base x Upper Register	(PCI_AWBasexU)	(Offset: H'04/24/44/64/84/A4/C4/E4)
AXI Window Mask x Lower Register	(PCI_AWMaskxL)	(Offset: H'08/28/48/68/88/A8/C8/E8)
AXI Window Mask x Upper Register	(PCI_AWMaskxU)	(Offset: H'0C/2C/4C/6C/8C/AC/CC/EC)
AXI Destination x Lower Register	(PCI_ADestxL)	(Offset: H'10/30/50/70/90/B0/D0/F0)
AXI Destination x Upper Register	(PCI_ADestxU)	(Offset: H'14/34/54/74/94/B4/D4/F4)

(x = 0 to 7)



Note: The values assigned in the figure (values in blue) are examples.

Figure 34.6 Example Settings of the AXI Window


AXI Space
PCIe Space
H'FFFF_FFFF_FFFF_FFFF
Base Address Mask0
(H'FFFF_FFFF_FFFF_FFFF)
H'0000_0000_8008_7FFF
AWMask0
(H'0000_0000_0000_7FFF)
AXI Window
H'0000_0000_8008_0000
AWBase0
(H'0000_0000_0008_0000)
H'0000_0000_0001_FFFF
AXI Window
H'0000_0000_8000_0000
AWDset0
(H'0000_0000_0001_8000)
H'0000_0000_0000_0000
Memory Base
(H'0000_0000_0000_0000)
RX
TX
Root Complex Macro
Endpoint Macro
TX
RX

34.5.2.2 PCIe Window Setting

PCIe Window is used to transfer data from the AXI bus to PCIe (TX side) as MWr/MRd.

It can be set by write access to the following registers from the AXI Slave I/F.

Up to 8 windows can be set. The following register settings are required for the required number of windows.

PCIe Window Base x Lower Register (PCI_PWBasexL) (Offset: H'100/120/140/160/180/1A0/1C0/1E0)

PCIe Window Base x Upper Register (PCI_PWBasexU) (Offset: H'104/124/144/164/184/1A4/1C4/1E4)

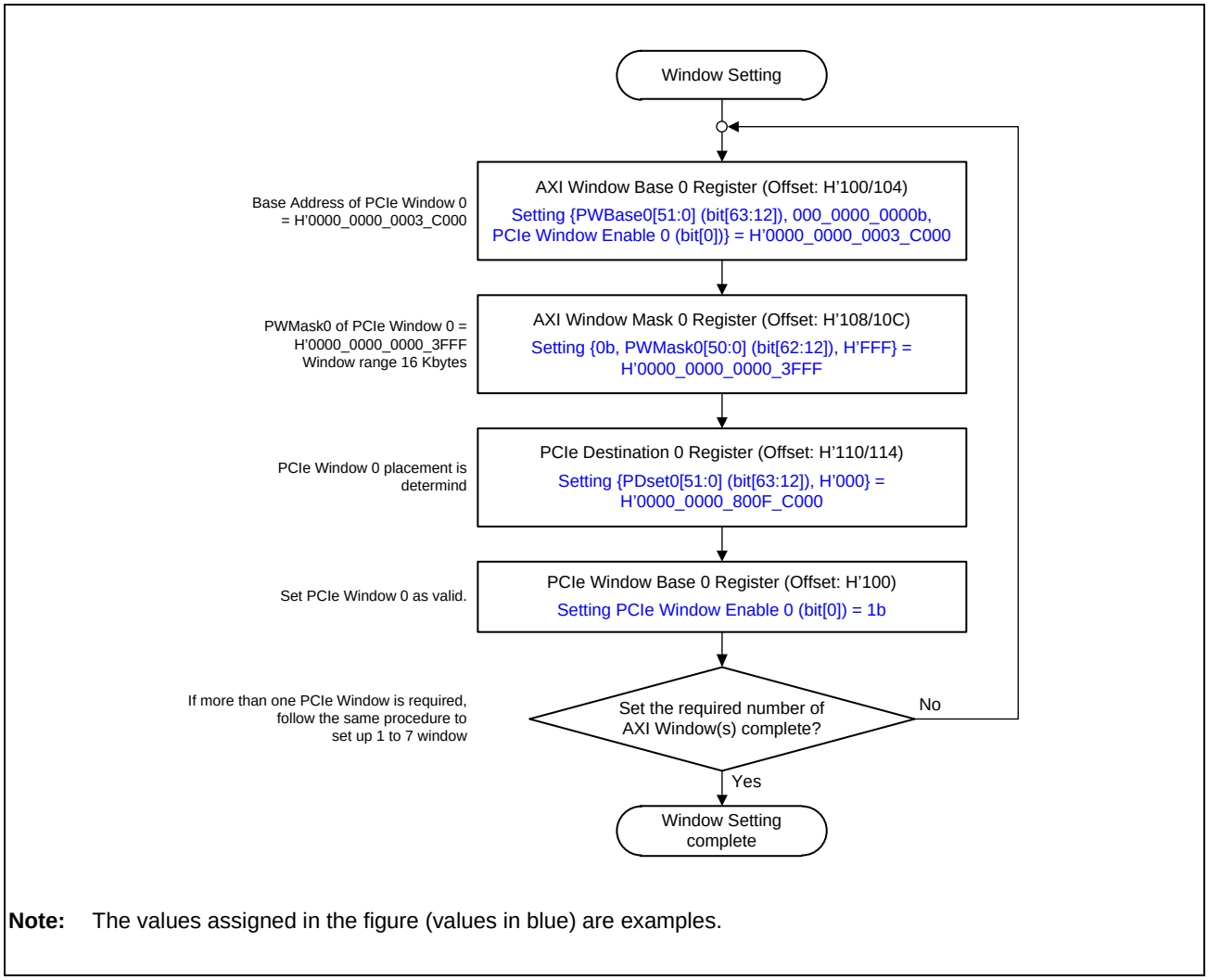
PCIe Window Mask x Lower Register (PCI_PWMaskxL) (Offset: H'108/128/148/168/188/1A8/1C8/1E8)

PCIe Window Mask x Upper Register (PCI_PWMaskxU) (Offset: H'10C/12C/14C/16C/18C/1AC/1CC/1EC)

PCIe Destination x Register (Lower) (PCI_PDestxL) (Offset: H'110/130/150/170/190/1B0/1D0/1F0)

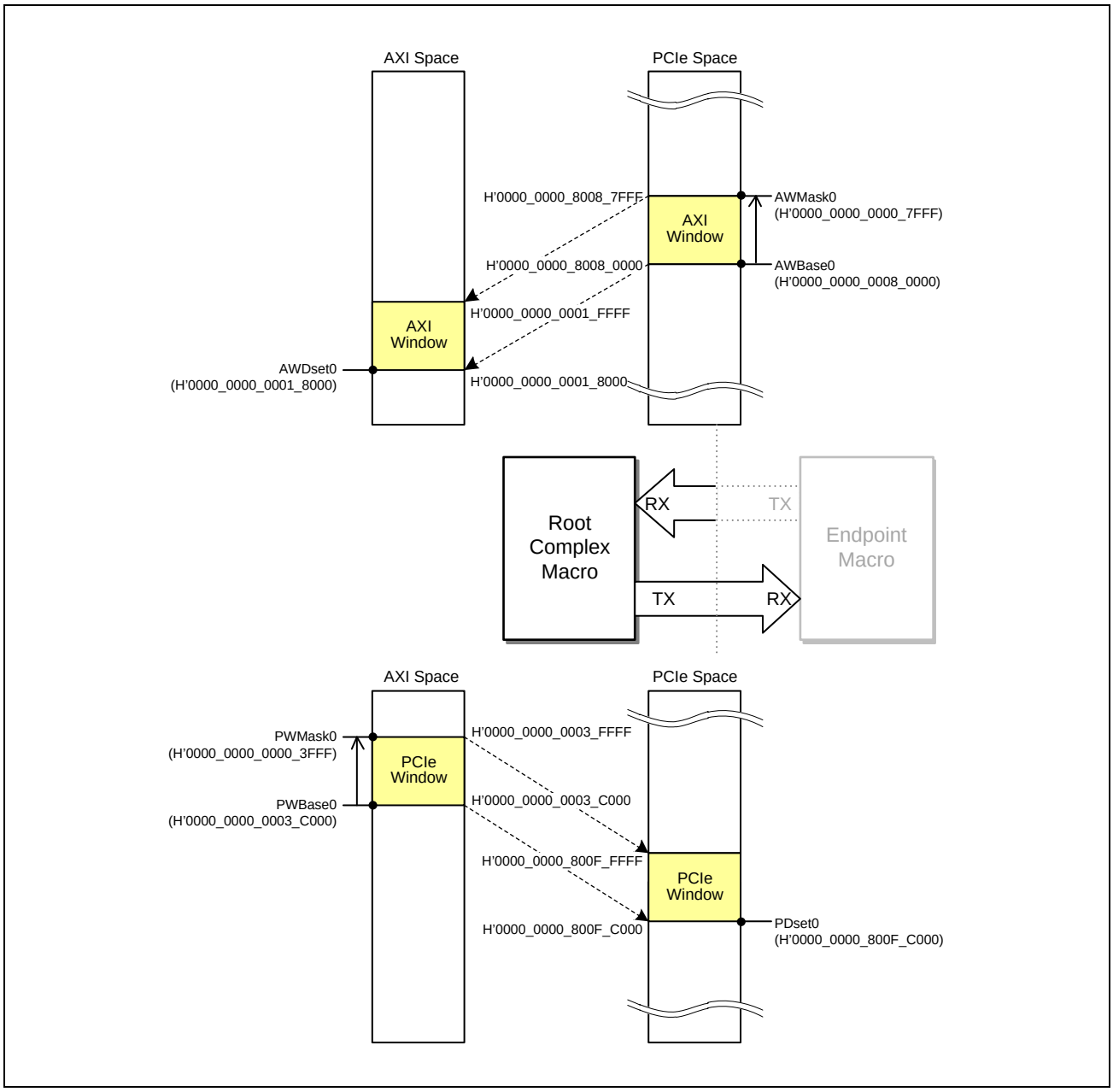
PCIe Destination x Register (Upper) (PCI_PDestxU) (Offset: H'114/134/154/174/194/1B4/1D4/1F4)

(x = 0 to 7)



Note: The values assigned in the figure (values in blue) are examples.

Figure 34.7 Example Settings of the PCIe Window



34.5.2.3 Device Search

On PC systems, the Root Complex (Host CPU) reads the Vendor ID/Device ID and Header Type in the oncoming device's Configuration Register to determine what product is connected as the oncoming device. Please check Base Spec for details.

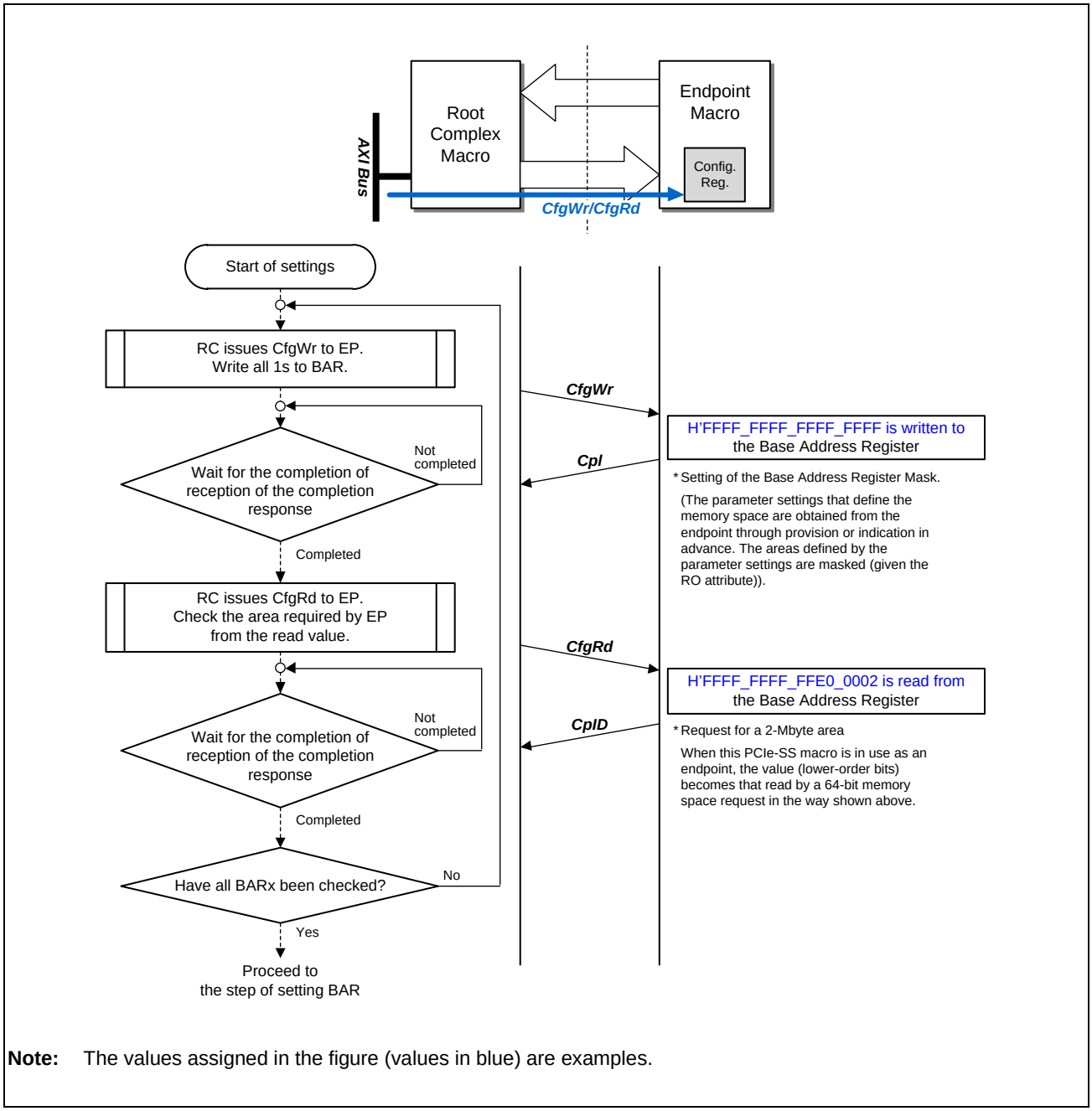
If it is clear that the oncoming device is an endpoint device, such as an embedded system, there is no need to perform a device search. However, the allocation of the bus number must be performed from the Root Complex side to the Endpoint side.

34.5.3 Setting the PCIe Address Space

A PC or similar system recognizes information on the areas of the memory space which are required by other-party endpoint devices by access to the configuration registers of the endpoints through software processing by the CPU on the root-complex side. In the case of embedded systems, etc., the areas required by the other-party devices will generally be known in advance, so this processing is not usually required.

The following is an example procedure for setting of the PCIe memory space for an other-party endpoint by the root complex.

Configuration requests are made to be issued to the other-party endpoint devices by register access through the AXI slave interface.



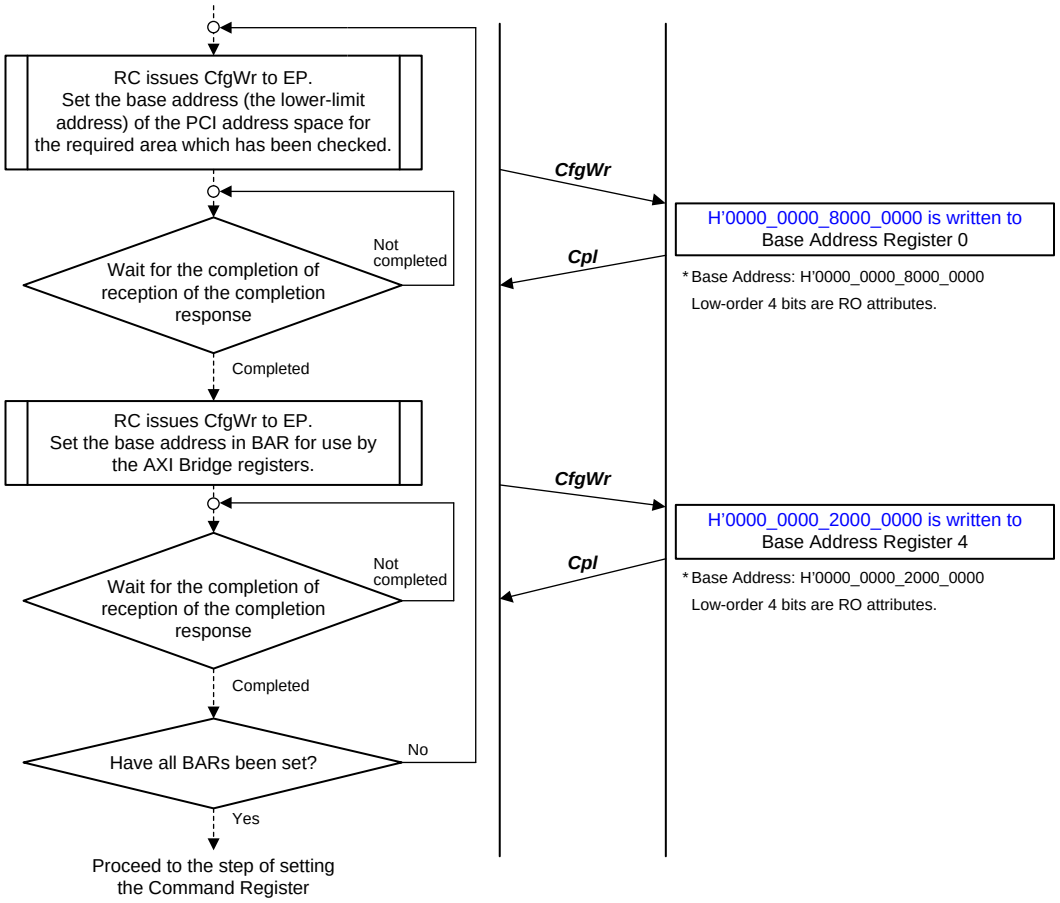
Note: The values assigned in the figure (values in blue) are examples.

Figure 34.8 Example Settings for the PCIe Address Space

After issuing a configuration write request for writing all 1s (H'FFFF_FFFF), a configuration read request is issued and if the result of reading is all 0s (H'0000_0000), the given base address register (BAR) is judged to be reserved and the corresponding space is considered to be unused.

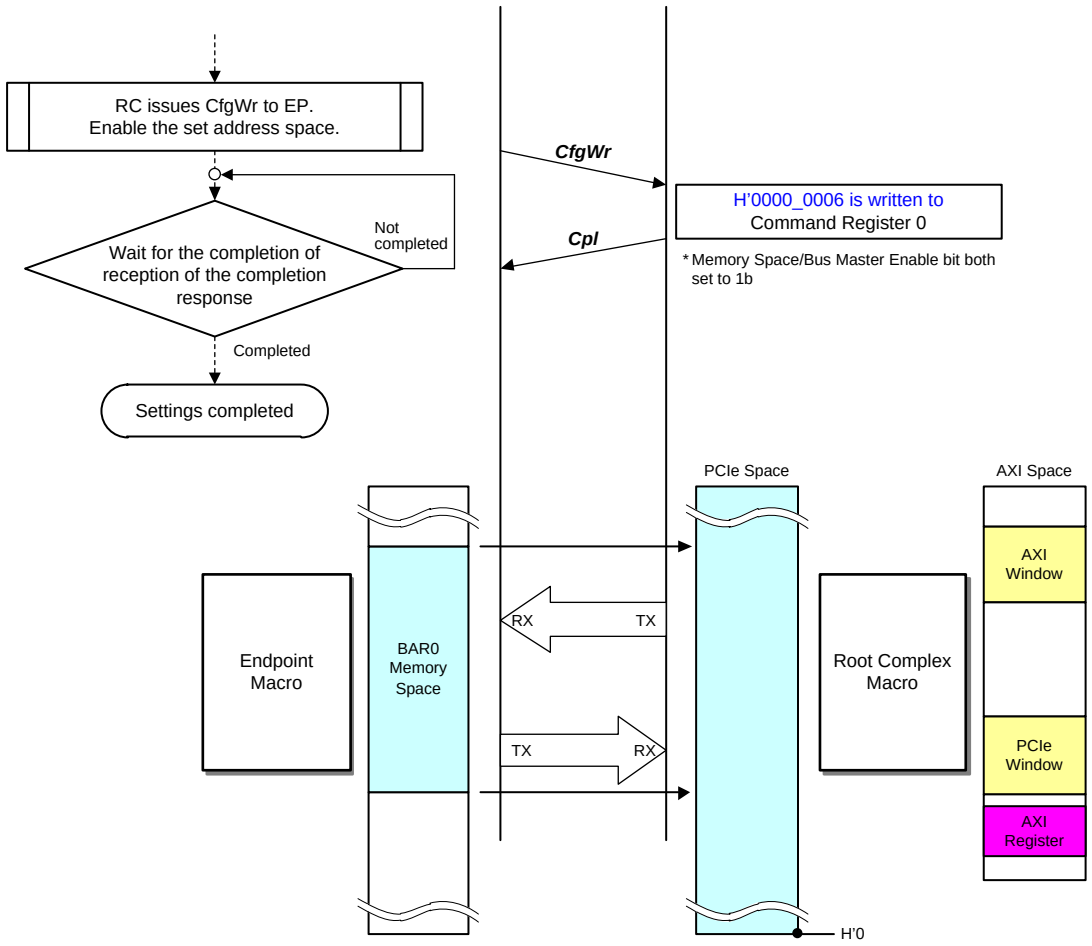
From the result of reading, software on the root complex side writes the base address (the lower-limit address) of the PCI address space to be allocated to the endpoint device by issuing a configuration write request.*[1] At this time, setting of a BAR for which the area requested by the endpoint device was found to be reserved and thus unused is not required. Also, if an address space which includes the other party has been prepared in advance, the above confirmation is not required. Set the base address directly.

Note 1. Setting a base address register to all 0s is prohibited.



Note: The values assigned in the figure (values in blue) are examples.

The completion of settings for the PCIe address space means that the root complex is ready for data transfer to the corresponding endpoint. Finally, set the Bus Master Enable bit and the Memory Space Enable bit of the given command register to enable the memory space.



Note: The values assigned in the figure (values in blue) are examples.

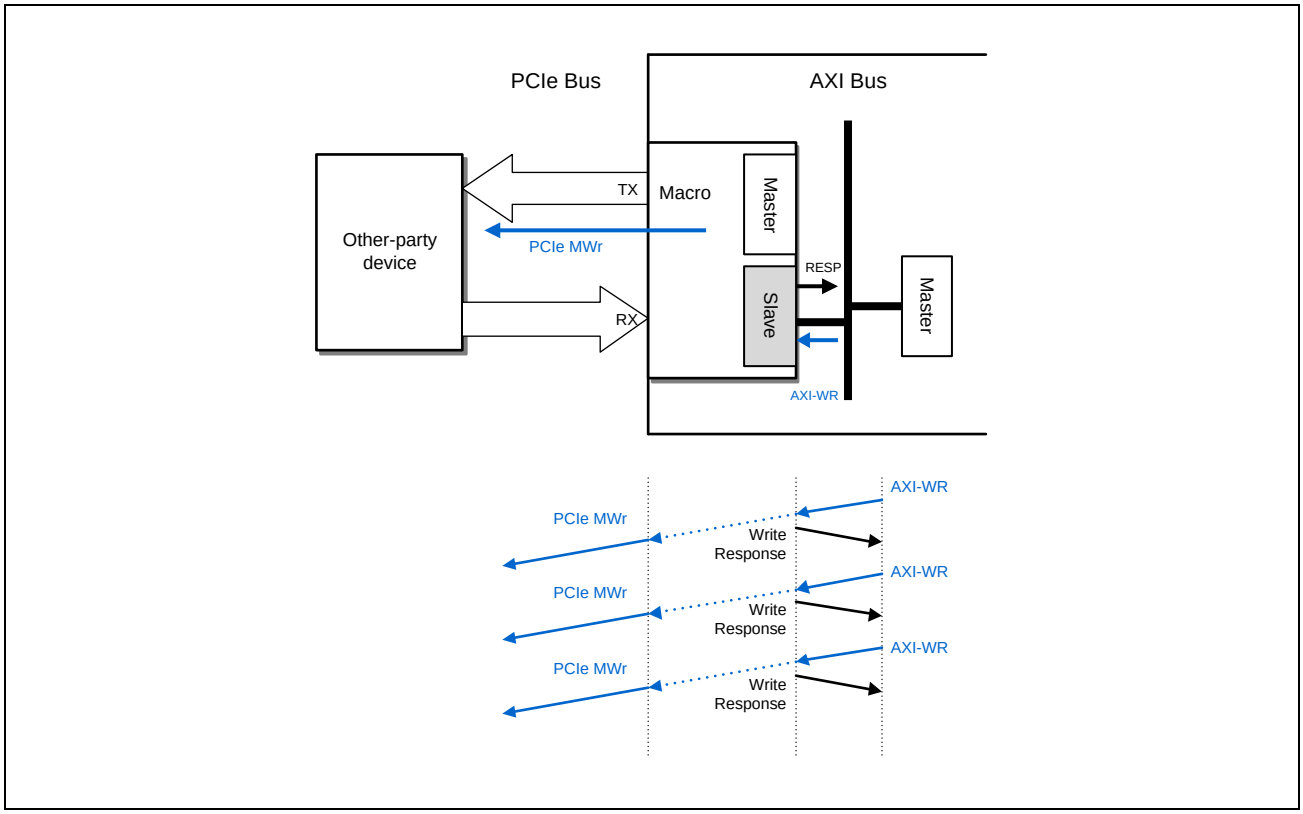
34.5.4 Data Transfer

This module has one port each for master and slave operation as an AXI interface. For PCIe requests which can be issued through the master and slave ports, see **Section 34.4.2, Issuing of PCIe Request and Register Access (From AXI)** and **Section 34.4.3, Issuing of AXI Transaction and Register Access (From PCIe)**.

The focus of this explanation is on normal memory data transfer.

34.5.4.1 PCIe MWr (when the AXI Slave is in Use)

A write transaction from the AXI bus (AXI-WR) via a window set as a PCIe window is converted into an MWr command (PCIe MWr) and then issued. If a PCIe MWr request is to be issued through the AXI interface slave port of the module by using a DMAC, etc. external to the module, the number of write transactions which can be accepted at a time is one, so the operation is as follows.



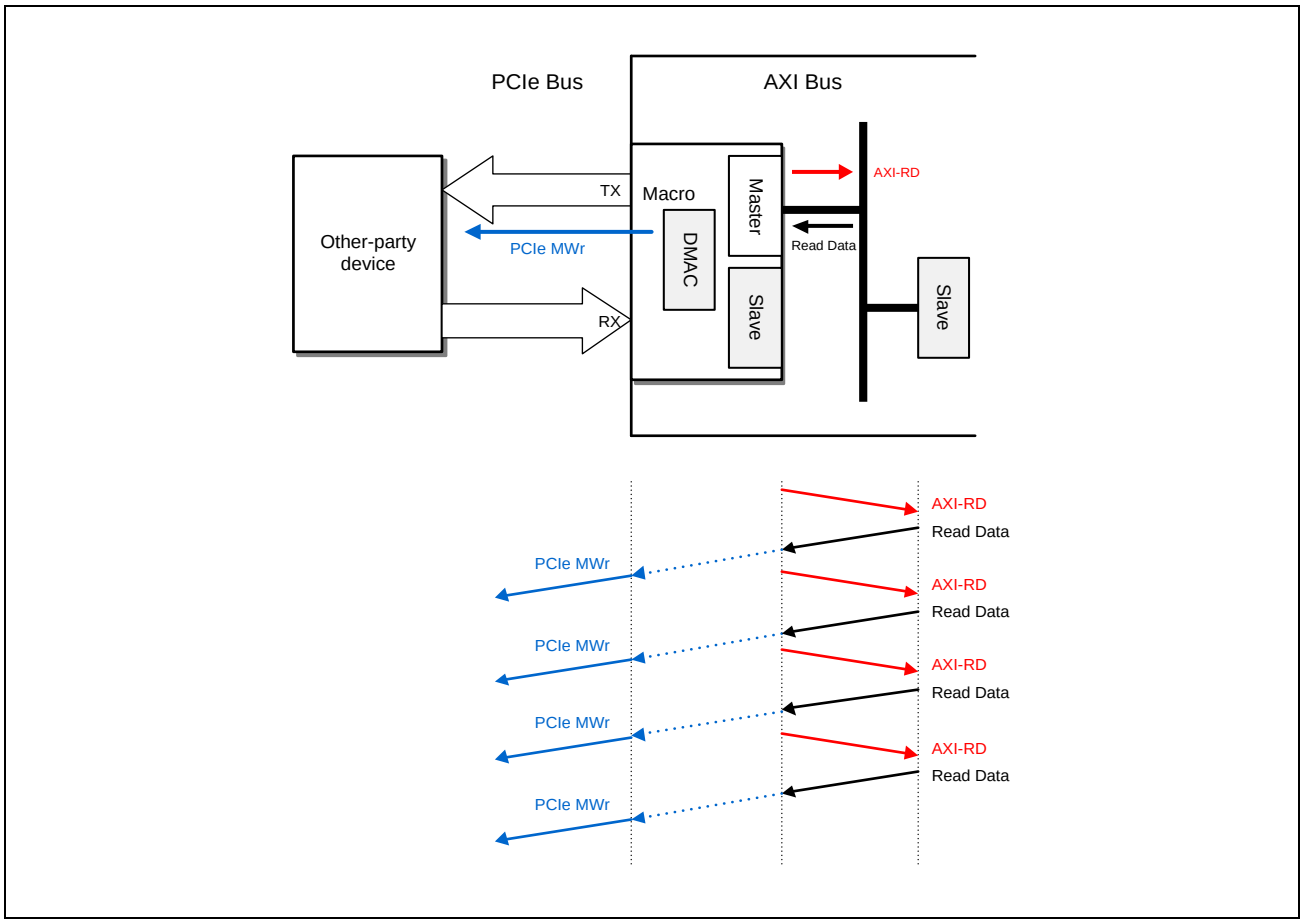
1. A write transaction from the AXI bus is issued through the AXI slave interface (write address channel, write data channel).
2. The transaction is converted into a PCIe MWr command and data are transferred to the other-party device via the PCIe bus.
3. After a wait for a response from the AXI slave interface (write response channel), a next AXI write transaction is issued.

A write response is issued after writing data to the transmission buffer (SRAM) within the module. If this transmission buffer is full, the module receives as much data as it can capture and then places the READY signal at the low level. Accordingly, the AXI bus may be placed in the hold state during transfer.

Since this depends not only on the specifications of the PCIe module (the number of lanes, rate, maximum payload size, etc.) but also on the size of the transmission buffer, the size of the reception buffer of the other-party device, and the external system configuration (the time for access to the external DRAM, etc.), care should be taken when considering the system configuration in general.

34.5.4.2 PCIe MWr (when the DMAC is in Use)

If a PCIe MWr request is to be issued through the AXI master port by using the DMAC within the module, the number of requests which can be read by the internal DMAC is one transaction per channel, so the operation is as follows.



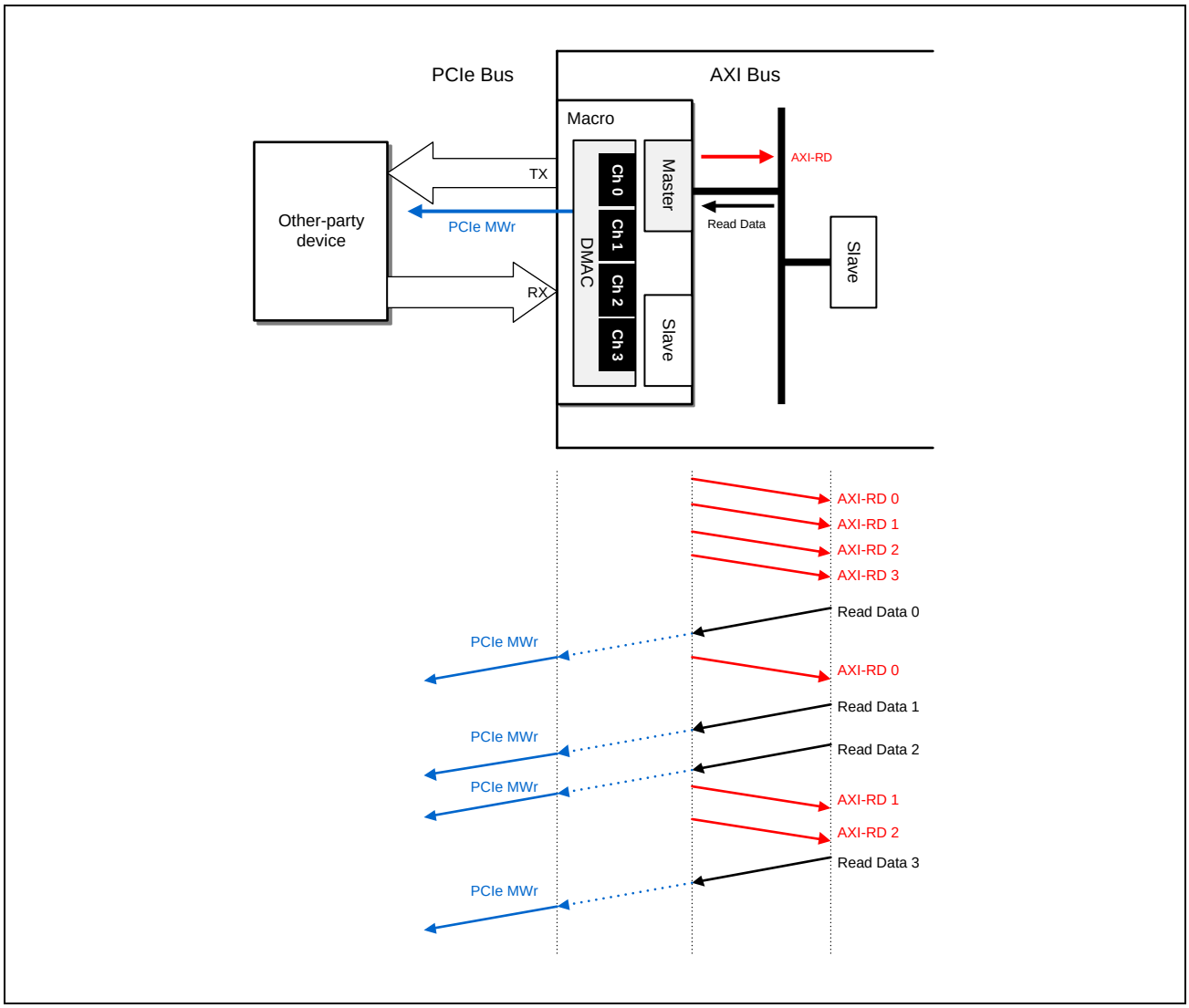
1. An AXI read (AXI-RD) request is issued through the AXI master interface (read address channel).
2. After the AXI master interface (read data channel) receives read data, this is converted into a PCIe MWr command and data are transferred to the other-party device via the PCIe bus.
3. Following the completion of the reception of read data, a next AXI-RD is issued.
4. The above steps are repeated until transfer of all bytes set in the DMA Size bits is completed.

The read data channel checks that the data that have been read are valid and indicates the readiness of the data for reception. Even when the transmission buffer (SRAM) within the module is full, the module issues read requests through the read address channel. Also, it receives as much data as it can capture and places subsequent data in the non-receivable state. Accordingly, depending on the state of the transmission buffer, data may not be transferred regardless of a request having been issued, or the AXI bus may be placed in the hold state during transfer. This lengthens the period of waiting for data that have been read, leading to the deterioration of transfer performance as well as the deterioration of overall system performance.

Likewise, contention with write access by the slave interface and contention in write access between channels may decrease performance in transfer.

Since this depends not only on the specifications of the PCIe module (the number of lanes, rate, maximum payload size, etc.) but also on the size of the transmission buffer, the size of the reception buffer of the other-party device, and the external system configuration (the time for access to the external DRAM, etc.), care should be taken when considering the system configuration in general.

The following is an example where the number of requests which can be read by the AXI master interfaced through four DMAC channels is 4.



1. An AXI read (AXI-RD0: DMAC ch. 0) request is issued through the AXI master interface (read address channel).
2. Since the number of requests which can be read by the module = 4, AXI read requests are subsequently issued through DMAC ch. 1, ch. 2, and ch. 3.

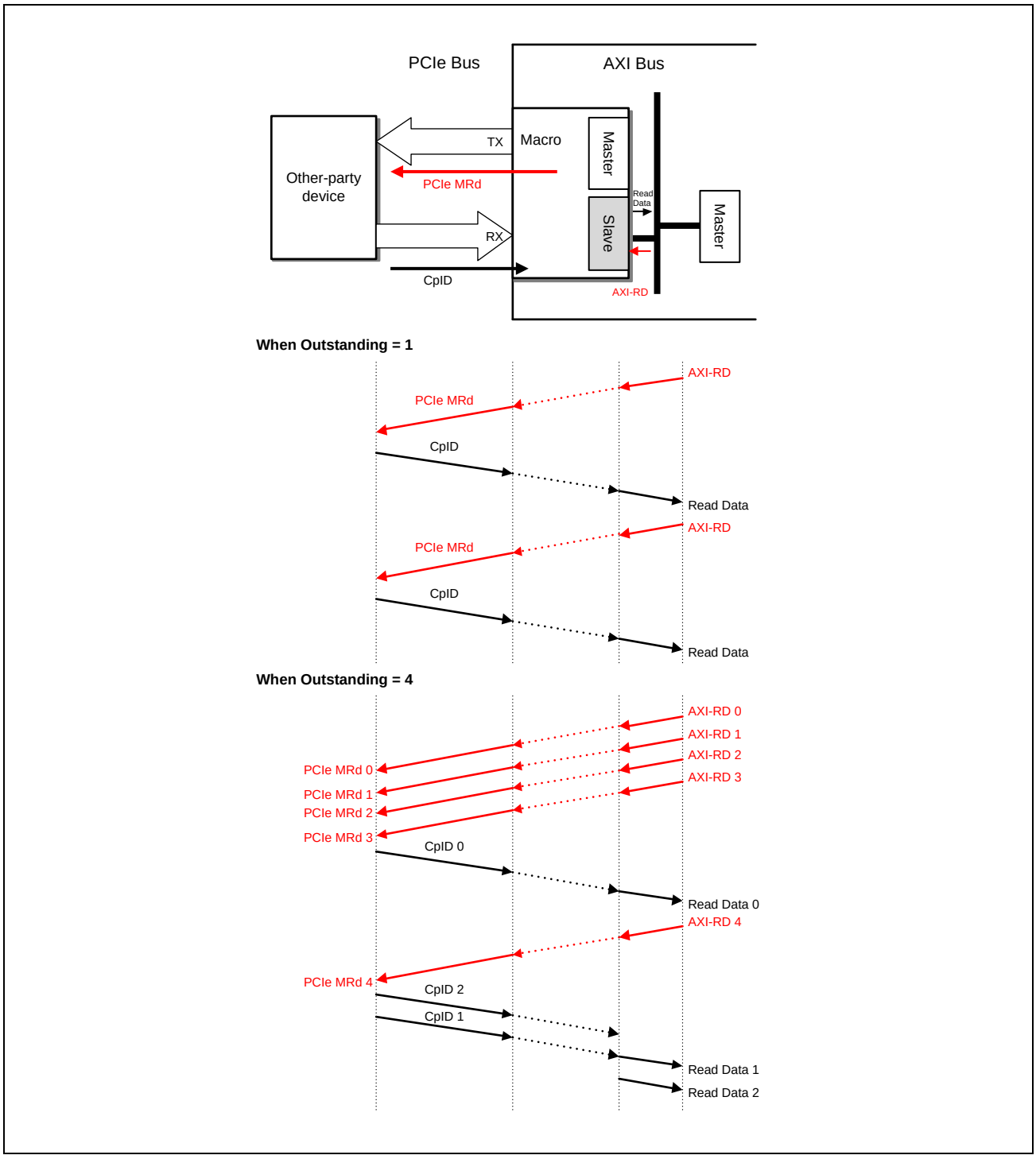
   Note: No order of priority applies to the issuing of requests through these channels.
3. Following the completion of the reception of data read in response to the read request through ch. 0, ch. 0 is able to issue a next AXI-RD0.
4. The above steps are repeated until transfer of all bytes set in the DMA Size bits for each channel is completed.

Configuring data transfer as described above allows increased performance, although such a configuration complicates software control.

34.5.4.3 PCIe MRd (when the AXI Slave is in Use)

In usage as an AXI slave, memory read requests (PCIe MRd) are issued to the PCIe interface, a completion (CplD) is received from the PCIe interface, and an AXI write transfer (AXI-WR) is initiated (this module supports eight outstanding transfers). Therefore, only the number of PCIe MRd requests that corresponds to this set number of outstanding transfers can be issued first, regardless of the reception of CplD. The received CplDs are stored in a data buffer (RAM) and then transferred through the AXI bus.

If a PCIe MRd is issued through the AXI interface slave port of the module, the number of read requests which can be accepted by the AXI slave at a time = 1 to 8 corresponding to the PCIe section.



When outstanding = 1

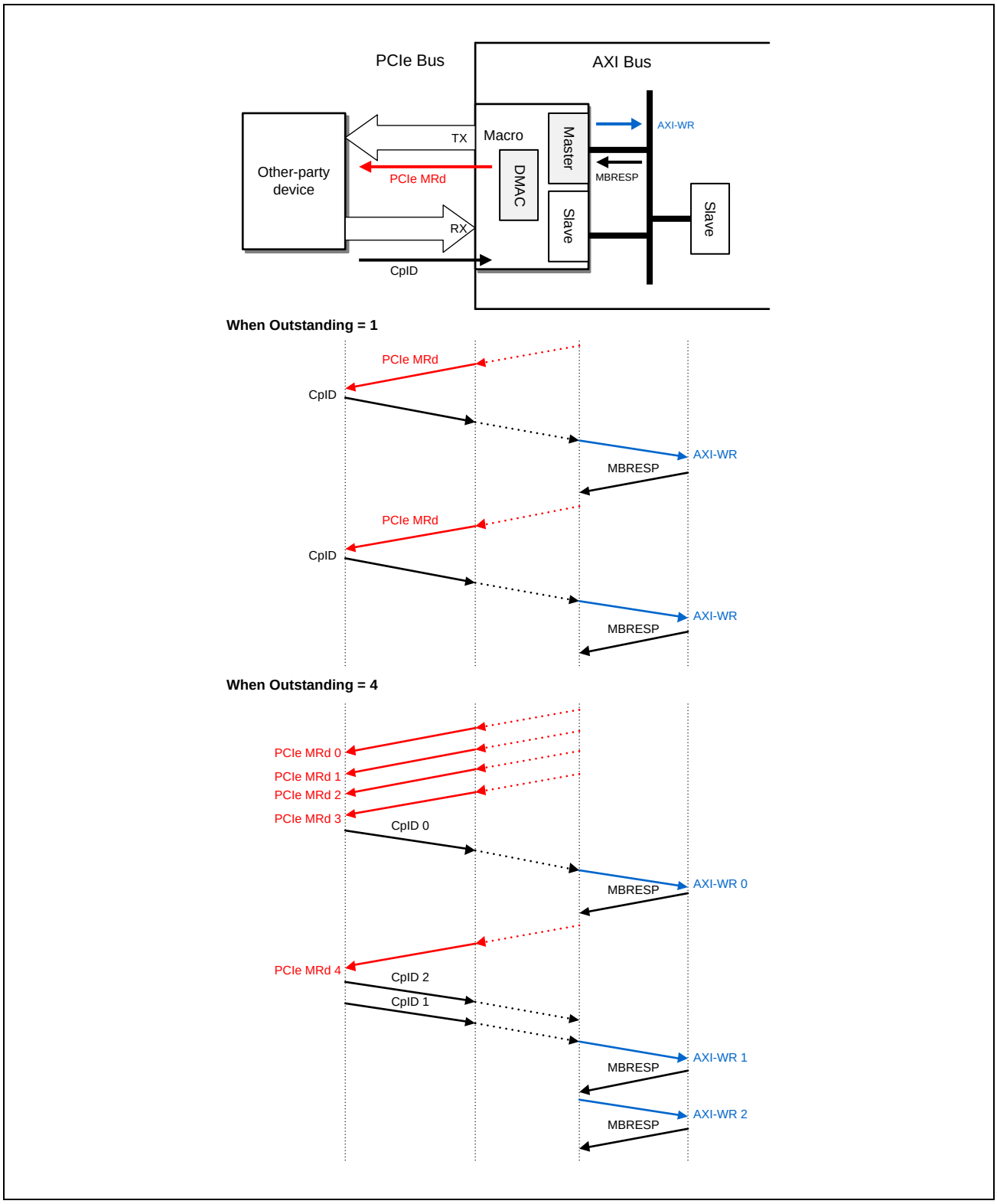
1. An AXI read transaction is issued through the AXI slave interface (read address channel).
2. A transaction is converted into a PCIe MRd command and the request for data reading is issued to the other-party device via the PCIe bus.
3. CplD is received from the other-party device and an AXI read response is issued through the AXI slave interface (read data channel).
4. Steps 1 to 3 are repeated.

When outstanding = 4

1. An AXI read transaction is issued through the AXI slave interface (read address channel).
2. A transaction is converted into a PCIe MRd command and the request for data reading is issued to the other-party device via the PCIe bus.
   Steps 1 and 2 can be repeated until up to four consecutive requests have been issued.
3. CplD is received from the other-party device and an AXI read response is issued through the AXI slave interface (read data channel).
   At this time, the PCIe module can operate with out-of-order transactions, so the order of CplDs from the other-party device may not be preserved. The module also includes a buffer to support in-order transactions with the AXI bus. If this is used and the CplD for a preceding PCIe MRd is never returned, however, the next request cannot be issued and performance deteriorates accordingly.
   After that, steps 1 to 3 are repeated.

34.5.4.4 PCIe MRd (when the DMAC is in Use)

If a PCIe MRd request is to be issued by using the DMAC within the module, the overall number of requests is also 1 to 8 when the DMAC is incorporated depending on the number of outstanding transfers.



1. Activate the internal DMAC through the AXI slave interface or from an other-party device.

When outstanding = 1

2. The DMAC issues a read request and this is converted to a PCIe MRd command within the module. The request for data reading is issued to the other-party device via the PCIe bus.
3. CplD is received from the other-party device and an AXI write transaction is issued through the AXI master interface (write address channel, write data channel).
4. Following the completion of the AXI write transfer, a next request is issued.
5. Steps 2 to 4 are repeated until transfer of all bytes set in the DMA Size bits is complete

When outstanding = 4

2. The DMAC issues a read request and this is converted to a PCIe MRd command within the module. The request for data reading is issued to the other-party device via the PCIe bus. Up to four consecutive requests can be issued.
3. CplD is received from the other-party device and an AXI write transaction is issued through the AXI master interface (write address channel, write data channel).
   At this time, the PCIe module can operate with out-of-order transactions, so the order of CplDs from the other-party device may not be preserved. The module also includes a buffer to support in-order transactions with the AXI bus. If this is used and the CplD for a preceding PCIe MRd is never returned, however, the next request cannot be issued and performance deteriorates accordingly.
4. Following the completion of the AXI write transfer, a next request is issued.
5. Steps 2 to 4 are repeated until transfer of all bytes set in the DMA Size bits is completed.

34.6 Usage Notes

34.6.1 Points for Caution and Prohibited Items in the Issuing of Requests

The following describes the restrictions in the issuing of various requests.

(1) Prohibition of fixed bursts of 2 or more beats

Fixed bursts for 2 or more beats are prohibited. If an attempt to use these is made, an OKAY response is returned to the AXI bus, but this may lead to issuing of the unexpected requests to the PCIe side, or unexpected register access which will change register values, and so on.

(2) Points for caution in the issuing of special requests

The requests listed below are issued by access to the internal registers. The registers are only accessible from the AXI side and writing to them while requests are being processed is prohibited. Attempted access from the PCIe side will be ignored.

[Special Requests]

- Zero-Length Memory Read Request
- Configuration Read
- Configuration Write
- Message Request
- Message Request with data payload

Remark Do not issue a special request for which issuing of the request is prohibited. Operation is not guaranteed if this is done.

34.6.2 Ordering Specifications of Received Non-Posted and Posted Requests

In the specifications of this module, a Non-Posted Request is not overtaken by a Posted Request on the receiving side. Requests received from another party are output to the higher-level bus in the order of reception.

34.6.3 Caution when Changing the Speed Spontaneously

Follow the procedure below. The following control bits are present in the PCI Core Control 2 Register and PCI Core Status 2 Register among the AXI-Bridge registers.

1. Wait until the home node is placed in the L0 state.
2. Read the value of the STATE_DATA_RATE_IDENTIFIER_RECEIVED bit to check the supported speed of the other-party node.
3. If the home node supports the speed to which the other-party node wants to change, set the UI_LINK_SPEED_CHANGE[1:0] bits and assert the UI_LINK_SPEED_CHANGE_REQ bit.
4. Wait for the UI_LINK_SPEED_CHANGE_DONE bit to be asserted (wait for the completion).

Remark UI_LINK_SPEED_CHANGE_REQ is retained until UI_LINK_SPEED_CHANGE_DONE is asserted. It should be de-asserted after checking the assertion of UI_LINK_SPEED_CHANGE_DONE.

34.6.4 Error Processing of Unsupported Requests

The following describes the flow of processing for error reporting and error logging.

(1) Access that straddles a 4-KB boundary

Data transfer to a memory space which straddles a 4-KB boundary cannot proceed (this is stipulated by the PCI Express Base Specification). In this module, the reception of a memory write or read request for a memory space which straddles a 4-KB boundary is handled as a malformed TLP.

(2) RCB violations

The PCI Express Base Specification states that error processing in response to violations of the read completion boundary (RCB) is optional. The PCIe module of this LSI chip does not support the detection of RCB violations at the time of the reception of completion responses. Support for this function should be handled by the user logic.

(3) Error processing in response to byte enable fields.

The PCI Express Base Specification states that error processing in response to Byte Enable fields is optional. This module does not support the detection of errors in the form of violations of Byte Enable rules. Support for this function should be handled by the user logic.

(4) Request that extends beyond the base address boundary

This module does not detect a request that starts within the range set in the Base Address Register but extends beyond the boundary from the base address as an error. Such a request should be handled by the user logic.

(5) Processing when a TPL that is ready to be transferred by the transmitter is a malformed TLP

Even if malformation of a TLP that is ready to be transferred by the transmitter is detected, due to the detection of parity errors when it is read from the FIFO buffer and so on, this unit does not suspend the transmission of such a TLP. Received malformed TLPs must be handled appropriately by other-party devices.

34.6.5 Processing on Reception of the Message

The following lists the messages to be detected as a silent drop or UR on reception of the given message.

Table 34.23 Operations on Message Reception

Received Message	Root Complex
Assert_INTx	Normal processing
Deassert_INTx	Normal processing
ERR_COR	Normal processing
ERR_NONFATAL	Normal processing
ERR_FATAL	Normal processing
UNLOCK	Silent Drop
Set_Slot_Power_Limit	Silent Drop
Vendor_Defined_Type0	UR
Vendor_Defined_Type1	Silent Drop
Ignore	Silent Drop
LTR	Normal processing
OBFF	UR
PM_PME	Normal processing
PME_TO_Ack	Normal processing
PM_Active_State_Nak	Silent Drop
PME_Turn_Off	Silent Drop
PTM_Request	UR
PTM_Response	UR
PTM_ResponseD	UR
Invalidate_Request	Silent Drop
Invalidate_Completion	Normal processing
Page_Request	UR
Page_Response	UR

Note: Silent Drop: Normal completion, data are not reflected
UR: Unsupported Request

34.6.6 Other Points for Caution

(1) Point for caution on register writing

In some cases of writing 2DW or more and skipping over with the use of byte enable to a register from the AXI side, writing might not proceed as expected.

Using a pin reset or register reset from the AXI side as a non-consecutive SWSTRB, restricts writing to no more than 1DW (32 bits).

(2) Point for caution on register reading

In the case of reading a register from the AXI side, values read from invalid byte lanes are from undefined outputs (meaningless garbage data).

(3) Generation of unexpected correctable errors

A correctable error may be detected at the time of EIOS reception following low power state transitions of this module such as from L0s to L1. If this happens, processing to send a message or assert an interrupt flag signal is to proceed. Take care on this point and respond appropriately so that a correctable error is not handled as a fatal error (a mask setting by a register to switch the notification of unexpected correctable errors off is recommended).

(4) Reset interval in transitions from a hot reset to detection in RC mode

In generations of PCI Express after Gen1, at the time of transitions from a hot reset to detection in accord with the operating rate, the base specification prescribes securing a 1-ms waiting interval for changes to the rate, so secure a reset interval of 1 ms.

(5) How to use the PCI_RST_OUT# terminal

By connecting this terminal to the “PERST#” input terminal on the Endpoint side of the opposite device in the Root Complex mode, initialization of the opposite device can be controlled by the Root Complex side.

(6) Retimer

One Retimer can be used.

(7) Point for caution on Window setting

Window addressing has the following precautions and limitations:

- When the BAR* register and the corresponding BAR Mask* register setting value are added, bit carry should not be allowed.
- When the AXI Window Base* register and the corresponding AXI Window Mask* register setting value are added, bit carry should not be allowed.
- When the PCIe Window Base* register and the corresponding PCIe Window Mask* register setting value are added, bit carry should not be allowed.
- When the AXI Window Mask* register and the corresponding AXI Destination* register setting value are added, bit carry should not be allowed.
- When the PCIe Window Mask* register and the corresponding PCIe Destination* register setting value are added, bit carry should not be allowed.

- The value of BAR Mask*, AXI Window Mask* and PCIe Window Mask* register should be set a power of 2 minus 1.

  Ex) H’0000_0007_FFFF_FFFF is OK. H’0000_0001_0FFF_FFFF is NG.
  – Each Window should be a single memory size with a power of 2.
  – The minimum space between BAR-BAR masks is 4 Kbytes.

35. Serial Sound Interface (SSIF-2)

35.1 Overview

The serial sound interface (SSIF-2) transmits and receives audio data to and from various devices that are compatible with I^2S format, monaural format and TDM format.

35.1.1 Features

Table 35.1 Features of SSIF-2

Item		Description
Number of interfaces		4
Communication mode		• Master/slave • Full-duplex communication is available in interfaces 0, 1, and 3. Half-duplex communication is only available in interface 2.
Communication format		• I^2S format • Monaural format • TDM format
Serial data		• MSB first • Data can be left-justified or right-justified. • Data delay (one clock cycle) or no delay selectable for the period from SSI_RCK to SSI_TXD/SSI_RXD • System word length: 8, 16, 24, 32, 48, 64, 128, or 256 bits • Data word length: 8, 16, 18, 20, 22, 24, or 32 bits • Padding polarity: Low or high
Bit clock (SSIBCK)	In master mode	• Clock source division ratio: 1/1, 1/2, 1/4, 1/6, 1/8, 1/12, 1/16, 1/24, 1/32, 1/48, 1/64, 1/96, and 1/128. • Supply/stop of SSI_BCK is selectable while communication is halted.
	In slave mode	SSI_BCK input method: Direct input or via the noise canceller
	In master/slave mode	Polarity: Rising/falling edge canceler
Left and right clock/frame synchronization (SSILRCK/SSIFS)	In master mode	• Polarity: Low/high level selectable • Supply/stop of SSI_RCK is selectable while communication is halted.
	In slave mode	Two input methods: Direct input or via the noise canceller
Transmit data (SSITxD) and receive data (SSIRxD)	Transmission	Mute: Transmission of transmit FIFO data or fixed value 0 selectable
	Reception (in slave mode)	Two input methods: Direct input or via the noise canceller
FIFO	Capacity	• Transmit FIFO/receive FIFO: 4 bytes × 32 stages, • Receive FIFO: 4 bytes × 32 stages
	Data alignment	Method of data transfer between FIFO and shift register (Left-justification and right-justification) selectable.
Interrupt	Interrupt output	• Communication error/idle mode error (level) • Receive data full interrupt (edge) • Transmit data empty interrupt (edge) • Receive data full/transmit data empty interrupt (edge)
	Interrupt capture spilling solution function	Edge Interrupt supports inhibition function of capture spilling.
Low power consumption function		Supply/stop of audio clock is selectable in master-mode communication.

The definition of words and phrases used by the format of the communication of SSIF-2 is as follows.

Table 35.2 Word Definition List

Word	Definition
Start trigger	The first edge that reaches value that set to LRCKP in SSI_RCK pin after SSIF-2 permit communication.
Frame boundary	When SSIF-2 begins to transmit the first data of one frame. When SSIF-2 forwarding ends final data of one frame.
Frame word number	Sound channel number in one frame
System word length	Bit length in one frame.
Data word length	Effective bit length in one frame.
Control bit of communication format	• SSICR register: FRM,DWL,SWL,LRCKP,SPDP,SDTA,PDTA,DEL • SSIFCR register: BSW • SSIOFR register: OMOD • SSISCR register: TDES, RDFS

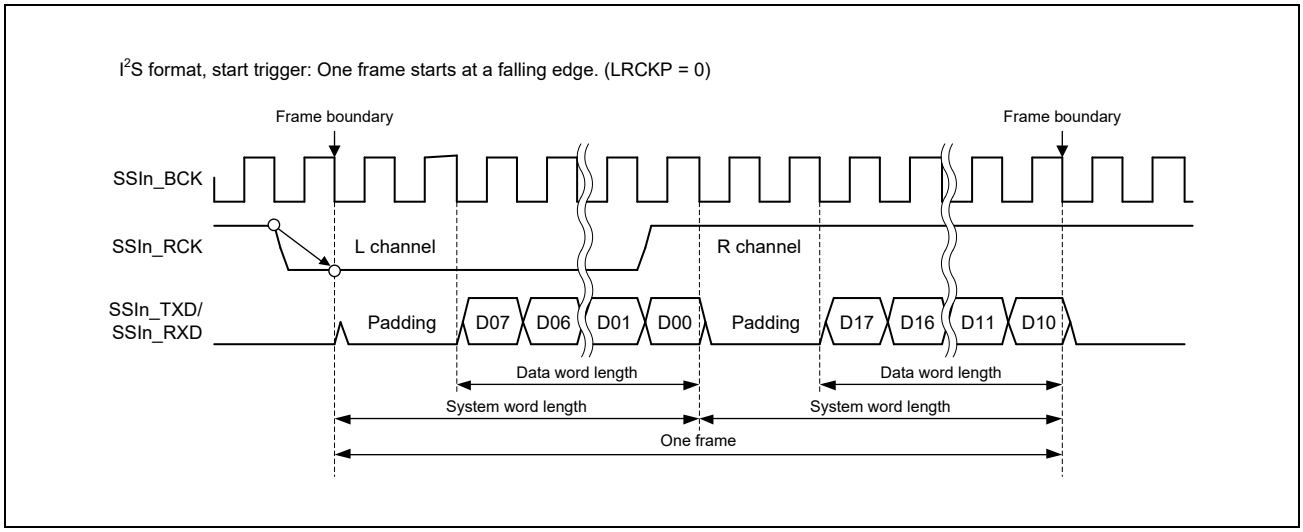
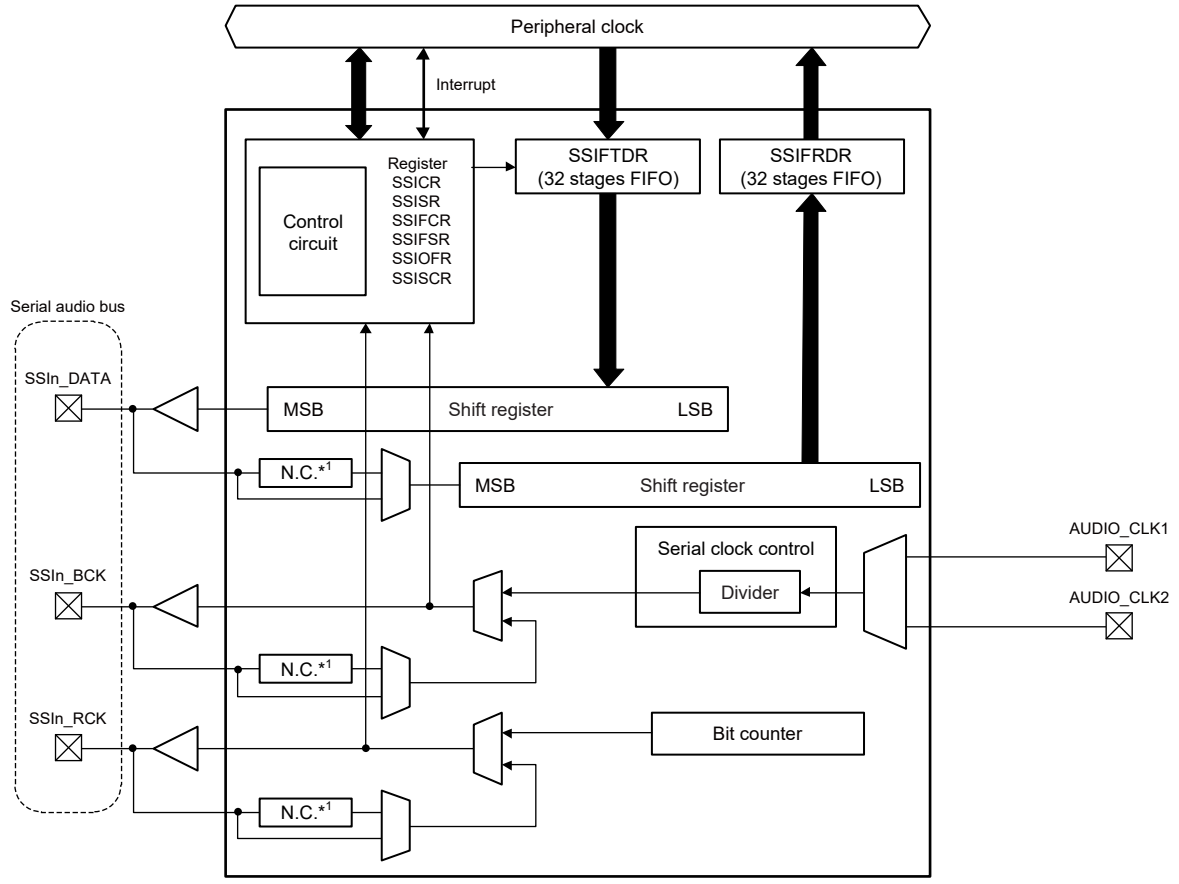


Figure 35.1 Communication Format Definition

35.1.2 Block Diagram

Figure 35.2 shows the block diagram of SSIF-2.



[Legend]

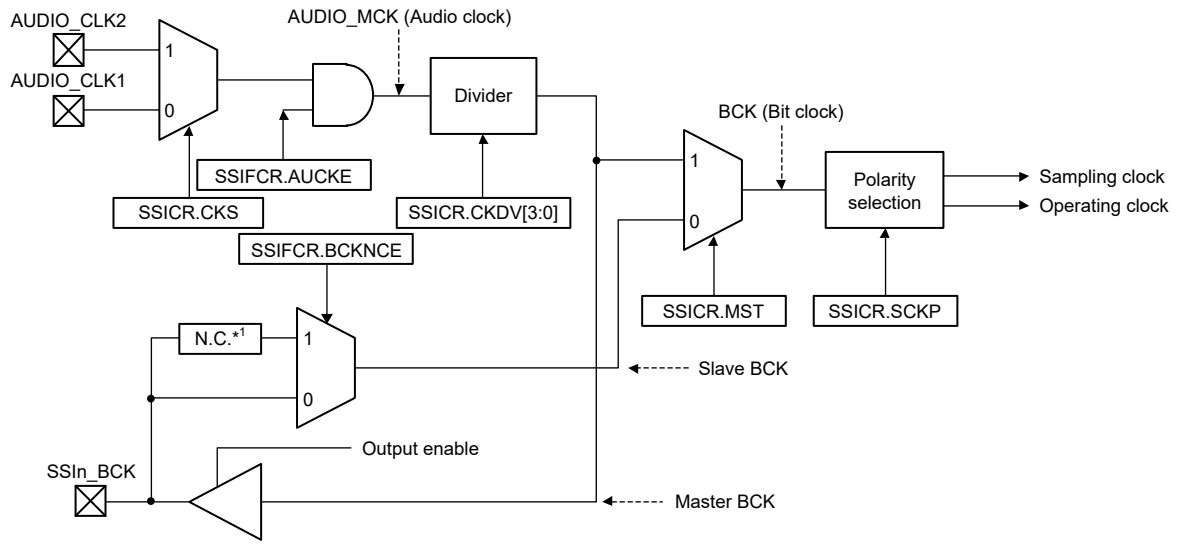
SSICR:	Control register
SSISR:	Status register
SSIFCR:	FIFO control register
SSIFSR:	FIFO status register
SSIOFR:	Audio format register
SSIFTDR:	Transmit FIFO data register
SSIFRDR:	Receive FIFO data register
SSISCR:	Status Control register

Note: In channels 0, 1, and 3, SSn_DATA can be used independently as SSIn_TXD for transmission and SSIn_RXD for reception.

Note 1. N.C.: Noise canceller

Figure 35.2 SSIF-2 Block Diagram

Figure 35.3 shows the clock configuration of SSIF-2.



[Legend]

SSICR: Control register

Note: Switch the above SFRs while communication is halted.

Note 1. N.C. (Noise canceller)

Figure 35.3 SSIF-2 Clock Configuration

35.2 Input/Output Pins

Table 35.3 shows the pin configuration

Table 35.3 Pin Configuration

Interface	Name	I/O	Function
0, 1, 3 (n = 0,1,3)	SSIn_BCK	I/O	Bit clock
	SSIn_RCK	I/O	LR clock/frame synchronization
	SSIn_TXD	Output	Serial data output
	SSIn_RXD	Input	Serial data input
2 (n = 2)	SSIn_BCK	I/O	Bit clock
	SSIn_RCK	I/O	LR clock/frame synchronization
	SSIn_DATA	I/O	Serial data I/O
Common	AUDIO_CLK1	Input	External clock1 for audio (The oversampling clock signal is input.)
	AUDIO_CLK2	Input	External clock2 for audio (The oversampling clock signal is input.)

35.3 List of Registers

Table 35.4 shows the register configuration of SSIF-2. The address of the SSIF-2 register is represented by the offset address from the base address. SSIF-2 base address is as follows:

- SSIF-2 base address: H'0_100A_8000 (ch0), H'0_100A_8400 (ch1), H'0_100A_8800 (ch2), H'0_100A_8C00 (ch3) (Overall Address Space)
- SSIF-2 base address: H'400A_8000 (ch0), H'400A_8400 (ch1), H'400A_8800 (ch2), H'400A_8C00 (ch3) (Cortex-M33/Cortex-M33_FPU Address Space Secure)
- SSIF-2 base address: H'500A_8000 (ch0), H'500A_8400 (ch1), H'500A_8800 (ch2), H'500A_8C00 (ch3) (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 35.4 List of Control Registers (1/2)

Channel	Register Name	Symbol	R/W	Initial Value	Address	Access Size
0	Control register	SSICR	RW	H'0000_0000	H'00	32 bits
	Status register *Note*: Bits 31, 30, and 25 to 0 are read only.	SSISR	RW	H'0200_0000	H'04	32 bits
	FIFO control register	SSIFCR	RW	H'0000_0000	H'10	32 bits
	FIFO status register *Note*: Bits 31 to 17 and 15 to 1 are read only.	SSIFSR	RW	H'0001_0000	H'14	32 bits
	Transmit FIFO data register	SSIFTDR	W	H'0000_0000	H'18	32 bits, 16 bits, 8 bits
	Receive FIFO data register	SSIFRDR	R	H'0000_0000	H'1C	32 bits, 16 bits, 8 bits
	Audio format register	SSIOFR	RW	H'0000_0000	H'20	32 bits
	Status control register	SSISCR	RW	H'0000_0000	H'24	32 bits
1	Control register	SSICR	RW	H'0000_0000	H'00	32 bits
	Status register *Note*: Bits 31, 30, and 25 to 0 are read only.	SSISR	RW	H'0200_0000	H'04	32 bits
	FIFO control register	SSIFCR	RW	H'0000_0000	H'10	32 bits
	FIFO status register *Note*: Bits 31 to 17 and 15 to 1 are read only.	SSIFSR	RW	H'0001_0000	H'14	32 bits
	Transmit FIFO data register	SSIFTDR	W	H'0000_0000	H'18	32 bits, 16 bits, 8 bits
	Receive FIFO data register	SSIFRDR	R	H'0000_0000	H'1C	32 bits, 16 bits, 8 bits
	Audio format register	SSIOFR	RW	H'0000_0000	H'20	32 bits
	Status control register	SSISCR	RW	H'0000_0000	H'24	32 bits

Table 35.4 List of Control Registers (2/2)

Channel	Register Name	Symbol	R/W	Initial Value	Address	Access Size
2	Control register	SSICR	RW	H'0000_0000	H'00	32 bits
	Status register *Note*: Bits 31, 30, and 25 to 0 are read only.	SSISR	RW	H'0200_0000	H'04	32 bits
	FIFO control register	SSIFCR	RW	H'0000_0000	H'10	32 bits
	FIFO status register *Note*: Bits 31 to 17 and 15 to 1 are read only.	SSIFSR	RW	H'0001_0000	H'14	32 bits
	Transmit FIFO data register	SSIFTDR	W	H'0000_0000	H'18	32 bits, 16 bits, 8 bits
	Receive FIFO data register	SSIFRDR	R	H'0000_0000	H'1C	32 bits, 16 bits, 8 bits
	Audio format register	SSIOFR	RW	H'0000_0000	H'20	32 bits
	Status control register	SSISCR	RW	H'0000_0000	H'24	32 bits
3	Control register	SSICR	RW	H'0000_0000	H'00	32 bits
	Status register *Note*: Bits 31, 30, and 25 to 0 are read only.	SSISR	RW	H'0200_0000	H'04	32 bits
	FIFO control register	SSIFCR	RW	H'0000_0000	H'10	32 bits
	FIFO status register *Note*: Bits 31 to 17 and 15 to 1 are read only.	SSIFSR	RW	H'0001_0000	H'14	32 bits
	Transmit FIFO data register	SSIFTDR	W	H'0000_0000	H'18	32 bits, 16 bits, 8 bits
	Receive FIFO data register	SSIFRDR	R	H'0000_0000	H'1C	32 bits, 16 bits, 8 bits
	Audio format register	SSIOFR	RW	H'0000_0000	H'20	32 bits
	Status control register	SSISCR	RW	H'0000_0000	H'24	32 bits

Note: Access to the addresses other than those listed above is prohibited.

35.4 Function Details

35.4.1 Register Descriptions

35.4.1.1 Control Register (SSICR)

This is a 32-bit readable/writable register. With this register, select an audio clock, control IRQ, select data formats, and set an operation mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved. Write 0. The read value is 0.
30	CKS	0	RW	Selects an audio clock for master-mode communication*1 0: Selects the AUDIO_CLK2 input 1: Selects the AUDIO_CLK1 input
29	TUIEN	0	RW	Transmit underflow interrupt output enable 0: Disables transmit underflow interrupt output 1: Enables transmit underflow interrupt output
28	TOIEN	0	RW	Transmit overflow interrupt output enable 0: Disables transmit overflow interrupt output 1: Enables transmit overflow interrupt output
27	RUIEN	0	RW	Receive underflow interrupt output enable 0: Disables receive underflow interrupt output 1: Enables receive underflow interrupt output
26	ROIEN	0	RW	Receive overflow interrupt output enable 0: Disables receive overflow interrupt output 1: Enables receive overflow interrupt output
25	IIEN	0	RW	Idle mode interrupt output enable 0: Disables idle mode interrupt output 1: Enables idle mode interrupt output
24	—	0	R	Reserved. Write 0. The read value is 0.
23, 22	FRM[1:0]	00b	RW	Select frame word number*1

Communication format (SSIOFR.OMOD[1:0])

FRM[1:0]	I²S (00b)	Monaural (10b)	TDM (01b)
00b	2	1	Set prohibition
01b	Set	Set prohibition	4
10b	prohibition		6
11b			8

Bit	Bit Name	Initial Value	R/W	Description
21 to 19	DWL[2:0]	000b	R/W	Selects data word length*1 000: 8 bits 001: 16 bits 010: 18 bits 011: 20 bits 100: 22 bits 101: 24 bits 110: 32 bits 111: Setting prohibited
18 to 16	SWL[2:0]	000b	RW	Selects system word length*1 000: 8 bits 001: 16 bits 010: 24 bits 011: 32 bits 100: 48 bits 101: 64 bits 110: 128 bits 111: 256 bits
15	—	0	R	Reserved. Write 0. The read value is 0.
14	MST	0	RW	Master enable*1 0: Slave-mode communication 1: Master-mode communication
13	BCKP	0	RW	Selects bit clock polarity*1 0: SSILRCK/SSIFS and SSIFTxD/SSIFRxD change at a falling edge (sampling SSILRCK/SSIFS and SSIFRxD at a rising edge of SSIBCK). 1: SSILRCK/SSIFS and SSIFTxD/SSIFRxD change at a rising edge (sampling SSILRCK/SSIFS and SSIFRxD at a falling edge of SSIBCK).
12	LRCKP	0	RW	Selects the initial value and polarity of LRCK/FS*1 0: The initial value is at a high level For SSILRCK/SSIFS, one frame start trigger is falling edge of SSILRCK/SSIFS. 1: The initial value is at a low level For SSILRCK/SSIFS, one frame start trigger is rising edge of SSILRCK/SSIFS.
11	SPDP	0	RW	Selects serial padding polarity*1 0: Padding data is at a low level 1: Padding data is at a high level
10	SDTA	0	RW	Selects serial data alignment*1 0: Transmits and receives serial data first and then padding bits. 1: Transmit and receives padding bits first and then serial data.
9	PDTA	0	RW	Selects placement data alignment*1 0: Left-justifies placement data (SSIFTDR, SSIFRDR) 1: Right-justifies placement data (SSIFTDR, SSIFRDR)
8	DEL	0	RW	Selects serial data delay*1 0: Delay of one cycle of SSIBCK between SSILRCK/SSIFS and SSIFTxD/SSIFRxD. 1: No delay between SSILRCK/SSIFS and SSIFTxD/SSIFRxD. This bit control SSILRCK/SSIFS in monaural format (Refer to **Section 35.4.2.2, Monaural Format** for details).

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	CKDV[3:0]	H'0	RW	Selects bit clock division ratio*1 CKDV[3:0] 0000: AUDIO_MCK 0001: AUDIO_MCK/2 0010: AUDIO_MCK/4 0011: AUDIO_MCK/8 0100: AUDIO_MCK/16 0101: AUDIO_MCK/32 0110: AUDIO_MCK/64 0111: AUDIO_MCK/128 1000: AUDIO_MCK/6 1001: AUDIO_MCK/12 1010: AUDIO_MCK/24 1011: AUDIO_MCK/48 1100: AUDIO_MCK/96 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited
3	MUEN	0	RW	Mute (silent) enable 0: Disables muting on the next frame boundary. 1: Enables muting on the next frame boundary.
2	—	0	R	Reserved. Write 0. The read value is 0.
1	TEN	0	RW	Transmission and reception enable*2
0	REN	0	RW	(TEN, REN): Operating 00: Disables transmission and reception 01: Enables reception (starts reception) 10: Enables transmission (starts transmission) 11: Enables transmission and reception (starts transmission and reception)

Note 1. Writing to these bits while SSIF-2 is in a communication state (SSISR.IIRQ = 0) is prohibited. If written, the operation performed immediately after writing is not guaranteed.

Note 2. If the TEN bit or REN bit is rewritten, make sure that the SSISR.IIRQ bit is in the desired status. If written, the operation performed immediately after writing is not guaranteed. For example, after enabling operation, make sure that SSISR.IIRQ = 0 and after disabling operation, make sure that SSISR.IIRQ = 1.

CKS Bit

This bit sets the audio clock in master-mode communication (MST = 1). In slave-mode communication (MST = 0), setting of this bit is invalid.

To write this bit, do so when AUDIO_MCK is not supplied. Refer to the AUCKE bit details explanation in **Section 35.4.1.3, FIFO Control Register (SSIFCR)** for detailed timing.

TUIEN Bit

This bit enables/disables output of transmit underflow interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.TUIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.TUIRQ = 1.

TOIEN Bit

This bit enables/disables output of transmit overflow interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.TOIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.TOIRQ = 1.

RUIEN Bit

This bit enables/disables output of receive underflow interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.RUIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.RUIRQ = 1.

ROIEN Bit

This bit enables/disables output of receive overflow interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.ROIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.ROIRQ = 1.

IIEN Bit

This bit enables/disables output of idle mode interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.IIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.IIRQ = 1.

FRM[1:0] Bits

These bits set system word number in one frame of the communication formats.

Rewrite these bits when the LR clock supply to the terminal SSILRCK stops. Refer to the LRCONT bit details explanation in **Section 35.4.1.7, Audio Format Register (SSIOFR)** for the output operation of the LR clock.

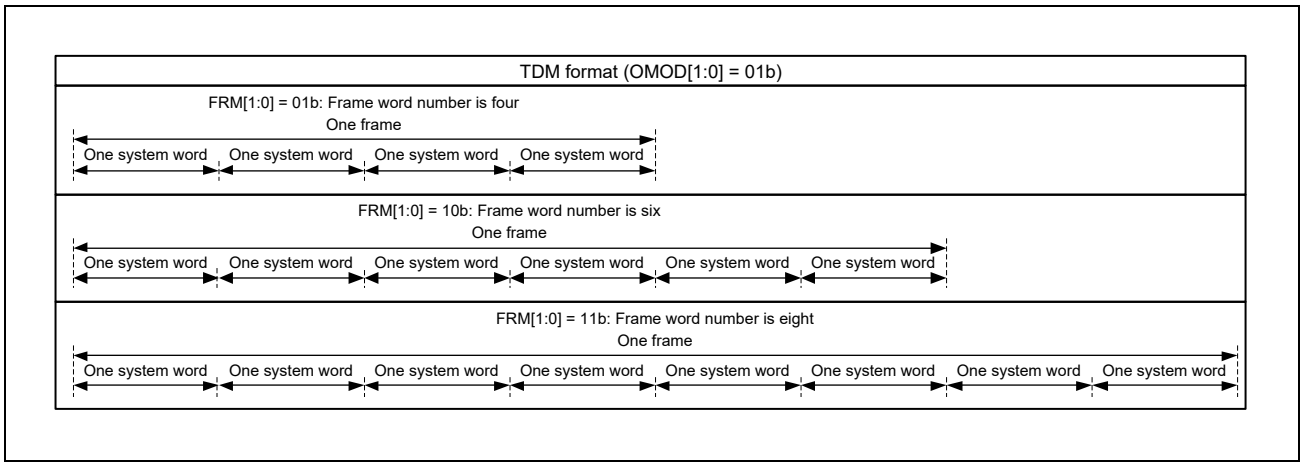


Figure 35.4 Frame Word Number

DWL[2:0] Bits

These bits set the number of bits in one data word. The setting whose number of data of system word lengths is shorter than that of the data word length is a prohibition. Refer to **Table 35.13** in **Section 35.4.2, Communication Formats** for details.

SWL[2:0] Bits

These bits set the number of bits in one system word. Padding bits are sent and received in relation with one data word set with DWL[2:0]. See **Table 35.13** in **Section 35.4.2, Communication Formats** for details.

Rewrite these bits when the LR clock supply to the terminal SSILRCK stops. Refer to the LRCONT bit details explanation in **Section 35.4.1.7, Audio Format Register (SSIOFR)** for the output operation of the LR clock.

MST Bit

This bit sets master-/slave-mode communication.

To write this bit, do so when AUDIO_MCK is not supplied. Refer to the AUCKE bit details explanation in **Section 35.4.1.3, FIFO Control Register (SSIFCR)** for detailed timing.

BCKP Bit

This bit sets the bit clock polarity (**Table 35.5**).

To write this bit, do so when AUDIO_MCK is not supplied. Refer to the AUCKE bit details explanation in **Section 35.4.1.3, FIFO Control Register (SSIFCR)** for detailed timing.

Table 35.5 Bit Clock Polarity

Communication	Master/Slave	Timing	BCKP = 0	BCKP = 1
Reception	Slave	At SSILRCK/SSIFS sampling	SSIBCK rising edge	SSIBCK falling edge
	Master/slave	At SSIFRxD sampling	SSIBCK rising edge	SSIBCK falling edge
Transmission	Master	At change of SSILRCK/SSIFS output	SSIBCK falling edge	SSIBCK rising edge
	Master/slave	At change of SSIFTxD output	SSIBCK falling edge	SSIBCK rising edge

LRCKP Bit

This bit sets the initial value and polarity of SSILRCK/SSIFS. Set this bit according to the communication format to be used in SSIF-2 (**Table 35.6**). Only the start trigger is used at slave communication (MST=0).

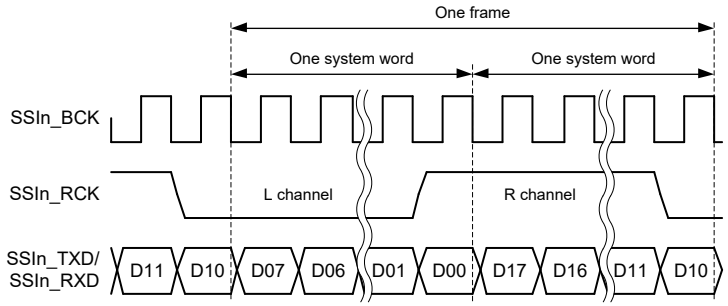
Rewrite this bit when the LR clock supply to the terminal SSILRCK stops. Refer to the LRCONT bit details explanation in **Section 35.4.1.7, Audio Format Register (SSIOFR)** for the output operation of the LR clock.

Table 35.6 Initial Output Value and Polarity of SSILRCK/SSIFS Pin

Communication Format	Expected Initial State	Setting Value of LRCKP
I^2S	H	0
Monaural	L	1
TDM	L	1

Note: Do the setting that can be communicated respectively to I^2S, monaural, and TDM at the compatible format.

System word length: 8 bits (SWL = 000b), Data word length: 8 bits (DWL = 000b),
LR clock polarity: L channel = low, R channel = high (LRCKP = 0)
Other control bits of communication format are at their initial values.



System word length: 8 bits (SWL = 000b), Data word length: 8 bits (DWL = 000b),
LR clock polarity: L channel = high, R channel = low (LRCKP = 1)
Other control bits of communication format are at their initial values.

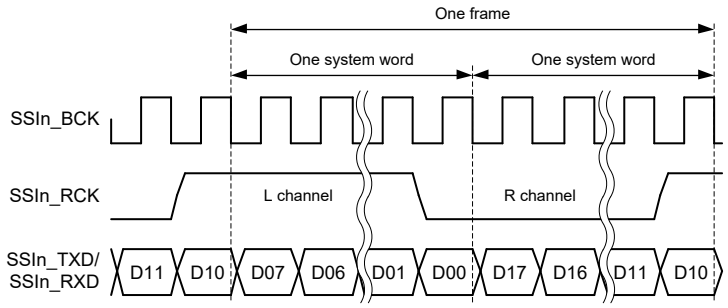


Figure 35.5 LRCK/FS Polarity Setting

SPDP Bit

This bit sets polarity of padding bits.

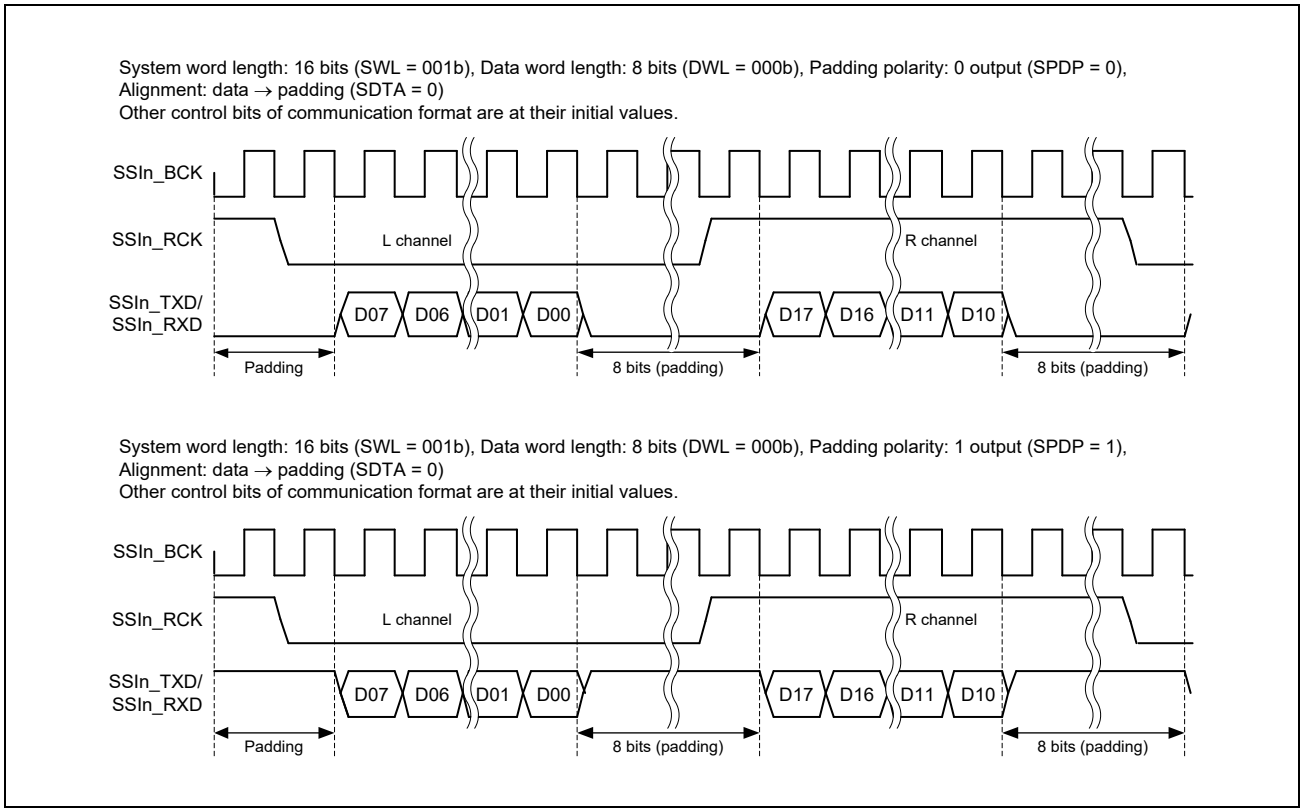


Figure 35.6 Padding Bit Polarity

SDTA Bit

This bit sets alignment of serial data (data bits and padding bits). For communication without padding bits, this bit is invalid.

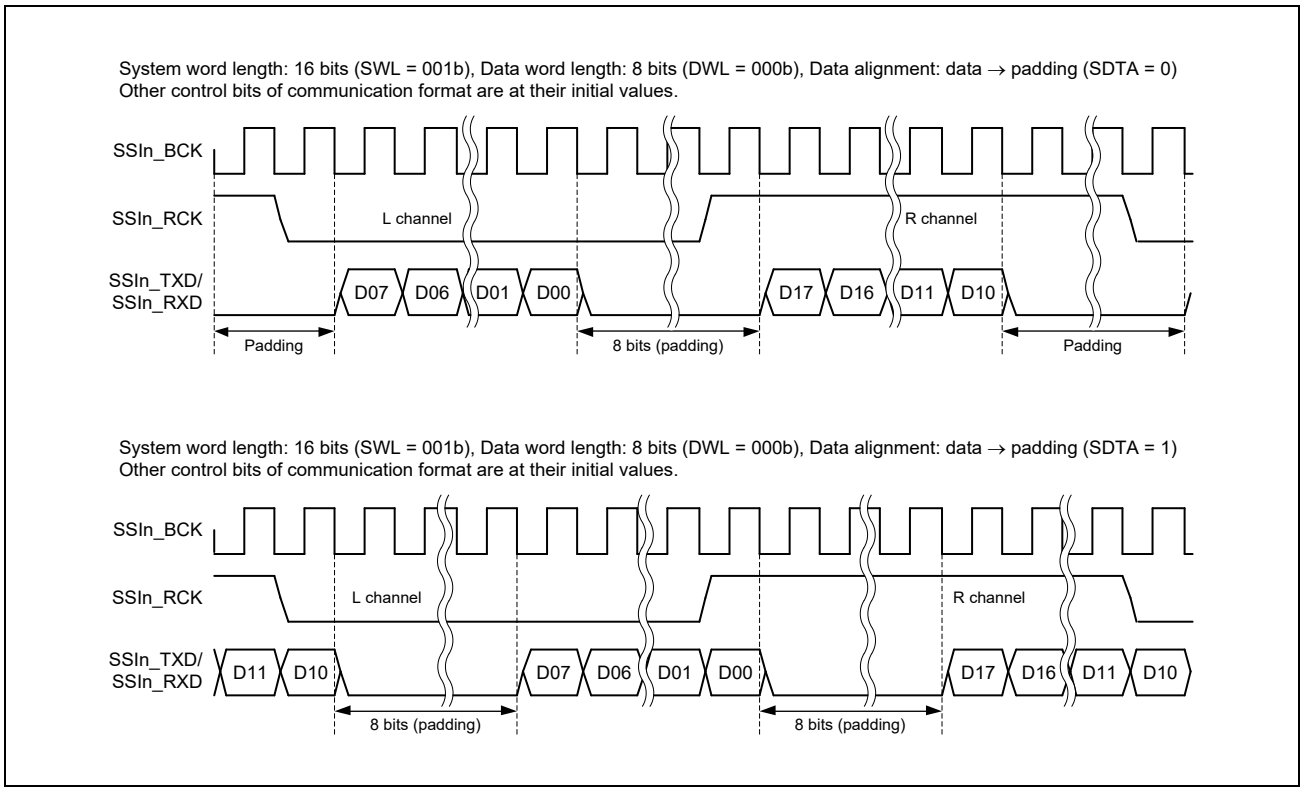


Figure 35.7 Alignment Setting of Serial Data with Padding Bits

PDTA Bit

This bit sets alignment of placement data. With the setting of data word length as 32 bits (DWL[2:0] = 110b), this bit is invalid.

- At transmission:

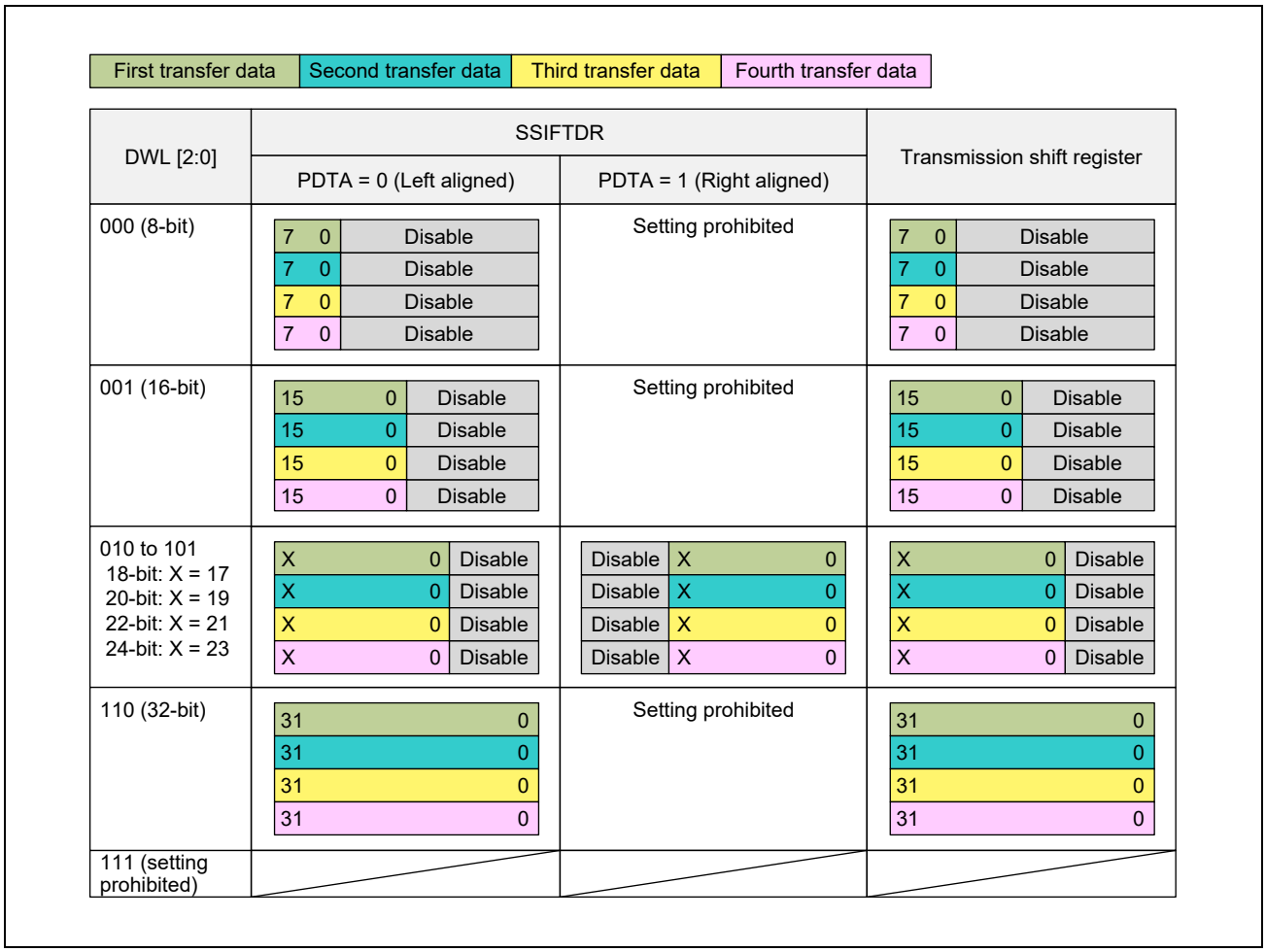


Figure 35.8 Alignment Setting of Placement Data in Transmission

- At reception:

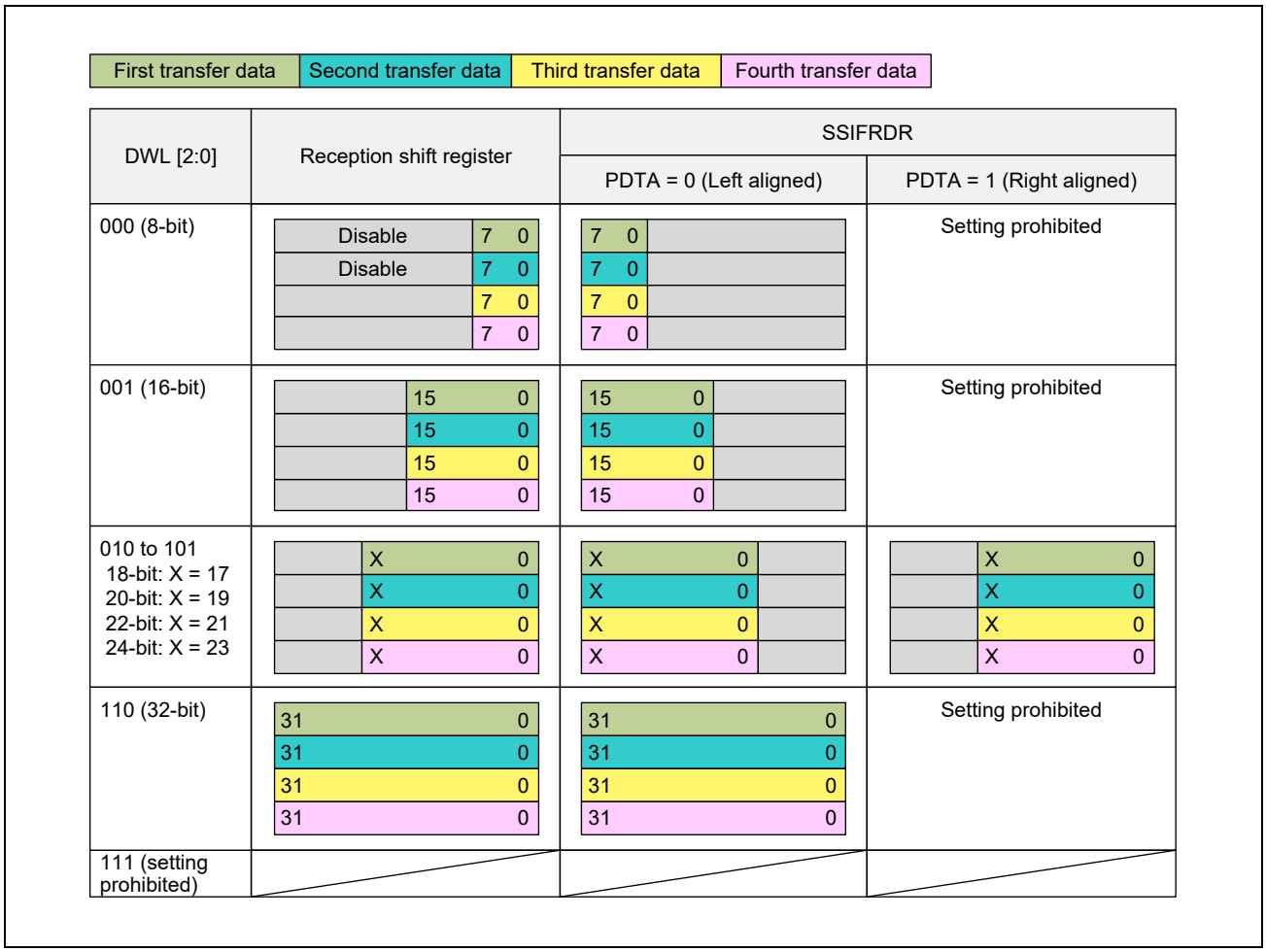


Figure 35.9 Alignment Setting of Placement Data in Reception

DEL Bit

This bit sets whether or not there will be a delay between SSILRCK/SSIFS and SSITxD/SSIRxD.

With the I²S format and TDM format, set 0 to DEL (**Figure 35.10**). With the monaural format, this bit changes the high period width. Refer to **Section 35.4.2.2, Monaural Format** for details. Do the setting that can be communicated at the compatible communication format.

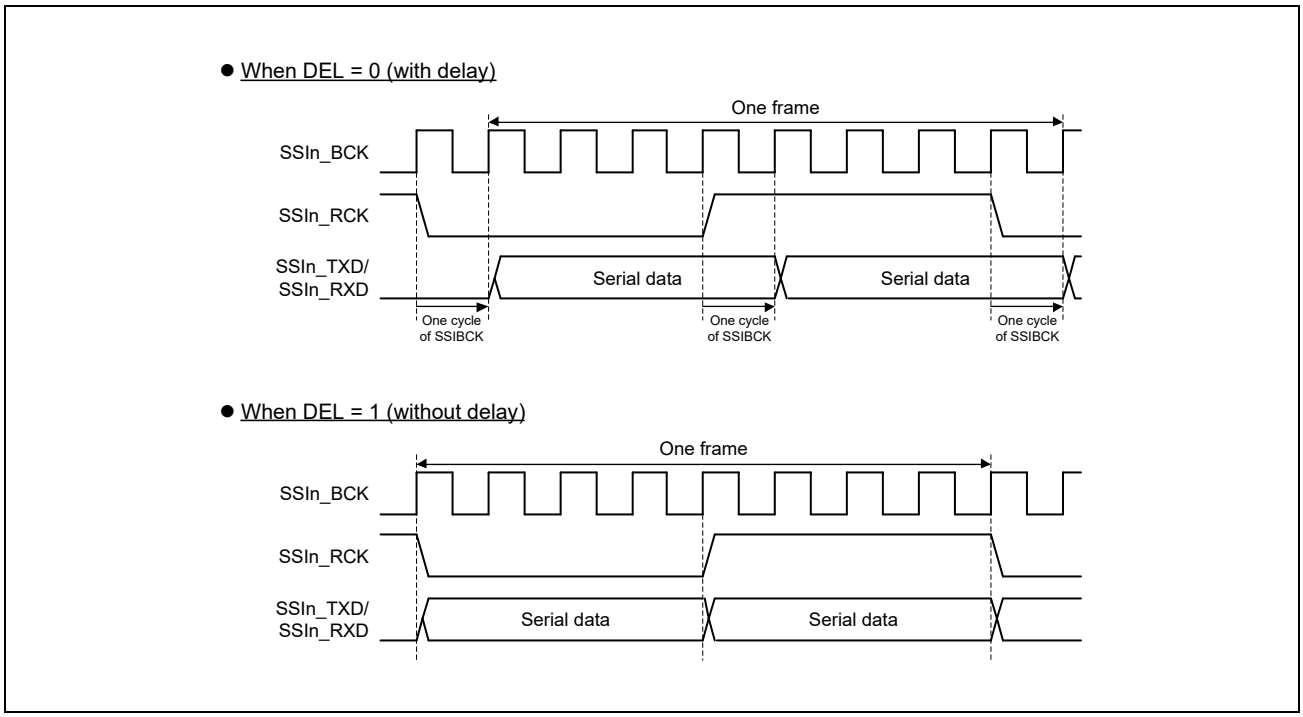


Figure 35.10 Setting of Delay in Serial Data

CKDV[3:0] Bits

These bits set the division ratio of the bit clock based on AUDIO_MCK in master-mode communication (MST = 1) (**Figure 35.11**). In slave-mode communication (MST = 0), setting of these bits are invalid.

To write this bit, do so when AUDIO_MCK is not supplied. Refer to the AUCKE bit details explanation in **Section 35.4.1.3, FIFO Control Register (SSIFCR)** for detailed timing.

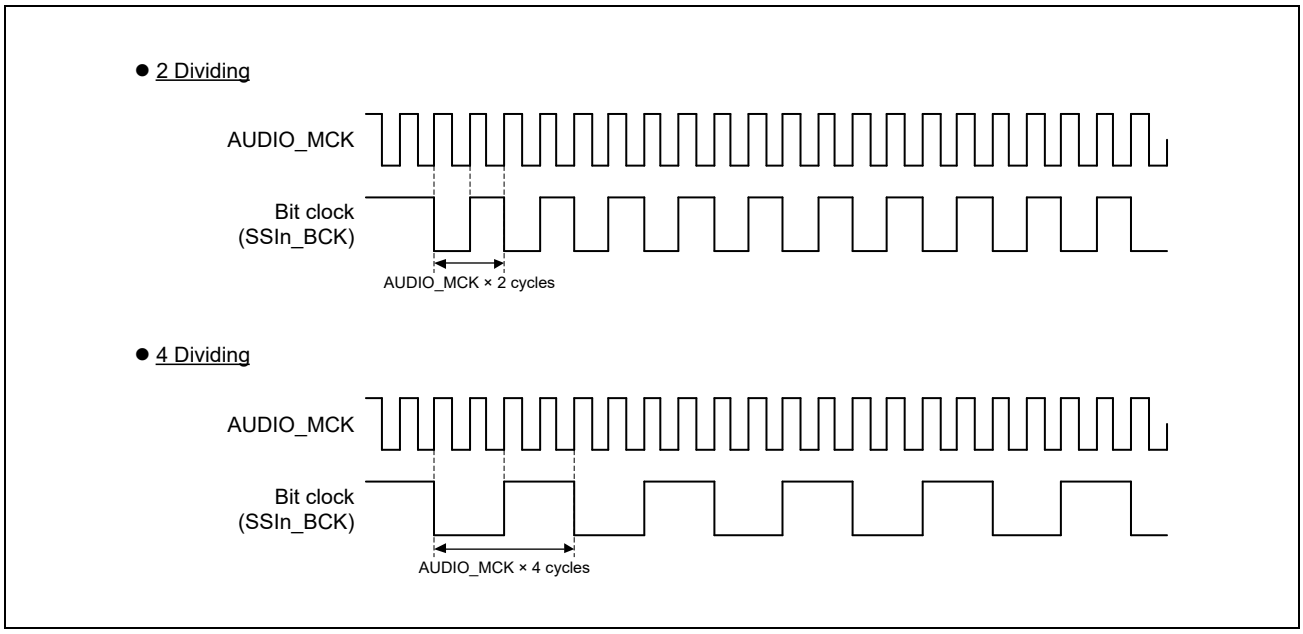


Figure 35.11 Sampling Frequencies in Master-mode Communication

MUEN Bit

This bit sets/clears the mute function for the data output from the terminal SSITxD. When you set this bit to 1, the output value of SSITxD becomes 0 from the next frame boundary. When you set this bit to 0, the output value of SSITxD can be made the data of the FIFO transmission data register from the following frame boundary.

Note that this bit controls data only. Status flags and interrupt signals are normally generated. Set this bit after setting completing the communication format.

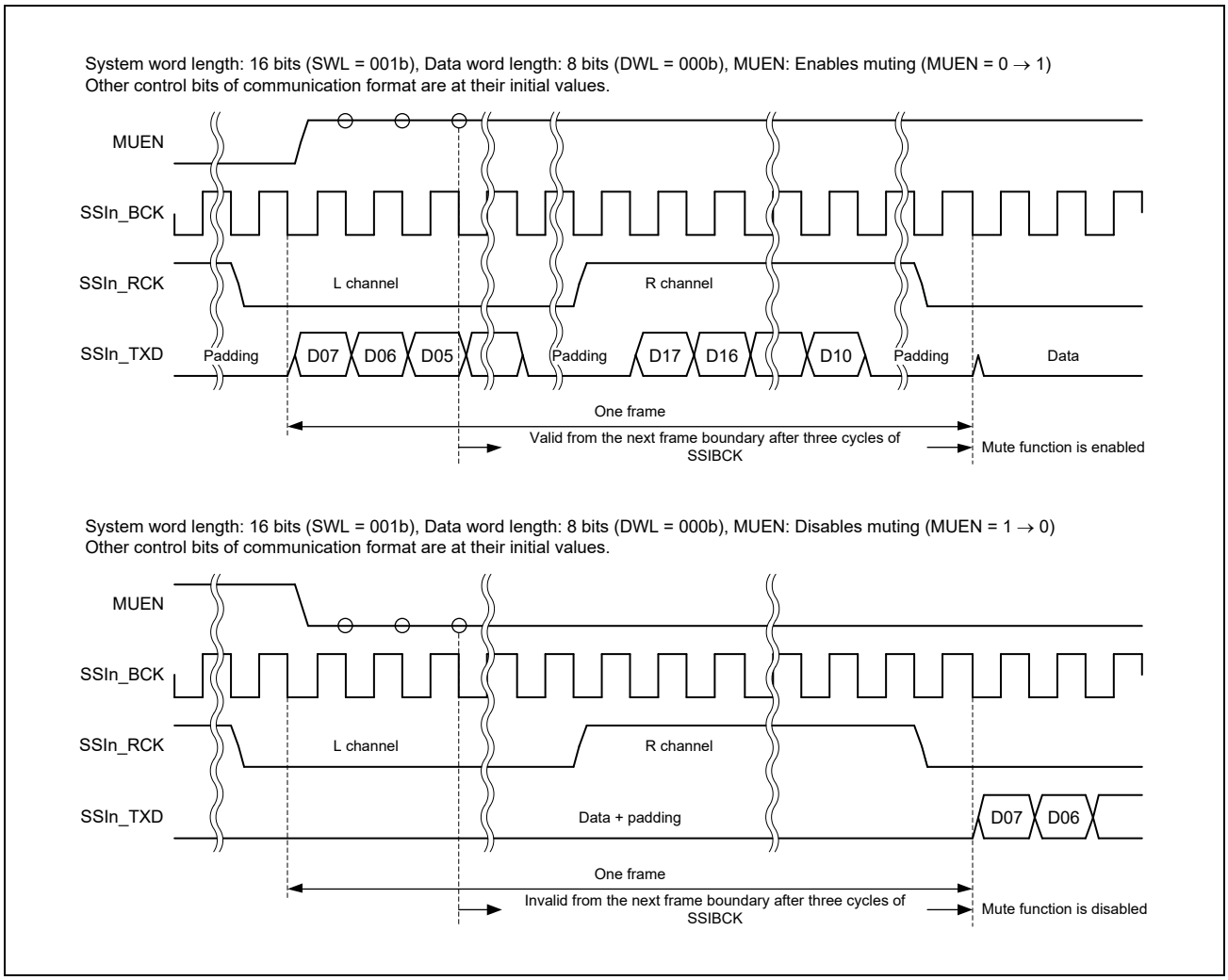


Figure 35.12 Transmit Data with the Mute Function Set

TEN and REN Bits

These bits enable/disable transmission and reception. Writing 1 to these bits, transmission starts in synchronization with start trigger. Refer to **Section 35.5.2.2, Transmission**, **Section 35.5.2.3, Reception**, and **Section 35.5.2.4, Transmission and Reception** for details. Writing 0 to these bits, transmission stops in synchronization with next frame boundary. Set 1 at the same time when you use SSIF-2 as transmission and reception. To stop communication, make sure to stop transmission and reception (TEN = 0 and REN = 0).

To stop communication without waiting for the frame boundary, follow the software reset procedure.

35.4.1.2 Status Register (SSISR)

This is a readable/writable 32-bit register. It is configured with status flags that indicate SSIF-2 operational state.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W0	R/W0	R/W0	R/W0	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00b	R	Reserved. Write 0. The read value is 0.
29	TUIRQ	0	RW0	Transmit underflow error status flag 0: No transmit underflow error is generated. 1: A transmit underflow error is generated.
28	TOIRQ	0	RW0	Transmit overflow error status flag 0: No transmit overflow error is generated. 1: A transmit overflow error is generated.
27	RUIRQ	0	RW0	Receive underflow error status flag 0: No receive underflow error is generated. 1: A receive underflow error is generated.
26	ROIRQ	0	RW0	Receive overflow error status flag 0: No receive overflow error is generated. 1: A receive overflow error is generated.
25	IIRQ	1	R	Idle mode status flag 0: In the communication state 1: In the idle state
24 to 0	—	H'000_0000	R	Reserved. Write 0. The read value is 0.

TUIRQ Bit

This is a status flag that indicates a transmit underflow error. This flag is set by automatic determination but it must be cleared by register access. This flag indicates that writing serial data necessary for one frame to SSIFTDR was slower than one frame of the transmission operation. Even if this flag is cleared after this flag becomes '1', the output value of SSIFTxD is 0. Follow the communication stop procedure (**Figure 35.55**) and the error procedure (**Figure 35.56**) to output the data written in transmission FIFO data register (SSIFTDR) to the terminal SSIFTxD. See **Section 35.5.2.6, Error Handling** for the error recovery procedure. This flag is not cleared by transmission FIFO data register reset (SSIFCR.TFRST).

[Priority order for setting and clearing]

Setting is prioritized.*1

[Clearing condition]

When either of the following is approved.

(1) Writing 0 to this flag after reading this flag as 1.*2

(2) When you do the communication permission (SSICR.TEN is changed from 0 to 1)

[Clearing timing]

Timing according to the above-mentioned clear condition.

(1) At completion of writing 0 to this flag after reading this flag as 1 (**Figure 35.13**).

(2) Write 1 in SSICR.TEN, complete, and after one cycle of P0ϕ (**Figure 35.13**).

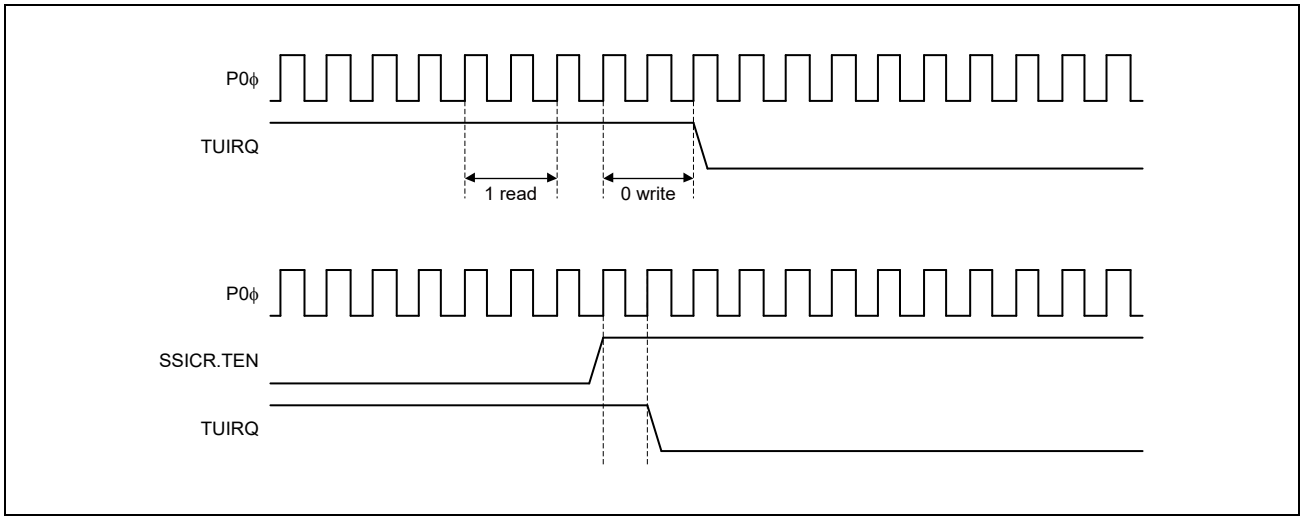


Figure 35.13 TUIRQ Clearing Timing

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1). The software reset has priority over all the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following three conditions being met:
- Software reset (SSIFCR.SSIRST=1).
- Completion of writing 0 to this flag after having read it as 1.
- One cycle of P0ϕ elapsing after the completion of writing 1 to SSICR.TEN.

Remark After the communication is permitted (SSICR.TEN is written and '1' is written from '0'), the transmission error flag (TOIRQ and TUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the transmission error flag might not be able to be read.

[Setting condition]

When data to be transmitted in the next frame is not fully written to SSIFTDR at the frame boundary of continuance communication.

[Setting timing]

After three cycles of P0ϕ from frame boundary (**Figure 35.14**).

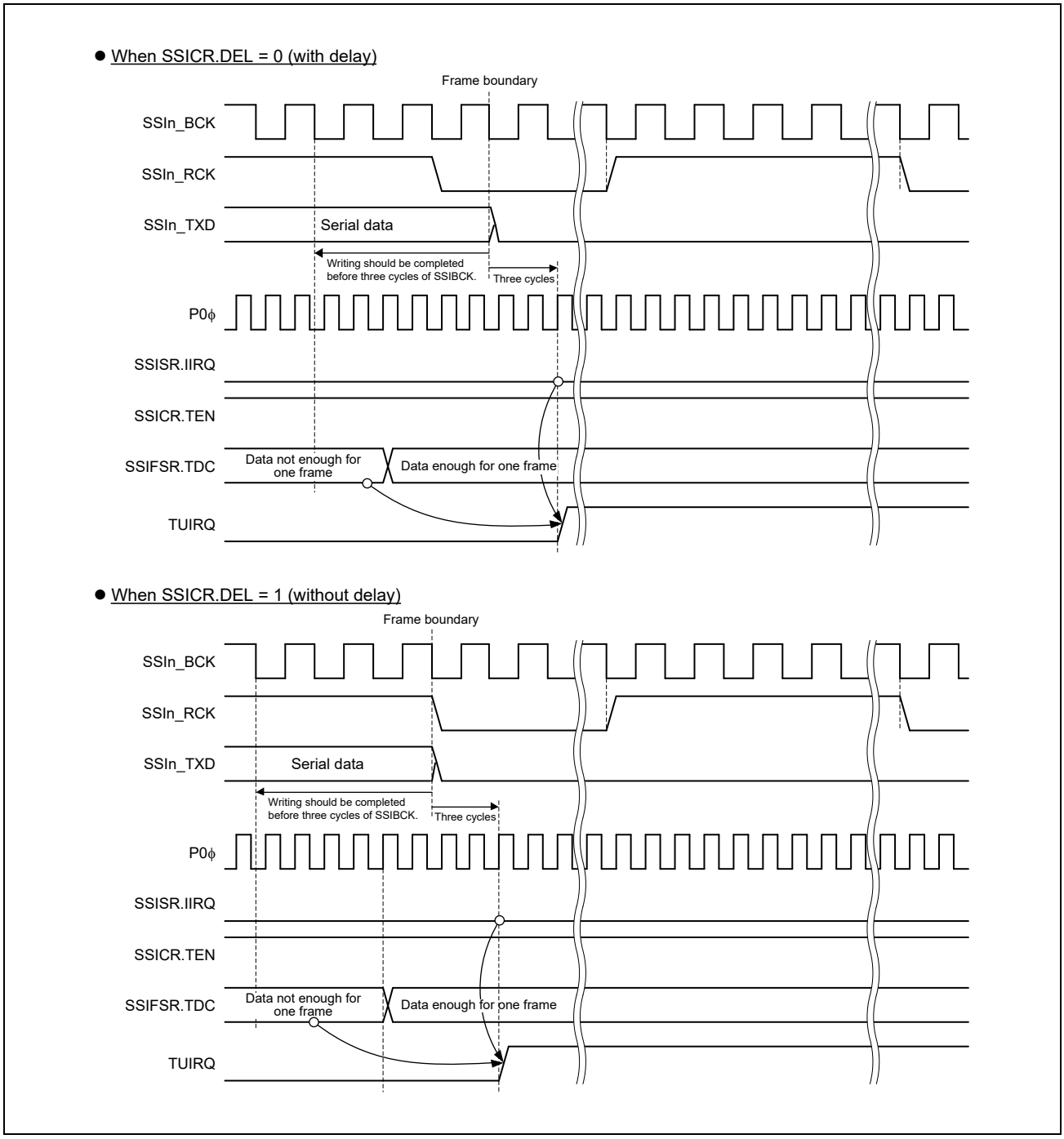


Figure 35.14 TUIRQ Setting Timing (when you continue communicating)

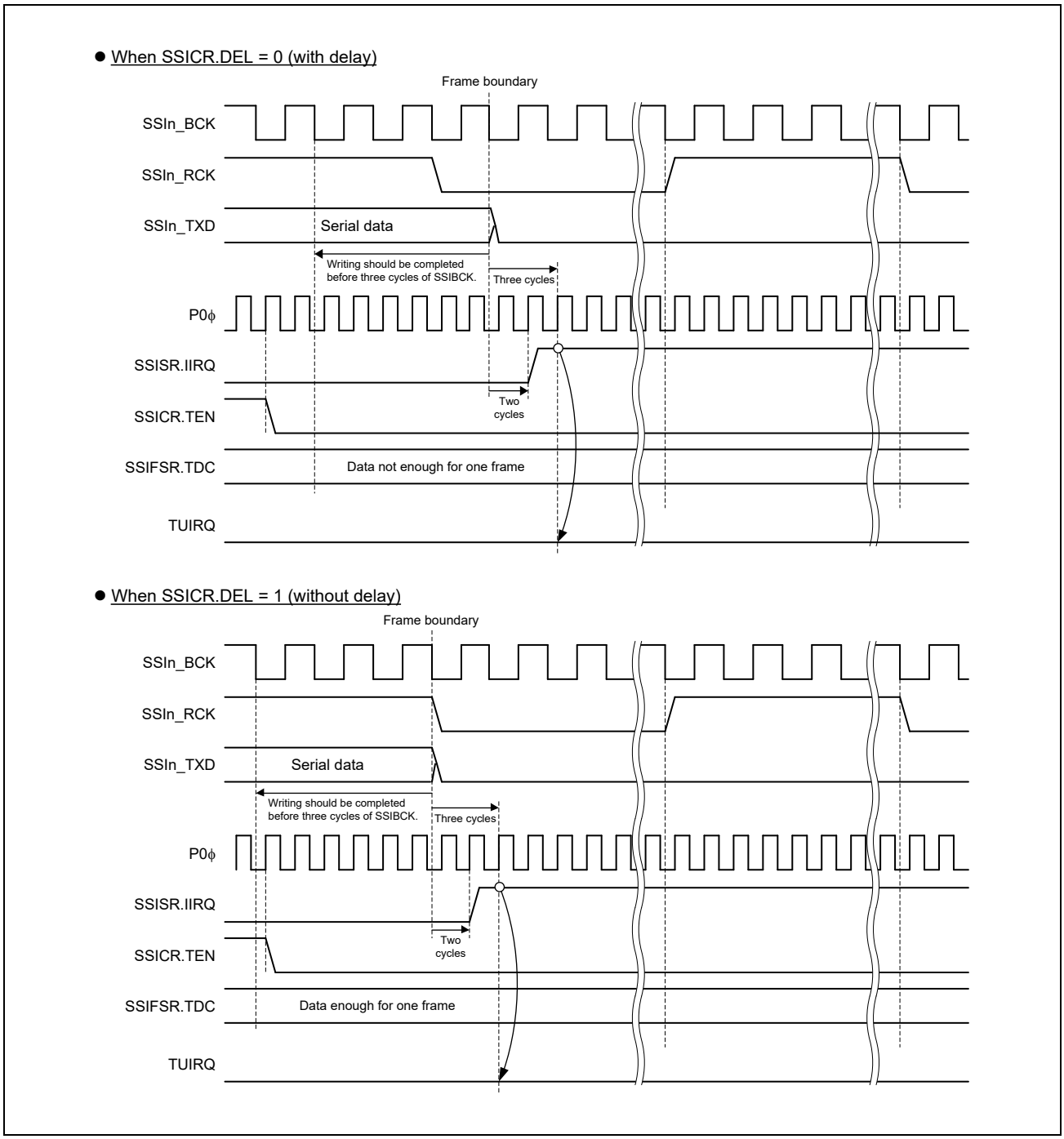


Figure 35.15 TUIRQ Setting Timing (when you stop communicating)

TOIRQ Bit

This is a status flag that indicates a transmit overflow error. This flag is set by automatic determination but it must be cleared by register access. This flag indicates that serial data is written to SSIFTDR when it is full during transmission (SSICR.TEN=1). Data is not written to SSIFTDR where a transmit overflow error is generated. See **Section 35.5.2.6, Error Handling** for the error recovery procedure. This flag is not cleared by transmission FIFO data register reset (SSIFCR.TFRST).

[Priority order for setting and clearing]

Setting is prioritized.*1

[Clearing condition]

When either of the following is approved.

(1) Writing 0 to this flag after reading this flag as 1.*2

(2) When you do the communication permission (SSICR.TEN is changed from 0 to 1)

[Clearing timing]

Timing according to the above-mentioned clear condition.

(1) At completion of writing 0 to this flag after reading this flag as 1 (same as **Figure 35.13**).

(2) Write 1 in SSICR.TEN, complete, and after one cycle of P0ϕ (same as **Figure 35.13**).

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1). The software reset has priority over all the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following three conditions being met:
- Software reset (SSIFCR.SSIRST=1).
- Completion of writing 0 to this flag after having read it as 1.
- One cycle of P0ϕ elapsing after the completion of writing 1 to SSICR.TEN.

Remark After the communication is permitted (SSICR.TEN is written and '1' is written from '0'), the transmission error flag (TOIRQ and TUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the transmission error flag might not be able to be read.

[Setting condition]

During transmission operation (SSICR.TEN = 1), data is written to SSIFTDR while it is full.

[Setting timing]

At completion of writing to SSIFTDR (**Figure 35.16**)

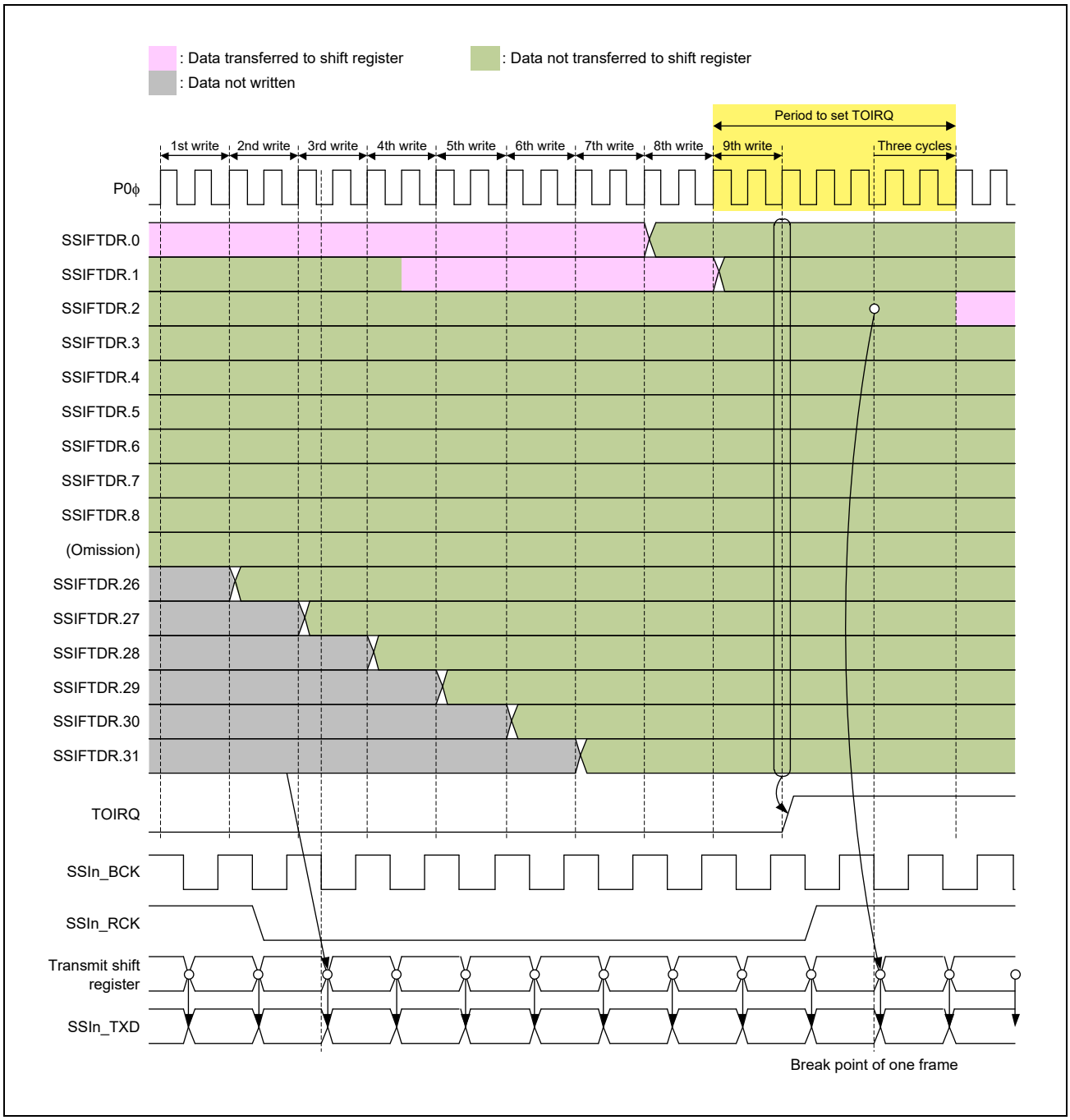


Figure 35.16 TOIRQ Setting Timing

RUIRQ Bit

This is a status flag that indicates a receive underflow error. This flags is set by automatic determination but it must be cleared by register access. This flag indicates that SSIFRDR is read while it is empty. Data read from SSIFRDR where a receive underflow error is generated is invalid. See **Section 35.5.2.6, Error Handling** for the error recovery procedure. This flag is not cleared by reception FIFO data register reset (SSIFCR.RFRST). Even if SSIFRDR is read to (SSIFCR.RFRST=1) while resetting reception FIFO data register, it doesn't set 1.

[Priority order for setting and clearing]

Setting is prioritized.*1

[Clearing condition]

When either of the following is approved.

(1) Writing 0 to this flag after reading this flag as 1.*2

(2) When you do the communication permission (SSICR.REN is changed from 0 to 1)

[Clearing timing]

Timing according to the above-mentioned clear condition.

(1) At completion of writing 0 to this flag after reading this flag as 1 (same as **Figure 35.13**).

(2) Write 1 in SSICR.REN, complete, and after one cycle of P0ϕ (same as **Figure 35.13**).

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1). The software reset has priority over all the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following three conditions being met:
- Software reset (SSIFCR.SSIRST=1).
- Completion of writing 0 to this flag after having read it as 1.
- One cycle of P0ϕ elapsing after the completion of writing 1 to SSICR.TEN.

Remark After the communication is permitted (SSICR.REN is written and '1' is written from '0'), the transmission error flag (ROIRQ and RUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the reception error flag might not be able to be read.

[Setting condition]

Reading from SSIFRDR while it is empty.

[Setting timing]

At completion of reading from SSIFRDR (**Figure 35.17**).

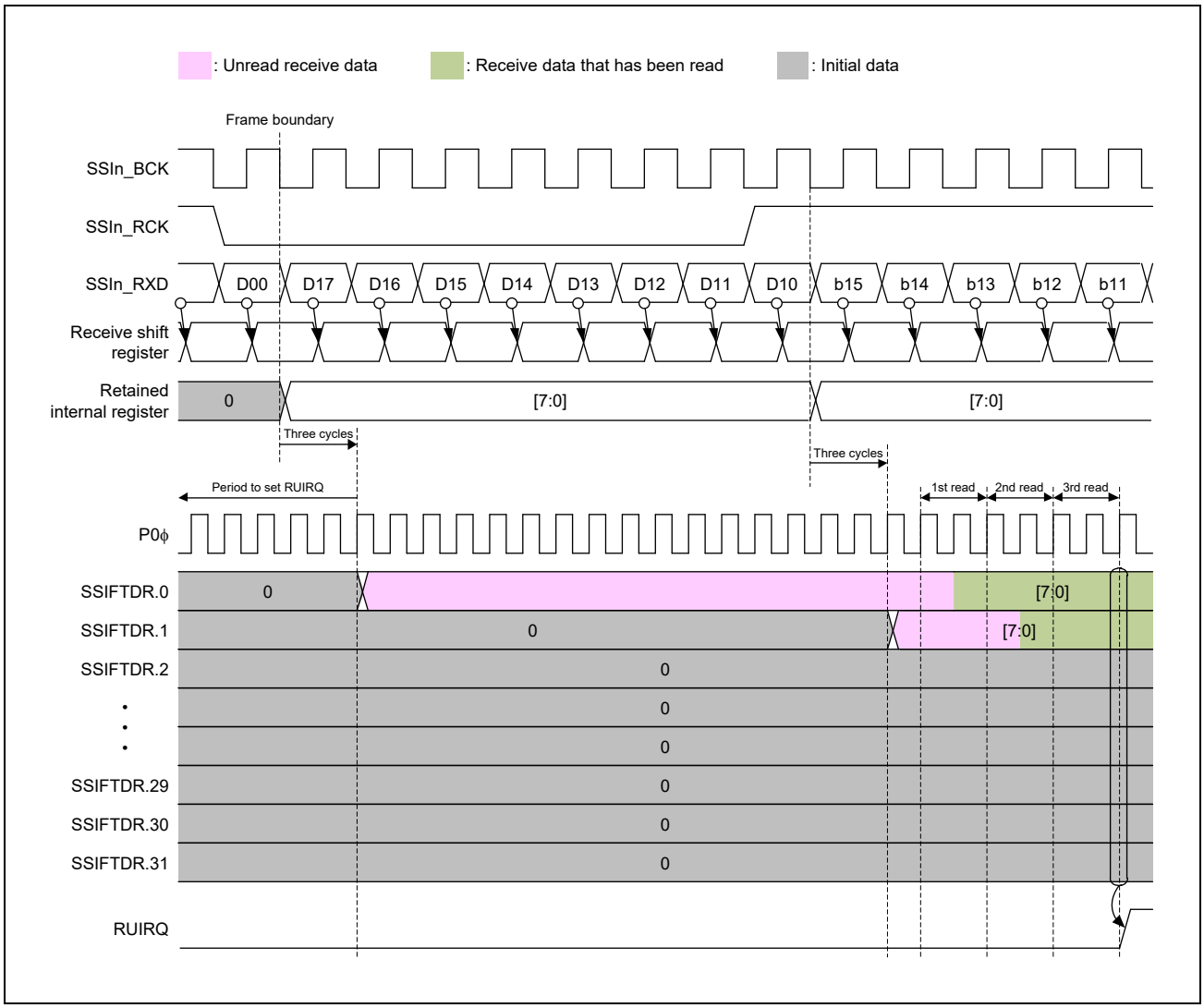


Figure 35.17 RUIRQ Setting Timing

ROIRQ Bit

This is a status flag that indicates a receive overflow error. This flags is set by automatic determination but it must be cleared by register access. This flag indicates that received data is supplied at a higher rate than requested. Data is not transferred from the receive shift register to SSIFRDR where a receive overflow error is generated. See **Section 35.5.2.6, Error Handling** for the error recovery procedure. This flag is not cleared by reception FIFO data register reset (SSIFCR.RFRST).

[Priority order for setting and clearing]

Setting is prioritized.

[Clearing condition]

Writing 0 to this flag after reading this flag as 1.

[Clearing timing]

At completion of writing 0 after reading 1 from an SFR (same as **Figure 35.13**).

[Priority order for setting and clearing]

Setting is prioritized.*1

[Clearing condition]

When either of the following is approved.

(1) Writing 0 to this flag after reading this flag as 1.*2

(2) When you do the communication permission (SSICR.REN is changed from 0 to 1)

[Clearing timing]

Timing according to the above-mentioned clear condition.

(1) At completion of writing 0 to this flag after reading this flag as 1 (same as **Figure 35.13**).

(2) Write 1 in SSICR.REN, complete, and after one cycle of P0ϕ (same as **Figure 35.13**).

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1). The software reset has priority over all the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following three conditions being met:
- Software reset (SSIFCR.SSIRST=1).
- Completion of writing 0 to this flag after having read it as 1.
- One cycle of P0ϕ elapsing after the completion of writing 1 to SSICR.TEN.

Remark After the communication is permitted (SSICR.REN is written and '1' is written from '0'), the transmission error flag (ROIRQ and RUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the reception error flag might not be able to be read.

[Setting condition]

At completion of receiving new data while SSIFRDR is full.

[Setting timing]

Three cycles of P0ϕ after reception is completed (**Figure 35.18**).

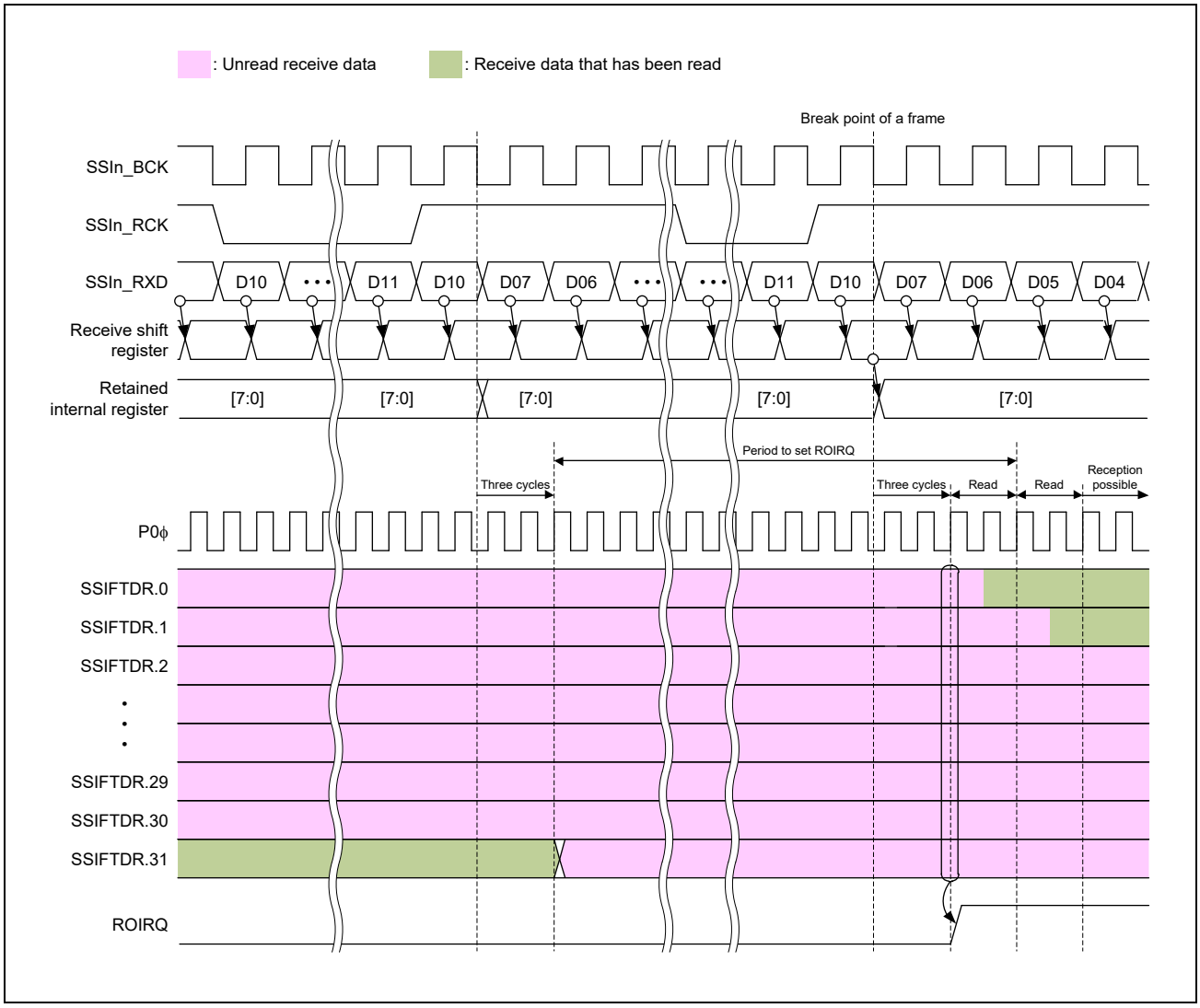


Figure 35.18 ROIRQ Setting Timing

IIRQ Bit

This is a status flag that indicates the idle state. It indicates whether SSIF-2 is in the idle state or communication state (**Figure 35.19** and **Figure 35.20**).

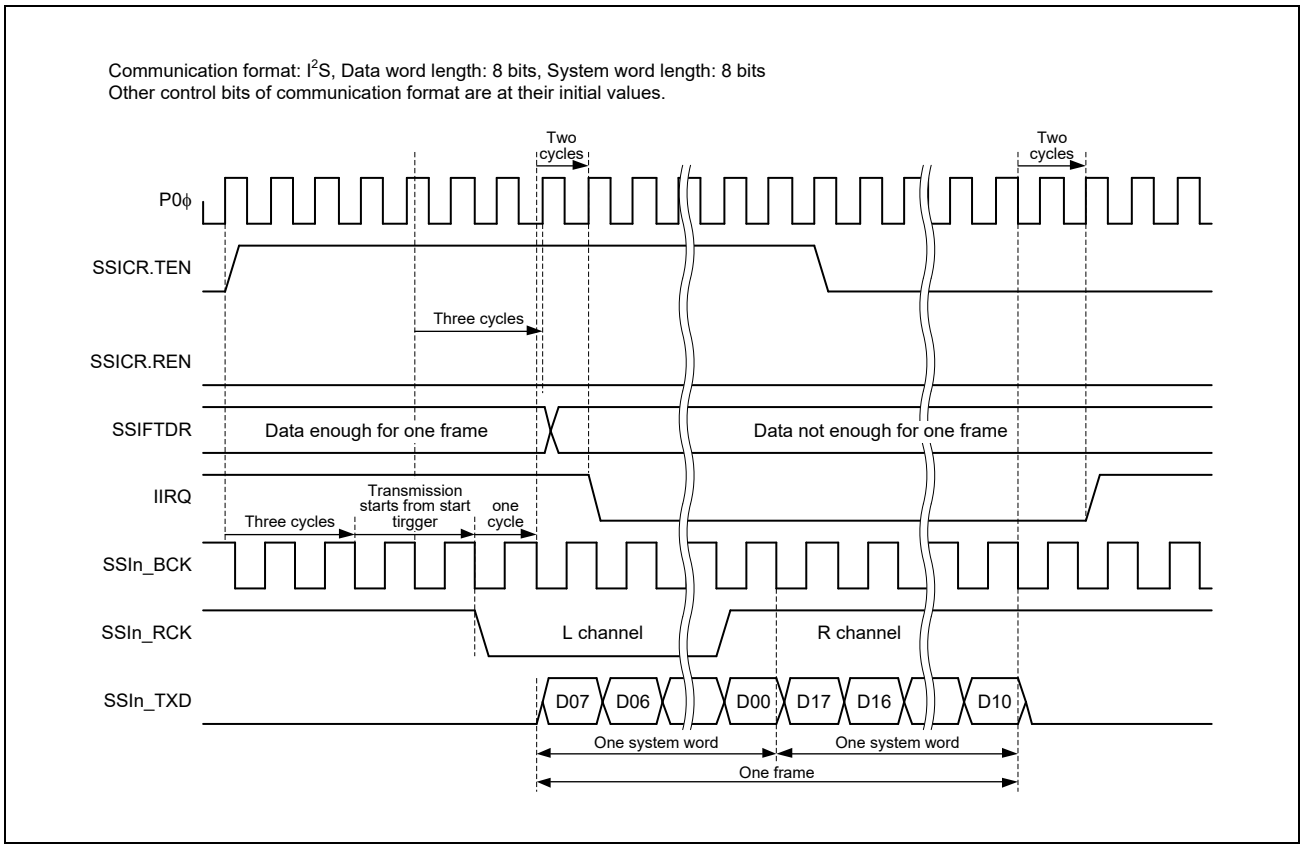


Figure 35.19 IIRQ Setting Timing (Transmission)

Transmitter (dedicated to transmission)

[Clearing condition]

While transmission is enabled (SSICR.TEN = 1 and SSICR.REN = 0), one frame of data to be transmitted has been written to SSIFTDR, and a start trigger is generated in SSILRCK/SSIFS.

[Clearing timing]

Two cycles of P0ϕ after one cycle of SSIBCK where a start trigger is generated.

[Setting condition]

One frame of data has been transmitted while transmission and reception are disabled (SSICR.TEN = 0 and SSICR.REN = 0).

[Setting timing]

Two cycles of P0ϕ after transmission is complete according to the setting condition (frame boundary).

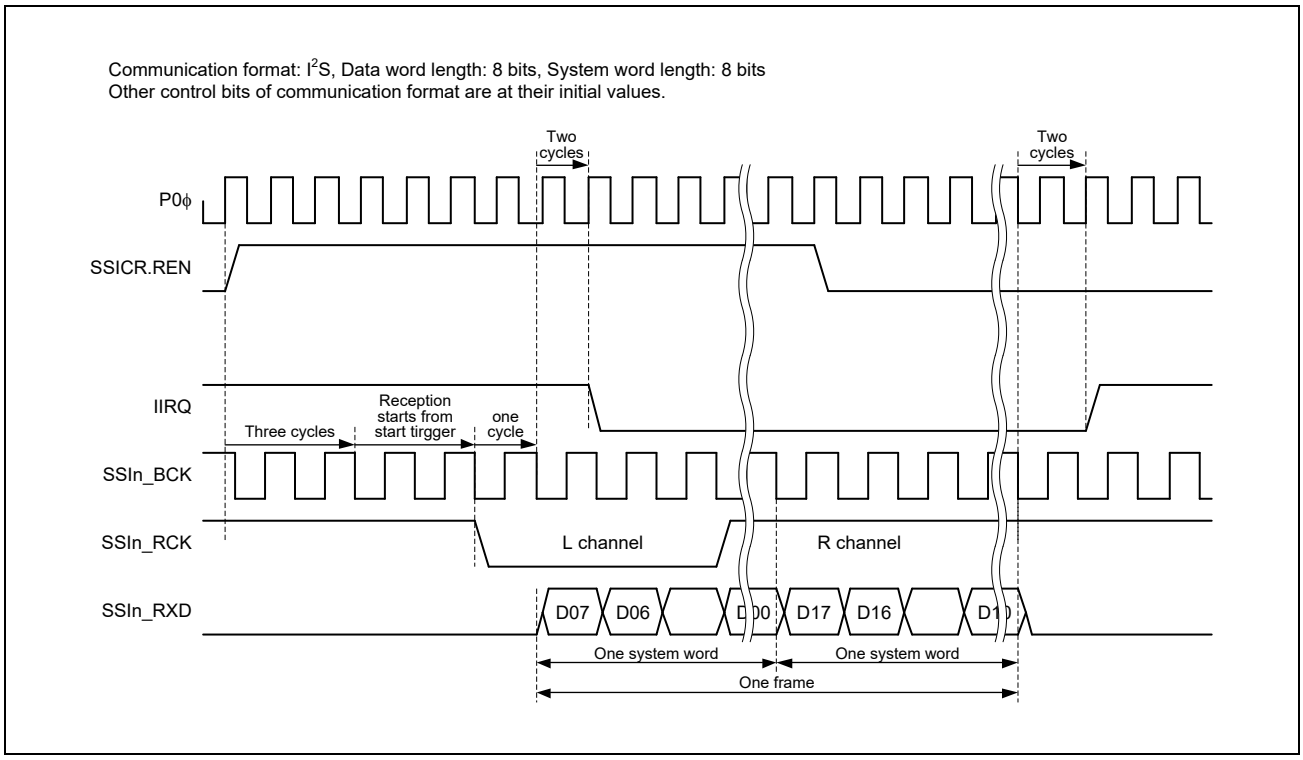


Figure 35.20 IIRQ Setting Timing (Reception)

Receiver (dedicated to reception)

[Clearing condition]

While reception is enabled (SSICR.TEN = 0 and SSICR.REN = 1), and a start trigger is generated in SSILRCK/SSIFS.

[Clearing timing]

Two cycles of P0ϕ after one cycle of SSIBCK when a start trigger of the clearing condition is generated.

[Setting condition]

One frame of data has been received while transmission and reception are disabled (SSICR.TEN = 0 and SSICR.REN = 0).

[Setting timing]

Two cycles of P0ϕ after reception of the setting condition is completed (frame boundary).

Transceiver (transmission and reception)

[Clearing condition]

While transmission and reception are enabled (SSICR.TEN = 1 and SSICR.REN = 1), one frame of data to be transmitted has been written to SSIFTDR, and a start trigger is generated in SSILRCK/SSIFS.

[Clearing timing]

Two cycles of P0ϕ after one cycle of SSIBCK when a start trigger of the clearing condition is generated.

[Setting condition]

One frame of data has been transmitted while transmission and reception are disabled (SSICR.TEN = 0 and SSICR.REN = 0).

[Setting timing]

Two cycles of P0ϕ after transmission is completed according to the setting condition (frame boundary).

35.4.1.3 FIFO Control Register (SSIFCR)

This is a readable/writable 32-bit register. It sets a software reset, byte swap, noise canceller, and enable/disable of interrupt requests.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	BSW	BCKNCE	LRCKNCE	RXDNCE	—	—	—	—	TIE	RIE	TFRST	RFRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	AUCKE	0	RW	AUDIO_MCK Enable in Master-mode Communication*1 0: Disables supply of AUDIO_MCK 1: Enables supply of AUDIO_MCK
30 to 17	—	H'0000	R	Reserved. Write 0. The read value is 0.
16	SSIRST	0	RW	Software Reset 0: Clears a software reset condition 1: Sets a software reset condition
15 to 12	—	H'0	R	Reserved. Write 0. The read value is 0.
11	BSW	0	RW	Byte Swap Enable*1 0: Disables byte swap 1: Enables byte swap
10	BCKNCE	0	RW	Noise Canceller Enable in Slave-mode Communication (SSIBCK)*1*2 0: Disables (bypasses) the noise canceller 1: Enables the noise canceller
9	LRCKNCE	0	RW	Noise Canceller Enable in Slave-mode Communication (SSILRCK/ SSIFS)*1*2 0: Disables (bypasses) the noise canceller 1: Enables the noise canceller
8	RXDNCE	0	RW	Received Data Input Noise Canceller Enable in Slave-mode Communication (SSIRxD)*1*2 0: Disables (bypasses) the noise canceller 1: Enables the noise canceller
7 to 4	—	H'0	R	Reserved. Write 0. The read value is 0.
3	TIE	0	RW	Transmit Data Empty Interrupt Output Enable 0: Disables transmit data empty interrupts 1: Enables transmit data empty interrupts
2	RIE	0	RW	Receive Data Full Interrupt Output Enable 0: Disables receive data full interrupts 1: Enables receive data full interrupts
1	TFRST	0	RW	Transmit FIFO data register reset*1 0: Clears a transmit data FIFO reset condition 1: Sets a transmit data FIFO reset condition
0	RFRST	0	RW	Receive FIFO data register reset*1 0: Clears a receive data FIFO reset condition 1: Sets a receive data FIFO reset condition

Note 1. Writing to these bits while SSIF-2 is in a communication state (SSISR.IIRQ = 0) is prohibited. If written, the operation performed immediately after writing is not guaranteed.

Note 2. This bit can only be set to 1 in slave-mode communication (SSICR.MST = '0') except in the case of transmission (SSICR.TEN = '1') while SSICR.DEL = '1'. Be sure to clear this bit to 0 in master-mode communication (SSICR.MST = '1') or during transmission in slave mode (SSICR.MST = '0', SSICR.TEN = '1') while SSICR.DEL = '1'. Set the BCKNCE, LRCKNCE, and RXDNCE bits to the same value.

AUCKE Bit

This bit enables/disables supply to AUDIO_MCK while in master-mode communication (MST = 1). Select AUDIO_MCK (SSICR.CKS bit) before writing 1 to this bit.

Rewrite this bit after completing the setting (CKS, MST, BCKP, and CKDV of the SSICR register) that relates to AUDIO_MCK.

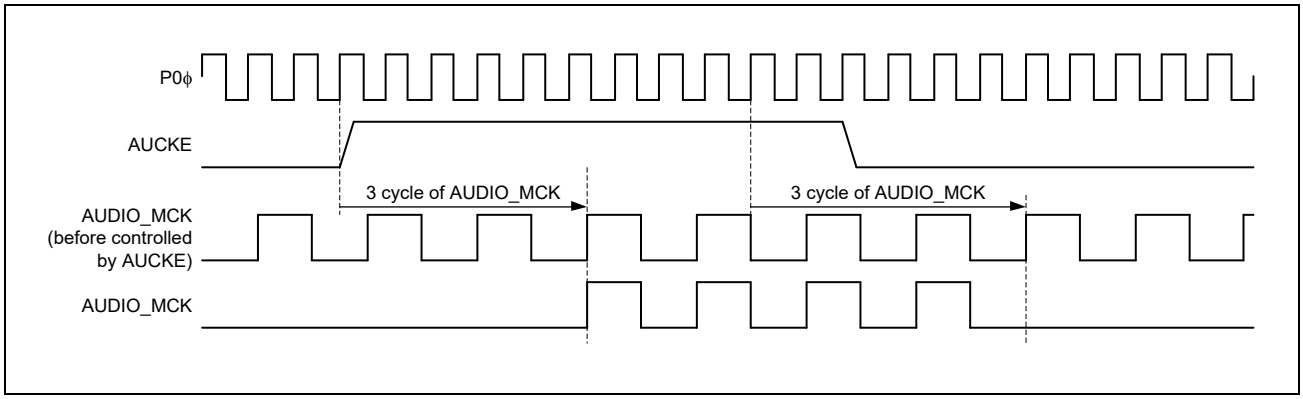


Figure 35.21 Stop/Resume of AUDIO_MCK

CAUTION

- In slave-mode communication (SSICR.MST = 0), SSIF-2 needs supply of SSIBCK. To stop BCK on the master side, make sure that SSIF-2 is in the idle state (SSISR.IIRQ = 1). If BCK is stopped before SSIF-2 becomes idle, take the procedure to start communication (**Figure 35.52**) or wait for an idle state by taking the procedure to resume communication (**Figure 35.57**).
- In master-mode communication (SSICR.MST = 1), SSIF-2 operates with the audio clock (AUDIO_MCK). To stop SSIF-2 completely, make sure that SSIF-2 is in the idle state (SSISR.IIRQ = 1) and then write 0 to SSIFCR.ADCKE. If 0 is written to SSIFCR.ADCKE before SSIF-2 becomes idle, take the procedure to start communication (**Figure 35.52**) or wait for the idle state by taking the procedure to resume communication (**Figure 35.57**).

Figure 35.22 and **Figure 35.23** show timing until outputting it to the terminal SSIBCK after this bit is set to ‘1’.

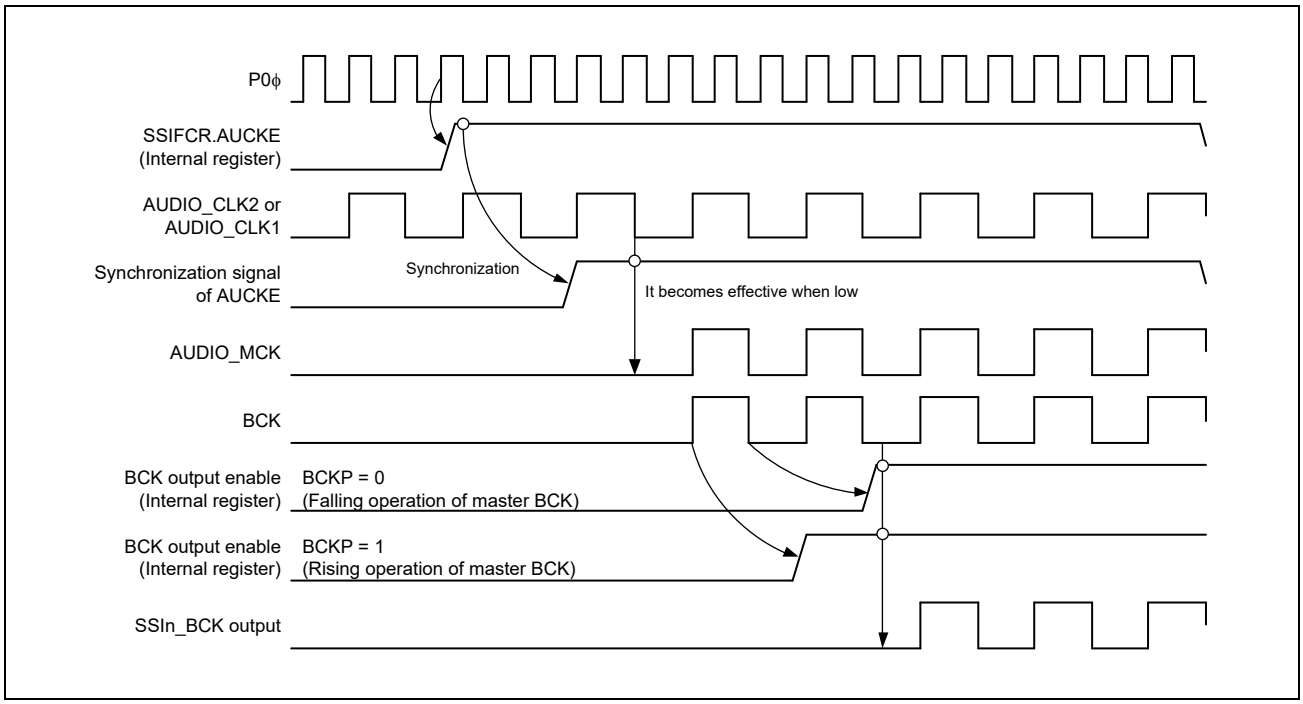


Figure 35.22 Timing Chart when It begins to Communicate Master from System Reset

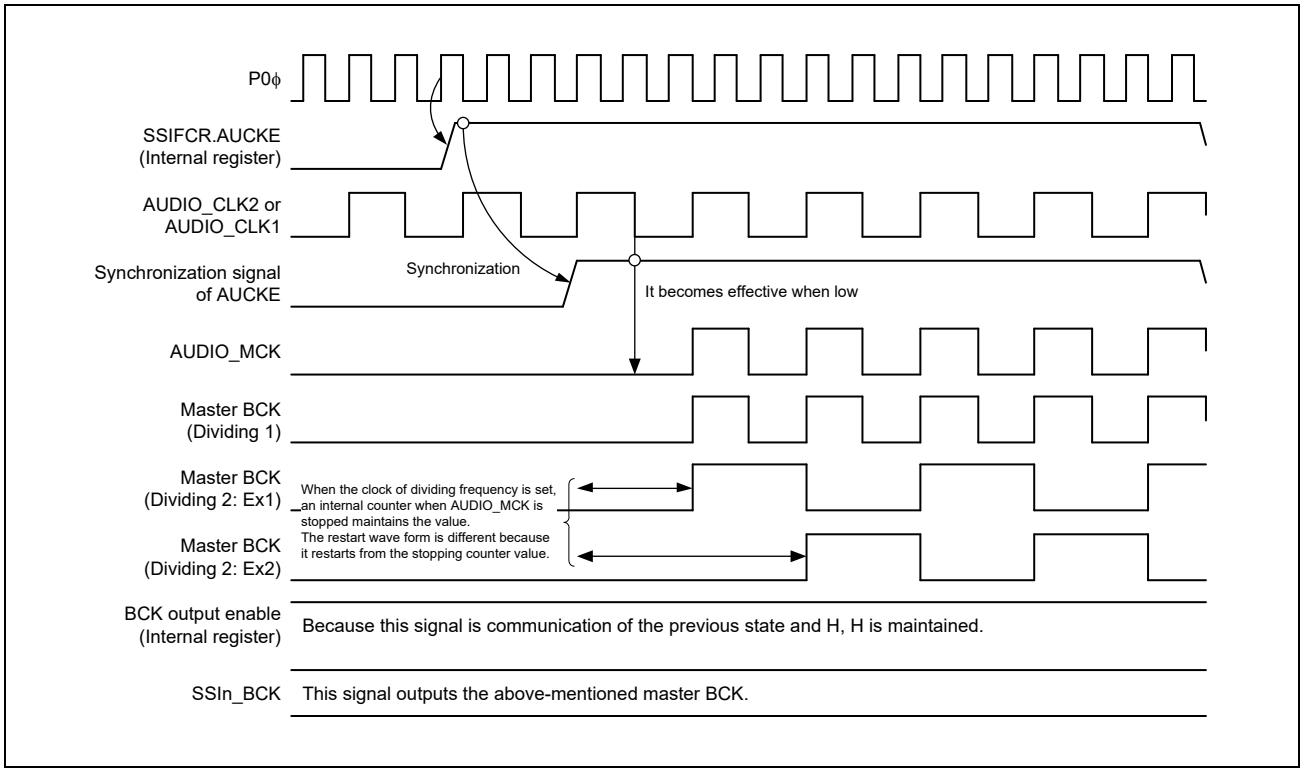


Figure 35.23 Timing Chart when It begins to Communicate Master from Communication Stop

Remark When the supply of AUDIO_MCK is stopped, the terminal SSIBCK maintains the value. The terminal SSIBCK might stop in the state of “H”.

SSIRST Bit

This bit sets a software reset of SSIF-2. Writing 1 to this bit initializes the internal state of SSIF-2. After a reset by writing 1, write 0 to release the reset because this bit is not automatically cleared to 0. After writing 0 to this bit, make sure that the bit becomes 0 before the next procedure.

To stop communication of SSIF-2 immediately, after turning off the peripheral functions, write 1 to this bit. Initialization by a software reset is performed without any relation with the bit clock.

Table 35.7 Bits Initialized by Software Reset of the SSIFCR.SSIRST Bit

Symbol	Address (Base+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	H'00	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	H'04	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	H'10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	BCKNCE	LRCKNCE	RxDNCE	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	H'14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	H'18	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	H'1C	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR	H'20	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	H'24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

BSW Bit

This bit enables/disables byte swap of register access for the transmit FIFO data register (SSIFTDR) and the receive FIFO data register (SSIFRDR). This bit is valid only with 16-bit access or 32-bit access to SSIFTDR and SSIFRDR (**Figure 35.24**).

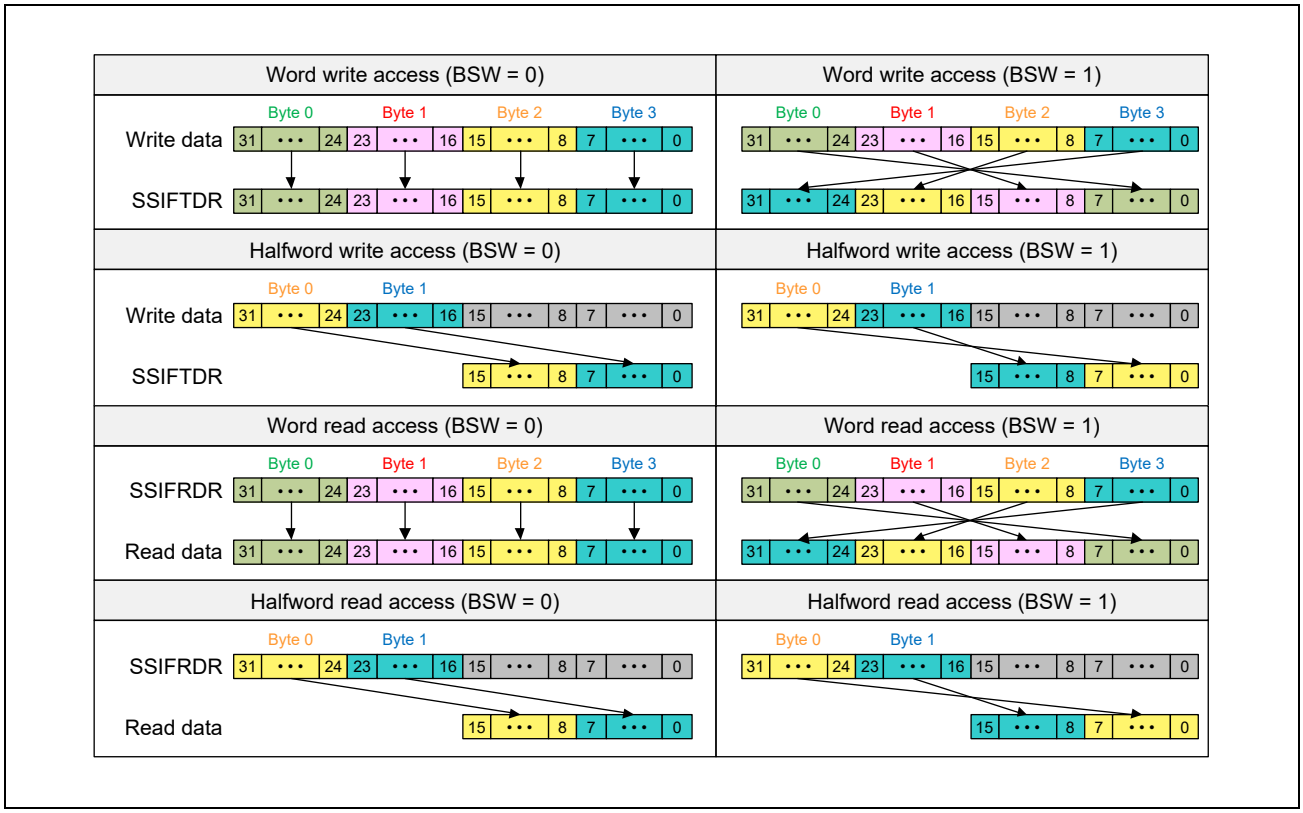


Figure 35.24 Operation Example of Byte Swap

BCKNCE Bit

This bit enables or disables the noise cancellation function supporting the suppression of transients 15-ns in length (typ.) in input signals on the SSIBCK pin in slave-mode communication (SSICR.MST = ‘0’) except in the case of transmission (SSICR.TEN = ‘1’) while SSICR.DEL = ‘1’. The setting of this bit in master-mode communication (SSICR.MST = ‘1’) is disabled.

Be sure to clear this bit to 0 in master-mode communication or during transmission in slave mode (SSICR.MST = ‘0’, SSICR.TEN = ‘1’) while SSICR.DEL = ‘1’. Set the BCKNCE, LRCKNCE, and RXDNCE bits to the same value.

LRCKNCE Bit

This bit enables or disables the noise cancellation function supporting the suppression of transients 15-ns in length (typ.) in input signals on the SSILRCK and SSIFS pins in slave-mode communication (SSICR.MST = ‘0’) except in the case of transmission (SSICR.TEN = ‘1’) while SSICR.DEL = ‘1’. The setting of this bit in master-mode communication (SSICR.MST = ‘1’) is disabled.

Be sure to clear this bit to 0 in master-mode communication or during transmission in slave mode (SSICR.MST = ‘0’, SSICR.TEN = ‘1’) while SSICR.DEL = ‘1’. Set the BCKNCE, LRCKNCE, and RXDNCE bits to the same value.

RXDNCE Bit

This bit enables or disables the noise cancellation function supporting the suppression of transients 15-ns in length (typ.) in input signals on the SSIRxD pin in slave-mode communication (SSICR.MST = ‘0’) except in the case of transmission (SSICR.TEN = ‘1’) while SSICR.DEL = ‘1’ and while reception is enabled (SSICR.REN = ‘1’).

The setting of this bit in master-mode communication (SSICR.MST = ‘1’) or while reception is disabled (SSICR.REN = ‘0’) is disabled.

Be sure to clear this bit to 0 in master-mode communication or during transmission in slave mode (SSICR.MST = ‘0’, SSICR.TEN = ‘1’) while SSICR.DEL = ‘1’. Set the BCKNCE, LRCKNCE, and RXDNCE bits to the same value.

TIE Bit

This bit enables/disables output of transmit data empty interrupts. Transmission data empty interrupt is used as an interrupt factor written in the FIFO transmission data register. Set this bit to 1 after setting set condition (SSISCR.TDES) of transmission data empty interrupt. See **Figure 35.25** for the generation timing of a transmit data empty interrupt.

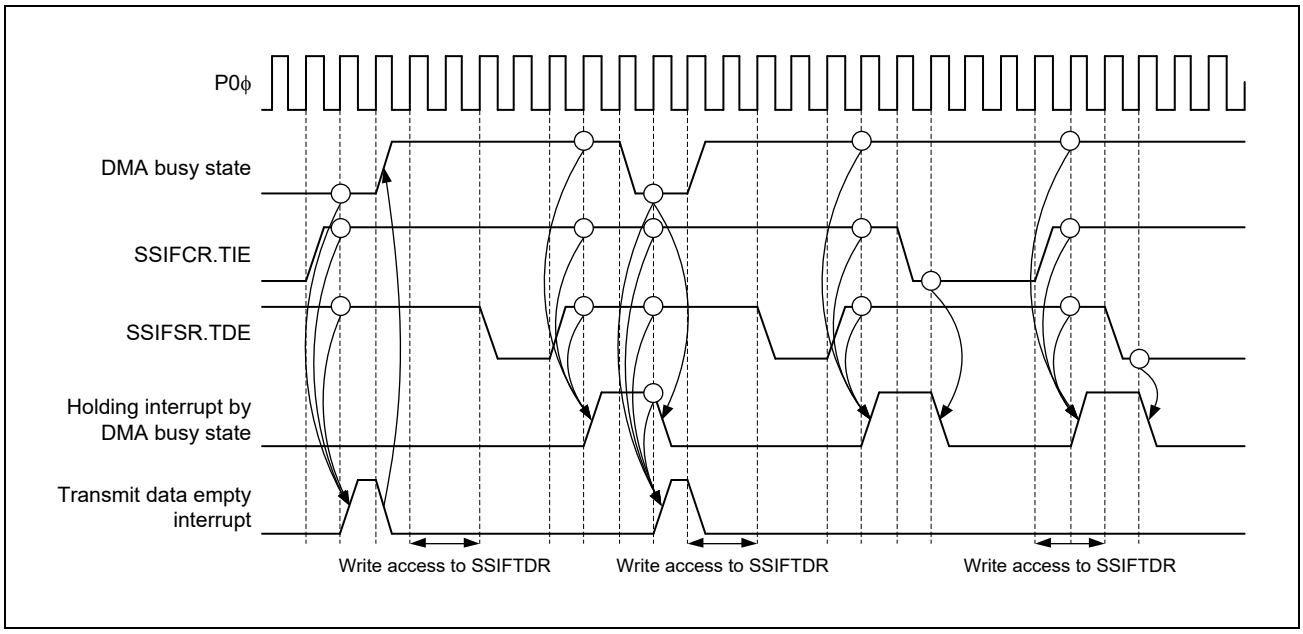


Figure 35.25 Generation Timing of Transmit Data Empty Interrupt

RIE Bit

This bit enables/disables output of receive data full interrupts. Receive data empty interrupt is used as an interrupt factor to read the FIFO receive data register. Set this bit to 1 after setting set condition (SSISCR.RDFS) of Receive data empty interrupt. See **Figure 35.26** for the generation timing of a receive data full interrupt.

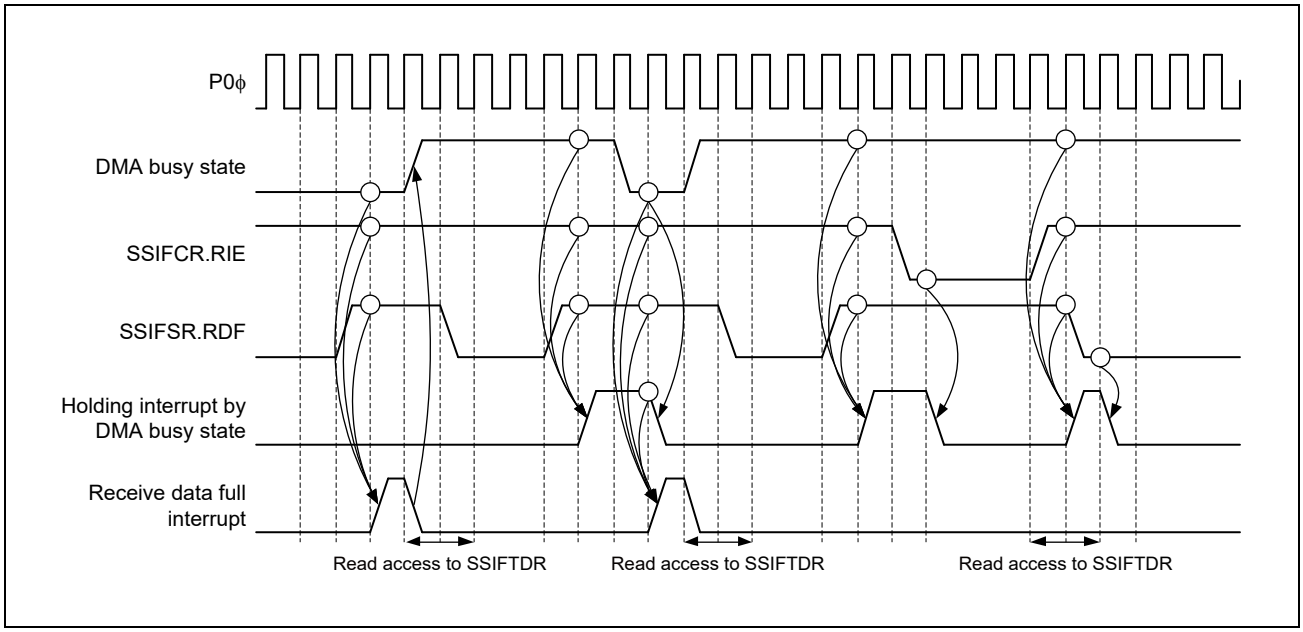


Figure 35.26 Generation Timing of Receive Data Empty Interrupt

TFRST Bit

This bit sets a software reset of the transmit FIFO data register (SSIFTDR). Writing 1 to this bit initializes the internal state related to SSIFTDR. See **Table 35.8** for the registers to which this software reset is applied. After a reset by writing 1, write 0 to clear the reset condition because this bit is not automatically cleared to 0. After writing 0 to this bit, make sure that the bit becomes 0 before the next procedure.

This bit is a bit of SSIRST for reset. Software reset by SSIRST becomes and writing this bit becomes invalid because of priority.

Table 35.8 Bits Initialized by Software Reset of the SSIFCR.TFRST Bit

Symbol	Address (Base+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	H'00	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	H'04	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	H'10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	BCKNCE	LRCKNCE	RxDNCE	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	H'14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	H'18	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	H'1C	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR	H'20	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	H'24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

RFRST Bit

This bit sets a software reset of the receive FIFO data register (SSIFRDR). Writing 1 to this bit initializes the internal state related to SSIFRDR. After a reset by writing 1, write 0 to clear the reset condition because this bit is not automatically cleared to 0. After writing 0 to this bit, make sure that the bit becomes 0 before the next procedure.

This bit is a bit of SSIRST for reset. Software reset by SSIRST becomes and writing this bit becomes invalid because of priority.

Table 35.9 Bits Initialized by Software Reset of the SSIFCR.TFRST Bit

Symbol	Address (Base+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	H'00	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	H'04	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	H'10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	BCKNCE	LRCKNCE	RxDNCE	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	H'14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	H'18	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	H'1C	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR	H'20	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	H'24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

35.4.1.4 IFO Status Register (SSIFSR)

This register is configured with status flags that indicate the status of the transmit FIFO data register and the receive FIFO data register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W0

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	H'0	R	Reserved. Write 0. The read value is 0.
29 to 24	TDC[5:0]	H'00	R	Number of Transmit FIFO Data Indication Flag
23 to 17	—	H'00	R	Reserved. Write 0. The read value is 0.
16	TDE	1	RW0	Transmit Data Empty Flag 0: The free space of SSIFTDR is not more than the value of SSISCR.TDES 1: The free space of SSIFTDR is not less than the value of SSISCR.TDES plus one.
15, 14	—	H'0	R	Reserved. Write 0. The read value is 0.
13 to 8	RDC[4:0]	H'00	R	Number of Receive FIFO Data Indication Flag
7 to 1	—	H'00	R	Reserved. Write 0. The read value is 0.
0	RDF	0	RW0	Receive Data Full Flag 0: The size of received data in SSIFRDR is not more than the value of SSISCR.RDFS 1: The size of received data in SSIFRDR is not less than the value of SSISCR.RDFS plus one.

TDC[5:0] Bits

These bits indicate the number of valid data that are stored in the transmit FIFO data register (SSIFTDR). With this flag as H'0, there is no data to be transmitted. With H'10, there is no space to write data.

TDE Bit

This bit indicates that the transmit FIFO data register (SSIFTDR) has free space not less than the amount set with the SSIFCR.TTRG bit plus one. This flag is set by automatic determination but it must be cleared by register access.
[Priority order for setting and clearing]

Clearing is prioritized.*1

[Clearing condition]

Either of the following two:

(1) Writing 0 to this bit after reading this bit as 1.*2

(2) When writing data to SSIFTDR.

[Clearing timing]

Timing according to the above-mentioned clear condition.

(1) At completion of writing 0 to this bit after reading this bit as 1 (**Figure 35.13**).

(2) When writing data to SSIFTDR.

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1) and transmit FIFO data register reset (SSIFCR.TFRST = 1). The software reset and transmit FIFO data register reset have priority over all the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following four conditions being met:
- Software reset (SSIFCR.SSIRST = 1)
- Resetting of the transmit FIFO data register (SSIFCR.TFRST = 1)
- Completion of writing 0 to this flag after having read it as 1.
- When writing data to SSIFTDR.

[Setting condition]

SSIFTDR has free space not less than the amount set with the SSIFCR.TTRG bit plus one.

[Setting timing]

While operating on P0ϕ, SSIFTDR is found to have free space not less than “size set in the SSISCR.TDES bits + 1.”

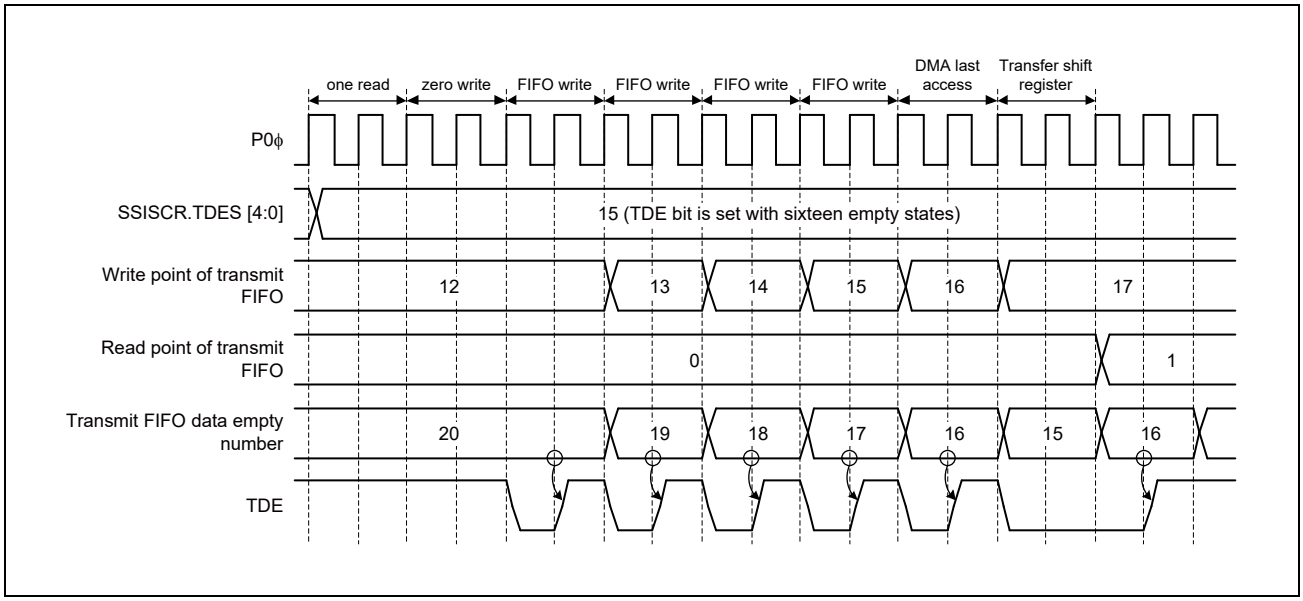


Figure 35.27 Set Timing and Clear Timing of TDE

RDC[5:0] Bits

These bits indicate the number of valid data that are stored in the receive FIFO data register (SSIFRDR). With this flag as H’0, there is no received data. With H’20, the register is filled with received data and there is no free space.

RDF Bit

This bit indicates that the receive FIFO data register (SSIFRDR) has unread received data not less than the amount set with the SSISCR.RDFS bit plus one. This flag is set by automatic determination but it must be cleared by register access.

[Priority order for setting and clearing]

Clearing is prioritized.*1

[Clearing condition]

Either of the following two:

(1) Writing 0 to this bit after reading this bit as 1.*2

(2) When reading data from SSIFRDR.

[Clearing timing]

Timing according to the above-mentioned clear condition.

(1) At completion of writing 0 to this bit after reading this bit as 1 (**Figure 35.13**).

(2) After reading data from SSIFRDR.

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1) and receive FIFO data register reset (SSIFCR.RFRST = 1). Clearing conditions available for these bits are the software reset and receive FIFO data register reset as well as the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following four conditions being met:
- Software reset (SSIFCR.SSIRST = 1)
- Receive FIFO data register reset (SSIFCR.RFRST = 1)
- Completion of writing 0 to this flag after having read it as 1.
- When reading data from SSIFRDR.

[Setting condition]

SSIFRDR has data not less than the amount set with the SSISCR.RDFS bit plus one.

[Setting timing]

At completion of transfer from the shift register that results in SSIFRDR having data not less than the amount set with the SSISCR.RDFS bit plus one.

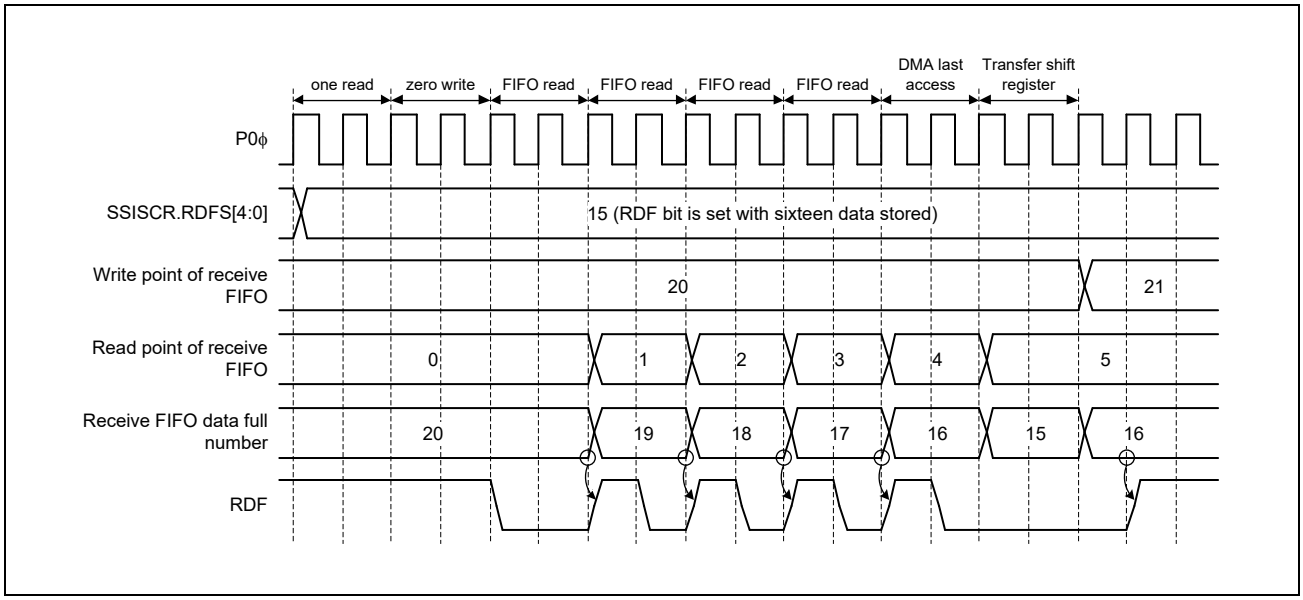
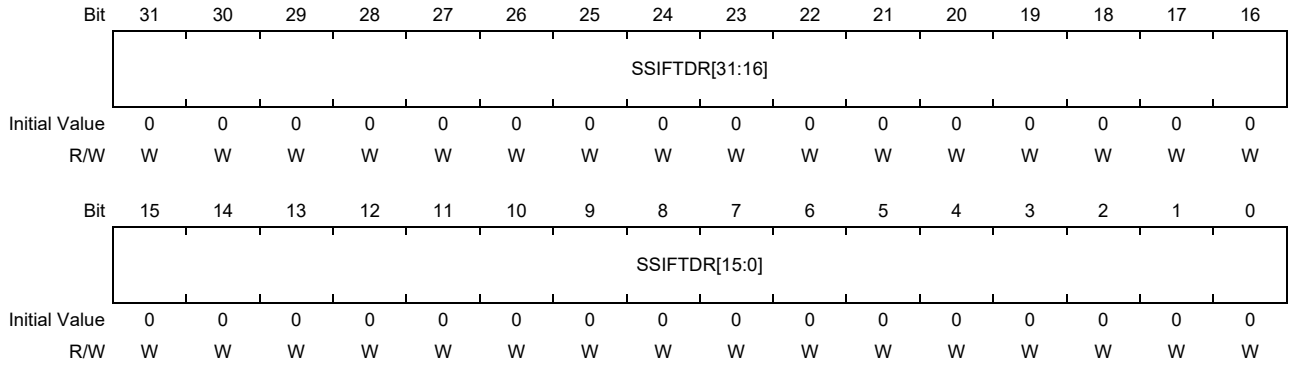


Figure 35.28 Set Timing and Clear Timing of RDF

35.4.1.5 Transmit FIFO Data Register (SSIFTDR)

This is a 32-bit writable register. 0 is returned when this register is read. This register stores data to be serially transmitted.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SSIFTDR[31:0]	H'0000_0000	W	Transmit FIFO data

To use this register for transmission, set DMA operation handling of a transmit data empty interrupt as writing to this register. Determine the access size to this register according to the data word length to be communicated (**Table 35.10**).

Table 35.10 Register Access Restriction to FIFOs

Access Size				
SSICR.DWL[2:0]	Data Word Length	Byte	Halfword	Word
000b	8	✓	—	—
001b	16	—	✓	—
010b	18	—	—	✓
011b	20	—	—	✓
100b	22	—	—	✓
101b	24	—	—	✓
110b	32	—	—	✓
111b	Setting prohibited	—	—	—

Figure 35.29 shows register access to the transmit FIFO data register.

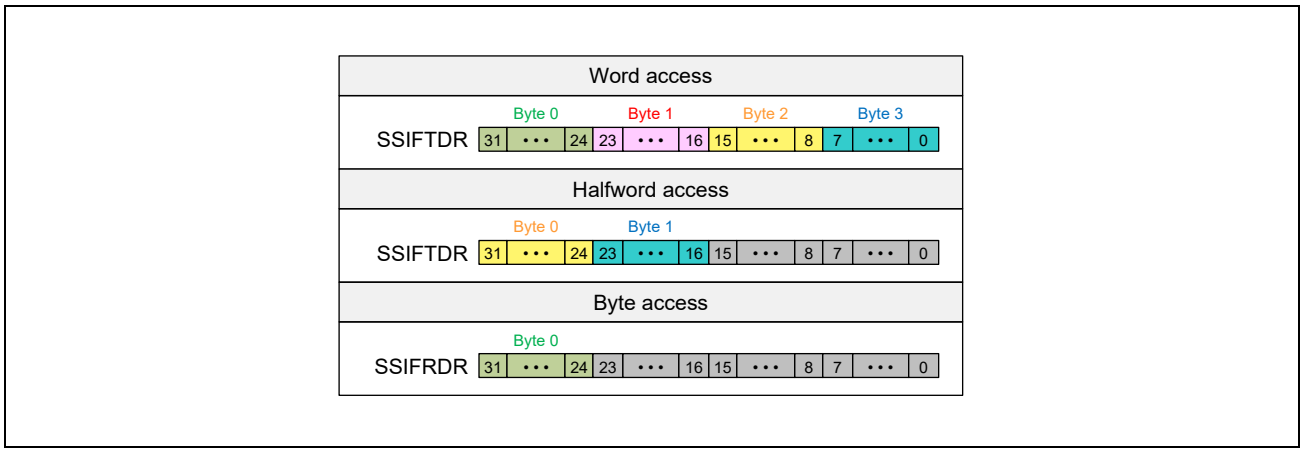


Figure 35.29 Example of Register Access to the Transmit FIFO Data Register

Figure 35.30 shows the configurations and operation examples of the transmit FIFO data register and transmit shift register. The configurations are for storing data to FIFO and not related with communication.

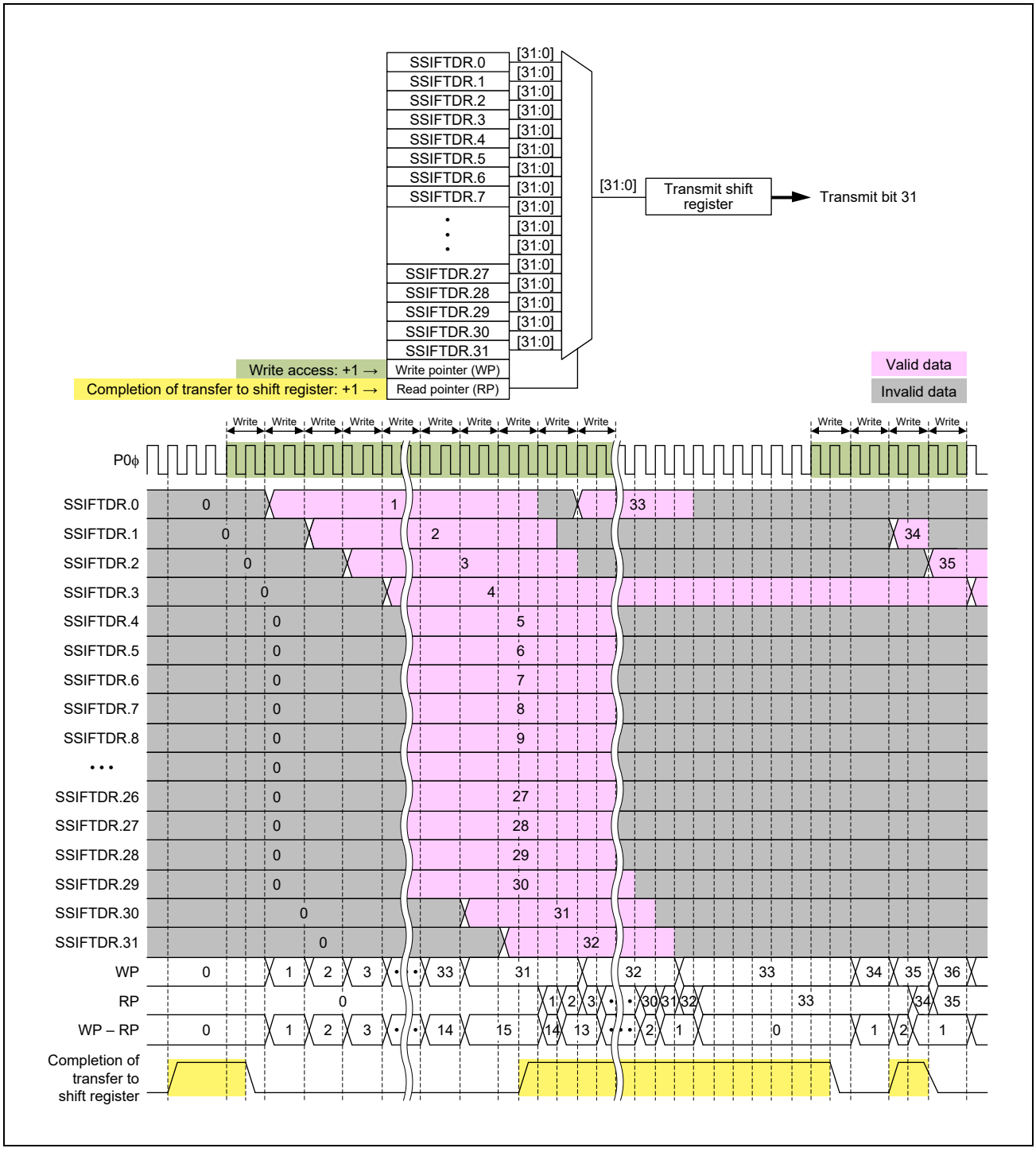


Figure 35.30 Configuration of the Transmit FIFO Data Register and Transmit Shift Register, and FIFO Operation Example

35.4.1.6 Receive FIFO Data Register (SSIFRDR)

This is a readable 32-bit register. This register stores received serial data.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SSIFRDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SSIFRDR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SSIFRDR [31:0]	H'0000_0000	R	Receive FIFO data

To use this register for reception, set DMA operation handling of a receive data full interrupt as reading from this register. Determine the access size to this register according to the data word length to be communicated (**Table 35.10**). Register access to the receive FIFO data register is same as for the transmit FIFO data register (**Figure 35.29**).

Figure 35.31 shows the configurations and operation examples of the receive FIFO data register and receive shift register.

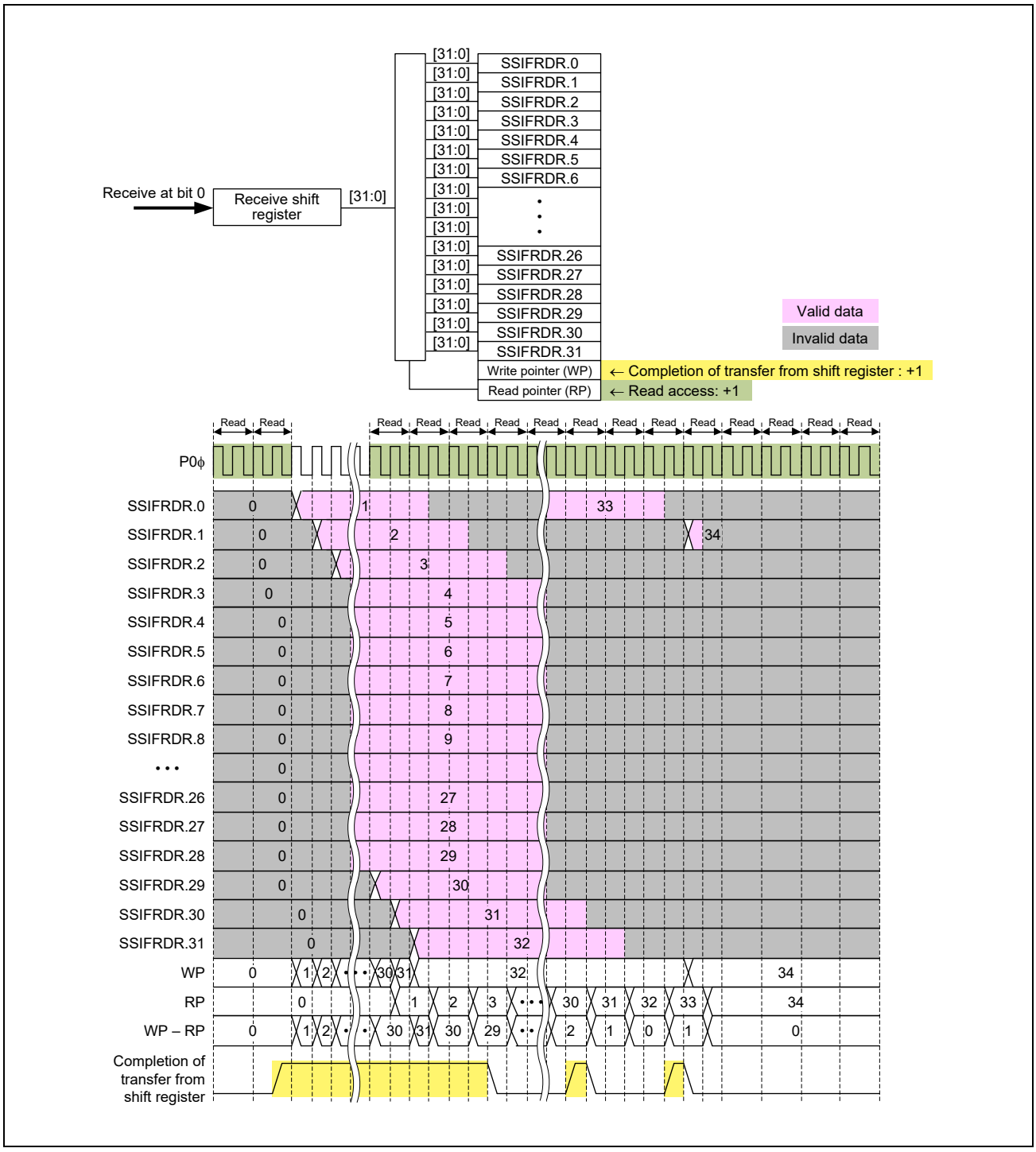


Figure 35.31 Configuration of the Receive FIFO Data Register and Receive Shift Register, and FIFO Operation Example

35.4.1.7 Audio Format Register (SSIOFR)

This is a readable/writable 32-bit register. It sets an audio format including the communication format, LRCK/FS continuation mode, and BCK output stop.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	BCKSTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	H'00_0000	R	Reserved. Write 0. The read value is 0.
9	BCKASTP	0	RW	BCK Output Stop Enable*1,*2 0: Enables output of BCK 1: Disables output of BCK
8	LRCONT	0	RW	LRCK/FS Continuation Enable*1,*2 0: Disables LRCK continuation 1: Enables LRCK continuation
7 to 2	—	H'00	R	Reserved. Write 0. The read value is 0.
1, 0	OMOD[1:0]	0	RW	Audio Format Select*3,*4 OMOD[1:0]: Format 00: I^{2}S format 01: TDM format 10: Monaural format 11: Setting prohibited

Note 1. This bit is valid only in master-mode communication (SSICR.MST = 1). The setting is invalid in slave-mode communication (SSICR.MST = 0).

Note 2. BCKASTP=1 and LRCONT=1 is prohibitions.

Note 3. Writing in this bit is a prohibition when SSIF-2 is communication state (SSISR.IIRQ=0). Operation is not guaranteed when rewriting it.

Note 4. Use the communication format setting that can be communicated when the format of the communication of the opposing device is compatible format of SSIF-2.

BCKASTP Bit

This bit turns on or off the function to output BCK to the SSIBCK pin according to the communication shown in **Figure 35.32** and **Figure 35.33** in master-mode communication (SSICR.MST = 1). Set this bit after setting completing the communication format.

This bit must defend the following usage. Start communication with BCKASTP = 0, and set 1 to BCKASTP while communicating. As a result, the bit clock output to the terminal SSIBCK stops by the automatic operation when the communication stop is done. When you will restart the communication, set 0 to BCKASTP with idle state (SSICR.IIRQ=1) and the AUDIO_MCK supply state (SSIFCR.AUCKE = 1)

When master communication (SSICR.MST=1) and idle state (SSICR.IIRQ=1).

Table 35.11 BCKASTP State, and Output state of SSIBCK Pin

BCKASTP bit	Output State of SSIBCK Pin
0	Enable
1	Disable

Note: When the opposing device who is the slave needs the clock of the terminal SSIBCK before the communication operates, it is not possible to use it. Use it to stop clocking after the communication ends (**Figure 35.33**). Refer to **Figure 35.32** for timing that the function becomes effective.

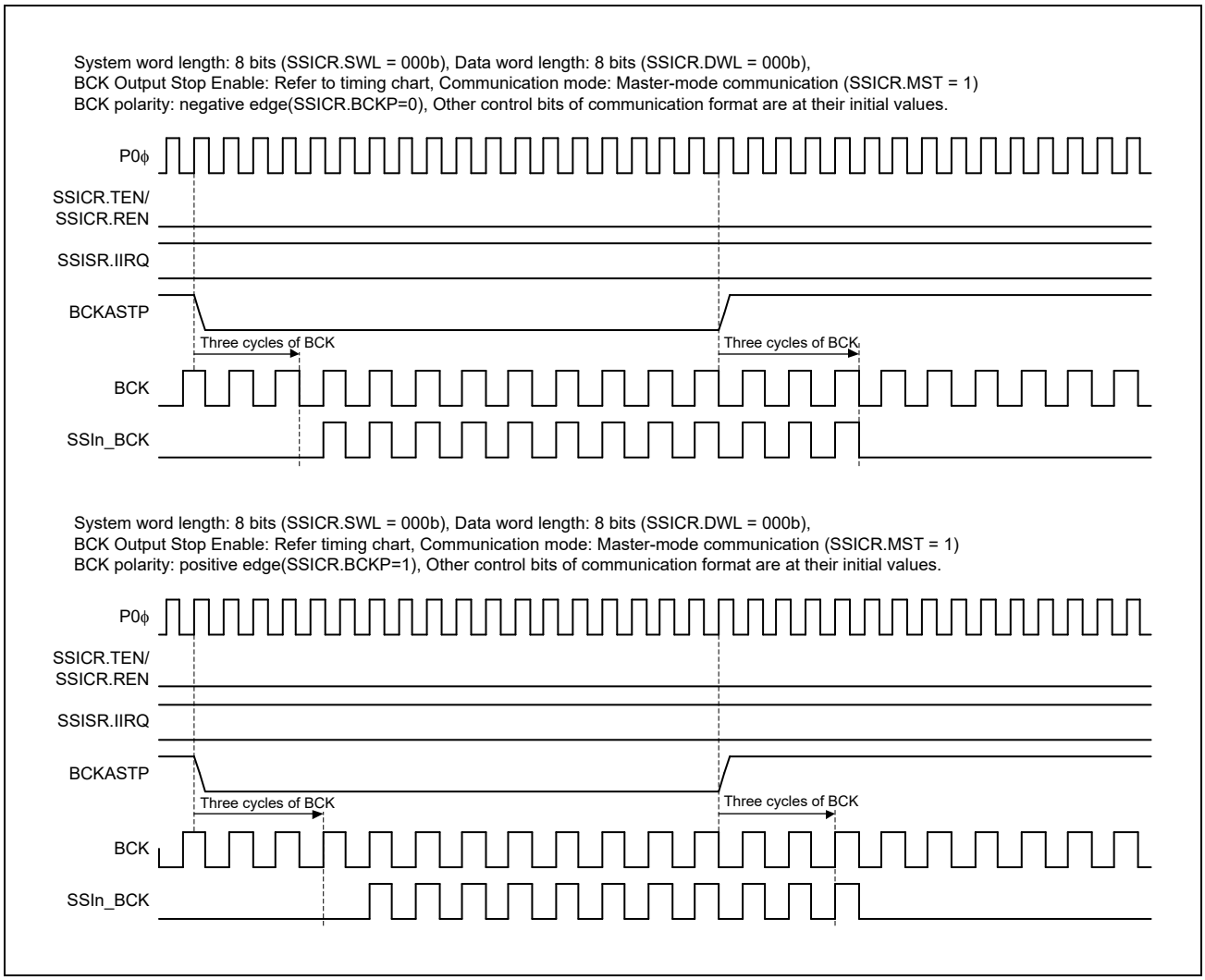


Figure 35.32 Example Operation of the BCKASTP Bit (in Idle State)

When master communication (SSICR.MST=1) and the BCK output automatic operation stop function (BCKASTP = 1) (**Figure 35.33**).

Details of the BCK output to SSIBCK pin are as follows.

- Output start timing: To generate an effective edge in timing that LRCK/FS is changed into the valid value, BCK is output.
- Output stop timing: Frame boundary from 1 to 1.5 clock

Refer to the timing chart of **Figure 35.33** for detailed timing.

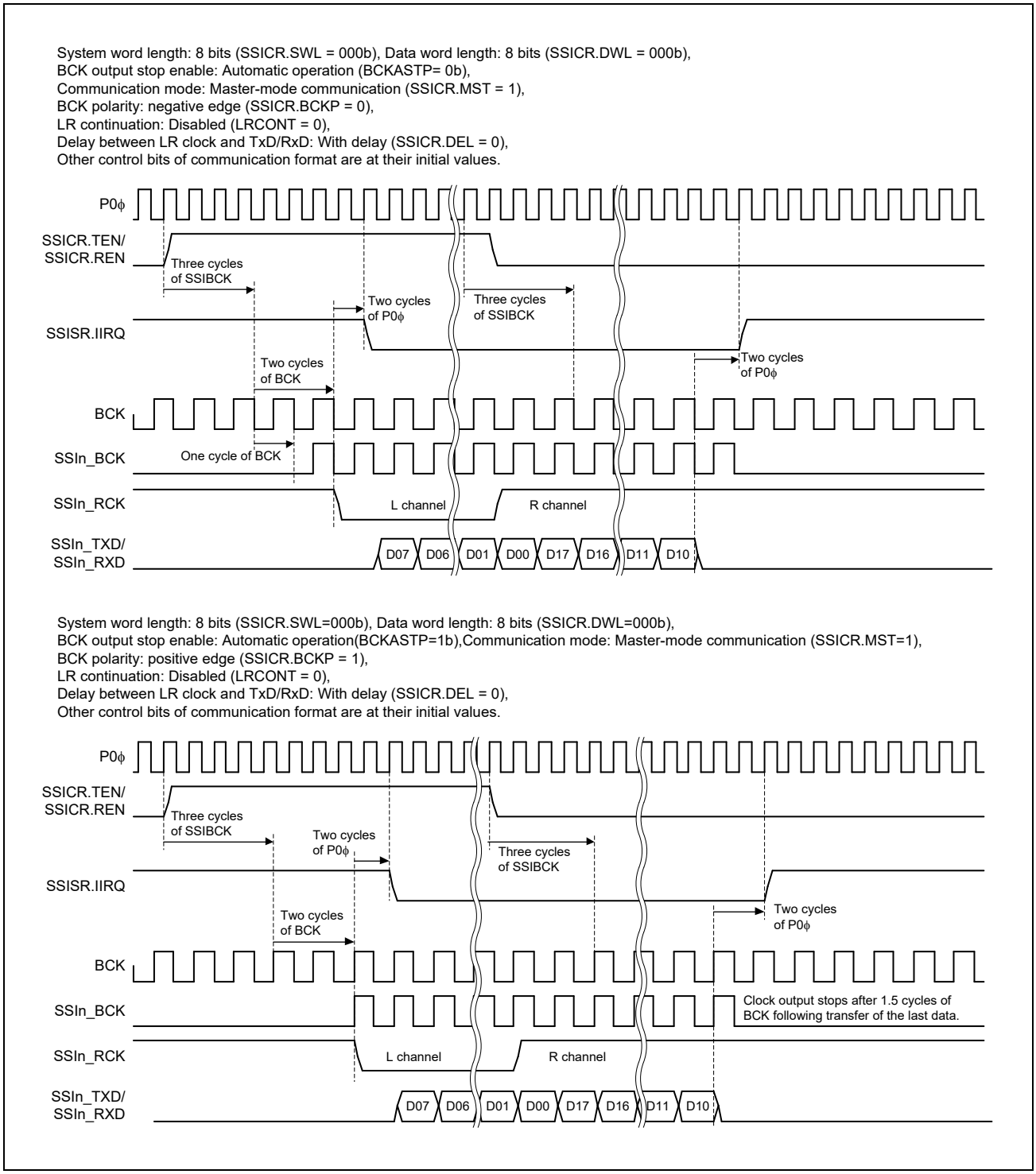


Figure 35.33 Example Operation of the BCKASTP Bit (Communication behavior in BCKASTP=1)

LRCONT Bit

This bit enables/disables the output from the SSILRCK/SSIFS pin when SSIF-2 is master communication (SSICR.MST=1) and idle state (SSISR.IIRQ=1). Writing 1 to this bit (enables LRCK/FS continuation) in master mode (SSICR.MST = 1) enables continuation of the output from the SSILRCK/SSIFS pins even when idle state.

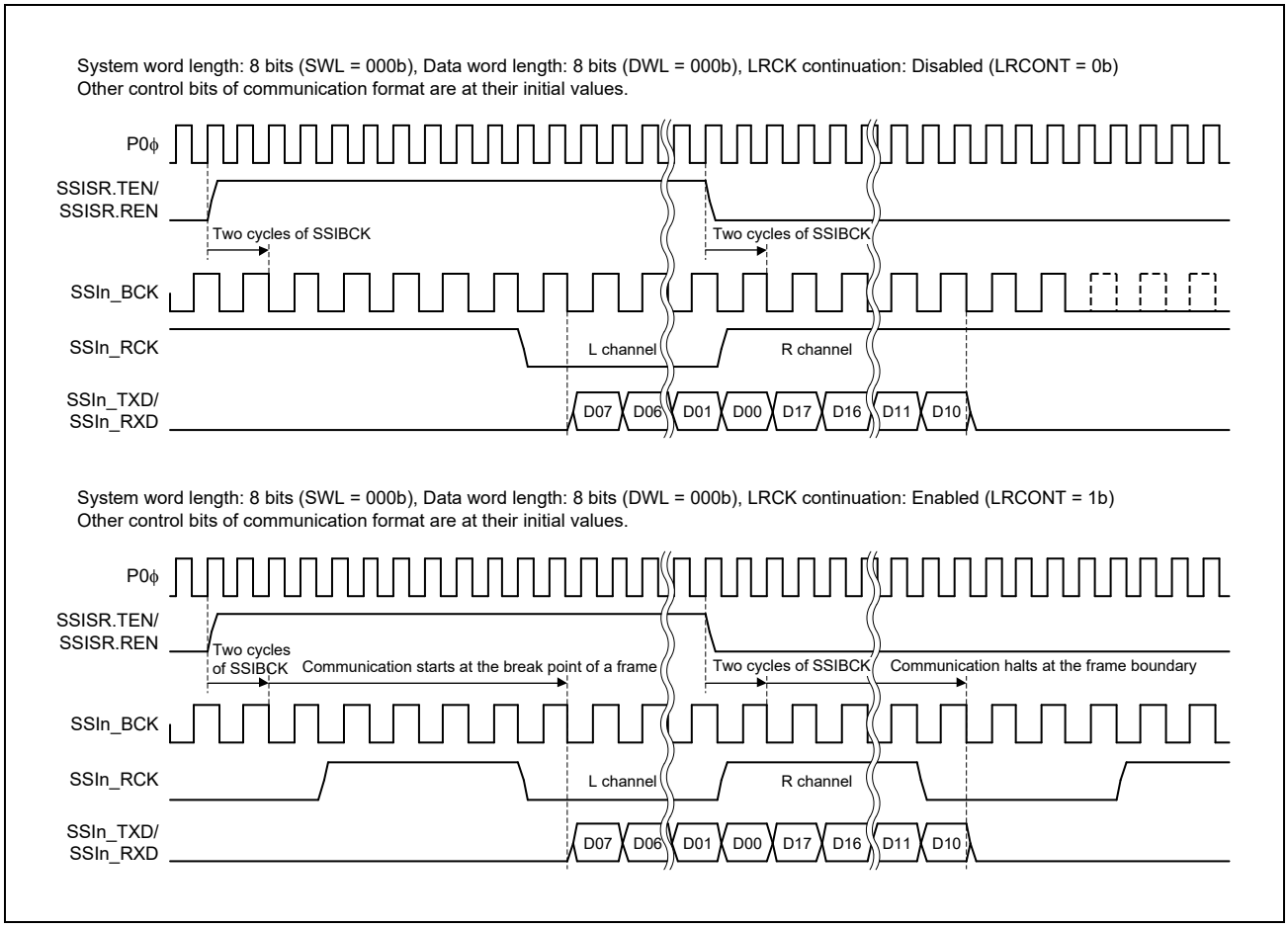


Figure 35.34 Operation Example of LRCK/FS Continuation (in Idle State)

OMOD[1:0] Bits

These bits set an audio format.

Rewrite this bit when the LR clock supply to the terminal SSILRCK stops. Refer to the LRCONT bit details explanation in **Section 35.4.1.7, Audio Format Register (SSIOFR)** for the output operation of the LR clock.

35.4.1.8 Status Control Register (SSISCR)

This is a readable/writable 32-bit register. It sets the operation of the TDE and RDF flags.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	H'0_0000	R	Reserved. Write 0. The read value is 0.
12 to 8	TDES[4:0]	H'00	RW	TDE Setting Condition Select*1 TDES[4:0]: Operation 00000b: SSIFTDR has one stage or more free space 00001b: SSIFTDR has two stages or more free space (snip) 11110b: SSIFTDR has thirty-one stages or more free space 11111b: SSIFTDR has thirty-two stages or more free space
7 to 5	—	H'0	R	Reserved. Write 0. The read value is 0.
4 to 0	RDFS[4:0]	H'00	RW	RDF Setting Condition Select*1 RDFS[4:0]: Operation 00000b: SSIFRDR has one stage or more data size 00001b: SSIFRDR has two stages or more data size (snip) 01110b: SSIFRDR has thirty-one or more data size 01111b: SSIFRDR has thirty-two stages or more data size

Note 1. Writing to these bits while SSIF-2 is in a communication state (SSISR.IIRQ = 0) is prohibited. If written, the operation performed immediately after writing is not guaranteed.

TDES[4:0] Bits

These bits set the setting condition of the transmit data empty flag (TDE). Set these bits in byte units.

RDFS[4:0] Bits

These bits set the setting condition of the receive data full flag (RDF). Set these bits in byte units.

35.4.2 Communication Formats

SSIF-2 supports three communication formats (**Table 35.12**).

Table 35.12 Supported Communication Formats

Communication Format	SSIOFR.OMOD[1:0]
I^2S format	00
TDM format	01
Monaural format	10

The following figure shows the serial data configuration common to communication formats. Serial data is determined according to the system word length (SSICR.SWL[2:0]) and the data word length (SSICR.DWL[2:0]). When the system word length is longer than the data word length, padding bits are transferred (**Figure 35.35**).

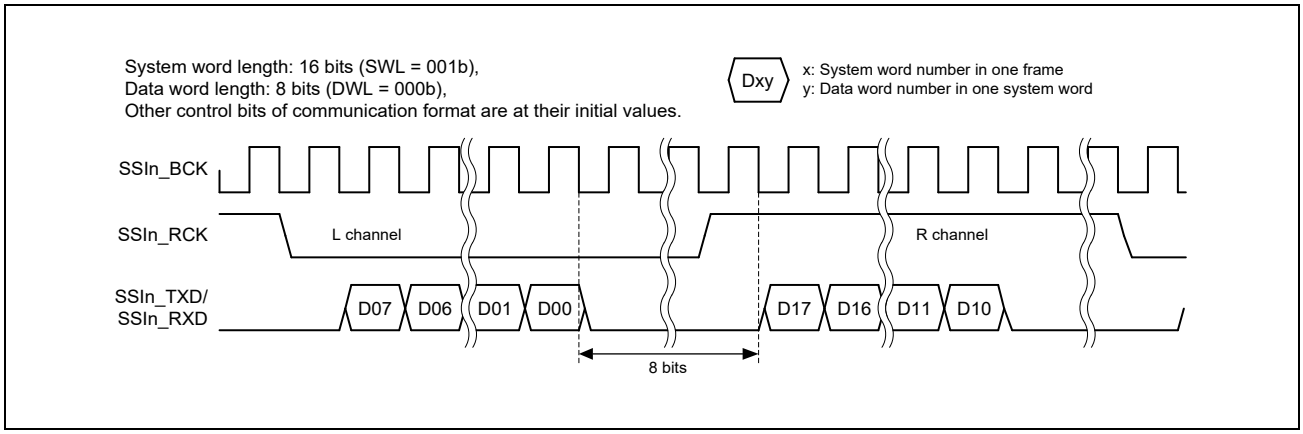


Figure 35.35 Example of Padding Bit Transfer (I^2S Format: System Word Length > Data Word Length)

Table 35.13 lists the number of padding bits to be transferred with each combination of system word length (SSICR.SWL[2:0]) and data word length (SSICR.DWL[2:0]). "-" indicates that the setting is prohibited.

Table 35.13 Number of Padding Bits

	SSICR.DWL[2:0]	000b	001b	010b	011b	100b	101b	110b	111b
SSICR.SWL[2:0]	System Word Length	8	16	18	20	22	24	32	Set Prohibition
000b	8	0	—	—	—	—	—	—	—
001b	16	8	0	—	—	—	—	—	—
010b	24	16	8	6	4	2	0	—	—
011b	32	24	16	14	12	10	8	0	—
100b	48	40	32	30	28	26	24	16	—
101b	64	56	48	46	44	42	40	32	—
110b	128	120	112	110	108	106	104	96	—
111b	256	248	240	238	236	234	232	224	—

35.4.2.1 I²S Format

The I²S format is a communication format used for connection with I²S-compatible serial devices. With this format setting (SSIOFR.OMOD[1:0] = 00b), one frame is configured with two system words, one for the channel L and the other for channel R. The SSILRCK/SSIFS signals are at a low level for the channel L and at a high level for the channel R. Set the polarity of the signals with the SSICR.LRCKP bit. **Figure 35.36** shows the I²S format without padding. See **Figure 35.35** for the format with padding.

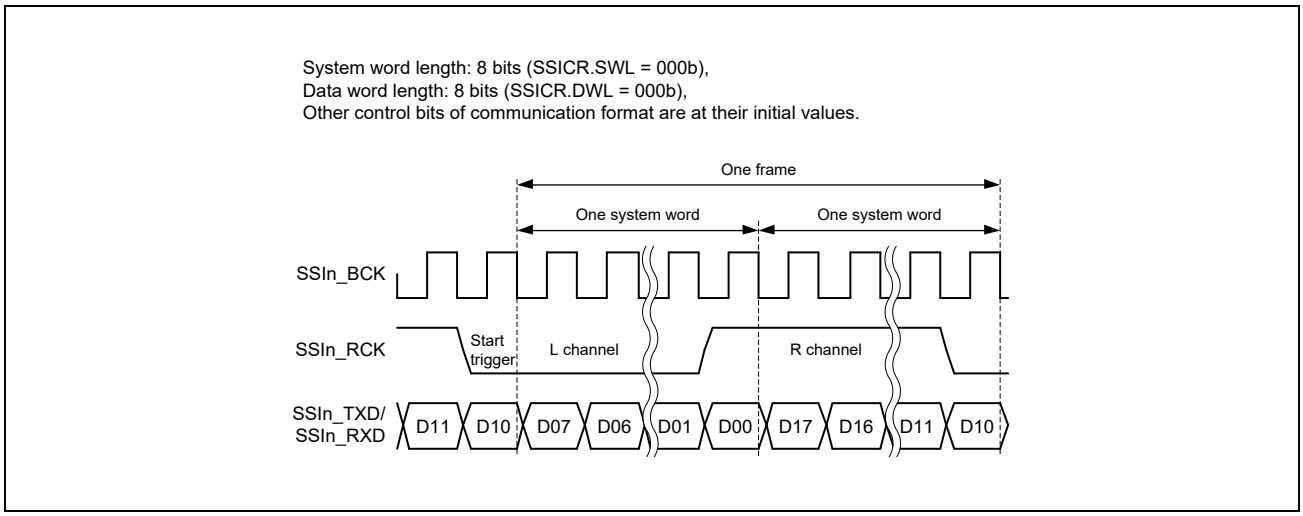


Figure 35.36 I²S Format (without Padding: System Word Length = Data Word Length)

For the state of external pins when SSIF-2 is in the idle state, see **Section 35.5.1.1, Idle State**.

NOTE

SSIF-2 has the terminal SSILRCK/SSIFS that shows the synchronization of the communication. Match communication format of the opposing device and SSIF-2 when SSIF-2 is slave communication (SSICR.MST=0). SSIF-2 uses the signal of SSILRCK/ SSIFS only as a communication start trigger.

35.4.2.2 Monaural Format

The monaural format is a communication format used for connection with monaural-compatible serial devices. With this format setting (SSIOFR.OMOD[1:0] = 10b), one frame is configured with the number of system words set with the SSICR.FLM[1:0] bits (**Figure 35.37**). A rising of the SSILRCK/SSIFS signal means a start trigger. **Figure 35.37** and **Figure 35.38** respectively show the monaural formats without and with padding.

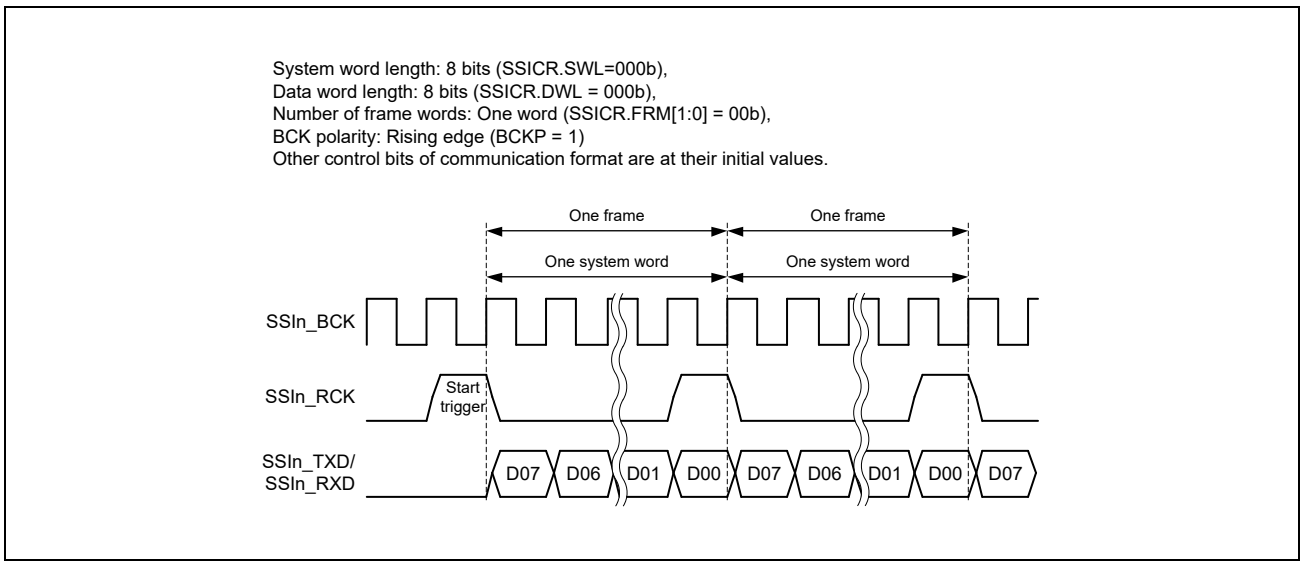


Figure 35.37 Short Frame in Monaural Format (without Padding: System Word Length = Data Word Length)

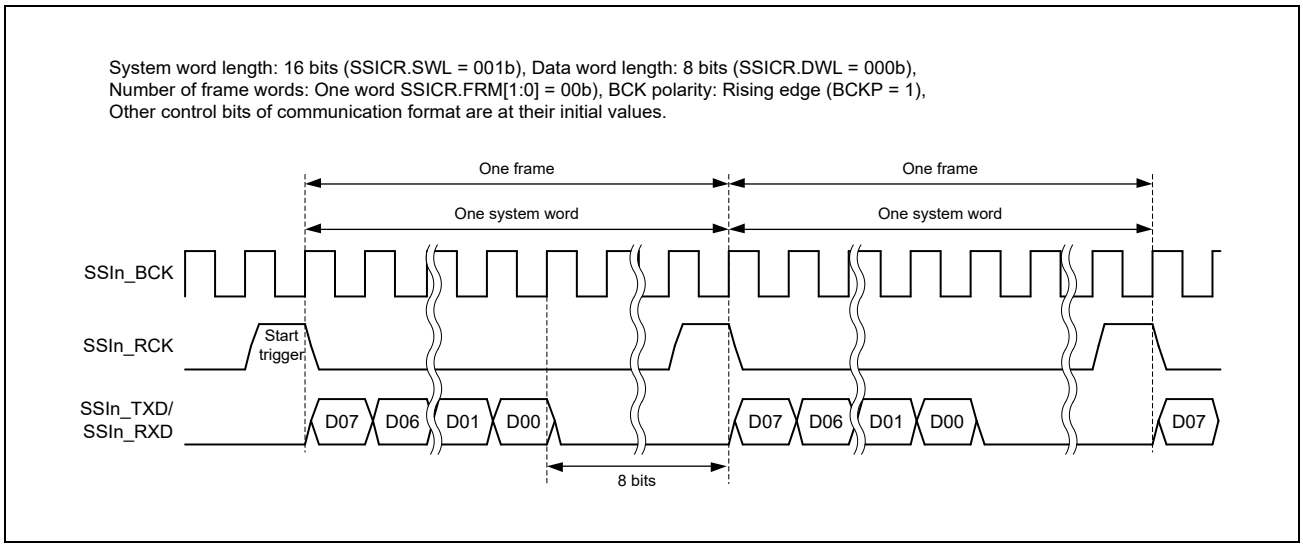


Figure 35.38 Short Frame in Monaural Format (with Padding: System Word Length > Data Word Length)

The monaural formats supported by SSIF-2 consist of short frames and long frames. See **Section 35.4.2.2(1), Short Frame** and **Section 35.4.2.2(2), Long Frame** for the difference between these two frames.

For the state of external pins state when SSIF-2 is in the idle state, see **Section 35.5.1.1, Idle State**.

NOTE

SSIF-2 has the terminal SSILRCK/SSIFS that shows the synchronization of the communication. Match communication format of the opposing device and SSIF-2 when SSIF-2 is slave communication (SSICR.MST=0). SSIF-2 uses the signal of SSILRCK/SSIFS only as a communication start trigger.

(1) Short Frame

With the short frame (SSICR.DEL = 0), the SSILRCK/SSIFS signal that indicates a start of serial data is at a high level for one cycle of SSIBCK. Data transfer starts at the falling edge of the signal (**Figure 35.37** and **Figure 35.38**).

(2) Long Frame

With the long frame (SSICR.DEL = 1), the SSILRCK/SSIFS signal that indicates a start of serial data is at a high level for two cycles of SSIBCK. Data transfer starts at the rising edge of the signal (**Figure 35.39**).

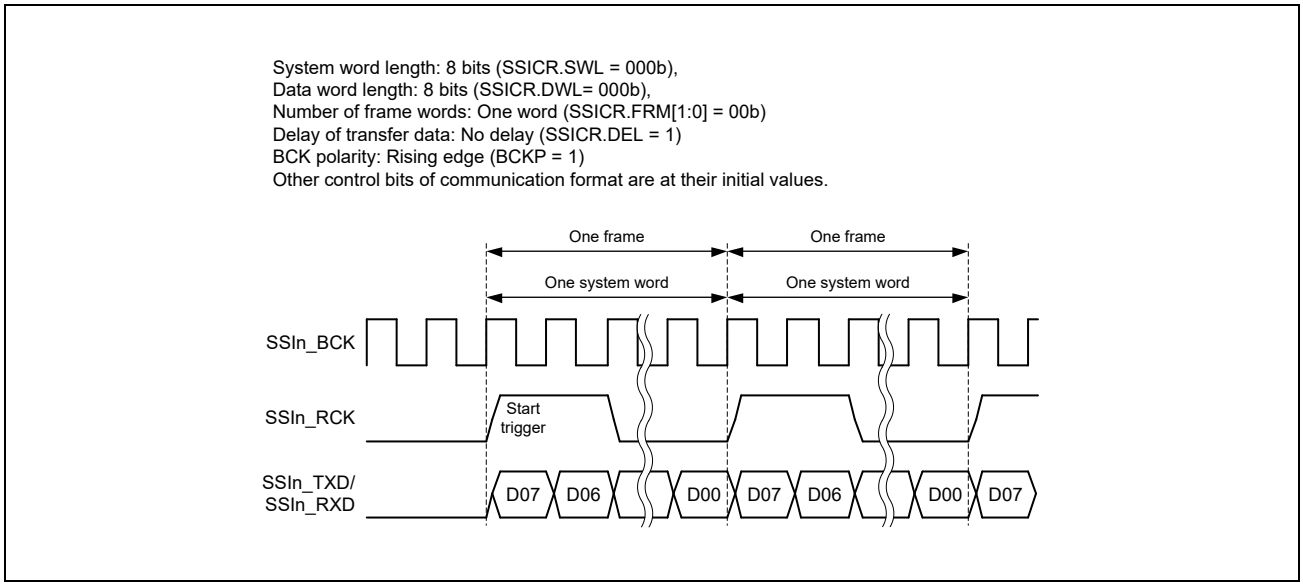


Figure 35.39 Long Frame in Monaural Format (without Padding)

35.4.2.3 TDM Format

The TDM format is a communication format used for connection with TDM-compatible multi-channel devices. With this format setting (SSIOFR.OMOD[1:0] = 01b), one frame is configured with four to eight system words set with the SSICR.FRM[1:0] bits. With this format, the SSILRCK/SSIFS signal is at a high level for the first one system word and at a low level for the rest. The pulse generated on the SSILRCK/SSIFS signal is defined as the SYNC pulse and its rising edge means a start of one frame. **Figure 35.40** and **Figure 35.41** respectively show the TDM formats with and without padding.

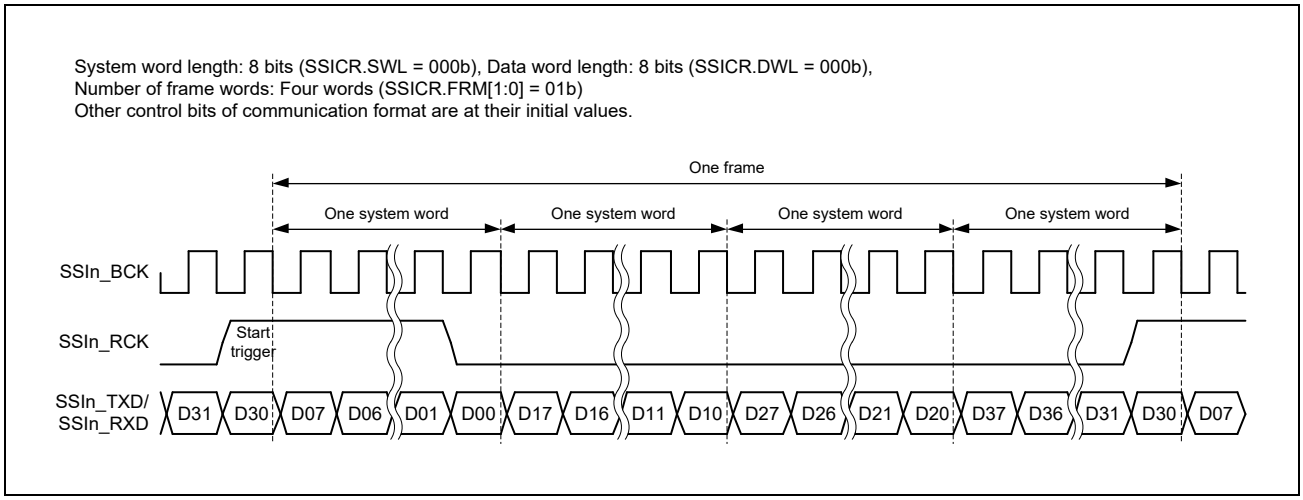


Figure 35.40 TDM Format (without Padding: System Word Length = Data Word Length)

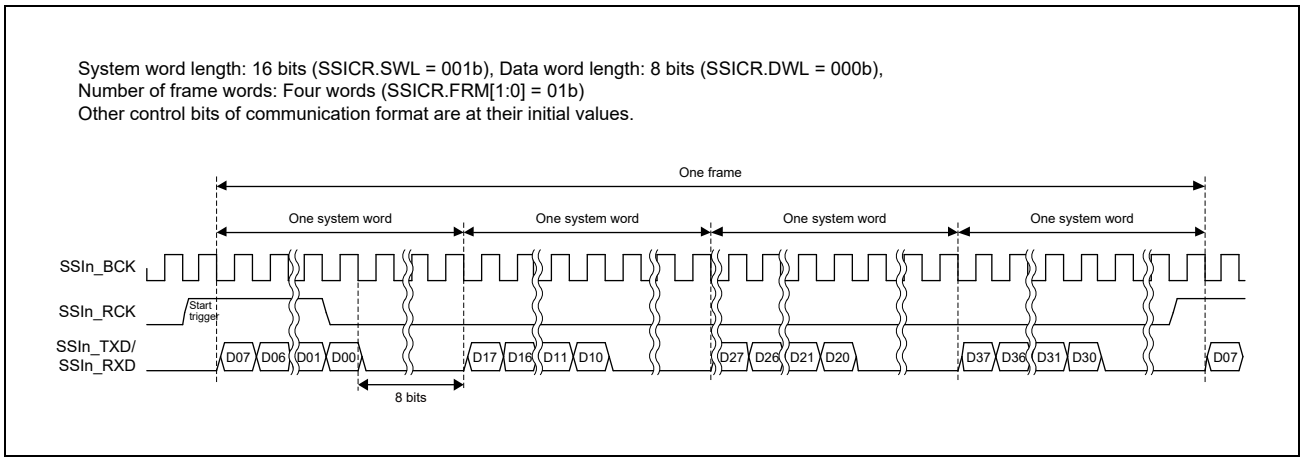


Figure 35.41 TDM Format (with Padding: System Word Length > Data Word Length)

For the state of external pins when SSIF-2 is in the idle state, see **Section 35.5.1.1, Idle State**.

NOTE

SSIF-2 has the terminal SSILRCK/SSIFS that shows the synchronization of the communication. Match communication format of the opposing device and SSIF-2 when SSIF-2 is slave communication (SSICR.MST=0). SSIF-2 uses the signal of SSILRCK/SSIFS only as a communication start trigger.

35.4.3 Communication Modes

SSIF-2 supports the following communication modes. **Table 35.15** lists the control bits that are not available with each communication mode. See **Section 35.4.3.1** to **Section 35.4.3.5** for details of these communication modes.

Table 35.14 Communication Modes

Communication Mode	SSICR.MST Bit	SSICR.REN Bit	SSICR.TEN Bit
Slave-mode transmission	0	0	1
Slave-mode reception	0	1	0
Slave-mode transmission and reception	0	1	1
Master-mode transmission	1	0	1
Master-mode reception	1	1	0
Master-mode transmission and reception	1	1	1

Table 35.15 Control Bits that Cannot be Used in Each Communication Mode

Communication Mode Control Bit	Slave-mode Reception	Slave-mode Transmission	Slave-mode Transmission and Reception	Master-mode Reception	Master-mode Transmission	Master-mode Transmission and Reception
SSICR.CKS	Invalid	Invalid	Invalid	Available	Available	Available
SSICR.CKDV	Invalid	Invalid	Invalid	Available	Available	Available
SSICR.MUEN	Invalid	Available	Available	Invalid	Available	Available
SSICR.TEN	Invalid	Available	Available	Invalid	Available	Available
SSICR.REN	Available	Invalid	Available	Available	Invalid	Available
SSIFCR.AUCKEN	Invalid	Invalid	Invalid	Available	Available	Available
SSIFCR.BCKNCE	Available	Available*1	Available*2	Invalid	Invalid	Invalid
SSIFCR.LRCKNCE	Available	Available*1	Available*2	Invalid	Invalid	Invalid
SSIFCR.RXDNCE	Available	Invalid	Available	Invalid	Invalid	Invalid
SSIFCR.TIE	Invalid	Available	Available	Invalid	Available	Available
SSIFCR.RIE	Available	Invalid	Available	Available	Invalid	Available
SSIFCR.TFRST	Invalid	Available	Available	Invalid	Available	Available
SSIFCR.RFRST	Available	Invalid	Available	Available	Invalid	Available
SSIOFR.BCKASTP	Invalid	Invalid	Invalid	Available	Available	Available
SSIOFR.LRCONT	Invalid	Invalid	Invalid	Available	Available	Available
SSIOFR.OMOD	Available	Available	Available	Available	Available	Available
SSISCR.TDES	Invalid	Available	Available	Invalid	Available	Available
SSISCR.RDFS	Available	Invalid	Available	Available	Invalid	Available

Note 1. Disabled when SSICR.DEL = ‘1’

Note 2. Note 2. Disabled during transmission (SSICR.TEN = ‘1’) when SSICR.DEL = ‘1’

35.4.3.1 Slave-Mode Communication

SSIF-2 operates in slave mode with SSICR.MST = 0. Supply the SSIBCK signal and the SSILRCK/SSIFS signal used for the serial data communication from external devices. If these signals do not match the communication format set for SSIF-2, operation is not guaranteed.

35.4.3.2 Master-Mode Communication

SSIF-2 operates in master mode with SSICR.MST = 1. The SSIBCK signal and SSILRCK/SSIFS signal used for the serial data communication are internally generated from the audio clock. These signals use the format according to the setting of SSIF-2. If the communication format of another device working as the slave device does not match the communication format set for SSIF-2, operation is not guaranteed.

35.4.3.3 Transmission

SSIF-2 transmits serial data to opposing device when SSICR.TEN = 1 and SSICR.REN = 0. If the communication format of the opposing device does not match the communication format set for SSIF-2, operation is not guaranteed.

35.4.3.4 Reception

SSIF-2 receives serial data from opposing device when SSICR.TEN = 0 and SSICR.REN = 1. If the communication format of the opposing device does not match the communication format set for SSIF-2, operation is not guaranteed.

35.4.3.5 Transmission and Reception

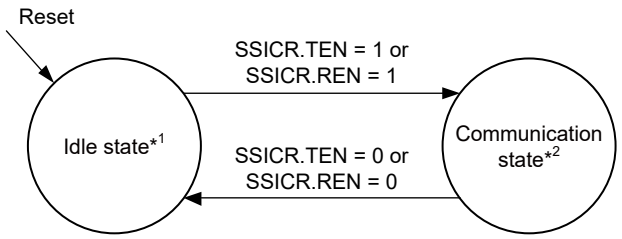
SSIF-2 transmits and receives serial data to and from opposing device when SSICR.TEN = 1, SSICR.REN = 1. If the communication format of opposing device does not match the communication format set for SSIF-2, operation is not guaranteed.

35.5 Operation

35.5.1 Operational State

SSIF-2 has the following two main operation states (**Figure 35.42**).

- Idle state (SSISR.IIRQ = 1)
- Communication state (SSISR.IIRQ = 0)



Note 1. See **Section 35.5.1.1** for details of the idle state.

Note 2. See **Section 35.5.1.2** for details of the communication state.

Figure 35.42 SSIF-2 State Transition

35.5.1.1 Idle State

In this state, communication of SSIF-2 is halted. However, when SSICR.MST = 1, output of BCK and LRCK/FS to external pins is enabled depending on the settings in the SSIOFR.BCKASTP bit and the SSIOFR.LRCONT bit (**Table 35.16**). This function is common to all formats.

Table 35.16 Output from External Pins in the Idle State

			Output from Pins		
SSICR.MST	SSIOFR.BCKASTP	SSIOFR.LRCONT	SSIBCK	SSILRCK/SSIFS	SSITxD
0	—	—	Stop	Stop	Stop
1	0	0	Supply	Stop	Stop
1	0	1	Supply	Supply	Stop
1	1	0	Stop	Stop	Stop
1	1	1	Stop	Supply	Stop

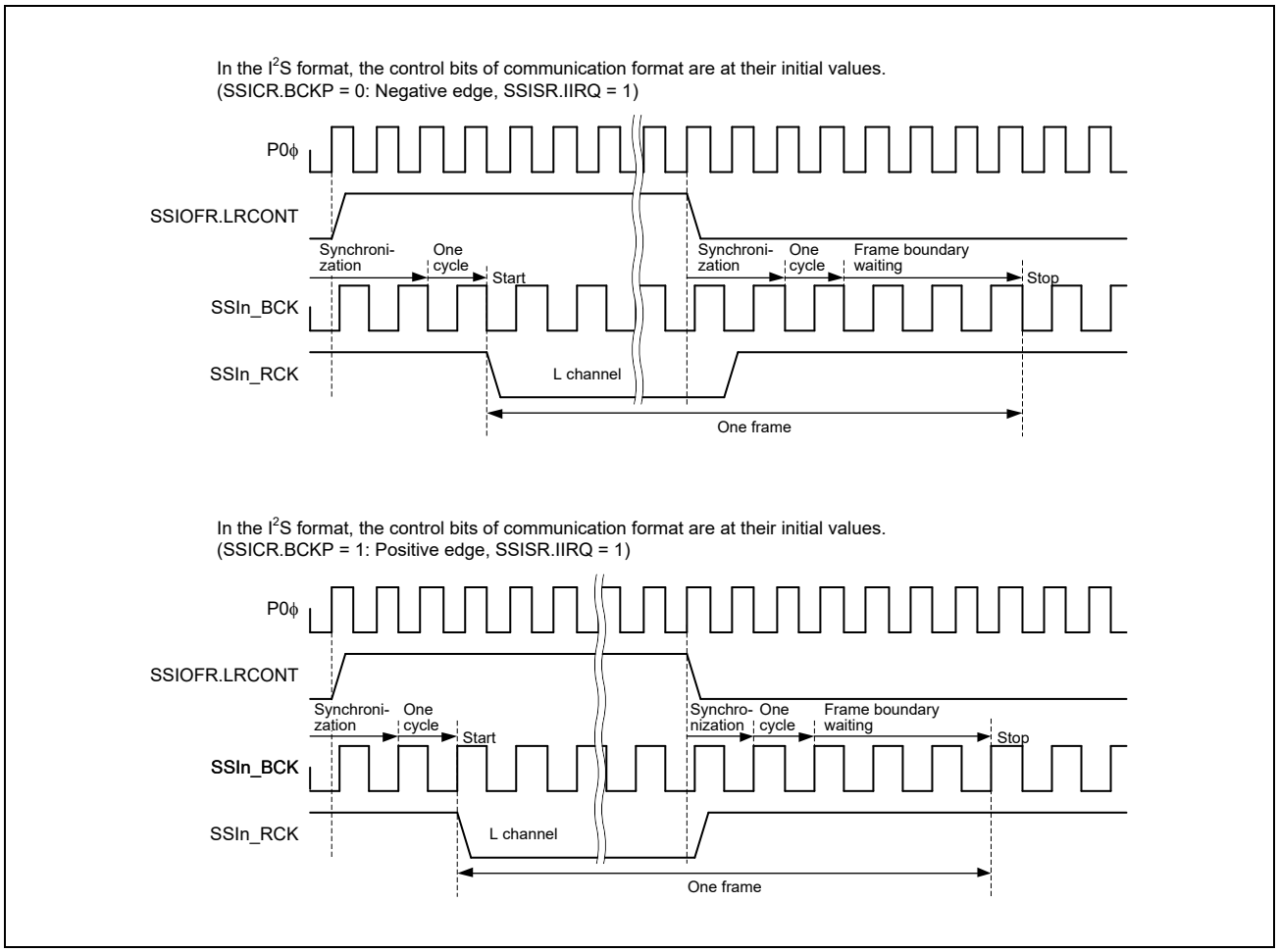


Figure 35.43 Example of LRCK/FS Continuation Release with SSIOFR.LRCONT

NOTE

To stop the output to the SSILRCK/SSIFS pin with SSIOFR.LRCONT when SSIF-2 is in the idle state in master- mode communication (SSICR.MST = 1), note the following: The output stops when 0 is written from 1 to SSIOFR.LRCONT. (**Figure 35.43**). Make sure that the remote device is not affected.

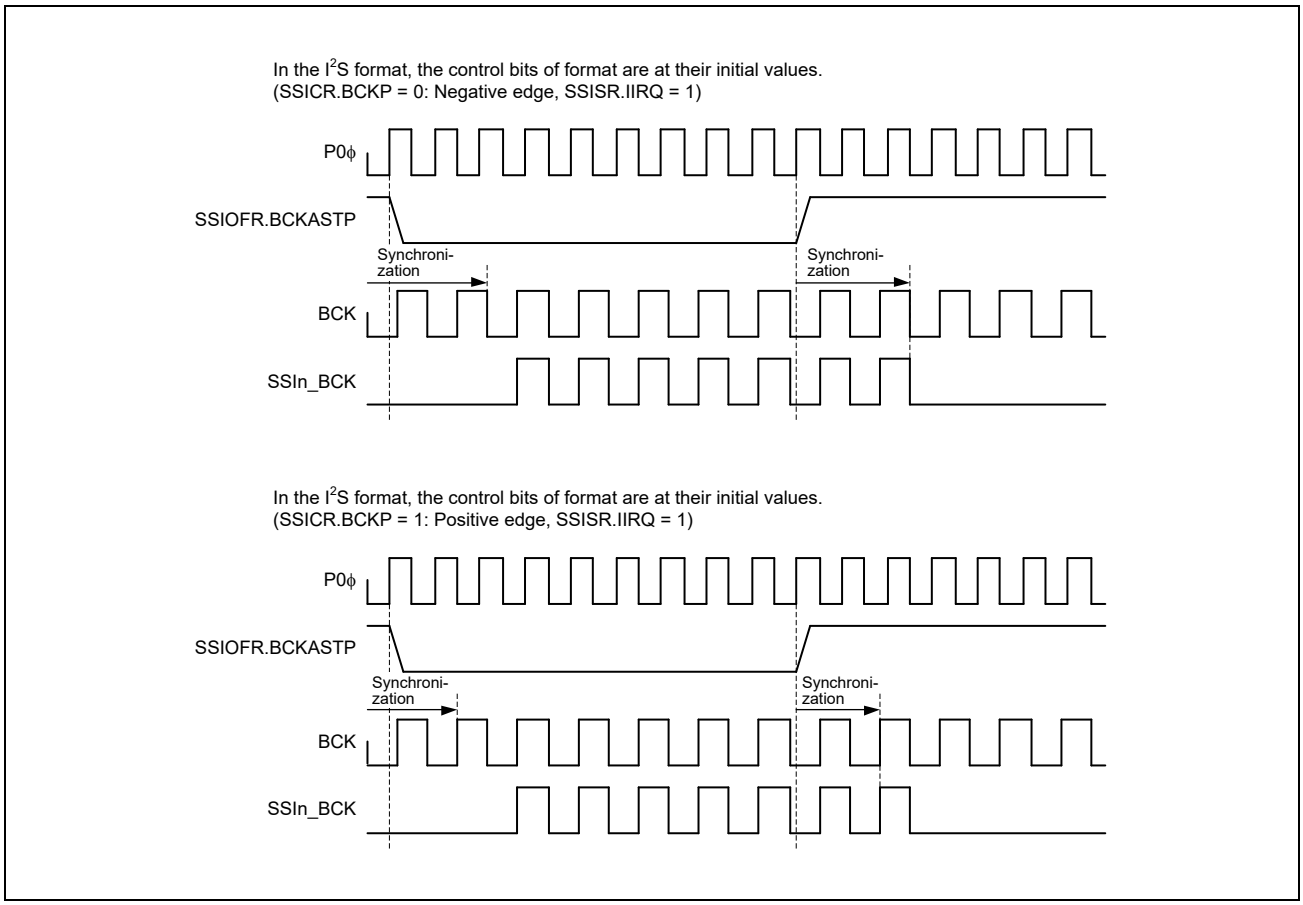


Figure 35.44 Example of Stopping SSIBCK with SSIOFR.BCKASTP

NOTE

To stop the output to the SSIBCK pin with SSIOFR.BCKASTP in master-mode communication (SSICR.MST = 1) and while SSIF-2 is in the idle state, note the following; By writing 0 to SSIOFR.BCKASTP while it is 1, the output stops immediately (**Figure 35.44**). So, make sure that the remote device is not affected.

35.5.1.2 Communication States

In this state, SSIF-2 is during communication. **Figure 35.45** shows transitions of communication states and **Table 35.17** lists the conditions for transition. If the transition condition is not satisfied, the state does not transit.

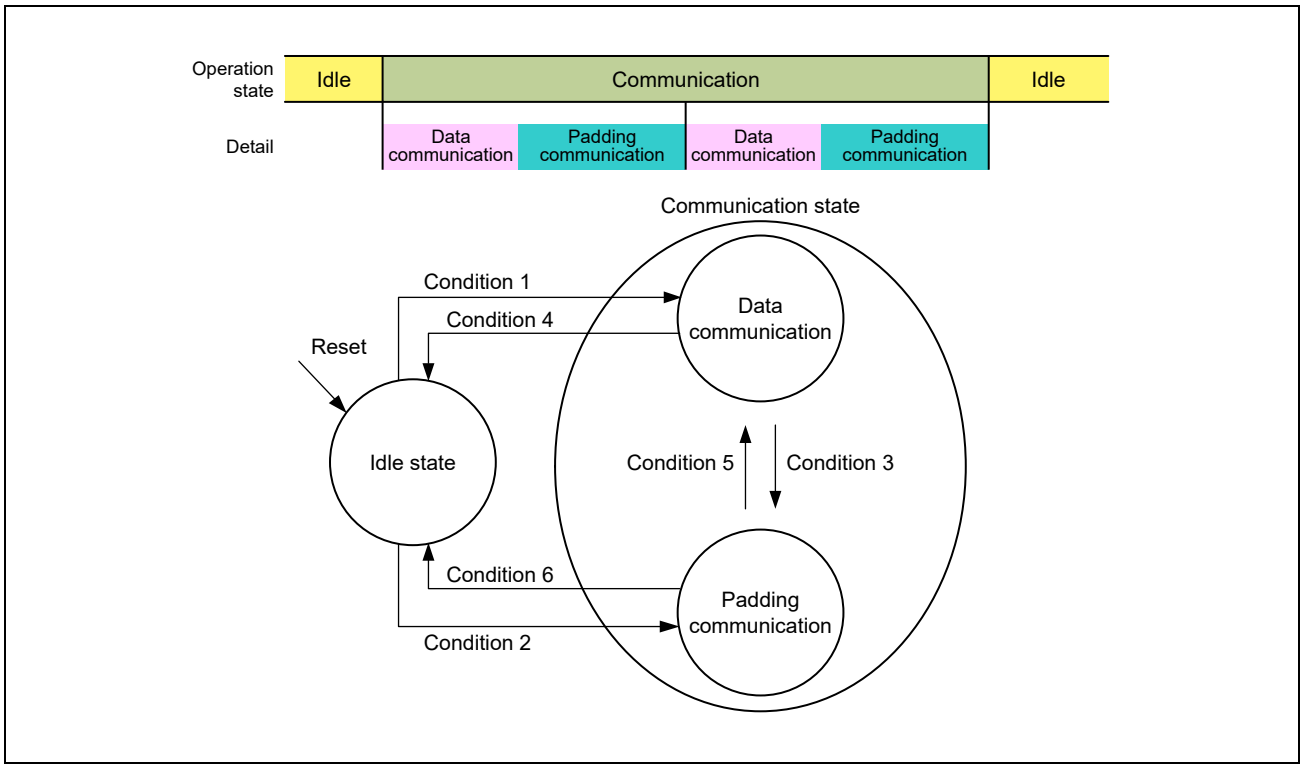


Figure 35.45 Communication State Transition

Table 35.17 Condition for Communication State Transition

Condition No.	Condition for Transition
1	Writing SSICR.TEN = 1 or SSICR.REN = 1 while SSICR.SDTA = 0 or in the setting without padding bits.
2	Writing SSICR.TEN = 1 or SSICR.REN = 1 while SSICR.SDTA = 1 and in the setting with padding bits.
3	The following all of three conditions are satisfied: ● SSICR.TEN = 1 or SSICR.REN = 1 ● In the setting without padding bits ● Transfer data words of the last data bit is completed.
4	The following all of two conditions are satisfied: ● SSICR.SDTA = 1 or without padding bits ● Transfer data words of the last data bit of one frame is completed while SSICR.TEN = 0 and SSICR.REN = 0
5	Transfer of the last padding bit is completed while SSICR.TEN = 1 or SSICR.REN = 1
6	The following all of two conditions are satisfied: ● SSICR.SDTA = 0 and with padding bits ● Transfer of the last data bit of one frame is completed while SSICR.TEN = 0 and SSICR.REN = 0.

See **Table 35.13** for the setting with/without padding bits.

(1) Data Communication State

In this state, SSIF-2 is during communication. Data of the data word length set with the SSICR.DWL[2:0] bits is transmitted, received, or transmitted and received.

State Transition in the Setting without Padding Bits

During communication (SSISR.IIRQ = 0), SSIF-2 is during data communication for all the time (**Figure 35.46**). By disabling transmission and reception (SSICR.TEN = 0 and SSICR.REN = 0), SSIF-2 transits to the idle state (**Figure 35.47**).

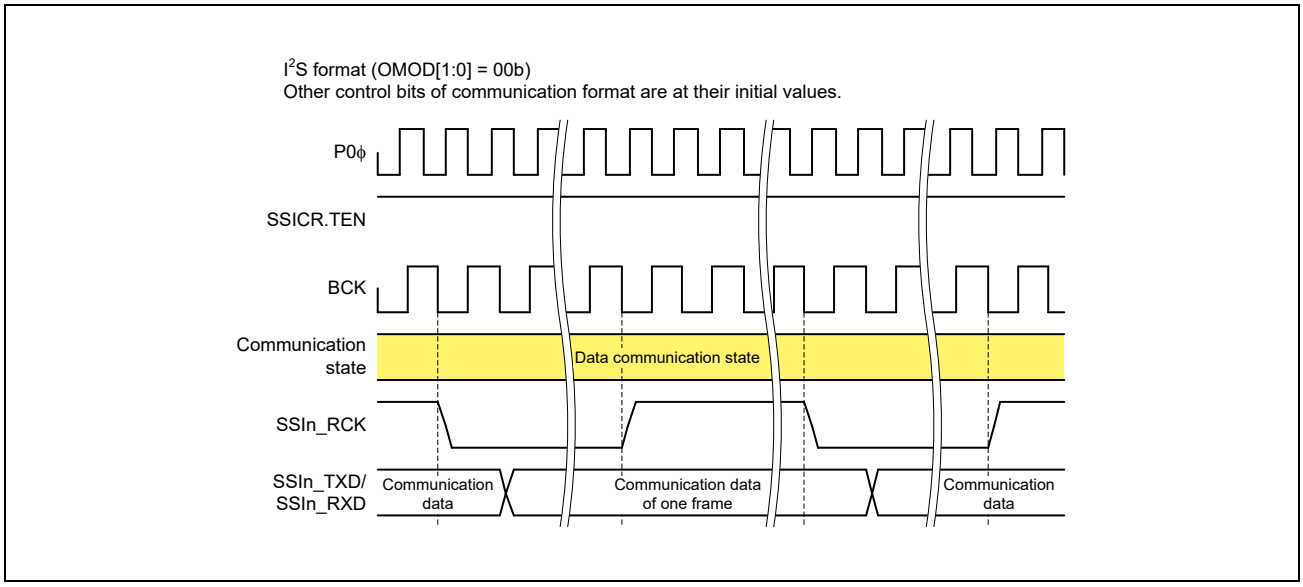


Figure 35.46 Continuation of the Data Communication State

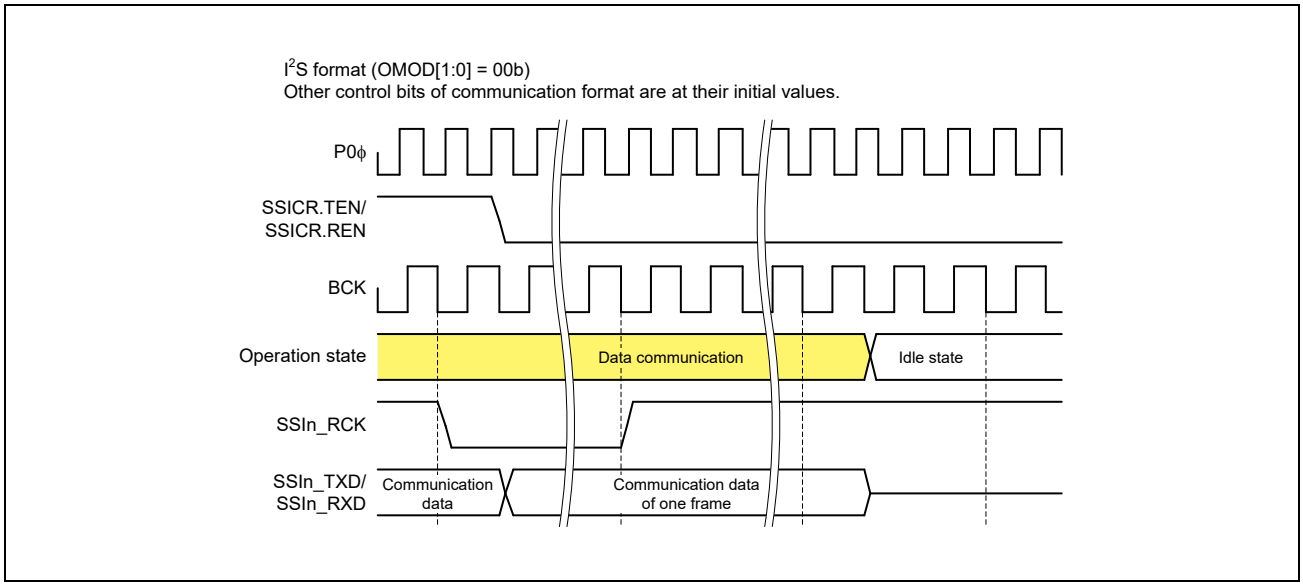


Figure 35.47 Halt from the Data Communication State (without Padding Bits)

State Transition in the Setting with Padding Bits

During communication (SSISR.IIRQ = 0), after completing transfer of the last bit of a data word, SSIF-2 transits to the padding communication state from data communication state (**Figure 35.48**). By disabling transmission and reception (SSICR.TEN = 0 and SSICR.REN = 0) while SSICR.SDTA = 1, SSIF-2 transits to the idle state from data communication state when the communication stop is done. (**Figure 35.49**).

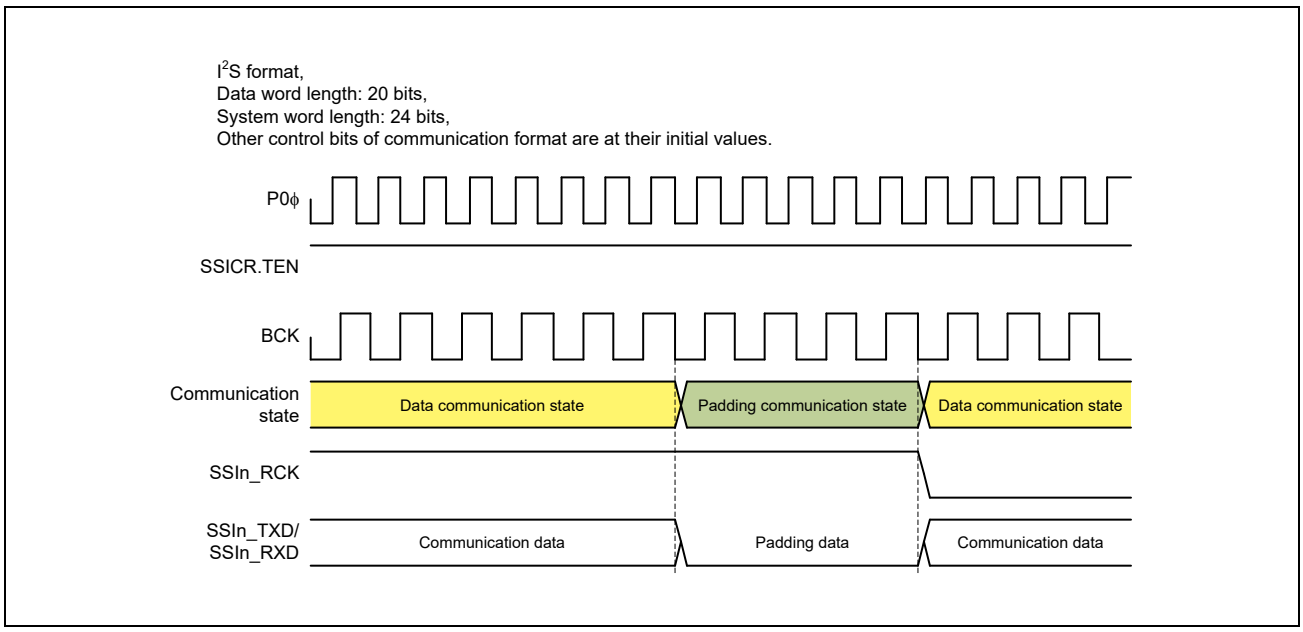


Figure 35.48 Transition from Data Communication to Padding Communication

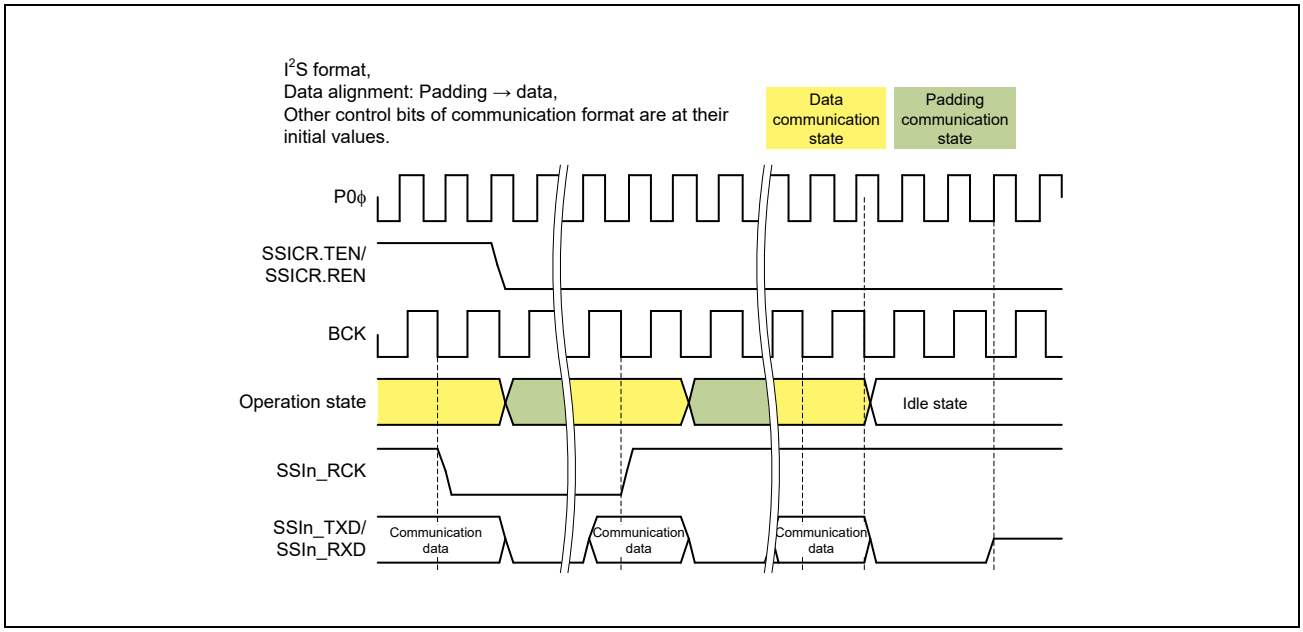


Figure 35.49 Halt from Data Communication (with Padding Bits)

(2) Padding Communication State

In this state, SSIF-2 is during communication. The padding bits set with the SSICR.SWL[2:0] bits and SSICR.DWL[2:0] bits are transmitted, received, or transmitted and received.

State Transition in the Setting with Padding Bits

During communication (SSISR.IIRQ = 0), after completing transfer of the last padding bit, SSIF-2 transits to data communication state (**Figure 35.48**). By disabling transmission and reception (SSICR.TEN = 0 and SSICR.REN = 0) while SSICR.SDTA = 0, SSIF-2 transits to the idle state from padding communication when the communication stop is done. (**Figure 35.50**).

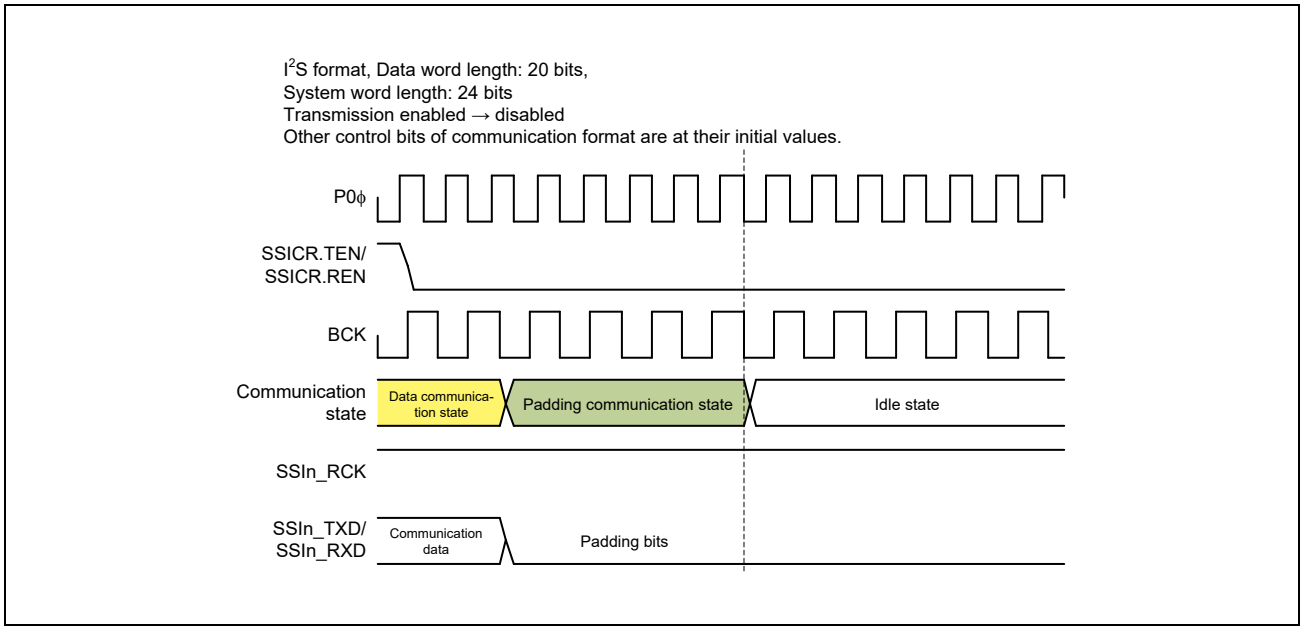


Figure 35.50 Halt from the Padding Communication State

35.5.2 Communication Operation

Figure 35.51 shows the communication flow of SSIF-2.

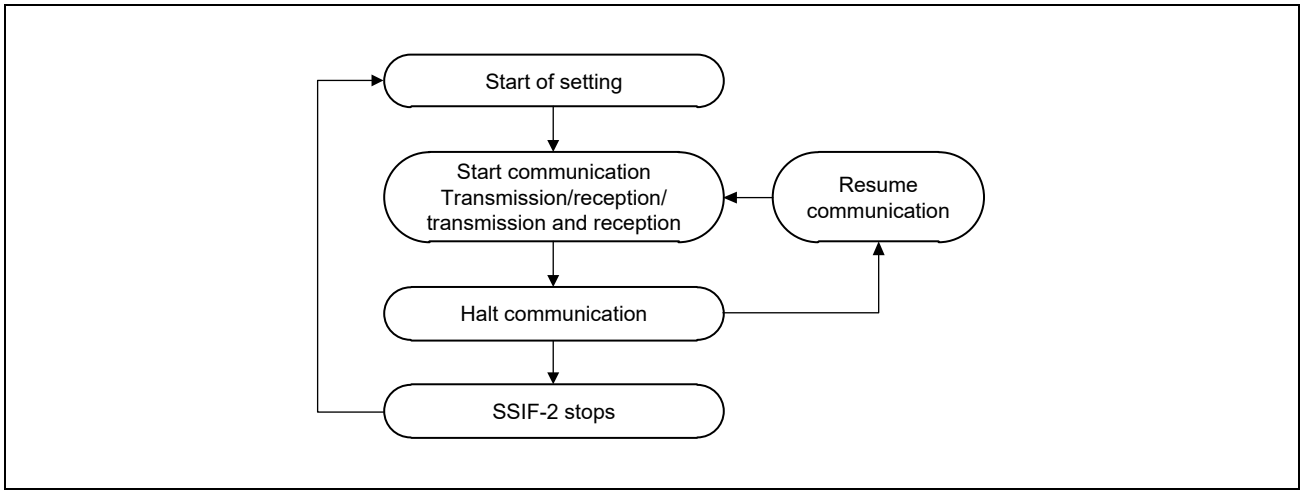
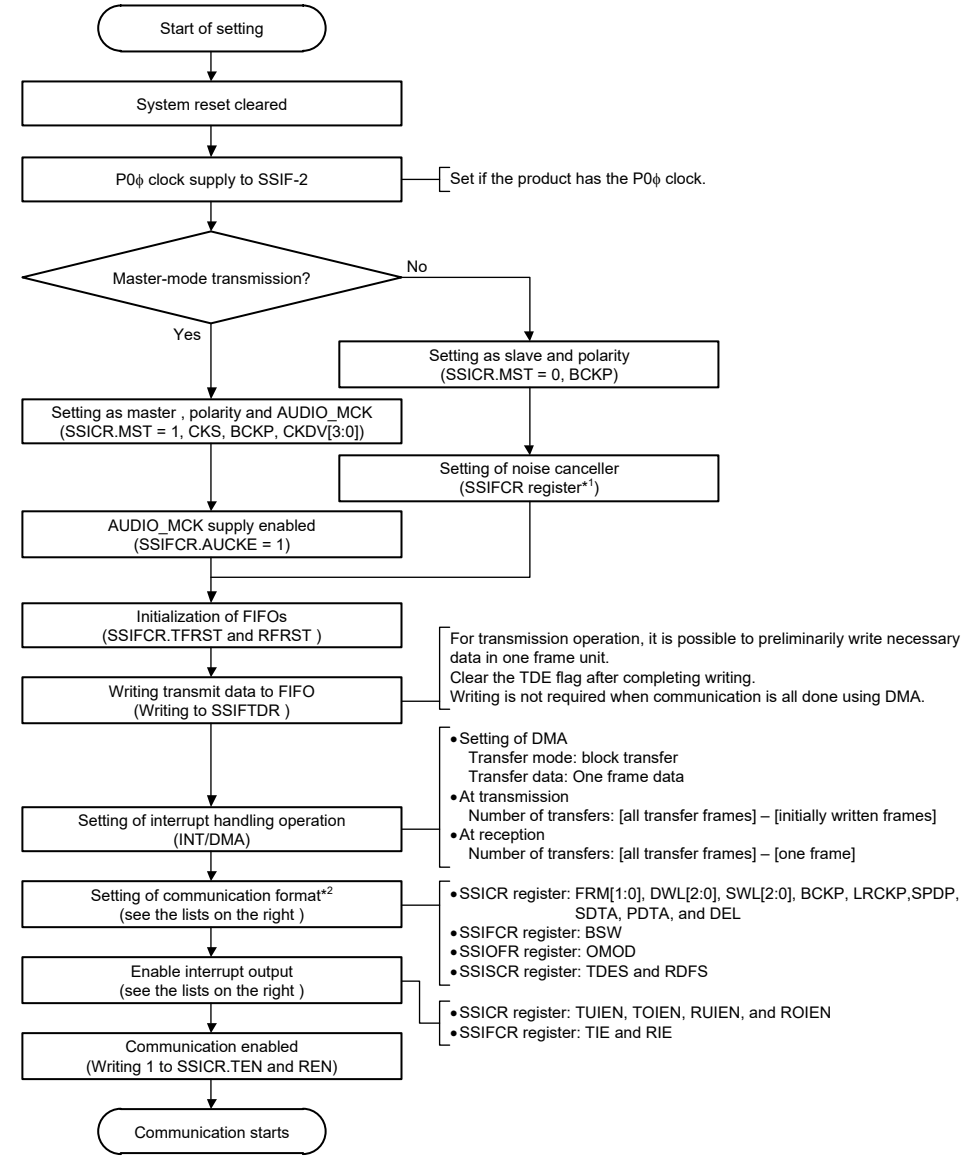


Figure 35.51 SSIF-2 Communication Operation

The procedure of each operation is described in **Section 35.5.2.1, Start Communication** to **Section 35.5.2.7, Resume Communication**.

35.5.2.1 Start Communication

This section describes how to start communication of SSIF-2. **Figure 35.52** shows the procedure to start communication. Be sure to follow the procedure. See **Section 35.5.2.2** for transmission operation and **Section 35.5.2.3** for reception operation.



Note 1. BCKNCE, LRCKNCE, LRCKNCE
Change all the settings of the noise cancellation to the same setting.

Note 2. Set SSICR.MUEN, SSIOFR.BCKASTP, and SSIOFR.LRCONT after completing the setting of the communication format.

Figure 35.52 Procedure to Start Communication (CPU Procedure)

For SSIF-2 communication, continuous communication is possible with DMA interrupt handling. For transmission, write 1 to SSIFCR.TIE, SSICR.TUIEN, and SSICR.TOIEN. For reception, write 1 to SSIFCR.RIE, SSICR.RUIEN, and SSICR.ROIEN.

35.5.2.2 Transmission

Follow the transmission operation procedure (**Figure 35.53**) while the transmission is operating.

Transmission starts in synchronization with the start trigger of the SSILRCK/SSIFS signal while transmission is enabled (SSICR.TEN = 1 and SSICR.REN = 0) and while there is one or more frame of serial data in the transmit FIFO data register (SSIFTDR).

SSIF-2 outputs a transmit data empty interrupt to DMA according to the setting condition of TDE (SSISCR.TDES) and transmit data empty interrupt (SSIFCR.TIE) being enabled.

This interrupt requests writing to the transmit FIFO data register (SSIFTDR). By writing to SSIFTDR at each generation of an interrupt, serial data transmission is realized.

In the procedure to start communication, set DMA handling for transmit data empty interrupts as writing to the transmit FIFO data register (SSIFTDR). As a result, SSIF-2 is able not to mediate CPU and to transmit continuous data. When it becomes empty capacity of transmission FIFO data register set with SSISCR.TDES, transmission data empty interrupt is generated. Set the writing number of times according to empty capacity of transmission FIFO data register that transmission data empty interrupt shows. In a case of an error, stop receive procedure (**Section 35.5.2.5**) and then take the error procedure (**Section 35.5.2.6**).

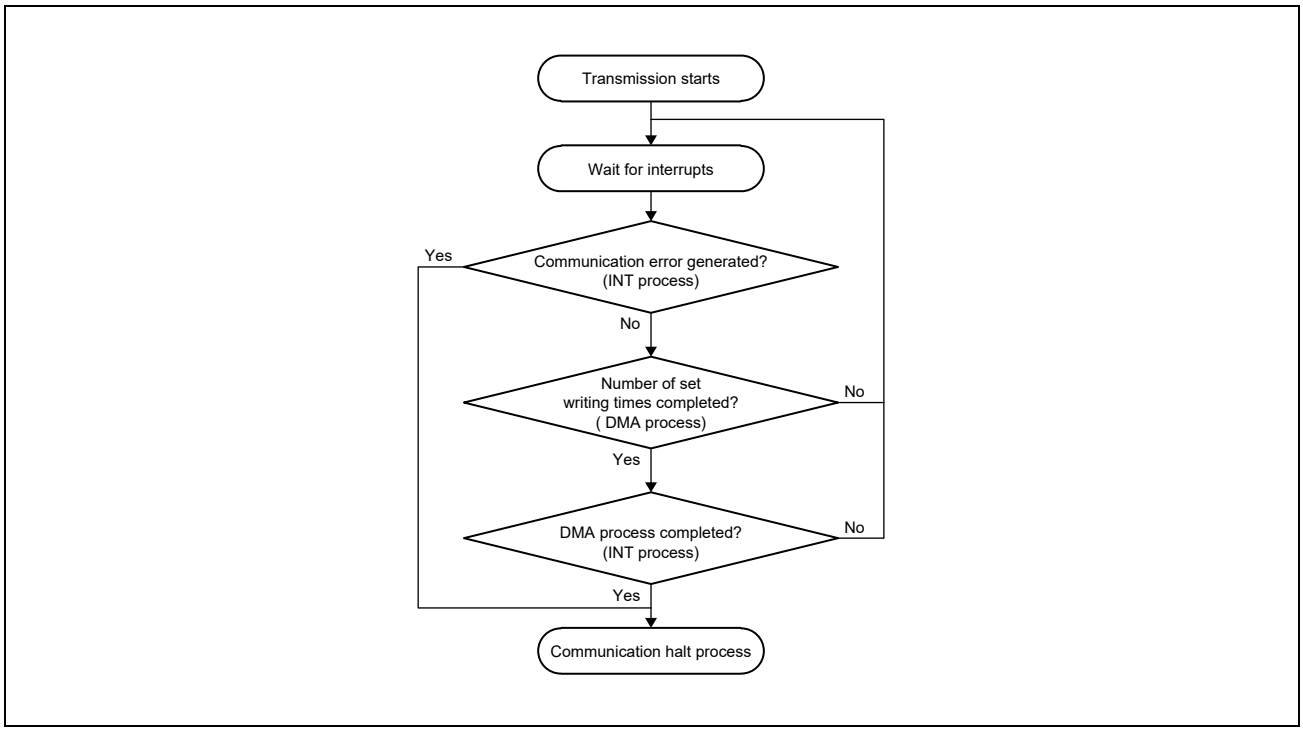


Figure 35.53 Transmission Procedure

NOTE

The communication flow defined in SSIF-2 uses the DMA. In the case the DMA is not used, observe one of SSIFSR.TDE, and write it in SSIFTDR. Follow empty capacity of transmission FIFO data register set with SSISCR.TDES about the number of times written in SSIFTDR detecting 1 of SSIFSR.TDE. After writing the transmission data corresponding to the free space to SSIFTDR, polling ‘1’ of SSIFSR.TDE and repeating this enables continuous transmission operation.

35.5.2.3 Reception

Follow the reception operation procedure (**Figure 35.54**) while the reception is operating.

Reception starts in synchronization with the start trigger of the SSILRCK/SSIFS signal while reception is enabled (SSICR.TEN = 0 and SSICR.REN = 1). SSIF-2 outputs a receive data full interrupt to DMA according to the setting of RDF (SSISCR.RDFS) and receive data empty interrupt (SSIFCR.RIE) being enabled. This interrupt requests reading from the receive FIFO data register (SSIFRDR). By reading from SSIFRDR at each generation of an interrupt, serial data reception is realized.

In the procedure to start communication, set DMA handling for receive data full interrupts as reading from receive FIFO data register (DDIFRDR). As a result, SSIF-2 is able not to mediate CPU and to receive continuous data. If the data capacity of reception FIFO data register set with SSISCR.RDFS is stored, receive data full interrupt is generated. Set the writing number of times according to the data capacity of reception FIFO data register that receive data full interrupt shows.

In the case of an error, stop the receive operation (**Section 35.5.2.6**) and then take the error procedure (**Section 35.5.2.6**).

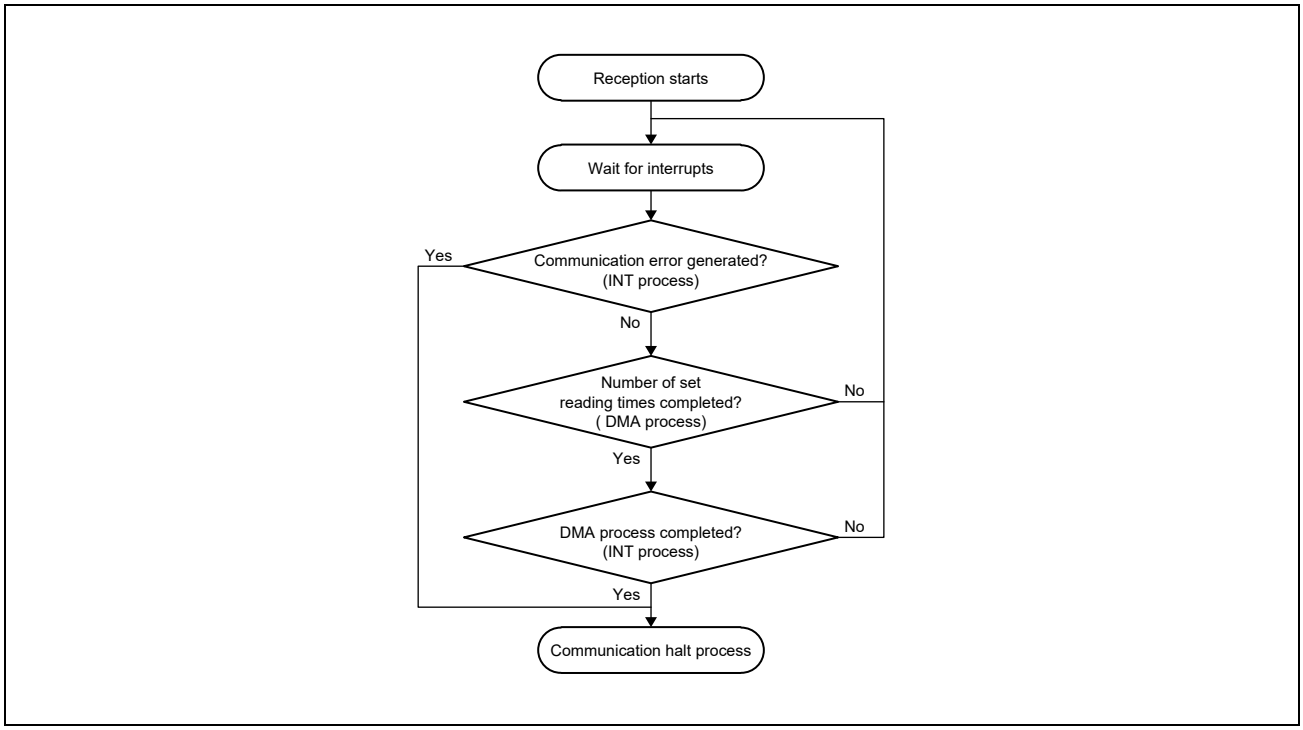


Figure 35.54 Reception Procedure

NOTE

The communication flow defined in SSIF-2 uses the DMA. In the case the DMA is not used, observe 1 of SSIFSR.RDF, and read SSIFRDR. Follow the receive data capacity of reception FIFO data register set with SSISCR.RDFS about the number of times in which SSIFRDR is read detecting 1 of SSIFSR.RDF. After reading the received data from SSIFRDR, Continuous reception operation is possible by polling ‘1’ of SSIFSR.RDF and repeating this.

35.5.2.4 Transmission and Reception

Transmission and reception starts in synchronization with the start trigger of the SSILRCK/SSIFS signal while transmission and reception are enabled (SSICR.TEN = 1 and SSICR.REN = 1) and while there is one or more frames of serial data in the transmit FIFO data register (SSIFTDR). The transmission and reception operation of SSIF-2 can be transmitted and received continuously by each doing the procedure for showing in the transmission operation (**Section 35.5.2.2**) and the reception operation (**Section 35.5.2.3**). As transmission and reception are independent in SSIF-2, see sections of transmission and reception respectively. See **Section 35.5.2.5, Halt Communication** to halt communication.

35.5.2.5 Halt Communication

This section describes how to halt communication of SSIF-2. **Figure 35.55** shows the procedure to halt communication. Be sure to follow the procedure.

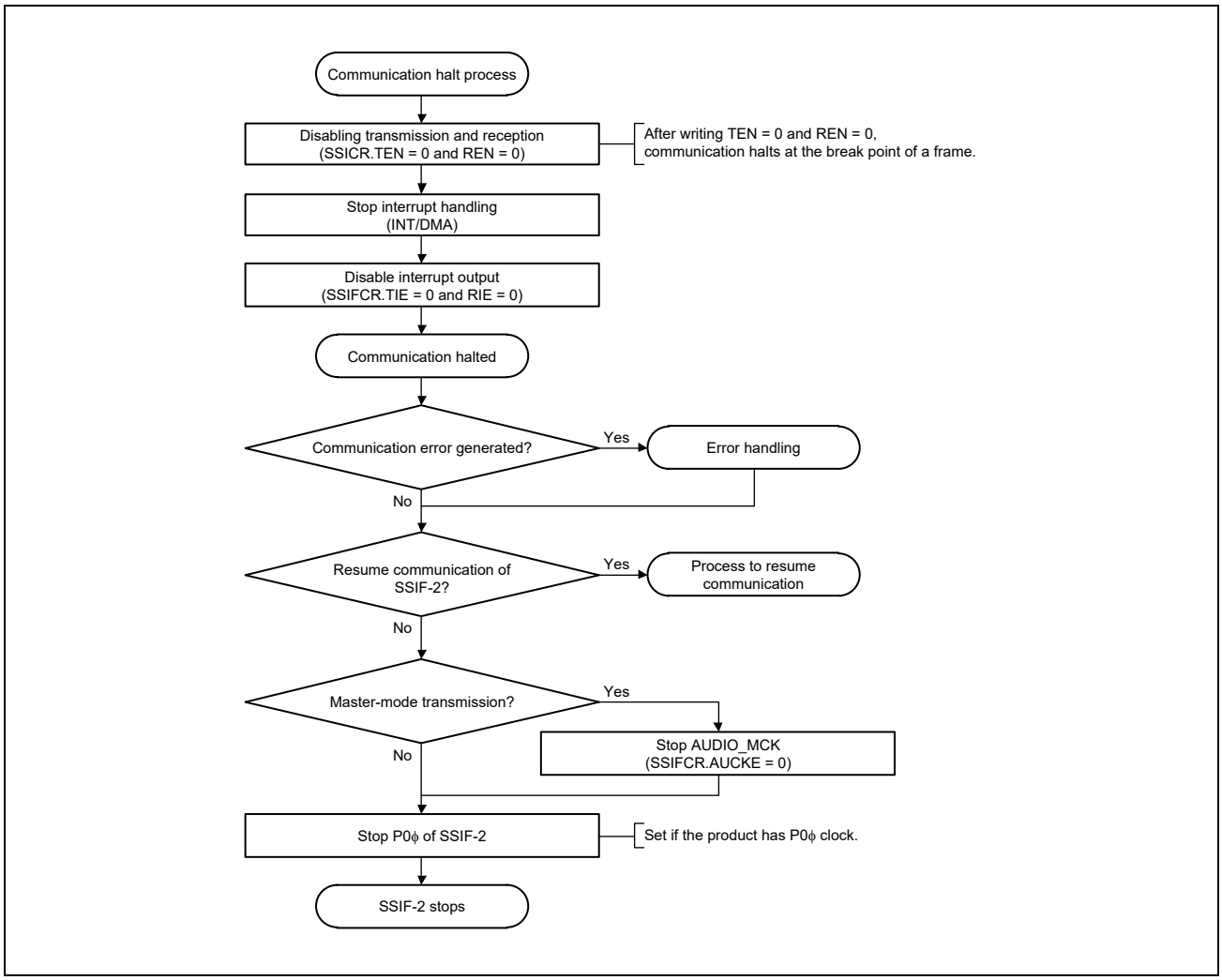


Figure 35.55 Procedure to Halt Communication (CPU Procedure)

To halt the communication of SSIF-2, supply of the following clocks are required until the SSISR.IIRQ bit indicates an idle state.

- Input clock from the SSIBCK pin when SSICR.MST = 0
- AUDIO_MCK when SSICR.MST = 1

To resume communication of SSIF-2 in the previous setting, see **Section 35.5.2.7, Resume Communication**.

NOTE

When communication of SSIF-2 is halted according to the procedure to halt communication in **Figure 35.55**, resume communication according to the procedure to start communication in **Figure 35.52**.

35.5.2.6 Error Handling

SSIF-2 has the following four errors.

- Transmit underflow error
- Transmit overflow error
- Receive underflow error
- Receive overflow error

When an underflow error or overflow error is generated, SSIF-2 need to be restarted. Take the error procedure (**Figure 35.56**) according to the procedure to halt communication (**Figure 35.55**).

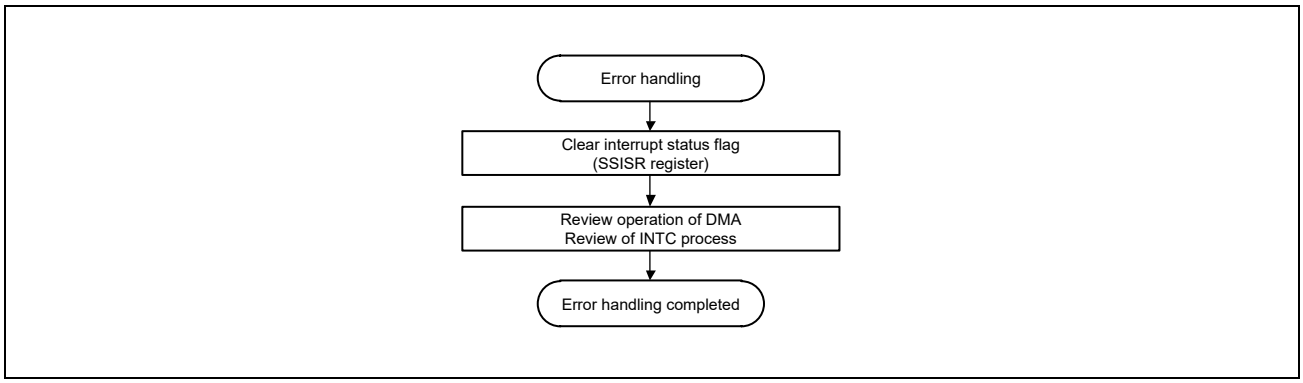


Figure 35.56 Error Handling Procedure

Four error operations are described as follows. When the interrupt output enable bit of the SSICR register is enabled and error flags are set, an error interrupt is generated. See descriptions of flags in **Section 35.4.1.2, Status Register (SSISR)** for the setting conditions of error flags.

(1) Transmit Underflow Error

In the case a transmit underflow error is generated during transmission (SSICR.TEN = 1, SSISR.IIRQ = 0), review the number of write operations to the transmit FIFO data register (SSIFTDR) according to serial data empty interrupts. After generation of a transmit underflow error, data to be transmitted from SSIF-2 are 0 output. To transmit data written to SSIFTxD pin normally, run the procedure to halt communication (**Figure 35.55**) and then take the error procedure (**Figure 35.56**). When this error is generated, serial data is consumed as usual. To resume transmission of SSIF-2, write data from the beginning of one frame.

(2) Transmit Overflow Error

In the case a transmit overflow error is generated, review the number of write operations to the transmit FIFO data register (SSIFTDR) according to transmit data empty interrupts. Data is not written to SSIFTDR where a transmit overflow error is generated. This error is generated regardless of while the transmission is operating. To resume transmission of SSIF-2, take the procedure to halt communication (**Figure 35.55**) and then take the error procedure (**Figure 35.56**). When communication is resumed, consider the lost serial data.

(3) Receive Underflow Error

In the case a receive underflow error is generated, review the number of reading operations from the receive FIFO data register (SSIFRDR) according to receive data full interrupts. The value read from SSIFRDR is undefined where a receive underflow error is generated. This error is generated regardless of while the reception is operating. To resume reception of SSIF-2, take the procedure to halt communication (**Figure 35.55**) and then take the error procedure (**Figure 35.56**).

(4) Receive Overflow Error

In the case a receive overflow error is generated during reception (SSICR.REN = 1, SSISR.IIRQ = 0), review the number of reading operations from the receive FIFO data register (SSIFRDR) according to receive data full interrupts. Data is not stored in SSIFRDR when a receive overflow error is generated. To resume reception of SSIF-2, take the procedure to halt communication (**Figure 35.55**) and then take the error handling procedure (**Figure 35.56**).

35.5.2.7 Resume Communication

To resume communication of SSIF-2, follow the procedure shown in **Figure 35.57**. After communication is halted, the same setting is applied when it is resumed. To change the settings of clocks and slave/master mode, follow the procedure to start communication (**Figure 35.52**). For communication after resume, see **Section 35.5.2.2** and **Section 35.5.2.3**.

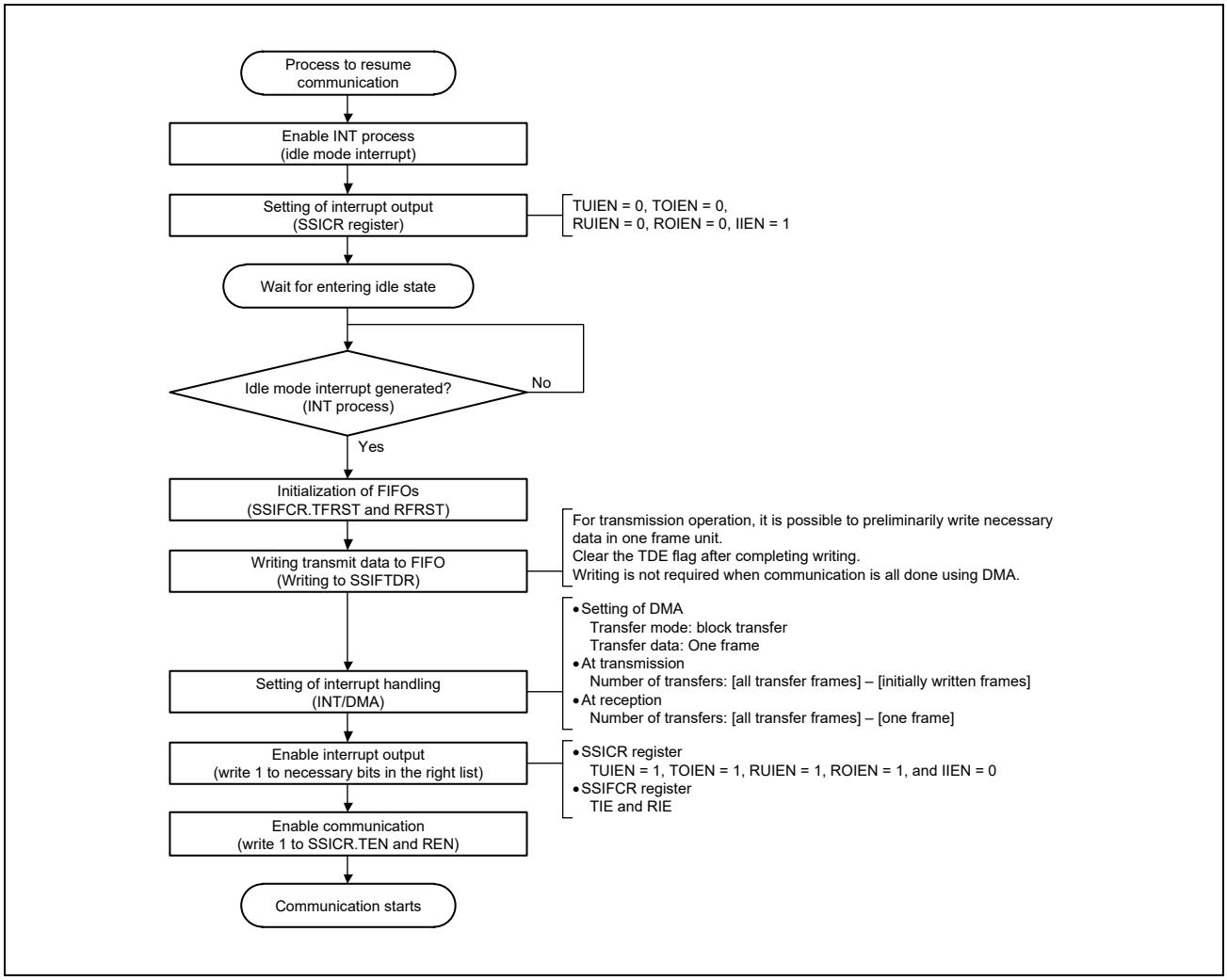


Figure 35.57 Procedure to Resume Communication (CPU Procedure)

35.5.3 Interrupt Sources

Table 35.18 lists the interrupt sources. Set enable/disable of interrupt output of each source with the TUIEN, TOIEN, RUIEN, ROIEN, and IIEN bits in the SSICR register and the TIE and RIE bits in the SSIFCR register.

Table 35.18 Interrupt Sources

Interrupt Source	Interrupt	Interrupt Flag	Interrupt Sense	DMA Activation
INT_ssif_int_req	Transmit underflow interrupt/ Transmit overflow interrupt/ Receive underflow interrupt/ Receive overflow interrupt/ Idle mode interrupt	SSISR.TUIRQ SSISR.TOIRQ SSISR.RUIRQ SSISR.ROIRQ SSISR.IIRQ	Level	Not available
INT_ssif_dma_tx	Transmit data empty interrupt	SSIFSR.TDE	Edge	Available
INT_ssif_dma_rx	Receive data full interrupt	SSIFSR.RDF	Edge	Available
INT_ssif_dma_rt	Receive data full interrupt/ Transmit data empty interrupt	SSIFSR.TDE/SSIFSR.RDF	Edge	Available

35.5.3.1 INT_ssif_int_req Interrupt

This interrupt source combines five interrupts. Enable output of necessary interrupts before using SSIF-2. This interrupt moves from the flag and the output enable permission that five factors have respectively (**Figure 35.58**). To clear an interrupt, set the interrupt enable to 0 or clear the interrupt flag to 0.

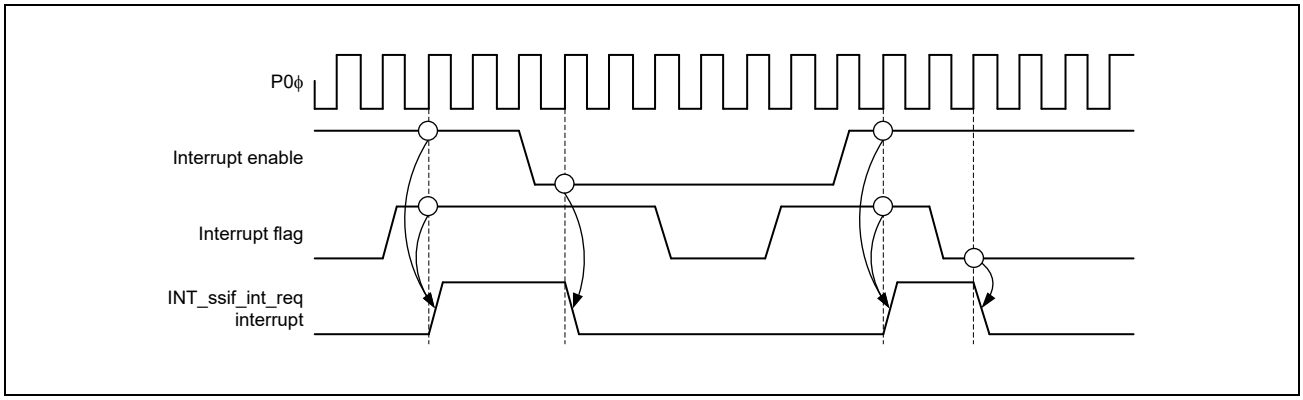


Figure 35.58 Timing Chart of the Common Interrupt Source, INT_ssif_int_req

(1) Transmit Underflow Interrupt

As the transmit underflow interrupt, SSISR.TUIRQ is output while SSICR.TUIEN = 1. To restart transmission (SSICR.TEN = 1), enable the output of this interrupt (SSICR.TUIRQ = 1). In the case this interrupt is generated, take the procedure to halt communication (**Figure 35.55**) and then take the error procedure (**Figure 35.56**).

(2) Transmit Overflow Interrupt

As the transmit overflow interrupt, SSISR.TOIRQ is output while SSICR.TOIRQ = 1. To restart transmission (SSICR.TEN = 1), enable the output of this interrupt (SSICR.TOIRQ = 1). In the case this interrupt is generated, take the procedure to halt communication (**Figure 35.55**) and then take the error procedure (**Figure 35.56**).

(3) Receive Underflow Interrupt

As the receive underflow interrupt, SSISR.RUIRQ is output while SSICR.RUIRQ = 1. To restart reception (SSICR.REN = 1), enable the output of this interrupt (SSICR. RUIRQ = 1). In the case this interrupt is generated, take the procedure to halt communication (**Figure 35.55**) and then take the error procedure (**Figure 35.56**).

(4) Receive Overflow Interrupt

As the receive overflow interrupt, SSISR.ROIRQ is output while SSICR.ROIRQ = 1. To restart reception (SSICR.REN = 1), enable the output of this interrupt (SSICR. ROIRQ = 1). In the case this interrupt is generated, take the procedure to halt communication (**Figure 35.55**) and then take the error procedure (**Figure 35.56**).

(5) Idle Mode Interrupt

As the idle mode interrupt, SSISR.IIRQ is output while SSICR.IIEN = 1. When the thing that the communication has stopped completely is confirmed, this interrupt is used (**Figure 35.57**).

35.5.3.2 INT_ssif_dma_tx Interrupt [full-duplex communication]

As the transmit data empty interrupt, a pulse interrupt is output by the following condition.

- SSIFSR.TDE and SSIFCR.TIE becomes 1.
  SSIF-2 operation: SSIFSR.TDE becomes 1 from 0 while SSIFCR.TIE = 1
  CPU operation: SSIFCR.TIE is changed to 1 from 0 while SSIFSR.TDE = 1

An interrupt suppression function is available for this interrupt to prevent the thing that cannot be taken. When the DMA is busy state (Interrupt cannot be accepted), and the interruption condition occurs, output of this interrupt is held. This interrupt is output after the DMA becomes acceptable (**Figure 35.59**).

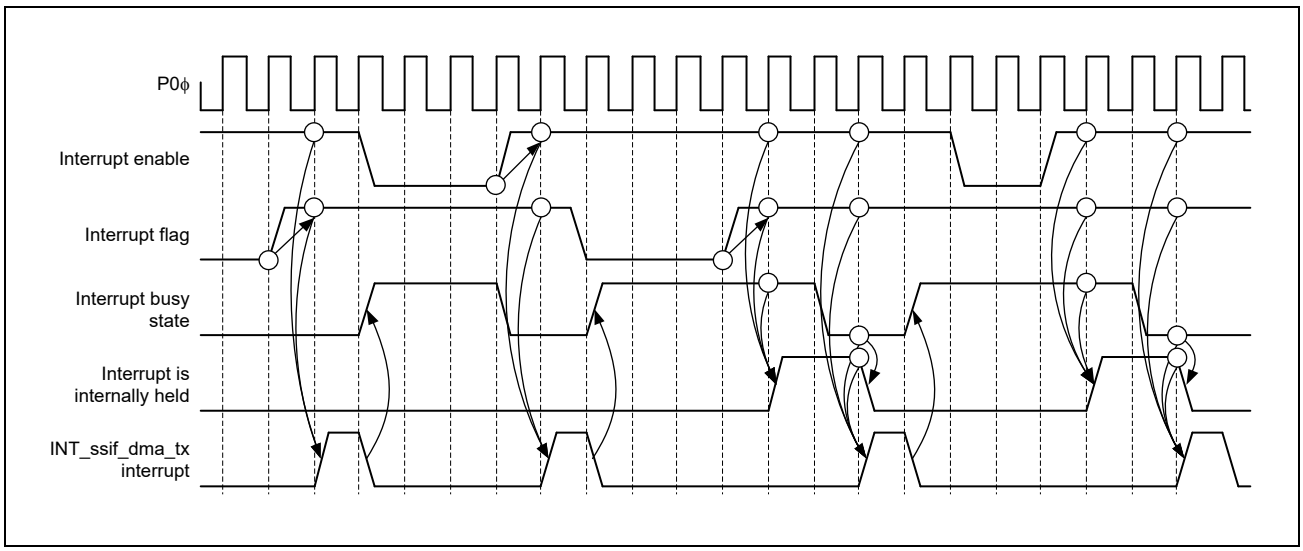


Figure 35.59 INT_ssif_dma_tx Interrupt Timing Chart

35.5.3.3 INT_ssif_dma_rx Interrupt [full-duplex communication]

As the receive data full interrupt, a pulse interrupt is output by the following condition.

- SSIFSR.RDF and SSIFCR.RIE becomes 1.
  SSIF-2 operation: SSIFSR.RDF becomes 1 from 0 while SSIFCR.RIE = 1.
  CPU operation: SSIFCR.RIE is changed to 1 from 0 while SSIFSR.RDF = 1.

An interrupt suppression function is available for this interrupt to prevent the thing that cannot be taken. When the DMA is busy state (Interrupt cannot be accepted), and the interruption condition occurs, output of this interrupt is held. This interrupt is output after the DMA becomes acceptable. It operates in the same way as in **Figure 35.59**.

35.5.3.4 INT_ssif_dma_rt Interrupt [half-duplex communication]

This interrupt is output by two sources, transmit data empty interrupt and receive data full interrupt. When this interrupt is generated, read the interrupt flag and specify the interrupt source.

An interrupt suppression function is available for this interrupt to prevent the thing that cannot be taken. When the DMA is busy state (Interrupt cannot be accepted), and the interruption condition occurs, output of this interrupt is held. This interrupt is output after the DMA becomes acceptable (**Figure 35.59**).

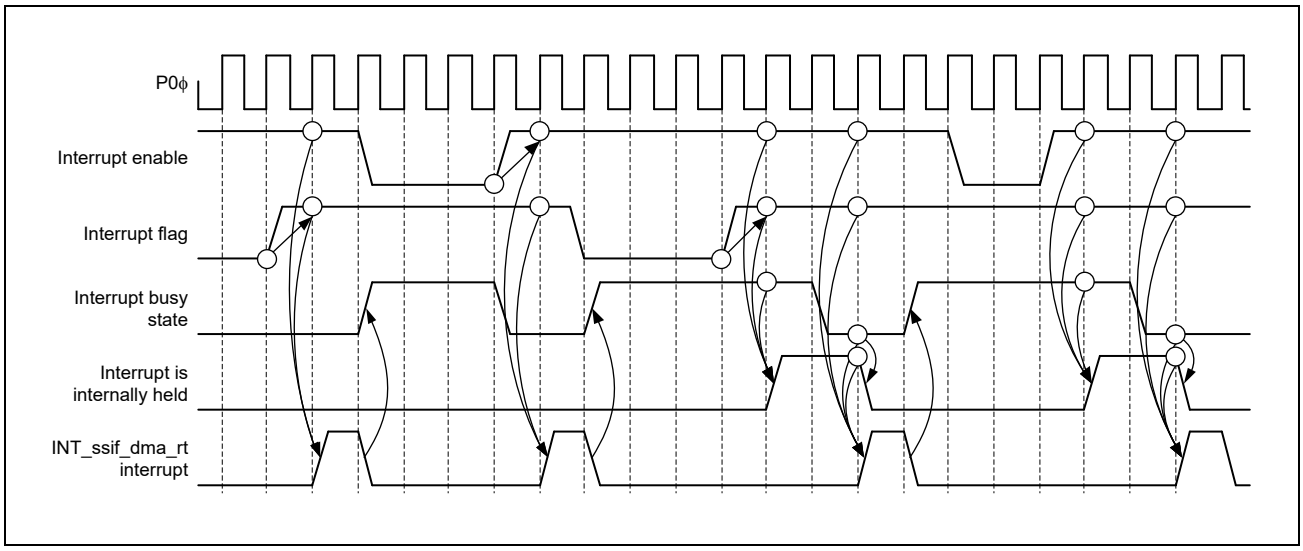


Figure 35.60 INT_ssif_dma_tx Interrupt Timing Chart

35.5.4 Software Resets

SSIF-2 has three software reset bits to reset its states.

- SSIF-2 software reset (SSIFCR.SSIRST)
- Transmit FIFO data register reset (SSICR.TFRST)
- Receive FIFO data register reset (SSIFCR.RFRST)

It explains the procedure of three software reset.

35.5.4.1 Software Reset Procedure

(1) SSIF-2 Software Reset

For the SSIF-2 software reset bit (SSIFCR.SSIRST), follow the procedure shown in **Figure 35.61**. After a reset, the same setting is applied when it is resumed. To change the settings of clocks and slave/master mode, follow the procedure to start communication (**Figure 35.52**). See **Section 35.5.2.2, Transmission** and **Section 35.5.2.3, Reception** respectively for transmission and reception after communication is resumed.

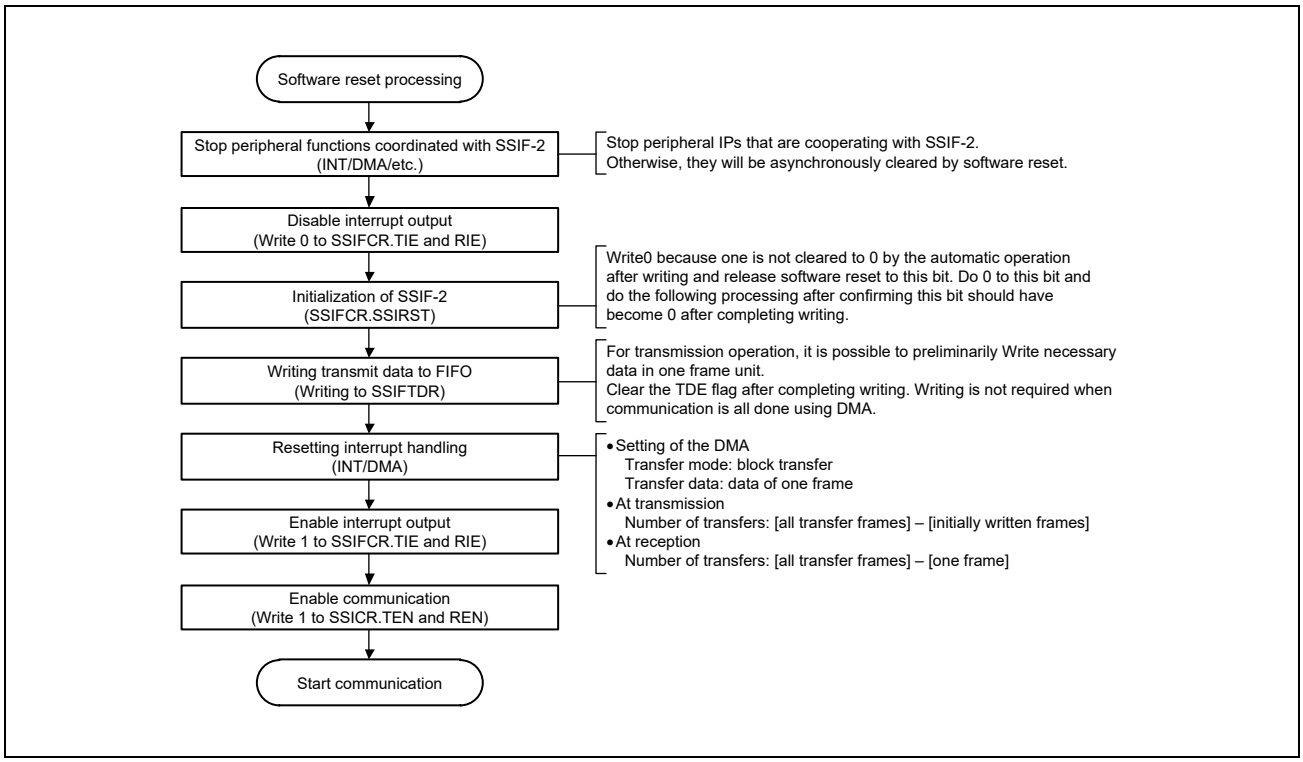


Figure 35.61 Software Reset Procedure (CPU Procedure)

(2) Transmit FIFO Data Register Reset

To initiate a transmit FIFO data register reset, follow the procedure to start communication (**Figure 35.52**) and resume communication (**Figure 35.57**).

(3) Receive FIFO Data Register Reset

To initiate a receive FIFO data register reset, follow the procedure to start communication (**Figure 35.52**) and resume communication (**Figure 35.57**).

35.6 Notes

35.6.1 Notes

35.6.1.1 Attention by Communication Using DMA

SSIF-2 assumes the DMA access as follows.

Table 35.19 Interrupt to Expect DMA Access

Communication Operation	Interrupt	DMA Access	LSI Specification	Remarks
Transmission	INT_ssif_dma_tx	Write to SSIFTDR	Full duplex transmission	
	INT_ssif_dma_rt		Half duplex transmission	
Reception	INT_ssif_dma_rx	Read SSIFRDR	Full duplex transmission	
	INT_ssif_dma_rt		Half duplex transmission	
Transmission and Reception	INT_ssif_dma_tx	Write to SSIFTDR	Full duplex transmission	
	INT_ssif_dma_rx	Read SSIFRDR	Half duplex transmission	

35.6.1.2 Notes for Slave-Mode Communication

(1) ADCKE Control

In slave-mode communication (SSICR.MST = 0), SSIF-2 needs supply of SSIBCK. To stop BCK on the master side, make sure that SSIF-2 is in the idle state (SSISR.IIRQ = 1). If BCK is stopped before SSIF-2 becomes idle, take the procedure to start communication (**Figure 35.52**) or wait for an idle state by taking the procedure to resume communication (**Figure 35.57**).

(2) SSI_RCK

SSIF-2 has the terminal SSI_RCK that shows the synchronization of the communication. Match communication format of the opposing device and SSIF-2 when SSIF-2 is slave communication (SSICR.MST=0). SSIF-2 uses the signal of SSI_RCK only as a communication start trigger.

35.6.1.3 Notes for Master-Mode Communication

(1) ADCKE Control

In master-mode communication (SSICR.MST = 1), SSIF-2 operates with the audio clock (AUDIO_MCK). To stop SSIF- 2 completely, make sure that SSIF-2 is in the idle state (SSISR.IIRQ = 1) and then write 0 to SSIFCR.ADCKE. If 0 is written to SSIFCR.ADCKE before SSIF-2 becomes idle, take the procedure to start communication (**Figure 35.52**).

(2) LRCONT Control

To stop the output to the SSI_RCK pin with SSIOFR.LRCONT when SSIF-2 is in the idle state in master-mode communication (SSICR.MST = 1), note the following: The output stops when 0 is written from 1 to SSIOFR.LRCONT. (**Figure 35.43**). Make sure that the remote device is not affected.

(3) BCKASTP Control

To stop the output to the SSIBCK pin with SSIOFR.BCKASTP in master-mode communication (SSICR.MST = 1) and while SSIF-2 is in the idle state, note the following; By writing 0 to SSIOFR.BCKASTP while it is 1, the output stops immediately (**Figure 35.44**). So, make sure that the remote device is not affected.

NOTE

When the opposing device who is the slave needs the clock of the terminal SSI_BCK before the communication operates, it is not possible to use it.

35.6.1.4 Notes for Communication Flow

(1) When an Error Interrupt is Generated

SSIF-2 has the following four errors.

- Transmit underflow error
- Transmit overflow error
- Receive underflow error
- Receive overflow error

When an underflow error or overflow error is generated, SSIF-2 need to be restarted. Take the error procedure (**Figure 35.56**) according to the procedure to halt communication (**Figure 35.55**).

(a) Transmit Underflow Error

In the case a transmit underflow error is generated during transmission (SSICR.TEN = 1, SSISR.IIRQ = 0), review the number of write operations to the transmit FIFO data register (SSIFTDR) according to serial data empty interrupts. After generation of a transmit underflow error, data to be transmitted from SSIF-2 are 0 output. To transmit data written to SSIFTxD pin normally, run the procedure to halt communication (**Figure 35.55**) and then take the error procedure (**Figure 35.56**). When this error is generated, serial data is consumed as usual. To resume transmission of SSIF-2, write data from the beginning of one frame.

(b) Transmit Overflow Error

In the case a transmit overflow error is generated, review the number of write operations to the transmit FIFO data register (SSIFTDR) according to transmit data empty interrupts. Data is not written to SSIFTDR where a transmit

overflow error is generated. This error is generated regardless of while the transmission is operating. To resume transmission of SSIF-2, take the procedure to halt communication (**Figure 35.55**) and then take the error procedure (**Figure 35.56**). When communication is resumed, consider the lost serial data.

(c) Receive Underflow Error

In the case a receive underflow error is generated, review the number of reading operations from the receive FIFO data register (SSIFRDR) according to receive data full interrupts. The value read from SSIFRDR is undefined where a receive underflow error is generated. This error is generated regardless of while the reception is operating. To resume reception of SSIF-2, take the procedure to halt communication (**Figure 35.55**) and then take the error procedure (**Figure 35.56**).

(d) Receive Overflow Error

In the case a receive overflow error is generated during reception (SSICR.REN = 1, SSISR.IIRQ = 0), review the number of reading operations from the receive FIFO data register (SSIFRDR) according to receive data full interrupts. Data is not stored in SSIFRDR when a receive overflow error is generated. To resume reception of SSIF-2, take the procedure to halt communication (**Figure 35.55**) and then take the error handling procedure (**Figure 35.56**).

(2) Transmit Data Empty Interrupt

The communication flow defined in SSIF-2 uses the DMA. In the case the DMA is not used, observe one of SSIFSR.TDE, and write it in SSIFTDR. Follow empty capacity of transmission FIFO data register set with SSISCR.TDES about the number of times written in SSIFTDR detecting 1 of SSIFSR.TDE. Clear the transmission data that corresponds to empty capacity and clear the SSIFSR.TDE flag to SSIFTDR after writing. Transmission operation consecutive by the repeated thing is possible. SSIFSR.TDE is not cleared when not clearing.

(3) Receive Data Full Interrupt

The communication flow defined in SSIF-2 uses the DMA. In the case the DMA is not used, observe 1 of SSIFSR.RDF, and read SSIFRDR. Follow the receive data capacity of reception FIFO data register set with SSISCR.RDFS about the number of times in which SSIFRDR is read detecting 1 of SSIFSR.RDF. Clear data that does the reception completion and clear the SSIFSR.RDF flag from SSIFRDR after reading. Reception operation consecutive by the repeated thing is possible. SSIFSR.RDF is not cleared when not clearing.

(4) Switching Transfer Modes

1. For state transition from transmission, reception, and transmission and reception, disable transmission and reception (SSICR.TEN = 0, SSICR.REN = 0).
2. Confirm it is in the idle state (SSISR.IIRQ = 1).
3. In the idle state, set the SSICR.TEN bit and the SSICR.REN bit again and resume transfer.

(5) Resume Communication after Halting SSIF-2

When communication of SSIF-2 is halted according to the procedure to halt communication in **Figure 35.55**, resume communication according to the procedure to resume communication in **Figure 35.52**.

35.6.1.5 Write Access Restriction

(1) SSICR Register

If the TEN bit or REN bit is rewritten, make sure that the SSISR.IIRQ bit is in the desired status. If written, the operation performed immediately after writing is not guaranteed. For example, after enabling operation, make sure that SSISR.IIRQ = 0 and after disabling operation, make sure that SSISR.IIRQ = 1.

(a) TEN Bit and REN Bit

Writing 1 to these bits, transmission starts in synchronization with start trigger. Refer to **Section 35.5.2.2, Transmission**, **Section 35.5.2.3, Reception**, and **Section 35.5.2.4, Transmission and Reception** for details. Writing 0 to these bits, transmission stops in synchronization with next frame boundary. Set 1 at the same time when you use SSIF-2 as transmission and reception. To stop communication, make sure to stop transmission and reception (TEN = 0 and REN = 0).

(2) SSICR Register

(a) Clear operation of TUIRQ and TOIRQ

After the communication is permitted (SSICR.TEN is written and '1' is written from '0'), the transmission error flag (TOIRQ and TUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the transmission error flag might not be able to be read.

(b) Clear operation of RUIRQ and ROIRQ

After the communication is permitted (SSICR.REN is written and '1' is written from '0'), the transmission error flag (ROIRQ and RUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the reception error flag might not be able to be read.

(3) Communication State

Writing to the bits with shaded area in **Table 35.20** is prohibited. If written, the operation performed immediately after writing is not guaranteed.

Table 35.20 Bits Protected from Writing during Communication

Symbol	Address (Base+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	H'00	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	H'04	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	H'10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	BCKNCE	LRCKNCE	RxDNCE	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	H'14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	H'18	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	H'1C	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR		+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	H'24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	TDES[5:0]						—	—	RDFS[5:0]					

36. Sampling Rate Converter (SRC)

The sampling rate converter (SRC) handles sampling rate conversion.

36.1 Features

The following lists the features of the SRC.

■ Data format

16-bit stereo/16-bit monaural

■ Sampling rates

- Input:
  Selectable from 8 kHz, 11.025 kHz, 12 kHz, 16 kHz, 22.05 kHz, 24 kHz, 32 kHz, 44.1 kHz, and 48 kHz
- Output:
  Selectable from 8 kHz*1, 16 kHz*1, 32 kHz, 44.1 kHz, and 48 kHz

Note 1. This frequency is only available when the input sampling rate is 44.1 kHz.

■ Processing capacity

A sample output interval is a minimum of 4.62 µs (462 clock cycles at Pϕ = 100 MHz).

■ SNR

80 dB or higher

■ Five interrupt sources

Input data FIFO buffer empty, output data FIFO buffer full, output data FIFO buffer overwrite, output data FIFO buffer underflow, and conversion end

■ Two DMA transfer sources

Input data FIFO buffer empty and output data FIFO buffer full

Figure 36.1 shows the configuration of the SRC.

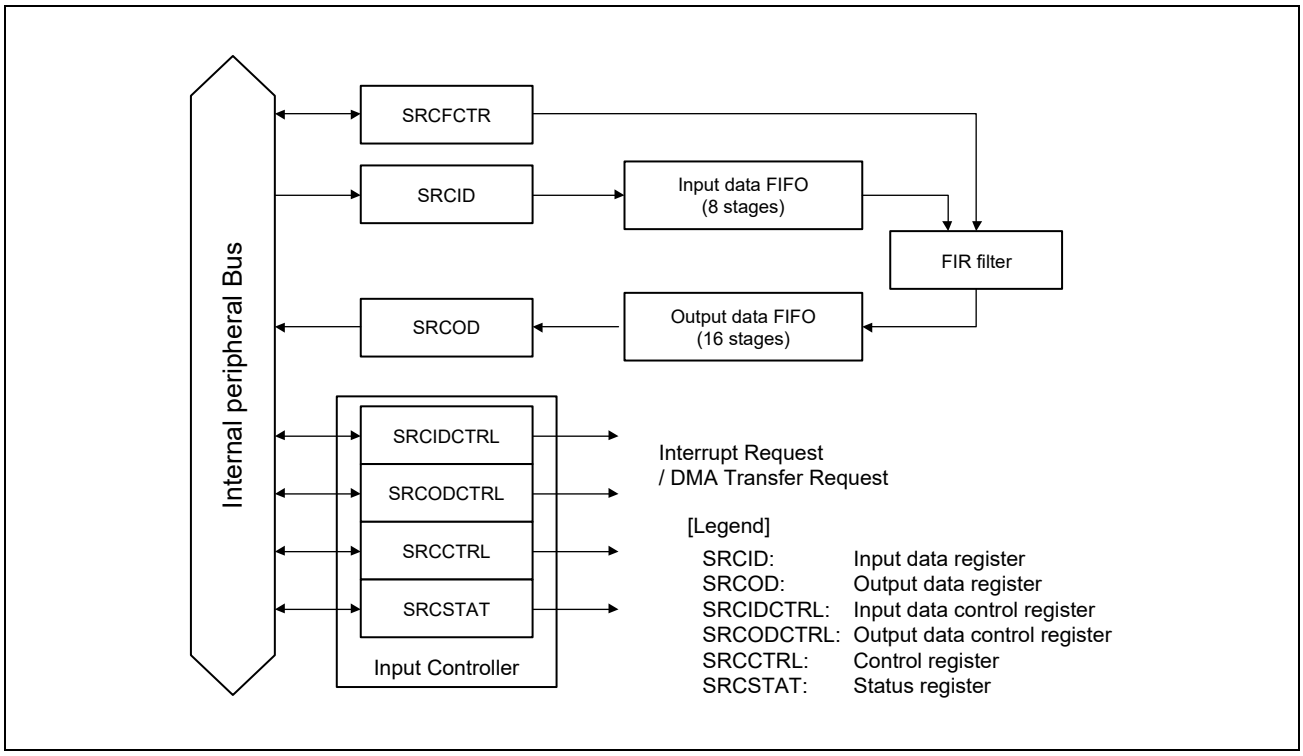


Figure 36.1 Block Diagram of SRC

36.2 Register Configuration

Table 36.1 shows the register configuration. The SRC address space is offset from the base address. The base address of SRC is as follows.

SRC base address: H'0_100A_0000 (Overall Address Space)

SRC base address: H'400A_0000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)

SRC base address: H'500A_0000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 36.1 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Higher-order bits in the input data register	SRCID*3	R/W	H'0000	H'7000	16
Lower-order bits in the input data register			H'0000	H'7002	16
Higher-order bits in the output data register	SRCOD*4	R	H'0000	H'7004	16
Lower-order bits in the output data register			H'0000	H'7006	16
Input data control register	SRCIDCTRL	R/W	H'0000	H'7008	16
Output data control register	SRCODCTRL	R/W	H'0000	H'700A	16
Control register	SRCCTRL	R/W	H'0000	H'700C	16
Status register	SRCSTAT	R/(W)*1	H'0002	H'700E	16
Filter coefficient tables	SRCFCTRs	R/W	Undefined	H'0000 to H'56BF	32*2

Note 1. Bits 15 to 6 and 4 are read-only. Only 0 can be written to bits 5, 3, 1 and 0 after having read as 1.

Note 2. The actual width of read and written data is 22 bits.

Note 3. For the input data register (SRCID), the 16 higher-order bits and the 16 lower-order bits must be written in that order.

Note 4. For the output data register (SRCOD), the 16 higher-order bits and the 16 lower-order bits must be read in that order.

36.3 Register Descriptions

36.3.1 Input Data Register (SRCID)

SRCID is a 32-bit readable and writable register that is used to input the data before sampling rate conversion. All the bits are read as 0. The data input to SRCID is stored in the 8-stage input data FIFO buffer. When the number of data units in the input data FIFO buffer is 8, writing to SRCID has no effect.

For stereo data, bits 31 to 16 are for Lch data, and bits 15 to 0 are for Rch data. For monaural data, data in bits 31 to 16 is valid, and data in bits 15 to 0 is invalid.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The data subject to sampling rate conversion is aligned differently depending on the IED bit setting in SRCIDCTRL. **Table 36.2** shows the relationship between the IED bit setting and data alignment.

Table 36.2 Alignment of Data before Sampling Rate Conversion

IED	Left channel [15:8]	Left channel [7:0]	Right channel [15:8]*1	Right channel [7:0]*1
0	SRCID[31:24]	SRCID[23:16]	SRCID[15:8]	SRCID[7:0]
1	SRCID[23:16]	SRCID[31:24]	SRCID[7:0]	SRCID[15:8]

Note 1. When processing monaural data, the data in these bits is invalid.

36.3.2 Output Data Register (SRCOD)

SRCOD is a 32-bit read-only register used to output the data after sampling rate conversion. The data in the output data FIFO buffer is read through SRCOD. When the number of data units in the 16-stage output data FIFO buffer is zero after the start of conversion, the value previously read is read again.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

The data in SRCOD is aligned differently depending on the OCH and OED bit setting in SRCODCTRL. **Table 36.3** shows the correspondence between the OCH and OED bit setting and data alignment in SRCOD.

Table 36.3 Alignment of Data in SRCOD

OCH	OED	SRCOD[31:24]	SRCOD[23:16]	SRCOD[15:7]	SRCOD[7:0]
0	0	Left channel [15:8]	Left channel [7:0]	Right channel [15:8]*2	Right channel [7:0]*2
	1	Left channel [7:0]	Left channel [15:8]	Right channel [7:0]*2	Right channel [15:8]*2
1*1	0	Right channel [15:8]	Right channel [7:0]	Left channel [15:8]	Left channel [7:0]
	1	Right channel [7:0]	Right channel [15:8]	Left channel [7:0]	Left channel [15:8]

Note 1. When processing monaural data, do not set the bit to 1.

Note 2. When processing monaural data, the data in these bits is invalid.

36.3.3 Input Data Control Register (SRCIDCTRL)

SRCIDCTRL is a 16-bit readable and writable register that specifies the endian format of input data, enables or disables the interrupt requests, and specifies the triggering number of data units.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	IED	IEN	—	—	—	—	—	—	IFTRG[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	IED	0	R/W	Input Data Endian Specifies the endian format of the input data. 0: Little endian 1: Big endian *Note*: This bit must be rewritten while the setting of the SRCEN bit is 0.
8	IEN	0	R/W	Input Data FIFO Buffer Empty Interrupt Enable Enables or disables the input data FIFO buffer empty interrupt request to be issued when the number of data units in the input FIFO buffer becomes equal to or smaller than the triggering number specified by the IFTRG1 and IFTRG0 bits, thus resulting in the IINT bit in the status register (SRCSTAT) being set to 1. 0: Input data FIFO buffer empty interrupt is disabled. 1: Input data FIFO buffer empty interrupt is enabled.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	IFTRG[1:0]	00	R/W	Input FIFO Buffer Data Triggering Number Specifies the condition in terms of the number on which the IINT bit in the status register (SRCSTAT) is set to 1. When the number of data units in the input FIFO buffer becomes equal to or smaller than the triggering number listed below, the IINT bit is set to 1. 00: 0 01: 2 10: 4 11: 6

36.3.4 Output Data Control Register (SRCODCTRL)

SRCODCTRL is a 16-bit readable and writable register that specifies whether to exchange the channels for the output data, specifies the endian format of output data, enables or disables the interrupt requests, and specifies the triggering number of data units.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	OCH	OED	OEN	—	—	—	—	—	—	OFTRG[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	OCH	0	R/W	Output Data Channel Exchange Specifies whether to exchange the channels for the output data register (SRCOD). 0: Does not exchange the channels (the same order as data input) 1: Exchanges the channels (the opposite order from data input) *Note 1.* When processing monaural data, do not set this bit to 1. *Note 2.* This bit must be rewritten while the setting of the SRCEN bit is 0.
9	OED	0	R/W	Output Data Endian Specifies the endian format of the output data. 0: Little endian 1: Big endian *Note*: This bit must be rewritten while the setting of the SRCEN bit is 0.
8	OEN	0	R/W	Output Data FIFO Buffer Full Interrupt Enable Enables or disables the output data FIFO buffer full interrupt request to be issued when the number of data units in the output FIFO buffer becomes equal to or greater than the number specified by the OFTRG1 and OFTRG0 bits, thus resulting in the OINT bit in the status register (SRCSTAT) being set to 1. 0: Output data FIFO buffer full interrupt is disabled. 1: Output data FIFO buffer full interrupt is enabled.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	OFTRG[1:0]	00	R/W	Output FIFO Buffer Data Trigger Number Specifies the condition in terms of the number on which the OINT bit in the status register (SRCSTAT) is set to 1. When the number of data units in the output FIFO buffer becomes equal to or greater than the number listed below, the OINT bit is set to 1. 00: 1 01: 4 10: 8 11: 12

36.3.5 Control Register (SRCCTRL)

SRCCTRL is a 16-bit readable and writable register that enables or disables access to the filter coefficient tables, enables or disables module operation, enables or disables the interrupt requests, specifies processing to flush or clear the internal working memory, and sets the input and output sampling rates.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FICRAE	—	CEEN	SRCEN	UDEN	OVEN	FL	CL	IFS[3:0]				—	OFS[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	FICRAE	0	R/W	Access to the Filter Coefficient Tables Enable Enables or disables reading from and writing to the filter coefficient table RAMs. 0: Disables reading from and writing to the filter coefficient table RAMs. 1: Enables reading from and writing to the filter coefficient table RAMs. *Note*: This bit must be rewritten while the setting of the SRCEN bit is 0.
14	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	CEEN	0	R/W	Conversion End Interrupt Enable Enables or disables the conversion end interrupt to be generated when the CEF bit in SRCSTAT is set to 1 after flush processing is completed and all the output data is read. 0: Disables conversion end interrupt requests. 1: Enables conversion end interrupt requests. *Note*: For details of flush processing, see the description of the FL bit in this register.
12	SRCEN	0	R/W	Module Enable Enables or disables this module operation. Writing 1 while SRCEN = 0 clears the internal working memory. 0: Disables this module operation. 1: Enables this module operation. *Note*: When SRCEN = 1, do not change the settings of the following bits. Register / Bit / Bit Name SRCIDCTRL / 9 / IED SRCODCTRL / 10, 9 / OCH, OED SRCCTRL / 15, 7 to 4, 2 to 0 / FICRAE, IFS[3:0], OFS[2:0]
11	UDEN	0	R/W	Output Data FIFO Buffer Underflow Interrupt Enable Enables or disables the output data FIFO buffer underflow interrupt to be generated when output data FIFO buffer is read and the UDF bit in SRCSTAT is set to 1 while the number of data units in the output data FIFO buffer is zero. 0: Disables output data FIFO buffer underflow interrupt requests. 1: Enables output data FIFO buffer underflow interrupt requests.

Bit	Bit Name	Initial Value	R/W	Description
10	OVEN	0	R/W	Output Data FIFO Buffer Overwrite Interrupt Enable Enables or disables the output data FIFO buffer overwrite interrupt request to be issued when the conversion for the next data has been completed while the number of data units in the output FIFO buffer is eight, thus setting the OVF bit in the status register (SRCSTAT) to 1. 0: Output data FIFO buffer overwrite interrupt is disabled. 1: Output data FIFO buffer overwrite interrupt is enabled. • When OVEN = 1: Conversion processing is stopped until the OVF bit is cleared by the CPU accessing to SRCSTAT when the output data FIFO buffer overwrite interrupt is generated. At this time, conversion result writing to the output data FIFO buffer is also stopped. • OVEN = 0: The OVF bit is automatically cleared when the output data FIFO buffer has space, and conversion processing can be continued.
9	FL	0	R/W	Internal Working Memory Flush Writing 1 to this bit starts converting the sampling rate of all the data in the input FIFO buffer, input buffer memory, and intermediate memory (i.e., flush processing). After all external input data are eliminated, sampling rate conversion processing proceeds for the data left in the internal memory. This bit is always read as 0. When SRCEN = 0, writing 1 to this bit does not trigger flush processing. In addition, when 1 is written to the FL bit while the number of data units in the input buffer memory is less than the values shown in **Table 36.6**, valid output data cannot be received. Thus the internal working memory is cleared without triggering the flush processing.
8	CL	0	R/W	Internal Working Memory Clear Writing 1 to this bit clears the input FIFO buffer, output FIFO buffer, input buffer memory, intermediate memory, and accumulator, after which this bit is cleared to 0. This bit is always read as 0. Even when SRCEN = 0, writing 1 to this bit clears the processing.
7 to 4	IFS[3:0]	0000	R/W	Input Sampling Rate Specifies the input sampling rate. 0000: 8.0 kHz 0001: 11.025 kHz 0010: 12.0 kHz 0011: Setting prohibited 0100: 16.0 kHz 0101: 22.05 kHz 0110: 24.0 kHz 0111: Setting prohibited 1000: 32.0 kHz 1001: 44.1 kHz 1010: 48.0 kHz 1011: Setting prohibited 1100: Setting prohibited 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited
3	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	OFS[2:0]	000	R/W	Output Sampling Rate Specifies the output sampling rate. 000: 44.1 kHz 001: 48.0 kHz 010: 32.0 kHz 011: Setting prohibited 100: 8.0 kHz*1 101: 16.0 kHz*1 110: Setting prohibited 111: Setting prohibited *Note 3.* This setting of OFS[2:0] is only valid when the setting of IFS[3:0] is 1001.

After flush processing has been completed, the number of output data units obtained as a result of conversion can be calculated by using the following formula.

$$\frac{\text{Number of output data units} - 1}{\text{Output sampling rate}} = \frac{\text{Number of input data units} \times \text{n} - 1}{\text{Input sampling rate} \times \text{n}}$$

$$\text{Number of output data units} = \left[(\text{Number of input data units} \times \text{n} - 1) \times \frac{\text{Output sampling rate}}{\text{Input sampling rate} \times \text{n}}\right] + 1$$

Table 36.4 Values of the n in the Formula Above

OFS Setting (Output Sampling Rate [kHz])	IFS Setting (Input Sampling Rate [kHz])								
	0000 (8.0)	0001 (11.025)	0010 (12.0)	0100 (16.0)	0101 (22.05)	0110 (24.0)	1000 (32.0)	1001 (44.1)	1010 (48.0)
000 (44.1)	6	4	4	3	2	2	3	—	1
001 (48.0)	6	4	4	3	2	2	3	1	—
010 (32.0)	4	8	4	2	4	2	—	2	1
100 (8.0)	—	—	—	—	—	—	—	1	—
101 (16.0)	—	—	—	—	—	—	—	1	—

Conversion processing by this module does not start and thus output data are not obtained until the specified number of data units have been input. The minimum number of input data units required for both obtaining the first output data and performing flush processing depends on the IFS and OFS bit settings.

Table 36.5 and **Table 36.6** show the relation between the settings of the IFS and OFS bits and the number of input data units required.

Table 36.5 Relation between Sampling Rate Settings and Number of Initial Input Data Units Required

OFS Setting (Output Sampling Rate [kHz])	IFS Setting (Input Sampling Rate [kHz])								
	0000 (8.0)	0001 (11.025)	0010 (12.0)	0100 (16.0)	0101 (22.05)	0110 (24.0)	1000 (32.0)	1001 (44.1)	1010 (48.0)
000 (44.1)	38	40	40	43	48	48	43	—	63
001 (48.0)	38	40	40	43	48	48	43	32	—
010 (32.0)	40	37	40	48	40	48	—	48	63
100 (8.0)	—	—	—	—	—	—	—	63	—
101 (16.0)	—	—	—	—	—	—	—	63	—

Table 36.6 Relation between Sampling Rate Settings and Number of Input Data Units Required for Flush Processing

OFS Setting (Output Sampling Rate [kHz])	IFS Setting (Input Sampling Rate [kHz])								
	0000 (8.0)	0001 (11.025)	0010 (12.0)	0100 (16.0)	0101 (22.05)	0110 (24.0)	1000 (32.0)	1001 (44.1)	1010 (48.0)
000 (44.1)	27	24	24	22	16	16	22	—	1
001 (48.0)	27	24	24	22	16	16	22	32	—
010 (32.0)	24	29	24	16	24	16	—	16	1
100 (8.0)	—	—	—	—	—	—	—	1	—
101 (16.0)	—	—	—	—	—	—	—	1	—

36.3.6 Status Register (SRCSTAT)

SRCSTAT is a 16-bit readable and writable register that indicates the number of data units in the input and output data FIFO buffers, whether the various interrupt sources have been generated or not, and the flush processing status.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFDN[4:0]					IFDN[3:0]				—	CEF	FLF	UDF	OVF	IINT	OINT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R/(W)*1	R	R/(W)*1	R/(W)*1	R/(W)1*	R/(W)1*

Note 3. Only 0 can be written after having read as 1.

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	OFDN[4:0]	00000	R	Output FIFO Buffer Data Count Indicates the number of data units in the output FIFO buffer.
10 to 7	IFDN[3:0]	0000	R	Input FIFO Buffer Data Count Indicates the number of data units in the input FIFO buffer.
6	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	CEF	0	R/(W) *1	Conversion End Flag Indicates that all the output data is read after flush processing is completed. [Clearing conditions] • 0 has been written to the CEF bit after reading CEF = 1 • 1 has been written to the CL bit in SRCCTRL • 1 has been written to the SRCEN bit in SRCCTRL while SRCEN is 0 [Setting condition] • The number of data units in the output data FIFO buffer is zero on completion of flush processing
4	FLF	0	R	Flush Processing Status Flag Indicates whether flush processing is in progress or not. [Clearing conditions] • Flush processing has been completed • 1 has been written to the CL bit in SRCCTRL • 1 has been written to the SRCEN bit in SRCCTRL while SRCEN is 0 [Setting condition] • 1 has been written to the FL bit in SRCCTRL Note that this bit will never be set when flush processing is not performed.
3	UDF	0	R/(W) *1	Output FIFO Buffer Underflow Interrupt Request Flag Indicates that the output data FIFO buffer is read when the number of data units in the output data FIFO buffer is zero. [Clearing conditions] • 0 has been written to the UDF bit after reading UVF = 1 • 1 has been written to the CL bit in SRCCTRL • 1 has been written to the SRCEN bit in SRCCTRL while SRCEN is 0 [Setting condition] • The output data FIFO buffer is read while the number of data units in the output FIFO buffer is zero

Bit	Bit Name	Initial Value	R/W	Description
2	OVF	0	R/(W) *1	Output Data FIFO Buffer Overwrite Interrupt Request Flag Indicates that the sampling rate conversion for the next data has been completed when the output data FIFO buffer is full. The conversion is stopped until the OVF flag is cleared. [Clearing conditions] • 0 has been written to the OVF bit after reading OVF = 1 while the OVEN bit in SRCCTRL is 1 • The number of data units in the output FIFO buffer decreases after reading SRCOD while the OVEN bit in SRCCTRL is 0, that is, empty space is available in the output FIFO buffer • 1 has been written to the CL bit in SRCCTRL • 1 has been written to the SRCEN bit in SRCCTRL while SRCEN is 0 [Setting condition] • The sampling rate conversion for the next data has been completed when the output FIFO buffer is full
1	IINT	1	R/(W) *1	Input Data FIFO Buffer Empty Interrupt Request Flag Indicates that the number of data units in the input FIFO buffer has become equal to or smaller than the triggering number specified by the IFTRG[1:0] bits in the input data control register (SRCIDCTRL). [Clearing conditions] • 0 has been written to the IINT bit after reading IINT = 1 • Handling of the last transfer by DMA transfer [Setting conditions] • The number of data units in the input FIFO buffer has become equal to or smaller than the specified triggering number • 1 has been written to the CL bit in SRCCTRL • 1 has been written to the SRCEN bit in SRCCTRL while SRCEN is 0
0	OINT	0	R/(W) *1	Output Data FIFO Buffer Full Interrupt Request Flag Indicates that the number of data units in the output FIFO buffer has become equal to or greater than the triggering number specified by the OFTRG[1:0] bits in the output data control register (SRCODCTRL). [Clearing conditions] • 0 has been written to the OINT bit after reading OINT = 1 • Handling of the last transfer by DMA transfer • 1 has been written to the CL bit in SRCCTRL • 1 has been written to the SRCEN bit in SRCCTRL while SRCEN is 0 [Setting condition] • The number of data units in the output FIFO buffer has become equal to or greater than the specified triggering number

Note 1. Only 0 can be written after having read as 1.

36.3.7 Filter Coefficient Tables (SRCFCTRs)

SRCFCTRs are 32-bit readable and writable random access memory (RAMs) that hold the filter coefficients to be used in the sampling rate conversion processing. The peripheral bus only has access to the RAMs when the FICRAE and SRCEN bits in the control register (SRCCTRL) are set to 1 and 0, respectively. Bits 31 to 22 are reserved and always read as 0. Bits 21 to 0 are used as the register that holds filter coefficient values, of which the initial values are undefined.

Each filter coefficient table can only hold 22 bits × 5552 words of coefficient data. The sampling rate conversion operation and its characteristics are only guaranteed when the separately provided coefficient data are used.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	SRCFCOE[21:16]					
Initial Value	0	0	0	0	0	0	0	0	0	0	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCFCOE[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

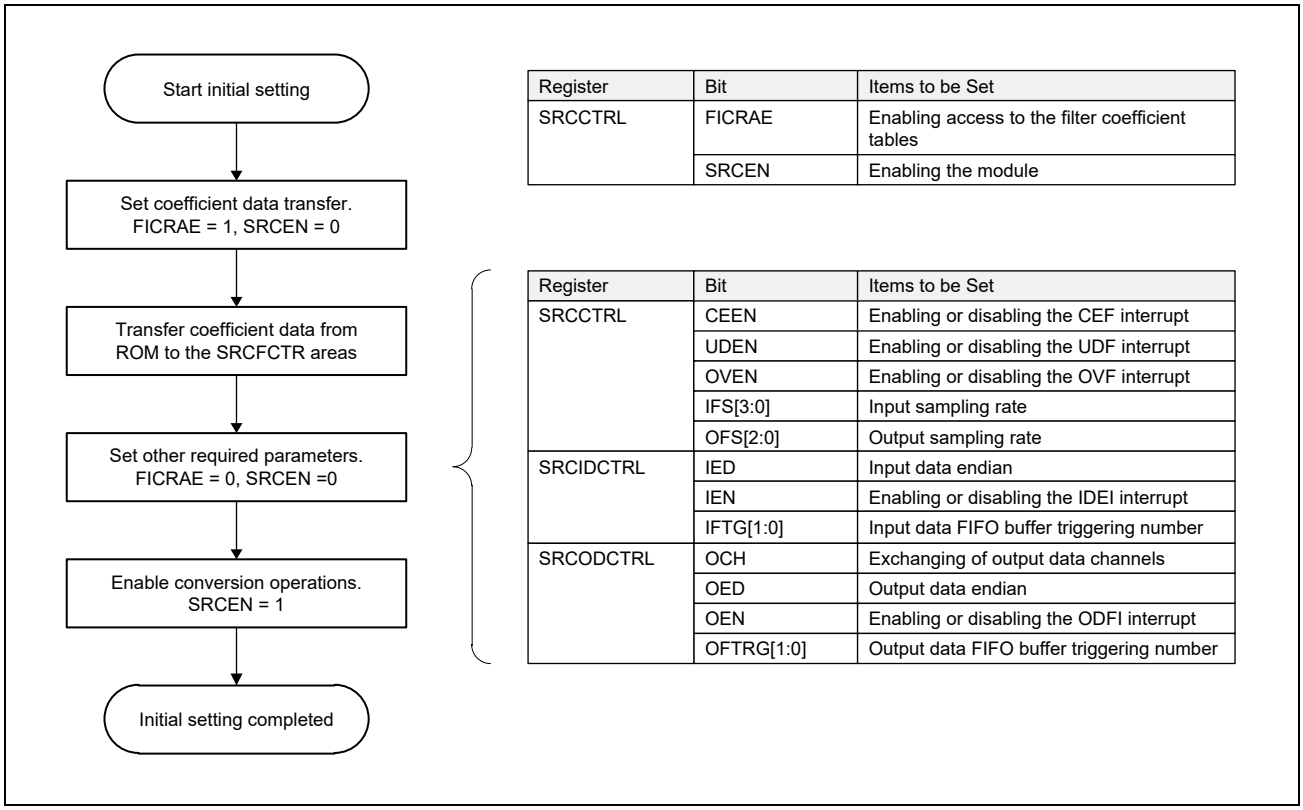
x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21 to 0	SRCFCOE [21:0]	Undefined	R/W	Filter coefficient tables

36.4 Operation

36.4.1 Initial Setting

Figure 36.2 shows a sample flowchart for initial setting. After the reset release and before conversion operations in the SRC start, the filter coefficient data stored in a medium such as flash memory (ROM) must be transferred to the filter coefficient tables (SRCFCTRs). When the coefficient data have already been stored in the filter coefficient tables, the required parameters can be set immediately.



Register	Bit	Items to be Set
SRCCTRL	FICRAE	Enabling access to the filter coefficient tables
	SRCEN	Enabling the module

Register	Bit	Items to be Set
SRCCTRL	CEEN	Enabling or disabling the CEF interrupt
	UDEN	Enabling or disabling the UDF interrupt
	OVEN	Enabling or disabling the OVF interrupt
	IFS[3:0]	Input sampling rate
	OFS[2:0]	Output sampling rate
SRCIDCTRL	IED	Input data endian
	IEN	Enabling or disabling the IDEI interrupt
	IFTG[1:0]	Input data FIFO buffer triggering number
SRCODCTRL	OCH	Exchanging of output data channels
	OED	Output data endian
	OEN	Enabling or disabling the ODFI interrupt
	OFTRG[1:0]	Output data FIFO buffer triggering number

Figure 36.2 Sample Flowchart for Initial Setting

36.4.2 Data Input

Figure 36.3 is a sample flowchart for data input.

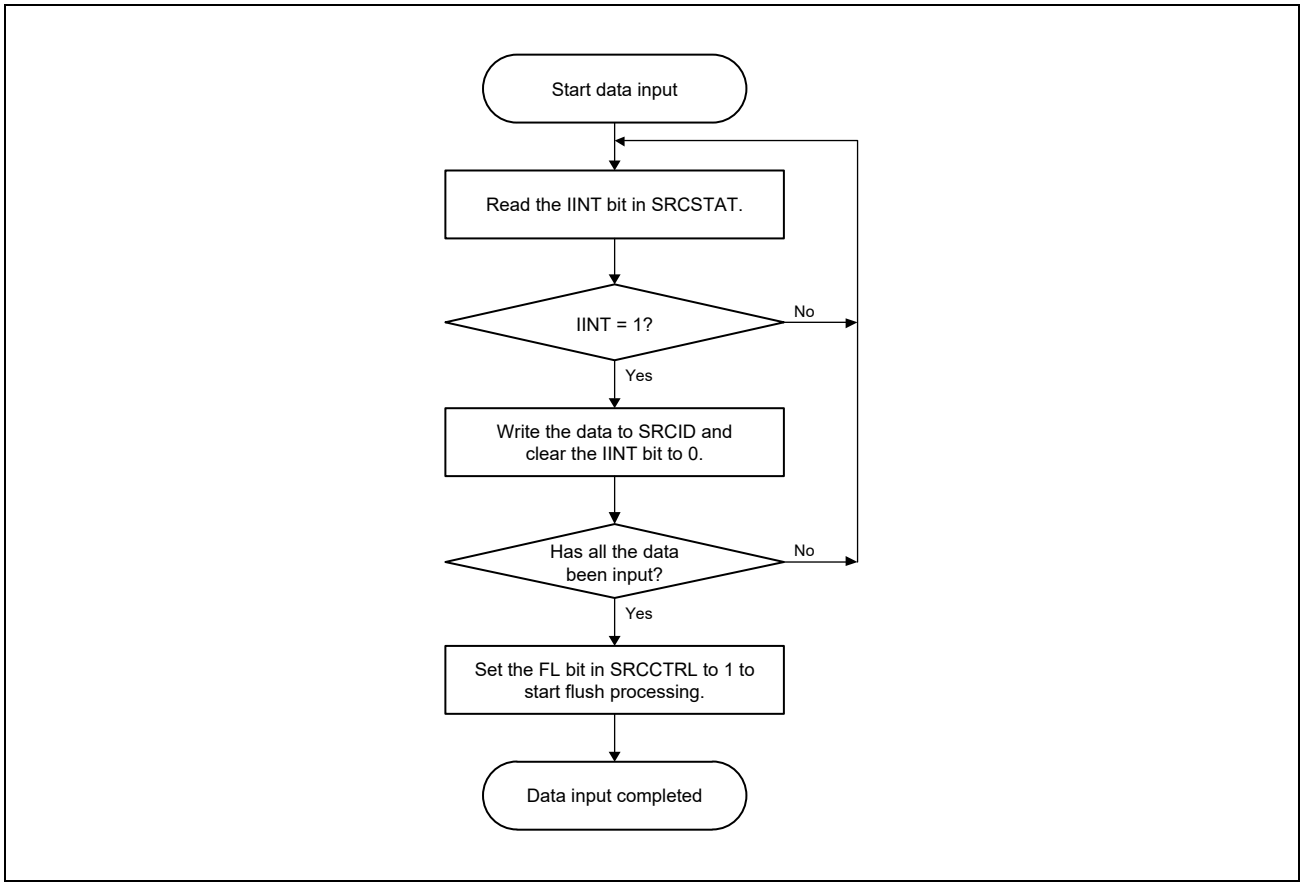


Figure 36.3 Sample Flowchart for Data Input

(1) When Interrupts are Issued to CPU

1. Set the IEN bit in SRCIDCTRL to 1.
2. When the IINT bit in SRCSTAT is set to 1, an IDEI interrupt request is issued. In the interrupt processing routine, read the IINT bit and confirm that it is 1, write data to SRCID. Then return from the interrupt processing routine.
3. Transfer the data by repeating step 2 as many times as required.
4. When all data have been input, write 1 to the FL bit in SRCCTRL.

(2) When Interrupts are Used to Activate Direct Memory Access Controller

1. Assign the IDEI interrupt request of this module to one channel of the direct memory access controller.
2. Set the IEN bit in SRCIDCTRL to 1.
3. When the IINT bit in SRCSTAT is set to 1, an IDEI interrupt request, that is, a DMA transfer request, is issued thus activating the direct memory access controller. When the direct memory access controller has written data to the SRCID thus resulting in the last data unit having been written, the IINT bit is cleared to 0.
4. Transfer the data by repeating step 3 as many times as required.
5. When all data have been input, write 1 to the FL bit in SRCCTRL.

NOTE

The number of stages of the input FIFO buffer is eight. Moreover, the possible number of data units that can be transferred in response to an IDEI interrupt request, that is, the number of empty FIFO buffer stages, depends on the setting of the IFTRG[1:0] bits in SRCCTRL. As the input FIFO buffer has no functionality to avoid or detect being overwritten, the data in it will be destroyed when it is overwritten. The number of contiguous data units to be transferred by DMA transfer, therefore, must be set in consideration of the setting of the IFTRG[1:0] bits.

36.4.3 Data Output

Figure 36.4 is a sample flowchart for data output.

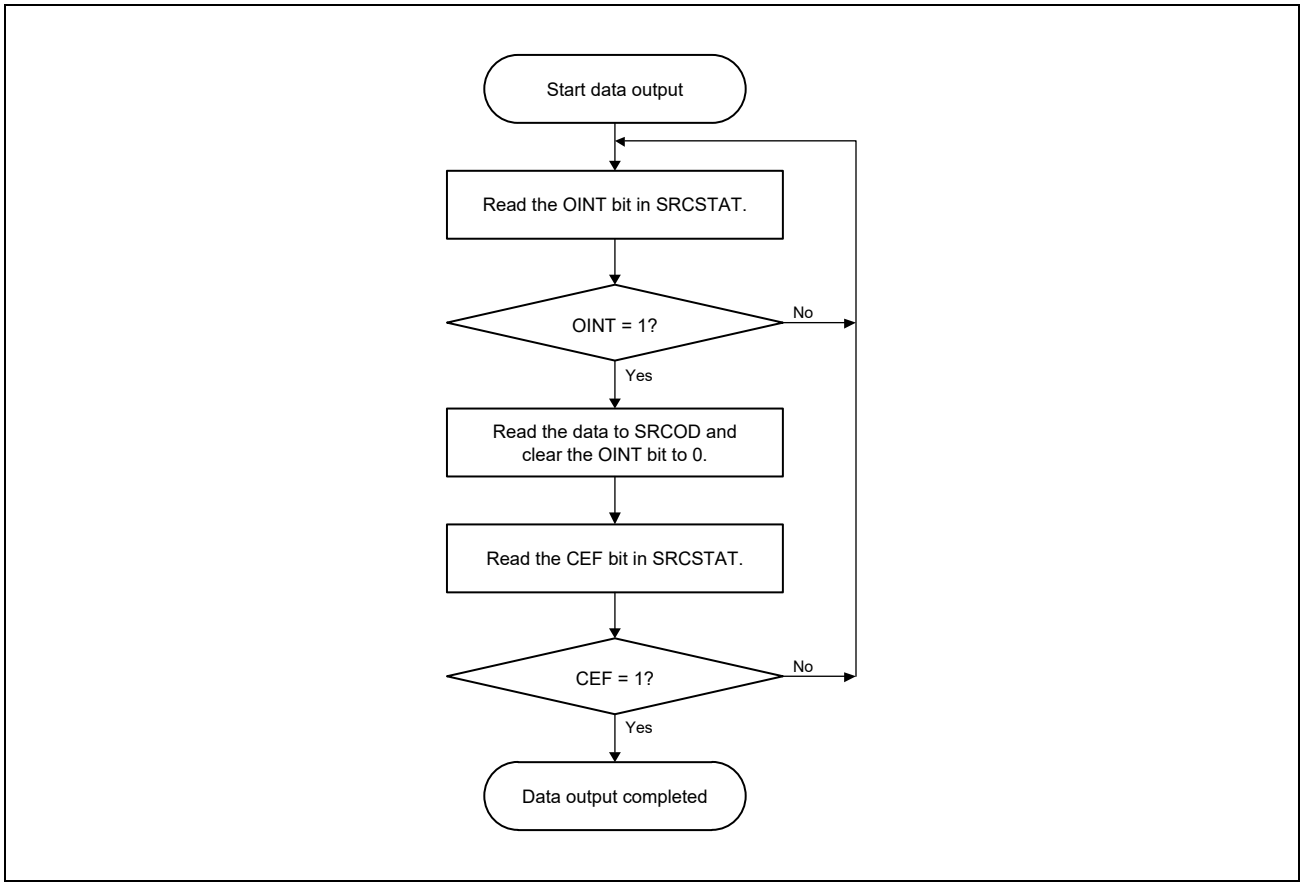


Figure 36.4 Sample Flowchart for Data Output

(1) When Interrupts are Issued to CPU

1. Set the OEN bit in SRCODCTRL to 1.
2. When the OINT bit in SRCSTAT is set to 1, an ODFI interrupt request is issued. In the interrupt processing routine, read the OINT bit and confirm that it is 1, read data from SRCOD. Then return from the interrupt processing routine.
3. Read the data by repeating step 2 as many times as required.
4. When all data have been read after flush processing starts, the CEF bit in SRCSTAT is set to 1. When the CEF bit is read as 1, the data output has been completed.

(2) When Interrupts are Used to Activate Direct Memory Access Controller

1. Assign the ODFI interrupt request of this module to one channel of the direct memory access controller.
2. Set the OEN bit in SRCODCTRL to 1.
3. When the OINT bit in SRCSTAT is set to 1, an ODFI interrupt request, that is, a DMA transfer request, is issued thus activating the direct memory access controller. When the direct memory access controller has read data from SRCOD thus resulting in the last data unit having been read, the OINT bit is cleared to 0.
4. Read the data by repeating step 3 as many times as required.
5. When all data have been read after flush processing starts, the CEF bit in SRCSTAT is set to 1. When the CEF bit is read as 1, the data output has been completed.

NOTES

1. The number of stages of the output FIFO buffer is sixteen. When the output FIFO buffer overflows, that is, when it is overwritten without data in it having been read, conversion processing is stopped. Though the output FIFO buffer can be read even when it has overflowed, resuming conversion processing requires additional processing that depends on the setting of the OVEN bit. For details, see the description of the OVEN bit in SRCCTR.
2. When the output FIFO buffer is read while the number of data units in it is zero, the value read is invalid. The number of contiguous data units to be read by DMA transfer must be set in consideration of the setting of the OFTRG[1:0] bits.

36.4.4 Interrupts

This module has five interrupt sources. **Table 36.7** summarizes the interrupts.

Table 36.7 Interrupt Requests and Generation Conditions

Interrupt Request	Abbreviation	Interrupt Condition	Direct Memory Access Controller Activation	Interrupt Request Signal
Input data FIFO buffer empty	IDEI	IINT = 1, IEN = 1, and SRCEN = 1	Possible	Edge
Output data FIFO buffer full	ODFI	OINT = 1, OEN = 1, and SRCEN = 1	Possible	Edge
Output data FIFO buffer overwrite	OVF	OVF = 1, OVEN = 1, and SRCEN = 1	Not possible	Level
Output data FIFO buffer underflow	UDF	UDF = 1, UDEN = 1, and SRCEN = 1	Not possible	Level
Conversion end	CEF	CEF = 1, CEEN = 1, and SRCEN = 1	Not possible	Level

When one of the interrupt conditions is met, the corresponding pin outputs an interrupt request signal that is detected as an edge or level. In response to the signal, the CPU executes the corresponding interrupt exception handling routine. The corresponding interrupt source flag, IINT, OINT, OVF, UDF, or CEF should be cleared from within the routine.

The IDEI and ODFI interrupts can activate the direct memory access controller (DMA) when the DMA is set to allow this. The IINT and OINT flags must not be cleared by writing from the CPU during DMA transfer. Clearing here refers to writing 0 to a bit after having read it as 1.

36.5 Usage Notes

36.5.1 Note on Accessing Registers

After the following write access to SRCCTRL, three cycles of the peripheral clock elapse before the corresponding bit in SRCSTAT is updated.

- Before the FLF bit in SRCSTAT is set after 1 is written to the FL bit in SRCCTRL
- Before each bit in SRCSTAT is initialized after 1 is written to the CL bit in SRCCTRL
- Before each bit in SRCSTAT is initialized after 1 is written to the SRCEN bit in SRCCTRL while the SRCEN bit is 0

As the CPU executes any subsequent instruction without waiting for the completion of the register writing, an instruction that immediately follows that used to write to SRCCTRL cannot accurately detect the updated state of SRCSTAT. To check the updated SRCSTAT state, dummy-read SRCCTRL or SRCSTAT after the instruction used to write to SRCCTRL.

36.5.2 Note on Flush Processing

When 1 is written to the FL bit in SRCCTRL, this module continues conversion processing by adding 0-data to the input data end point. Flush processing, therefore, should be performed when the audio data end point is input and there is no subsequent data.

To perform conversion again after flush processing, clear the internal working memory in either of the following ways.

- Write 1 to the CL bit in SRCCTRL.
- Write 0 and then 1 to the SRCEN bit in SRCCTRL.

36.5.3 Note on CPU Operations during Transfer by the DMAC

When the DMAC is used for the data transfer to the input and from the output data registers (SRCID and SRCOD), the IINT and OINT bits (bits 1 and 0) in the status register (SRCSTAT) must not be cleared by the CPU during transfer. Clearing here refers to writing 0 to a bit after having read it as 1.

36.5.4 Note on Access while the SRC is Operating

Access to the filter coefficient tables must not proceed while the SRC is operating, that is, while the setting of the SRCEN bit is 1.

36.5.5 Note on After rest

The coefficient data is not initialized by reset, so reloading of the coefficient data is not necessary.

37. PDM Interface (PDM)

This section describes the PDM interface.

37.1 Overview

This PDM interface is a unit for receiving voice data from an external PDM (pulse density modulation) device. For features and configuration, see the following sections.

37.1.1 Feature

Table 37.1 lists the features of this PDM interface.

Table 37.1 PDM Interface Specifications

Item	Description
Number of channels	3 channels
Functions	• Capable of filtering 1-bit digital input data PDM_DATn (n = 0, 1, 2) and converting them into 20-bit or 16-bit digital data. The bit order is little-endian. • This unit supports stereo microphone (L/R sampling by rising/falling clock edge). • This unit supports sound activity detector. • Each channel includes programmable filters: 4th order sinc filter, high-pass filter (for suppression of DC bias), correction filter (for sinc passband distortion), half-band decimation filter (for aliasing distortion). – This unit supports programmable and flexible decimation ratios. – The sinc filter is selectable as first-, second-, third-, and fourth-order. • This unit supports DMA operation through APB. • Each channel has an internal buffer. – 64 stages when designed – Capable of storing voice data during low power mode • Error detection functions can be used for debugging.
Interrupt sources	7 sources 1. Data reception interrupt per channel (Maximum 3 interrupts) 2. Sound detection interrupt (shared between channels) 3. Error detection interrupt per channel (Maximum 3 interrupts)
Low power consumption function	This unit supports microphone low power configuration (slower clock). This unit can switch to the higher clock speed after sound detection recognized.
Bus interface	This unit has an APB interface which is confirmed to APB4.

37.1.2 Block Diagram

The following is a block diagram of the PDM interface.

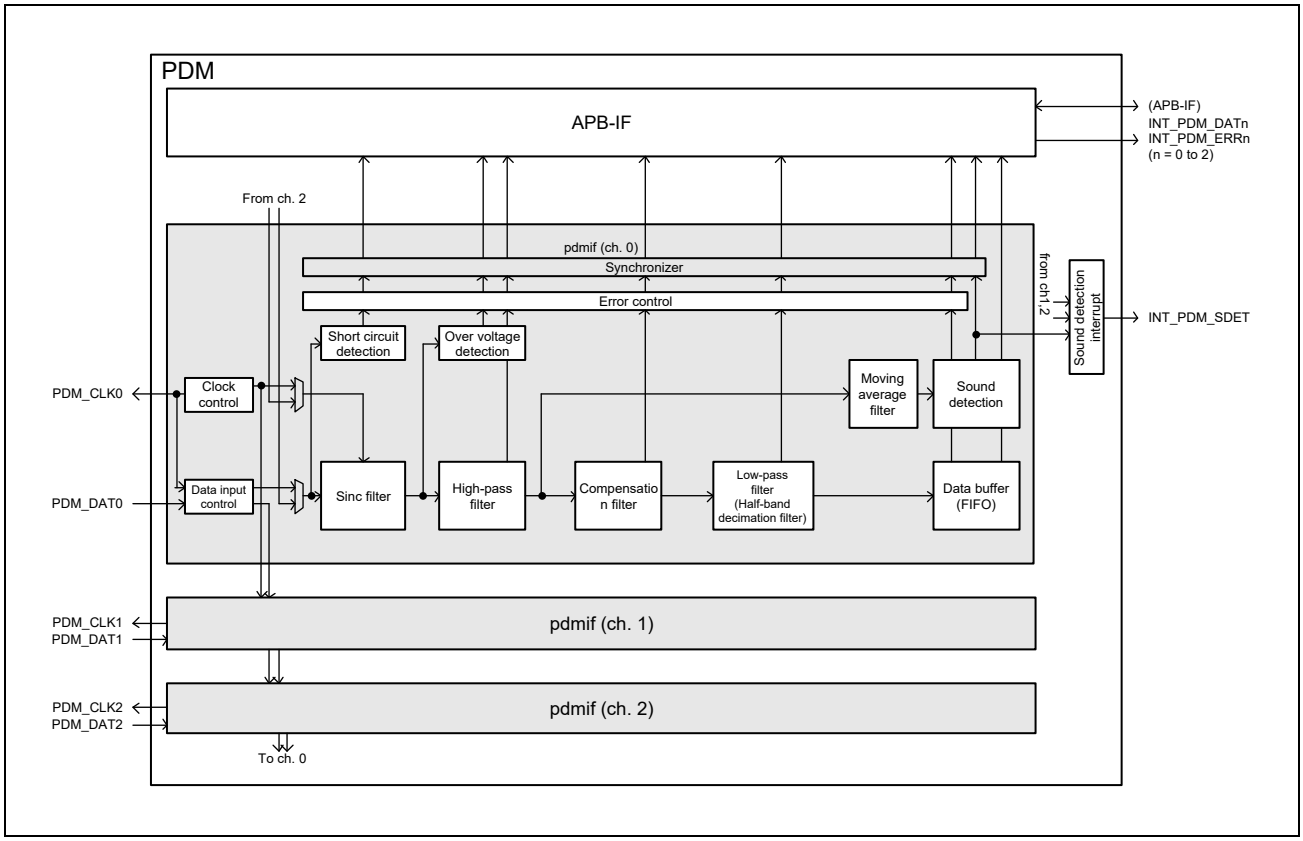


Figure 37.1 Three-Channel PDM-IF Block Diagram

Each channel of PDM-IF consists of following blocks:

- Clock control: Generates PDM_CLKn (n = 0, 1, 2) to the microphone, and internal timing signals to the sinc filter.
- Data input control: Receives PDM_DATn (n = 0, 1, 2) from the microphone.
- Sinc filter: Converts the 1-bit digital data stream input from PDM_DATn (n = 0, 1, 2) to 34-bit signed data. Then, the 34-bit signed data is clipped to 20-bit signed data.
- High-pass filter: A filter for suppression of DC bias
- Compensation filter: A filter for sinc passband distortion
- Low-pass (half-band decimation) filter: A filter for aliasing distortion
- Moving average filter: A simple filter for noise reduction before sound detection
- Sound detection: Generates the wakeup interrupt to recover from low power mode.
- Data buffer: Stores 20-bit or 16-bit data filtered by some filters.
- Short circuit detection: Detects the short circuit error which indicates consecutive 0's or 1's input from PDM_DATn (n = 0, 1, 2).
- Overvoltage detection: Detects the overvoltage error.

- Error control: Generates error flags from other blocks.

The 1-bit digital data stream input from PDM_DATn pin (n = 0, 1, 2) is converted to 34-bit signed data by the sinc filter and then the data is clipped to 20-bit signed data. The 20-bit singed data is used for sound detection and data storing in buffer.

The sound detection block receives 20-bit signed data through high-pass filter block and moving average filter block. The block compares received data to the programmed thresholds and generates a wakeup interrupt in case that the data is above the upper threshold or below the lower threshold.

The data buffer block receives 20-bit signed data through high-pass filter, compensation filter, and low-pass filter. The block stores the 20-bit signed data or the clipped 16-bit signed data. And then the block sends the 20-bit signed data or 16-bit signed data to APB-IF once APB read access occurs.

37.1.3 External Pins

Table 37.2 lists the PDM external pin specifications.

Table 37.2 Pin Configuration

External Pin Name	I/O	Function	Pin Name in This Section
PDM0_DAT	Input	External PDM Ch0 data reception pin	PDM_DAT0
PDM1_DAT	Input	External PDM Ch1 data reception pin	PDM_DAT1
PDM2_DAT	Input	External PDM Ch2 data reception pin	PDM_DAT2
PDM0_CLK	Output	External PDM Ch0 clock output pin	PDM_CLK0
PDM1_CLK	Output	External PDM Ch1 clock output pin	PDM_CLK1
PDM2_CLK	Output	External PDM Ch2 clock output pin	PDM_CLK2

37.1.4 Connected Unit

Table 37.3 lists the connected units.

Table 37.3 Connected Unit

Unit Name	Connected Unit Name	Function of Connected Unit
PDMm	CPG	Clock pulse generator (CPG)
	LSI Internal Bus	LSI Internal Bus
	Interrupt Controller	Interrupt control unit
	GPIO	Pin function controller

37.2 Registers

The register addresses of PDM are given as offsets from the individual base addresses <PDMm_base>. The register base addresses of each PDM are listed in the following table.

Table 37.4 Register Base Addresses

Base Address Name	Unit	Base Address
<PDM0_base>	PDM0	H'0_1005_A000 (H'5005_A000*1, H'4005_A000*2)

Note 1. Cortex-M33 address space (non-secure)

Note 2. Cortex-M33 address space (secure)

Note: Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

37.2.1 List of Registers

The list of PDM registers and the memory addresses are as follows.

For the actual addresses, the offset values indicated in the following table are added to the base addresses.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size (bits)
Channel Software Start Trigger Register	PDMm_PDCSTRTR	H'0000_0000	H'0000	32
Channel Software Stop Trigger Register	PDMm_PDCSTPTR	H'0000_0000	H'0004	32
Channel Software Change Trigger Register	PDMm_PDCCHGTR	H'0000_0000	H'0008	32
Channel Interrupt Control Register	PDMm_PDCICR	H'0000_0000	H'000C	32
Channel Status Register	PDMm_PDCSR	H'0000_0000	H'0010	32
Channel Status Clear Register	PDMm_PDCSCR	H'0000_0000	H'0014	32
Reserve	—	—	H'0018 to H'001F	—
Channel Sound Detection Control Register	PDMm_PDCSDCR	H'0000_0000	H'0020	32
Channel Data Read Control Register	PDMm_PDCDRCR	H'0000_0000	H'0024	32
Channel Data Clear Register	PDMm_PDCDCR	H'0000_0000	H'0028	32
Reserve	—	—	H'002C to H'007F	—
Version Register	PDMm_PDVR	H'0000_0010	H'0080	32
Reserve	—	—	H'0084 to H'00FF	—
Software Start Trigger Register Channel n	PDMm_PDSTRTRCHn	H'0000_0000	H'0100 + n × H'0100	32
Software Stop Trigger Register Channel n	PDMm_PDSTPTRCHn	H'0000_0000	H'0104 + n × H'0100	32
Software Change Trigger Register Channel n	PDMm_PDCHGTRCHn	H'0000_0000	H'0108 + n × H'0100	32
Interrupt Control Register Channel n	PDMm_PDICRCHn	H'0000_0000	H'010C + n × H'0100	32
Status Detection Control Register Channel n	PDMm_PDSDCRCHn	H'0000_0000	H'0110 + n × H'0100	32
Status Register Channel n	PDMm_PDSRCHn	H'0000_0000	H'0114 + n × H'0100	32
Status Clear Register Channel n	PDMm_PDSCRCHn	H'0000_0000	H'0118 + n × H'0100	32
Reserve	—	—	H'011C + n × H'0100 to H'011F + n × H'0100	—
Mode Setting Register Channel n	PDMm_PDMDSRCHn	H'0000_0000	H'0120 + n × H'0100	32
Sinc filter Control Register Channel n	PDMm_PDSFCRCHn	H'057C_0000	H'0124 + n × H'0100	32
High-pass filter Coefficient s(0) Register Channel n	PDMm_PDHFCS0RCHn	H'0000_3F61	H'0128 + n × H'0100	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size (bits)
High-pass filter Coefficient k(1) Register Channel n	PDMm_PDHFCK1RCHn	H'0000_3EC1	H'012C + n × H'0100	32
High-pass filter Coefficient h(0) Register Channel n	PDMm_PDHFCH0RCHn	H'0000_4000	H'0130 + n × H'0100	32
High-pass filter Coefficient h(1) Register Channel n	PDMm_PDHFCH1RCHn	H'0000_C000	H'0134 + n × H'0100	32
Compensation filter Coefficient h(0) Register Channel n	PDMm_PDCFCH00RCHn	H'0000_1FE8	H'0138 + n × H'0100	32
Compensation filter Coefficient h(1) Register Channel n	PDMm_PDCFCH01RCHn	H'0000_0039	H'013C + n × H'0100	32
Compensation filter Coefficient h(2) Register Channel n	PDMm_PDCFCH02RCHn	H'0000_003C	H'0140 + n × H'0100	32
Compensation filter Coefficient h(3) Register Channel n	PDMm_PDCFCH03RCHn	H'0000_1E56	H'0144 + n × H'0100	32
Compensation filter Coefficient h(4) Register Channel n	PDMm_PDCFCH04RCHn	H'0000_01DC	H'0148 + n × H'0100	32
Compensation filter Coefficient h(5) Register Channel n	PDMm_PDCFCH05RCHn	H'0000_06E1	H'014C + n × H'0100	32
Compensation filter Coefficient h(6) Register Channel n	PDMm_PDCFCH06RCHn	H'0000_01DC	H'0150 + n × H'0100	32
Compensation filter Coefficient h(7) Register Channel n	PDMm_PDCFCH07RCHn	H'0000_1E56	H'0154 + n × H'0100	32
Compensation filter Coefficient h(8) Register Channel n	PDMm_PDCFCH08RCHn	H'0000_003C	H'0158 + n × H'0100	32
Compensation filter Coefficient h(9) Register Channel n	PDMm_PDCFCH09RCHn	H'0000_0039	H'015C + n × H'0100	32
Compensation filter Coefficient h(10) Register Channel n	PDMm_PDCFCH10RCHn	H'0000_1FE8	H'0160 + n × H'0100	32
Low-pass filter Coefficient h0(10) Register Channel n	PDMm_PDLFCH010RCHn	H'0000_0400	H'0164 + n × H'0100	32
Low-pass filter Coefficient h1(0) Register Channel n	PDMm_PDLFCH10RCHn	H'0000_1FF8	H'0168 + n × H'0100	32
Low-pass filter Coefficient h1(1) Register Channel n	PDMm_PDLFCH101RCHn	H'0000_000A	H'016C + n × H'0100	32
Low-pass filter Coefficient h1(2) Register Channel n	PDMm_PDLFCH102RCHn	H'0000_1FF0	H'0170 + n × H'0100	32
Low-pass filter Coefficient h1(3) Register Channel n	PDMm_PDLFCH103RCHn	H'0000_0018	H'0174 + n × H'0100	32
Low-pass filter Coefficient h1(4) Register Channel n	PDMm_PDLFCH104RCHn	H'0000_1FDC	H'0178 + n × H'0100	32
Low-pass filter Coefficient h1(5) Register Channel n	PDMm_PDLFCH105RCHn	H'0000_0034	H'017C + n × H'0100	32
Low-pass filter Coefficient h1(6) Register Channel n	PDMm_PDLFCH106RCHn	H'0000_1FB3	H'0180 + n × H'0100	32
Low-pass filter Coefficient h1(7) Register Channel n	PDMm_PDLFCH107RCHn	H'0000_0076	H'0184 + n × H'0100	32
Low-pass filter Coefficient h1(8) Register Channel n	PDMm_PDLFCH108RCHn	H'0000_1F2E	H'0188 + n × H'0100	32
Low-pass filter Coefficient h1(9) Register Channel n	PDMm_PDLFCH109RCHn	H'0000_0289	H'018C + n × H'0100	32
Low-pass filter Coefficient h1(10) Register Channel n	PDMm_PDLFCH110RCHn	H'0000_0289	H'0190 + n × H'0100	32
Low-pass filter Coefficient h1(11) Register Channel n	PDMm_PDLFCH111RCHn	H'0000_1F2E	H'0194 + n × H'0100	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size (bits)
Low-pass filter Coefficient h1(12) Register Channel n	PDMm_PDLFCH112RCHn	H'0000_0076	H'0198 + n × H'0100	32
Low-pass filter Coefficient h1(13) Register Channel n	PDMm_PDLFCH113RCHn	H'0000_1FB3	H'019C + n × H'0100	32
Low-pass filter Coefficient h1(14) Register Channel n	PDMm_PDLFCH114RCHn	H'0000_0034	H'01A0 + n × H'0100	32
Low-pass filter Coefficient h1(15) Register Channel n	PDMm_PDLFCH115RCHn	H'0000_1FDC	H'01A4 + n × H'0100	32
Low-pass filter Coefficient h1(16) Register Channel n	PDMm_PDLFCH116RCHn	H'0000_0018	H'01A8 + n × H'0100	32
Low-pass filter Coefficient h1(17) Register Channel n	PDMm_PDLFCH117RCHn	H'0000_1FF0	H'01AC + n × H'0100	32
Low-pass filter Coefficient h1(18) Register Channel n	PDMm_PDLFCH118RCHn	H'0000_000A	H'01B0 + n × H'0100	32
Low-pass filter Coefficient h1(19) Register Channel n	PDMm_PDLFCH119RCHn	H'0000_1FF8	H'01B4 + n × H'0100	32
Sound Detection Lower Threshold Register Channel n	PDMm_PDSDLTRCHn	H'0000_0000	H'01B8 + n × H'0100	32
Sound Detection Upper Threshold Register Channel n	PDMm_PDSDUTRCHn	H'0000_0000	H'01BC + n × H'0100	32
Data Buffer Control Register Channel n	PDMm_PDDBCRCHn	H'0000_0000	H'01C0 + n × H'0100	32
Short Circuit Threshold Setting Register Channel n	PDMm_PDSCTSRCHn	H'0000_0000	H'01C4 + n × H'0100	32
Overvoltage Lower Threshold Register Channel n	PDMm_PDOVLTRCHn	H'0000_0000	H'01C8 + n × H'0100	32
Overvoltage Upper Threshold Register Channel n	PDMm_PDOVUTRCHn	H'0000_0000	H'01CC + n × H'0100	32
Reserve	—	—	H'01D0 + n × H'0100 to H'01DF + n × H'0100	—
Data Read Control Register Channel n	PDMm_PDDRCRCHn	H'0000_0000	H'01E0 + n × H'0100	32
Data Clear Register Channel n	PDMm_PDDCRCHn	H'0000_0000	H'01E4 + n × H'0100	32
Data Read Register Channel n	PDMm_PDDRRCHn	H'0000_0000	H'01E8 + n × H'0100	32
Data Status Register Channel n	PDMm_PDDSRCHn	H'0000_0000	H'01EC + n × H'0100	32

37.2.2 Register Description

The prefix (PDMm_) of the register names is omitted in this and subsequent sections.

37.2.2.1 Channel Software Start Trigger Register (PDMm_PDCSTRTR) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0000

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	STRTRG[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	STRTRG[2:0]	H'0	R0W	The read value is always 0b. Channel n start trigger 0b: Do nothing 1b: Start channel n

37.2.2.2 Channel Software Stop Trigger Register (PDMm_PDCSTPTR) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0004

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	STPTRG[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	STPTRG[2:0]	H'0	R0W	The read value is always 0b. Channel n stop trigger 0b: Do nothing 1b: Stop channel n

37.2.2.3 Channel Software Change Trigger Register (PDMm_PDCCHGTR) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0008

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CHGTRG[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0W	R0W	R0W	R0	R0	R0	R0	R0	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10 to 8	—	All 0	R0W	Reserved. This register is readable and the initial value can be written to it.
7 to 3	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	CHGTRG[2:0]	H'0	R0W	The read value is always 0b. Channel n change trigger 0b: Do nothing 1b: Change clock (PDM_CLKn) setting

37.2.2.4 Channel Interrupt Control Register (PDMm_PDCICR) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'000C

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	IEDE[2:0]			—	—	—	—	—	IDRE[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	RW	RW	RW	R0	R0	R0	R0	R0	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ISDE[2:0]			—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	RW	RW	RW	R0	R0	R0	R0	R0	R0	R0	R0

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
26 to 24	IEDE[2:0]	H'0	RW	Channel n error detection interrupt enable bit 0b: Don't allow to issue INT_PDM_ERRn interrupt 1b: Allow to issue INT_PDM_ERRn interrupt
23 to 19	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	IDRE[2:0]	H'0	RW	Channel n data reception interrupt enable bit 0b: Don't allow to issue INT_PDM_DATn interrupt 1b: Allow to issue INT_PDM_DATn interrupt
15 to 11	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10 to 8	ISDE[2:0]	H'0	RW	Channel n sound detection interrupt enable bit 0b: Don not allow to issue INT_PDM_SDET interrupt when the sound for channel n is detected. 1b: Allow to issue INT_PDM_SDET interrupt when the sound for channel n is detected.
7 to 0	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.

37.2.2.5 Channel Status Register (PDMm_PDCSR) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0010

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	EDF[2:0]			—	—	—	—	—	DRF[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R	R	R	R0	R0	R0	R0	R0	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SDF[2:0]			—	—	—	—	—	STATE[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R	R	R	R0	R0	R0	R0	R0	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
26 to 24	EDF[2:0]	H'0	R	Channel n error detection flag 0b: Indicates that no data is detected. 1b: Indicates that an error is detected. Refer to the PDSRCHn register for details of errors.
23 to 19	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	DRF[2:0]	H'0	R	Channel n data reception flag 0b: Indicates that the number of data stored in buffer does not exceed the threshold. 1b: Indicates that the number of data stored in buffer exceeded the threshold.
15 to 11	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10 to 8	SDF[2:0]	H'0	R	Channel n sound detection flag 0b: Indicates that the sound which exceeded the threshold is not detected. 1b: Indicates that the sound which exceeded the threshold is detected.
7 to 3	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	STATE[2:0]	H'0	R	Channel n state 0b: Channel n stop 1b: Channel n in operation

EDFn

[Condition to become "1b"]

- When any error occurs. Refer to the PDSRCHn register for the kind of error.

[Condition to become "0b"]

- When all error flags are cleared by writing the PDSCRCHn, this flag is also cleared.

DRFn

Refer to **Section 37.3.7, Data Buffer** for details.

[Condition to become “1b”]

- When PDCDRCR.DATREn = 1b and PDDSRCHn.DATNUM[7:0] is greater than or equal to a threshold which is configured by PDDBCRCHn.DATRITHR[2:0].

[Condition to become “0b”]

- When PDCDRCR.DATREn = 0b or PDDSRCHn.DATNUM[7:0] is less than a threshold.

SDFn

Refer to **Section 37.3.6, Sound Detection** for details.

[Conditions to become “1b”]

- When PDCSDCR.SDEn = 1b and a moving average filter result is greater than or equal to PDSDUTRCHn.SDETU[19:0].
- When PDCSDCR.SDEn = 1b and a moving average filter result is less than or equal to PDSDLTRCHn.SDETL[19:0].

[Condition to become “0b”]

- When writing 1b to a flag clear register bit (PDCSCR.SDFCn).

STATEn

Refer to **Section 37.3.2, Channel Start and Channel Stop** for details.

[Condition to become “1b”]

- When “1b” is written to PDCSTRTR.STRTRGn.

[Condition to become “0b”]

- When “1b” is written to PDCSTPTR.STPTRGn.

37.2.2.6 Channel Status Clear Register (PDMm_PDCSCR) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0014

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SDFC[2:0]			—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0W	R0W	R0W	R0	R0	R0	R0	R0	R0	R0	R0

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10 to 8	SDFC[2:0]	H'0	R0W	The read value is always 0b. Channel n sound detection flag clear 0b: Do nothing 1b: Clear PDCSR.SDFn
7 to 0	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.

37.2.2.7 Channel Sound Detection Control Register (PDMm_PDCSDCR) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0020

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	SDE[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	SDE[2:0]	H'0	RW	Channel n sound detection enable bit 0b: Do not allow to detect sound 1b: Allow to detect sound

37.2.2.8 Channel Data Read Control Register (PDMm_PDCDRCR) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0024

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DATRE[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	DATRE[2:0]	H'0	RW	Channel n data read enable bit 0b: Do not allow to read data from data buffer 1b: Allow to read data from data buffer

37.2.2.9 Channel Data Clear Register (PDMm_PDCDCR) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0028

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DATC[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	DATC[2:0]	H'0	R0W	The read value is always 0b. Channel n data clear 0b: Do nothing 1b: Clear data

37.2.2.10 Version Register (PDMm_PDVR) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0080

Initial Value: H'0000_0010

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VER[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R0	R0	R0	R0	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	VER[11:0]	H'010	R	IP version is shown.

37.2.2.11 Software Start Trigger Register Channel n (PDMm_PDSTRTRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0100

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STRTRG
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	STRTRG	H'0	R0W	The read value is always 0b. Start trigger 0b: Do nothing 1b: Start the channel

37.2.2.12 Software Stop Trigger Register Channel n (PDMm_PDSTPTRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0104

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STPTRG
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	STPTRG	H'0	R0W	The read value is always 0b. Stop trigger 0b: Do nothing 1b: Stop the channel

37.2.2.13 Software Change Trigger Register Channel n (PDMm_PDCHGTRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0108

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CHGTRG
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0W	R0	R0	R0	R0	R0	R0	R0	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	—	H'0	R0W	Reserved This register is readable and the initial value can be written to it.
7 to 1	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	CHGTRG	H'0	R0W	The read value is always 0b. Change trigger 0b: Do nothing 1b: Change settings of PDM_CLKn and sinc filter

37.2.2.14 Interrupt Control Register Channel n (PDMm_PDICRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'010C

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IEDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	IDRE	ISDE	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	R0

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	IEDE	H'0	RW	Error detection interrupt enable bit 0b: Do not allow to issue INT_PDM_ERRn interrupt 1b: Allow to issue INT_PDM_ERRn interrupt
15 to 3	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	IDRE	H'0	RW	Data reception interrupt enable bit 0b: Do not allow to issue INT_PDM_DATn interrupt 1b: Allow to issue INT_PDM_DATn interrupt
1	ISDE	H'0	RW	Sound detection interrupt enable bit 0b: Do not allow to issue INT_PDM_SDET interrupt 1b: Allow to issue INT_PDM_SDET interrupt
0	—	H'0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.

37.2.2.15 Status Detection Control Register Channel n (PDMm_PDSDCRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0110

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	BFOWDE	—	—	—	—	—	—	—	—	OVUDE	OVLDE	SCDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDE	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27	BFOWDE	H'0	RW	Buffer overwriting detection enable bit 0b: Do not allow to detect buffer overwriting. 1b: Allow to detect buffer overwriting.
26 to 19	—	All 0	RW	Reserved This register is readable and the initial value can be written to it.
18	OVUDE	H'0	RW	Overvoltage upper limit exceeded detection enable bit 0b: Do not allow to detect data exceeding the upper limit. 1b: Allow to detect data exceeding the upper limit
17	OVLDE	H'0	RW	Overvoltage lower limit exceeded detection enable bit 0b: Do not allow to detect data falling below the lower limit. 1b: Allow to detect data falling below the lower limit
16	SCDE	H'0	RW	Short circuit detection enable bit 0b: Do not allow to detect the PDM_DATn pin short circuit 1b: Allow to detect the PDM_DATn pin short circuit
15 to 2	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SDE	H'0	RW	Sound detection enable bit 0b: Do not allow to detect sound 1b: Allow to detect sound
0	—	H'0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.

37.2.2.16 Status Register Channel n (PDMm_PDSRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0114

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	BFOWDF	—	—	—	—	—	—	—	—	OVUDF	OVLDF	SCDF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DRF	SDF	STATE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27	BFOWDF	H'0	R	Buffer overwriting detection flag 0b: Indicates that buffer overwriting does not occur. 1b: Indicates that buffer overwriting occurred.
26 to 19	—	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18	OVUDF	H'0	R	Overvoltage upper limit exceeded detection flag 0b: Indicates that the data is not above the upper limit 1b: Indicates that the data has exceeded the upper limit
17	OVLDF	H'0	R	Overvoltage lower limit exceeded detection flag 0b: Indicates that the data is not below the lower limit 1b: Indicates that the data has falling below the lower limit
16	SCDF	H'0	R	Short circuit detection flag 0b: Indicates that the PDM_DATn pin short circuit is not detected 1b: Indicates that a short circuit on the PDM_DATn pin has been detected
15 to 3	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	DRF	H'0	R	Data reception flag 0b: Indicates that the number of data stored in buffer is less than a threshold 1b: Indicates that the number of data stored in buffer is greater than or equal to a threshold
1	SDF	H'0	R	Sound detection flag 0b: Indicates that the sound is not detected 1b: Indicates that the sound is detected
0	STATE	H'0	R	State 0b: Channel stop 1b: Channel in operation

BFOWDF

Refer to **Section 37.3.7, Data Buffer** for details.

[Condition to become “1b”]

- When PDSDCRCHn.BFOWDE = 1b and data buffer is full and data is written to the buffer without reading.

[Condition to become “0b”]

- When writing 1b to a flag clear register bit (PDSCRCHn.BFOWDFC).

OVUDF

Refer to **Section 37.3.9, Overvoltage Detection** for details.

[Condition to become “1b”]

- When PDSDCRCHn.OVUDE = 1b and a clipped sinc filter result is larger than an upper limit (PDOVUTRCHn.OVDU[19:0]).

[Condition to become “0b”]

- When writing 1b to a flag clear register bit (PDSCRCHn.OVUDFC).

OVLDF

Refer to **Section 37.3.9, Overvoltage Detection** for details.

[Condition to become “1b”]

- When PDSDCRCHn.OVLDE = 1b and a clipped sinc filter result is smaller than a lower limit (PDOVLTRCHn.OVDL[19:0]).

[Condition to become “0b”]

- When writing 1b to a flag clear register bit (PDSCRCHn.OVLDFC).

SCDF

Refer to **Section 37.3.8, Short-Circuit Detection** for details.

[Conditions to become “1b”]

- When PDSDCRCHn.SCDE = 1b and 1b is continuously input to the PDM_DATn pin, exceeding the number of times set in PDSCTSRCHn.SCDH[12:0].
- When PDSDCRCHn.SCDE = 1b and 0b is continuously input to the PDM_DATn pin, exceeding the number of times set in PDSCTSRCHn.SCDL[12:0].

[Condition to become “0b”]

- When writing 1b to a flag clear register bit (PDSCRCHn.SCDFC).

DRF

Refer to **Section 37.3.7, Data Buffer** for details.

[Condition to become “1b”]

- When PDDRCRCHn.DATRE = 1b and PDDSRCHn.DATNUM[7:0] is greater than or equal to a threshold which is configured by PDDBCRCHn.DATRITHR[2:0].

[Condition to become “0b”]

- When PDDRCRCHn.DATRE = 0b or PDDSRCHn.DATNUM[7:0] is less than a threshold.

SDF

Refer to **Section 37.3.6, Sound Detection** for details.

[Conditions to become “1b”]

- When PDSDCRCHn.SDE = 1b and a moving average filter result is PDSDUTRCHn.SDETU[19:0] or greater.
- When PDSDCRCHn.SDE = 1b and a moving average filter result is PDSDLTRCHn.SDETL[19:0] or less.

[Condition to become “0b”]

- When writing 1b to a flag clear register bit (PDSCRCHn.SDFC).

STATE

Refer to **Section 37.3.2, Channel Start and Channel Stop** for details.

[Condition to become “1b”]

- When “1b” is written to PDSTRTRCHn.STRTRG.

[Condition to become “0b”]

- When “1b” is written to PDSTPTRCHn.STPTRG.

37.2.2.17 Status Clear Register Channel n (PDMm_PDSCRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0118

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	BFOWDFC	—	—	—	—	—	—	—	—	OVUDFC	OVLDFC	SCDFC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDFC	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0W	R0

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27	BFOWDFC	H'0	R0W	The read value is always 0b. Buffer overwriting detection flag clear 0b: Do nothing 1b: Clear PDSRCHn.BFOWDF
26 to 19	—	All 0	R0W	Reserved This register is readable and the initial value can be written to it.
18	OVUDFC	H'0	R0W	The read value is always 0b. Overvoltage upper limit exceeded detection flag clear 0b: Do nothing 1b: Clear PDSRCHn.OVUDF
17	OVLDFC	H'0	R0W	The read value is always 0b. Overvoltage lower limit exceeded detection flag clear 0b: Do nothing 1b: Clear PDSRCHn.OVLDF
16	SCDFC	H'0	R0W	The read value is always 0b. Short circuit detection flag clear 0b: Do nothing 1b: Clear PDSRCHn.SCDF
15 to 2	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SDFC	H'0	R0W	The read value is always 0b. Sound detection flag clear 0b: Do nothing 1b: Clear PDSRCHn.SDF
0	—	H'0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.

37.2.2.18 Mode Setting Register Channel n (PDMm_PDMDSRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0120

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DBIS[3:0]				—	—	SDMAMD[1:0]		—	—	—	—	—	—	LFIS[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R0	R0	RW	RW	R0	R0	R0	RW	R0	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CFIS[1:0]		—	—	HFIS[1:0]		—	SFMD[2:0]			—	—	—	INPSEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	RW	RW	RW	R0	RW	RW	RW	R0	RW	RW	RW	R0	R0	R0	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	DBIS[3:0]	H'0	RW	Data buffer input shift setting 0000b: 20-bit mode, {1{S},[18:0]} 0001b: 20-bit mode, {2{S},[18:1]} 0010b: 20-bit mode, {3{S},[18:2]} 0011b: 20-bit mode, {4{S},[18:3]} (0100b-0111b : Setting prohibited) 1000b: 16-bit mode, {S,D[18:4]} 1001b: 16-bit mode, {S,D[17:3]} 1010b: 16-bit mode, {S,D[16:2]} 1011b: 16-bit mode, {S,D[15:1]} 1100b: 16-bit mode, {S,D[14:0]} (1101b-1111b: Setting prohibited) *Note*: S: Sign bit
27, 26	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25, 24	SDMAMD [1:0]	H'0	RW	Moving average mode for sound detection data 00b: 1-order (filter is skipped) (default) 01b: 2-order 10b: 4-order 11b: Setting prohibited
23 to 21	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	—	H'0	RW	Reserved This register is readable and the initial value can be written to it.
19	—	H'0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18	—	H'0	RW	Reserved This register is readable and the initial value can be written to it.
17, 16	LFIS[1:0]	H'0	RW	Low-pass (half-band decimation) filter input shift setting 00b: No shift 01b: 1-bit right shift 10b: 2-bit right shift 11b: 3-bit right shift

Bit	Bit Name	Initial Value	R/W	Description
15	—	H'0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14	—	H'0	RW	Reserved This register is readable and the initial value can be written to it.
13, 12	CFIS[1:0]	H'0	RW	Compensation filter input shift setting 00b: No shift 01b: 1-bit right shift 10b: 2-bit right shift 11b: 3-bit right shift
11	—	H'0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10	—	H'0	RW	Reserved This register is readable and the initial value can be written to it.
9, 8	HFIS[1:0]	H'0	RW	High-pass filter input shift setting 00b: No shift 01b: 1-bit right shift 10b: 2-bit right shift 11b: 3-bit right shift
7	—	H'0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 4	SFMD[2:0]	H'0	RW	Sinc filter mode setting 001b: 1-order 010b: 2-order 011b: 3-order Other: 4-order (default)
3 to 1	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	INPSEL	H'0	RW	Input data select 0b: Rise-edge data of channel n 1b: Fall-edge data of channel n-1 (In case of n = 0, fall-edge data of channel 2 is selected.)

37.2.2.19 Sinc filter Control Register Channel n (PDMm_PDSFCRCHn) (m = 0)

See **Table 37.8** for the setting values of SINCDEC[7:0] and SINCRNG[4:0].

Access Size: 32 bits

Address(es): <PDMm_base> + H'0124

Initial Value: H'057C_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SINCRNG[4:0]					SINCDEC[7:0]							
Initial Value	0	0	0	0	0	1	0	1	0	1	1	1	1	1	0	0
R/W	R0	R0	R0	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CKDIV[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 24	SINCRNG [4:0]	H'05	RW	Sinc filter output valid range

00000b: {S, [32:14]}		10000b:	{S, [16:0], 00b}
00001b: {S, [31:13]}		10001b:	{S, [15:0], 000b}
00010b: {S, [30:12]}		10010b:	{S, [14:0], 0000b}
00011b: {S, [29:11]}		10011b:	{S, [13:0], 00000b}
00100b: {S, [28:10]}		10100b:	{S, [12:0], 000000b}
00101b: {S, [27:9]} (default)		10101b:	{S, [11:0], 0000000b}
00110b: {S, [26:8]}		10110b:	{S, [10:0], 00000000b}
00111b: {S, [25:7]}		10111b:	{S, [9:0], 000000000b}
01000b: {S, [24:6]}		11000b:	{S, [8:0], 0000000000b}
01001b: {S, [23:5]}		11001b:	{S, [7:0], 00000000000b}
01010b: {S, [22:4]}		11010b:	{S, [6:0], 000000000000b}
01011b: {S, [21:3]}		11011b:	{S, [5:0], 0000000000000b}
01100b: {S, [20:2]}		11100b:	{S, [4:0], 00000000000000b}
01101b: {S, [19:1]}		11101b:	{S, [3:0], 000000000000000b}
01110b: {S, [18:0]}		11110b:	{S, [2:0], 0000000000000000b}
01111b: {S, [17:0], 0b}		11111b:	{S, [1:0], 00000000000000000b} (prohibited)

Note: S: Signed bit.
When the overflow occurs, H'7FFFF is output. When the underflow occurs, H'80000 is output.

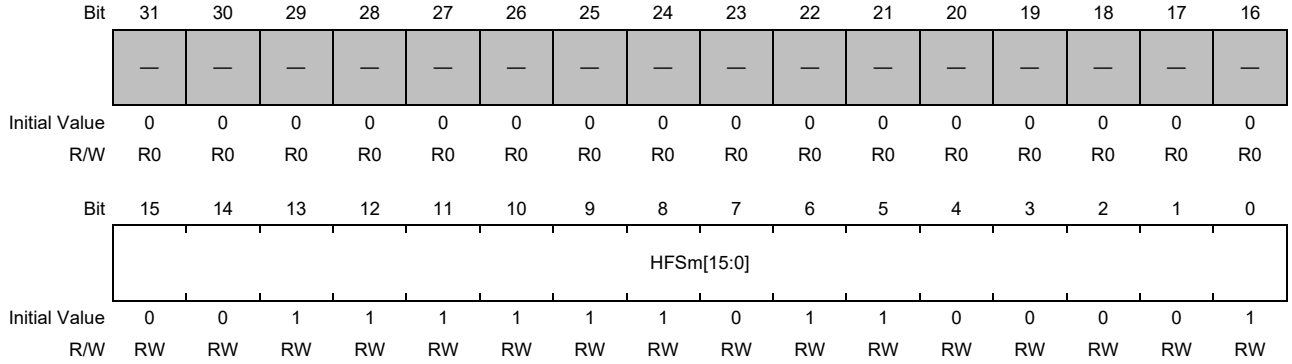
Bit	Bit Name	Initial Value	R/W	Description
23 to 16	SINCDEC [7:0]	H'7C	RW	Sinc filter decimation ratio M indicates the decimation ratio. Decimation ratio M is set as follows. M = SINCDEC + 1 H'00: M = 1 (prohibited) H'01: M = 2 (prohibited) H'02: M = 3 (prohibited) H'03: M = 4 … H'7C: M = 125 (default) … H'FF: M = 256 Setting a value less than M = 4 is prohibited. If such a value is set, operation of the sinc filter is not guaranteed. The setting should satisfy D × M > 12. Otherwise, operation of other filters is not guaranteed. *Note*: D: PDM_CLKn's division ratio
15 to 4	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3 to 0	CKDIV[3:0]	H'0	RW	PDM_CLKn's division ratio to core clock 0000b: 1/2 (default) 0001b: 1/4 0010b: 1/6 … 1101b: 1/28 1110b: 1/30 1111b: 1/32

37.2.2.20 High-Pass Filter Coefficient s(0) Register Channel n (PDMm_PDHFCS0RCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0128

Initial Value: H'0000_3F61



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	HFSm[15:0]	H'3F61	RW	High-pass filter coefficient s0 16-bit signed data (Fraction bit [13:0]) For the initial value, see Value after Reset of Each Coefficient.

● Value after Reset of Each Coefficient

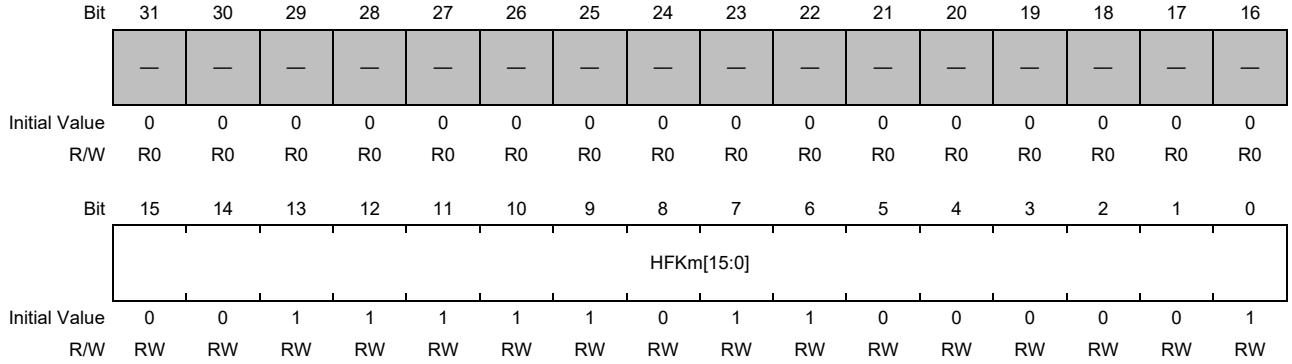
Symbol Name	Coefficient Name	Value after Reset		Comment
		Hex	Decimal	
HFS0	s(0)	H'3F61	0.99029541015625	Cut-off 100 Hz

37.2.2.21 High-Pass Filter Coefficient k(1) Register Channel n (PDMm_PDHFCK1RCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'012C

Initial Value: H'0000_3EC1



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	HFKm[15:0]	H'3EC1	RW	High-pass filter coefficient k1 16-bit signed data (Fraction bit [13:0]) For the initial value, see Value after Reset of Each Coefficient.

● Value after Reset of Each Coefficient

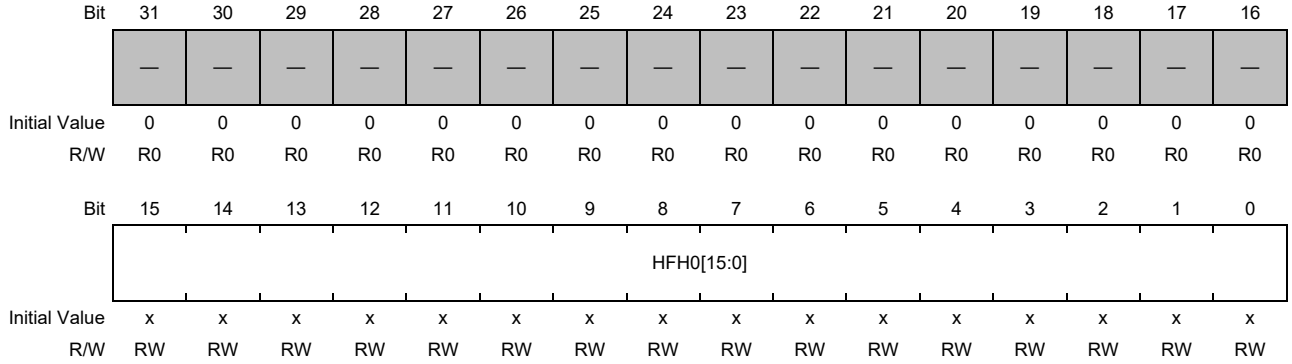
Symbol Name	Coefficient Name	Value after Reset		Comment
		Hex	Decimal	
HFK1	k(1)	H'3EC1	0.98052978515625	Cut-off 100 Hz

37.2.2.22 High-Pass Filter Coefficient h(p) Register Channel n (PDMm_PDHFCHpRCHn) (m = 0, p = 0, 1)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0130 + p × H'0004

Initial Value: H'0000_xxxx



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	HFH0[15:0]	All x	RW	High-pass filter coefficient h(p) 16-bit signed data (Fraction bit [13:0]) For the initial value, see Value after Reset of Each Coefficient.

Remark x: Undefined

● Value after Reset of Each Coefficient

Symbol Name	Coefficient Name	Value after Reset		Comment
		Hex	Decimal	
HFH0	h(0)	H'4000	1	
HFH1	h(1)	H'C000	−1	

37.2.2.23 Compensation Filter Coefficient h(p) Register Channel n (PDMm_PDCFCHpRCHn) (m = 0, p = 00 to 10)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0138 + p × H'0004

Initial Value: H'0000_xxxx

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CFH00[12:0]												
Initial Value	0	0	0	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R0	R0	R0	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	CFH00[12:0]	All x	RW	Compensation filter coefficients h(p) 13-bit signed data (Fraction bit [10:0]) For the initial value, see Value after Reset of Each Coefficient.

Remark x: Undefined

● Value after Reset of Each Coefficient

Symbol Name	Coefficient Name	Value after Reset		Comment
		Hex	Decimal	
CFH00	h(0)	H'1FE8	−0.01171875	
CFH01	h(1)	H'0039	0.02783203125	
CFH02	h(2)	H'003C	0.029296875	
CFH03	h(3)	H'1E56	−0.2080078125	
CFH04	h(4)	H'01DC	0.232421875	
CFH05	h(5)	H'06E1	0.85986328125	
CFH06	h(6)	H'01DC	0.232421875	
CFH07	h(7)	H'1E56	−0.2080078125	
CFH08	h(8)	H'003C	0.029296875	
CFH09	h(9)	H'0039	0.02783203125	
CFH10	h(10)	H'1FE8	−0.01171875	

37.2.2.24 Low-Pass Filter Coefficient h0(10) Register Channel n (PDMm_PDLFCH010RCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0164

Initial Value: H'0000_0400

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	LFH010[12:0]												
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	LFH010[12:0]	H'0400	RW	Low-pass (half-band decimation) filter coefficient h0(m) 13-bit signed data (Fraction bit [10:0]) For the initial value, see Value after Reset of Each Coefficient.

● Value after Reset of Each Coefficient

Symbol Name	Coefficient Name	Value after Reset		Comment
		Hex	Decimal	
LFH010	h0(10)	H'0400	0.5	Cut-off 7 kHz

37.2.2.25 Low-Pass Filter Coefficient h1(p) Register Channel n (PDMm_PDLFCH1pRCHn) (m = 0, p = 0 to 19)

Access Size: 32 bits

Address(es): <PDMm_base> + H'0168 + p × H'0004

Initial Value: H'0000_xxxx

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	LFH100[12:0]												
Initial Value	0	0	0	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R0	R0	R0	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	LFH100[12:0]	All x	RW	Low-pass (half-band decimation) filter coefficient h1(p) 13-bit signed data (Fraction bit [10:0]) For the initial value, see Value after Reset of Each Coefficient.

Remark x: Undefined

● Value after Reset of Each Coefficient

Symbol Name	Coefficient Name	Value after Reset		Comment
		Hex	Decimal	
LFH100	h1(0)	H'1FF8	−0.00390625	Cut-off 7 kHz
LFH101	h1(1)	H'000A	0.0048828125	
LFH102	h1(2)	H'1FF0	−0.0078125	
LFH103	h1(3)	H'0018	0.01171875	
LFH104	h1(4)	H'1FDC	−0.017578125	
LFH105	h1(5)	H'0034	0.025390625	
LFH106	h1(6)	H'1FB3	−0.03759765625	
LFH107	h1(7)	H'0076	0.0576171875	
LFH108	h1(8)	H'1F2E	−0.1025390625	
LFH109	h1(9)	H'0289	0.31689453125	
LFH110	h1(10)	H'0289	0.31689453125	
LFH111	h1(11)	H'1F2E	−0.1025390625	
LFH112	h1(12)	H'0076	0.0576171875	
LFH113	h1(13)	H'1FB3	−0.03759765625	
LFH114	h1(14)	H'0034	0.025390625	
LFH115	h1(15)	H'1FDC	−0.017578125	
LFH116	h1(16)	H'0018	0.01171875	
LFH117	h1(17)	H'1FF0	−0.0078125	
LFH118	h1(18)	H'000A	0.0048828125	
LFH119	h1(19)	H'1FF8	−0.00390625	

37.2.2.26 Sound Detection Lower Threshold Register Channel n (PDMm_PDSDLTRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'01B8

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	SDETL[19:16]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDETL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 0	SDETL[19:0]	All 0	RW	Sound detection lower limit This bit sets the lower limit value (20-bit signed data) for sound detection.

37.2.2.27 Sound Detection Upper Threshold Register Channel n (PDMm_PDSDUTRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'01BC

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	SDETU[19:16]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDETU[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 0	SDETU[19:0]	All 0	RW	Sound detection upper limit This bit sets the upper limit value (20-bit signed data) for sound detection.

37.2.2.28 Data Buffer Control Register Channel n (PDMm_PDDBCRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'01C0

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DATRITHR[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	R0	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	—	H'0	RW	Reserved This register is readable and the initial value can be written to it.
3	—	H'0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	DATRITHR [2:0]	H'0	RW	Data reception interrupt threshold 000b: Output interrupt when receiving 1 or more data 001b: Output interrupt when receiving 2 or more data 010b: Output interrupt when receiving 4 or more data 011b: Output interrupt when receiving 8 or more data Others: Output interrupt when receiving 16 or more data

37.2.2.29 Short Circuit Threshold Setting Register Channel n (PDMm_PDSCTSRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'01C4

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SCDH[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SCDL[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	SCDH[12:0]	All 0	RW	Short circuit detection High Continuous detection count Set the upper limit for the number of times “1” are input continuously to the PDM_DATn pin.
15 to 13	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	SCDL[12:0]	All 0	RW	Continuous detection count Set the upper limit for the number of times “0” are input continuously to the PDM_DATn pin.

37.2.2.30 Overvoltage Lower Threshold Register Channel n (PDMm_PDOVLTRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'01C8

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	OVDL[19:16]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OVDL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 0	OVDL[19:0]	All 0	RW	Overvoltage detection lower limit These bits set the lower limit of the active voltage value (20-bit signed data) for overvoltage detection.

37.2.2.31 Overvoltage Upper Threshold Register Channel n (PDMm_PDOVUTRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'01CC

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	OVDU[19:16]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OVDU[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 0	OVDU[19:0]	All 0	RW	Overvoltage detection upper limit These bits set the upper limit of the active voltage value (20-bit signed data) for overvoltage detection.

37.2.2.32 Data Read Control Register Channel n (PDMm_PDDRCRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'01E0

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DATRE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	DATRE	H'0	RW	Data read enable bit 0b: Do not allow to read data from buffer 1b: Allow to read data from buffer

37.2.2.33 Data Clear Register Channel n (PDMm_PDDCRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'01E4

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DATC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	DATC	H'0	R0W	The read value is always 0b. Data clear 0b: Do nothing 1b: Clear data

37.2.2.34 Data Read Register Channel n (PDMm_PDDRRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'01E8

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	DAT[19:16]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R0	R0	R0	R0	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DAT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 20	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 0	DAT[19:0]	All 0	R	Data 20-bit signed sound data or 16-bit signed sound data is read out. In case of 20-bit mode (PDMDSRCHn.DBIS[3:0] = 0xxxb), the signed bit is [19]. In case of 16-bit mode (PDMDSRCHn.DBIS[3:0] = 1xxxb), the signed bits are [19:15]. Writing is invalid.

37.2.2.35 Data Status Register Channel n (PDMm_PDDSRCHn) (m = 0)

Access Size: 32 bits

Address(es): <PDMm_base> + H'01EC

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DATNUM[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	DATNUM[7:0]	All 0	R	Number of data stored in buffer When PDDRCRCHn.DATRE = 0b, 0b is read out. When PDDRCRCHn.DATRE = 1b, the number of data stored in buffer is read out. In case that buffer size is 64 stages, the maximum value is 63.

37.2.3 Mirroring Register

The common registers except for the version register have mirroring registers in each channel. PDCSR.EDFn is a register which ORs all the error flags of the mirroring registers in each channel.

Use the common registers for synchronous control between channels.

Table 37.5 Mirroring Register

Common Register		Mirroring Register in Each Channel	
Register Name	Bit Name	Register Name	Bit Name
PDCSTRTR	STRTRGn	PDSTRTRCHn	STRTRG
PDCSTPTR	STPTRGn	PDSTPTRCHn	STPTRG
PDCCHGTR	CHGTRGn	PDCHGTRCHn	CHGTRG
PDCICR	ISDEn	PDICRCHn	ISDE
	IDREn		IDRE
	IEDEn		IEDE
PDCSR	STATEn	PDSRCHn	STATE
	SDFn		SDF
	DRFn		DRF
	EDFn		SCDF or OVLDF or OVUDF or BFOWDF
PDCSCR	SDFCn	PDSCRCHn	SDFC
PDCSDCR	SDEn	PDSDCRCHn	SDE
PDCDRCR	DATREn	PDDRCRCHn	DATRE
PDCDCR	DATCn	PDDCRCHn	DATC

37.3 Operation

This section describes the PDM basic control method. See **Section 37.4.6, Setting PDM_CLKn (n = 0, 1, 2) Frequency and Sampling Frequency Generated by the Unit** for information on setting the PDM_CLKn (n = 0, 1, 2) frequency and the sampling frequency generated by the PDM unit, and see **Section 37.4.7, DMA Transfer Method** for information on transferring data with the DMA master.

37.3.1 Clock Specifications

The frequency-divided core clock is generated and output to PDM_CLKn.

Table 37.6 lists the guaranteed clock frequency and clock mode in PDM-IF. Core clock (CCLK) and PCLK are asynchronous.

In this section, read “PCLK” as “PDM_PCLK”, “CCLK” as “PDM_CCLK”.

Table 37.6 Guaranteed Clock Frequency and Clock Mode

CCLK	PCLK PCLK_SFR	PDM_CLKn		
Frequency (MHz)	Frequency (MHz)	Frequency (MHz)	Supported Core Clock Division Ratio	Setting Value (PDSFCRCHn.CKDIV[3:0])
8	1 to 100	4	2	0000b
		2	4	0001b
		1.33	6	0010b
		1	8	0011b
		0.8	10	0100b
		0.67	12	0101b
		0.57	14	0110b
		0.5	16	0111b
		0.44	18	1000b
		0.4	20	1001b
		0.36	22	1010b
		0.33	24	1011b
		0.31	26	1100b
		0.29	28	1101b
		0.27	30	1110b
		0.25	32	1111b

37.3.2 Channel Start and Channel Stop

The channel start and the channel stop is controlled by the registers listed in **Table 37.7**.

Set the channels to be synchronized to 1 at the same time by using the common registers.

Set the channel start according to **Section 37.4.1, Start Flow**.

Set the channel stop according to **Section 37.4.2, Stop Flow**.

Table 37.7 Channel Start Trigger and Channel Stop Trigger

	Register Bit Name	
Category	Common Register	Channel Register
Channel start	PDCSTRTR.STRTRG0	PDSTRTRCH0.STRTRG
	PDCSTRTR.STRTRG1	PDSTRTRCH1.STRTRG
	PDCSTRTR.STRTRG2	PDSTRTRCH2.STRTRG
Channel stop	PDCSTPTR.STPTRG0	PDSTPTRCH0.STPTRG
	PDCSTPTR.STPTRG1	PDSTPTRCH1.STPTRG
	PDCSTPTR.STPTRG2	PDSTPTRCH2.STPTRG

37.3.3 PDM_CLKn's Frequency Change

The external microphone transitions the power mode according to the input clock frequency. PDM-IF enables the transition of microphone's power mode by changing PDM_CLKn's operating frequency which is output to the microphone when operating. The PDM_CLKn's operating frequency can be changed by using the PDSFCRCHn.CKDIV[3:0] bits. When the frequency is changed, the decimation ratio of the sinc filter also needs to be changed. Set PDSFCRCHn.SINCDEC[7:0] in accordance with the sampling frequency after the decimation. Furthermore, when the decimation ratio of the sinc filter is changed, its output range is changed. Set PDSFCRCHn.SINCRNG[4:0] together in accordance with the output range. Refer to **Table 37.8,** which lists example settings when the sampling frequency after the decimation is 30 kHz*[1].

For the method of calculating the set values of PDSFCRCHn.CKDIV[3:0], PDSFCRCHn.SINCDEC[7:0], and PDSFCRCHn.SINCRNG[4:0] for the expected sampling frequency of PDM_CLKn, refer to **Section 37.4.6, Setting PDM_CLKn (n = 0, 1, 2) Frequency and Sampling Frequency Generated by the Unit**.

After changing the configuration of these registers, software needs to wait for the settling time of microphone and PDM-IF channel. Refer to **Section 37.4.4, Low Power Mode Transition Flow** and **Section 37.4.5, Normal Mode Transition Flow** for details.

Note 1. If the same filter coefficient has been set in filtering after the sinc filter, the characteristics of the filter changes with the change of the sampling frequency.

37.3.4 Filter

37.3.4.1 Data Clipping and Shifting between Filters

Voice sample data is clipped or shifted between filters. **Figure 37.2** shows this operation.

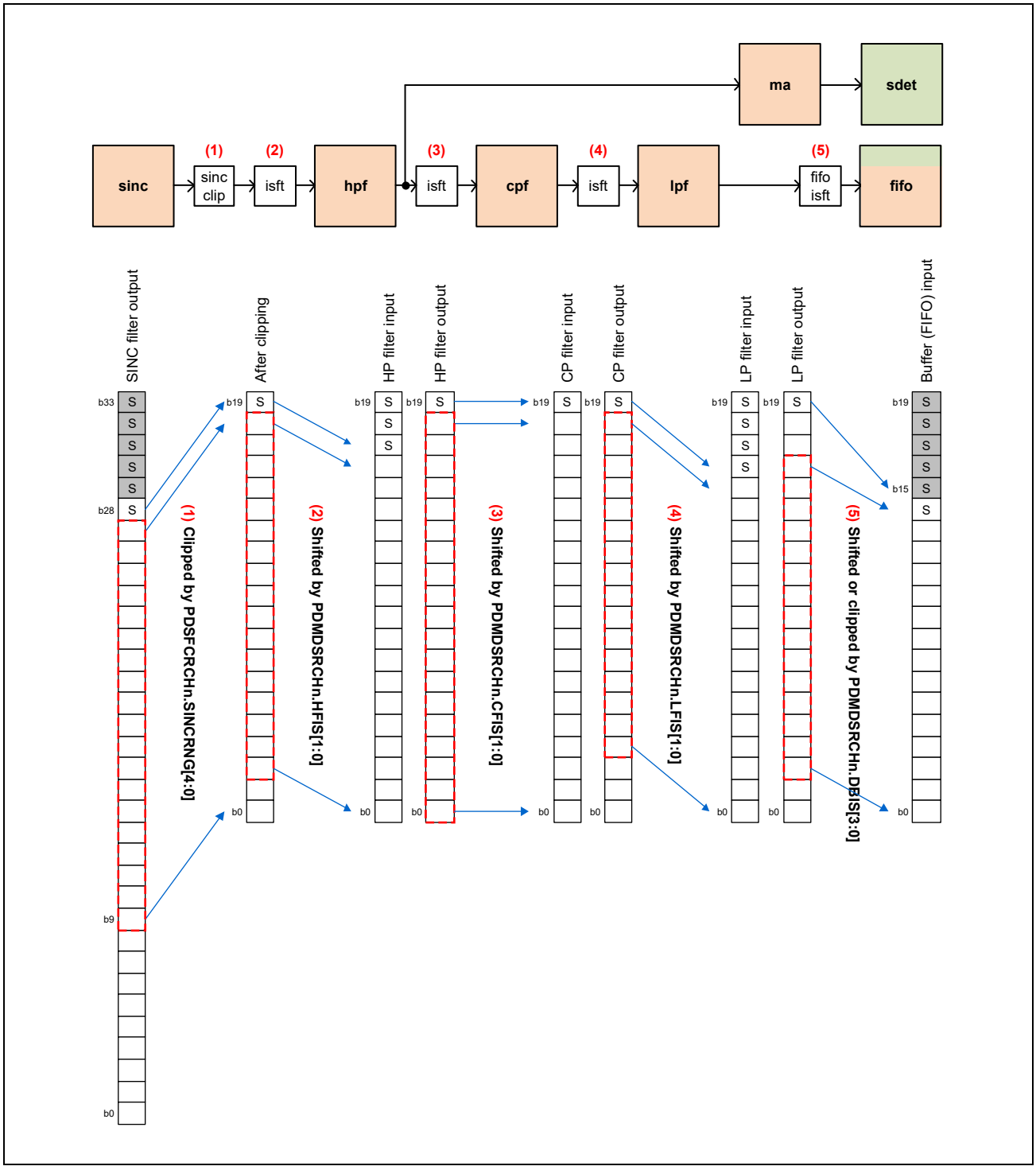


Figure 37.2 Data Clipping and Shifting between Filters

a) Sinc filter output clip

Table 37.8 lists the available settings of the sinc filter and data clipping.

It is controlled by PDSFCRCHn.SINCRNG[4:0].
It is prohibited to set other than those in **Table 37.8**.

Table 37.8 Sinc Filter Output Clip (1/2)

CCLK (MHz)	Sinc Filter Order	PDMDSRCHn. SFMD[2:0]	PDM_CLK's Division Ratio	PDSFCRCHn. CKDIV[3:0]	Sinc filter Decimation Ratio	PDSFCRCHn. SINCDEC[7:0]	Sinc Filter Output Effective Bits	Sinc Filter Output Clipping (20 bits)	PDSFCRCHn. SINCRNG[4:0]	Resolution	PDM_CLKn Frequency (MHz)	PDMIF Output Sampling Frequency (kHz)
8	4	H'0	2	H'0	125	H'7C	b28 to b0	b28 to b9	00101b	20	4.0	32
			4	H'1	62	H'3D	b24 to b0	b24 to b5	01001b	20	2.0	32.26
			6	H'2	42	H'29	b22 to b0	b22 to b3	01011b	20	1.33	31.75
			8	H'3	31	H'1E	b20 to b0	b20 to b1	01101b	20	1.0	32.26
			10	H'4	25	H'18	b19 to b0	b19 to b0	01110b	20	0.8	32
			12	H'5	21	H'14	b18 to b0	b18 to b0, 0b	01111b	19	0.67	31.75
			14	H'6	18	H'11	b17 to b0	b17 to b0, 00b	10000b	18	0.57	31.75
			16	H'7	16	H'0F	b17 to b0	b17 to b0, 00b	10000b	18	0.50	31.25
			18	H'8	14	H'0D	b16 to b0	b16 to b0, 000b	10001b	17	0.44	31.75
			20	H'9	12	H'0B	b15 to b0	b15 to b0, 0000b	10010b	16	0.40	33.33
			22	H'A	11	H'0A	b14 to b0	b14 to b0, 00000b	10011b	15	0.36	33.06
			24	H'B	10	H'09	b14 to b0	b14 to b0, 00000b	10011b	15	0.33	33.33
			26	H'C	9	H'08	b13 to b0	b13 to b0, 000000b	10100b	14	0.31	34.19
			28	H'D	9	H'08	b13 to b0	b13 to b0, 000000b	10100b	14	0.29	31.75
			30	H'E	8	H'07	b13 to b0	b13 to b0, 000000b	10100b	14	0.27	33.33
			32	H'F	8	H'07	b13 to b0	b13 to b0, 000000b	10100b	14	0.25	31.25
8	3	H'3	2	H'0	125	H'7C	b21 to b0	b21 to b2	01100b	20	4.0	32
			4	H'1	62	H'3D	b18 to b0	b18 to b0, 0b	01111b	19	2.0	32.26
			6	H'2	42	H'29	b17 to b0	b17 to b0, 00b	10000b	18	1.33	31.75
			8	H'3	31	H'1E	b15 to b0	b15 to b0, 0000b	10010b	16	1.0	32.26
			10	H'4	25	H'18	b14 to b0	b14 to b0, 00000b	10011b	15	0.8	32
			12	H'5	21	H'14	b14 to b0	b14 to b0, 00000b	10011b	15	0.67	31.75
			14	H'6	18	H'11	b13 to b0	b13 to b0, 000000b	10100b	14	0.57	31.75
			16	H'7	16	H'0F	b13 to b0	b13 to b0, 000000b	10100b	14	0.50	31.25
			18	H'8	14	H'0D	b12 to b0	b12 to b0, 0000000b	10101b	13	0.44	31.75
			20	H'9	12	H'0B	b11 to b0	b11 to b0, 00000000b	10110b	12	0.40	33.33
			22	H'A	11	H'0A	b11 to b0	b11 to b0, 00000000b	10110b	12	0.36	33.06
			24	H'B	10	H'09	b10 to b0	b10 to b0, 000000000b	10111b	11	0.33	33.33
			26	H'C	9	H'08	b10 to b0	b10 to b0, 000000000b	10111b	11	0.31	34.19
			28	H'D	9	H'08	b10 to b0	b10 to b0, 000000000b	10111b	11	0.29	31.75
			30	H'E	8	H'07	b10 to b0	b10 to b0, 000000000b	10111b	11	0.27	33.33
			32	H'F	8	H'07	b10 to b0	b10 to b0, 000000000b	10111b	11	0.25	31.25

Table 37.8 Sinc Filter Output Clip (2/2)

CCLK (MHz)	Sinc Filter Order	PDMDSRCHn. SFMD[2:0]	PDM_CLK's Division Ratio	PDSFCRCHn. CKDIV[3:0]	Sinc filter Decimation Ratio	PDSFCRCHn. SINCDEC[7:0]	Sinc Filter Output Effective Bits	Sinc Filter Output Clipping (20 bits)	PDSFCRCHn. SINCRNG[4:0]	Resolution	PDM_CLKn Frequency (MHz)	SINC Filter Output Sampling Frequency (kHz)
8	2	H'2	2	H'0	125	H'7C	b14 to b0	b14 to b0, 00000b	10011b	15	4.0	32
			4	H'1	62	H'3D	b12 to b0	b12 to b0, 0000000b	10101b	13	2.0	32.26
			6	H'2	42	H'29	b11 to b0	b11 to b0, 00000000b	10110b	12	1.33	31.75
			8	H'3	31	H'1E	b10 to b0	b10 to b0, 000000000b	10111b	11	1.0	32.26
			10	H'4	25	H'18	b10 to b0	b10 to b0, 000000000b	10111b	11	0.8	32
			12	H'5	21	H'14	b9 to b0	b9 to b0, 0000000000b	11000b	10	0.67	31.75
			14	H'6	18	H'11	b9 to b0	b9 to b0, 0000000000b	11000b	10	0.57	31.75
			16	H'7	16	H'0F	b9 to b0	b9 to b0, 0000000000b	11000b	10	0.50	31.25
			18	H'8	14	H'0D	b8 to b0	b8 to b0, 00000000000b	11001b	9	0.44	31.75
			20	H'9	12	H'0B	b8 to b0	b8 to b0, 00000000000b	11001b	9	0.40	33.33
			22	H'A	11	H'0A	b7 to b0	b7 to b0, 000000000000b	11010b	8	0.36	33.06
			24	H'B	10	H'09	b7 to b0	b7 to b0, 000000000000b	11010b	8	0.33	33.33
			26	H'C	9	H'08	b7 to b0	b7 to b0, 000000000000b	11010b	8	0.31	34.19
			28	H'D	9	H'08	b7 to b0	b7 to b0, 000000000000b	11010b	8	0.29	31.75
			30	H'E	8	H'07	b7 to b0	b7 to b0, 000000000000b	11010b	8	0.27	33.33
			32	H'F	8	H'07	b7 to b0	b7 to b0, 000000000000b	11010b	8	0.25	31.25
8	1	H'1	2	H'0	125	H'7C	b7 to b0	b7 to b0, 000000000000b	11010b	8	4.0	32
			4	H'1	62	H'3D	b6 to b0	b6 to b0, 0000000000000b	11011b	7	2.0	32.26
			6	H'2	42	H'29	b6 to b0	b6 to b0, 0000000000000b	11011b	7	1.33	31.75
			8	H'3	31	H'1E	b5 to b0	b5 to b0, 00000000000000b	11100b	6	1.0	32.26
			10	H'4	25	H'18	b5 to b0	b5 to b0, 00000000000000b	11100b	6	0.8	32
			12	H'5	21	H'14	b5 to b0	b5 to b0, 00000000000000b	11100b	6	0.67	31.75
			14	H'6	18	H'11	b5 to b0	b5 to b0, 00000000000000b	11100b	6	0.57	31.75
			16	H'7	16	H'0F	b5 to b0	b5 to b0, 00000000000000b	11100b	6	0.50	31.25
			18	H'8	14	H'0D	b4 to b0	b4 to b0, 000000000000000b	11101b	5	0.44	31.75
			20	H'9	12	H'0B	b4 to b0	b4 to b0, 000000000000000b	11101b	5	0.40	33.33
			22	H'A	11	H'0A	b4 to b0	b4 to b0, 000000000000000b	11101b	5	0.36	33.06
			24	H'B	10	H'09	b4 to b0	b4 to b0, 000000000000000b	11101b	5	0.33	33.33
			26	H'C	9	H'08	b4 to b0	b4 to b0, 000000000000000b	11101b	5	0.31	34.19
			28	H'D	9	H'08	b4 to b0	b4 to b0, 000000000000000b	11101b	5	0.29	31.75
			30	H'E	8	H'07	b4 to b0	b4 to b0, 000000000000000b	11101b	5	0.27	33.33
			32	H'F	8	H'07	b4 to b0	b4 to b0, 000000000000000b	11101b	5	0.25	31.25

An overflow or an underflow never occurs as long as the combinations which are shown in **Table 37.8** are set to PDMDSRCHn.SFMD[2:0], PDSFCRCHn.CKDIV[3:0], PDSFCRCHn.SINCDEC[7:0] and PDSFCRCHn.SINCRNG[4:0]. However, when other combinations are set to them, the overflow or the underflow might occur. When the overflow occurs, H'7FFFF is output. When the underflow occurs, H'80000 is output. Use the overvoltage detection function if you want to know if the overflow or the underflow occurs.

b) High-pass filter input shift

The clipped sinc filter output is right-shifted and input to the high-pass filter.

It is controlled by PDMDSRCHn.HFIS[1:0].

Table 37.9 High-pass Filter Input Shift

PDMDSRCHn.HFIS[1:0]	Function	Shifted Data*1
H'0	No shift	S, b18-b0
H'1	1-bit right-shift	S, S, b18-b1
H'2	2-bit right-shift	S, S, S, b18-b2
H'3	3-bit right-shift	S, S, S, S, b18-b3

Note 1. S: Signed bit (= bit 19)

c) Compensation filter input shift

The high-pass filter output is right-shifted and input to the compensation filter.

It is controlled by PDMDSRCHn.CFIS[1:0].

Table 37.10 Compensation Filter Input Shift

PDMDSRCHn.CFIS[1:0]	Function	Shifted Data*1
H'0	No shift	S, b18-b0
H'1	1-bit right-shift	S, S, b18-b1
H'2	2-bit right-shift	S, S, S, b18-b2
H'3	3-bit right-shift	S, S, S, S, b18-b3

Note 1. S: Signed bit (= bit 19)

d) Low-pass filter input shift

The compensation filter output is right-shifted and input to the low-pass (half-band decimation) filter.

It is controlled by PDMDSRCHn.LFIS[1:0].

Table 37.11 Low-pass Filter Input Shift

PDMDSRCHn.LFIS[1:0]	Function	Shifted Data*1
H'0	No shift	S, b18-b0
H'1	1-bit right-shift	S, S, b18-b1
H'2	2-bit right-shift	S, S, S, b18-b2
H'3	3-bit right-shift	S, S, S, S, b18-b3

Note 1. S: Signed bit (= bit 19)

e) Buffer input shift or clip

The compensation filter output is right-shifted, or 16-bit data is clipped from the compensation filter output. Then, the shifted or clipped data is input to buffer.

It is controlled by PDMDSRCHn.DBIS[3:0].

Table 37.12 Buffer Input Shift or Clip

PDMDSRCHn.DBIS[3:0]	Function	Valid Data Width	Shifted or Clipped Data*1
0000b	No shift	20-bit	S, b18-b0
0001b	1-bit right-shift	20-bit	S, S, b18-b1
0010b	2-bit right-shift	20-bit	S, S, S, b18-b2
0011b	3-bit right-shift	20-bit	S, S, S, S, b18-b3
1000b	Signed bit and b18-b4 is clipped	16-bit	S, b18-b4
1001b	Signed bit and b17-b3 is clipped*2	16-bit	S, b17-b3
1010b	Signed bit and b16-b2 is clipped*2	16-bit	S, b16-b2
1011b	Signed bit and b15-b1 is clipped*2	16-bit	S, b15-b1
1100b	Signed bit and b14-b0 is clipped*2	16-bit	S, b14-b0

Note 1. S: Signed bit (= bit 19)

Note 2. When an overflow occurs, H'7FFF is input to the buffer.
When an underflow occurs, H'8000 is input to the buffer.

37.3.4.2 Sinc Filter

The operating clock of the sinc filter is CCLK. Sinc filter is operated by using the 1-bit stream data latched with PDM_CLKn's rise-edge or fall-edge. Refer to **Section 37.3.5, Data Input Control for Stereo Sound** for details.

Figure 37.3 shows a sinc filter block diagram.

The differentiation stage is operated by decimation clock (the frequency is 1/M of PDM_CLKn). M is the decimation ratio and is set in the PDSFCRCHn.SINCDEC[7:0] bits. The filter result is output in register every decimation clock.

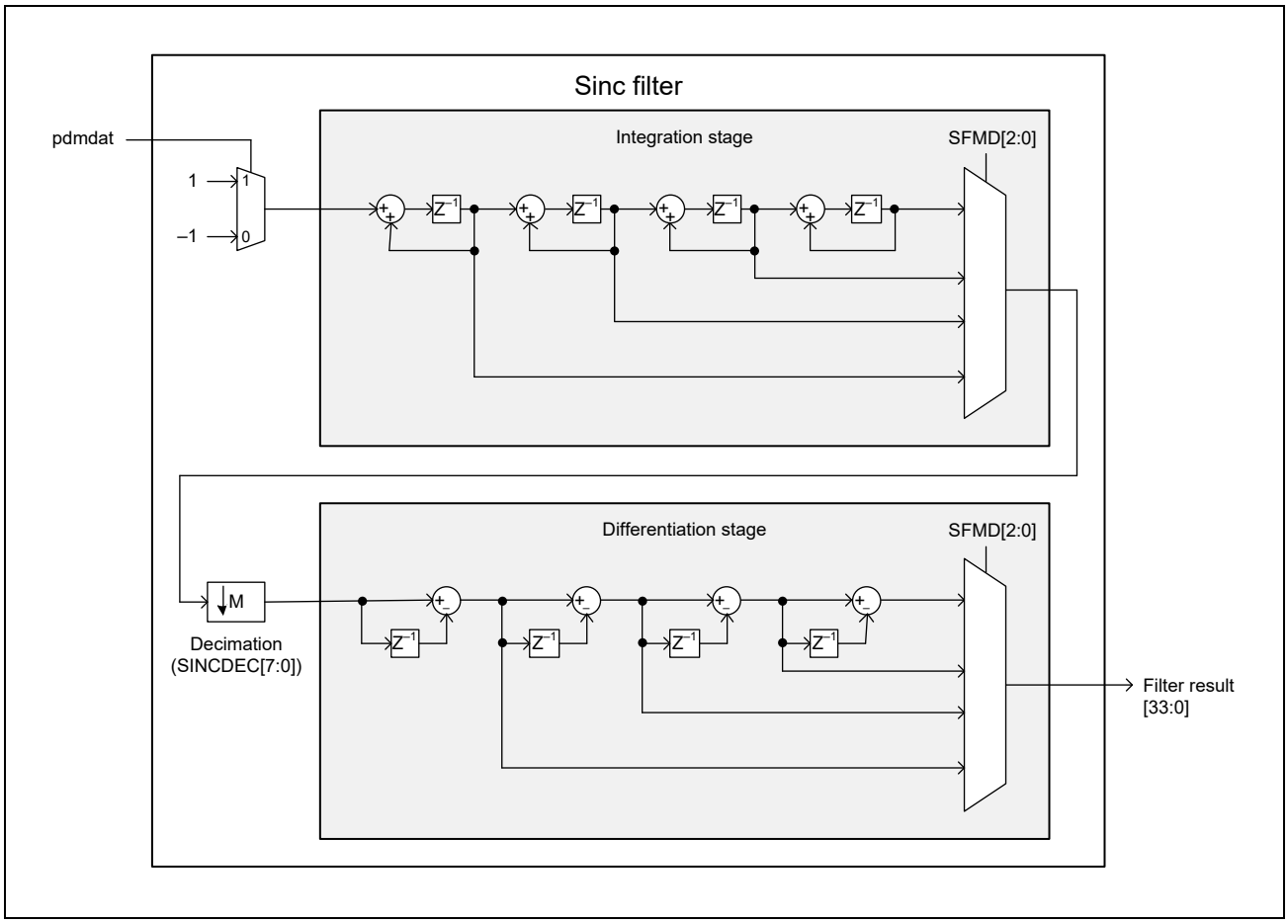


Figure 37.3 Sinc Filter Block Diagram

37.3.4.3 High-pass Filter

The operating clock of the high-pass filter is CCLK. The high-pass filter is operated by using the result of the sinc filter.

Figure 37.4 shows a high-pass filter block diagram.

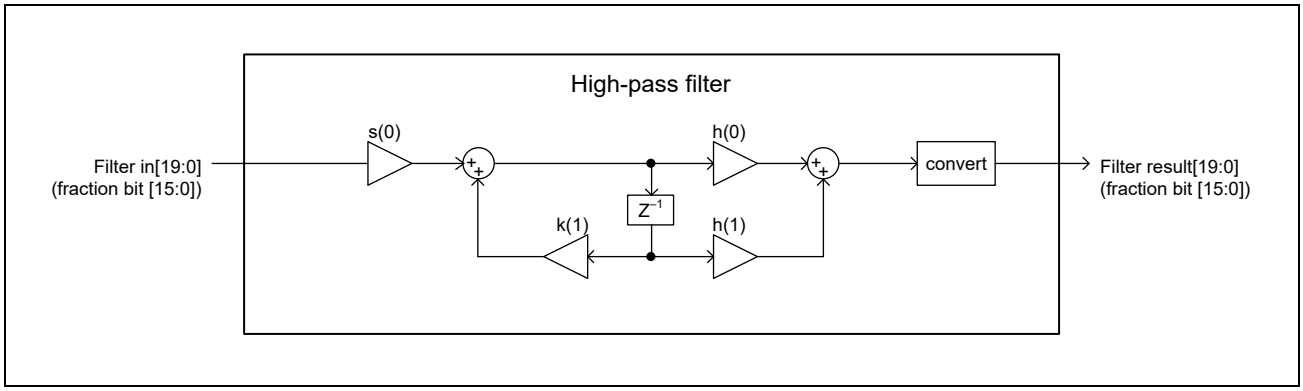


Figure 37.4 High-pass Filter Block Diagram

Table 37.13 lists the high-pass filter specification.

Table 37.13 High-pass Filter Specification

Function		Specification
Input	Bit width	Signed 20-bit (fraction 16-bit)
Output	Bit width	Signed 20-bit (fraction 16-bit)
Coefficient	Bit width	Signed 16-bit (fraction 14-bit) Refer to **Section 37.2.2.20, High-Pass Filter Coefficient s(0) Register Channel n (PDMm_PDHFCS0RCHn) (m = 0)**, **Section 37.2.2.21, High-Pass Filter Coefficient k(1) Register Channel n (PDMm_PDHFCK1RCHn) (m = 0)**, and **Section 37.2.2.22, High-Pass Filter Coefficient h(p) Register Channel n (PDMm_PDHFCHpRCHn) (m = 0, p = 0, 1)** for details.
Delay Z	Bit width	Signed 32-bit (fraction 18-bit)
Multiplier	Input A bit width	Signed 32-bit (fraction 18-bit)
	Input B bit width	Signed 16-bit (fraction 14-bit)
	Output bit width	Signed 47-bit (fraction 32-bit)
Adder	Input A bit width	Signed 32-bit (fraction 18-bit)
	Input B bit width	Signed 32-bit (fraction 18-bit)
	Output bit width	Signed 33-bit (fraction 18-bit)
Output conversion	Input bit width	Signed 32-bit (fraction 18-bit)
	Output bit width	Signed 20-bit (fraction 16-bit)
Excessive fraction bits processing		Excessive fraction bits are rounded down at functional unit input or final output.

37.3.4.4 Compensation Filter

The operating clock of the compensation filter is CCLK. The compensation filter is operated by using the result of the high-pass filter.

Figure 37.5 shows a compensation filter block diagram.

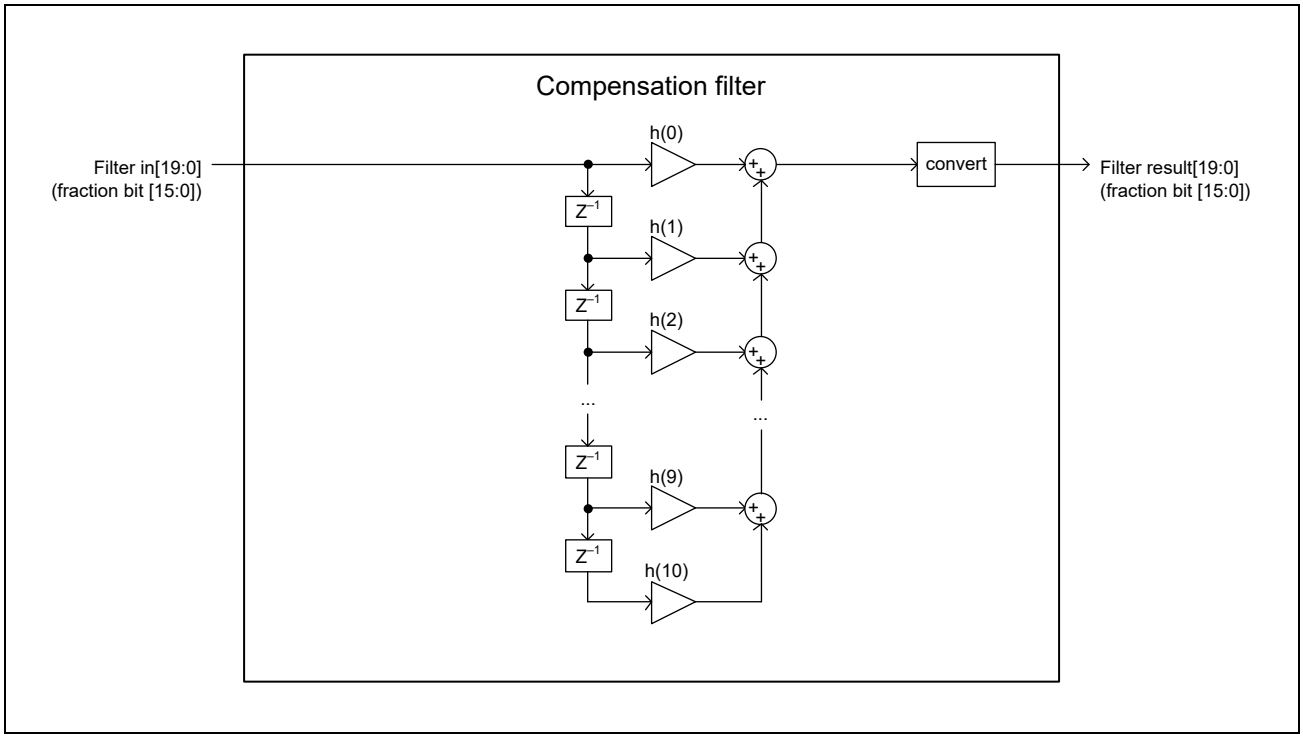


Figure 37.5 Compensation Filter Block Diagram

Table 37.14 shows a compensation filter specification.

Table 37.14 Compensation Filter Specification

Function		Specification
Input	Bit width	Signed 20-bit (fraction 16-bit)
Output	Bit width	Signed 20-bit (fraction 16-bit)
Coefficient	Bit width	Signed 13-bit (fraction 11-bit) Refer to **Section 37.2.2.23, Compensation Filter Coefficient h(p) Register Channel n (PDMm_PDCFCHpRCHn) (m = 0, p = 00 to 10)** for details.
Delay Z	Bit width	Signed 20-bit (fraction 16-bit)
Multiplier	Input A bit width	Signed 20-bit (fraction 16-bit)
	Input B bit width	Signed 13-bit (fraction 11-bit)
	Output bit width	Signed 32-bit (fraction 27-bit)
Adder	Input A bit width	Signed 22-bit (fraction 18-bit)
	Input B bit width	Signed 22-bit (fraction 18-bit)
	Output bit width	Signed 23-bit (fraction 18-bit)
Output conversion	Input bit width	Signed 22-bit (fraction 18-bit)
	Output bit width	Signed 20-bit (fraction 16-bit)
Excessive fraction bits processing		Excessive fraction bits are rounded down at functional unit input or final output.

37.3.4.5 Low-Pass (Half-Band Decimation) Filter

The operating clock of the low-pass filter is CCLK. The low-pass filter is operated by using the result of the compensation filter. Output data is decimated to a half of input data.

Figure 37.6 shows a low-pass filter block diagram.

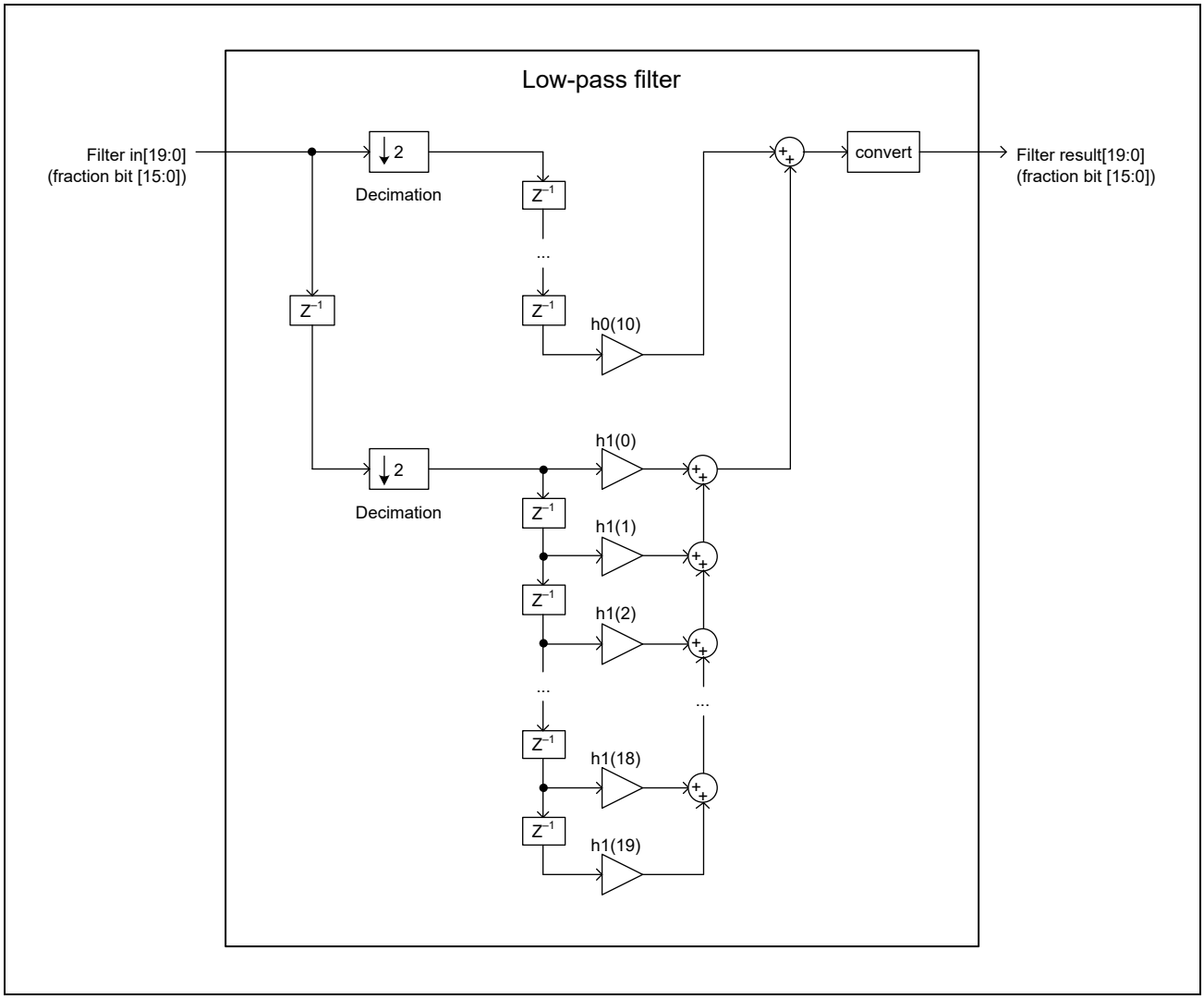


Figure 37.6 Low-Pass Filter Block Diagram

Table 37.15 lists a low-pass filter specification.

Table 37.15 Low-Pass Filter Specification

Function		Specification
Input	Bit width	Signed 20-bit (fraction 16-bit)
Output	Bit width	Signed 20-bit (fraction 16-bit)
Coefficient	Bit width	Signed 13-bit (fraction 11-bit) Refer to **Section 37.2.2.33, Data Clear Register Channel n (PDMm_PDDCRCHn) (m = 0)** for detail.
Delay Z	Bit width	Signed 20-bit (fraction 16-bit)
Multiplier	Input A bit width	Signed 20-bit (fraction 16-bit)
	Input B bit width	Signed 13-bit (fraction 11-bit)
	Output bit width	Signed 32-bit (fraction 27-bit)
Adder	Input A bit width	Signed 22-bit (fraction 18-bit)
	Input B bit width	Signed 22-bit (fraction 18-bit)
	Output bit width	Signed 23-bit (fraction 18-bit)
Output conversion	Input bit width	Signed 22-bit (fraction 18-bit)
	Output bit width	Signed 20-bit (fraction 16-bit)
Excessive fraction bits processing		Excessive fraction bits are rounded down at functional unit input or final output.

37.3.4.6 Moving Average Filter

The operating clock of the moving average filter is CCLK. The moving average filter is operated by using the result of the high-pass filter. The order of the moving average filter can be changed by the setting of the PDMDSRCHn.SDMAMD[1:0] bits.

Figure 37.7 shows a moving average filter block diagram.

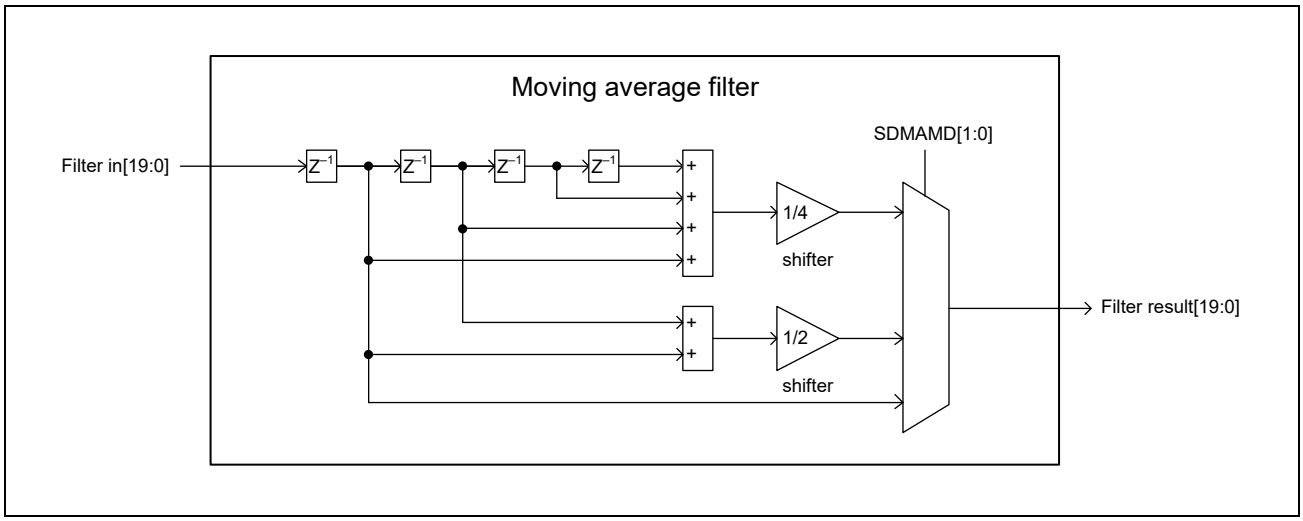


Figure 37.7 Moving Average Filter Block Diagram

Table 37.16 lists a moving average filter specification.

Table 37.16 Moving Average Filter Specification

Function		Specification
Input	Bit width	Signed 20-bit (fraction 16-bit)
Output	Bit width	Signed 20-bit (fraction 16-bit)
Delay Z	Bit width	Signed 20-bit (fraction 16-bit)
Excessive fraction bits processing		Excessive fraction bits are rounded down at shifter.

37.3.4.7 Settling Time of Channel Activation / Setting Change

The settling time of channel activation is the time after a start trigger PDCSTRTR.STRTRGn (or PDSTRTRCHn.STRTRG) is set before the filter result is stable and output. The settling time of channel setting change is the time after a change trigger PDCCHGTR.CHGTRGn (or PDCHGTRCHn.CHGTRG) is set before the filter result is stable and output. **Figure 37.8** shows the settling time measurement point.

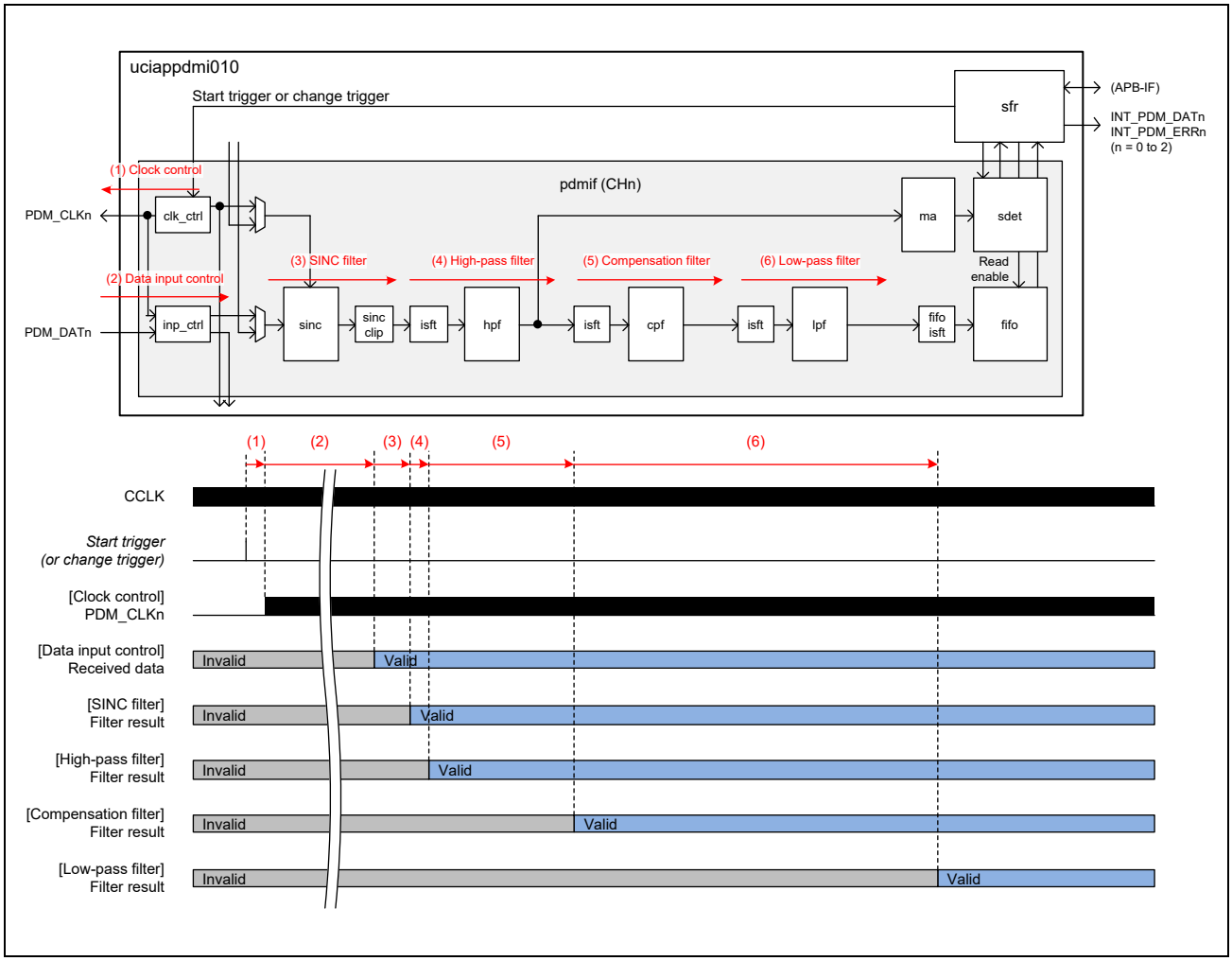


Figure 37.8 Settling Time of Channel Activation

The settling time is calculated as shown in **Table 37.17**.

Table 37.17 Settling Time of Channel Activation / Setting Change

No.	Function	Settling Time	
		Channel Activation	Channel Setting Change
(1)	Clock control	(4 + (D/2)) × Cp	(3 + Do + (D/2)) × Cp
(2)	Data input control	(D/2) × Cp + (Stable time for microphone*1)	
(3)	Sinc filter	(K × D × M) × Cp	
(4)	High-pass filter	(D × M + 5) × Cp	
(5)	Compensation filter	(10 × D × M + 12) × Cp	
(6)	Low-pass filter	(38 × D × M + 22) × Cp	

Note: Cp: CCLK period
D: PDM_CLKn's division ratio
Do: PDM_CLKn's division ratio (old setting)
K: Sinc filter order
M: Sinc filter decimation ratio

Note 1. Wakeup time from low power mode, Switching time, etc. Refer to the datasheet of microphone for details.

37.3.5 Data Input Control for Stereo Sound

The microphone which supports the stereo sound outputs the right sound synchronized with PDM_CLKn's rise-edge and the left sound synchronized with PDM_CLKn's fall-edge.

PDM-IF can process the right sound and the left sound using different channels.

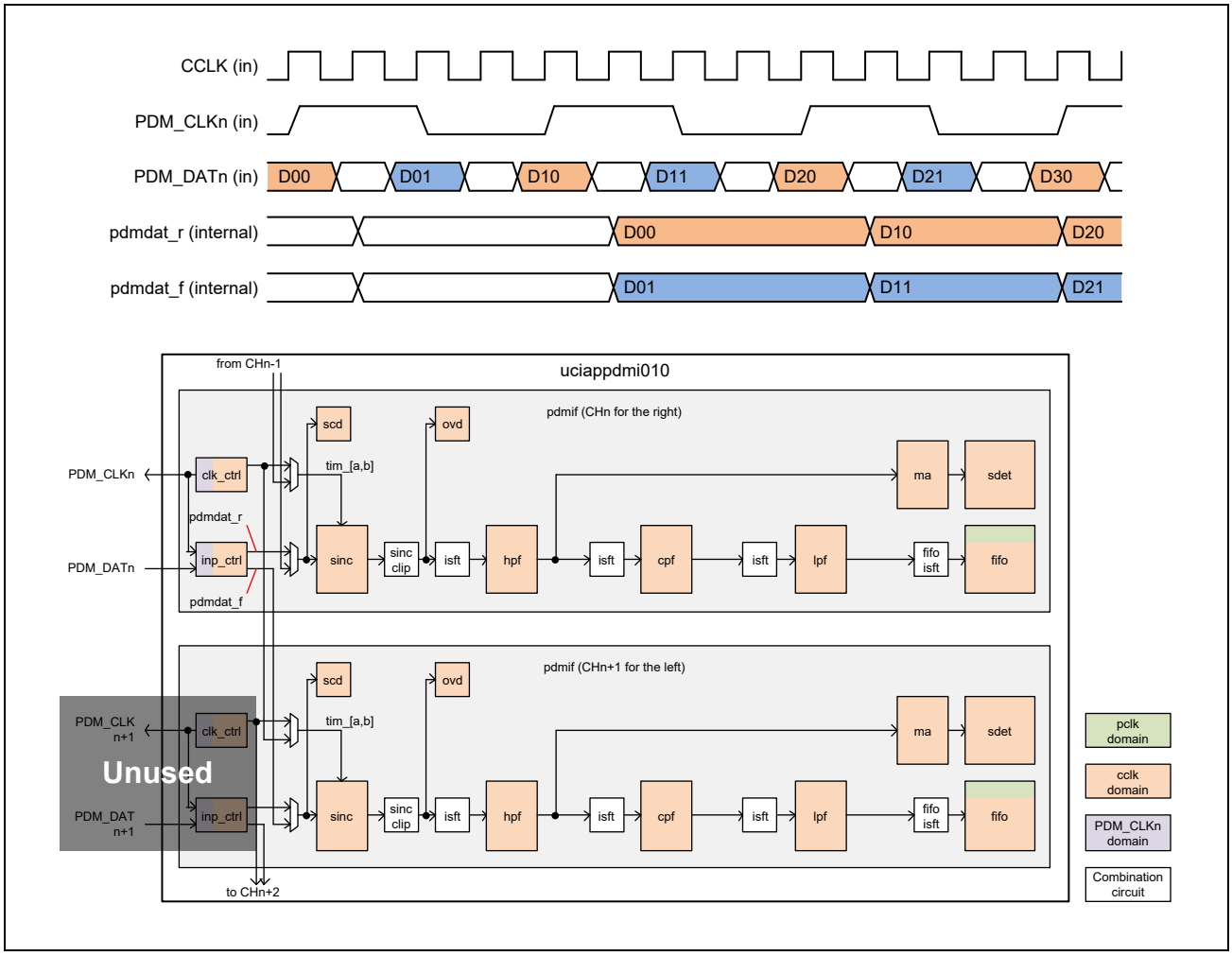


Figure 37.9 Data Input Control for Stereo Sound

Table 37.18 lists the channel combinations of stereo sound processing.

Table 37.18 Channel Combinations of Stereo Sound Processing

Input	Three Channels PDM-IF		Two Channels PDM-IF	
	Data Processing Latched at Rise-Edge (Right Sound)	Data Processing Latched at Fall-Edge (Left Sound)	Data Processing Latched at Rise-Edge (Right Sound)	Data Processing Latched at Fall-Edge (Left Sound)
PDM_CLK0 PDM_DAT0	Channel 0	Channel 1	Channel 0	Channel 1
PDM_CLK1 PDM_DAT1	Channel 1	Channel 2	Channel 1	Channel 0
PDM_CLK2 PDM_DAT2	Channel 2	Channel 0	Not implemented	

Table 37.19 lists an example of register settings for stereo sound processing using channel 0 and channel 1.

Table 37.19 Register Settings for Stereo Sound Processing

Register	Bit	Channel 0 Setting	Channel 1 Setting
PDICRCHn	All bits	Channel unique settings	Channel unique settings
PDSDCRCHn	All bits	Channel unique settings	Channel unique settings
PDMDSRCHn	INPSEL	0 (Use channel 0 data synchronized with rise-edge)	1 (Use channel 0 data synchronized with fall-edge)
	SFMD[2:0]	Same settings for channel 0 and channel 1 (Mandatory)	
	HFIS[1:0]	Channel unique settings	Channel unique settings
	HFMD	Same settings for channel 0 and channel 1 (Mandatory)	
	CFIS[1:0]	Channel unique settings	Channel unique settings
	CFMD	Same settings for channel 0 and channel 1 (Mandatory)	
	LFIS[1:0]	Channel unique settings	Channel unique settings
	LFMD	Same settings for channel 0 and channel 1 (Mandatory)	
	SDHFMD	Same settings for channel 0 and channel 1 (Mandatory)	
	SDMAMD[1:0]	Same settings for channel 0 and channel 1 (Mandatory)	
	DBIS[3:0]	Channel unique settings	Channel unique settings
PDSFCRCHn	CKDIV[3:0]	Setting of the clock which is supplied to the microphone connected to channel 0	Unuse
	SINCDEC[7:0]	Same settings for channel 0 and channel 1 (Mandatory)	
	SINCRNG[4:0]	Same settings for channel 0 and channel 1 (Mandatory)	
PDHFCS0RCHn	All bits	Channel unique settings	Channel unique settings
PDHFCK1RCHn	All bits	Channel unique settings	Channel unique settings
PDHFCH0RCHn	All bits	Channel unique settings	Channel unique settings
PDHFCH1RCHn	All bits	Channel unique settings	Channel unique settings
PDCFCHmRCHn	All bits	Channel unique settings	Channel unique settings
PDLFCH0mRCHn	All bits	Channel unique settings	Channel unique settings
PDLFCH1mRCHn	All bits	Channel unique settings	Channel unique settings
PDSDLTRCHn	All bits	Channel unique settings	Channel unique settings
PDSDUTRCHn	All bits	Channel unique settings	Channel unique settings
PDDBCRCHn	All bits	Same settings for channel 0 and channel 1 (Mandatory)	
PDSCTSRCHn	All bits	Channel unique settings	Channel unique settings
PDOVLTRCHn	All bits	Channel unique settings	Channel unique settings
PDOVUTRCHn	All bits	Channel unique settings	Channel unique settings

37.3.6 Sound Detection

The operating clock of the sound detection function is CCLK. When PDSDCRCHn.SDE = 1b, the function judges the sound detection using the moving average filter results.

A sound detection flag (PDSRCHn.SDF) is set to 1b when the result is above the upper limit PDSDUTRCHn.SDETU[19:0]. The sound detection flag is also set to 1b when the result is below the lower limit PDSDLTRCHn.SDETL[19:0]. The sound detection flag is cleared to 0b when 1b is written to PDSCRCHn.SDFC.

Figure 37.10 shows the sound detection function.

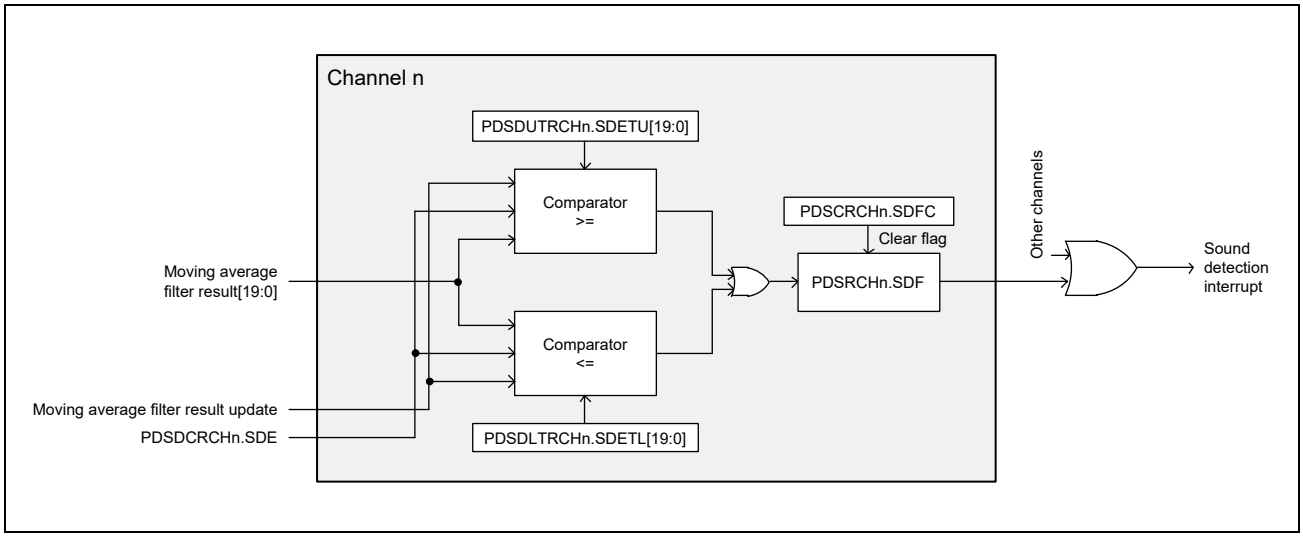


Figure 37.10 Sound Detection Function

37.3.7 Data Buffer

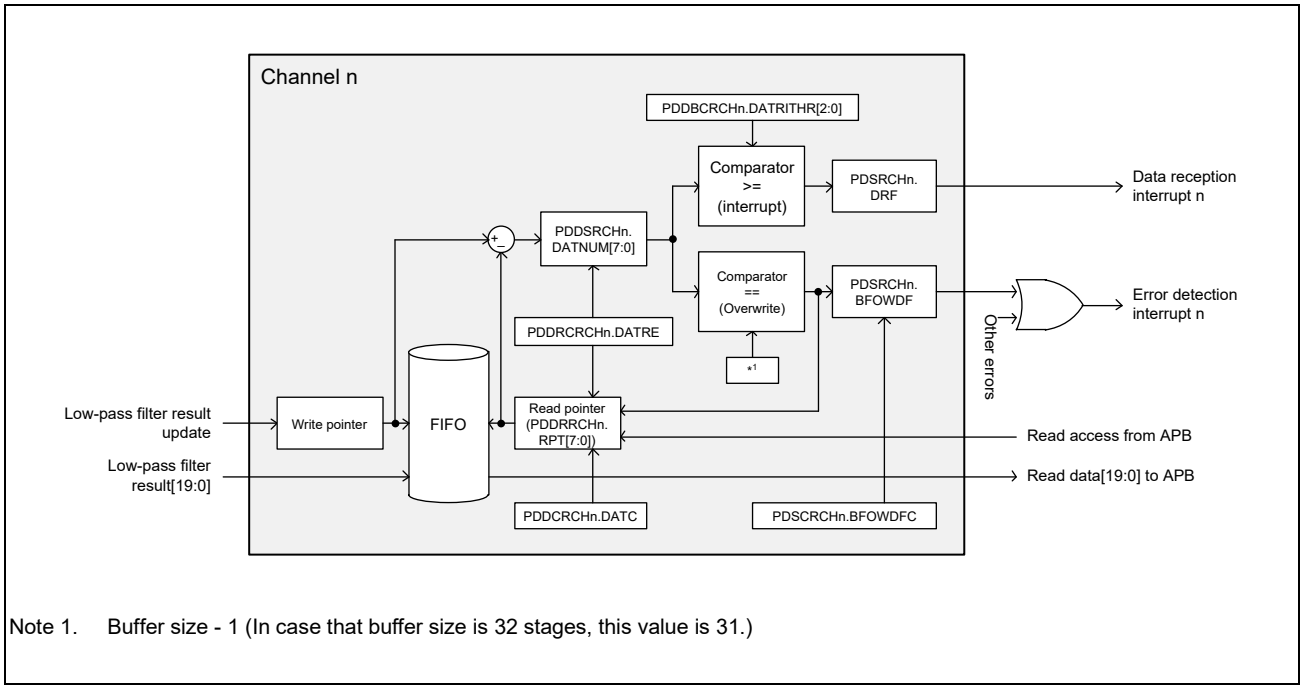
Data buffer writes sound data to FIFO synchronizing with CCLK and reads sound data from FIFO synchronizing with PCLK. Data buffer stores sound data in FIFO until it is read out.

While PDDRCRCHn.DATRE = 0b (hereinafter, DATRE), PDDSRCHnDATNUM[7:0] (hereinafter, DATNUM[7:0]) shows 0 and PDM-IF does not allow to read sound data. After DATRE is changed from 0b to 1b, DATNUM[7:0] shows the buffer size minus 2 and PDM-IF allows to read sound data. While DATRE = 1b, data buffer increments DATNUM[7:0] when data is written and decrements DATNUM[7:0] when data is read.

When DATNUM[7:0] is greater than or equal to a threshold which is configured by PDDBCRCHn.DATRITHR[2:0], a data reception flag PDSRCHn.DRF (hereinafter, DRF) is set to 1b. When DATNUM[7:0] is less than the threshold by reading data or etc., a data reception flag DRF is cleared to 0b.

When DATNUM[7:0] is the buffer size minus 2 and data is written without reading, data buffer overwrites the oldest data with a new one and a buffer overwriting detection flag PDSRCHn.BFOWDF (hereinafter, BFOWDF) is set to 1b. And then DATNUM[7:0] is set to the buffer size minus 2 again after overwriting. However, data buffer overwrites but BFOWDF is not set after DATRE changes from 0b to 1b to before the first data is read.

Figure 37.11 shows the data buffer function.



Note 1. Buffer size - 1 (In case that buffer size is 32 stages, this value is 31.)

Figure 37.11 Data Buffer Function

37.3.8 Short-Circuit Detection

When Short-circuit detection enable (PDSDCRCHn.SCDE) is valid, the short-circuit detection can operate.

A dedicated 13-bit counter counts consecutive 0's or 1's input from pdm_dat which is outputs by data input control block of channel n or channel n+1.

When the number of consecutive 0's exceeds the value set in the PDSCTSRCHn.SCDL[12:0] register or the number of consecutive 1's exceeds the value set in the PDSCTSRCHn.SCDH[12:0] register, a short-circuit detection interrupt request is generated.

Figure 37.12 shows the configuration for the short-circuit detection function.

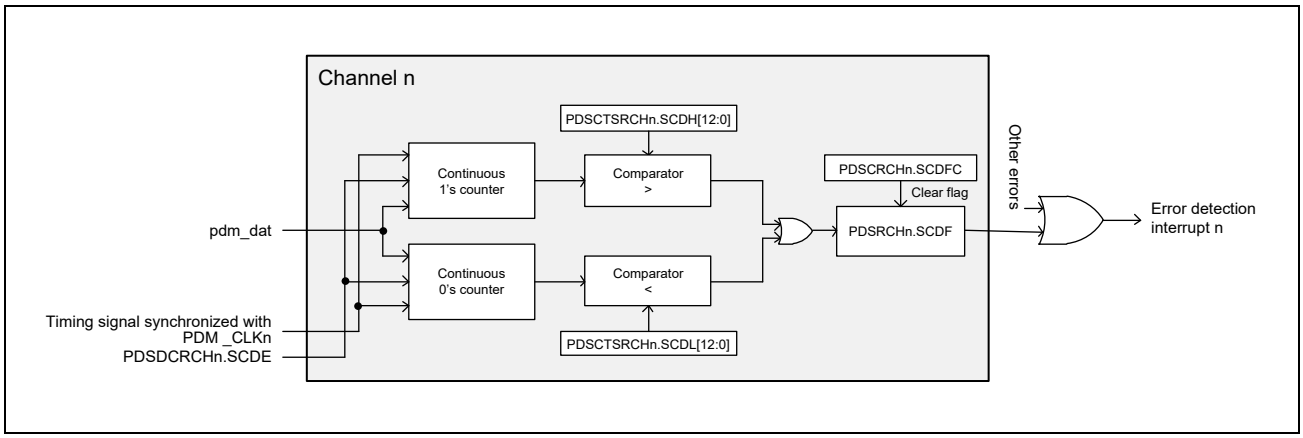


Figure 37.12 Short-Circuit Detection Function

37.3.9 Overvoltage Detection

When the overvoltage upper limit detection enable (PDSDCRCHn.OVUDE) is valid, the overvoltage upper limit detection can operate. When the overvoltage lower limit detection enable (PDSDCRCHn.OVLDE) is valid, the overvoltage lower limit detection can operate.

The sound data is clipped as a 20-bit data from 34-bit data output from sinc filter by the setting of the PDSFCRCHn.SINCRNG[4:0] bits. Refer to **Table 37.8** for details.

An overvoltage upper limit detection flag is set to 1b when the clipped sound data is greater than PDOVUTRCHn.OVDU[19:0].

An overvoltage lower limit detection flag is set to 1b when the clipped sound data is less than PDOVLTRCHn.OVDL[19:0].

Figure 37.13 shows the overvoltage detection function.

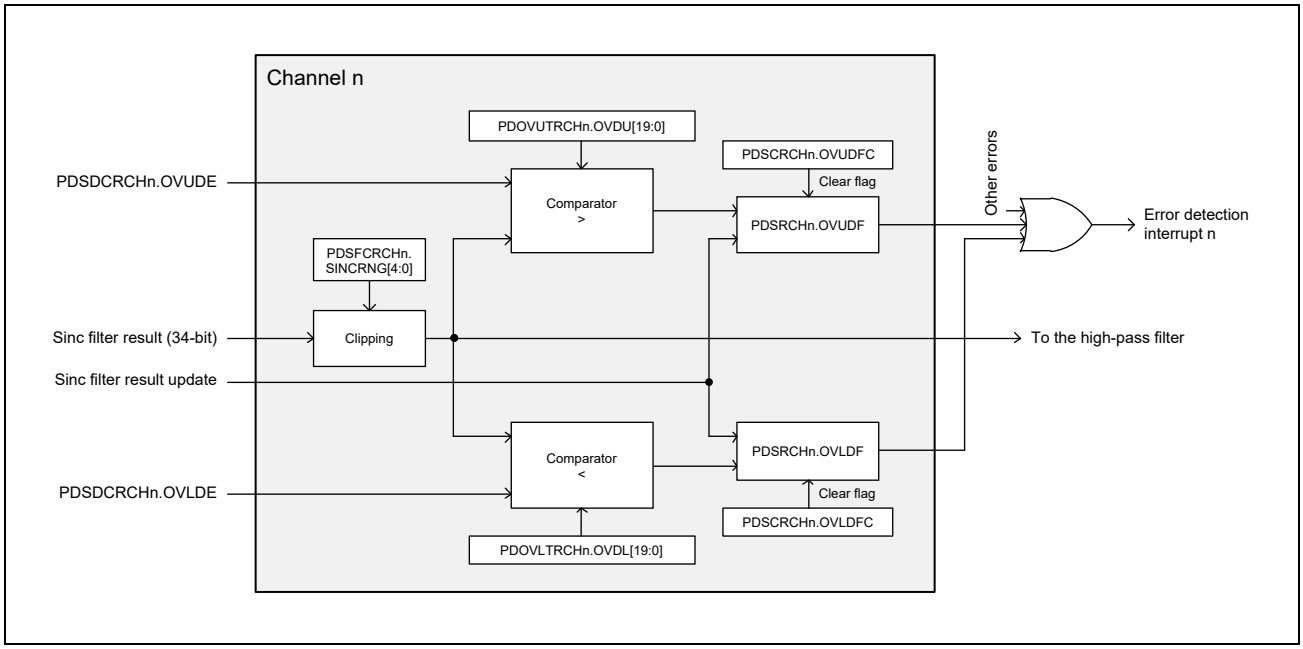


Figure 37.13 Overvoltage Detection Function

37.3.10 Error

Table 37.20 shows a PDM-IF error list.

Table 37.20 PDM-IF Error List

No.	Register Name	Description	Comment	Function (Related Section)
1	PDSRCHn.SCDF	Short circuit detection flag		**Section 37.3.8, Short-Circuit Detection**
2	PDSRCHn.OVLDF	Overvoltage lower limit detection flag		**Section 37.3.9, Overvoltage Detection**
3	PDSRCHn.OVUDF	Overvoltage upper limit detection flag		**Section 37.3.9, Overvoltage Detection**
4	PDSRCHn.BFOWDF	Buffer overwriting detection flag	For debugging	**Section 37.3.7, Data Buffer**

The errors in No.4 are for debugging. Software needs to set registers and operate PDM-IF so that these errors do not occur. If any of these error occur, stop PDM-IF following **Section 37.4.2, Stop Flow** and change the settings.

37.3.11 Interrupt Function

PDM-IF has the following interrupt function.

Table 37.21 PDM-IF Interrupt List

No.	Signal Name	Interrupt Name	Number of Signals	Pulse or Level	Related Register Bit Name		Function (Related Section)
					Common Register	Channel Register	
1	INT_PDM_DATn (n = 0, 1, 2)	Data reception interrupt CHn	3	Level	PDCSR.DRFn	PDSRCHn.DRF	**Section 37.3.7, Data Buffer**
2	INT_PDM_SDET	Sound detection interrupt (3-channel status flags are ORed.)	1	Level	PDCSR.SDFn	PDSRCHn.SDF	**Section 37.3.6, Sound Detection**
3	INT_PDM_ERRn (n = 0, 1, 2)	Error detection interrupt CHn	3	Level	PDCSR.EDFn	PDSRCHn.SCDF	**Section 37.3.8, Short-Circuit Detection**
						PDSRCHn.OVLDF	**Section 37.3.9, Overvoltage Detection**
						PDSRCHn.OVUDF	**Section 37.3.9, Overvoltage Detection**
						PDSRCHn.BFOWDF	**Section 37.3.7, Data Buffer**

37.4 Setting Flow

Figure 37.14 and **Figure 37.15** show the example of the main procedure of PDM-IF.

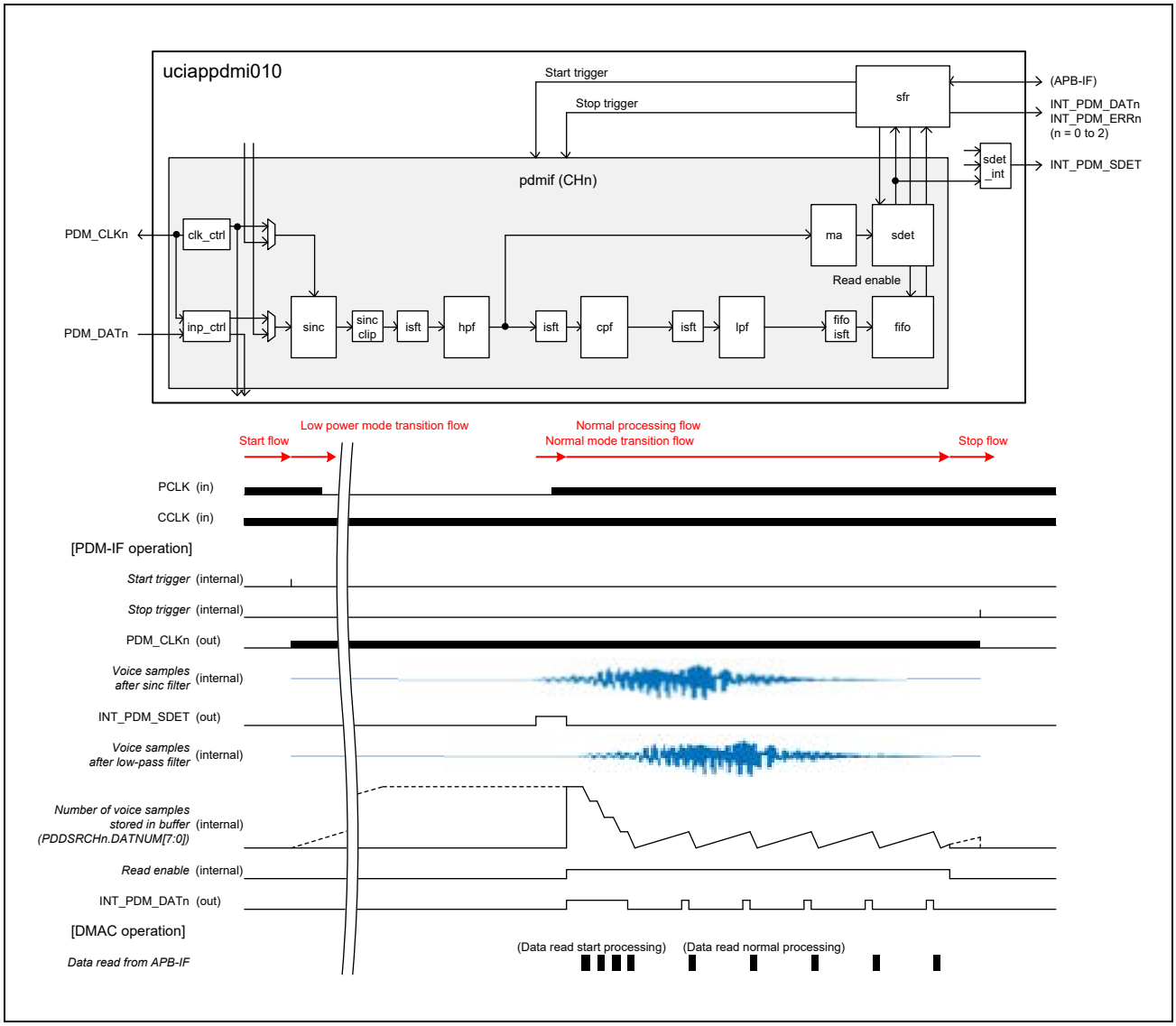


Figure 37.14 PDM-IF Operation Example

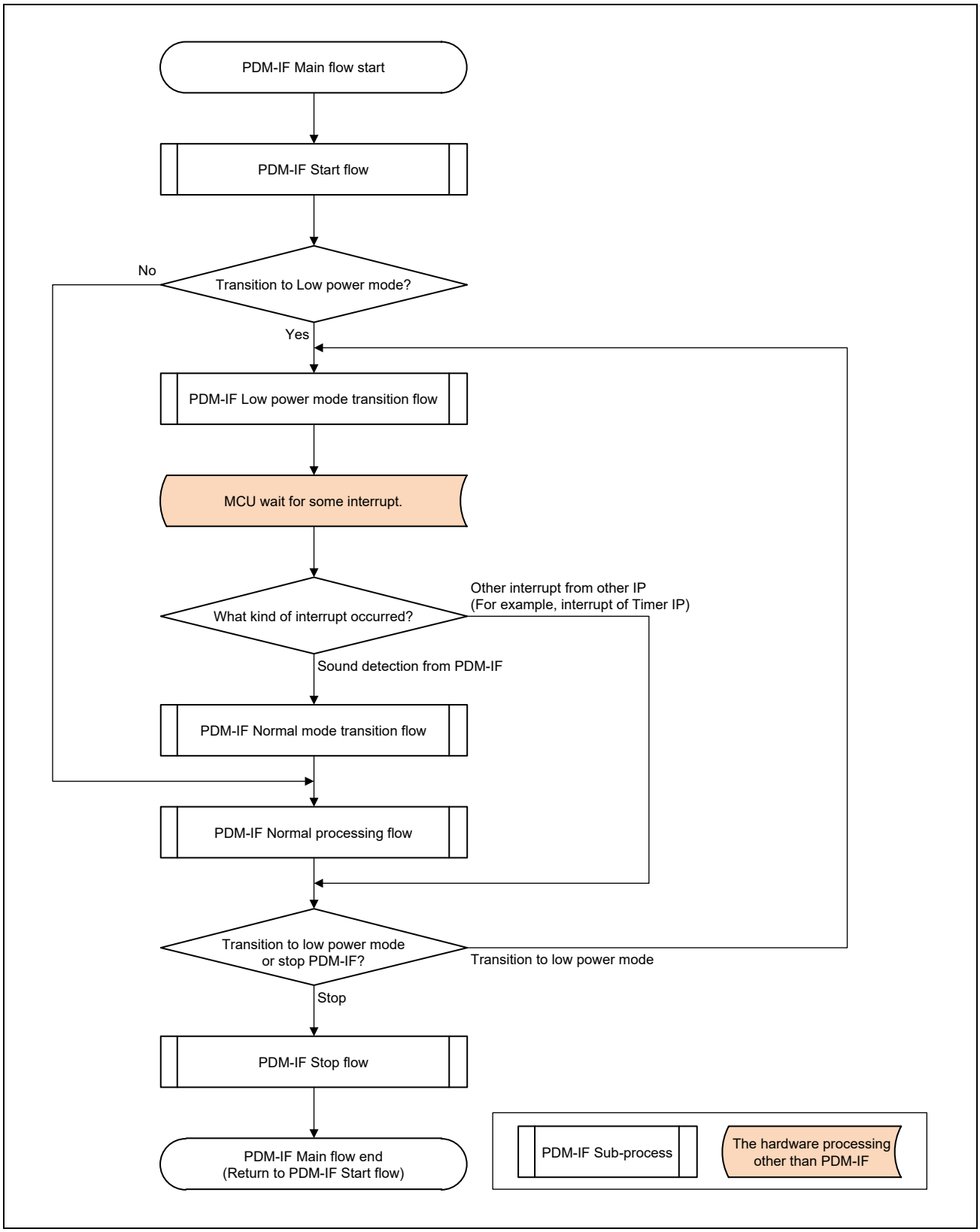


Figure 37.15 PDM-IF Setting Main Flow with Low Power Mode Transition

Figure 37.16 shows the example of the main procedure without low-power mode transition.

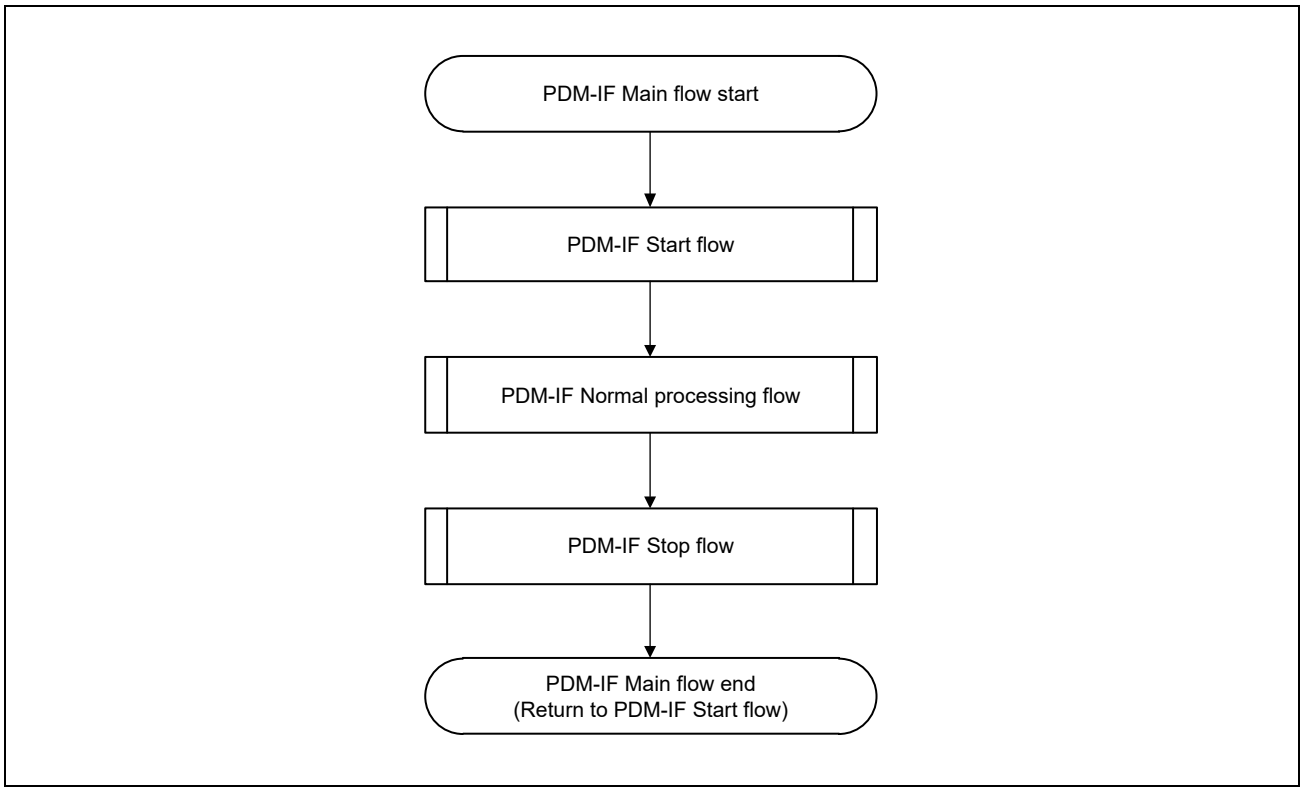
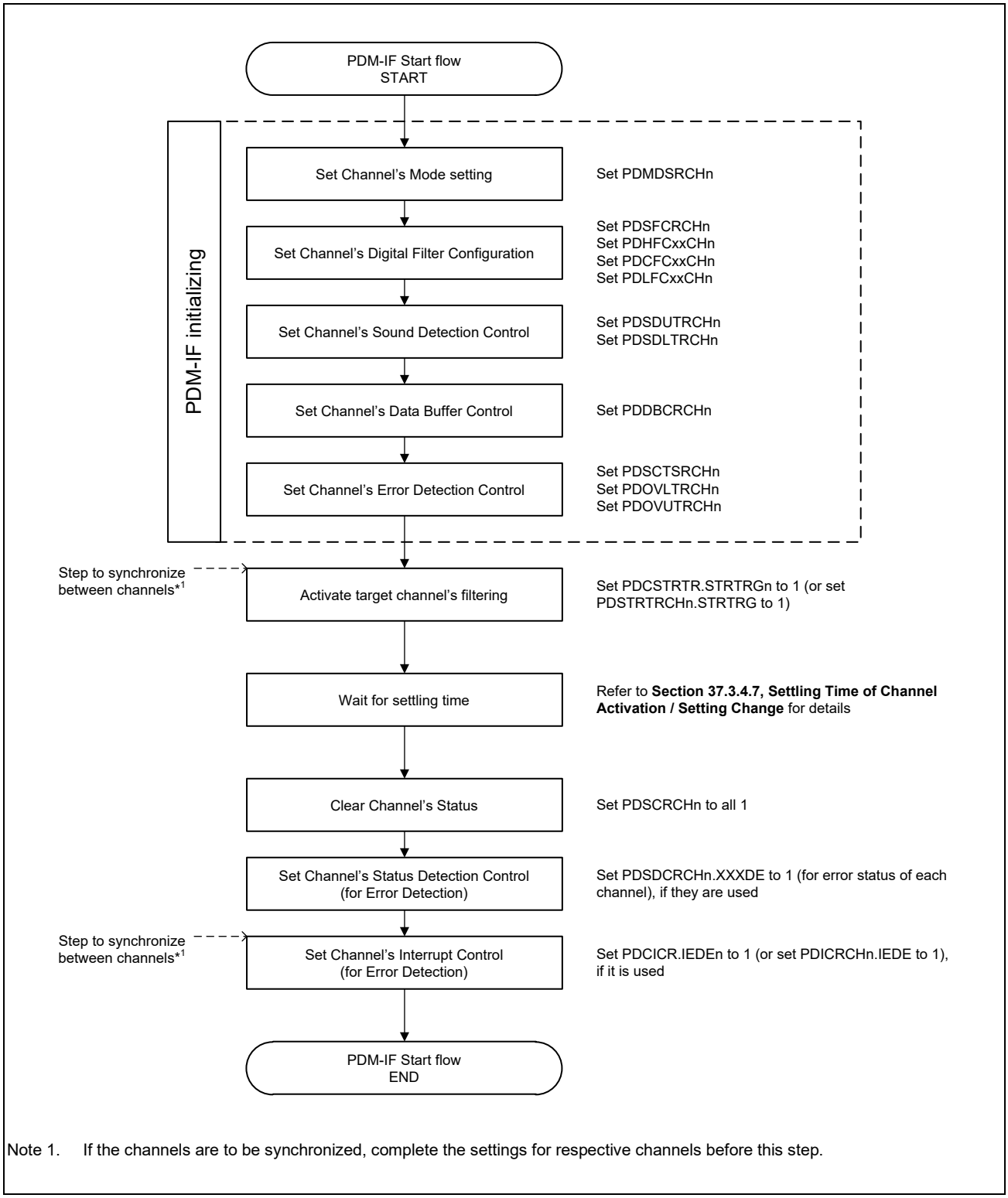


Figure 37.16 PDM-IF Setting Main Flow without Low Power Mode Transition

37.4.1 Start Flow

Figure 37.17 shows the example of the start procedure of PDM-IF.



Note 1. If the channels are to be synchronized, complete the settings for respective channels before this step.

Figure 37.17 PDM-IF Start Flow

37.4.2 Stop Flow

Figure 37.18 shows the example of the stop procedure of PDM-IF.

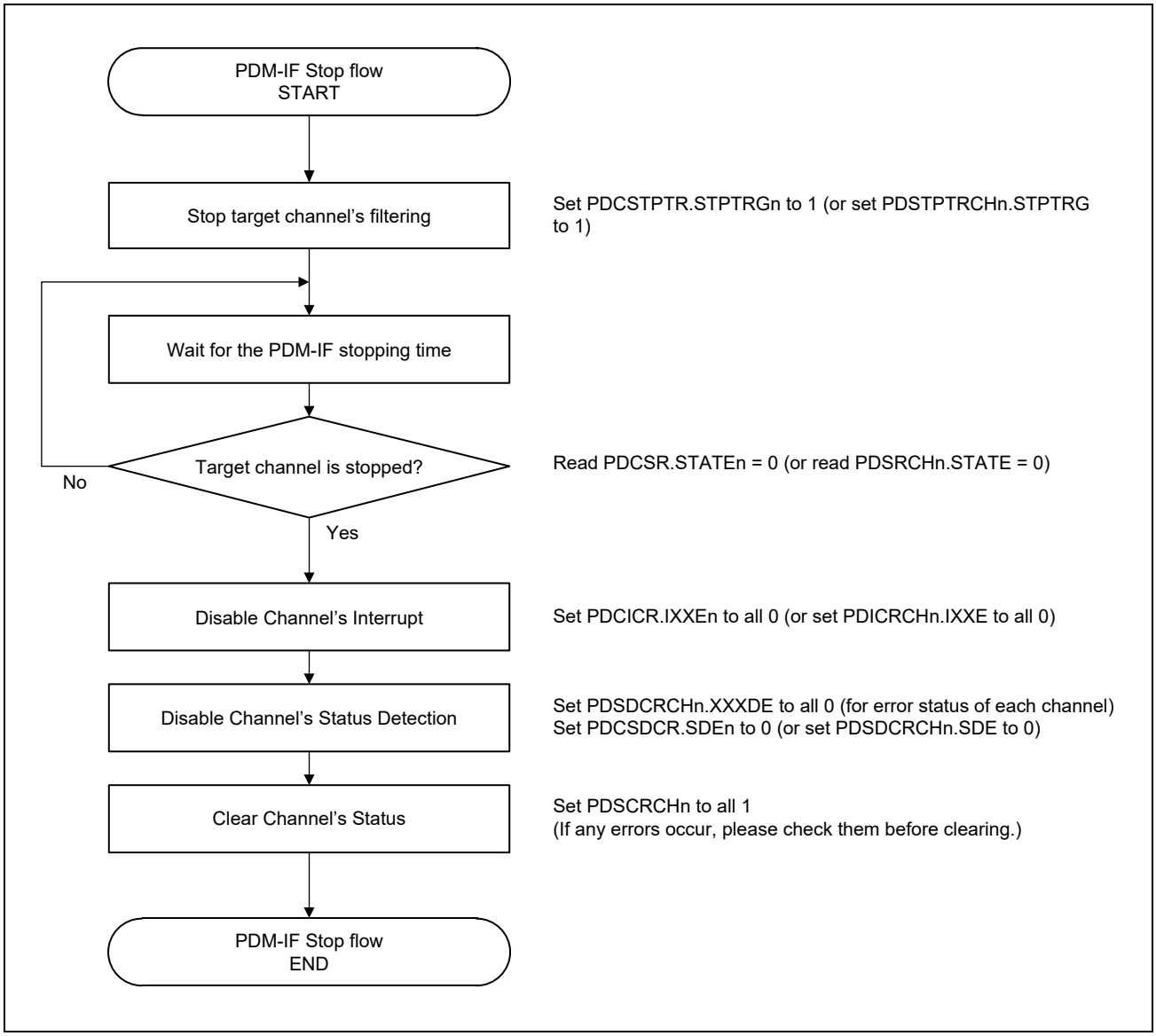


Figure 37.18 PDM-IF Stop Flow

37.4.3 Normal Processing Flow

Figure 37.19 shows the example of the normal processing procedure of PDM-IF.

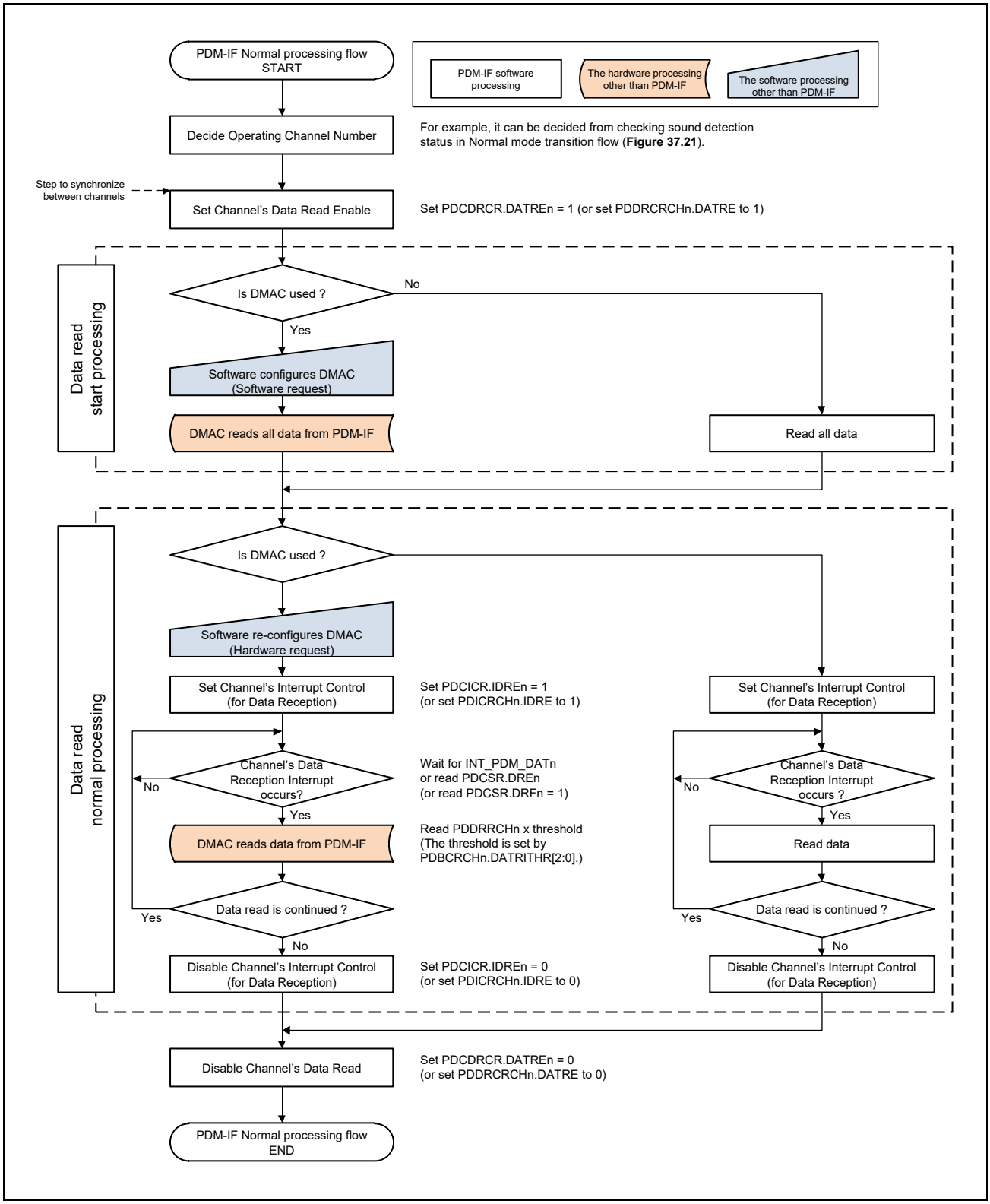
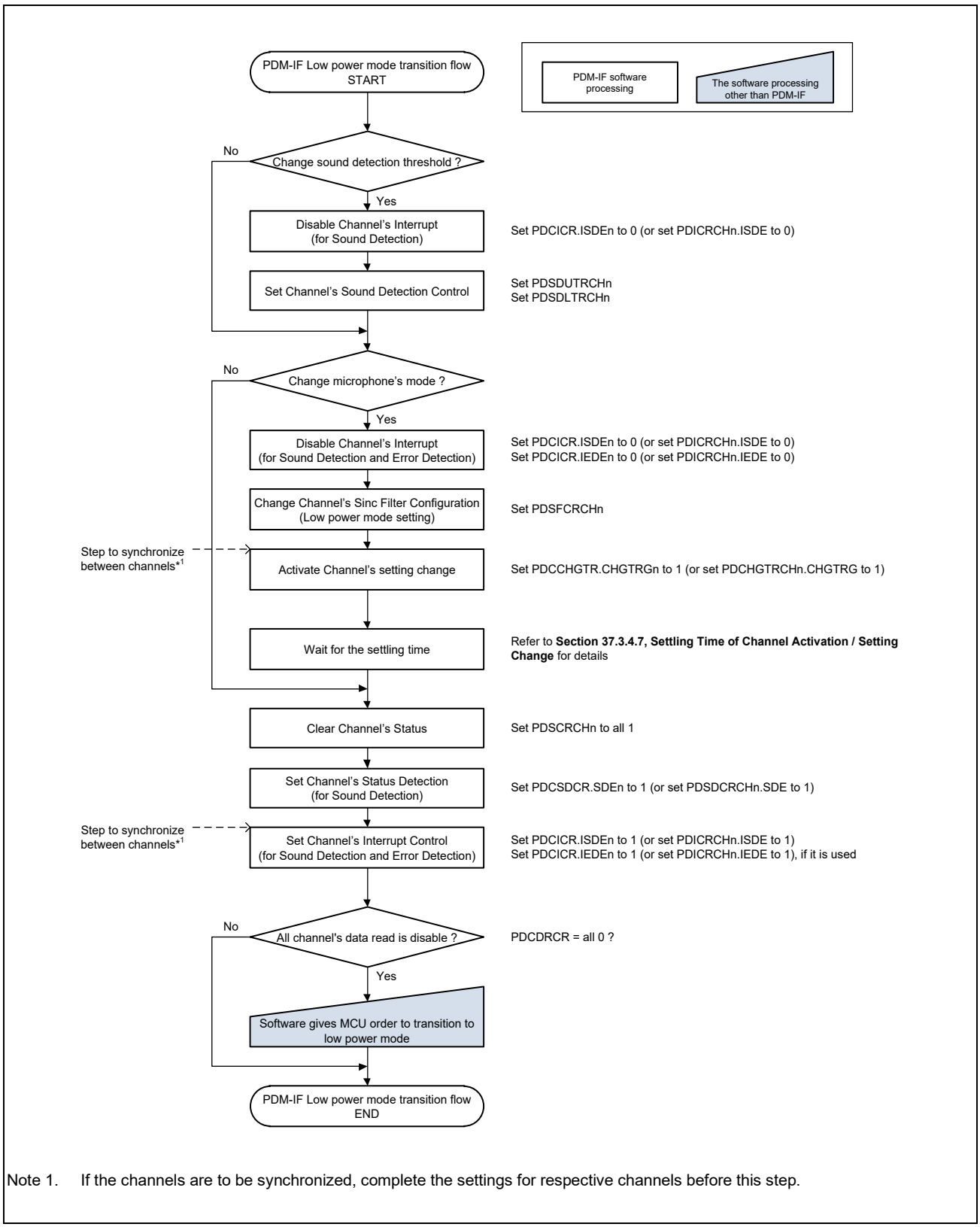


Figure 37.19 PDM-IF Normal Processing Flow

37.4.4 Low Power Mode Transition Flow

Figure 37.20 shows the example of the low power mode transition procedure of PDM-IF.

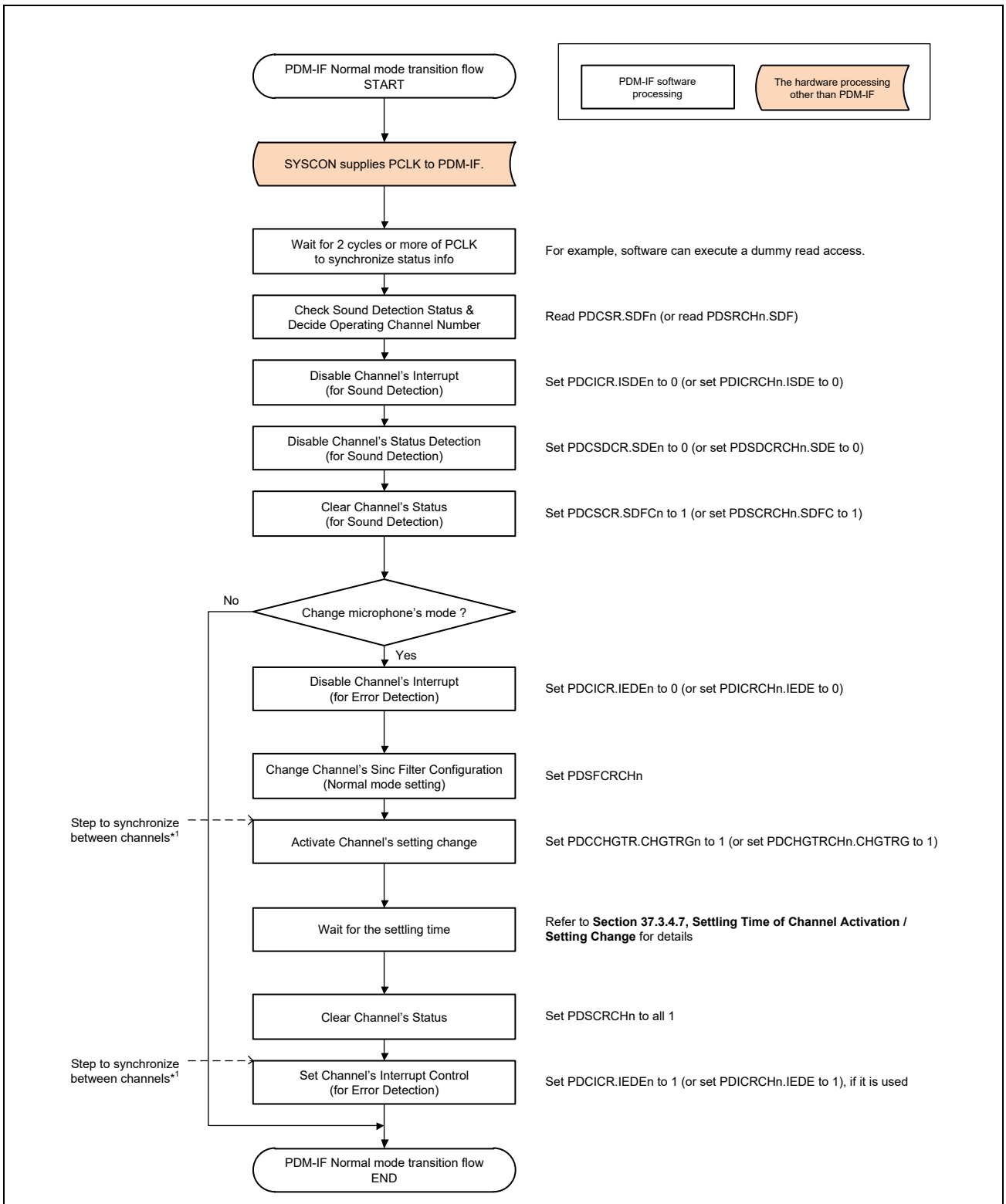


Note 1. If the channels are to be synchronized, complete the settings for respective channels before this step.

Figure 37.20 PDM-IF Low Power Mode Transition Flow

37.4.5 Normal Mode Transition Flow

Figure 37.21 shows the example of normal mode transition procedure of PDM-IF.



Note 1. If the channels are to be synchronized, complete the settings for respective channels before this step.

Figure 37.21 PDM-IF Normal Mode Transition Flow

37.4.6 Setting PDM_CLKn (n = 0, 1, 2) Frequency and Sampling Frequency Generated by the Unit

The frequency of PDM_CLKn (n = 0, 1, 2), which is the reference for receiving data from the external device, and the sampling frequency generated by the PDM unit for external input PDM data can be changed by setting the register. The PDM_CLKn (n = 0, 1, 2) frequency must match the clock input specifications of the external device. The sampling frequency generated by the unit should be set to the expected value.

In this LSI, the PDM reference clock CCLK is fixed at 8 MHz, so the PDM_CLK output frequency and the unit-generation sampling frequency follow the equation below when implementing the circuit.

- PDM_CLKn (n = 0, 1, 2) frequency

$$= \frac{\text{CCLK frequency}}{(\text{CKDIV} + 1) \times 2};$$

- Unit-generation sampling frequency

$$= \frac{\text{PDM\_CLK frequency}}{(\text{SINCDEC} + 1) \times 2^{*1}} = \frac{\text{CCLK frequency}}{(\text{CKDIV} + 1) \times 2} \div (\text{SINCDEC} + 1) \times 2^{*1};$$

Remark: CKDIV: PDSFCRCHn.CKDIV[3:0] register set value (Set the PDM_CLKn division ratio to CCLK)
SINCDEC: PDSFCRCHn.SINCDEC[7:0] register set value (Set the decimation ratio of the Sinc filter)

Note 1. Reason for ×2: Low-pass filter decimation ratio

- PDMIF sampling rate:

$$\frac{\text{CCLK frequency}}{\big((\text{CKDIV} + 1) \times 2\big) \times (\text{SINCDEC} + 1) \times 2}$$

The output range of the sinc filter is changed by changing the decimation ratio of the sinc filter (SINCDEC) with the change of the sampling frequency above. Since the maximum number of effective bits for processing of audio data within the PDM following sinc filtering is 20 bits, PDSFCRCHn.SINCRNG[4:0] must also be changed according to the output range of the sinc filter. This allows avoiding the occurrence of overflow and underflow and maximizing the effective audio amplitude.

Recommended setting: Set PDSFCRCHn.SINCRNG[4:0] such that the higher-order 20 bits of valid data of the sinc filter output are effective in calculation processing following sinc filtering.

For details on clipping or sifting of the sinc filter output, see **Section 37.3.4.1, Data Clipping and Shifting between Filters**.

- Output range of the sinc filter
  = ±decimation ratio (PDSFCRCHn.SINCDEC[7:0]) ^ order (PDMDSRCHn.SFMD[2:0]);

[Example]

When SINCDEC[7:0] = H'27 (decimation ratio: 40) and SFMD[2:0] = 0 (4-order), the output range of the sinc filter is ±2560000 (23 bits consisting of 22 bits of significant digits and a sign bit). The recommended setting of PDSFCRCHn.SINCRNG[4:0] in this case is 0_1011b (b22-b3).

37.4.7 DMA Transfer Method

This PDM unit can only receive voice data from external sources, and does not have a transmit function. The PDM unit does not have a dedicated REQ/ACK signal to the DMAC master, but only reads data from the PDM buffer read register via the APB I/F. Reading data via the CPU and reading data via the DMAC master are not different in terms of PDM functionality.

Figure 37.22 shows an overview of the PDM data read process.

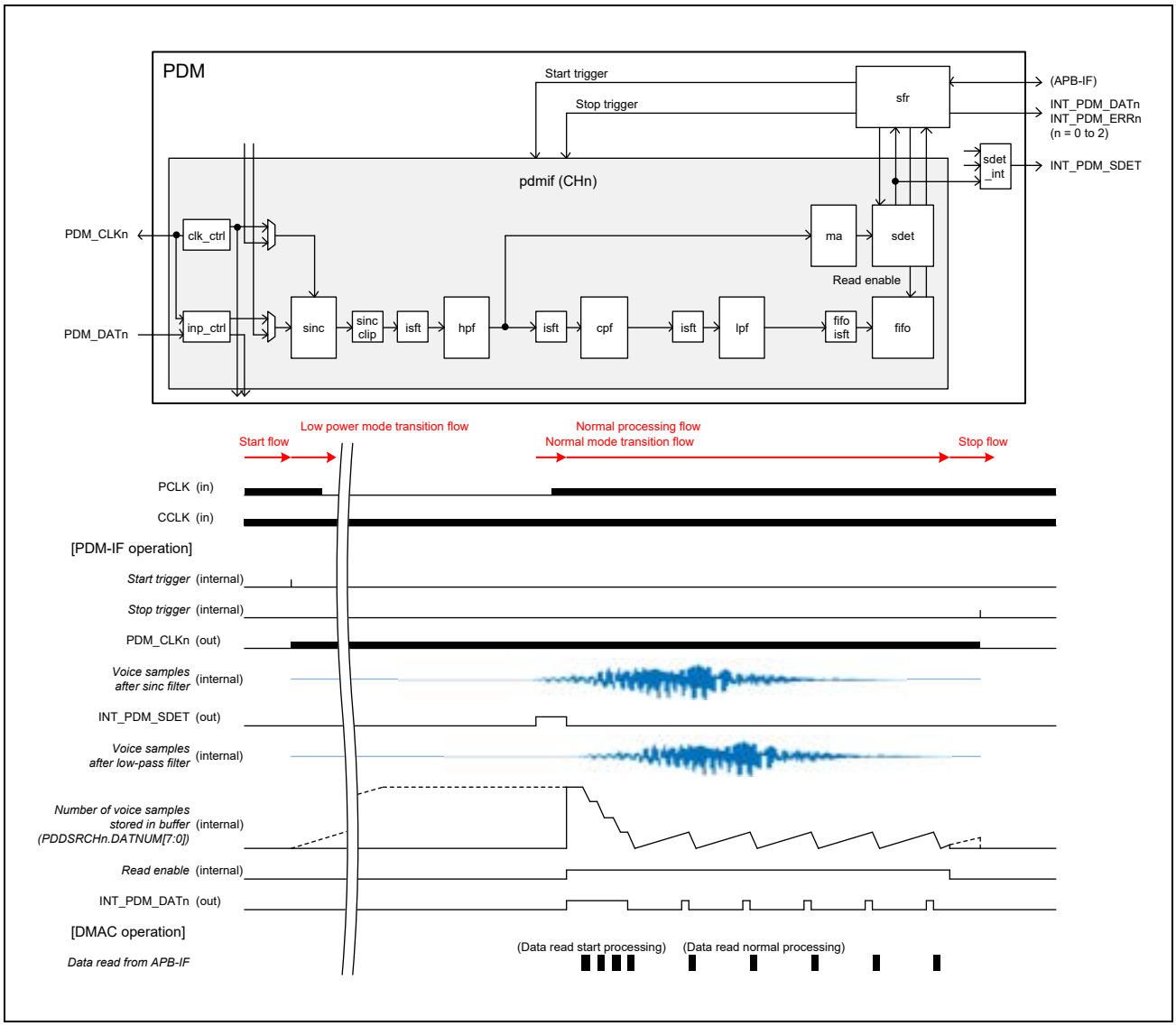


Figure 37.22 PDM-IF Operation Example

The DMA data transfer using the DMA master must follow the description in **Section 37.4.3, Normal Processing Flow**. The following explains the connection configuration with the DMA master, initialization settings on the DMA master, and recommended settings for the INT_PDM_DAT0/1/2 signal at PDM initialization.

The connection configuration with the DMA master in this LSI is as follows, issuing a DMA request to the DMA master based on the INT_PDM_DAT0/1/2 signal of the PDM. The INT_PDM_DAT0/1/2 interrupt signal is active (high) depending on the amount of data stored in the receive buffer in the PDM (the amount of interrupt generation buffer can be changed by setting the register / PDDBCRCHn (n = 0, 1, 2).DATRITHR[2:0]).

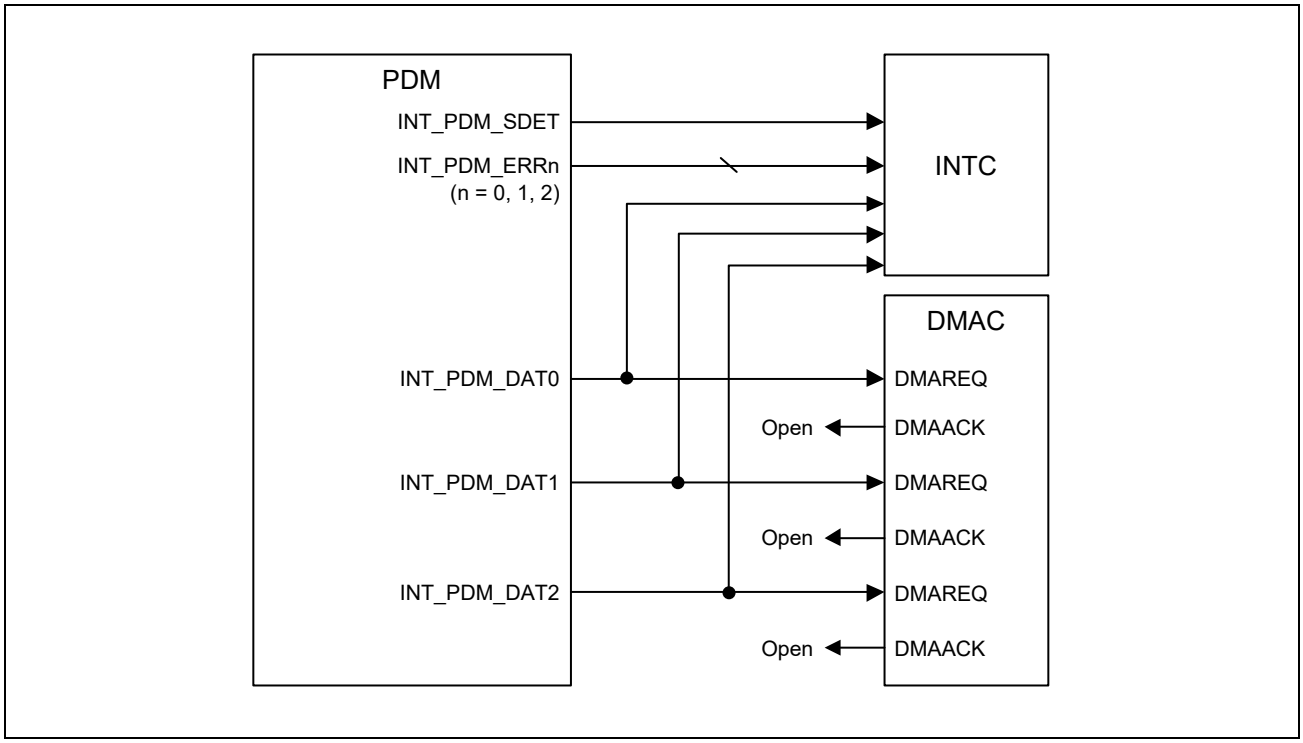


Figure 37.4-23 PDM Connection Configuration

This section describes the restrictions at DMA transfer and the active condition of the INT_PDM_DAT0/1/2 signal at PDM initialization.

1) Restrictions at DMA transfer:

The initialization setting on the DMA master is required in “Software configures DMAC/Software re-configures DMAC” in **Figure 37.19** of **Section 37.4.3, Normal Processing Flow**. The restrictions on PDM DMA transfers are described below.

- DMA transfer mode (1): Single

  (One DMA transfer process is performed for one DMA request.)
  Reference value for the DMA master setting: TM register = 0b

- DMA transfer mode (2): Bus cycle mode

  (DMAREQ is masked in the bus cycle period (while DMAACK is active.))
  Reference value for the DMA master setting: AM[2:0] = H’010

- One DMA transfer: 32 bits

  Reference value for the DMA master: DDS or SDS register = 32 bits

- DMA REQ signal attribute: Level/High

  (The active polarity of the INT_PDM_DAT0/1/2 signal is high, so the DMA request signal must also be high.)
  Reference value for the DMA master: LVL = 1b, HIEN = 1b, LOEN = 0b

- DMA REQ signal active release timing: After the completion of data read from the PDM unit

  (DMA REQ is masked while DMA ACK is enabled.)
  Reference value for the DMA master: REQD = 0b

- DMA transfer data read register: PDDRRCHn (n = 0, 1, 2).DAT[19:0] register

2) Interrupt setting of the INT_PDM_DAT0/1/2 signal in the PDM receive initialization setting:

The interrupt setting for the INT_PDM_DAT0/1/2 signal is required in “Set Channel’s Data Buffer Control” in **Figure 37.17** of **Section 37.4.1, Start Flow**.
As the receive buffer amount of PDM in this LSI is fixed at 64 stages, it is recommended to set PDDBCRCHn (n = 0, 1, 2).DATRITHR[2:0] = 4 (16 stages). Since a time difference may occur between the active timing of the INT_PDM_DAT0/1/2 signal and the request decision on the DMA master, the active timing of the INT_PDM_DAT0/1/2 signal must always be selected to be at least 2 stages.

37.4.8 Reset Clock Control

For the reset clock control, the following reset clock pins must be controlled.

Table 37.22 List of Reset Clocks

	Reset Signal	Clock Signal	Clock Source	Description
System 1	PRESETn	PCLK, PCLK_SFR	CPG unit (Divider clock for CPG PLL output)	For internal bus control circuit
System 2	CRESETn	CCLK	CPG unit (External system clock: 8 MHz fixed)	For external IF control circuit

The above reset clock pins are controlled by the CPG unit, and each reset clock can be controlled ON/OFF with the CPG register. When each reset is released, a feature is implemented on the CPG to stop the target clock before and after the reset is released.

For each system, reset and clock control should be performed according to the following procedure.

However, the reset and clock control between the two systems below are independent, and there are no restrictions on the order of settings between the systems.

Furthermore, when controlling the reset clock, the PDM unit should be stopped and accesses from related units should also be stopped.

1. Reset (PRESETn) and clock (PCLK, PCLK_SFR) for the internal bus control circuit:
   — PDM operation start condition (reset/clock supply): clock supply → reset release
   — PDM stop condition (reset/clock stop): reset enabled → clock stop
   — PDM reset (reset ON/OFF, clock supply continued): Reset can be enabled or disabled at any timing

2. Reset (CRESETn) and clock (CCLK) for the external IF control circuit:
   — PDM operation start condition (reset/clock supply): clock supply → reset release
   — PDM stop condition (reset/clock stop): reset enabled → clock stop
   — PDM reset (reset ON/OFF, clock supply continued): Reset can be enabled or disabled at any timing

As for PCLK/PCLK_SFR, the CPG unit can change the clock frequency variable (100 MHz, 50 MHz, 25 MHz, 12.5 MHz, 3.125 MHz). When lowering the PCLK/PCLK_SFR frequency, the performance of reading voice data from the PDM unit will be degraded. Use caution when lowering the frequency.

37.5 Interrupt

37.5.1 Interrupt List

Table 37.23 is a list of interrupt signals for this unit. For details on interrupts, refer to **Section 8, Interrupt Controller**.

Table 37.23 PDM-IF Interrupt List

No.	Signal Name	Interrupt Name	Number of Signals	Pulse or Level	Related Register Bit Name		Function (Related Section)
					Common Register	Channel Register	
1	INT_PDM_DATn (n = 0, 1, 2)	Data reception interrupt CHn (n = 0, 1, 2)	3	Level	PDCSR.DRFn	PDSRCHn (n = 0, 1, 2).DRF	**Section 37.3.7, Data Buffer**
2	INT_PDM_SDET	Sound detection interrupt (3-channel status flags are ORed.)	1	Level	PDCSR.SDFn	PDSRCHn (n = 0, 1, 2).SDF	**Section 37.3.6, Sound Detection**
3	INT_PDM_ERRn (n = 0, 1, 2)	Error detection interrupt CHn (n = 0, 1, 2)	3	Level	PDCSR.EDFn	PDSRCHn (n = 0, 1, 2).SCDF	**Section 37.3.8, Short-Circuit Detection**
						PDSRCHn (n = 0, 1 ,2).OVLDF	**Section 37.3.9, Overvoltage Detection**
						PDSRCHn (n = 0, 1, 2).OVUDF	**Section 37.3.9, Overvoltage Detection**
						PDSRCHn (n = 0, 1, 2) BFOWDF	**Section 37.3.7, Data Buffer**

37.6 Usage Notes

37.6.1 Data Reception Interrupt Threshold Register Setting

A data reception interrupt INT_PDM_DATn is issued as a level output. Therefore, if the number of samples in buffer still exceeds the data reception interrupt threshold set in PDDBCRCHn.DATRITHR[2:0] when data reading of DMA finishes, the interrupt controller might not detect the edge of INT_PDM_DATn. The situation happens if data reading time exceeds sampling rate*1 x threshold. Set the appropriate threshold so that the interrupt controller can detect the interrupt.

In case the threshold cannot avoid the situation, read PDDBCRCHn.DATRITHR[2:0] and re-start DMA by software after reading the data.

Note 1. Sampling rate: Time to spend to store a sample in buffer.

37.6.2 Frequency Characteristics of Filters with Default Settings

The following figure and table show the frequency characteristics of the filters with the default settings for reference information.

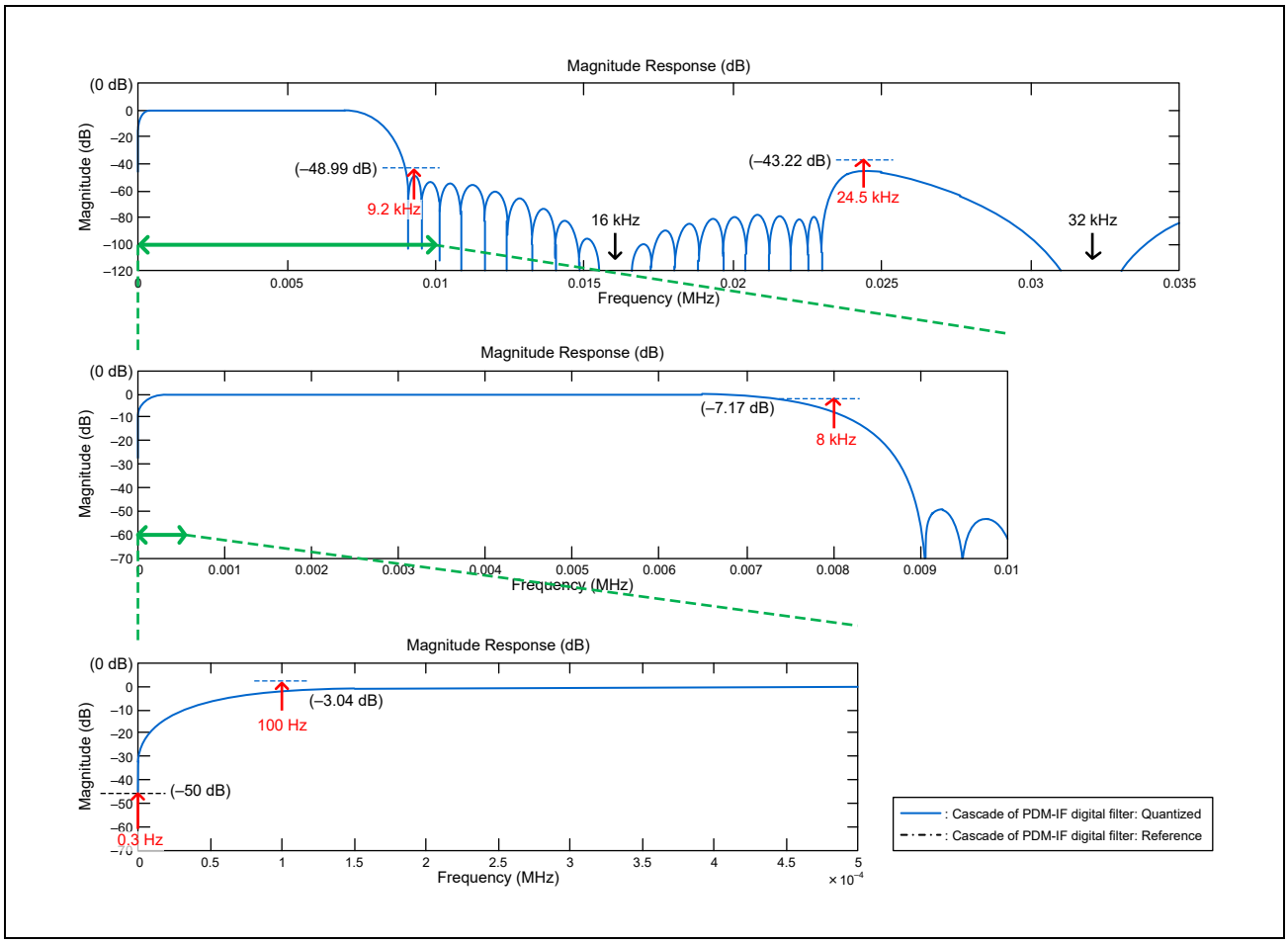


Figure 37.24 Frequency Characteristics of Filters with Default Settings

Table 37.24 Frequency Characteristics of Filters with Default Settings

Frequency (Hz)		Magnitude (dB)
Cut-off frequency of LPF	24.5 kHz	−43.22
	…	…
	9.2 kHz	−48.99
	8 kHz	−7.17
Passband	7 kHz	−0.39
	6 kHz	−0.07
	…	…
	150 Hz	−1.63
	100 Hz	−3.04
Cut-off frequency of HPF	50 Hz	−7.02
	0.3 Hz	−50.00

38. Renesas SPDIF Interface

38.1 Overview

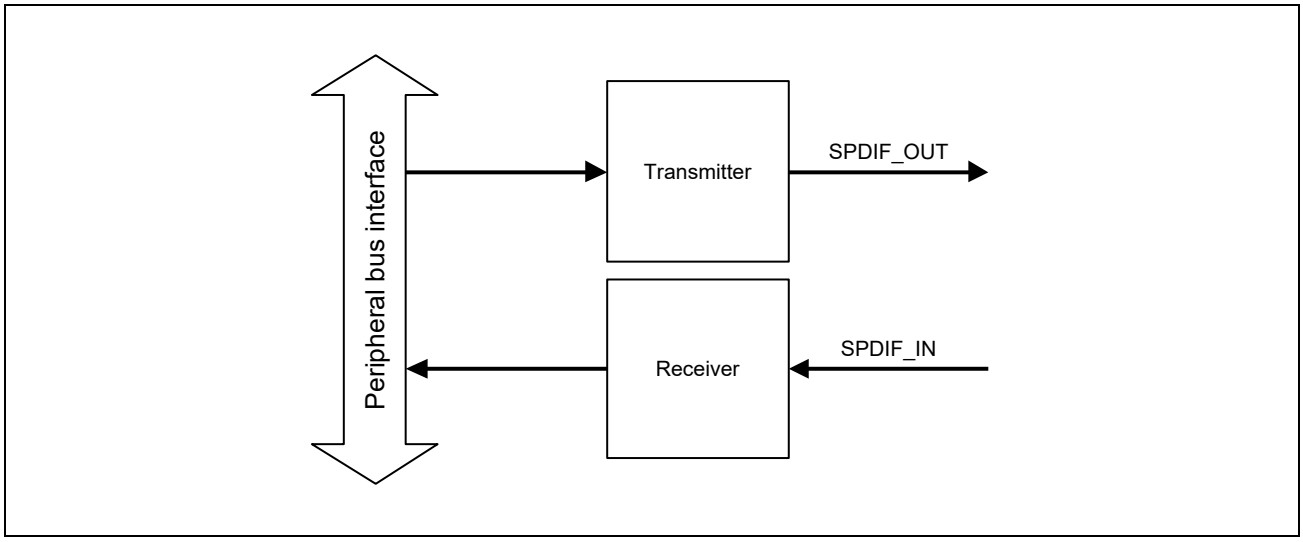


Figure 38.1 Overview Block Diagram

38.2 Features

- Supports the IEC 60958 standard (stereo and consumer use modes only).
- Supports sampling frequencies of 32 kHz, 44.1 kHz, and 48 kHz.
- Supports audio word sizes of 16 to 24 bits per sample.
- Biphase mark encoding.
- Double buffered data.
- Parity encoded serial data.
- Simultaneous transmit and receive
- Receiver autodetects IEC 61937 compressed mode data

38.3 Functional Block Diagram

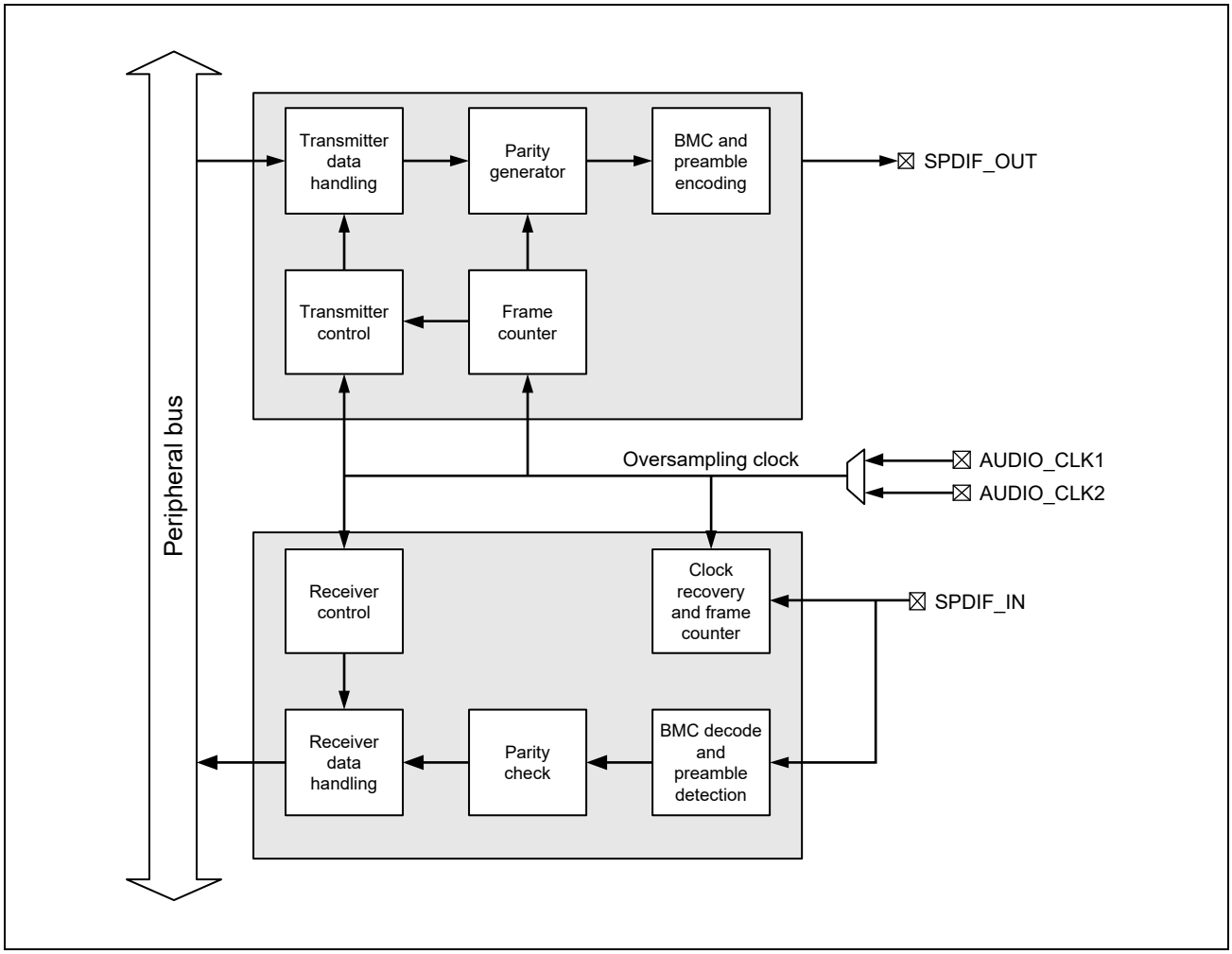


Figure 38.2 Functional Block Diagram

38.4 Input/Output Pins

Table 38.1 shows the pin configuration.

Table 38.1 Pin Configuration

Channel	Pin Name	I/O	Description
0	SPDIF_OUT	Output	Transmitter biphase-mark encoded SPDIF bitstream
1	SPDIF_IN	Input	Receiver biphase-mark encoded SPDIF bitstream
0, 1 (Common)	AUDIO_CLK1	Input	External clock for audio 1
	AUDIO_CLK2	Input	External clock for audio 2

38.5 Renesas SPDIF (IEC60958) Frame Format

The Renesas SPDIF frame consists of two subframes (for channels 1 and 2), each of which contains a 4-bit preamble, audio data of up to 24 bits, a V flag, a user bit, a channel status bit, and an even parity bit. **Figure 38.3** shows the subframe format. According to this format, the Renesas SPDIF performs biphase-mark modulation (channel coding) that will make the transmission line's DC component a minimum value.

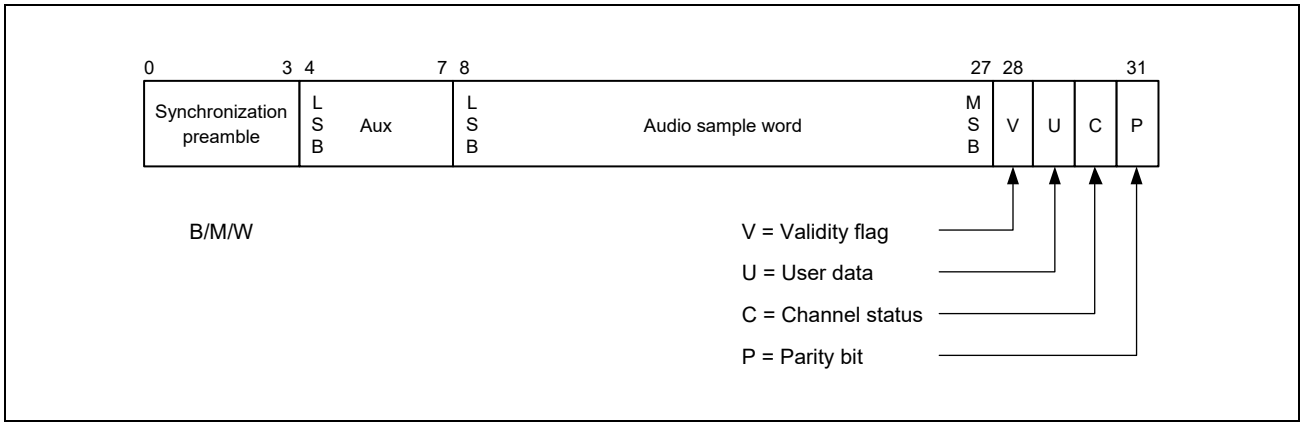


Figure 38.3 Subframe Format

Figure 38.4 shows the block format, which consists of 192 continuous frames. One block begins at the starting frame (preamble B) and ends at the 192nd frame (frame 191), and the preamble is used to identify all subframes. Each block has a total of 384 subframes, which are classified into three categories: subframe 0 indicating the beginning of a new block, subframe 1 (usually the channel 1), and subframe 2 (usually the channel 2). Usually, the music data sent and received by the SPDIF is continuous so that continuous blocks appear.

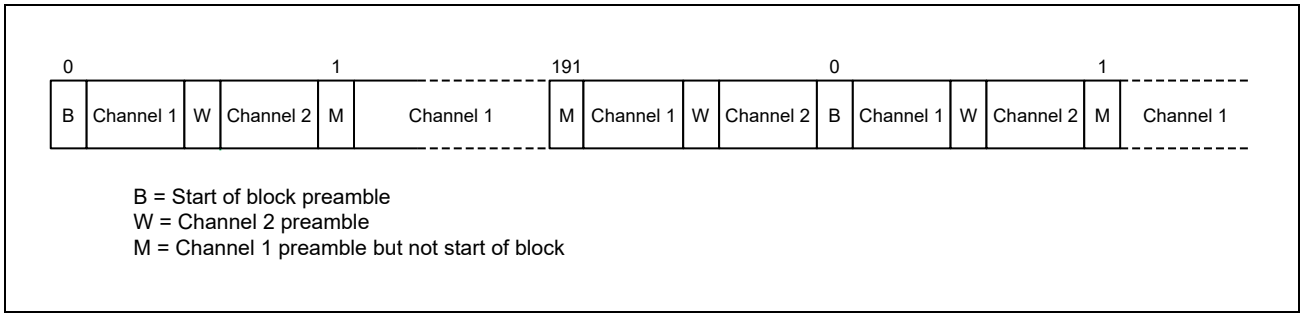


Figure 38.4 Block Format

Table 38.2 shows the binary values of the Renesas SPDIF preambles. The polarity of these preambles differs depending on the status of the preceding symbol (parity bit).

Table 38.2 Binary Preamble Values

Preamble	Preceding Symbol's Status = 0	Preceding Symbol's Status = 1
B	H'1110_1000	H'0001_0111
M	H'1110_0010	H'0001_1101
W	H'1110_0100	H'0001_1011

Note: As shown in **Figure 38.3**, the even parity bit at time slot 31 of a subframe determines the type of a preamble for one cycle of transmission. Usually, therefore, any one is selected from the set states that are sent through the Renesas SPDIF. However, IEC60958 requires decoding both types in view of connection with the preamble polarity reversed; the Renesas SPDIF has preambles decoded according to **Table 38.2**.

Channel status information is encoded at the rate of one bit per subframe, making the channel status information per block have a total of 192 bits for each of subframes 1 and 2. For the format of the channel status, refer to the IEC 60958 standard.

38.6 Register

Table 38.3 shows the register configuration. The SPDIF address space is offset from the base address. The base address of SPDIF is as follows.

SPDIF base address: H'0_100A_9000 (Overall Address Space)
SPDIF base address: H'400A_9000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
SPDIF base address: H'500A_9000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 38.3 Register Configuration

Channel	Register Name	Abbreviation	Offset Address	Access size
0 (Transmit)	Transmitter channel 1 audio register	TLCA	H'0000	32
	Transmitter channel 2 audio register	TRCA	H'0004	32
	Transmitter channel 1 status register	TLCS	H'0008	32
	Transmitter channel 2 status register	TRCS	H'000C	32
	Transmitter user data register	TUI	H'0010	32
1 (Receive)	Receiver channel 1 audio register	RLCA	H'0014	32
	Receiver channel 2 audio register	RRCA	H'0018	32
	Receiver channel 1 status register	RLCS	H'001C	32
	Receiver channel 2 status register	RRCS	H'0020	32
	Receiver user data register	RUI	H'0024	32
0, 1 (Common)	Control register	CTRL	H'0028	32
	Status register	STAT	H'002C	32
0, 1 (Common)	Transmitter DMA audio data register	TDAD	H'0030	32
	Receiver DMA audio data register	RDAD	H'0034	32

Note: All registers are longword registers and must be accessed as such.
A register diagram containing a 0 indicates that the write value should always be 0 (if the register is writable) and that the read value should always be 0 (if readable).

38.7 Register Descriptions

[Legend]

Initial Value: Register value after reset

—: Undefined value

R/W: Readable/writable register. The write value can be read.

R: Read only register. The write value should always be 0.

R/WC0: Readable/writable register. Writing 0 initializes the bit, but writing 1 is ignored.

R/WC1: Readable/writable register. Writing 1 initializes the bit, but writing 0 is ignored.

W: Write only register. Reading is prohibited. If this bit is reserved, the write value should always be 0.

—/W: Write only, Read value undefined

38.7.1 Control Register (CTRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	CKS	—	PB	RASS		TASS		RDE	TDE	NCSI	AOS	RME	TME
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	REIE	TEIE	UBOI	UBUI	CREI	PAEI	PREI	CSEI	ABOI	ABUI	RUII	TUII	RCSI	RCBI	TCSI	TCBI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	CKS	0	R/W	Oversampling clock select Selects oversampling clock supply source. 0: AUDIO_CLK1 1: AUDIO_CLK2
27	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	PB	0	R/W	Pass Back Passes transmitter SPDIF output into SPDIF receiver in SPDIF module. 0: Pass Back disabled 1: Pass Back enabled
25, 24	RASS	All 0	R/W	Receiver Audio Sample Bit Size These bits Indicate the receiver audio sample bit size (16, 20, or 24 bits), for data alignment purposes. 00: 16-bit sample 01: 20-bit sample 10: 24-bit sample 11: Reserved
23, 22	TASS	All 0	R/W	Transmitter Audio Sample Bit Size These bits Indicate the transmitter audio sample bit size (16, 20, or 24 bits), for data alignment purposes. 00: 16-bit sample 01: 20-bit sample 10: 24-bit sample 11: Reserved
21	RDE	0	R/W	Receiver DMA Enable Enables DMA requests for the receiver. 0: Receiver DMA disabled 1: Receiver DMA enabled
20	TDE	0	R/W	Transmitter DMA Enable Enables the DMA requests for the transmitter. 0: Transmitter DMA disabled 1: Transmitter DMA enabled
19	NCSI	0	R/W	New Channel Status Information Set this bit to 1 when new channel status information to be corrected is in the transmitter. 0: New channel status information has not been in transmitter 1: New channel status information has been in transmitter

Bit	Bit Name	Initial Value	R/W	Description
18	AOS	0	R/W	Audio Only Samples Clear this bit to 0 when audio channel 1 and channel 2 registers contain user information. When this bit is set to 1, all user bits are cleared to 0. 0: User information present 1: User information not present
17	RME	0	R/W	Receiver Module Enable Enables the receiver module. 0: Receiver module disabled 1: Receiver module enabled
16	TME	0	R/W	Transmitter Module Enable Enables the transmitter module. 0: Transmitter module disabled 1: Transmitter module enabled
15	REIE	0	R/W	Receiver Error Interrupt Enable Enables the receiver error interrupts. 0: Receiver error interrupt disabled 1: Receiver error interrupt enabled
14	TEIE	0	R/W	Transmitter Error Interrupt Enable Enables the transmitter error interrupts. 0: Transmitter error interrupt disabled 1: Transmitter error interrupt enabled
13	UBOI	0	R/W	User Buffer Overrun Interrupt Enable Enables the user buffer overrun interrupts. 0: User buffer overrun interrupt disabled 1: User buffer overrun interrupt enabled
12	UBUI	0	R/W	User Buffer Underrun Interrupt Enable Enables the user buffer underrun interrupts. 0: User buffer underrun interrupt disabled 1: User buffer underrun interrupt enabled
11	CREI	0	R/W	Clock Recovery Error Interrupt Enable Enables the clock recovery error interrupts. 0: Clock recovery error interrupt disabled 1: Clock recovery error interrupt enabled
10	PAEI	0	R/W	Parity Error Interrupt Enable Enables the parity check error interrupts. 0: Parity check error interrupt disabled 1: Parity check error interrupt enabled
9	PREI	0	R/W	Preamble Error Interrupt Enable Enables the preamble check error interrupts. 0: Preamble error interrupt disabled 1: Preamble error interrupt enabled
8	CSEI	0	R/W	Channel Status Error Interrupt Enable Enables the channel status error interrupts. 0: Channel status error interrupt disabled 1: Channel status error interrupt enabled
7	ABOI	0	R/W	Audio Buffer Overrun Interrupt Enable Enables the receiver audio buffer overrun interrupts. 0: Audio buffer overrun interrupt disabled 1: Audio buffer overrun interrupt enabled
6	ABUI	0	R/W	Audio Buffer Underrun Interrupt Enable Enables the transmitter audio buffer underrun interrupts. 0: Audio buffer underrun interrupt disabled 1: Audio buffer underrun interrupt enabled

Bit	Bit Name	Initial Value	R/W	Description
5	RUII	0	R/W	Receiver User Information Interrupt Enable Enables the receiver user information register full interrupts. 0: Receiver user information interrupt disabled 1: Receiver user information interrupt enabled
4	TUII	0	R/W	Transmitter User Information Interrupt Enable Enables the transmitter user information register empty interrupts. 0: Transmitter user information interrupt disabled 1: Transmitter user information interrupt enabled
3	RCSI	0	R/W	Receiver Channel Status Interrupt Enable Enables the receiver channel status register full interrupts. 0: Receiver channel status interrupt disabled 1: Receiver channel status interrupt enabled
2	RCBI	0	R/W	Receiver Channel Buffer Interrupt Enable Enables the receiver audio channel buffer full interrupts. 0: Receiver audio channel interrupt disabled 1: Receiver audio channel interrupt enabled
1	TCSI	0	R/W	Transmitter Channel Status Interrupt Enable Enables the transmitter channel status register empty interrupts. 0: Transmitter channel status interrupt disabled 1: Transmitter channel status interrupt enabled
0	TCBI	0	R/W	Transmitter Channel Buffer Interrupt Enable Enables the transmitter audio channel buffer empty interrupts. 0: Transmitter audio channel interrupt disabled 1: Transmitter audio channel interrupt enabled

38.7.2 Status Register (STAT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RIS	TIS	UBO	UBU	CE	PARE	PREE	CSE	ABO	ABU	RUIR	TUIR	CSRX	CBRX	CSTX	CBTX
Initial Value	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R	R	R	R	R	R

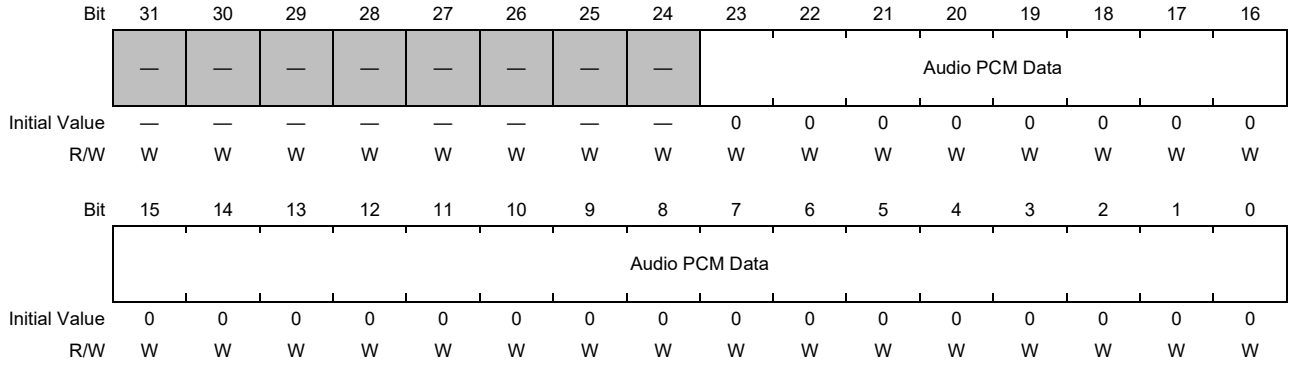
Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CMD	0	R	Compressed Mode Data Sets if the data being received is compressed mode data (When bit 1 = 1 in the V flag and channel status). 0: Data is not in compressed mode 1: Data is in compressed mode
15	RIS	1	R	Receiver Idle State Sets if the receiver is in the idle state. 0: Receiver is not in idle state 1: Receiver in idle state
14	TIS	1	R	Transmitter Idle State Sets if the transmitter is in the idle state. 0: Transmitter is not in idle state 1: Transmitter is in idle state
13	UBO	0	R/WC0	User Buffer Overrun*1 Sets if the receiver user buffer overruns. This bit is cleared by writing 0 to the register. If bit REIE and bit UBOI in the control register are set this causes an interrupt. 0: User buffer has not overrun 1: User buffer has overrun
12	UBU	0	R/WC0	User Buffer Underrun*1 Sets if the transmitter user buffer underrun. This bit is cleared by writing 0. If bits TEIE and UBUI in the control register are set this causes an interrupt. 0: User buffer has not underrun 1: User buffer has underrun
11	CE	0	R/WC0	Clock Error*1 Sets when the clock recovery falls out of synchronization. This bit is cleared by writing 0. If bits REIE and CREI in the control register are set this causes an interrupt. 0: Clock recovery stable 1: Clock recovery error
10	PARE	0	R/WC0	Parity Error*1 Sets when the parity checker produces a fail result. This bit is cleared by writing 0. If bits REIE and PAEI in the control register are set this causes an interrupt. 0: Parity check correct 1: Parity error

Bit	Bit Name	Initial Value	R/W	Description
9	PREE	0	R/WC0	Preamble Error*1 Sets when the start of word preamble fails to appear in the correct place. This bit is cleared by writing 0. If bits REIE and PREI in the control register are set this causes an interrupt. 0: Preamble is in the correct place 1: Preamble error *Note*: Only set after a start of block preamble has occurred.
8	CSE	0	R/WC0	Channel Status Error*1 Sets when the channel status information is written before the 32nd frame of the current block. This bit is cleared by writing 0. If bits TEIE and CSEI in the control register are set this causes an interrupt. 0: Channel status correct 1: Channel status error
7	ABO	0	R/WC0	Audio Buffer Overrun*1 Indicates that the receiver audio buffer is full in both the first and second stages and that data has been overwritten. This bit is cleared by writing 0. If bits REIE and ABOI in the control register are set then this causes an interrupt. 0: Receiver audio buffer has not overrun 1: Receiver audio buffer has overrun
6	ABU	0	R/WC0	Audio Buffer Underrun*1 Indicates that the transmitter audio buffer is empty in both the first and second stages and that the last data transmission has been repeated. This bit is cleared by writing 0. If bits TEIE and ABUI in the control register are set then this causes an interrupt. 0: Transmitter audio buffer has not underrun 1: Transmitter audio buffer has underrun
5	RUIR	0	R	Receiver User Information Register Status Indicates the status of the receiver user information register. This bit is cleared by reading from the receiver user register. If bit RUII in the control register is set then this causes an interrupt. 0: Receiver user information register is empty 1: Receiver user information register is full
4	TUIR	0	R	Transmitter User Information Register Status Indicates the status of the transmitter user information register. This bit is cleared by writing to the transmitter user register. If bit TUII in the control register is set then this causes an interrupt. 0: Transmitter user information register is full 1: Transmitter user information register is empty
3	CSRX	0	R	Channel 1 and Channel 2 Status for Receiver Indicates the status of the receiver channel status registers. This bit is cleared by reading from the receiver channel status registers. If bit RCSI in the control register is set this causes an interrupt. 0: Receiver channel status registers are empty 1: Receiver channel status registers are full
2	CBRX	0	R	Channel 1 and Channel 2 Buffers for Receiver Indicates the status of the receiver audio channel registers. This bit is cleared by reading from the receiver audio channel registers. If bit RCBI in the control register is set this causes an interrupt. 0: Receiver audio channel registers are empty 1: Receiver audio channel registers are full

Bit	Bit Name	Initial Value	R/W	Description
1	CSTX	0	R	Channel 1 and Channel 2 Status for Transmitter Indicates the status of the transmitter channel status registers. This bit is cleared by writing to the transmitter channel status registers. If bit TCSI in the control register is set this causes an interrupt. 0: Transmitter channel status register is full 1: Transmitter channel status register is empty
0	CBTX	0	R	Channel 1 and Channel 2 Buffers for Transmitter Indicates the status of the transmitter audio channel registers. This bit is cleared by writing to the transmitter audio channel registers. If bit TCBI in the control register is set this causes an interrupt. 0: Transmitter audio channel registers are full 1: Transmitter audio channel registers are empty

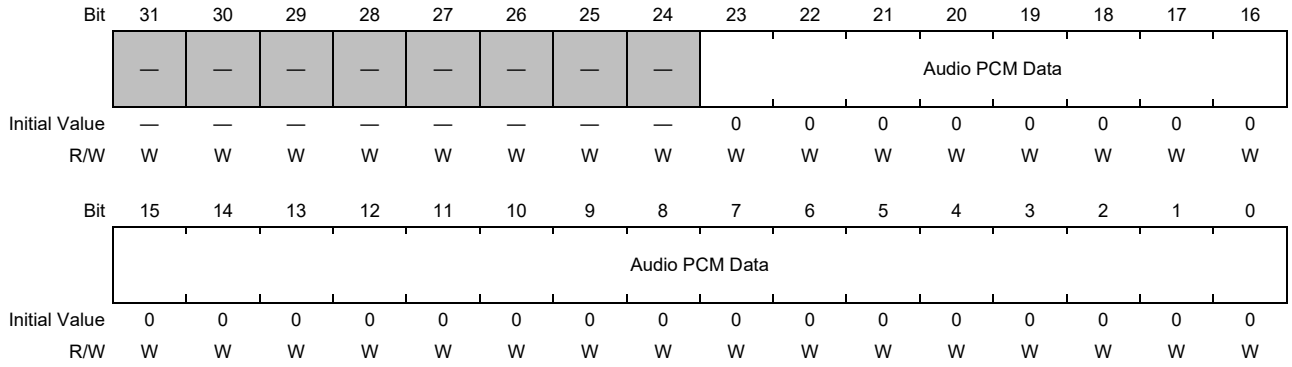
Note 1. When an error bit is detected during DMA transfer, DMA transfer settings must be made again. In this case, the Renesas SPDIF's module enable bit (either the RME or TME bit) and the DMA enable bit (either the RDE or TDE bit) must be disabled and the error status must be cleared before making DMA transfer settings again. Then the module enable bit should be set and DMA transfer can be started again.

38.7.3 Transmitter Channel 1 Audio Register (TLCA)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	W	Reserved
23 to 0	Audio PCM Data	All 0	W	Audio PCM Data LSB aligned PCM encoded audio data.

38.7.4 Transmitter Channel 2 Audio Register (TRCA)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	W	Reserved
23 to 0	Audio PCM Data	All 0	W	Audio PCM Data LSB aligned PCM encoded audio data.

38.7.5 Transmitter DMA Audio Data Register (TDAD)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	Audio PCM Data							
Initial Value	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Audio PCM Data															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	W	Reserved
23 to 0	Audio PCM Data	All 0	W	Audio PCM Data LSB aligned PCM encoded audio data.

38.7.6 Transmitter User Data Register (TUI)

U-bit data in subframes is written into this register. Because U-bit data is transmitted in a sequence of subframes 1 and 2, you need to update the data on a 16-frame basis. For the contents of the user bytes refer to the appropriate standard for the device in use. The user bits to be transmitted are set in sequence starting at the LSB.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	User Byte 4								User Byte 3							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	User Byte 2								User Byte 1							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	User Byte 4	All 0	W	U-bit information is stored here.
23 to 16	User Byte 3	All 0	W	
15 to 8	User Byte 2	All 0	W	
7 to 0	User Byte 1	All 0	W	

38.7.7 Transmitter Channel 1 Status Register (TLCS)

The 30-bit register stores the channel status information to be transmitted. For each channel, channel status information per frame consists of 192 bits. Because necessary data covers only the 30 bits that are set in the following register, zeros continue to be sent after the transmission of the first 30 bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CLAC[1:0]		FS[3:0]				CHNO[3:0]				SRCNO[3:0]			
Initial Value	—	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CATCD[7:0]								—	—	CTL[4:0]					—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	—	W	Reserved
29, 28	CLAC[1:0]	All 0	W	Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Reserved
27 to 24	FS[3:0]	All 0	W	Sample Frequency (FS) 0000: 44.1 kHz 0010: 48 kHz 0011: 32 kHz
23 to 20	CHNO[3:0]	All 0	W	Channel Number 0000: Don’t care 0001: A (left channel) 0010: B (right channel) 0011: C
19 to 16	SRCNO[3:0]	All 0	W	Source Number 0000: Don’t care 0001: 1 0010: 2 0011: 3
15 to 8	CATCD[7:0]	All 0	W	Category Code (Example) 00000000: 2-channel general format 00000001: 2-channel compact disc (IEC 908) 00000010: 2-channel PCM encoder/decoder 00000011: 2-channel digital audio tape recorder
7, 6	—	All 0	W	Reserved The write value should always be 0.
5 to 1	CTL[4:0]	All 0	W	Control The control bits are copied from the source (see IEC60958 standard).
0	—	0	W	Reserved The write value should always be 0.

38.7.8 Transmitter Channel 2 Status Register (TRCS)

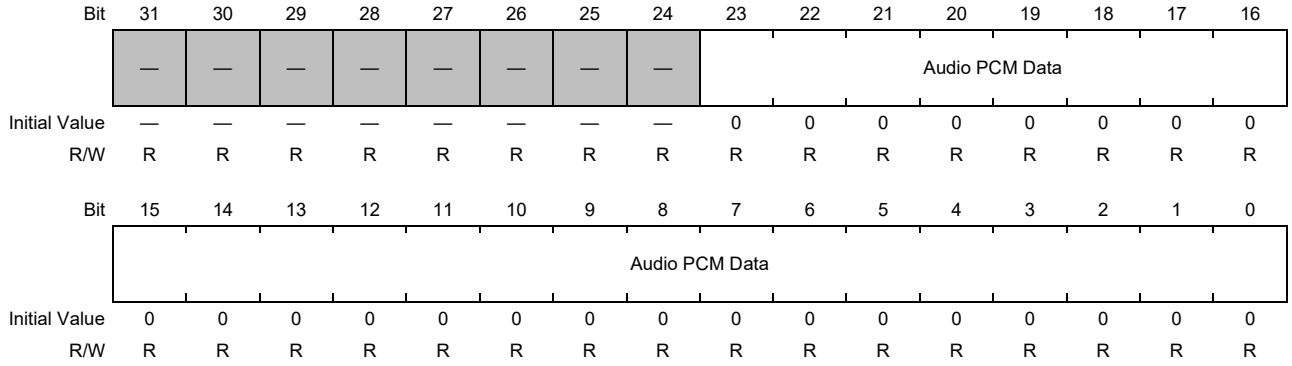
The 30-bit register stores the channel status information to be transmitted. For each channel, channel status information per frame consists of 192 bits. Because necessary data covers only the 30 bits that are set in the following register, zeros continue to be sent after the transmission of the first 30 bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CLAC[1:0]		FS[3:0]				CHNO[3:0]				SRCNO[3:0]			
Initial Value	—	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CATCD[7:0]								—	—	CTL[4:0]					—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

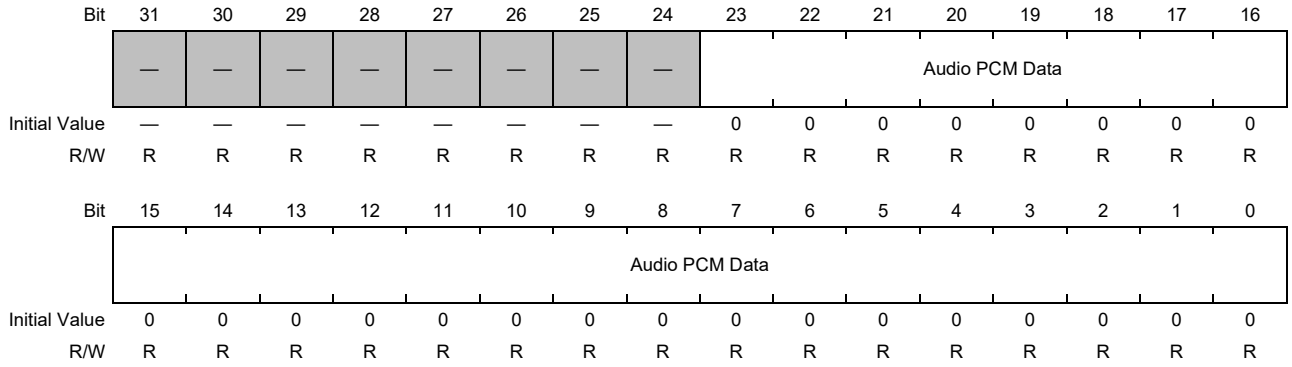
Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	—	W	Reserved
29, 28	CLAC[1:0]	All 0	W	Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Reserved
27 to 24	FS[3:0]	All 0	W	Sample Frequency (FS) 0000: 44.1 kHz 0010: 48 kHz 0011: 32 kHz
23 to 20	CHNO[3:0]	All 0	W	Channel Number 0000: Don’t care 0001: A (left channel) 0010: B (right channel) 0011: C
19 to 16	SRCNO[3:0]	All 0	W	Source Number 0000: Don’t care 0001: 1 0010: 2 0011: 3
15 to 8	CATCD[7:0]	All 0	W	Category Code (Example) 00000000: 2-channel general format 00000001: 2-channel compact disc (IEC 908) 00000010: 2-channel PCM encoder/decoder 00000011: 2-channel digital audio tape recorder
7, 6	—	All 0	W	Reserved The write value should always be 0.
5 to 1	CTL[4:0]	All 0	W	Control The control bits are copied from the source (see IEC60958 standard).
0	—	0	W	Reserved The write value should always be 0.

38.7.9 Receiver Channel 1 Audio Register (RLCA)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	R	Reserved
23 to 0	Audio PCM Data	All 0	R	Audio PCM Data LSB aligned PCM encoded audio data.

38.7.10 Receiver Channel 2 Audio Register (RRCA)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	R	Reserved
23 to 0	Audio PCM Data	All 0	R	Audio PCM Data LSB aligned PCM encoded audio data.

38.7.11 Receiver DMA Audio Data (RDAD)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	Audio PCM Data							
Initial Value	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Audio PCM Data															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	R	Reserved
23 to 0	Audio PCM Data	All 0	R	Audio PCM Data LSB aligned PCM encoded audio data.

38.7.12 Receiver User Data Register (RUI)

The register stores the U-bit data received through the Renesas SPDIF. Because U-bit data is stored in a sequence of subframes 1 and 2 starting at the LSB, you need to read the data on a 16-frame basis. For the contents of the user bytes refer to the appropriate standard for the device in use.

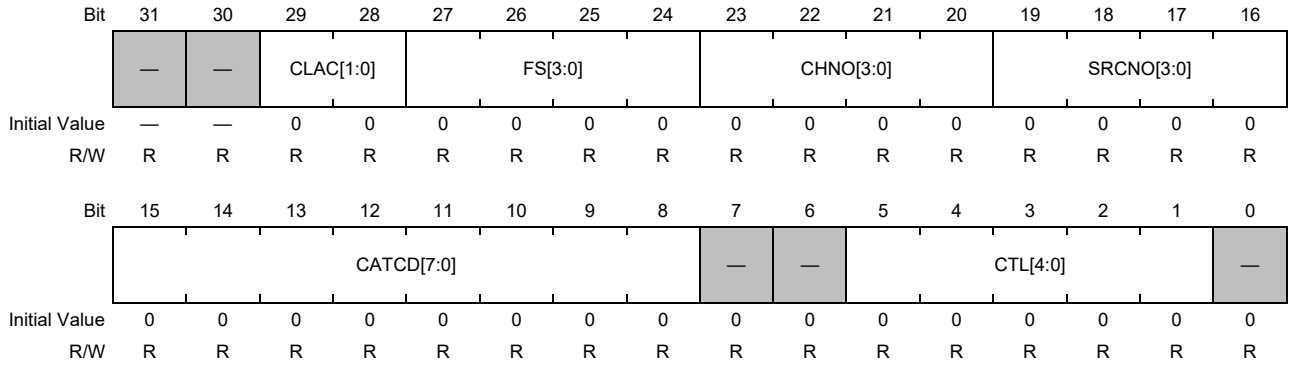
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	User Byte 4								User Byte 3							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	User Byte 2								User Byte 1							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	User Byte 4	All 0	R	U-bit information is stored here.
23 to 16	User Byte 3	All 0	R	
15 to 8	User Byte 2	All 0	R	
7 to 0	User Byte 1	All 0	R	

38.7.13 Receiver Channel 1 Status Register (RLCS)

The channel status is stored starting at the register's LSB in a way that subframe 1 received from the beginning of the block is stored. For the contents of the channel status register, refer to the IEC-60958 standard.



Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	—	R	Reserved
29, 28	CLAC[1:0]	All 0	R	Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Reserved
27 to 24	FS[3:0]	All 0	R	Sample Frequency (fS) 0000: 44.1 kHz 0010: 48 kHz 0011: 32 kHz
23 to 20	CHNO[3:0]	All 0	R	Channel Number 0000: Don't care 0001: A (left channel) 0010: B (right channel) 0011: C
19 to 16	SRCNO[3:0]	All 0	R	Source Number 0000: Don't care 0001: 1 0010: 2 0011: 3
15 to 8	CATCD[7:0]	All 0	R	Category Code (Example) 00000000: 2-channel general format 00000001: 2-channel compact disc (IEC 908) 00000010: 2-channel PCM encoder/decoder 00000011: 2-channel digital audio tape recorder
7, 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5 to 1	CTL[4:0]	All 0	R	Control The control bits are copied from the source (see IEC60958 standard).
0	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

38.7.14 Receiver Channel 2 Status Register (RRCS)

The channel status is stored starting at the register's LSB in a way that subframe 2 received from the beginning of the block is stored. For the contents of the channel status register, refer to the IEC-60958 standard.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CLAC[1:0]		FS[3:0]				CHNO[3:0]				SRCNO[3:0]			
Initial Value	—	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CATCD[7:0]								—	—	CTL[4:0]					—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	—	R	Reserved
29, 28	CLAC[1:0]	All 0	R	Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Reserved
27 to 24	FS[3:0]	All 0	R	Sample Frequency (fS) 0000: 44.1 kHz 0010: 48 kHz 0011: 32 kHz
23 to 20	CHNO[3:0]	All 0	R	Channel Number 0000: Don't care 0001: A (left channel) 0010: B (right channel) 0011: C
19 to 16	SRCNO[3:0]	All 0	R	Source Number 0000: Don't care 0001: 1 0010: 2 0011: 3
15 to 8	CATCD[7:0]	All 0	R	Category Code (Example) 00000000: 2-channel general format 00000001: 2-channel compact disc (IEC 908) 00000010: 2-channel PCM encoder/decoder 00000011: 2-channel digital audio tape recorder
7, 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5 to 1	CTL[4:0]	All 0	R	Control The control bits are copied from the source (see IEC60958 standard).
0	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

38.8 Functional Description — Transmitter

38.8.1 Transmitter Module

The transmitter module transmits PCM data and auxiliary information after encoding it according to the method of biphase-mark modulation that complies with the IEC60958 standard (SPDIF).

The clock for the transmitter module is an oversampling clock supplied from the outside. This clock usually selects a value that serves as an oversample at a frequency eight times larger than the clock frequency required for biphase-mark encoding. In this case, the clock frequency required to transmit 32 time slots in a subframe is 512 times as large as the sample frequency for audio data.

Audio data and channel status information are first written into the module's channel 1 and then into channel 2. Generally, the channel status need to be written only when the information changes. The SPDIF module requests that the channel status be written in 30 frames -- when all the current channel status data have been transmitted. You need to write somewhere between frame 31 and the beginning of the next block of 192 frames.

The audio data is stored in a double buffer arrangement. To make sure that the first stage buffer is empty, you can send an interrupt request or poll the status register. DMA transfers send channel 1 audio data on the first request and channel 2 data on the second.

The channel status information is stored in the 30-bit registers of channels 1 and 2. For each channel, the channel status information per frame consists of 192 bits. Because necessary data covers only 30 bits, zeros continue to be sent after the transmission of the first 30 bits until the block is completed.

User data forms a 32-bit double buffer arrangement. You can make sure that the first stage buffer is empty by either sending an interrupt request or polling the status register. Usually, information about the user data will become insufficient with the length of data between blocks. Transmission takes place in a sequence of channels 1 and 2. For the user data within a block, 384 bits are transmitted before the next block is continuously transmitted.

The audio data handled by the Renesas SPDIF module is a linear PCM, making it possible to set up to 24 bits. For this reason, the V flag indicating that audio data is a linear PCM remains to be 0. The V flag involves no register-based setting. An even parity is created for each 32 bits of serial output data (excluding the preamble).

NOTE

When transmitter user buffer underrun occurs, the current data in the buffer data of SPDIF is transmitted until the next data is filled.

38.8.2 Transmitter Module Initialization

The device defaults to an idle state when it comes out of reset, or can be put into an idle state when 0 is written to the TME bit in the CTRL register. When the transmitter module is idle, it has the following settings:

- The transmitter idle status bit (TIS) is set to 1, all other status bits are cleared to 0.
- Preamble generation is invalid.
- Synchronization between channels 1 and 2 is set to 0 (0 for channel 1, 1 for channel 2).
- Both word_count and frame_count are set to 0.
- The output from the biphase-mark encoder is set to 0.

Channel status, user and audio data registers will retain its value prior to putting the module into idle. To exit the idle state the user must write 1 to the TME bit in the CTRL register.

38.8.3 Initial Settings for Transmitter Module

When the TME bit is set to 1, the TUIR and CSTX bits are set to 1. After that, if data is written in the order of 1) TUI and 2) TLCS and TRCS, a channel status error will occur. To avoid this, be sure to write data in the order of 1) TLCS and TRCS and 2) TUI.

Before writing the first audio data (write access to TLCA or TRCA by the CPU or write access to TDAD by the DMA transfer) after setting the TME bit to 1, be sure to check that the CSTX and TUIR bits are cleared by writing to TLCS, TRCS, and TUI.

38.8.4 Transmitter Module Data Transfer

Once the transmitter module has left the idle state, it is ready for data transfer. Data transfer timing can be achieved in three ways. Either the transfer is done by interrupts, DMA requests or by polling the status register. There is a shared interrupt line (for both transmit and receive) and a single transmitter DMA request line.

Figure 38.5 shows a data transfer with an interrupt for the transmitter.

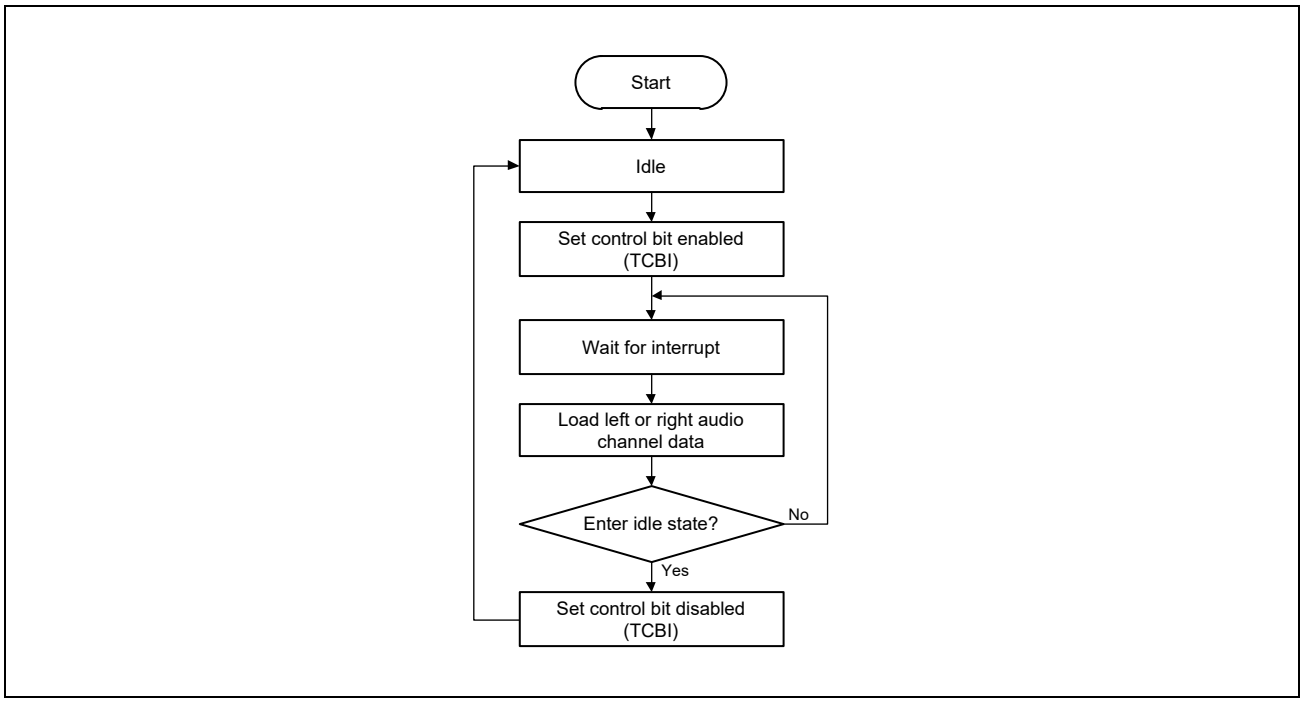


Figure 38.5 Transmitter Data Transfer Flow Diagram - Interrupt Driven

Figure 38.6 shows a data transfer with a DMA transfer for the transmitter.

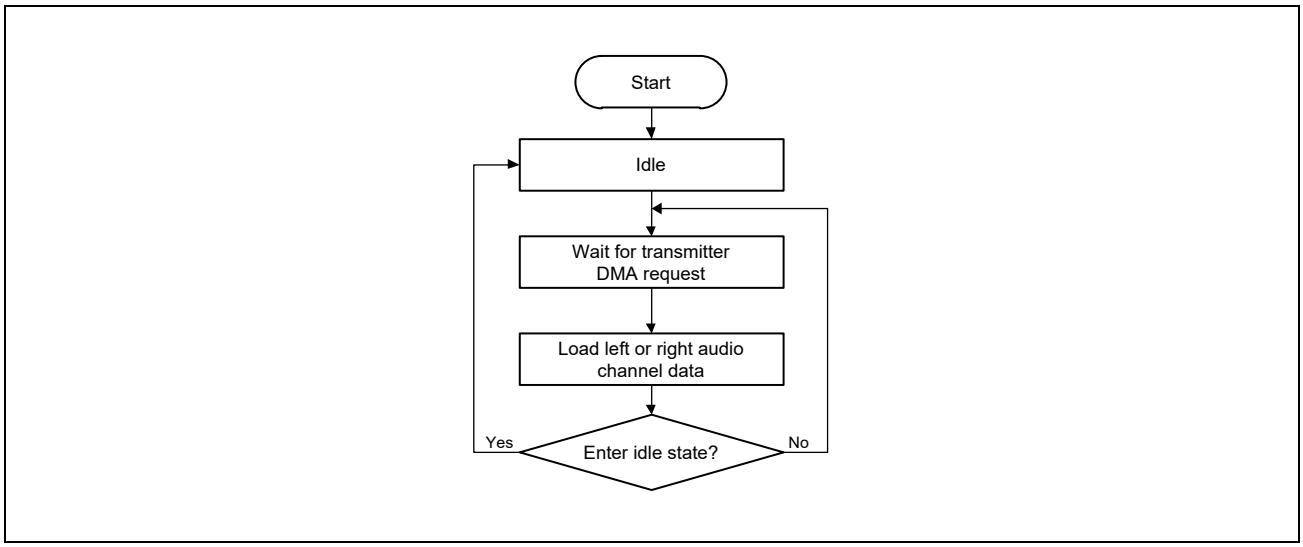


Figure 38.6 Transmitter Data Transfer Flow Diagram - DMA Request Driven

Channel status information is required to be updated when the information has changed. Because the updating needs to be done before the transmission of the next block, the channel status to be updated should be written after 30 frames have been sent; this is indicated either by an interrupt or by polling the status bit. If channel status is written before 30 frames have been sent (while current information is being sent) then an interrupt indicates that the channel status error bit (CSE) in the status register has been set.

NOTE

30 frames contains all the valid information in a single channel status block.

38.9 Functional Description — Receiver

38.9.1 Receiver Module

The receiver module demodulates data and clock signals from the input encoded according to the IEC60958 standard. The encoded data, shown in linear PCM format, is stored into the audio data register. The register also stores the channel status and user information being received simultaneously as auxiliary information.

The main clock for the receiver module is an oversampling clock supplied from the outside. The module operates at a frequency four times as large as the oversampling clock.

NOTE

The oversampling clock is the same for the transmitter and receiver.

Clock recovery is performed using a pulse width counter and averaging filters to produce a sampling pulse in the middle of each bit in the datastream. A clock error status bit indicates clock synchronization loss. Synchronization is achieved when a preamble occurs on the data stream for the first time. Continuous adjustment prevents jitter and/or clock drift from affecting clock recovery, provided that they fall within the clock recovery specifications.

Once the clock recovery is successful the biphase-mark decoder initiates its preamble detection. The decoder searches for the start of block preamble (see **Table 38.2**). A preamble error status bit indicates that following preambles have not appeared at the correct time, such failures are most likely caused by transmission loss or interference.

Even parity checking is performed on the decoded data. A discrepancy will result in the parity error status bit being set.

The SPDIF module acquires user data and channel status information in addition to audio data. The audio is stored in a double buffer arrangement. Either an interrupt request because of a full buffer or polling of the status bit will indicate when the data is ready to be read. DMA transfers receive channel 1 audio data on the first request and channel 2 data on the second.

Channel status is stored in a 30-bit register. Channel status information is received at 1-bit per subframe. Therefore, the registers will not be full until a total of 30 frames for each channel have been received. New channel status is compared with the current data to see if it has changed and is only read by the processor if it has. User data, which is also received at the same time, is stored into the register on a subframe basis, so that the reception is completed when 16 frames are reached.

NOTES

1. Channel status data requests do not support DMA.
2. When receiver user buffer overrun occurs, the current data in the buffer data of SPDIF is overwritten by the next incoming data from SPDIF interface.

38.9.2 Receiver Module Initialization

The device defaults to an idle state when it comes out of reset, or can be put into an idle state by writing 0 to bit RME in the CTRL register. Whilst idle the module has the following settings:

- The receiver idle status bit is set to 1, all other status bits are cleared to 0.
- Synchronization between channels 1 and 2 is set to 0 (0 for channel 1, 1 for channel 2).
- Both Word_count and frame_count are set to 0.

Channel status registers, user data registers and audio data registers will retain its value prior to putting the module into idle. To exit the idle state the user must write 1 to the bit RME in the CTRL register.

38.9.3 Receiver Module Data Transfer

Once the module has left the idle state it is ready for data transfer. Data transfer timing can be achieved in three ways. The transfer can be done by interrupts, or by polling the status register, or by DMA. There is a shared interrupt line (transmit and receive) and a single receiver DMA request line. Data transfer for the receiver can be interrupted by error signals caused by:

1. Clock recovery failure.
2. Transmission loss or interference – indicated by a preamble error.
3. Parity check failure.

Transmission loss or interference can cause the start of subframe or start of block preamble to be misplaced or not present.

Parity check failure occurs when the parity bit is incorrect, this can be caused by any of the above.

[Clock Recovery Deviation]

The receive margin for clock recovery is based on the following equation:

$$M = \left| \left(0.5 - \frac{1}{2N}\right) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100\%$$

where M = receive margin
N = oversampling rate
L = frame length = 33
D = duty cycle = 0.6
F = oversampling clock deviation = Level II accuracy = 1000 in $10e^{-6}$

Figure 38.7 indicates what the receive margin M represents

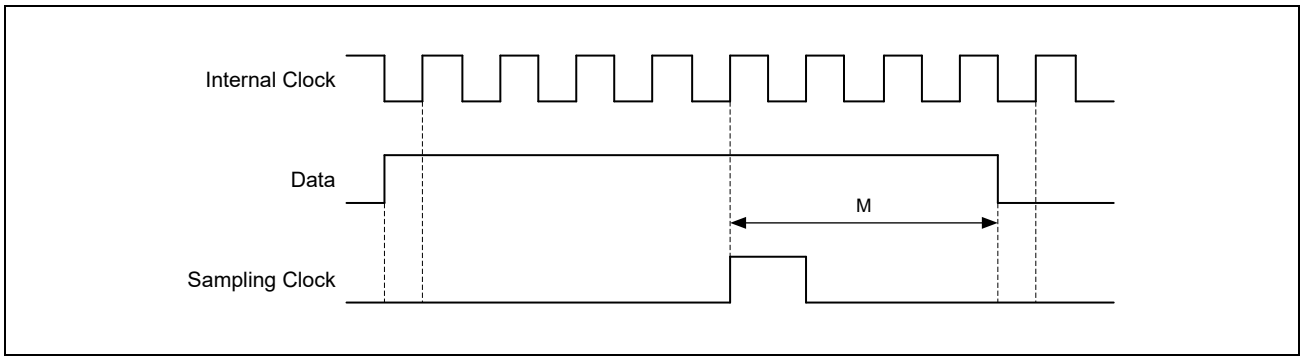


Figure 38.7 Receive Margin

Introducing jitter into the equation gives the following inequality.

$$J \leq \left| \left(0.5 - \frac{1}{2N}\right) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100\%$$

J = clock jitter

Eight times oversampling produces a receive margin = 39.25%
Four times oversampling produces a receive margin = 31.75%
Two times oversampling produces a receive margin = 16.75%

The fastest sample frequency is 48 kHz. This requires a clock speed of 128 × 48 kHz = 6.144 MHz. The worst case jitter in one cycle is specified at 40 ns = 24.5% of the period. This means that an oversampling rate of 4 or more will satisfy the inequality and therefore be sufficient for clock recovery.

Figure 38.8 illustrates the receiver data transfer using interrupts.

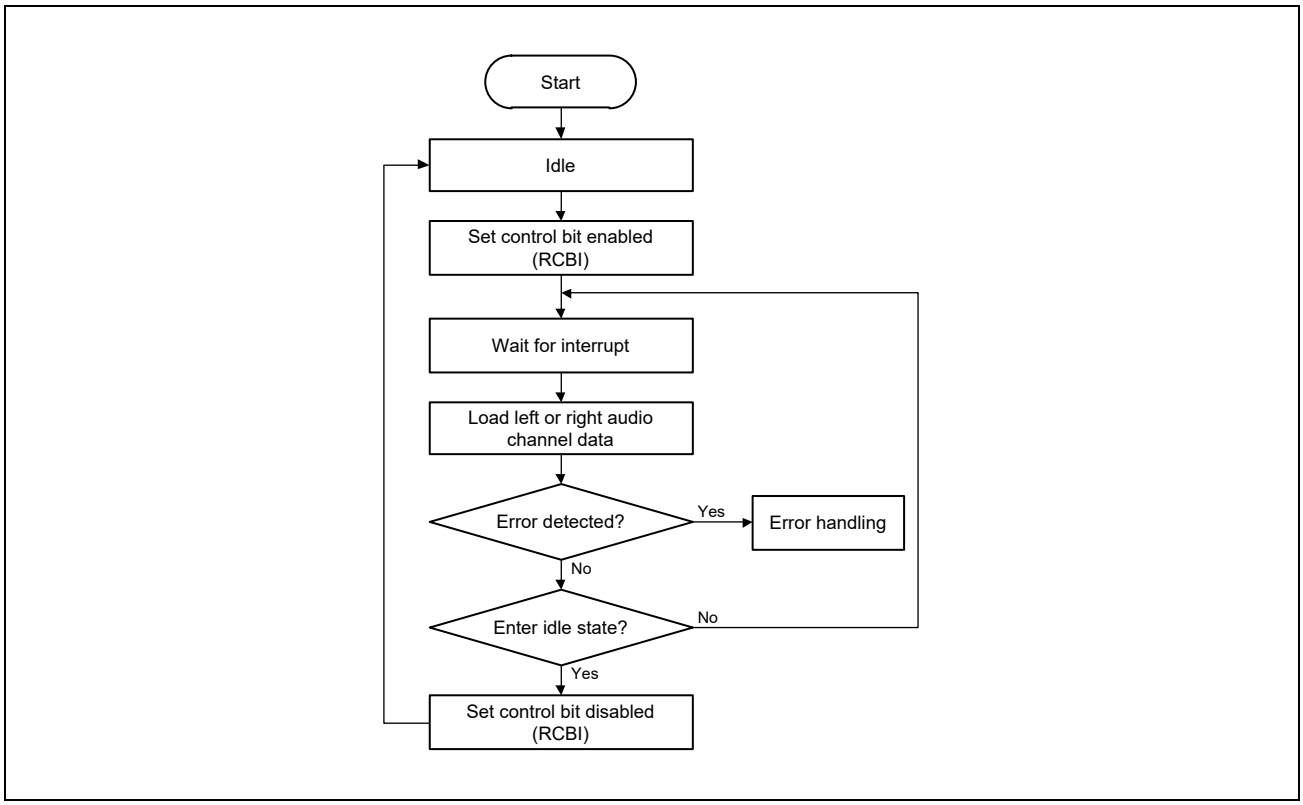


Figure 38.8 Receiver Data Transfer Flow Diagram - Interrupt Driven

Interrupts to indicate that the channel status information register is full occur after frame 30 has been received and only if the information has changed. When the first four bytes have been stored an interrupt occurs.

38.10 Disabling the Module

38.10.1 Transmitter and Receiver Idle

The transmitter or receiver modules can be disabled by writing 0 to the idle bit in the control register (TME for the transmitter and RME for the receiver). The idle state can be detected by polling the idle bit in the status register (TIS and RIS).

38.11 Compressed Mode Data

Compressed mode data is defined in the IEC 61937 specification. This module only detects compressed mode data. This is done by checking the parity flag (V flag) and bit 1 in the channel status data. If both are one then the data is in compressed mode. This is indicated by the setting of the CMD bit in the status register.

NOTE

Only the receiver detects compressed mode data since the information is not relevant to the transmitter.

38.12 References

- IEC60958 Digital Audio Interface
- IEC61937 Compressed Mode Digital Audio Interface

38.13 Usage Notes

38.13.1 Clearing TUIR

After TUI is written to, the TUIR bit is cleared only after transmission of a maximum of one frame is completed. When using a transmitter user information interrupt to write data to TUI, check that the TUIR bit is cleared before terminating the interrupt handling routine so that the interrupt is expectedly accepted again.

38.13.2 Frequency of Clock Input for Audio

The frequency of the clock input to the AUDIO_CLK1 or AUDIO_CLK2 must be lower than the P0Φ frequency.

39. A/D Converter

39.1 Features

This LSI has a successive approximation A/D converter with a 12-bit accuracy. Up to eight analog input channels and one dedicated channel to TSU can be selected.

■ Resolution

12 bits

■ Eight input channels and one dedicated input channel to TSU

■ Input voltage range

0 V to 1.8 V

■ Minimum conversion time

1.0 µs per channel (when A/D conversion clock ADC_ADCLK is 100 MHz)

■ Twelve data registers

The A/D conversion result is stored in a 32-bit data register corresponding to each channel.

■ Sample-and-hold function

■ Trigger mode

- Software trigger mode and hardware trigger mode are available.
- Software trigger mode to start A/D conversion by software
- Hardware trigger mode to start A/D conversion by an asynchronous trigger or synchronous trigger
  - Asynchronous trigger using an external pin (ADC_TRG) as the activation source
  - Synchronous trigger using the multi-function timer pulse unit 3 (MTU3a) or general-purpose
  - PWM timer (GPT) as the activation source

■ Operating mode

- Select mode and scan mode are available.
- Select mode to convert the specified single analog input channel
  - 1-buffer mode or 4-buffer mode can be specified for storing the A/D conversion result.
  - In 4-buffer mode, A/D conversion proceeds four times for the selected analog input and the A/D conversion results are stored in four registers.
- Scan mode to convert the analog inputs of arbitrarily selected channels in ascending order of channel number
  - 1-buffer mode can be specified for storing the A/D conversion result.

■ Conversion mode

- Single mode and repeat mode are available.
- Single mode to proceed A/D conversion only once
- Repeat mode to repeatedly proceed A/D conversion

■ A single interrupt source

- An A/D conversion end interrupt (INTAD) can be generated on completion of A/D conversion.
  - In the select mode with 1-buffer mode selected, an A/D conversion end interrupt is generated on completion of a single conversion.
  - In the select mode with 4-buffer mode selected, an A/D conversion end interrupt is generated on completion of four rounds of conversion.
  - In scan mode, an A/D conversion end interrupt is generated on completion of scanning of all the selected channels.

Table 39.1 shows the outline of the functions of the A/D converter.

Table 39.1 Functions of A/D Converter (1/2)

Item				Source
Analog Input Channel				ADC_CH0 to ADC_CH7 ADC_CH8(TSU)
Conditions for starting A/D conversion	Software	Software trigger		(Enabled by software)
	Asynchronous trigger	External trigger*1	Trigger input pin	ADC_TRG
	Synchronous trigger	Trigger from MTU3a	Compare match with or input capture to MTU0.TGRA	TRGA0N
			Compare match with or input capture to MTU1.TGRA	TRGA1N
			Compare match with or input capture to MTU2.TGRA	TRGA2N
			Compare match with or input capture to MTU3.TGRA	TRGA3N
			Compare match with or input capture to MTU4.TGRA or underflow of MTU4.TCNT in complementary PWM mode	TRGA4N
			Compare match with or input capture to MTU6.TGRA	TRGA6N
			Compare match with or input capture to MTU7.TGRA or underflow of MTU7.TCNT in complementary PWM mode	TRGA7N
			Compare match with MTU0.TGRE	TRG0N
			Compare match between MTU4.TADCORA and MTU4.TCNT	TRG4AN
			Compare match between MTU4.TADCORB and MTU4.TCNT	TRG4BN
			Compare match between MTU4.TADCORA and MTU4.TCNT or compare match between MTU4.TADCORB and MTU4.TCNT	TRG4AN or TRG4BN
			Compare match between MTU4.TADCORA and MTU4.TCNT and compare match between MTU4.TADCORB and MTU4.TCNT (when interrupt skipping function 2 is used)	TRG4ABN
			Compare match between MTU7.TADCORA and MTU7.TCNT	TRG7AN
			Compare match between MTU7.TADCORB and MTU7.TCNT	TRG7BN
			Compare match between MTU7.TADCORA and MTU7.TCNT or compare match between MTU7.TADCORB and MTU7.TCNT compare match	TRG7AN or TRG7BN
			Compare match between MTU7.TADCORA and MTU7.TCNT and compare match between MTU7.TADCORB and MTU7.TCNT (when interrupt skipping function 2 is used)	TRG7ABN

Table 39.1 Functions of A/D Converter (2/2)

Item				Source
Conditions for starting A/D conversion	Synchronous trigger	Trigger from GPT	Compare match with GPT0.GTADTRA	ADTRGA0
			Compare match with GPT0.GTADTRB	ADTRGB0
			Compare match with GPT1.GTADTRA	ADTRGA1
			Compare match with GPT1.GTADTRB	ADTRGB1
			Compare match with GPT2.GTADTRA	ADTRGA2
			Compare match with GPT2.GTADTRB	ADTRGB2
			Compare match with GPT3.GTADTRA	ADTRGA3
			Compare match with GPT3.GTADTRB	ADTRGB3
			Compare match with GPT0.GTADTRA or GPT0.GTADTRB	ADTRGA0 or ADTRGB0
			Compare match with GPT1.GTADTRA or GPT1.GTADTRB	ADTRGA1 or ADTRGB1
			Compare match with GPT2.GTADTRA or GPT2.GTADTRB	ADTRGA2 or DTRGB2
			Compare match with GPT3.GTADTRA or GPT3.GTADTRB	ADTRGA3 or ADTRGB3
Interrupt				INTAD

Note 1. To set ADC_TRG as the trigger for starting A/D conversion, set the general-purpose I/O function. For details, see **Section 45, General Purpose Input Output Port (GPIO)**.

Table 39.2 lists the I/O pins used in the A/D converter.

Table 39.2 I/O Pins of A/D Converter

Pin Name	I/O	Name	Function
ADC_CH0 to ADC_CH7	Input	Analog input pins	Analog input pins
ADC_TRG	Input	A/D conversion trigger input	External trigger input pin for starting A/D conversion
ADC_AVDD18	Input	Analog power supply voltage	A/D converter power supply pin
VSS	Input	Analog ground	A/D converter ground pin

39.2 List of Registers

Table 39.3 shows the register list of this module.

The base address is as follows.

Base address: H'0_1005_8000 (Cortex-A55 Address Space)
Base address: H'4005_8000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)
Base address: H'5005_8000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 39.3 List of Registers of A/D Converter

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
A/D converter mode register 0	ADM0	R/W	H'0000	H'0	32
A/D converter mode register 1	ADM1	R/W	H'0000	H'4	32
A/D converter mode register 2	ADM2	R/W	H'0000	H'8	32
A/D converter mode register 3	ADM3	R/W	H'0000	H'C	32
A/D converter of TSU	TSUMODE	R/W	H'0000	H'10	32
Reserved	—	R	H'0000	H'14 – 1F	32
A/D converter interrupt control register	ADINT	R/W	H'0000	H'20	32
A/D converter status register	ADSTS	R/W	H'0000	H'24	32
Reserved	—	R/W	H'0000	H'28	32
A/D converter external trigger pin filter control register	ADFIL	R/W	H'0000	H'2C	32
A/D conversion result register 0	ADCR0	R	H'0000	H'30	32
A/D conversion result register 1	ADCR1	R	H'0000	H'34	32
A/D conversion result register 2	ADCR2	R	H'0000	H'38	32
A/D conversion result register 3	ADCR3	R	H'0000	H'3C	32
A/D conversion result register 4	ADCR4	R	H'0000	H'40	32
A/D conversion result register 5	ADCR5	R	H'0000	H'44	32
A/D conversion result register 6	ADCR6	R	H'0000	H'48	32
A/D conversion result register 7	ADCR7	R	H'0000	H'4C	32
A/D conversion result register 8	ADCR8	R	H'0000	H'50	32
A/D conversion result register 9	ADCR9	R	H'0000	H'54	32
A/D conversion result register 10	ADCR10	R	H'0000	H'58	32
A/D conversion result register 11	ADCR11	R	H'0000	H'5C	32
Reserved	—	R	H'0000	H'60 – 6C	32
Renesas reserved area*1	—	R/W	Undefined	H'70 – 7F	32

Note: When writing successively to the same register, the notes in **Section 39.5.2, Timing Restrictions**, need to be observed.

Note 1. Writing to a Renesas reserved area is prohibited.

39.3 Register Descriptions

39.3.1 A/D Converter Mode Register 0 (ADM0)

The ADM0 is a 32-bit register that controls A/D conversion and the power-saving mode of the A/D converter, and also executes a software reset. This register can be read from and written to in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRESB	—	—	—	—	—	—	—	—	—	—	—	—	PWDWNB	ADBSY	ADCE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
15	SRESB	0	R/W	Issues a software reset to the A/D converter. Circuits (e.g. registers) other than those of the A/D converter will not be reset. To release the A/D converter from the reset state, write 1 to this bit. 0: A/D converter is reset. 1: A/D converter is released from the reset state. The A/D converter enters the power-saving mode when this bit is used to execute a reset.
14 to 3	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
2	PWDWNB	0	R/W	Specifies the A/D converter to enter or exit the power-saving mode. 0: Power-saving mode 1: Normal mode
1	ADBSY	0	R	Indicates the status of the A/D converter. This bit is read-only. Writing to this bit is ignored. 0: A/D converter is stopped. 1: A/D converter is busy. For operation of the ADBSY bit, see **Figure 39.1**.
0	ADCE	0	R/W	Stops or enables the A/D converter. 0: A/D converter is stopped. 1: A/D converter is enabled. When 0 is written to the ADCE bit, the A/D converter stops operating. For operation of the ADCE bit at reading, see **Figure 39.1**.

Note: Notes regarding the ADM0 register are given on the next page.

NOTES

1. To enable A/D conversion operation using the ADCE bit, first switch from PWDWNB = 0 (power-saving mode) to PWDWNB = 1 (normal mode) and then wait for a period equal to or longer than the stabilization wait time before setting the ADCE bit to 1. The stabilization wait time is 1 µs.
2. To make a transition to the power-saving mode using the PWDWNB bit, first stop operation of the A/D converter (ADCE = 0) and then confirm that A/D conversion is completed (ADBSY = 0) before setting the PWDWNB bit to 0.

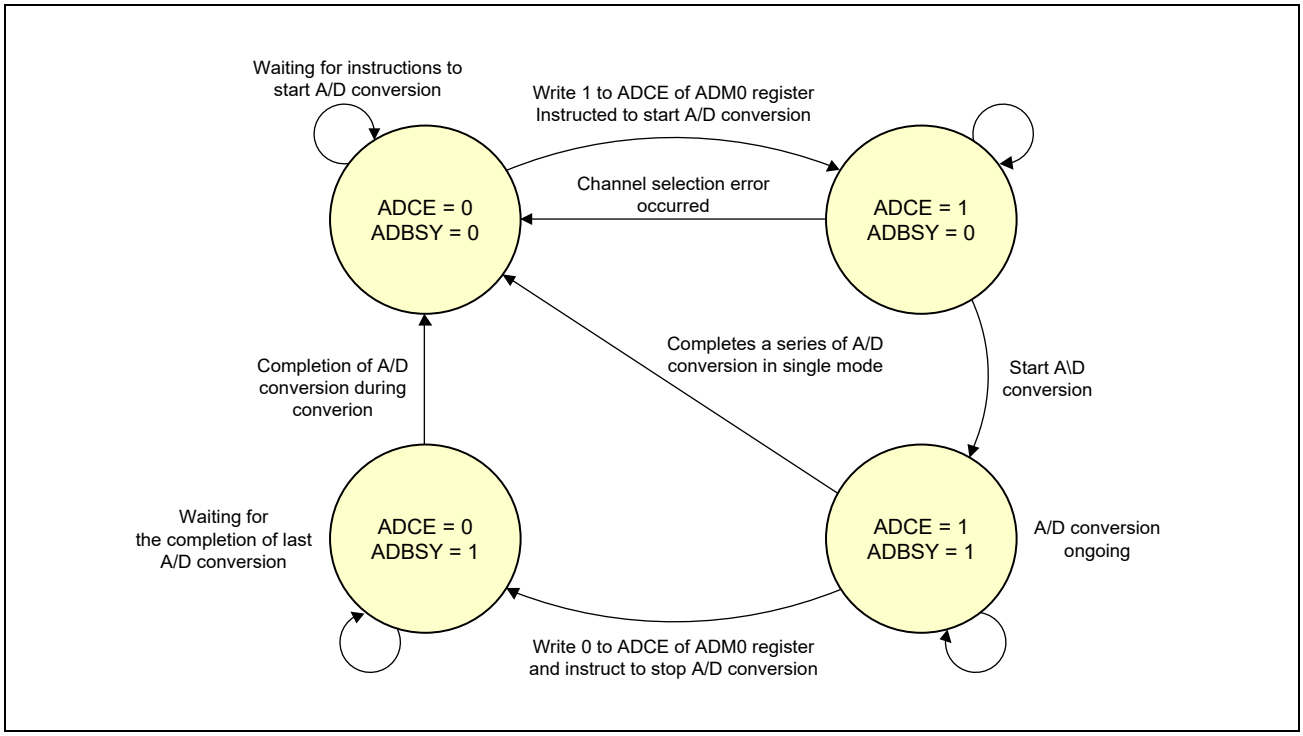


Figure 39.1 State Transitions of ADCE Bit and ADBSY Bit

39.3.2 A/D Converter Mode Register 1 (ADM1)

The ADM1 is a 32-bit register that controls A/D conversion and sets the mode for a hardware trigger. This register can be read from and written to in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	TRGEN5	TRGEN4	TRGEN3	TRGEN2	TRGEN1	TRGEN0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	EGA1	EGA0	—	—	—	—	—	—	—	BS	RPS	MS	TRGIN	TRG
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
21 to 16	TRGEN[5:0]	All 0	R/W	These bits select the hardware trigger pin. For details, see **Table 39.4**.
15, 14	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
13, 12	EGA[1:0]	00	R/W	These bits select the valid edge of the trigger pin.

EGA1	EGA0	Valid Edge
0	0	Hardware trigger is invalid
0	1	Falling edge
1	0	Rising edge
1	1	Both edges

Note: When the EGA[1:0] bits are set to 00 in hardware trigger mode, a trigger from the trigger pin will be ignored. When a trigger pin of the MTU3a or GPT is used, the falling edge is always valid and the setting of the EGA[1:0] bits (01b, 10b, or 11b) will be ignored. When the external trigger pin (ADC_TRG) is used, the valid edge is set by the EGA[1:0] bits.

Bit	Bit Name	Initial Value	R/W	Description
11 to 5	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
4	BS	0	R/W	Selects the buffer mode (valid in select mode). 0: 1-buffer mode 1: 4-buffer mode 4-buffer mode is prohibited in scan mode.
3	RPS	0	R/W	Selects the repeat number. 0: Single 1: Repeat
2	MS	0	R/W	Selects the operating mode. 0: Scan mode 1: Select mode
1	TRGIN	0	R/W	Selects the trigger input mode. 0: Auto mode 1: Step mode

Bit	Bit Name	Initial Value	R/W	Description
0	TRG	0	R/W	Sets the trigger mode. 0: Software trigger mode*1 1: Hardware trigger mode

Note 1. Set the EGA[1:0] bits to 00 when using software trigger mode.

Table 39.4 TRGEN[5:0] Bits and Trigger Source

Module	Source	Description	TRGEN[5:0] Bits					
			5	4	3	2	1	0
External pin	ADC_TRG	External trigger input	0	0	0	0	0	0
MTU	TRGA0N	Compare match with or input capture to MTU0.TGRA	0	0	0	0	0	1
	TRGA1N	Compare match with or input capture to MTU1.TGRA	0	0	0	0	1	0
	TRGA2N	Compare match with or input capture to MTU2.TGRA	0	0	0	0	1	1
	TRGA3N	Compare match with or input capture to MTU3.TGRA	0	0	0	1	0	0
	TRGA4N	Compare match with or input capture to MTU4.TGRA or underflow (trough) of MTU4.TCNT in complementary PWM mode	0	0	0	1	0	1
	TRGA6N	Compare match with or input capture to MTU6.TGRA	0	0	0	1	1	0
	TRGA7N	Compare match with or input capture to MTU7.TGRA or underflow (trough) of MTU7.TCNT in complementary PWM mode	0	0	0	1	1	1
	TRG0N	Compare match with MTU0.TGRE	0	0	1	0	0	0
	TRG4AN	Compare match between MTU4.TADCORA and MTU4.TCNT	0	0	1	0	0	1
	TRG4BN	Compare match between MTU4.TADCORB and MTU4.TCNT	0	0	1	0	1	0
	TRG4AN or TRG4BN	Compare match between MTU4.TADCORA and MTU4.TCNT or compare match between MTU4.TADCORB and MTU4.TCNT	0	0	1	0	1	1
	TRG4ABN	Compare match between MTU4.TADCORA and MTU4.TCNT and compare match between MTU4.TADCORB and MTU4.TCNT (when interrupt skipping function 2 is used)	0	0	1	1	0	0
	TRG7AN	Compare match between MTU7.TADCORA and MTU7.TCNT	0	0	1	1	0	1
	TRG7BN	Compare match between MTU7.TADCORB and MTU7.TCNT	0	0	1	1	1	0
	TRG7AN or TRG7BN	Compare match between MTU7.TADCORA and MTU7.TCNT or compare match between MTU7.TADCORB and MTU7.TCNT	0	0	1	1	1	1
	TRG7ABN	Compare match between MTU7.TADCORA and MTU7.TCNT and compare match between MTU7.TADCORB and MTU7.TCNT (when interrupt skipping function 2 is used)	0	1	0	0	0	0
GPT	ADTRGA0	Compare match with GPT0.GTADTRA	0	1	0	0	0	1
	ADTRGB0	Compare match with GPT0.GTADTRB	0	1	0	0	1	0
	ADTRGA1	Compare match with GPT1.GTADTRA	0	1	0	0	1	1
	ADTRGB1	Compare match with GPT1.GTADTRB	0	1	0	1	0	0
	ADTRGA2	Compare match with GPT2.GTADTRA	0	1	0	1	0	1
	ADTRGB2	Compare match with GPT2.GTADTRB	0	1	0	1	1	0
	ADTRGA3	Compare match with GPT3.GTADTRA	0	1	0	1	1	1
	ADTRGB3	Compare match with GPT3.GTADTRB	0	1	1	0	0	0
	ADTRGA0 or ADTRGB0	Compare match with GPT0.GTADTRA or GPT0.GTADTRB	0	1	1	0	0	1
	ADTRGA1 or ADTRGB1	Compare match with GPT1.GTADTRA or GPT1.GTADTRB	0	1	1	0	1	0
	ADTRGA2 or ADTRGB2	Compare match with GPT2.GTADTRA or GPT2.GTADTRB	0	1	1	0	1	1
	ADTRGA3 or ADTRGB3	Compare match with GPT3.GTADTRA or GPT3.GTADTRB	0	1	1	1	0	0

39.3.3 A/D Converter Mode Register 2 (ADM2)

The ADM2 is a 32-bit register that specifies the analog input channels to be used in A/D conversion. This register can be read from and written to in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CHSEL 8	CHSEL 7	CHSEL 6	CHSEL 5	CHSEL 4	CHSEL 3	CHSEL 2	CHSEL 1	CHSEL 0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
8 to 0	CHSEL[8:0]	All 0	R/W	These bits select the A/D analog input channel n (n = 0 to 8). CHSEL8 is a dedicated channel for TSU. 0: Analog input channel n is not subjected to conversion. 1: Analog input channel n is subjected to conversion.

Note 1. Select only one channel in select mode.

Note 2. If more than one channel is selected in select mode, a channel select error occurs.

39.3.4 A/D Converter Mode Register 3 (ADM3)

The ADM3 is a 32-bit register that sets the sampling period of the A/D converter. Be sure to set this register before starting the A/D converter. This register can be read from and written to in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADIL7	ADIL6	ADIL5	ADIL4	ADIL3	ADIL2	ADIL1	ADIL0	ADCMP7	ADCMP6	ADCMP5	ADCMP4	ADCMP3	ADCMP2	ADCMP1	ADCMP0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ADSMP7	ADSMP6	ADSMP5	ADSMP4	ADSMP3	ADSMP2	ADSMP1	ADSMP0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ADIL[7:0]	All 0	R/W	These bits should be set to H'00.
23 to 16	ADCMP[7:0]	All 0	R/W	These bits should be set to H'1D.
15 to 8	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
7 to 0	ADSMP[7:0]	All 0	R/W	These bits set the sampling period (unit: ADIVCLK cycle). Set above H'4F if select ch.0-7. Set H'FF if measure TSU. T stands for the cycle of ADIVCLK (100 MHz).

Note: The ADIL[7:0] bits should be set to H'00 and the ADCMP[7:0] bits should be set to H'1D. If they are set to other values, normal operation cannot be guaranteed.

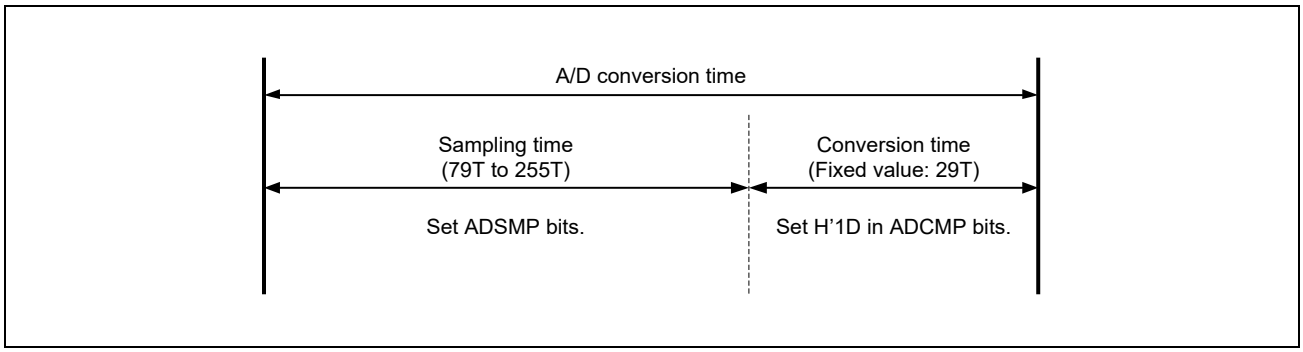


Figure 39.2 Correspondence between ADM3 Register and A/D Conversion Time

39.3.5 TSU Conversion Mode Register (TSUMODE)

The TSUMODE is a 32-bit register that control TSU conversion. This register can be read from and written to in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TSUEN	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
1	TSUEN	All 0	R/W	This bit set the TSU conversion mode. 0: TSU conversion mode is disabled.. 1: TSU conversion mode is enabled.
0	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.

39.3.6 A/D Converter Interrupt Control Register (ADINT)

The ADINT is a 32-bit register that controls interrupts. This register can be read from and written to in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INTS	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CSEEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	INTEN 11	INTEN 10	INTEN 9	INTEN 8	INTEN 7	INTEN 6	INTEN 5	INTEN 4	INTEN 3	INTEN 2	INTEN 1	INTEN 0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	INTS	0	R/W	Selects the attribute of the interrupt (INTAD) signal. 0: Pulse signal 1: Reserved This bit should be set to 0.
30 to 17	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
16	CSEEN	0	R/W	Enables or disables the A/D conversion channel select error interrupt. 0: Interrupt output is disabled. 1: Interrupt output is enabled.
15 to 12	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
11 to 0	INTEN[11:0]	All 0	R/W	These bits enable or disable the conversion end interrupt of channel n (n = 0 to 11). 0: Interrupt output is disabled. 1: Interrupt output is enabled.

The interrupt signal is specified as a pulse and that pulse is output at any of the following timing:

1. An A/D conversion request is detected while the ADM2 register has caused an A/D conversion channel select error
2. Completion of conversion on the channel for which interrupt output has been enabled in the ADINT register
3. An A/D conversion request is detected while 4-buffer mode has been specified in scan mode

39.3.7 A/D Converter Status Register (ADSTS)

The ADSTS is a 32-bit register that controls the state of the A/D converter. This register can be read from and written to in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TRGS	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CSEST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	INST8	INTST7	INTST6	INTST5	INTST4	INTST3	INTST2	INTST1	INTST0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	TRGS	0	R/W	• When reading 0: Nothing detected 1: A trigger was detected during the A/D conversion time in **Figure 39.2**. • When writing 0: No effect 1: The state is cleared.
30 to 17	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
16	CSEST	0	R/W	Indicates the status of the A/D conversion channel select error interrupt. It is the status of the interrupt for an error that occurs when A/D conversion is started with no channel being specified in the CHSEL[8:0] bits in the ADM2 register as a channel on which A/D conversion proceeds. • When reading 0: No A/D conversion channel select error occurred. 1: An A/D conversion channel select error occurred. • When writing 0: No effect 1: The status is cleared.
15 to 9	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
8 to 0	INTST[8:0]	All 0	R/W	Indicates the state of the conversion end interrupt of channel n (n = 0 to 8). • When reading 0: Conversion has not finished. 1: Conversion has finished. • When writing 0: No effect 1: The state is cleared.*1

Note 1. When an interrupt source is generated simultaneously with clearing of the interrupt source, clearing is ignored and the respective bit remains set to 1.

39.3.8 A/D Converter External Trigger Pin Filter Control Register (ADFIL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	FILNUM		—	—	—	FILONOFF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
5, 4	FILNUM[1:0]	00	R/W	These bits select the number of stages of the AD external trigger pin filter. The signal to be filtered is ADC_ADCLK(TSUϕ) (100 MHz). 00: 10.0 ns (100 MHz) × 4 stages 01: 10.0 ns (100 MHz) × 8 stages 10: 10.0 ns (100 MHz) × 12 stages 11: 10.0 ns (100 MHz) × 16 stages
3 to 1	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
0	FILONOFF	0	R/W	Enables or disables the AD external trigger pin filter 0: Filter is disabled. 1: Filter is enabled.

39.3.9 A/D Conversion Result Registers11 to 0 (ADCR11 to ADCR0)

The ADCR11 to ADCR0 are 32-bit registers that hold the A/D conversion results. The A/D converter has twelve 32-bit registers. These registers are read from in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	AD11	AD10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
11 to 0	AD[11:0]	All 0	R	These bits hold the A/D conversion results.

Analog Input Channel	ADCR7 to ADCR0 Registers	
	Select Mode with 1-Buffer Mode Selected or Scan Mode	Select Mode with 4-Buffer Mode Selected
0	ADCR0	ADCR3 to ADCR0
1	ADCR1	
2	ADCR2	
3	ADCR3	
4	ADCR4	ADCR7 to ADCR4
5	ADCR5	
6	ADCR6	
7	ADCR7	
8	ADCR8	ADC11 to ADCR8
9	—	
10	—	
11	—	

The following formula indicates the relationship between the analog input voltage that is input to analog input pins (ADC_CH8 to ADC_CH0) and the A/D conversion results (A/D conversion result registers (ADCR11 to ADCR0)).

$$\mathrm{ADCR} = \mathrm{INT}\left[\frac{\mathrm{Vin}}{\mathrm{ADC\_AVREF}} \times 2^{\mathrm{d}} \times 0.5\right]$$

or

$$(\mathrm{ADCR} - 0.5) \times \frac{\mathrm{ADC\_AVREF}}{2^{\mathrm{d}}} \leq \mathrm{Vin} < (\mathrm{ADCR} + 0.5) \times \frac{\mathrm{ADC\_AVREF}}{2^{\mathrm{d}}}$$

INT[ ]:	Function that returns the integer portion of the value enclosed in [ ]
Vin:	Analog input voltage
ADC_AVREF:	Voltage of power supply pin(ADC_AVDD18) for the analog unit
ADCR:	Value of A/D conversion result registers (ADCR11 to ADCR0)
d:	Resolution of A/D converter (d = 12 in this LSI)

The relationship between the analog input voltage and A/D conversion result is shown in the figure below.

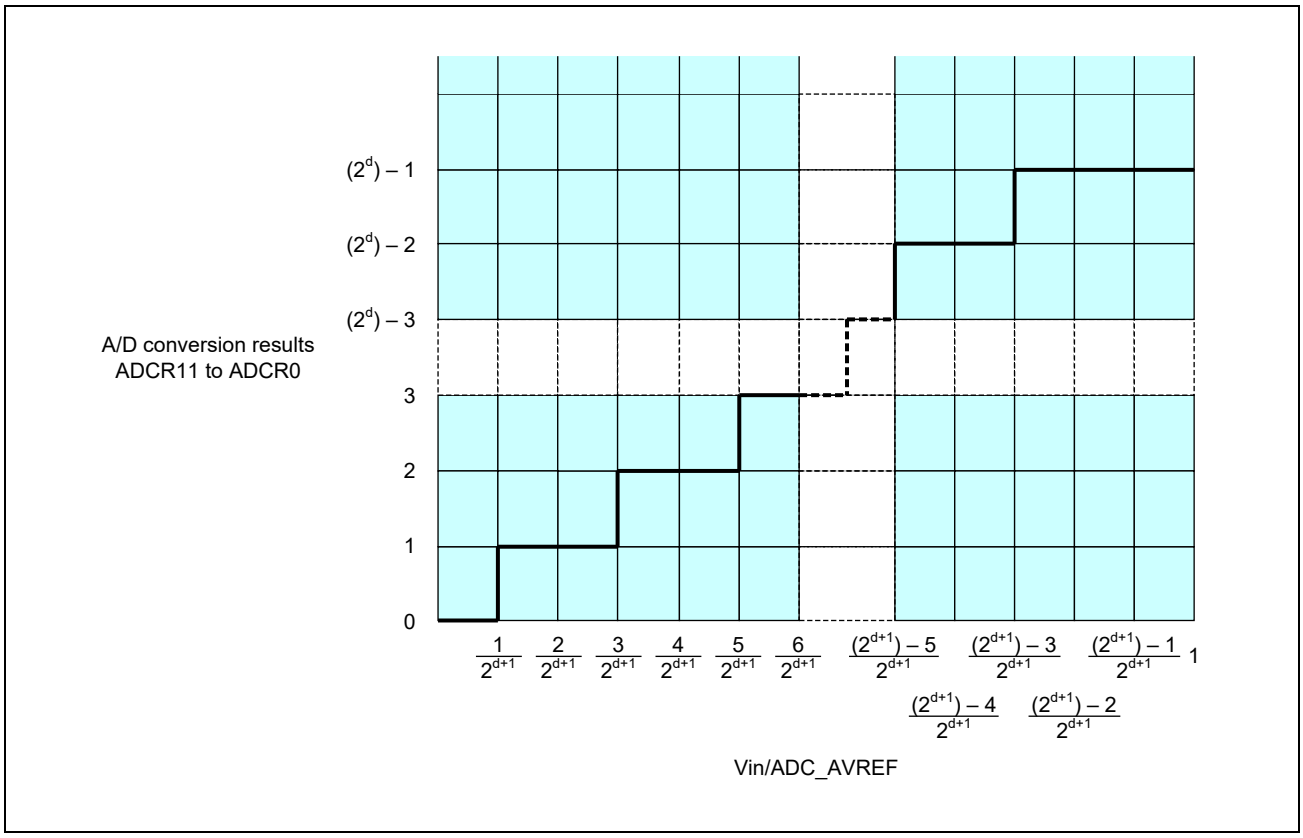


Figure 39.3 Relationship between Analog Input Voltage and A/D Conversion Result

39.4 Operation

39.4.1 Types of A/D Conversion Modes

The following A/D conversion modes can be specified. A/D conversion modes can be set in the ADM1 register.

Table 39.5 A/D Conversion Modes

Trigger Mode	Trigger Input Mode	Operating Mode	Conversion Count	Buffer Count	Operation	Example of A/D Conversion for Reference
Software trigger	—	Select	Single	1 buffer	Selected 1 channel × 1-time conversion	**Section 39.4.4.1, Example of A/D Conversion in Select Mode with Single Mode Selected**
				4 buffers	Selected 1 channel × 4-time conversion	—
			Repeat	1 buffer	(Selected 1 channel × 1-time conversion) × repeated	**Section 39.4.4.2, Example of A/D Conversion in Select Mode with Repeat Mode Selected**
				4 buffers	(Selected 1 channel × 4-time conversion) × repeated	**Section 39.4.4.3, Example of A/D Conversion in 4-Buffer Mode**
		Scan	Single	1 buffer	All selected channels × 1-time conversion	**Section 39.4.4.4, Example of A/D Conversion in Scan Mode with Single Mode Selected**
				4 buffers	(Setting prohibited)	
			Repeat	1 buffer	(All selected channels × 1-time conversion) × repeated	**Section 39.4.4.5, Example of A/D Conversion in Scan Mode with Repeat Mode Selected**
				4 buffers	(Setting prohibited)	
Hardware trigger	Auto (started with a single trigger input)	Select	Single	1 buffer	Selected 1 channel × 1-time conversion	—
				4 buffers	Selected 1 channel × 4-time conversion	—
			Repeat	1 buffer	(Selected 1 channel × 1-time conversion) × repeated	—
				4 buffers	(Selected 1 channel × 4-time conversion) × repeated	—
		Scan	Single	1 buffer	All selected channels × 1-time conversion	**Section 39.4.4.6, Example of A/D Conversion in Auto Mode**
				4 buffers	(Setting prohibited)	
			Repeat	1 buffer	(All selected channels × 1-time conversion) × repeated	—
				4 buffers	(Setting prohibited)	
	Step (conversion proceeds at every trigger input)	Select	Single	1 buffer	Selected 1 channel × 1-time conversion	—
				4 buffers	Selected 1 channel × 4-time conversion	—
			Repeat	1 buffer	(Selected 1 channel × 1-time conversion) × repeated	—
				4 buffers	(Selected 1 channel × 4-time conversion) × repeated	—
		Scan	Single	1 buffer	All selected channels × 1-time conversion	**Section 39.4.4.7, Example of A/D Conversion in Step Mode**
				4 buffers	(Setting prohibited)	
			Repeat	1 buffer	(All selected channels × 1-time conversion) × repeated	—
				4 buffers	(Setting prohibited)	

Note: When 4-buffer mode is set in scan mode, a channel select error occurs.

39.4.1.1 Trigger Modes

Software trigger mode and hardware trigger mode are the two types of trigger modes in which the trigger is the timing for starting the A/D conversion processing. These trigger modes can be set by the TRG bit in the ADM1 register.

(1) Software Trigger Mode

In software trigger mode, A/D conversion is started for the input of the ADC_CH8 to ADC_CH0 pins by setting the ADCE bit in the ADM0 register to 1. The ADBSY bit in the ADM0 register is 1 while the converter is operating.

The hardware trigger is invalid when software trigger mode is set. Set the EGA[1:0] bits in the ADM1 register to 00 when using software trigger mode. When a hardware trigger is input, A/D conversion is not started and the TRGS bit in the ADSTS register retains 0.

(2) Hardware Trigger Mode

There are two triggers in hardware trigger mode: asynchronous trigger using the ADC_TRG pin as the activation source and synchronous trigger using the multi-function timer pulse unit 3 (MTU3a) or general purpose PWM timer (GPT) as the activation source.

The valid edge of the selected hardware trigger can be set by the EGA[1:0] bits in the ADM1 register.

When the ADCE bit in the ADM0 register is set to 1, the A/D converter enters a standby state for a hardware trigger after 250 ns as passed and will start conversion operation upon detection of a valid edge. The ADBSY bit in the ADM0 register remains to be 1 during conversion operation.

The status of the TRGS bit in the ADSTS register can be cleared by writing 1 to the TRGS bit.

In hardware trigger mode, the available trigger input modes are auto mode and step mode. The trigger input mode can be set in the TRGIN bit in the ADM1 register.

(a) Auto mode

When a hardware trigger is input once, A/D conversion is repeated automatically for the conversion count in accordance with the specified conversion settings.

(b) Step mode

A/D conversion proceeds every time a hardware trigger is input. For example, if select mode, single mode, and 4-buffer mode are selected, conversion finishes when a hardware trigger has been input four times.

39.4.1.2 Operating mode

There are two operating modes: select mode and scan mode. The select mode has 1-buffer mode and 4-buffer mode as sub-modes. These modes can be set in the BS and MS bits in the ADM1 register.

(1) Select Mode

A/D conversion proceeds for the analog input specified in the CHSEL[8:0] bits in the ADM2 register. The A/D conversion results are stored in the ADCR11 to ADCR0 registers corresponding to the ADC_CH8 to ADC_CH0 pins. In select mode, 1-buffer mode or 4-buffer mode can be used as the method for storing the A/D conversion results.

Only one channel is selectable for A/D conversion in select mode.

If more than one analog input channel is selected in select mode, a channel select error occurs.

(a) 1-buffer mode

A/D conversion proceeds only once for the analog input specified in the ADM2 register and the conversion result is stored in the ADCR8 to ADCR0 registers corresponding to the ADC_CH8 to ADC_CH0 pins.

The ADC_CH8 to ADC_CH0 pins correspond with the ADCR8 to ADCR0 registers on a one-on-one basis. If interrupt output is enabled in the INTEN[8:0] bits in the ADINT register for the channels on which A/D conversion proceeds, an A/D conversion end interrupt (INTAD) occurs on completion of a single conversion.

This mode is usable for reading the result of a single conversion.

(b) 4-buffer mode

A/D conversion proceeds four times for the analog input specified in the ADM2 register and the conversion results are stored in the ADCR11 to ADCR8 or ADCR7 to ADCR4 or ADCR3 to ADCR0 registers. See **Table 39.7** for the correspondence between the analog inputs and A/D conversion result registers.

If interrupt output is enabled in the INTEN[11:0] bits in the ADINT register for the channels corresponding to the ADC11 to ADCR0 registers where the A/D conversion result is to be stored in 1-buffer mode, an A/D conversion end interrupt (INTAD) occurs when the A/D conversion result is stored in the ADCR11 to ADCR0 registers. For example, to generate an interrupt on the fourth A/D conversion when ADC_CH0 is selected, set the INTEN3 bit in the ADINT register to 1.

See **Table 39.6** for the relationship between the setting of the ADINT register and the output of an A/D conversion end interrupt in 4-buffer mode.

This mode is usable for obtaining the average of the A/D conversion results.

Table 39.6 ADINT Setting and Generation of A/D Conversion End Interrupt in 4-Buffer Mode

Channel for A/D Conversion	ADINT Register Setting		A/D Conversion End Interrupt
1 channel is selected from ADC_CH3 to ADC_CH0	INTEN[0]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 0.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 0.
	INTEN[1]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 1.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 1.
	INTEN[2]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 2.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 2.
	INTEN[3]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 3.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 3.
1 channel is selected from ADC_CH7 to ADC_CH4	INTEN[4]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 4.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 4.
	INTEN[5]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 5.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 5.
	INTEN[6]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 6.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 6.
	INTEN[7]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 7.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 7.
1 channel is selected from ADC_CH8	INTEN[8]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 8.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 8.
	INTEN[9]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 9.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 9.
	INTEN[10]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 10.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 10.
	INTEN[11]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 11.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 11.

Table 39.7 Correspondence between Analog Inputs and A/D Conversion Result Registers in 4-Buffer Mode

Analog Input	A/D Conversion Result Register
1 channel is selected from ADC_CH3 to ADC_CH0	ADCR0 (first time)
	ADCR1 (second time)
	ADCR2 (third time)
	ADCR3 (fourth time)
1 channel is selected from ADC_CH7 to ADC_CH4	ADCR4 (first time)
	ADCR5 (second time)
	ADCR6 (third time)
	ADCR7 (fourth time)
1 channel is selected ADC_CH8	ADCR8 (first time)
	ADCR9 (second time)
	ADCR10 (third time)
	ADCR11 (fourth time)

(2) Scan Mode

A/D conversion proceeds for the analog inputs of channels selected in the ADM2 register, in ascending order of channel number. The A/D conversion results are stored in the ADCR8 to ADCR0 registers corresponding to the analog inputs. If interrupt output is enabled in the INTEN[8:0] bits in the ADINT register for the channels on which A/D conversion proceeds, an A/D conversion end interrupt (INTAD) occurs when A/D conversion on the relevant channel finishes.

This mode is usable for constantly monitoring multiple analog signals.

In scan mode, only 1-buffer mode can be specified. If 4-buffer mode is specified in scan mode, a channel select error occurs.

39.4.1.3 Conversion Mode

Single mode and repeat mode are available. Use the RPS bit in the ADM1 register to set the conversion mode.

(1) Single Mode

When A/D conversion for the number of times of the conversion count finishes, the ADCE bit in the ADM0 register is automatically cleared to 0. For the number of times of conversion, see the description of operation in **Table 39.5**.

(2) Repeat Mode

When 1 is written to the ADCE bit in the ADM0 register, A/D conversion repeatedly proceeds in the A/D conversion mode specified in the ADM1 register. Immediately after 0 is written to the ADCE bit in the ADM0 register, A/D conversion is finished.

For the number of times of conversion, see the description of operation in **Table 39.5**.

39.4.2 Interrupt Functions

Table 39.8 lists the interrupt sources of the A/D converter and the conditions for generating them and the methods for confirming and clearing the interrupt sources.

The INTS bit in the ADINT register is used to set the operation for outputting an interrupt request of the INTAD.

Table 39.8 List of Interrupt Functions

Interrupt Source	Condition for Generation	Enabling Interrupt ADINT Register	Confirming Interrupt Source ADSTS Register*1	Method for Clearing Interrupt Source ADSTS Register
A/D conversion end	When A/D conversion on the specified channel is completed	INTEN[11:0]*2	INTST[11:0]	Write 1 to the bits that were 1 when the register was read.
A/D conversion channel select error	• When all of the CHSEL[7:0] bits in the ADM2 register were 0 at the start of A/D conversion*3 • When multiple analog input channels are selected in select mode*3 • When 4-buffer mode is specified in scan mode*3	CSEEN	CSEST	Write 1 to the CSEST bit.

Note 1. If an interrupt source is generated regardless of the settings of the INTEN[11:0] and CSEEN bits in the ADINT register, the bit corresponding to the interrupt source is set to 1 in the ADSTS register.

Note 2. The A/D conversion end interrupt can be enabled or disabled individually for each channel.

Note 3. When an A/D conversion channel select error occurs, the ADCE bit in the ADM0 register becomes 0 and A/D conversion operation is stopped.

39.4.3 Procedures of A/D Conversion

Follow the procedures given below for A/D conversion.

(1) Procedure for starting A/D conversion

After the A/D converter is released from the software reset state, use the ADM3 to ADM0 registers*1 to select the analog input channels on which A/D conversion proceeds and specify the trigger mode (software trigger or hardware trigger), operating mode (select or scan), etc.

Setting the ADCE bit in the ADM0 register to 1 will start A/D conversion in software trigger mode, whereas the A/D converter will enter a standby state for a trigger*2 in hardware trigger mode.

Note 1. Set the ADM3 to ADM0 registers when conversion by the A/D converter is stopped (ADBSY bit in the ADM0 register is 0).

Note 2. If the ADCE bit in the ADM0 register is set to 1 in hardware trigger mode, a transition is made to a trigger standby state. A/D conversion is started (ADBSY bit in ADM0 register is 1) by a hardware trigger signal and the A/D converter returns to the trigger standby state (ADBSY bit in ADM0 register is 0) on completion of A/D conversion.

A/D conversion operation is enabled.

After the A/D converter makes a transition from power-saving mode to normal mode, a period as long as the stabilization wait time is required. The stabilization wait time is 1 μs.

After the stabilization wait time has passed, set the ADCE bit in the ADM0 register to 1.

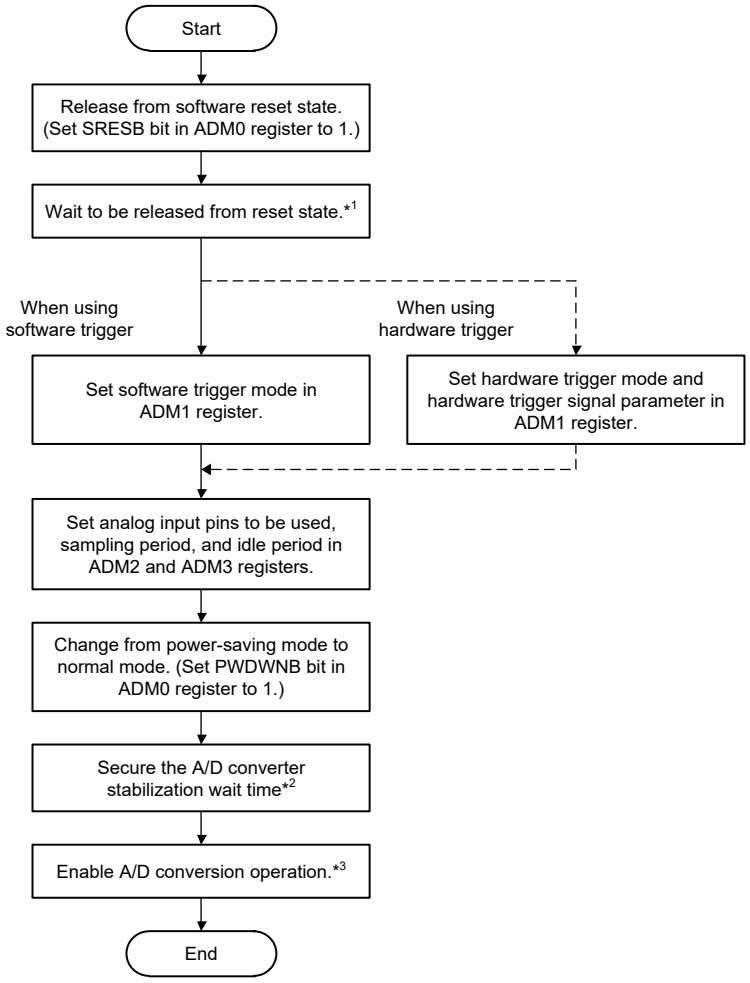
In software trigger mode, setting the ADCE bit to 1 starts A/D conversion.

In hardware trigger mode, setting the ADCE bit to 1 makes the A/D converter enter a standby state for a hardware trigger after 250 ns has passed, and conversion operation is started upon detection of a valid edge.

(2) When A/D conversion is finished, the A/D conversion results are stored in the ADCR11 to ADCR0 registers. Also, an A/D conversion end interrupt (INTAD) occurs on completion of A/D conversion on the channels specified in the INTEN[8:0] bits in the ADINT register.

39.4.3.1 Procedure for Starting A/D Conversion

Use the following flowchart to start A/D conversion.



Note 1. The time waiting for the A/D converter to be released from the reset state is 2 × ADC_ADCLK(TSUϕ).

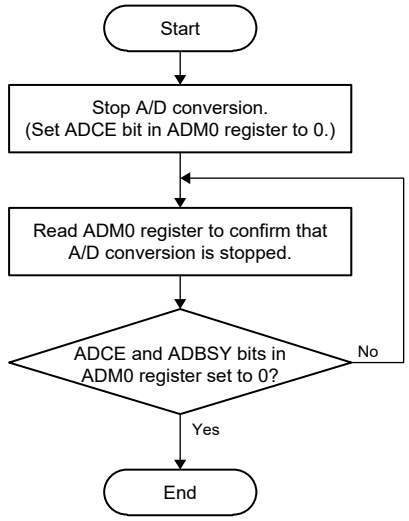
Note 2. The stabilization wait time is 1 µs.

Note 3. In software trigger mode, A/D conversion is started by setting the ADCE bit to 1.
In hardware trigger mode, after the ADCE bit is set to 1, A/D conversion is started upon detection of a valid edge.

Figure 39.4 Procedure for Starting A/D Conversion

39.4.3.2 Procedure for Stopping A/D Conversion

Use the following flowchart to stop A/D conversion.

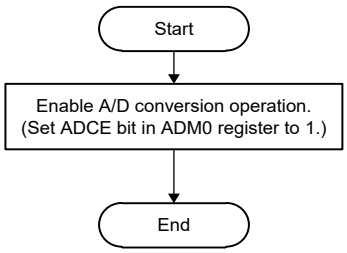


Note: For details on the operation for stopping conversion by writing 0 to the ADCE bit, see **Table 39.10**.

Figure 39.5 Procedure for Stopping A/D Conversion

39.4.3.3 Procedure for Stopping and Restarting A/D Conversion

Use the following flowchart to restart A/D conversion after it has been stopped with the procedure for stopping A/D conversion.



Note: For details on the operation for restarting conversion, see **Table 39.10**.

Figure 39.6 Procedure for Stopping and Restarting A/D Conversion

39.4.3.4 Procedure for Entering Power-saving Mode

Use the following flowchart to make a transition to the power-saving mode.

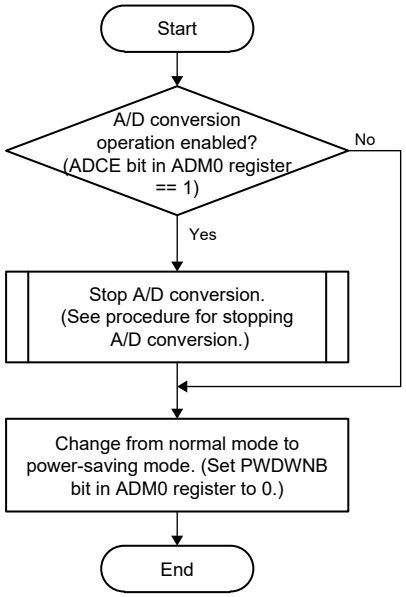
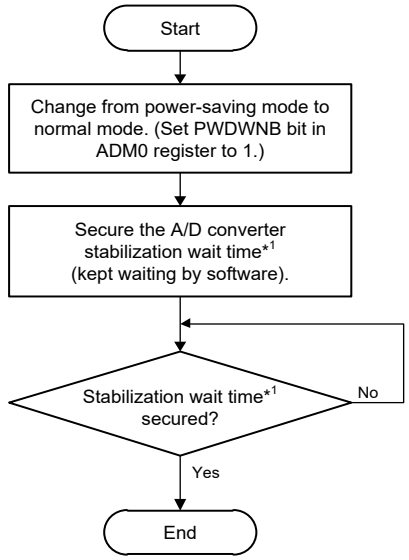


Figure 39.7 Procedure for Entering Power-Saving Mode

39.4.3.5 Procedure for Releasing from Power-saving Mode

Use the following flowchart to release the A/D converter from the power-saving mode.



Note 1. The stabilization wait time is 1 µs.

Figure 39.8 Procedure for Releasing from Power-Saving Mode

39.4.3.6 Procedure for a Software Reset

Use the following flowchart to execute a software reset.

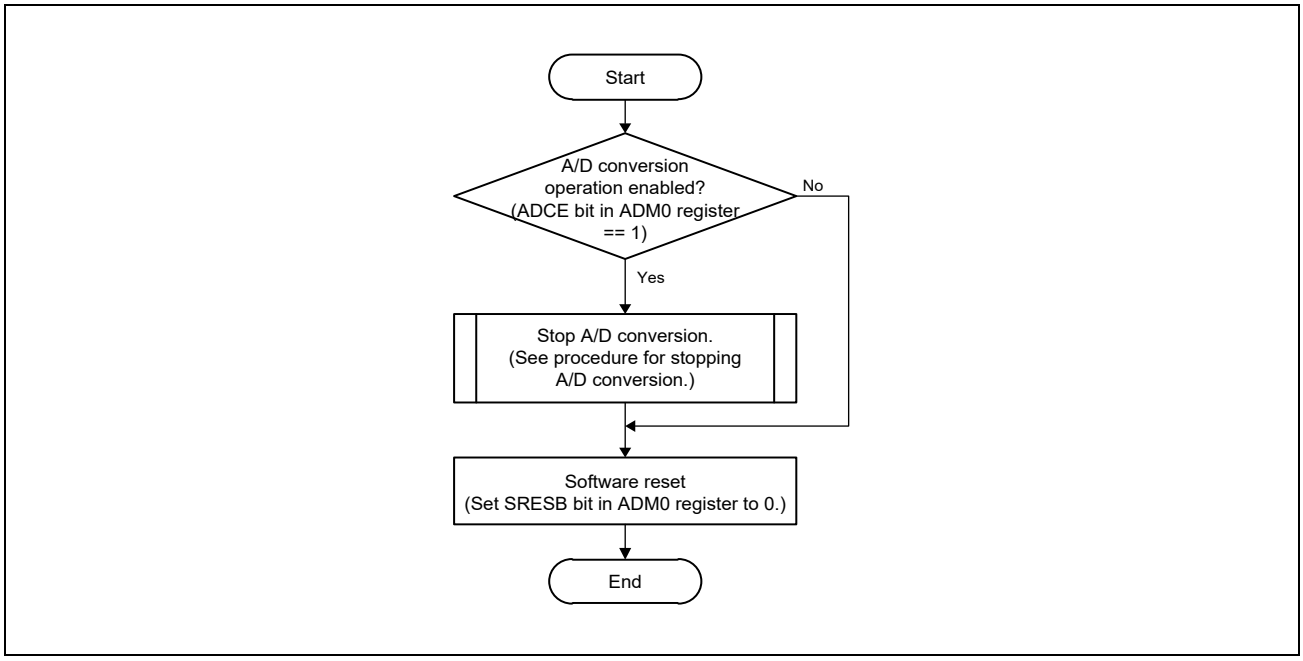
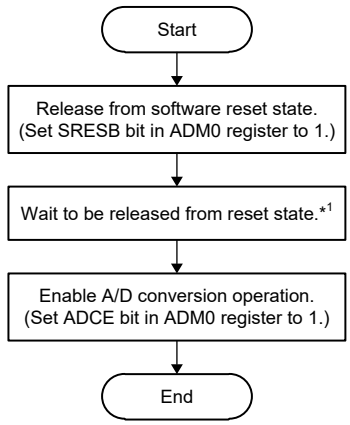


Figure 39.9 Procedure for a Software Reset

39.4.3.7 Procedure for Restarting A/D Conversion after a Software Reset

Use the following flowchart to start A/D conversion, then execute a software reset with the procedure for a software reset, and finally restart A/D conversion.



Note 1. The time waiting for the A/D converter to be released from the reset state is 2 × ADC_ADCLK(TSUϕ).

Figure 39.10 Procedure for Restarting A/D Conversion after a Software Reset

39.4.3.8 Procedure for Interrupt Processing

Use the flowchart in **Figure 39.11** to process an interrupt request that was output by the A/D converter.

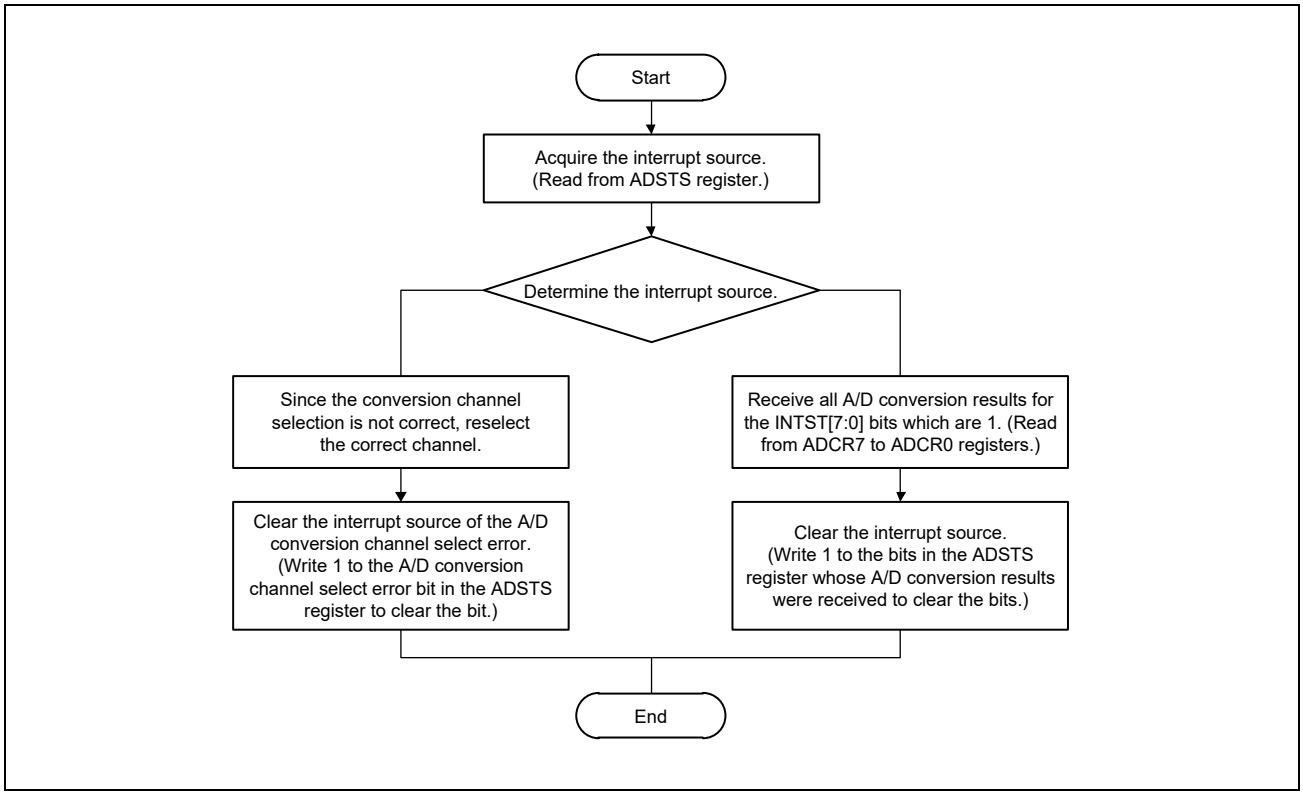


Figure 39.11 Procedure for Interrupt Processing

39.4.4 Examples of A/D Conversion

This section gives examples of A/D conversion.

1) Example of A/D conversion in select mode with single mode selected
2) Example of A/D conversion in select mode with repeat mode selected
3) Example of A/D conversion in 4-buffer mode
4) Example of A/D conversion in scan mode with single mode selected
5) Example of A/D conversion in scan mode with repeat mode selected
6) Example of A/D conversion in auto mode
7) Example of A/D conversion in step mode

Note to confirm the differences between modes shown below because not all possible combinations are included in this manual.

- For the differences between 1-buffer mode and 4-buffer mode, refer to 1) and 3).
- For the differences between single mode and repeat mode, refer to 1) and 2).
- For the differences between select mode and scan mode, refer to 1) and 4) or 2) and 5)
- For the differences between auto mode and step mode, refer to 6) and 7)
- For the differences between software trigger mode and hardware trigger mode, refer to 4) and 6).

39.4.4.1 Example of A/D Conversion in Select Mode with Single Mode Selected

Figure 39.12 shows an example of A/D conversion with software trigger mode, select mode, single mode and 1-buffer mode are selected. In the example in **Figure 39.12**, conversion proceeds with the analog input changed from ADC_CH1 to ADC_CH2.

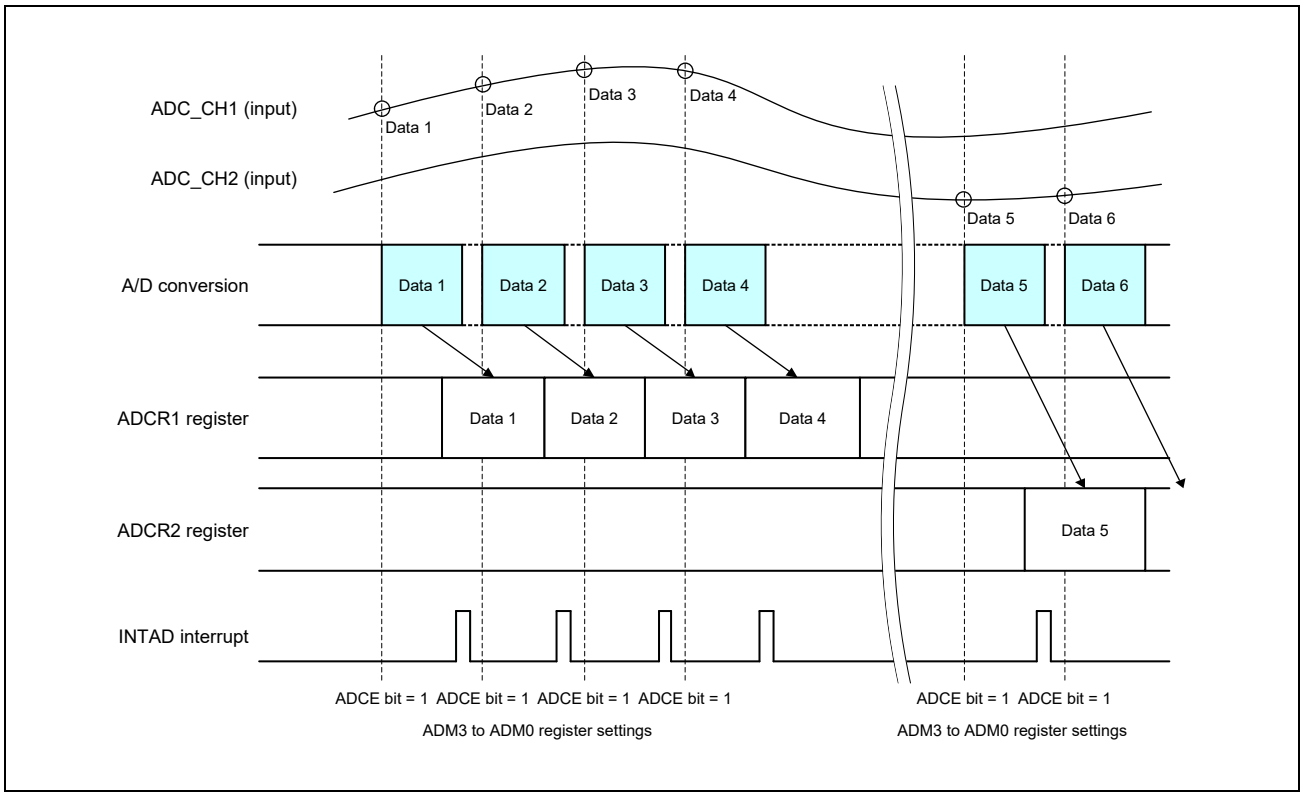


Figure 39.12 Example of A/D Conversion in Select Mode with Single Mode Selected

1. The SRESB bit in the ADM0 register is set to 1 to release the A/D converter from the software reset state.
2. The A/D converter waits to be released from the reset state (wait time is 2 × ADC_ADCLK(TSUϕ)).
3. In the ADM3 to ADM0 registers, software trigger mode, select mode, single mode, and 1-buffer mode are selected, and analog input pin ADC_CH1 is set.
4. In the ADINT and ADSTS registers, the interrupt signal is set as a pulse and interrupt output of channel 1 is enabled.
5. Normal mode is selected by setting the PWDWNB bit in the ADM0 register to 1.
6. The A/D converter is kept waiting for a period equal to or longer than the stabilization wait time*1 by software.
7. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
8. A/D conversion proceeds on ADC_CH1 (A/D conversion result is stored in the ADCR1 register).
9. An INTAD interrupt occurs.
10. Steps 7 to 9 above are repeated.
11. A/D conversion operation is stopped by setting the ADCE bit in the ADM0 register to 0.
12. In the ADM3 to ADM0 registers, software trigger mode, select mode, single mode, and 1-buffer mode are selected, and analog input pin ADC_CH2 is set.

13. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
14. A/D conversion proceeds on ADC_CH2 (A/D conversion result is stored in the ADCR2 register).
15. An INTAD interrupt occurs.
16. Steps 13 to 15 above are repeated.

Note 1. The stabilization wait time is 1 μs.

39.4.4.2 Example of A/D Conversion in Select Mode with Repeat Mode Selected

Figure 39.13 shows an example of A/D conversion with software trigger mode, select mode, repeat mode, and 1-buffer mode are selected. In the example in **Figure 39.13**, conversion proceeds with the analog input changed from ADC_CH1 to ADC_CH2.

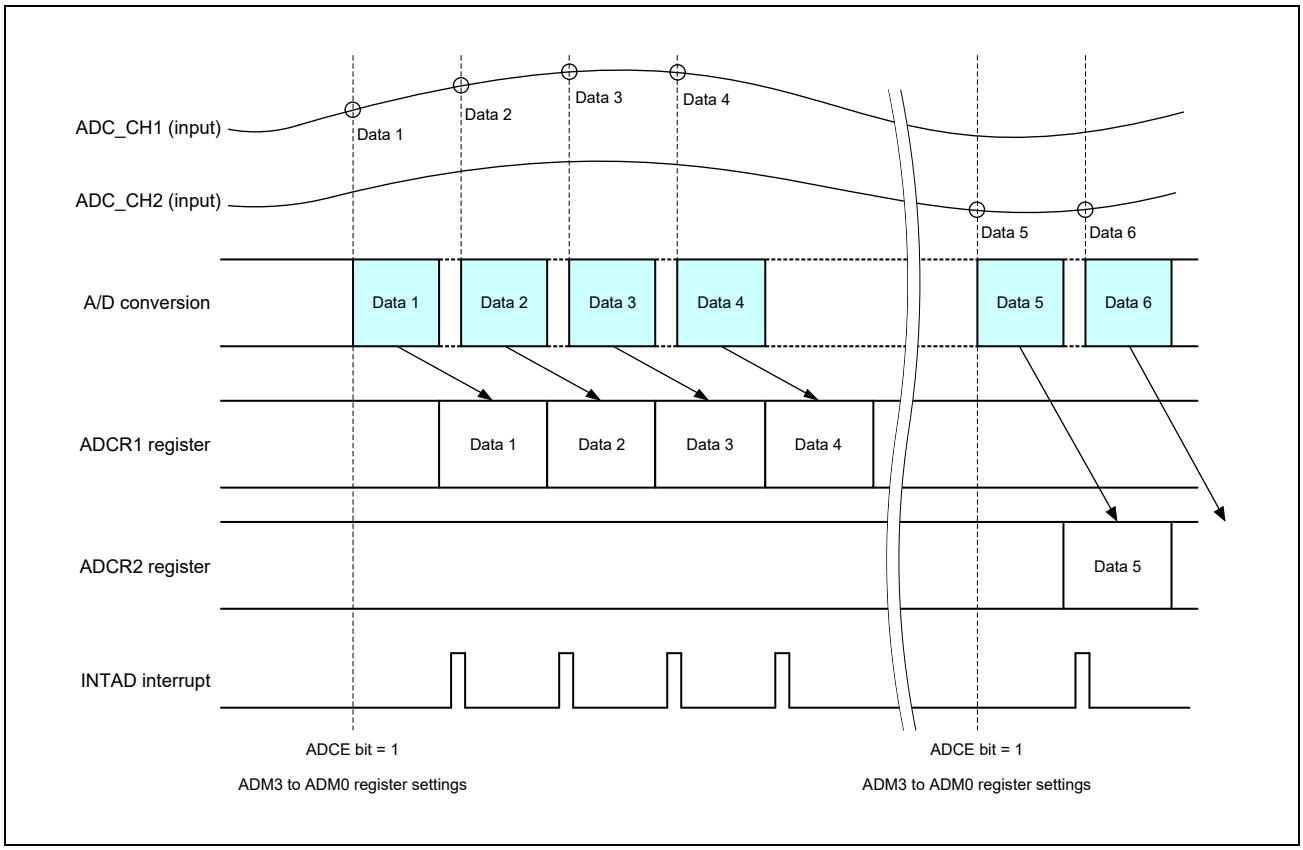


Figure 39.13 Example of A/D Conversion in Select Mode with Repeat Mode Selected

1. The SRESB bit in the ADM0 register is set to 1 to release the A/D converter from the software reset state.
2. The A/D converter waits to be released from the reset state (wait time is 2 × ADC_ADCLK(TSUϕ)).
3. In the ADM3 to ADM0 registers, software trigger mode, select mode, repeat mode, and 1-buffer mode are selected, and analog input pin ADC_CH1 is set.
4. In the ADINT and ADSTS registers, the interrupt signal is set as a pulse and interrupt output of channel 1 is enabled.
5. Normal mode is selected by setting the PWDWNB bit in the ADM0 register to 1.
6. The A/D converter is kept waiting for a period equal to or longer than the stabilization wait time*1 by software.
7. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
8. A/D conversion proceeds on ADC_CH1 (A/D conversion result is stored in the ADCR1 register).
9. An INTAD interrupt occurs.
10. Steps 8 and 9 above are repeated automatically.
11. A/D conversion operation is stopped by setting the ADCE bit in the ADM0 register to 0.

12. In the ADM3 to ADM0 registers, software trigger mode, select mode, repeat mode, 1-buffer mode are selected, and analog input pin ADC_CH2 is set.
13. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
14. A/D conversion proceeds on ADC_CH2 (A/D conversion result is stored in the ADCR2 register).
15. An INTAD interrupt occurs.
16. Steps 14 and 15 above are repeated.

Note 1. The stabilization wait time is 1 μs.

39.4.4.3 Example of A/D Conversion in 4-Buffer Mode

Figure 39.14 shows an example of A/D conversion with software trigger mode, select mode, repeat mode, and 4-buffer mode are selected. In the example in **Figure 39.14**, conversion proceeds with the analog input changed from ADC_CH2 to ADC_CH4.

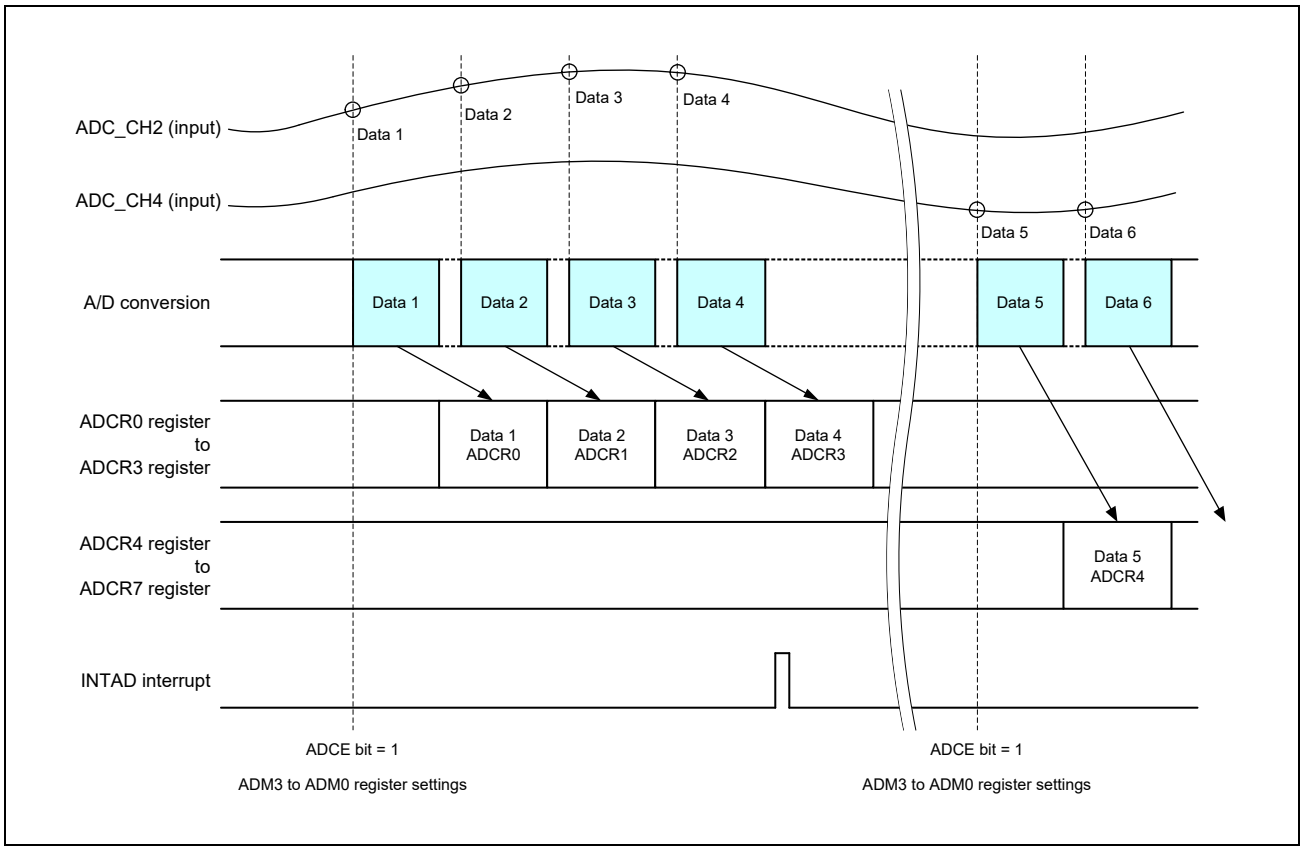


Figure 39.14 Example of A/D Conversion in 4-Buffer Mode

1. The SRESB bit in the ADM0 register is set to 1 to release the A/D converter from the software reset state.
2. The A/D converter waits to be released from the reset state (wait time is 2 × ADC_ADCLK(TSUϕ)).
3. In the ADM3 to ADM0 registers, software trigger mode, select mode, repeat mode, and 4-buffer mode are selected, and analog input pin ADC_CH2 is set.
4. In the ADINT and ADSTS registers, the interrupt signal is set as a pulse and interrupt output of channel 3 is enabled.
5. Normal mode is selected by setting the PWDWNB bit in the ADM0 register to 1.
6. The A/D converter is kept waiting for a period equal to or longer than the stabilization wait time*1 by software.
7. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
8. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 1) is stored in the ADCR0 register).
9. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 2) is stored in the ADCR1 register).
10. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 3) is stored in the ADCR2 register).
11. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 4) is stored in the ADCR3 register).

12. An INTAD interrupt occurs.
13. Steps 8 to 12 above are repeated automatically.
14. A/D conversion operation is stopped by setting the ADCE bit in the ADM0 register to 0.
15. In the ADM3 to ADM0 registers, software trigger mode, select mode, repeat mode, and 4-buffer mode are selected, and analog input pin ADC_CH4 is set.
16. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
17. A/D conversion proceeds on ADC_CH4 (A/D conversion result (data 5) is stored in the ADCR4 register).
18. A/D conversion proceeds on ADC_CH4 (A/D conversion result (data 6) is stored in the ADCR5 register).

...

Note 1. The stabilization wait time is 1 μs.

39.4.4.4 Example of A/D Conversion in Scan Mode with Single Mode Selected

Figure 39.15 shows an example of A/D conversion with software trigger mode, scan mode, and single mode are selected. In the example in **Figure 39.15**, conversion proceeds with ADC_CH3 to ADC_CH0 selected.

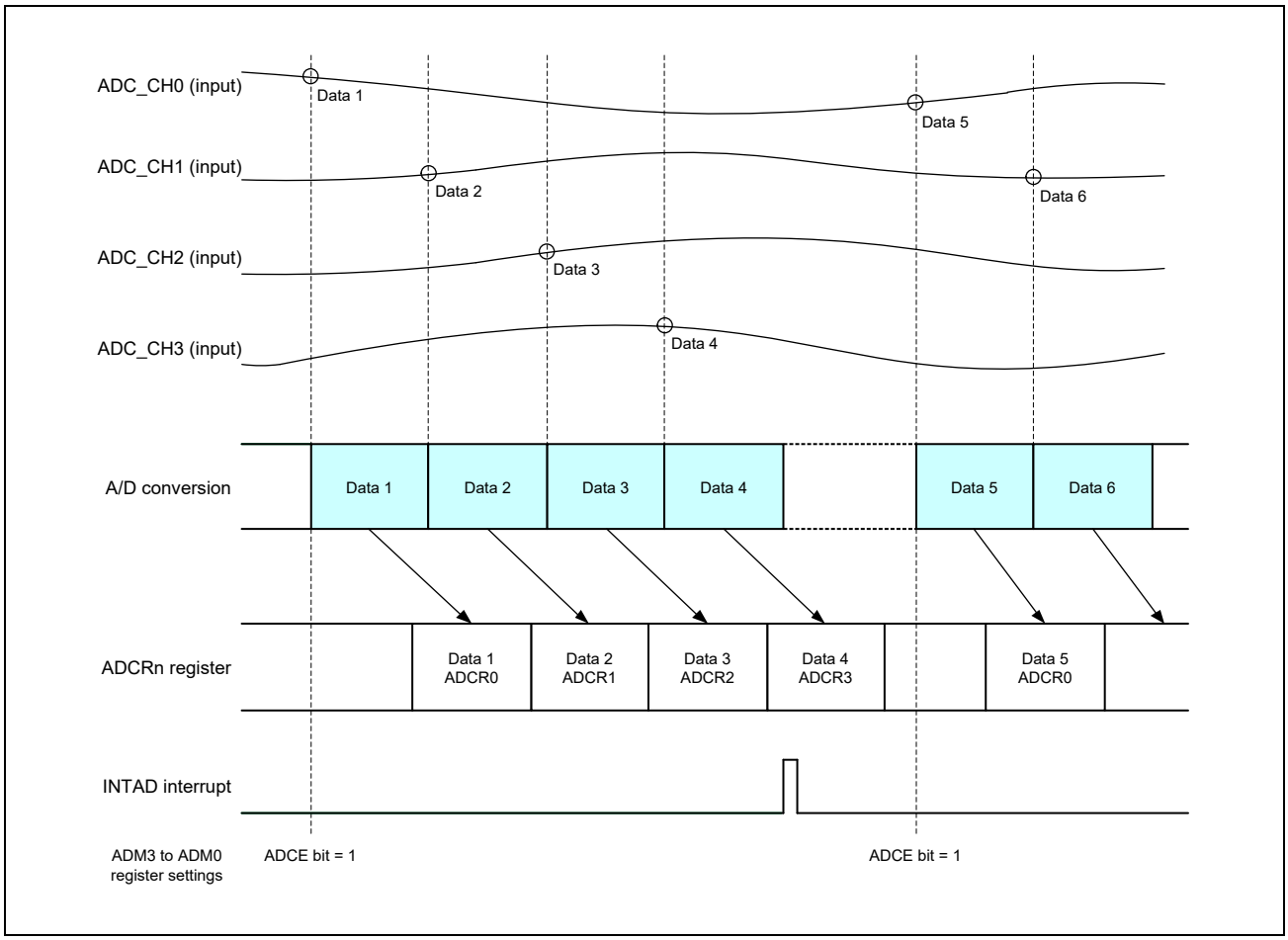


Figure 39.15 Example of A/D Conversion in Scan Mode with Single Mode Selected

1. The SRESB bit in the ADM0 register is set to 1 to release the A/D converter from the software reset state.
2. The A/D converter waits to be released from the reset state (wait time is 2 × ADC_ADCLK(TSUϕ)).
3. In the ADM3 to ADM0 registers, software trigger mode, scan mode, and single mode are selected, and analog input pins ADC_CH3 to ADC_CH0 are set.
4. In the ADINT and ADSTS registers, the interrupt signal is set as a pulse and interrupt output of channel 3 is enabled.
5. Normal mode is selected by setting the PWDWNB bit in the ADM0 register to 1.
6. The A/D converter is kept waiting for a period equal to or longer than the stabilization wait time*1 by software.
7. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
8. A/D conversion proceeds on ADC_CH0 (A/D conversion result (data 1) is stored in the ADCR0 register).
9. A/D conversion proceeds on ADC_CH1 (A/D conversion result (data 2) is stored in the ADCR1 register).
10. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 3) is stored in the ADCR2 register).

11. A/D conversion proceeds on ADC_CH3 (A/D conversion result (data 4) is stored in the ADCR3 register).
12. An INTAD interrupt occurs.
13. Steps 7 to 12 above are repeated.

Note 1. The stabilization wait time is 1 µs.

39.4.4.5 Example of A/D Conversion in Scan Mode with Repeat Mode Selected

Figure 39.16 shows an example of A/D conversion with software trigger mode, scan mode, and repeat mode are selected. In the example in **Figure 39.16**, conversion proceeds with ADC_CH3 to ADC_CH0 selected.

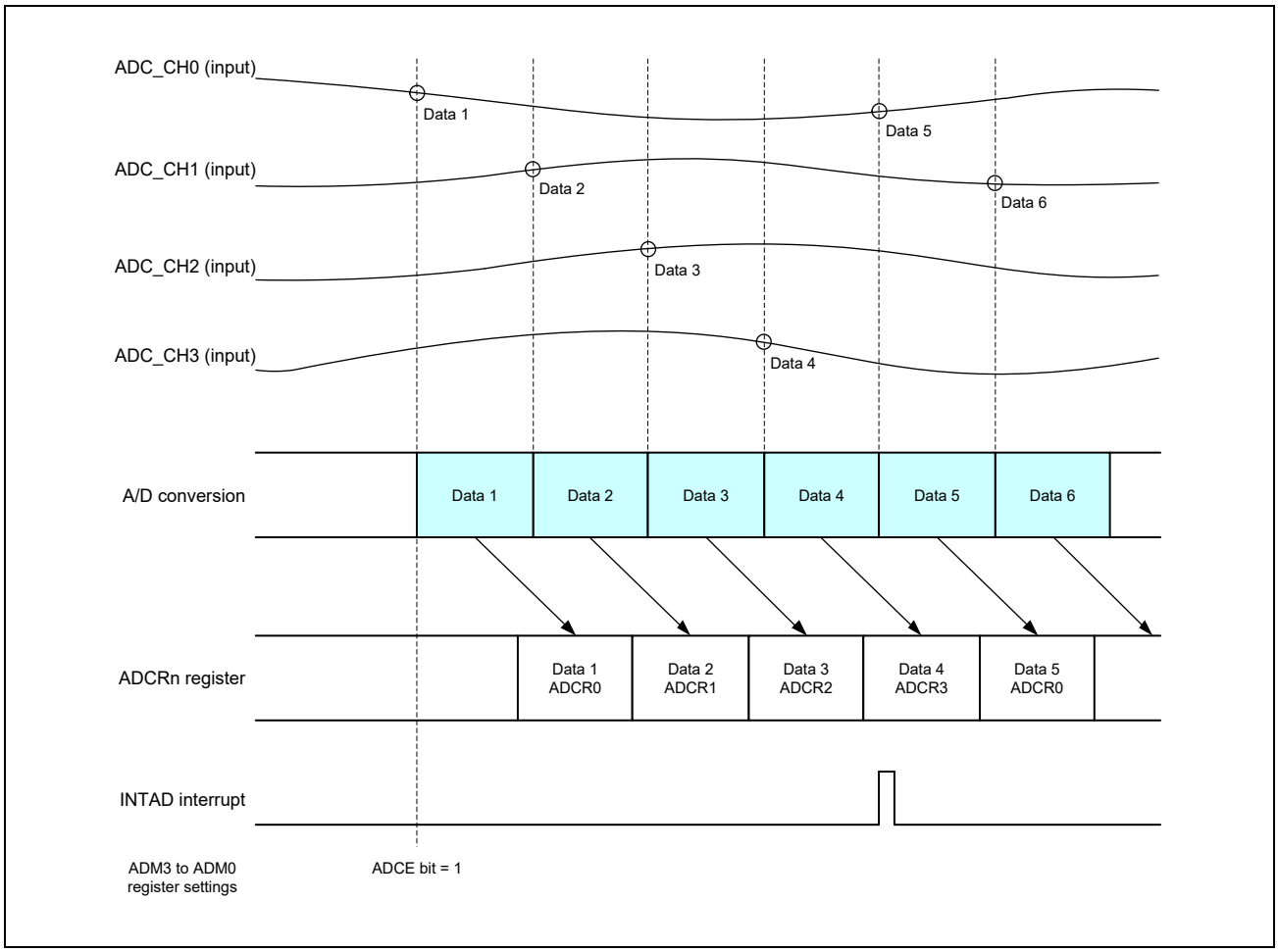


Figure 39.16 Example of A/D Conversion in Scan Mode with Repeat Mode Selected

1. The SRESB bit in the ADM0 register is set to 1 to release the A/D converter from the software reset state.
2. The A/D converter waits to be released from the reset state (wait time is 2 × ADC_ADCLK(TSUϕ)).
3. In the ADM3 to ADM0 registers, software trigger mode, scan mode, and repeat mode are selected and analog input pins ADC_CH3 to ADC_CH0 are set.
4. In the ADINT and ADSTS registers, the interrupt signal is set as a pulse and interrupt output of channel 3 is enabled.
5. Normal mode is selected by setting the PWDWNB bit in the ADM0 register to 1.
6. The A/D converter is kept waiting for a period equal to or longer than the stabilization wait time*1 by software.
7. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
8. A/D conversion proceeds on ADC_CH0 (A/D conversion result (data 1) is stored in the ADCR0 register).
9. A/D conversion proceeds on ADC_CH1 (A/D conversion result (data 2) is stored in the ADCR1 register).
10. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 3) is stored in the ADCR2 register).

11. A/D conversion proceeds on ADC_CH3 (A/D conversion result (data 4) is stored in the ADCR3 register).
12. An INTAD interrupt occurs.
13. Steps 8 to 12 above are repeated automatically.

Note 1. The stabilization wait time is 1 µs.

39.4.4.6 Example of A/D Conversion in Auto Mode

Figure 39.17 shows an example of A/D conversion with hardware trigger mode, auto mode, scan mode, and single mode are selected. In the example in **Figure 39.17**, conversion proceeds with ADC_CH3 to ADC_CH0 selected.

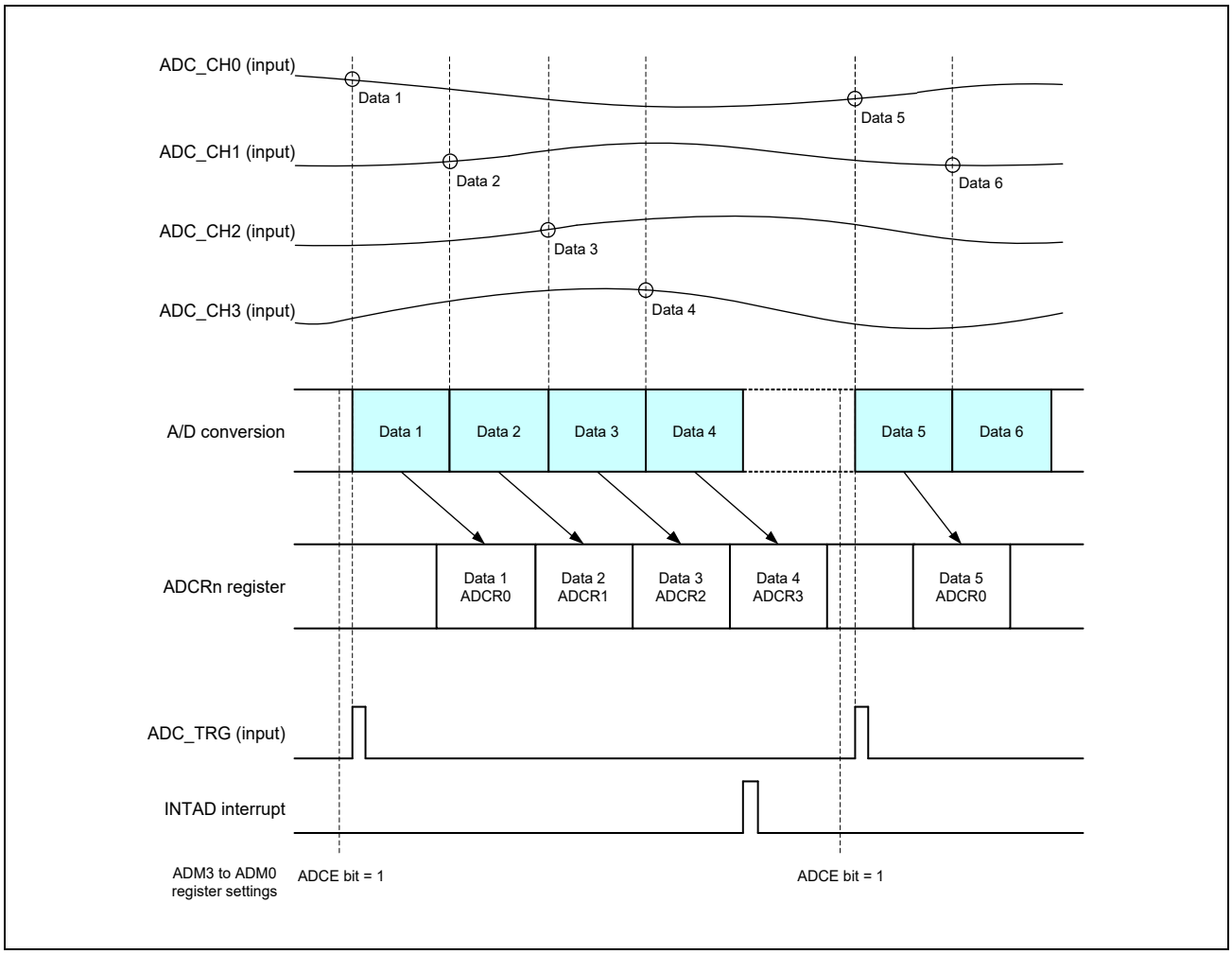


Figure 39.17 Example of A/D Conversion in Auto Mode

1. The SRESB bit in the ADM0 register is set to 1 to release the A/D converter from the software reset state.
2. The A/D converter waits to be released from the reset state (wait time is 2 × ADC_ADCLK(TSUϕ)).
3. In the ADM3 to ADM0 registers, hardware trigger mode, auto mode, scan mode, and single mode are selected, and analog input pins ADC_CH3 to ADC_CH0 are set, the trigger of the ADC_TRG pin is enabled, and rising-edge detection of the hardware trigger is set.
4. In the ADINT and ADSTS registers, the interrupt signal is set as a pulse and interrupt output of channel 3 is enabled.
5. Normal mode is selected by setting the PWDWNB bit in the ADM0 register to 1.
6. The A/D converter is kept waiting for a period equal to or longer than the stabilization wait time*[1] by software.
7. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).

8. After 250 ns that is the period it takes for accepting a trigger, a rising edge of the pulse is input to the ADC_TRG pin (A/D conversion is started).
9. A/D conversion proceeds on ADC_CH0 (A/D conversion result (data 1) is stored in the ADCR0 register).
10. A/D conversion proceeds on ADC_CH1 (A/D conversion result (data 2) is stored in the ADCR1 register).
11. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 3) is stored in the ADCR2 register).
12. A/D conversion proceeds on ADC_CH3 (A/D conversion result (data 4) is stored in the ADCR3 register).
13. An INTAD interrupt occurs.
14. Steps 7 to 13 above are repeated.

Note 1. The stabilization wait time is 1 µs.

39.4.4.7 Example of A/D Conversion in Step Mode

Figure 39.18 shows an example of A/D conversion with hardware trigger mode, step mode, scan mode, and single mode are selected. In the example in **Figure 39.18**, conversion proceeds with ADC_CH3 to ADC_CH0 selected.

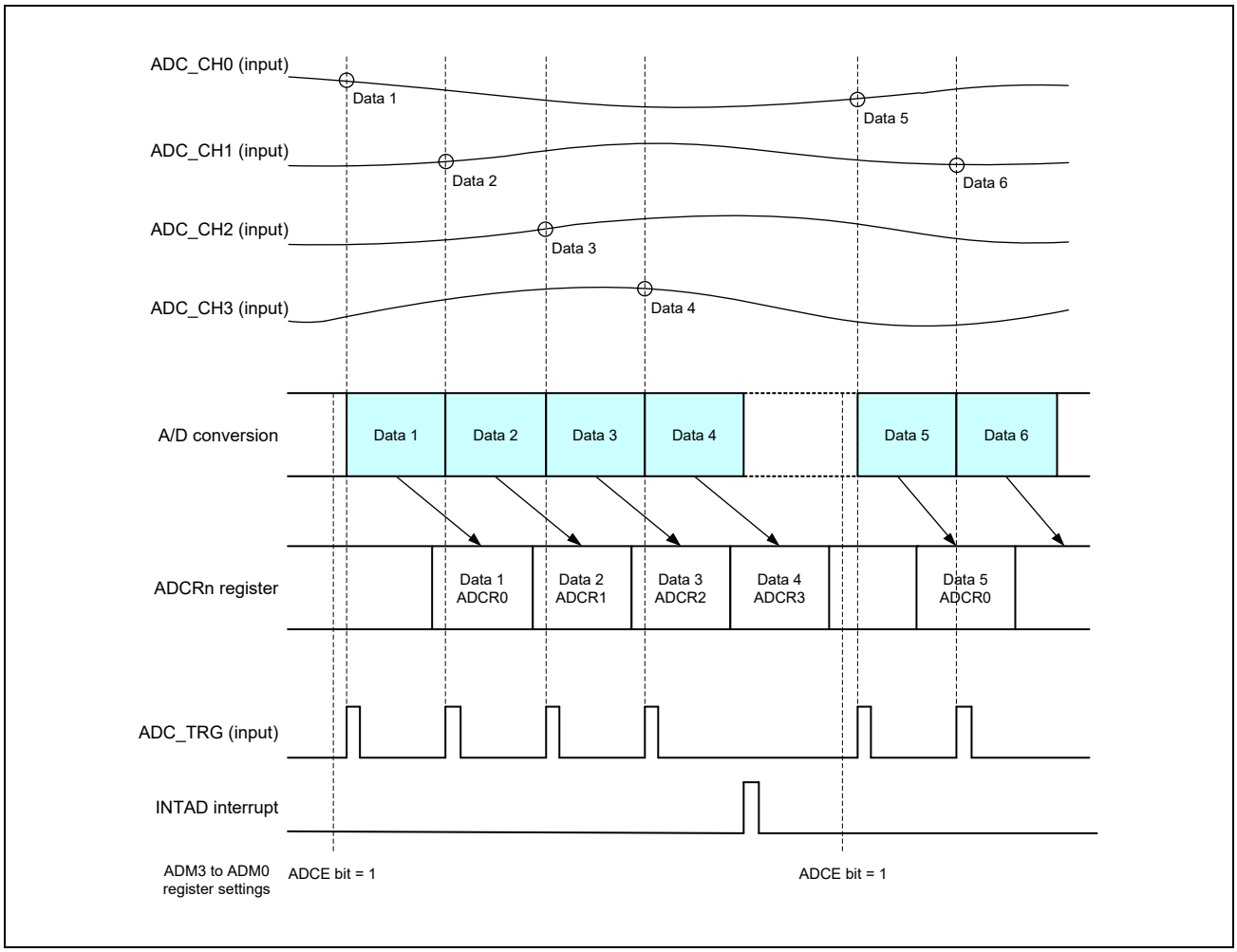


Figure 39.18 Example of A/D Conversion in Step Mode

1. The SRESB bit in the ADM0 register is set to 1 to release the A/D converter from the software reset state.
2. The A/D converter waits to be released from the reset state (wait time is 2 × ADC_ADCLK(TSUϕ)).
3. In the ADM3 to ADM0 registers, hardware trigger mode, step mode, scan mode, and single mode are selected, analog input pins ADC_CH3 to ADC_CH0 are set, the trigger of the ADC_TRG pin is enabled, and rising-edge detection of the hardware trigger is set.
4. In the ADINT and ADSTS registers, the interrupt signal is set as a pulse and interrupt output of channel 3 is enabled.
5. Normal mode is selected by setting the PWDWNB bit in the ADM0 register to 1.
6. The A/D converter is kept waiting for a period equal to or longer than the stabilization wait time*1 by software.
7. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1.
8. After 250 ns that is the period it takes for accepting a trigger, a rising edge of the pulse is input to the ADC_TRG pin (A/D conversion is started).

9. A/D conversion proceeds on ADC_CH0 (A/D conversion result (data 1) is stored in the ADCR0 register).
10. After 250 ns that is the period it takes for accepting a trigger, a rising edge of the pulse is input to the ADC_TRG pin (A/D conversion is started).
11. A/D conversion proceeds on ADC_CH1 (A/D conversion result (data 2) is stored in the ADCR1 register).
12. After 250 ns that is the period it takes for accepting a trigger, a rising edge of the pulse is input to the ADC_TRG pin (A/D conversion is started).
13. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 3) is stored in the ADCR2 register).
14. After 250 ns that is the period it takes for accepting a trigger, a rising edge of the pulse is input to the ADC_TRG pin (A/D conversion is started).
15. A/D conversion proceeds on ADC_CH3 (A/D conversion result (data 4) is stored in the ADCR3 register).
16. An INTAD interrupt occurs.
17. Steps 7 to 16 above are repeated.

Note 1. The stabilization wait time is 1 μs.

39.4.5 TSU Conversion

ADC convert the temperature that TSU IP output to 12bit data. There are two conversion methods: one is to set the TSUMODE register and the other is to convert with normal ADC operation. Start/stop the TSU every time you measure temperature or always keep the TSU on. (See chapter TSU for TSU control)

39.4.5.1 TSUMODE register

1. Stop an operation of ADC. (See **Section 39.4.3.2, Procedure for Stopping A/D Conversion**)
   If the temperature sensor is not operating, start control from the temperature sensor register.
2. Set TSUMODE.TSUEN to 1b.
3. Set ADM3.ADSMP[7:0] to H'FF.
4. Set ADM0.ADCE to 1b to start ADC operation.
   The temperature data is converted every 3 µs, and 12-bit conversion data is stored in ADCR8 register.
5. Read data from ADCR8 register. Calculate the temperature with Tj correction. (See the section "Procedure for Measuring the Temperature" in chapter TSU)
6. After temperature calculation, set ADM0.ADCE to 0b to stop ADC operation.
7. Set TSUMODE.TSUEN to 0b.
   In the case of stopping the temperature sensor, stop it with temperature sensor register.
8. Restore ADM3.ADSMP[7:0] to the value before TSU conversion.
9. When ADM0.ADCE is set to 1b, operation resumes with the settings before TSU conversion.

39.4.5.2 ADM2.CHSEL8 selection

If ADM2.CHSEL8 is selected, the dedicated channel for TSU is A/D converted like other channels. Start the temperature sensor before measurement. If ADM2.CHSEL8 is selected, set ADM3.ADSMP[7:0] to H'FF to extend the sampling time.

When converting in scan mode, not only TSU but also the conversion time of other channels are extended together. When setting to 1buffer mode, the conversion result is stored in the ADCR8 register.

When setting to 4buffer mode, the conversion result is stored in ADCR8, ADCR9, ADCR10 and ADCR11 registers.

Read the data from the register, perform Tj correction and calculate the temperature. (See the section "Procedure for Measuring the Temperature" in chapter TSU)

39.5 Notes

39.5.1 Interval of Hardware Trigger

The interval (interval of input time) of a trigger in hardware trigger mode must be longer than the A/D conversion time. For the A/D conversion time, see **Section 39.3.4, A/D Converter Mode Register 3 (ADM3)**.

39.5.2 Timing Restrictions

This A/D converter has restrictions on the timing interval, which are caused by the asynchronism countermeasure circuit. The restrictions shown in **Table 39.9** need to be met. If an access is made in a period shorter than this interval, the register values on the ADC_PCLK (P0ϕ) side are not reflected in the registers on the ADC_ADCLK (TSUϕ) side because the notification of register update does not reach the ADC_ADCLK (TSUϕ) side.

Table 39.9 Restrictions on Timing Interval

Condition	Restriction
Writing to the same register successively	(Interval of write access) ≥ (6 × ADC_PCLK(P0ϕ) + 6 × ADC_ADCLK(TSUϕ))
Interval of A/D conversion	(A/D conversion time) ≥ (6 × ADC_PCLK(P0ϕ) + 6 × ADC_ADCLK(TSUϕ))

Note: For the A/D conversion time, see **Figure 39.2**.

39.5.3 Restrictions on ADC_PCLK(P0ϕ) and ADC_ADCLK(TSUϕ) Cycles

The ADC_PCLK(P0ϕ) and ADC_ADCLK(TSUϕ) cycles need to satisfy the relation in the following formula.

$$\frac{\text{ADC\_ADCLK(TSU}\phi\text{)} \leq \text{(A/D conversion time)}}{6 - \text{ADC\_PCLK(P0}\phi\text{)}}$$

Note: For the A/D conversion time, see **Figure 39.2**.

39.5.4 Operation when Stopping or Restarting A/D Conversion

When 0 is written to the ADCE bit in the ADM0 register, the A/D converter stops A/D conversion. A/D conversion is restarted by writing 0 to the ADCE bit in the ADM0 register once and then writing 1 to the same bit. Even though there was an instruction to stop A/D conversion, A/D conversion may not stop immediately. **Table 39.10** shows the operation for stopping conversion and also the operation for restarting conversion after it has been stopped.

Table 39.10 Operation when Stopping or Restarting A/D Conversion with the ADCE Bit

A/D Conversion Modes					Timing for Stopping Conversion when Stop is Instructed with ADCE = 0	Operation for Restarting Conversion
Trigger		Operating Mode	Conversion Count	Buffer Count		
Mode	Input					
Software	—	Select	Single	1	Stopped on completion of 1-time conversion.	Conversion proceeds on the channel selected in the ADM2 register.
				4	Stopped on completion of conversion in which there was an instruction to stop in the middle of 4-time conversion.	Same as above
			Repeat	1	Stopped on completion of 1-time conversion.	Same as above
				4	Stopped on completion of conversion in which there was an instruction to stop in the middle of 4-time conversion.	Same as above
		Scan	Single	1	Stopped on completion of conversion in which there was an instruction to stop in the middle of scanning.	Conversion proceeds from the smallest channel number selected in the ADM2 register.
				4	(Setting prohibited)	(Setting prohibited)
			Repeat	1	Stopped on completion of conversion in which there was an instruction to stop in the middle of scanning.	Conversion proceeds from the smallest channel number selected in the ADM2 register.
				4	(Setting prohibited)	(Setting prohibited)
Hardware	Auto mode	Select	Single	1	Same as software trigger mode	Same as software trigger mode
				4	Same as above	Same as above
			Repeat	1	Same as above	Same as above
				4	Same as above	Same as above
		Scan	Single	1	Same as above	Same as above
				4	Same as above	Same as above
			Repeat	1	Same as above	Same as above
				4	Same as above	Same as above
	Step mode	Select	Single	1	Same as software trigger mode	Same as software trigger mode
				4	Same as above	Same as above
			Repeat	1	Same as above	Same as above
				4	Same as above	Same as above
		Scan	Single	1	Same as above	Same as above
				4	Same as above	Same as above
			Repeat	1	Same as above	Same as above
				4	Same as above	Same as above

40. Thermal Sensor Unit (TSU)

This LSI chip incorporates a thermal sensor unit (TSU) that measures the temperature inside the LSI.

40.1 Overview

40.1.1 Features

- This LSI has a single TSU.
- The thermal sensor in this unit measures temperatures in the range from −40°C to 125°C with an accuracy of ±7°C.
- The analog output of TSU is connected to channel 8 of ADC. By controlling ADC, it converts the analog output with 12bit digital data and outputs the result. (See **Section 39, A/D Converter**.)

40.1.2 Block Diagram

Figure 40.1 is a block diagram of the TSU.

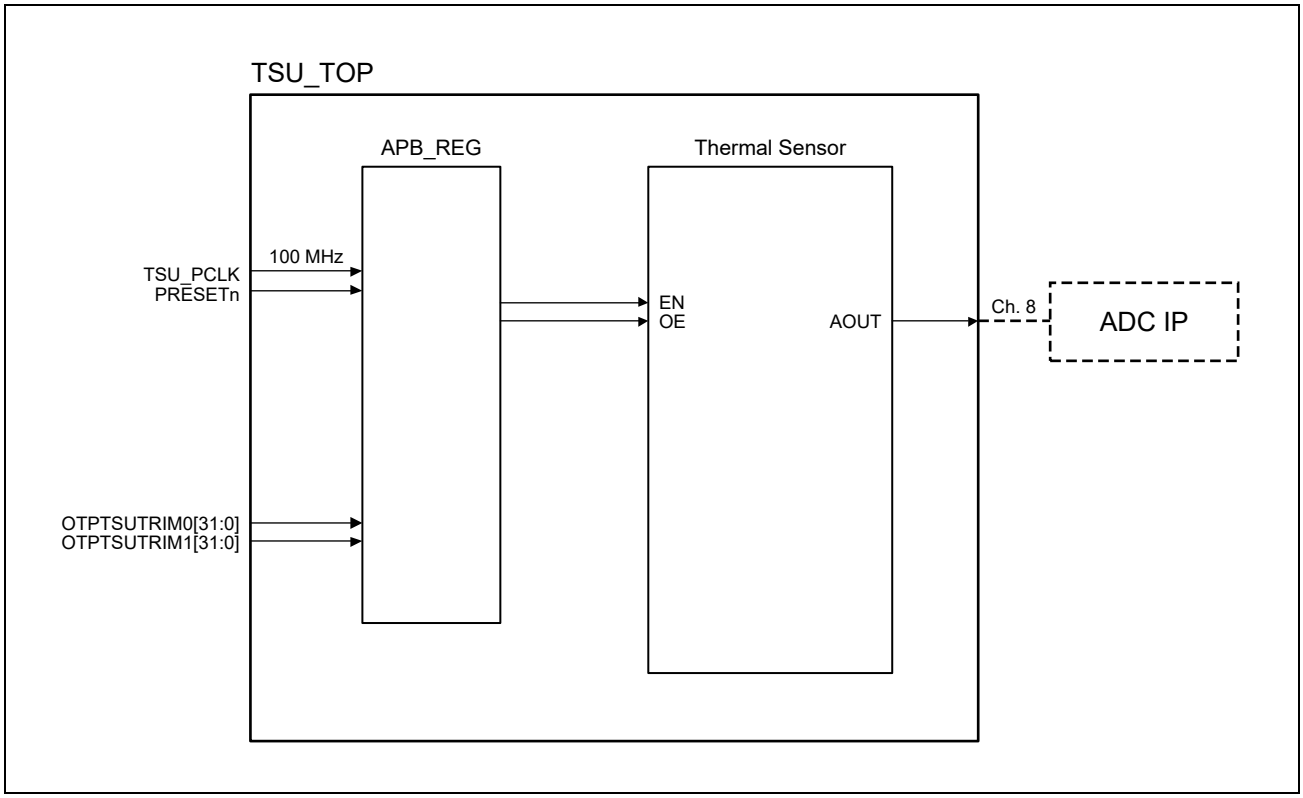


Figure 40.1 Block Diagram of the TSU

The TSU consists of the following blocks.

- APB_REG
- Thermal Sensor

40.1.2.1 APB_REG

- The APB_REG block provides an external interface for the TSU.
- This block contains control and status registers for the thermal sensor.

40.1.2.2 Thermal Sensor

- The thermal sensor block includes the internal probe, A/D converter and other items.
- The analog output of TSU is connected to channel 8 of ADC. By controlling ADC, it converts the analog output with 12bit digital data.

40.1.3 External Pins

The TSU has no external pins.

40.1.4 Interrupts

The TSU has no interrupts.

40.2 Register Configuration

This section describes the registers for use in controlling the TSU.

Base Address: H'0_1005_9000 (Cortex-A55 Address Space)
Base Address: H'5005_9000 (Cortex-M33 Address Space Non-Secure)
Base Address: H'4005_9000 (Cortex-M33 Address Space Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

40.2.1 List of Register

Table 40.1 lists the TSU registers. All registers are accessible in 32-bit units.

Table 40.1 List of the TSU Registers

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
Sensor Mode Register	TSU_SM	R/W	H'0000_0000	H'00	32
TSU OTP Calibration Register 0	OTPTSUTRIM0_REG	R	—	H'18	32
TSU OTP Calibration Register 1	OTPTSUTRIM1_REG	R	—	H'1C	32

40.3 Register Descriptions

40.3.1 Sensor Mode Register (TSU_SM)

This register is used to specify the operating mode of the thermal sensor.

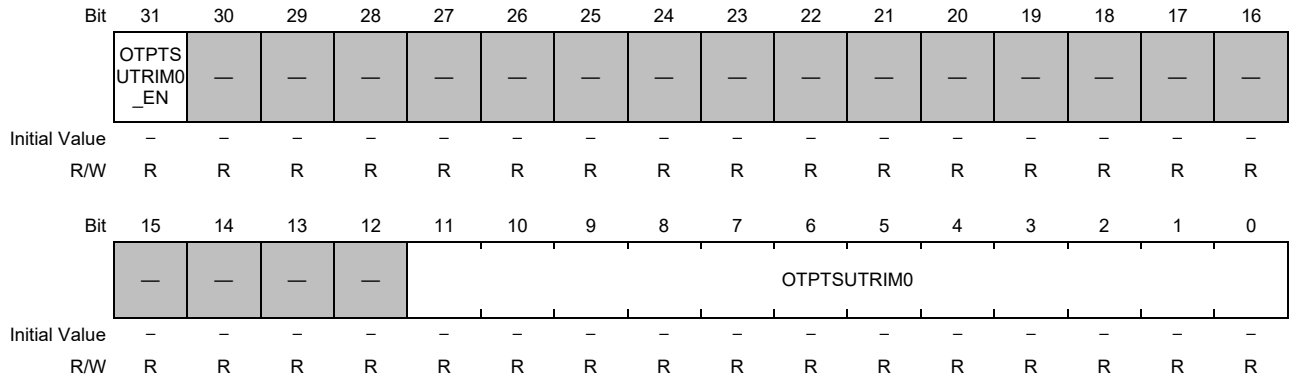
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OE	EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	OE	0b	R/W	TS IP Output Enable 1b: output enable 0b: output disable
0	EN	0b	R/W	TS IP Enable 1b: Thermal sensor enable 0b: Thermal sensor disable

40.3.2 TSU OTP Calibration Register 0 (OTPTSUTRIM0_REG)

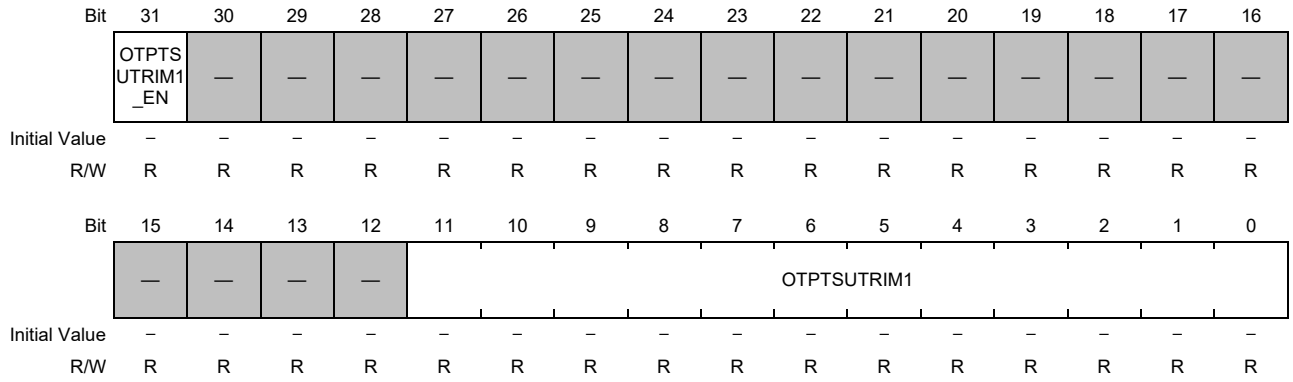
This register holds the correction value for calibration to be used in calculating the temperature.



Bit	Bit Name	Initial Value	R/W	Description
31	OTPTSUTRIM0_EN	—	R	This bit indicates whether [11:0] OTPTSUTRIM0 is valid or invalid. 0b: [11:0] OTPTSUTRIM0 is invalid. 1b: [11:0] OTPTSUTRIM0 is valid.
30 to 12	—	—	R	Reserved When read, the value is indefinite value. The written value will be ignored.
11 to 0	OTPTSUTRIM0	—	R	These bits hold the correction value for calibration to be used in calculating the temperature.

40.3.3 TSU OTP Calibration Register 1 (OTPTSUTRIM1_REG)

This register holds the correction value for calibration to be used in calculating the temperature.



Bit	Bit Name	Initial Value	R/W	Description
31	OTPTSUTRIM1_EN	—	R	This bit indicates whether [11:0] OTPTSUTRIM1 is valid or invalid. 0b: [11:0] OTPTSUTRIM1 is invalid. 1b: [11:0] OTPTSUTRIM1 is valid.
30 to 12	—	—	R	Reserved When read, the value is indefinite value. The written value will be ignored.
11 to 0	OTPTSUTRIM1	—	R	These bits hold the correction value for calibration to be used in calculating the temperature.

40.4 Operation

40.4.1 Procedure for Starting the TSU

This section describes the procedure for starting the TSU.

1. Set the EN bit in the TSU_SM register to 1b to place the thermal sensor in the normal operating mode.
2. Wait for 30 µs or more, set the OE bit in the TSU_SM register to 1b to enable the output.
3. Wait for 50 µs or more to stable the output level.
4. ADC to convert 12bit digital data.
5. Read converted data from ADC register.

40.4.2 Procedure for Stopping the TSU

This section describes the procedure for stopping the TSU.

1. Set the EN and OE bits in the TSU_SM register to 0 at the same time to place the thermal sensor in the standby mode.

40.4.3 Setting A/D Converter

See **Section 39, A/D Converter** for more information.

40.4.4 Procedure for Measuring the Temperature

To calculate the temperature Tj, read registers ADCR of ADC, OTPTSUTRIM0_REG, and OTPTSUTRIM1_REG of TSU, and follow the procedure below.

1. Read the value of the ADCR of ADC register eight times every at least 3 µs. The read values are defined as TSCode[0] to TSCode[7].

2. Calculate the average of the values read, TSCodeAVE, by using the following formula.

$$\mathrm{TSCodeAVE} = \frac{(\mathrm{TSCode}[0] + \cdots + \mathrm{TSCode}[7])}{8}$$

3. Read the correction values for use in calibration from registers OTPTSUTRIM0_REG and OTPTSUTRIM1_REG and use the following formula to calculate Tj.

$$\mathrm{Tj} = (\mathrm{TSCodeAVE} - \mathrm{OTPTSUTRIM1}) \times \left(\frac{165}{(\mathrm{OTPTSUTRIM0} - \mathrm{OTPTSUTRIM1})}\right) - 40$$

Note: If the ADC_AVDD18 is not 1.80 V on board, use the following formula to calculate the corrected TSCode.
Corrected TSCode = (ADC_AVDD18 voltage / 1.80) × TSCodeAve

$$\mathrm{Tj} = (\mathrm{Corrected\ TSCode} - \mathrm{OTPTSUTRIM1}) \times \left(\frac{165}{(\mathrm{OTPTSUTRIM0} - \mathrm{OTPTSUTRIM1})}\right) - 40$$

40.5 Usage Note

- Do not set OE to 1 during EN = 0.

41. Low Power Consumption

This LSI has several functions to achieve low power consumption. By combining with power supply isolation, further power consumption can be reduced.

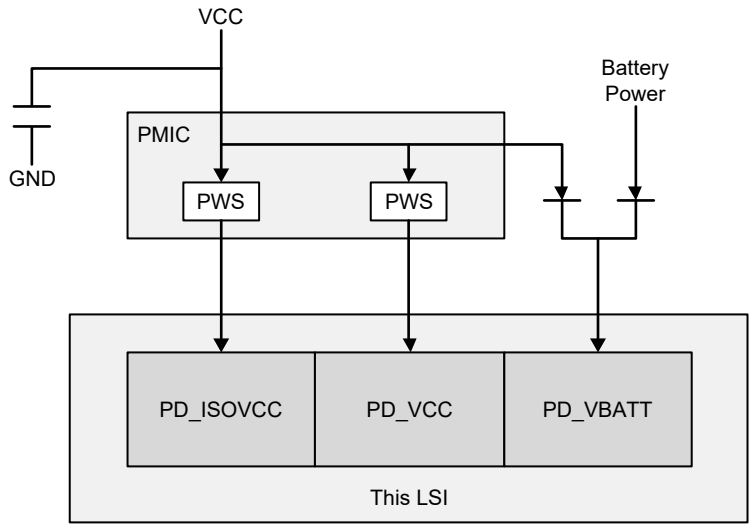
41.1 Overview

41.1.1 Power Domain

A power domain of this LSI is divided into PD_VCC domain, PD_ISOVCC domain and PD_VBATT domain.

- PD_VCC domain always powered on area.
- PD_ISOVCC domain is the area where the power can be turned off.
- PD_VBATT domain is the area where the RTC/backup register is located, works on battery power when the power of PD_VCC and PD_ISOVCC domain are turned off.

The following figure shows one example relationship of the power domain and power supply.



Note: PMIC: Power Management IC
PWS: Power Switch

Figure 41.1 Power Domain and Power Supply

The **Table 41.1** shows the relationship of the power domain and power supply pins.

The power-on/power-off sequence should be followed to **Section 47.3, Power-On/Power-Off Sequence**.

Table 41.1 Power Domain to which Power Supply Pins Belong

Group	Pin Name	Power Domain
G1	VBATT_VDD	PD_VBATT
G2	VDD	PD_VCC
	PLL16_AVDD	
	PLL23_AVDD	
	PLL4_AVDD	
G3	PVDD18	
	ADC_AVDD18	
	OTP_AVDD18	
G4	PVDD33	
	JTAG_PVDD	
	XSPI_PVDD	
	I3C_PVDD	
G5	VDD_ISO	PD_ISOVCC
	PCIE_VDD09	
G6	SDn_PVDD (n = 0, 1)	
	PVDD182533_n (n = 0, 1)	
G7	USB_AVDD18	
	USB_VDD18	
	PCIE_VDD18	
G8	USB_VDD33	
G9	DDR_VAA	
G10	DDR_VDDQ	

The following figure shows the power domain to which each module belongs.

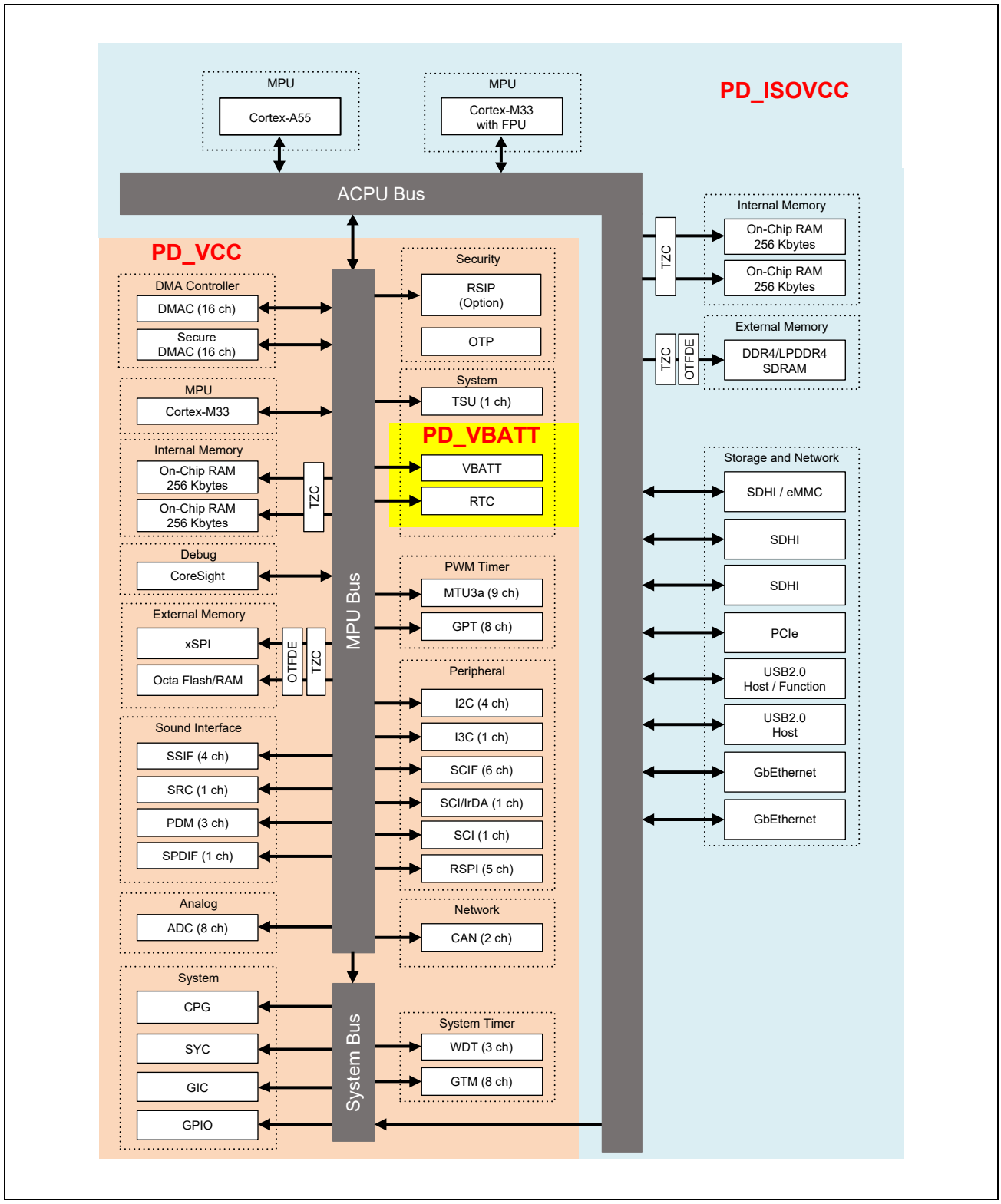


Figure 41.2 Modules in Power Domain

41.1.2 Power mode

This LSI has the following four power modes depending on the power on/off combination of the power domain.

Table 41.2 Power Mode

Power Mode	Power Domain		
	PD_ISOVCC	PD_VCC	PD_VBATT
ALL_ON	ON	ON	ON
AWO	OFF	ON	ON
VBATT	OFF	OFF	ON
ALL_OFF	OFF	OFF	OFF

Note: • ALL_OFF mode: All CPUs and peripheral modules can not be worked.
• ALL_ON mode: All CPUs and peripheral modules can be worked.
• AWO mode: Cortex-M33 and peripheral modules in PD_VCC and PD_VBATT domain can be worked.
• VBATT mode: Only RTC, tamper detection and backup registers can be worked.

The following figure shows the transition of the power mode. Regarding to the operation for the power mode transition, refer to **Section 41.6, Power Mode Transition**.

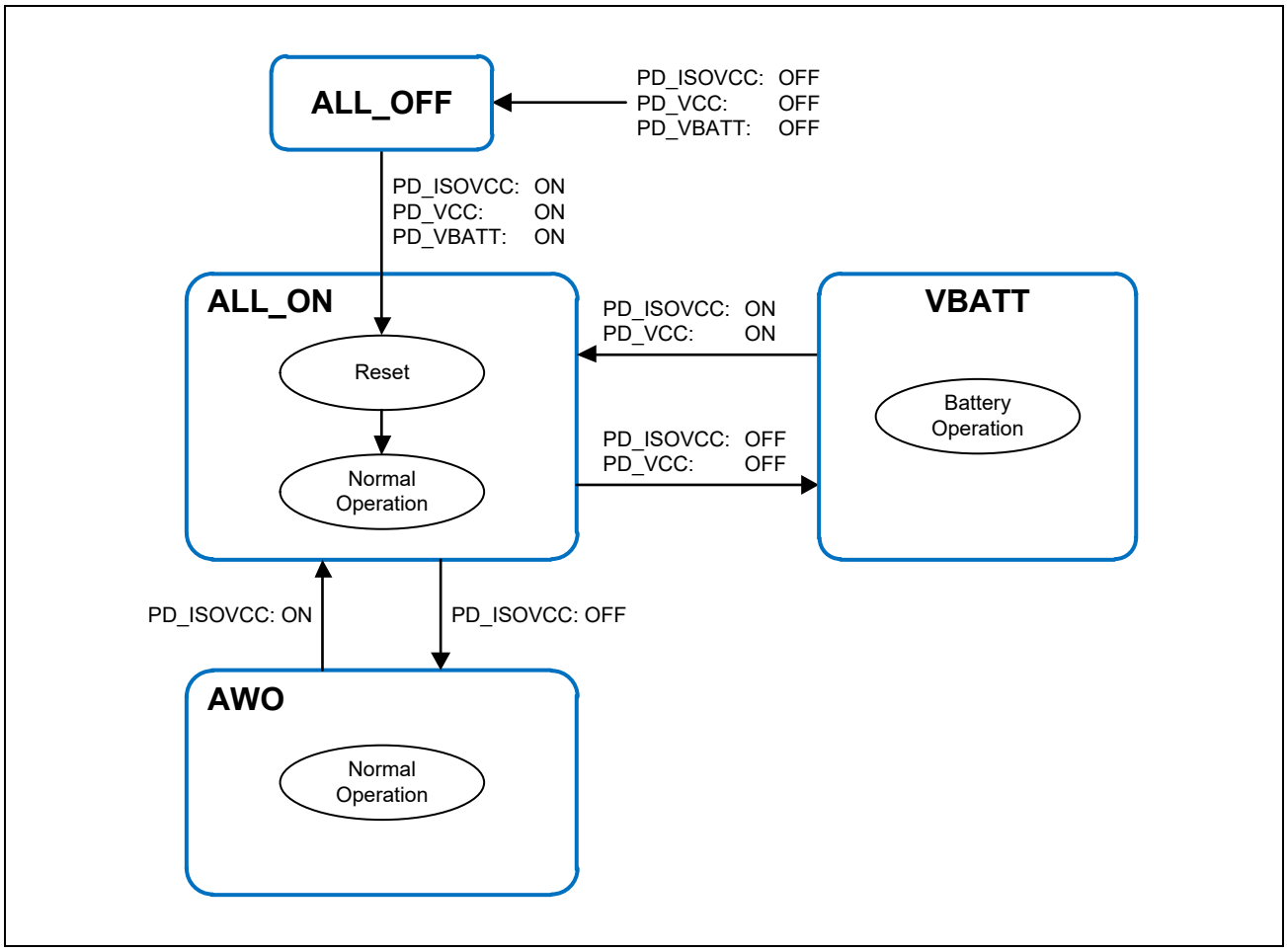


Figure 41.3 Power Mode Transition

41.1.3 Low Power Function

This LSI has the following low power function.

■ Module Standby Mode

This mode stops the clock of specific peripheral modules except CPU to reduce the power consumption.

Stop the clock by setting the CPG register.

Refer to **Section 41.2, Module Standby Mode**.

■ Low Frequency Mode

This mode slows down the clock of the peripheral modules which synchronized to the system bus clock, and CPU to reduce the power consumption.

Change the clock frequency by setting the CPG register.

Refer to **Section 41.3, Low Frequency Mode**.

■ Cortex-A55 Sleep Mode

This mode puts Cortex-A55 into a low power state to reduce the power consumption.

Put into a low power state by issuing a WFI instruction.

The return factor is interrupt.

Refer to **Section 41.4, Cortex-A55 Sleep Mode**.

■ Cortex-M33/Cortex-M33_FPU Sleep Mode

This mode puts Cortex-M33/Cortex-M33_FPU into a low power state to reduce the power consumption.

Put into a low power state (Deep Sleep) by issuing a WFI instruction.

The return factor is interrupt and SysTick Timer.

Refer to **Section 41.5, Cortex-M33/Cortex-M33_FPU Sleep Mode**.

41.2 Module Standby Mode

41.2.1 Overview

The Module Standby Mode is a mode that requests the clock stop of the module specified by the master.

The purpose of this mode is to reduce power consumption by stopping unnecessary functions.

The master which can request the clock stop is Cortex-A55 and Cortex-M33/Cortex-M33_FPU. So, Cortex-A55 and Cortex-M33/Cortex-M33_FPU cannot be placed in the Module Standby Mode. To reduce the power consumption in the CPUs, use the CPU sleep modes to be described later.

This mode is implemented by stopping the clock supply by the CPG register and switching the MSTOP signal of the bus through CPG register settings.

If the master accesses a module that has the clock stopped and the MSTOP bit set, a bus error will occur.

The following registers are used in the Module Standby Mode.

Table 41.3 Module Standby Mode Related Registers

Register Name	Abbreviation	Description
Clock Control Register	CPG_CLKON_***	The detail information of each register is described in **Section 6, System Controller (SYSC)** and **Section 7, Clock Pulse Generator (CPG)**. The usage of these registers is described in **Section 41.2.2, Operation**.
MSTOP Register	CPG_BUS_***_MSTOP CPG_MHU_STOP	

The following figure shows the block connection overview related Module Standby Mode.

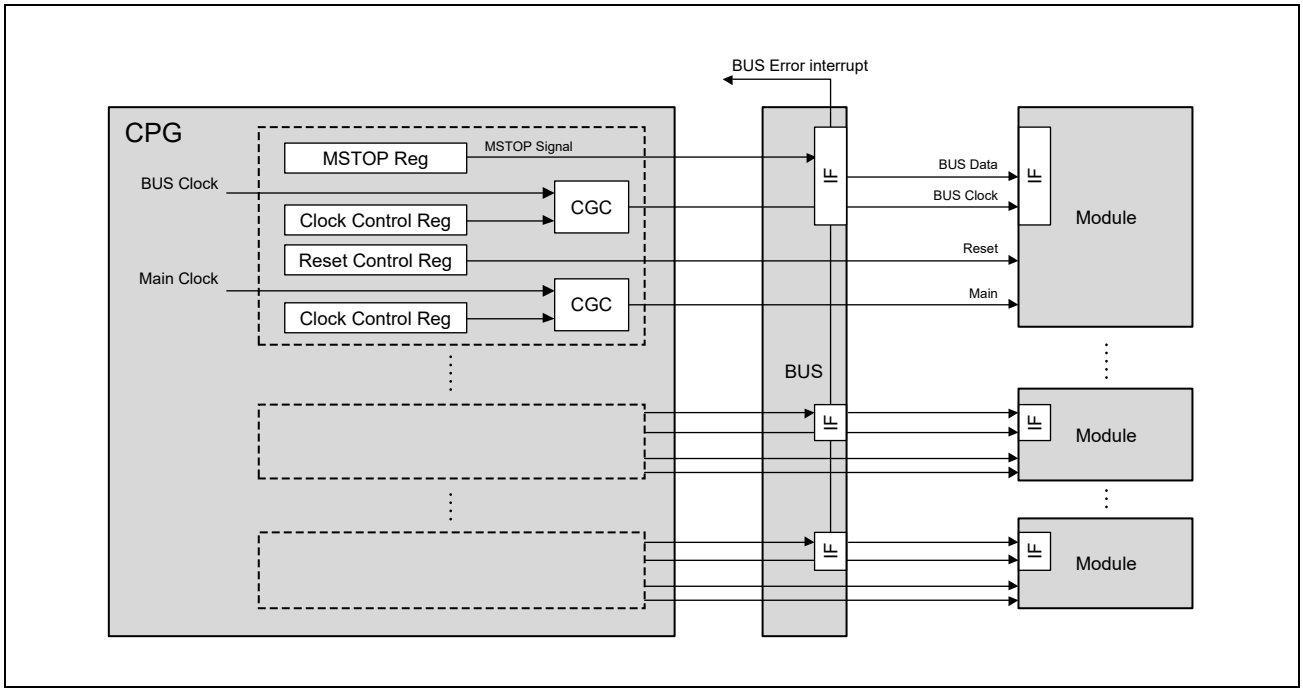


Figure 41.4 Block Connection Overview for Module Standby Mode

41.2.2 Operation

The following chart shows the procedure to enter and return from the Module Standby Mode.

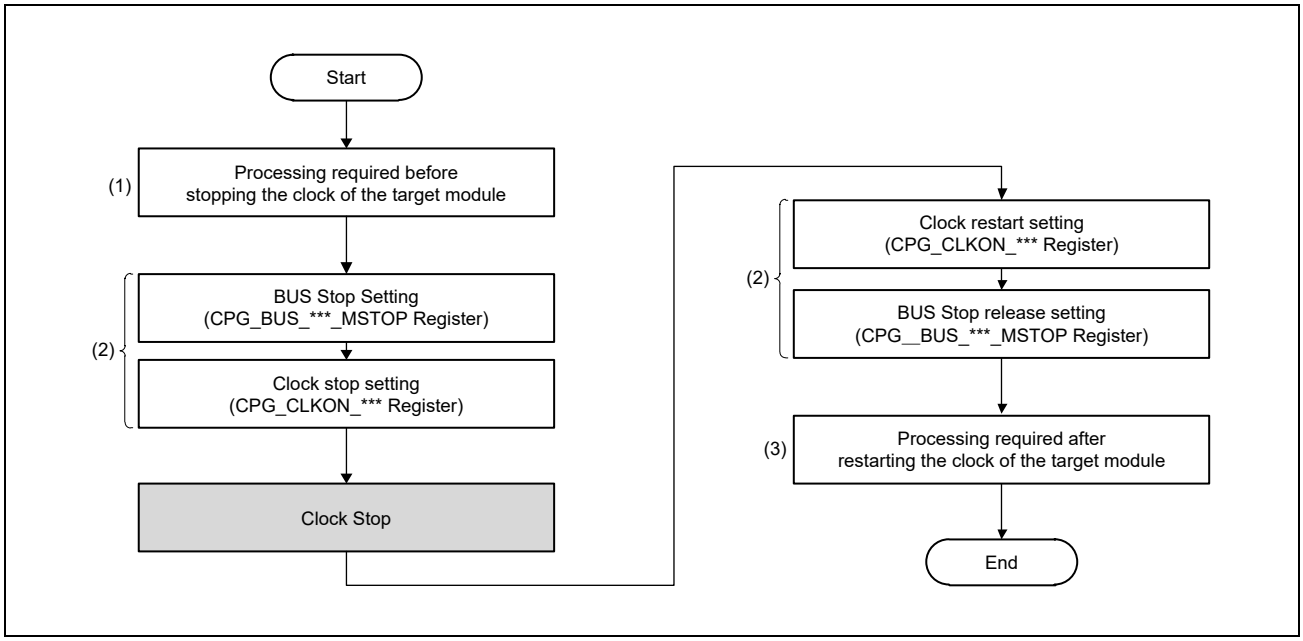


Figure 41.5 Module Standby Mode Procedure

(1) Depend on the module, there may be a request to stop during communication and various processing. In this case, if the managed CPU can be stopped unilaterally for each module, a stop request is immediately issued to the CPG. For those that cannot be stopped until a certain process is completed, after confirming that the managed CPU has stopped for each module, a stop request is made to the CPG.

(2) Refer to **Section 41.2.2.1, Setting of Clock Control and MSTOP Register** regarding to the setting of Clock Control Register and MSTOP Register.

(3) Whether the reset control is required depends on the module; check the specifications of the target module. Set the reset control register to 0 to enter the reset state. Set 1 for the release from the reset state.

41.2.2.1 Setting of Clock Control and MSTOP Register

The following table shows the combination of the Clock Control register and the MSTOP register to stop the clock of the target module.

The Module Standby Mode can be applied only to the modules shown in the table below.

- When entering the Module Standby Mode:

  Set the MSTOP Register bit of the target module to 1.
  Set the Clock Control Register bit of the target module to 0.

- When exiting the Module Standby Mode:

  Set the MSTOP Register bit of the target module to 0.
  Set the Clock Control Register bit of the target module to 1.

Table 41.4 Registers for Module Standby Mode (1/2)

Target Module for Module Standby Mode	Clock Control Register		MSTOP Register	
	Register Name (CPG_CLKON_***)	bit	Register Name (CPG_BUS_***_MSTOP)	bit
GIC	GIC600	[0]	ACPU	[3]
IA55	IA55	[1:0]	PERI_CPU	[13]
IM33	IM33	[1:0]	PERI_CPU	[14]
IM33_FPU	IM33	[9:8]	PERI_CPU2	[1]
MHU	MHU	[0]	CPG_MHU_STOP*1	[0]
CoreSight	CST	[13:0]	PERI_CPU	[2]
SYC	SYC	[0]	PERI_CPU	[3]
DMAC	DMAC_REG	[1:0]	REG1	[3:0]
GTM ch0	GTM	[0]	REG0	[4]
GTM ch1		[1]		[5]
GTM ch2		[2]		[6]
GTM ch3		[3]		[7]
GTM ch4		[4]		[8]
GTM ch5		[5]		[9]
GTM ch6		[6]		[10]
GTM ch7		[7]		[11]
MTU	MTU	[0]	MCPU1	[2]
POE3	POE3	[0]	MCPU1	[9]
GPT	GPT	[0]	MCPU1	[4]
POEGA	POEG	[0]	MCPU1	[5]
POEGB		[1]		[6]
POEGC		[2]		[7]
POEGD		[3]		[8]
WDT ch0	WDT	[1:0]	REG0	[0]
WDT ch1		[3:2]		[1]
WDT ch2		[5:4]		[2]
xSPI	SPI	[3:0]	PERI_CPU2	[3]

Table 41.4 Registers for Module Standby Mode (2/2)

Target Module for Module Standby Mode	Clock Control Register		MSTOP Register	
	Register Name (CPG_CLKON_***)	bit	Register Name (CPG_BUS_***_MSTOP)	bit
SDHI ch0	SDHI	[3:0]	PERI_COM	[0]
SDHI ch1		[7:4]		[1]
SDHI ch2		[11:8]		[11]
SSI ch0	SSI	[1:0]	MCPU1	[10]
SSI ch1		[3:2]		[11]
SSI ch2		[5:4]		[12]
SSI ch3		[7:6]		[13]
SRC	SRC	[0]	MCPU1	[3]
USB2.0 ch0	USB	[0], [2]	PERI_COM	[6:5]
USB2.0 ch1		[1]		[7]
USB2.0 PHY*2		[3]		[4]
ETHER ch0	ETH	[0]	PERI_COM	[2]
ETHER ch1		[1]		[3]
I2C ch0	I2C	[0]	MCPU2	[10]
I2C ch1		[1]		[11]
I2C ch2		[2]		[12]
I2C ch3		[3]		[13]
SCIF ch0	SCIF	[0]	MCPU2	[1]
SCIF ch1		[1]		[2]
SCIF ch2		[2]		[3]
SCIF ch3		[3]		[4]
SCIF ch4		[4]		[5]
SCIF ch5		[5]	MCPU3	[4]
SCI ch0	SCI	[0]	MCPU2	[7]
SCI ch1		[1]		[8]
IRDA	IRDA	[0]	MCPU2	[6]
RSPI ch0	RSPI	[0]	MCPU1	[14]
RSPI ch1		[1]		[15]
RSPI ch2		[2]	MCPU2	[0]
RSPI ch3		[3]	MCPU3	[2]
RSPI ch4		[4]		[3]
CANFD	CANFD	[1:0]	MCPU2	[9]
ADC	ADC	[1:0]	MCPU2	[14]
TSU	TSU	[0]	MCPU2	[15]
OCTA	OCTA	[1:0]	MCPU3	[0]
PDM	PDM	[1:0]	MCPU3	[9]
PCI	PCI	[1:0]	PERI_COM	[10]
SPDIF	SPDIF	[0]	MCPU3	[6]
I3C	I3C	[1:0]	MCPU3	[10]
VBAT	VBAT	[0]	MCPU3	[8]

Note 1. CPG_MHU_STOP itself is the register name.

Note 2. The clock for USB2.0 PHY can only be stopped when the clock for USB2.0 ch0 and ch1 are stopped.

41.3 Low Frequency Mode

41.3.1 Overview

This mode reduces the power consumption by reducing the frequency of the clock signal supplied to the CPU and the specified peripheral module which synchronized to the system bus. Refer to **Table 41.5**.

Table 41.5 Clocks and Registers Related to Low Frequency Mode

Clock Symbol	CPG Register	Division Ratio	Related Clock*1
Iϕ	CPG_PL1_DDIV[1:0]	1/1, 1/2, 1/4, 1/8	Cortex-A55 main clock
I2ϕ	CPG_PL6_DDIV[2:0]	1/1, 1/2, 1/4, 1/8, 1/32	Coetex-M33 main clock
I3ϕ	CPG_PL6_DDIV[6:4]	1/1, 1/2, 1/4, 1/8, 1/32	Coetex-M33_FPU main clock
P0ϕ	CPG_PL2_DDIV[6:4]	1/1, 1/2, 1/4, 1/8, 1/32	Peripheral modules and related system bus clock
P1ϕ	CPG_PL3_DDIV[2:0]	1/1, 1/2, 1/4, 1/8, 1/32	Peripheral modules and related system bus clock
P2ϕ	CPG_PL3_DDIV[6:4]	1/1, 1/2, 1/4, 1/8, 1/32	Peripheral modules and related system bus clock
P3ϕ	CPG_PL3_DDIV[10:8]	1/1, 1/2, 1/4, 1/8, 1/32	Peripheral modules and related system bus clock

Note 1. For more detailed information, refer to another file for the clock list.

41.3.2 Operation

It is achieved by having software control registers of the CPG to switch the frequency divisors for the various domains. Regarding to the operation of the register, refer to **Section 7.4.5, Procedure for Switching the Division Ratio of the Dynamic Switching Frequency Dividers**.

41.4 Cortex-A55 Sleep Mode

41.4.1 Overview

Refer to **Section 2.3.1, Cortex-A55 Sleep Mode** for detail.

The following table shows the registers related to the Cortex-A55 Sleep Mode.

Refer to **Section 6, System Controller (SYSC)** for detail.

Table 41.6 Cortex-A55 Sleep Mode Related Registers

Register Name	Abbreviation	Bit Name
Lowpower Sequence Control Register1	SYS_LP_CTL1	bit[24]: CA55SLEEP_ACK bit[8]: CA55SLEEP_REQ
Lowpower Sequence Control Register2	SYS_LP_CTL2	bit[0]: CA55_STBYCTL
Lowpower Sequence Control Register5	SYS_LP_CTL5	bit[8]: CA55SLEEP0_F
Lowpower Sequence Control Register6	SYS_LP_CTL6	bit[8]: CA55SLEEP0_E

41.4.2 Operation

This section shows an example procedure.

The following figure is one example how to use the SYSC registers. The SYS_LP_CTL1, the SYS_LP_CTL5 and the SYS_CTL6 registers can be used to share the transition state of Cortex-A55 with Cortex-M33. Referring to these registers, Cortex-M33 can perform processing if need when Cortex-A55 enters and exits the Cortex-A55 Sleep Mode.

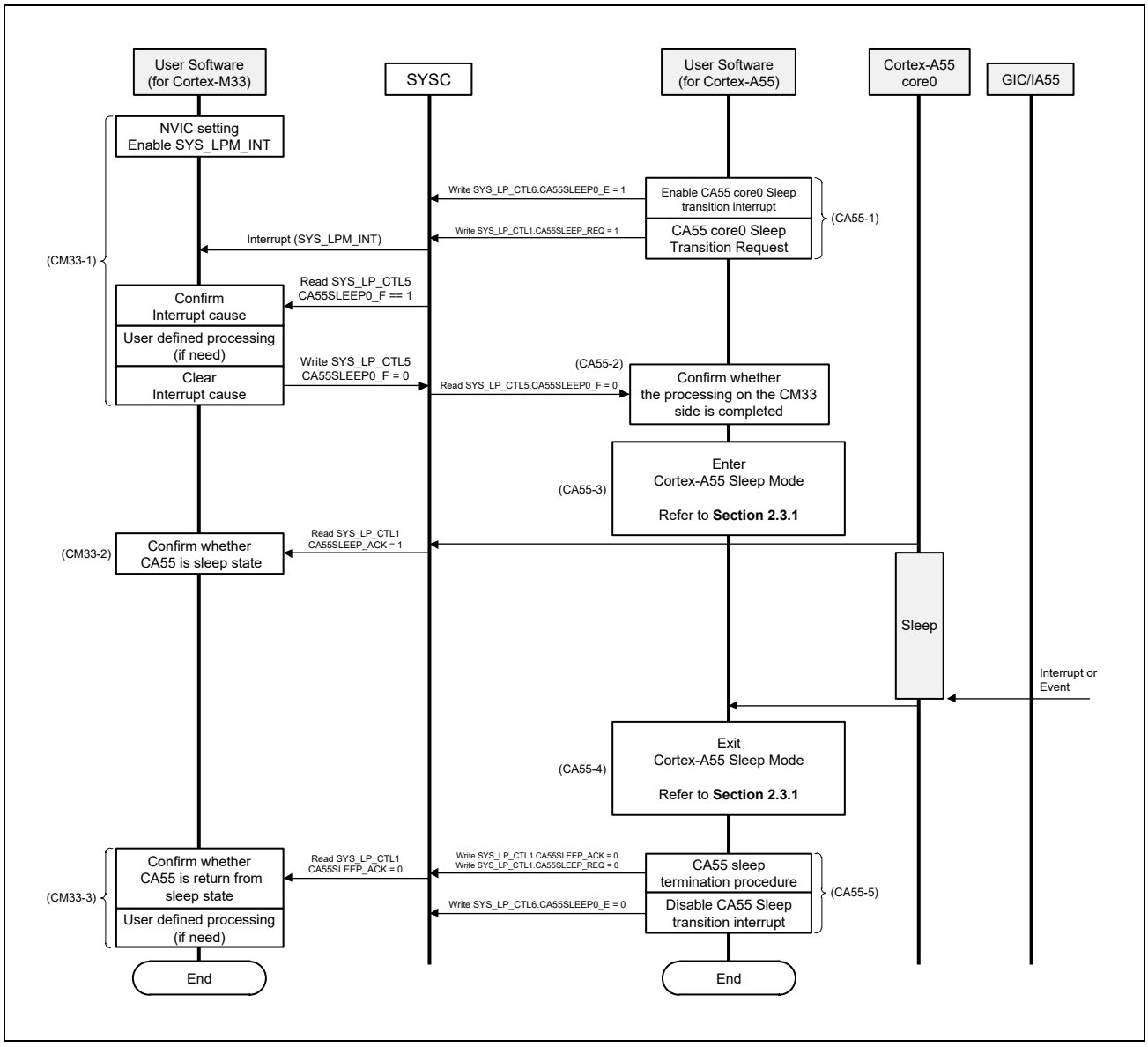


Figure 41.6 The Example Procedure How to Use the SYSC Registers (Cortex-A55 Sleep Mode)

(CA55-1) Set the CA55SLEEP0_E bit of the SYS_LP_CTL6 register to 1 to enable the transition request interrupt to the Cortex-A55 Sleep Mode (core0).
Setting the CA55SLEEP_REQ bit of the SYS_LP_CTL1 register to 1 causes a SYS_LPM_INT interrupt.
The CA55SLEEP0_F bit of the SYS_LP_CTL5 register is set to 1 as an interrupt cause.

(CM33-1) When a SYS_LPM_INT interrupt is detected, the SYS_LP_CTL5 register is read to check the interrupt cause. (Enable the SYS_LPM_INT interrupt on the NVIC in advance.)

By confirming that the CA55SLEEP0_F bit is 1, the software on Cortex-M33 can know that Cortex-A55 core0 is about to transition to Sleep Mode.
Perform the process required when Cortex-A55 is the sleep mode. (if need)
And then clear the CA55SLEEP0_F bit of the SYS_LP_CTL5 register.

(CA55-2) Read the SYS_LP_CTL5 register.
By confirming that the CA55SLEEP0_F is cleared, the software on Cortex-A55 can confirm that the processing on the Cortex-M33 side is completed.

(CA55-3) Enter the Cortex-A55 Sleep Mode.
Refer to **Section 2.3.1, Cortex-A55 Sleep Mode** procedure (1) to (3).

(CM33-2) Read SYS_LP_CTL1 register.
By confirming that the CA55SLEEP_ACK bit is 1, the software on Cortex-M33 can confirm that Cortex-A55 core0 completed transiting to the sleep mode.

(CA55-4) Exit the Cortex-A55 Sleep Mode.
Refer to **Section 2.3.1, Cortex-A55 Sleep Mode** procedure (4) to (6).

(CA55-5) Clear the CA55SLEEP_ACK bit of the SYS_LP_CTL1 register to indicate returning from the sleep mode.
Clear the CA55SLEEP_REQ bit of the SYS_LP_CTL1 register to release the Cortex-A55 sleep mode transition request.
Set the CA55SLEEP_E bit of the SYS_LP_CTL6 register to 0 to disable the transition request interrupt to the Cortex-CA55 Sleep Mode.

(CM33-3) Read SYS_LP_CTL1 register.
By confirming that the CA55SLEEP_ACK bit is cleared, the software on Cortex-M33 can confirm that Cortex-A55 completed returning from the sleep mode.
Perform the process required when Cortex-A55 returned from the sleep mode. (if need)

41.5 Cortex-M33/Cortex-M33_FPU Sleep Mode

41.5.1 Overview

Refer to **Section 3.4.4, Cortex-M33/Cortex-M33_FPU Sleep Mode** for detail.

The following table shows the registers related to the Cortex-M33/Cortex-M33_FPU Sleep Mode.

Refer to **Section 6, System Controller (SYSC)** for detail.

Table 41.7 Cortex-M33/Cortex-M33_FPU Sleep Mode Related Registers

Register Name	Abbreviation	Bit Name
Lowpower Sequence Control Register1	SYS_LP_CTL1	bit[12]: CM33SLEEP_REQ bit[13]: CM33FPUSLEEP_REQ bit[28]: CM33SLEEP_ACK bit[29]: CM33FPUSLEEP_ACK
Lowpower Sequence Control Register2	SYS_LP_CTL5	bit[10]: CM33SLEEP_F bit[11]: CM33USLEEP_F
Lowpower Sequence Control Register5	SYS_LP_CTL6	bit[10]: CM33SLEEP_E bit[11]: CM33FPUSLEEP_E
Lowpower Sequence Control Register6	SYS_LP_CTL7	bit[0]: IM33_MASK bit[1]: IM33_MASK

41.5.2 Operation

This section shows an example procedure.

The following figure is one example how to use the SYSC registers. The SYS_LP_CTL1, the SYS_LP_CTL5 and the SYS_CTL6 registers can be used to share the transition state of Cortex-M33 with Cortex-A55. Referring to these registers, Cortex-A55 can perform processing if need when Cortex-M33 enters and exits the Cortex-M33 Sleep Mode.

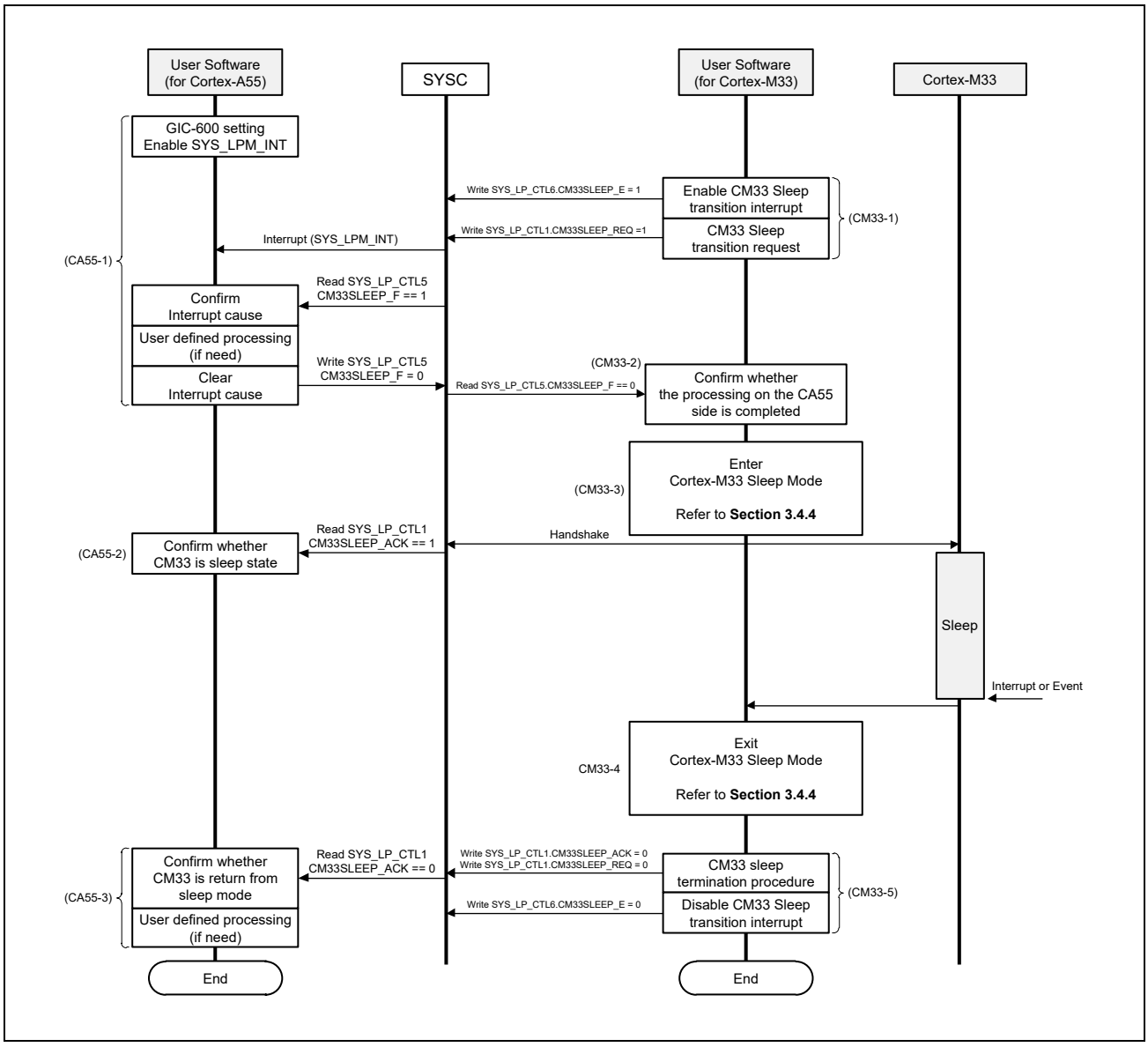


Figure 41.7 The Example Procedure How to Use the SYSC Registers (Cortex-M33 Sleep Mode)

(CM33-1) Set the CM33SLEEP_E bit of the SYS_LP_CTL6 register to 1 to enable the transition request interrupt to the Cortex-M33 Sleep Mode.
Setting the CM33SLEEP_REQ bit of the SYS_LP_CTL1 register to 1 causes a SYS_LPM_INT interrupt.
The CM33SLEEP_F bit of the SYS_LP_CTL5 register is set to 1 as an interrupt cause.

(CA55-1) When a SYS_LPM_INT interrupt is detected, the SYS_LP_CTL5 register is read to check the interrupt cause. (Enable the SYS_LPM_INT interrupt on the GIC in advance.)

By confirming that the CM33SLEEP_F bit is 1, the software on Cortex-A55 can know that Cortex-M33 is about to transition to Sleep Mode.
Perform the process required when Cortex-M33 is the sleep mode. (if need)
And then clear the CM33SLEEP_F bit of the SYS_LP_CTL5 register.

(CM33-2) Read the SYS_LP_CTL5 register.
By confirming that the CM33SLEEP_F is cleared, the software on Cortex-M33 can confirm that the processing on Cortex-A55 side is completed.

(CM33-3) Enter the Cortex-M33 Sleep Mode.
Refer to **Section 3.4.4, Cortex-M33/Cortex-M33_FPU Sleep Mode** procedure (1) to (7).

(CA55-2) Read SYS_LP_CTL1 register.
By confirming that the CM33SLEEP_ACK bit is 1, the software on Cortex-A55 can confirm that Cortex-M33 completed transiting to the sleep mode.

(CM33-4) Exit the Cortex-M33 Sleep Mode.
Refer to **Section 3.4.4, Cortex-M33/Cortex-M33_FPU Sleep Mode** procedure (8) to (12).

(CM33-5) Clear the CM33SLEEP_ACK bit of the SYS_LP_CTL1 register to indicate returning from the sleep mode.
Clear the CM33SLEEP_REQ bit of the SYS_LP_CTL1 register to release the Cortex-M33 sleep mode transition request.
Set the CM33SLEEP_E bit of the SYS_LP_CTL6 register to 0 to disable the transition request interrupt to the Cortex-M33 Sleep Mode.

(CA55-3) Read SYS_LP_CTL1 register.
By confirming that the CM33SLEEP_ACK bit is cleared, the software on Cortex-A55 can confirm that Cortex-M33 completed returning from the sleep mode.
Perform the process required when Cortex-M33 returned from the sleep mode. (if need)

41.6 Power Mode Transition

The power mode transitions are accomplished by combining the following control:

1. Internal control by register control inside this LSI.
2. Power supply control for each power domain.
3. Control flow branches considering the state before state transition and the state after transition.

As for 2 and 3, it depends on the system configuration including this LSI and power control system, so it is necessary to implement it according to the user's system.

41.6.1 ALL_ON to VBATT

The following table shows an example transition flow outline from ALL_ON mode to VBATT mode.

Table 41.8 Example Transition Flow Outline from ALL_ON Mode to VBATT Mode

Step	Controlled by	Operation	Description
1	Cortex-A55	Stop DMA transfer of all IPs.	Depend on IPs.
2	Cortex-A55	Enter DDR IO retention Mode. (if needed)	IO Retention mode setting.
3	Cortex-M33	Transition to Cortex-M33 Sleep Mode. (if booting)	Refer to step (1) to (7) in **Section 3.4.4, Cortex-M33/Cortex-M33_FPU Sleep Mode**.
4	Cortex-M33_FPU	Transition to Cortex-M33_FPU Sleep Mode. (if booting)	Refer to step (1) to (7) in **Section 3.4.4, Cortex-M33/Cortex-M33_FPU Sleep Mode**.
5	Cortex-A55	Activate VBATT module. (if inactive)	CPG_CLKON_VBAT, CPG_RST_VBA, CPG_BUS_MCPU3_MSTOP
6	Cortex-A55	USB PHY PWRRDY signal control (if using USB)	SYS_USB_PWRRDY
7	Cortex-A55	PCIe RST_RSM_B signal control (if using PCIe)	SYS_PCIE_RST_RSM_B
8	Cortex-A55	Enable the isolation cell for PD_VBATT.	ISOENPROT, ISOEN
9	Cortex-A55	Set the VBATT flag*1 to the external device (ex. register in PMIC) via I2C.	VBATT flag is used when returning to ALL_ON mode.
10	Cortex-A55	Shut down the power supply for PD_VCC and PD_ISOVCC from PMIC via I2C.	Regarding to the power off sequence, refer to **Figure 47.4**.

Note 1. The implementation method of VBATT flag depends on the user's system.

41.6.2 VBATT to ALL_ON

The following table shows an example transition flow outline from VBATT mode to ALL_ON mode.

Table 41.9 Example Transition Flow Outline from VBATT Mode to ALL_ON Mode

Step	Controlled by	Operation	Description
1	External	Start the power supply for PD_VCC and PD_ISOVCC from PMIC.	Regarding to the power on sequence, refer to **Figure 47.4**. The isolation cell for PD_VBATT is automatically disabled after power on.
2	External	Release system reset and boot Cortex-A55.	De-assert PRST#.
3	Cortex-A55	Confirm the VBATT flag*1 via I2C to check if normal boot or return from VBATT mode.	The VBATT flag is set when transition to VBATT mode.
4	Cortex-A55	Exit from DDR IO retention mode. (if using the DDR retention mode)	The setting exiting from DDR IO retention mode.
5	Cortex-A55	Activate the other CPUs and the peripheral modules.	Depend on the user system.

Note 1. The implementation method of VBATT flag depends on the user's system.

41.6.3 ALL_ON to AWO

The following table shows an example transition flow outline from ALL_ON mode to AWO mode.

Table 41.10 Example Transition Flow Outline from ALL_ON Mode to AWO Mode

Step	Controlled by	Operation	Description
1	Cortex-A55	Stop DMA transfer of all IPs.	Depend on IPs.
2	Cortex-A55	Enter DDR retention Mode. (if needed)	Retention mode setting.
4	Cortex-M33_FPU	Transition to Cortex-M33_FPU Sleep Mode. (if booting)	Refer to step (1) to (7) in **Section 3.4.4, Cortex-M33/Cortex-M33_FPU Sleep Mode**.
5	Cortex-A55	Module stop setting for peripheral modules and system bus in PD_ISOVCC.	CPG_BUS_***_MSTOP (***: ACPU, PERI_COM, PERI_DDR, TZCDDR), CPG_MHU_MSTOP.
6	Cortex-A55	Clock stop setting for peripheral modules and system bus in PD_ISOVCC.	CPG_CLKON_*** (***: GIC600, MHU, SDHI, USB, ETH, DDR, PCI, AXI_COM_BUS, PERI_COM, AXI_TZCDDR, OTFDE_DDR)
7	Cortex-A55	Reset assert setting for peripheral modules and system bus in PD_ISOVCC.	CPG_RST_*** (***: GIC600, MHU, SDHI, USB, ETH, DDR, PCI, AXI_COM_BUS, PERI_COM, AXI_TZCDDR, OTFDE_DDR)
	Cotex-A55	Clock stop setting and reset assert setting for Cortex-M33_FPU	CPG_CLKON_CM33, CPG_RST_CM33
8	Cortex-A55	USB PHY PWRRDY signal control (if using USB)	SYS_USB_PWRRDY
9	Cortex-A55	PCIe RST_RSM_B signal control (if using PCIe)	SYS_PCIE_RST_RSM_B
11	Cortex-A55	Transition to Cortex-A55 Sleep Mode.	Refer to step (1) to (3) in **Section 2.3.1, Cortex-A55 Sleep Mode**.
12	Cortex-M33	Cortex-A55 Core P-Channel Control Cortex-A55 Cluster P-Channel Control	CPG_CORE0_PCHCTL CPG_CLUSTER_PCHCTL
13	Cortex-M33	Cortex-A55 Q-Channel Control	SYS_LP_CA55CK_CTLn (n=1,2,3)
14	Cortex-M33	Clock stop setting for Cortex-A55	CPG_CLKON_CA55.
15	Cortex-M33	Reset assert setting for Cortex-A55	CPG_RST_CA55.
16	Cortex-M33	Enable the isolation cell for PD_ISOVCC.	SYS_PD_ISO_CTRL
17	Cortex-M33	Shut down the power supply for PD_ISOVCC from PMIC via I2C.	Regarding to the power off sequence, refer to **Figure 47.2** and **Figure 47.3**.

41.6.4 AWO to ALL_ON

The following table shows an example transition flow outline from AWO mode to ALL_ON mode.

Table 41.11 Example Transition Flow Outline from AWO Mode to ALL_ON Mode

Step	Controlled by	Operation	Description
1	Cortex-M33	Supply the power for PD_ISOVCC from PMIC via I2C	Regarding to the power on sequence, refer to **Figure 47.2** and **Figure 47.3**.
2	Cortex-M33	Disable the isolation cell for PD_ISOVCC	SYS_PD_ISO_CTRL
3	Cortex-M33	Initial setting for return to ALL_ON mode	CPG_RETn_REG (n = 0, 1, 2)
4	Cortex-M33	Clock start setting for Coretex-A55	CPG_CLKON_CA55.
5	Cortex-M33	Reset release setting for Cortex-A55	CPG_RST_CA55.
6	Cortex-M33	Cortex-A55 Q-Channel Control	SYS_LP_CA55CK_CTLn (n = 1, 2, 3)
7	Cortex-M33	Cortex-A55 Cluster P-Channel Control Cortex-A55 Core P-Channel Control	CPG_CLUSTER_PCHCTL CPG_CORE0_PCHCTL
8	Cortex-A55	Exit from DDR retention mode (when using)	Setting for exiting form DDR retention mode
9	Cortex-A55	USB PHY PWRRDY signal control (if use USB)	SYS_USB_PWRRDY
10	Cortex-A55	PCIe RST_RSM_B signal control (if use PCIe)	SYS_PCIE_RST_RSM_B
11	Cortex-A55	Clock start setting for system bus and desired peripheral modules in PD_ISOVCC	CPG_CLKON_*** (***: GIC600, MHU, SDHI, USB, ETH, DDR, PCI, AXI_COM_BUS, PERI_COM, AXI_TZCDDR, OTFDE_DDR)
12	Cortex-A55	Release reset setting for system bus and desired peripheral modules in PD_ISOVCC	CPG_RST_*** (***: GIC600, MHU, SDHI, USB, ETH, DDR, PCI, AXI_COM_BUS, PERI_COM, AXI_TZCDDR, OTFDE_DDR)
13	Cortex-A55	Release MSTOP bit for system bus and desired peripheral modules in PD_ISOVCC	CPG_BUS_***_MSTOP (***: ACPU, PERI_COM, PERI_DDR, TZCDDR), CPG_MHU_MSTOP.
14	Cortex-A55	Clock start setting and reset release setting for Cortex-M33_FPU (if use Cortex-M33_FPU)	CPG_CLKON_CM33, CPG_RST_CM33

42. Battery Backup Function (VBATTB)

Backup power supplied by a battery can keep the realtime clock (RTC) operating and retain the important data while the VCC power is turned off. The tamper detection function can be used to erase the backed-up data in cases where an intrusion to the system has been detected.

Regarding the Real time clock, refer to **Section 22, Realtime Clock (RTCA-3)** for details.

42.1 Overview

Table 42.1 shows the specifications of the VBATTB.

Table 42.1 VBATTB Specifications

Item	Description
Scope of backup	All modules in the backup domain (PD_VBATT) • Backup registers • 32kHz-clock oscillator • Tamper detector • Realtime clock (RTC)
Backup registers	128 bytes • The backup registers can immediately be erased in response to the detection of physical tampering.
Tamper event detection	When the tamper detector has detected an intrusion to the system, it indicates this by setting a flag and, if this is selected, generating an interrupt. • Tamper event can acquire a timestamp on tamper detection. • Tamper input pin (TAMPIN) • Noise filter (sampling rate: 32.768 kHz; passes signal transitions in response to three consecutive samples with the same level)

Figure 42.1 shows a block diagram of the VBATTB, and **Table 42.2** shows the external pins of the VBATTB.

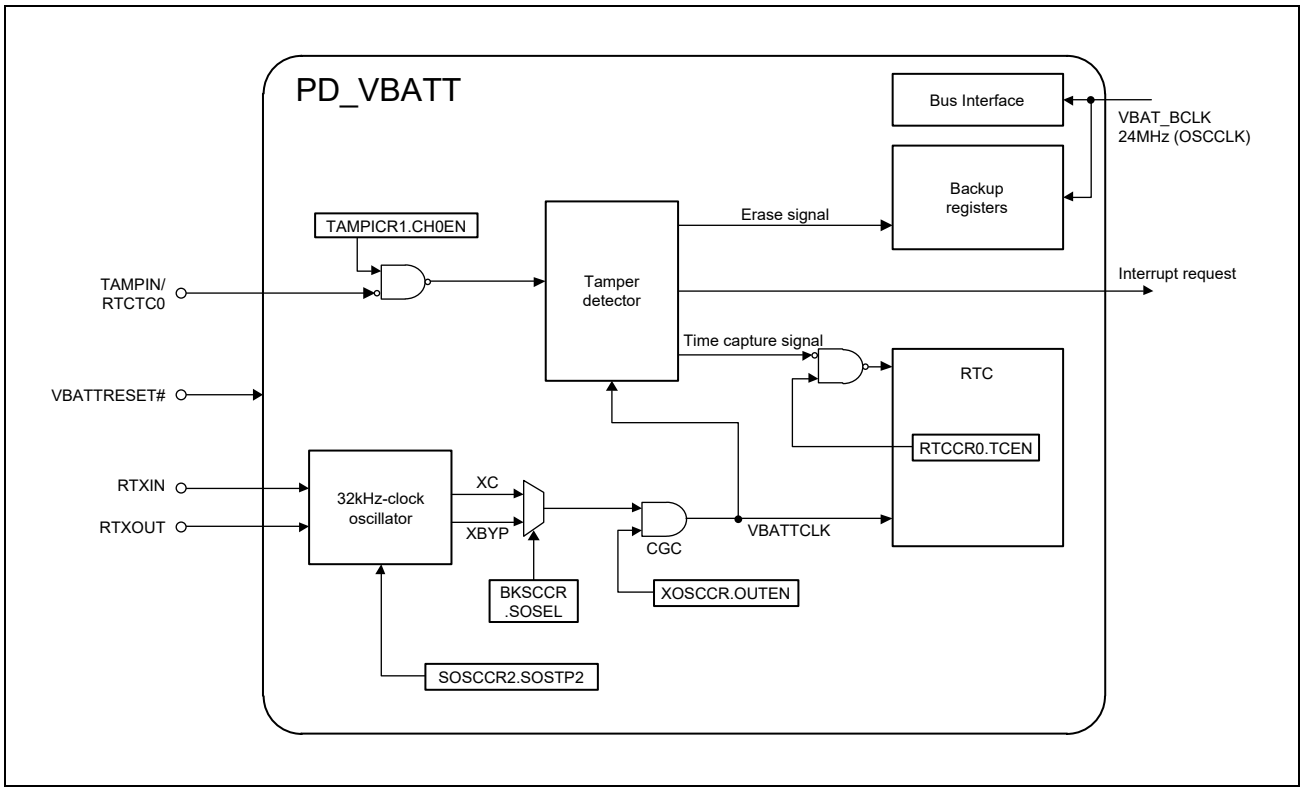


Figure 42.1 Block Diagram of VBATTB

Table 42.2 External Pins

Pin Name	I/O	Function
VBATTRESET#	Input	Backup domain reset
RTXIN	Input	Pins for the 32kHz-clock • Oscillation mode : Connect 32.768-kHz crystal to RTXIN and RTXOUT pins. • Bypass mode (RTXOUT to XBYP): Connect external clock to RTXOUT pin.
RTXOUT	Input /Output	
TAMPIN	Input	Tamper input pin (TAMPIN) / Time capture event input pin for RTC (RTCTC0)

42.2 Register Configuration

NOTE

- Registers in the battery backup function (VBATTB) can only be initialized by the backup domain reset (VBATTRESET#).
- Resets generated by other sources such as the PRST# pin or software will not affect these registers.

The Base Address is as follows.

Base Address: H'0_1005_C000 (Cortex-A55 Address Space)
Base Address: H'4005_C000 (Cortex-M33/Cortex-M33_FPU Address Space Secure)
Base Address: H'5005_C000 (Cortex-M33/Cortex-M33_FPU Address Space Non-Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

Table 42.3 shows the register list of the VBATTB. Prohibit to access the undefined area.

Table 42.3 Register List

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Backup Domain Power Status Register	BKPSR	R/W	H'0000_00x1	H'00	32
Tamper Status Register	TAMPSR	R/W	H'0000_0000	H'04	32
Tamper Control Register	TAMPCR	R/W	H'0000_0000	H'08	32
Time Capture Event Control Register	TCECR	R/W	H'0000_0000	H'0C	32
Tamper/RTCTC Input Control Register 1	TAMPICR1	R/W	H'0000_0000	H'10	32
Tamper/RTCTC Input Control Register 2	TAMPICR2	R/W	H'0000_0070	H'14	32
Tamper/RTCTC Input Monitoring Register	TAMPIMR	R	H'0000_000x	H'18	32
Backup Domain 32kHz-Clock Control Register	BKSCCR	R/W	H'0000_0000	H'1C	32
32kHz-Clock Oscillator Control Register 2	SOSCCR2	R/W	H'0000_0000	H'24	32
Isolation Enable Control Register	ISOEN	R/W	H'0000_0000	H'28	32
Isolation Enable Write Protect Control Register	ISOENPROT	R/W	H'0000_0000	H'2C	32
Oscillator Control Register	XOSCCR	R/W	H'0000_0000	H'30	32
Backup Register n (n = 0 to 31)	BKRn	R/W	H'0000_0000	H'80 to H'FC	32

42.3 Register Descriptions

42.3.1 Backup Domain Power Status Register (BKPSR)

The BKPSR register indicates the VBATRESET# status monitor.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DDRIO CTL	—	—	—	—	—	—	VBT RSTF
Initial Value	0	0	0	0	0	0	0	0	0	0	x	1	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	DDRIOCTL	0	R/W	This bit is used when DDR IO retention control. Set to 0 when exit from DDR IO retention mode. Set to 1 when enter DDR IO retention mode. *Note*: DDR cannot be accessed when DDRIOCTL is 1.
6	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
5	—	x	R	Reserved When read, the initial value is undefined. The written value will be ignored.
4	—	1	R	Reserved When read, the initial value is read. The written value will be ignored.
3 to 1	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	VBTRSTF	1	R/W	VBATTRESET# status flag. This bit indicates VBATTRESET# status. When an assert of VBATTRESET# is detected, VBTRSTF is set to 1. This status flag can be cleared writing 0. [Read Operation] 0: VBATTRESET# is de-asserted. 1: VBATTRESET# is asserted. [Write operation] 0: VBTRSTF status flag is cleared (VBTRSTF is set to 0). 1: VBTRSTF is set to 1.

Note: This register can only be initialized by the backup domain reset (VBATTRESET# pin).

42.3.2 Tamper Status Register (TAMPSR)

The TAMPSR register indicates the status of the tamper detection events.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TAMP0F
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 1	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	TAMP0F	0	R/W	Tamper 0 Detection Flag 0: Tamper 0 event has not been detected. 1: Tamper 0 event has been detected. Only 0 can be written to this bit, to clear the flag. To clear this flag, confirm that its value is 1 and then write 0 to it. TAMP0F flag indicates that a trigger (an effective edge) was input to the TAMPIN pin. The trigger is selected by the TAMPICR2.CH0TRG bit. [Setting condition] • The trigger set by the TAMPICR2.CH0TRG bit being detected on the TAMPIN pin. [Clearing conditions] • 0 being written to the TAMP0F flag after having confirmed that its value is 1.

Note: This register can only be initialized by the backup domain reset (VBATTRESET# pin).

42.3.3 Tamper Control Register (TAMPCR)

The TAMPCR register controls operations in response to the detection of tampering.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TAMP0EE	—	—	—	TAMP0IE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 5	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
4	TAMP0EE	0	R/W	Tamper 0 Erase Enable 0: Backup registers are not erased in response to a tamper 0 event. 1: Backup registers are erased in response to a tamper 0 event. The TAMP0EE bit selects whether the backup registers should be or should not be erased in response to a tamper 0 event.
3 to 1	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	TAMP0IE	0	R/W	Tamper 0 Detection Interrupt Enable. 0: Tamper 0 detection interrupt is disabled. 1: Tamper 0 detection interrupt is enabled. The TAMP0IE bit enables or disables the tamper n detection interrupt.

Note: This register can only be initialized by the backup domain reset (VBATTRESET# pin).

42.3.4 Time Capture Event Control Register (TCECR)

The TCECR register selects the sources for time capture events of the RTC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TCE0S
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 1	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	TCE0	0	R/W	Time Capture Event 0 Source Select 0: Input signal from the RTCTC0 pin 1: Tamper 0 event The TCE0S bit selects the source of time capture event 0 for the RTC. When this bit is set to 0, a signal input to the RTCTC0 pin is input to the RTC. Selection of this setting is recommended when tamper detection is not to be used. When this bit is set to 1, the tamper 0 event detection signal is input to the RTC. Selection of this setting is recommended when a tamper event is to cause a timestamp to be recorded.

Note: This register can only be initialized by the backup domain reset (VBATTRESET# pin).

42.3.5 Tamper/RTCTC Input Control Register 1 (TAMPICR1)

The TAMPICR1 register enables or disables input from the TAMPIN/RTCTC0 pins.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CH0EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 1	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	CH0EN	0	R/W	Channel 0 Input Enable 0: The TAMPIN/RTCTC0 signal input is disabled. 1: The TAMPIN/RTCTC0 signal input is enabled. The CH0EN bit enables or disables the input from the TAMPIN/RTCTC0 pin. Set each bit to 1 not only if tamper events are to be detected but also if the RTC time capture function is to be used.

Note: This register can only be initialized by the backup domain reset (VBATTRESET# pin).

42.3.6 Tamper/RTCTC Input Control Register 2 (TAMPICR2)

The TAMPICR2 register is used to enable or disable the noise filters for the TAMPIN/RTCTC0 pin and select the triggers for tamper event detection.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	CH0 TRG	—	—	—	CH0 NFE
Initial Value	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
6 to 5	—	11b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
4	CH0TRG	1	R/W	Channel 0 Trigger Select Select the trigger for tamper 0 event detection. 0: A falling edge of the input on the TAMPIN pin 1: A rising edge of the input on the TAMPIN pin The CH0TRG bit is used to select the effective edge for the input signal from the TAMPIN pin for use as a trigger of tamper event detection.
3 to 1	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	CH0NFE	0	R/W	Channel 0 Noise Filter Enable 0: Noise filter for the TAMPIN/RTCTC0 pin is disabled. 1: Noise filter for the TAMPIN/RTCTC0 pin is enabled. The CH0NFE bit enables or disables the noise filter for the TAMPIN/RTCTC0 pin. When tamper 0 event detection is not to be used, set the bit to 0. Five cycles of the 32kHz-clock are necessary for the output of the noise filter to become stable after this bit has been set to 1.

Note: This register can only be initialized by the backup domain reset (VBATTRESET# pin).

42.3.7 Tamper/RTCTC Input Monitoring Register (TAMPIMR)

The TAMPIMR register is used to monitor the input levels on the TAMPIN pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CH0LV
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 3	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2 to 1	—	x	R	Reserved When read, the initial value is undefined. The written value will be ignored.
0	CH0LV	x	R	Channel 0 Level Monitoring Flag 0: The low level is being input on the TAMPIN pin. 1: The high level is being input on the TAMPIN pin. CH0LVL flag indicates the input level on an TAMPIN pin.

Note: This register can only be initialized by the backup domain reset (VBATTRESET# pin).

42.3.8 Backup Domain 32kHz-Clock Control Register (BKSCCR)

The BKSCCR register is used to select the output of the 32kHz-clock oscillator provided to each module in the backup domain.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SOSEL	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
6	SOSEL	0	R/W	32kHz-Clock Oscillator output Select 0: The 32kHz-clock oscillator XC output is selected. 1: The 32kHz-clock oscillator XBYP output is selected. The SOSEL bit is used to select the output of the 32kHz-clock oscillator provided to each module in the backup domain. When using external clock, select the XBYP output.
5 to 0	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

Note: This register can only be initialized by the backup domain reset (VBATTRESET# pin).

42.3.9 32kHz-Clock Oscillator Control Register 2 (SOSCCR2)

The SOSCCR2 register controls the 32kHz-clock oscillation.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOSTP2
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 1	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	SOSTP2	0	R/W	32kHz-Clock Oscillator Stop 0: 32kHz-clock oscillator is operating. 1: 32kHz-clock oscillator is stopped.

Note: This register can only be initialized by the backup domain reset (VBATTRESET# pin).

42.3.10 Isolation Enable Control Register (ISOEN)

The ISOEN register controls the isolation enable signal before entering the battery backup mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ISOEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	ISOEN	0	R/W	Isolation Enable 0: Isolation OFF 1: Isolation ON The ISOEN bit should be set to 1 before entering the battery backup mode. This register can be written when released the write protection by the ISOENPROT register.

Note: This register can only be initialized by the backup domain reset (VBATTRESET# pin).

42.3.11 Isolation Enable Write Protect Control Register (ISOENPROT)

The IOSENPROT register is used to remove write-protection from the ISOEN register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WPROT[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WPROT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	WPROT	0	R/W	ISOEN register write protect control After writing H'15AF_FA51 to this register, ISOEN register can be written. When ISOEN register is written, this register is cleared to zero.

Note: This register can only be initialized by the backup domain reset (VBATTRESET# pin).

42.3.12 Oscillator Control Register (XOSCCR)

The Oscillator Control register controls the 32-kHz oscillator.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OUTEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

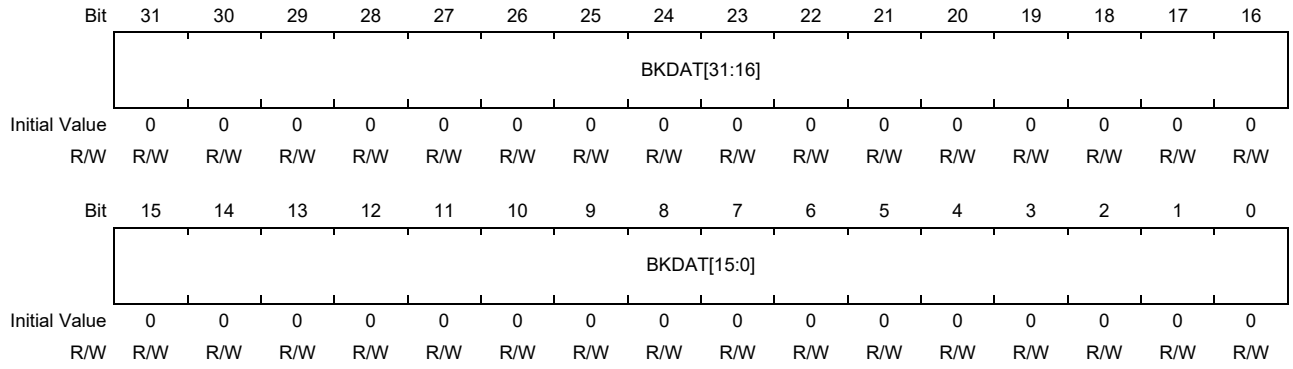
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	XSEL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	OUTEN	0	R/W	32kHz-clock output control 0: Disable 32kHz-clock output 1: Enable 32kHz-clock output
15 to 2	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1 to 0	XSEL	0	R/W	Oscillation mode control 00: CL = 4 pF 01: CL = 7 pF 10: CL = 9 pF 11: CL = 12.5 pF

Note: This register can only be initialized by the backup domain reset (VBATTRESET# pin).

42.3.13 Backup Register n (BKRn) (n = 0 to 31)

The BKRn (0 to 31) registers are backup registers.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BKDAT	0	R/W	Backup Data The contents of the BKRn registers are retained even in the battery backup mode. The BKRn registers can be erased when a tamper event is detected.

Note: This register can only be initialized by the backup domain reset (VBATTRESET# pin).

42.4 Operation

42.4.1 Battery Backup Function

This function is used to retain the operations of the modules in the backup domain by supplying power from the VBATTB pin while the main power supply is turned off.

The following modules are in the backup domain.

- Backup registers
- 32kHz-clock oscillator
- Tamper detector
- RTC

42.4.2 Tamper Detctor

This LSI has TAMPIN pin which remains functional in the battery backup mode and can detect physical intrusions.

The tamper inputs can be configured for rising or falling edge detection and with or without noise filtering.

The following operations can be selected in response to the detection of tampering.

- Generating an interrupt request
- Erasing the backup registers
- Generating RTC timestamp events

Figure 42.2 is a block diagram of the tamper detector.

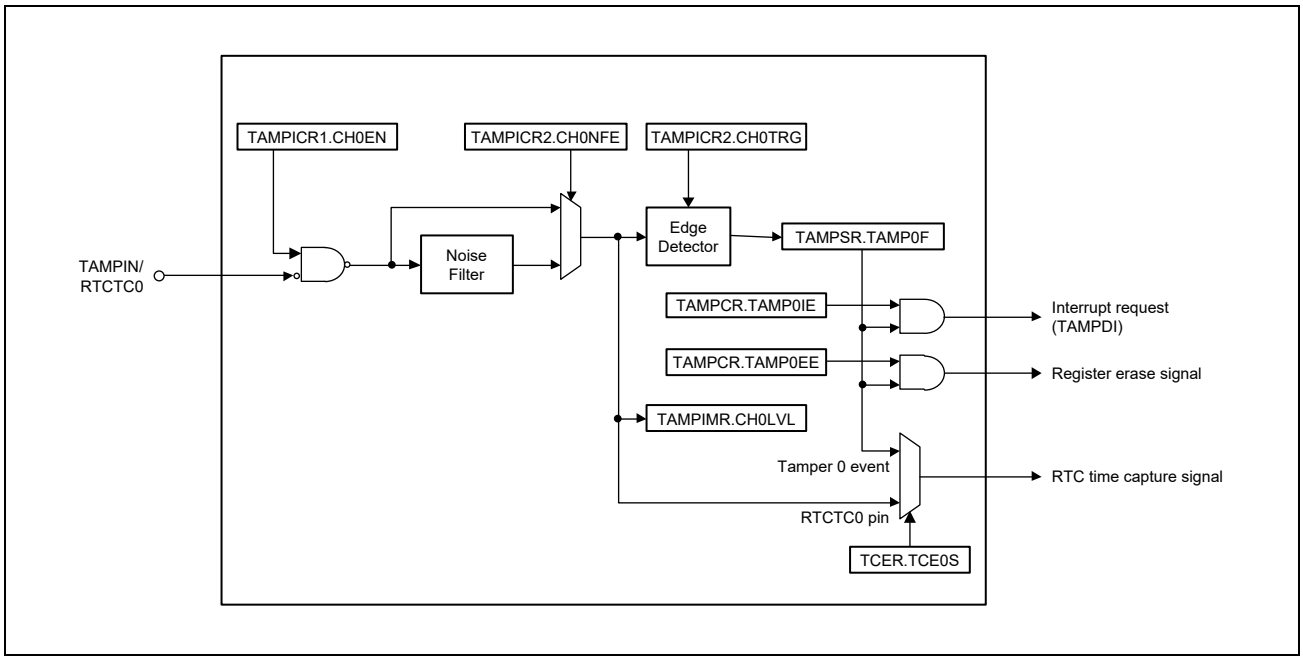


Figure 42.2 Tamper Detector

If the value of the TAMPICR1.CH0EN bit or the CH0NFE or CH0TRG bit in the TAMPICR2 register is changed, a TAMPSR.TAMP0F flag may become 1. Set the TAMP0IE and TAMP0EE bits in the TAMPCR register to 0 before setting these bits. Clear the TAMP0F flags after setting these bits.

The TAMPIMR.CH0LVL flags are for monitoring the levels being input on the TAMPIN pin.

When an effective edge is detected, the TAMP0F flag becomes 1. If the TAMP0IE bit is 1 at this point, the TAMPDI interrupt is generated. If the TAMP0EE bit is 1, the backup registers are erased to H'00.

Setting the TCECR.TCE0S bit to 1 enables a timestamp on tamper detection by using the time capture function of the RTC. For details on the time capture function, refer to **Section 22, Realtime Clock (RTCA-3)**.

Figure 42.3 shows the setting flow when using the tamper detector.

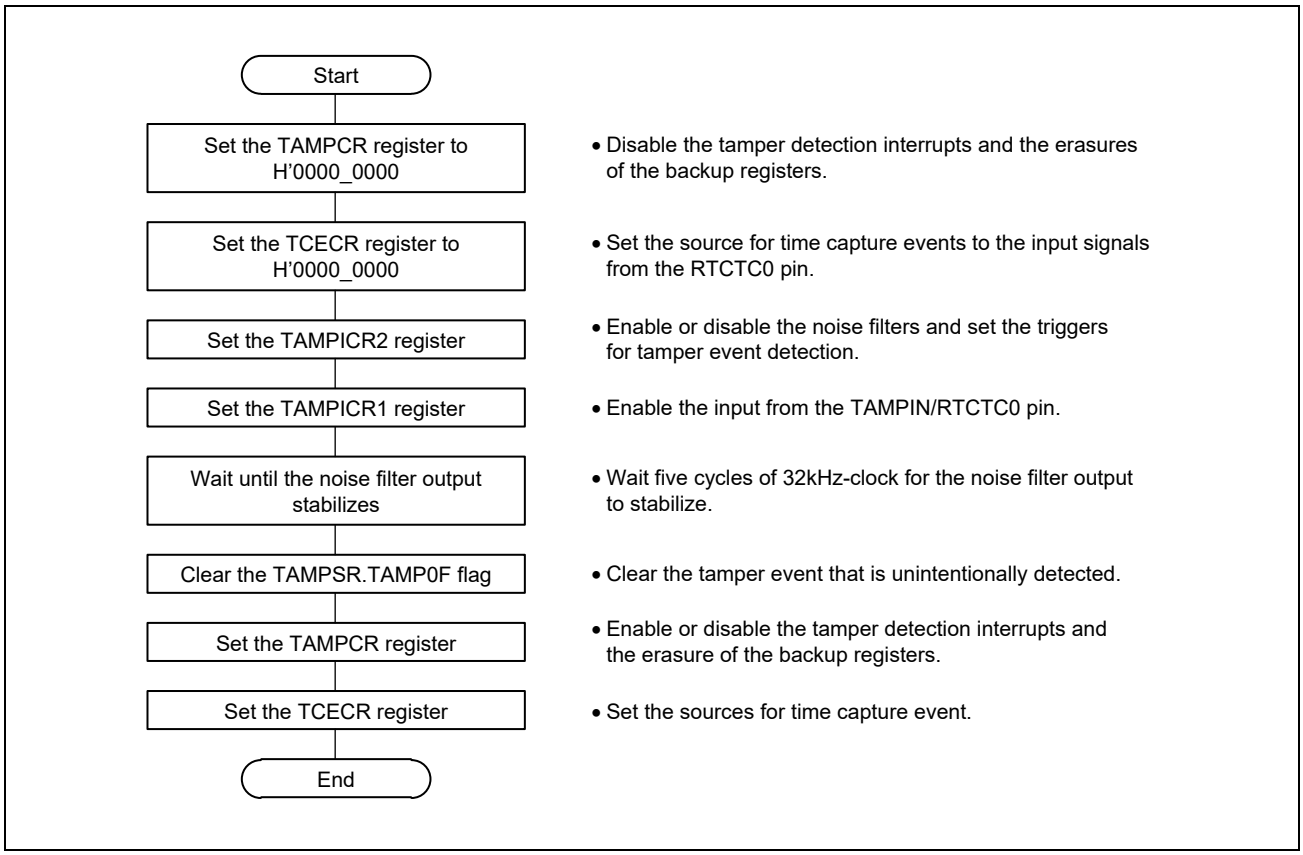


Figure 42.3 Setting for Tamper Detector

42.4.3 Isolation Control

Before entering the battery backup mode (VBATT mode), the following procedure should be performed.

1. Write “H’15AF_FA51” to ISOENPROT register to release the write protection of ISOEN register.
2. Set the ISOEN bit of ISOEN register to 1.

Refer to the **Section 41, Low Power Consumption** regarding to the battery backup mode.

42.4.4 Interrupt

The VBATTB has interrupt source listed in **Table 42.4.**

Table 42.4 VBATTB Interrupt Source

Name	Symbol	Interrupt Source	Interrupt Flag	Interrupt Enable Bit
Tamper detection interrupt	TAMPDI	Tamper 0 detection	TAMP0F	TAMP0IE

43. Renesas Secure IP (RSIP-E05A)

The Renesas Security IP (RSIP-E05A) consists of the access management circuit, encryption engine, and random number generation circuit. In combination with the RSIP-E05A library, the RSIP-E05A can prevent eavesdropping (to maintain confidentiality), falsification of information (to ensure integrity), and impersonation (to verify authenticity).

Because key information required for encryption and decryption is stored only in the RSIP-E05A and all accesses from the outside can be blocked, RSIP-E05A enables building a more robust security system.

43.1 Overview

Table 43.1 summarizes the specifications of the RSIP-E05A. **Figure 43.1** shows a block diagram of the RSIP-E05A.

Table 43.1 Specifications of RSIP-E05A (1/2)

Item	Description
Access control	Access management circuit • In case of irregular access to the RSIP-E05A due to a tampered program or CPU runaway, this circuit blocks all subsequent accesses and stops data output from the RSIP-E05A
Encryption engine	AES: Compliant with NIST FIPS PUB 197 • Key length: 128, 192, or 256 bits • Data block size: 128 bits • AES supports following block cipher mode – ECB, CBC, CTR: Compliant with NIST SP 800-38A – CCM: Compliant with NIST SP 800-38C – GCM: Compliant with NIST SP 800-38D – XTS: Compliant with IEEE 1619-2007 • AES supports following authentication algorithm – CMAC: Compliant with NIST SP 800-38B – GMAC: Compliant with NIST SP 800-38D • AES does not support the KEY size of 192-bit in XTS, CMAC, GMAC, CCM, GCM and CMAC.
Random number generation	128-bit true random number generation circuit
Signature generation and verification	RSA • Maximum number of operable bits: 4224 bits • Signature generation, signature verification, public-key encryption, private-key decryption (Add RSA compliance specification(s)) ECC • Maximum number of operable bits: 576 bits • Signature generation, signature verification, key generation
Message digest computation	HASH • Block size: – 512-bit (SHA-1, SHA-224, SHA-256) – 1024-bit (SHA-512/224, SHA-512/256, SHA-384, SHA-512) • Key size: 512-bit or less • HASH supports following secure hash algorithms – SHA-1, SHA-224, SHA-256, SHA-512/224, SHA-512/256, SHA-384 and SHA-512: Compliant with FIPS PUB 180-4 • HASH supports following message authentication algorithm – HMAC: Compliant with FIPS PUB 198

Table 43.1 Specifications of RSIP-E05A (2/2)

Item	Description
Hardware Unique Key	• Key derivation functions combine the Hardware Unique Key with the key generation information. The derived keys implement the key wrapping for user key secure storage. • The HUK uniqueness prevents the illicit cloning and copying of keys to another LSI of the LSI group. • The HUK itself is stored in wrapped (encrypted, non-plain) format in an isolated memory area. Therefore it is protected from illicit access and copy.
Application Key Management	• Wrapped keys are only valid within the RSIP-E05A.
Unique ID	• An LSI (Unique ID) is accessible from the access management circuit. • Key derivation functions combine the Unique ID with the key generation information. Such derived keys are used to unwrap the HUK within the RSIP-E05A boundary.
On-the-fly Decryption / Encryption (OTFDE)	• The RSIP-E05A outputs the key data of the On-the-fly Decryption / Encryption IP (OTFDE) through a dedicated bus
SPA/DPA Protections	• SPA/DPA protections can optionally be enabled for crypto engine processing.
Low power consumption	• Setting of the module stop state is possible.

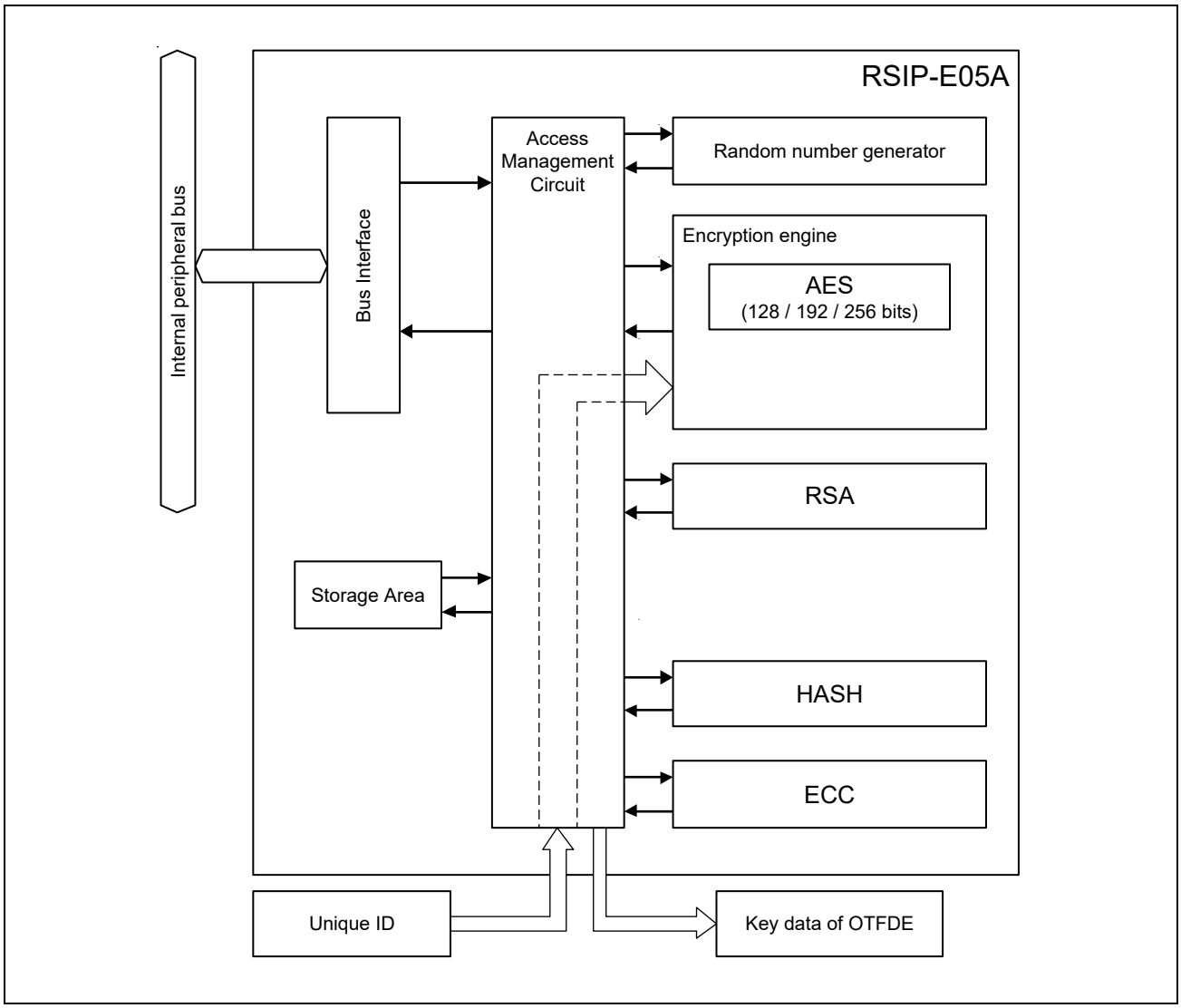


Figure 43.1 RSIP-E05A Block Diagram

43.2 Operation

43.2.1 Encryption Engine

Figure 43.2 shows conceptual block diagram of the encryption engine installed in the RSIP-E05A.

The encryption engine uses the input key information, and converts the plaintext data to ciphertext or ciphertext data to plaintext through the hardware.

The encryption/decryption process can be completed without exposing the key data and the process's intermediate data outside of the RSIP-E05A. This process is performed by the secure engine and storage area of the RSIP-E05A internally.

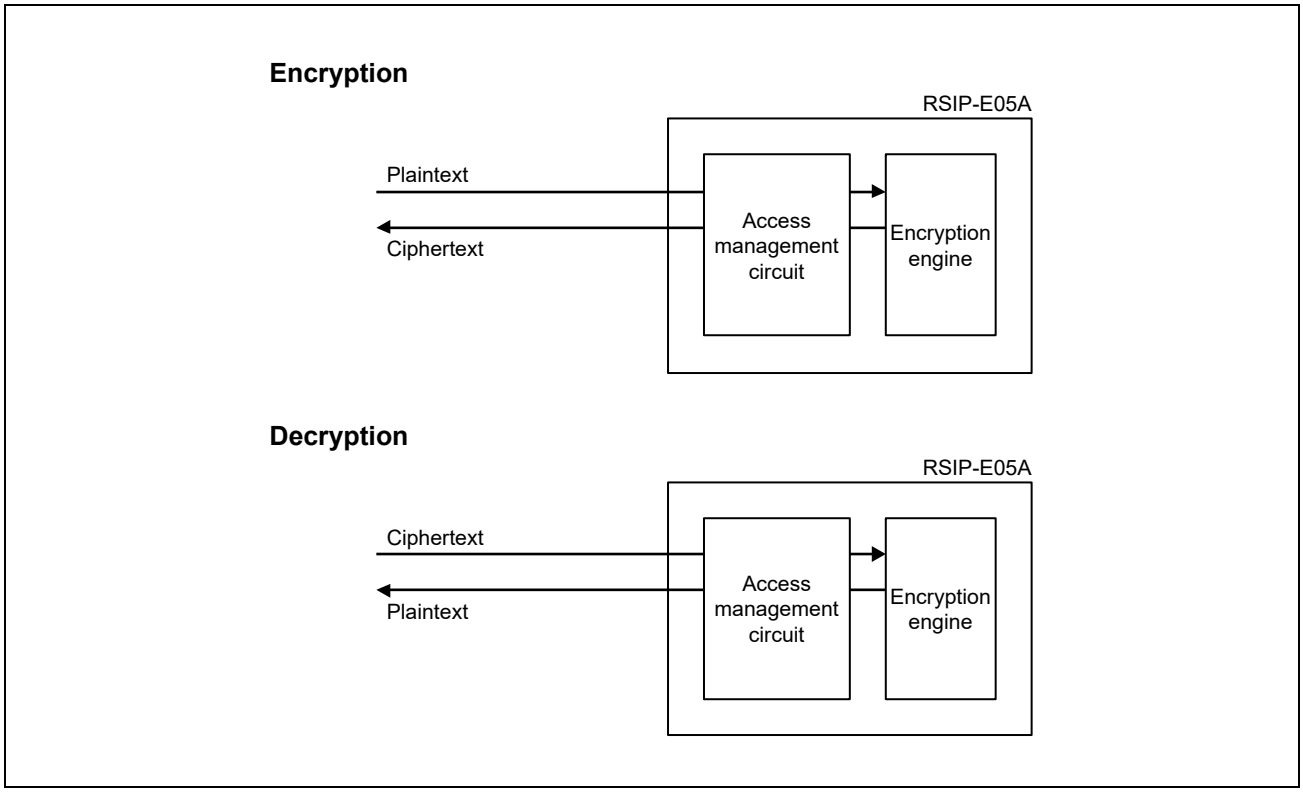


Figure 43.2 Encryption and Decryption Processes by Encryption Engine

43.2.2 Encryption and Decryption

Follow the procedure below to encrypt and decrypt the data:

(1) Input the key information to the RSIP-E05A.

(2) Input the target data to the RSIP-E05A. Plaintext data is converted to ciphertext and ciphertext data to plaintext.

(3) Read the converted data.

The encryption engine has input and output buffers, and can perform encryption/decryption in parallel with data input/output. **Figure 43.3** shows the encryption engine timing.

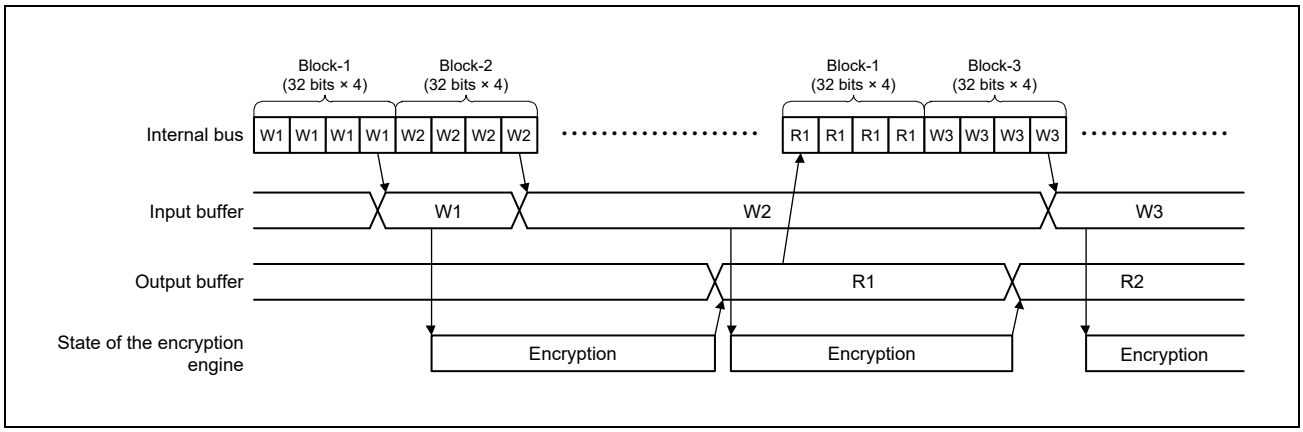


Figure 43.3 Encryption and Decryption Timing Diagram

43.3 Usage Notes

43.3.1 SPA/DPA Protections

SPA/DPA protections can optionally be enabled for crypto engine processing.

43.3.2 Renesas Secure IP(RSIP-E05A) Driver

Use of the RSIP-E05A requires the RSIP-E05A driver provided by Renesas Electronics. Please contact our sales office for information regarding the RSIP-E05A driver.

44. On-The-Fly Decryption/Encryption (OTFDE)

44.1 Overview

On-The-Fly Decryption/Encryption (OTFDE) module encrypts and decrypts data stored in specific address areas of external memory connected to this LSI to ensure data security.

The supported encryption algorithm is AES. The AES key is managed by RSIP.

This LSI has the OTFDE modules for DDR controller and xSPI or Octa controller as shown in **Figure 44.1**.

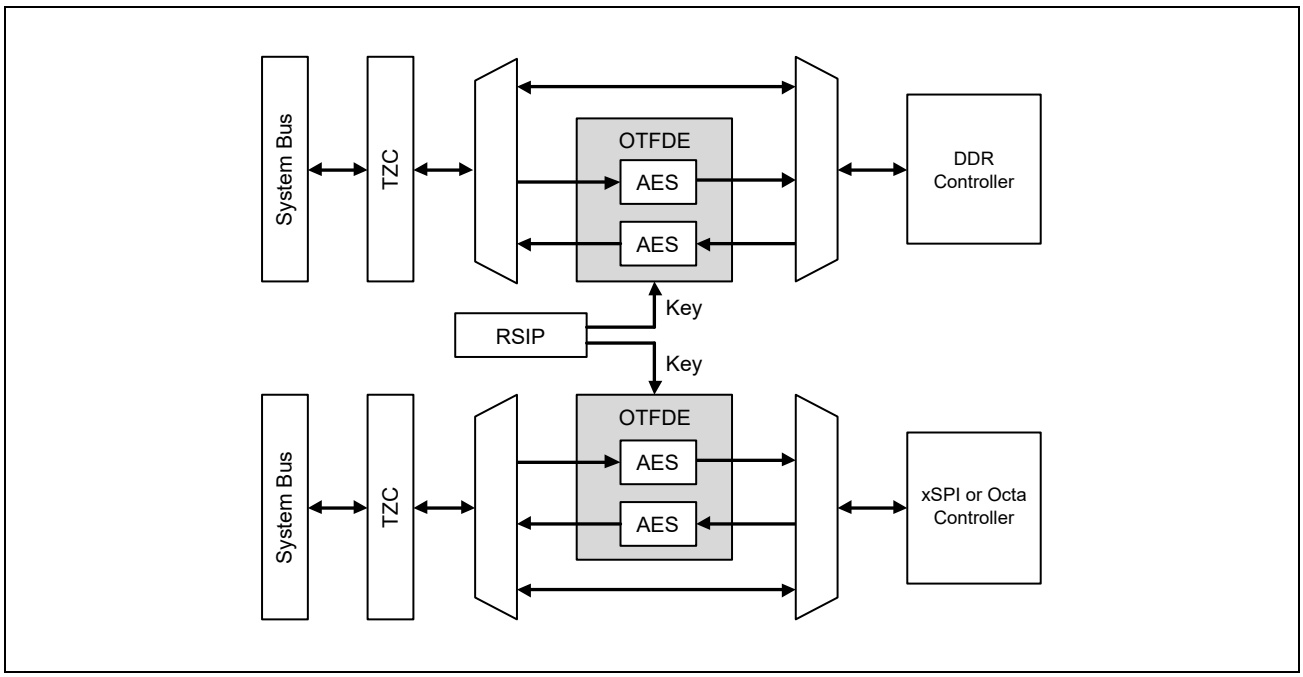


Figure 44.1 On-The-Fly Decryption/Encryption Block Diagram

44.2 Renesas Secure IP(RSIP) Driver

Use of the OTFDE requires the RSIP driver provided by Renesas Electronics. Please contact our sales office for information regarding the RSIP driver.

45. General Purpose Input Output Port (GPIO)

This LSI can use up to 82 general-purpose I/O ports.

Each port of the general-purpose I/O port is multiplexed with the terminal of the peripheral function, and either the general-purpose I/O port function or the peripheral function can be selected by setting the register.

45.1 Features

45.1.1 General Purpose Input Output Port (GPIO)

General-purpose input/output ports (GPIOs) provide general-purpose ports that can be configured as inputs or outputs.

When set as an output, it can be written to an internal register to control the state driven by the output pin. When set as an input, you can read the state of the input by reading the state of the internal registers. You can also generate interrupts using the GPIO input pin.

- GPIO Output Control (Port Register [P_m])
- GPIO Input, Output Enable (Port Mode Register [PM_m])
- GPIO Input Monitor (Port Input Register [PIN_m])
- Interrupt used the GPIO input (Interrupt Enable Control Register[ISEL_m])

Also, it able to control the following on some ports

- IO voltage mode control for each port
- Buffer drive ability control for each port
- Pull-up/Pull-down Switching for each port
- Digital noise filter control for each port

45.1.2 Port Function Control

Switching of each port function can be controlled by the following register settings.

- GPIO function and peripheral function selection (Port Mode Control Register[PMC_m])
- Selection of each peripheral function (Port Function Control Register[PFC_m]) [Function1 to 8]

NOTE

Please refer to another excel file for the pin function for the peripheral functions that can be selected.
Please refer to above excel file for the “Pin function list” described in each chapter.
Do not select the same peripheral function for multiple different external terminals.

45.1.3 Special Purpose Port Function Control

The settings of the following specific ports can be controlled by registers.

- IO voltage mode control for each port
- Buffer drive ability control for each port
- Digital noise filter control for each port
- Ether MII/RGMII mode control

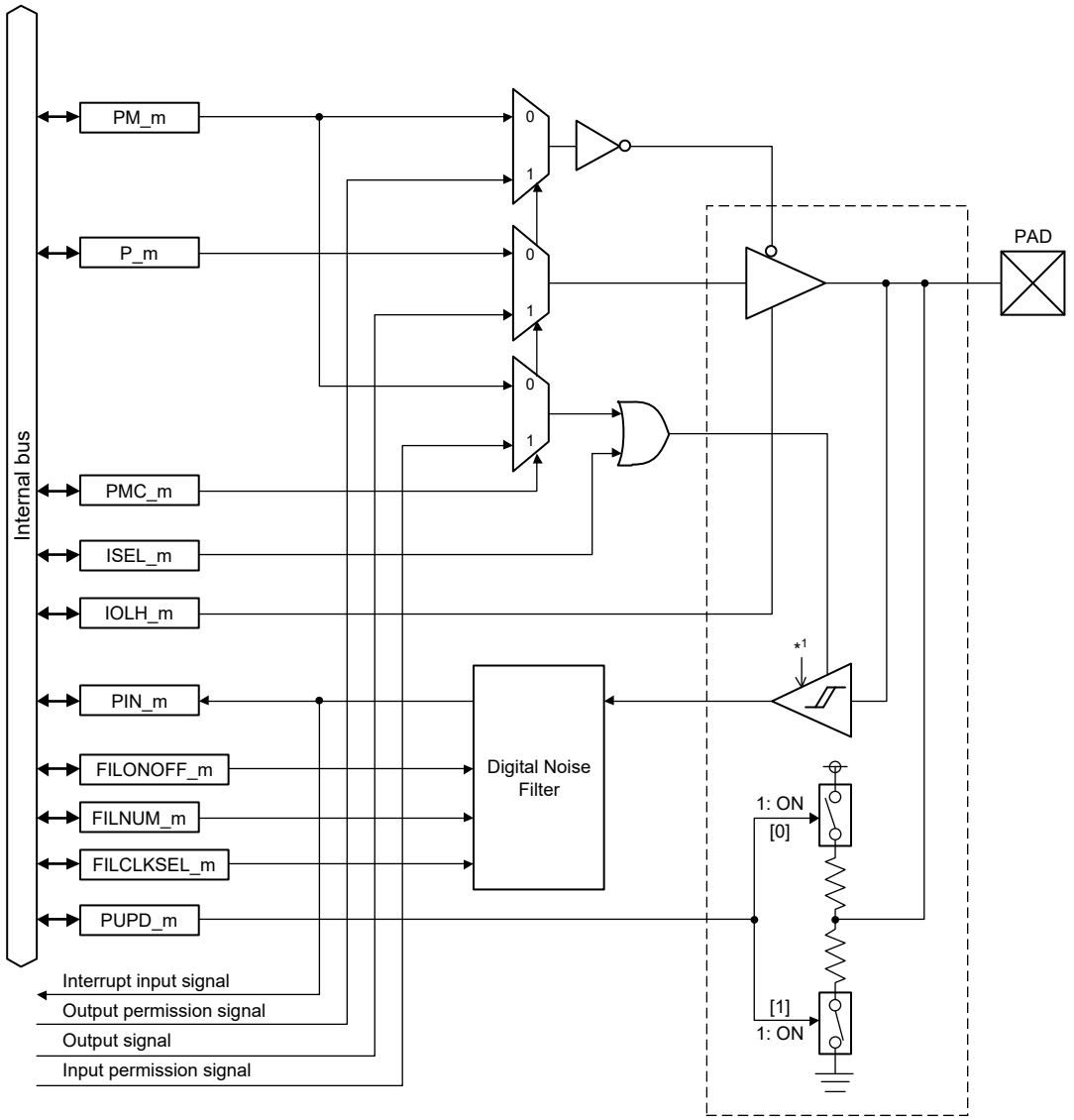
The specific ports shown below are targeted.

- Target Functions: Ether ch0/ch1, SDHI ch0/ch1, XSPI, RIIC, I3C
- Target Signals: NMI, TMS/SWDIO, TDO, WDTOVF_PERROUT#, AUDIO_CLK1, AUDIO_CLK2

45.1.4 General Purpose Input Output Port Configuration

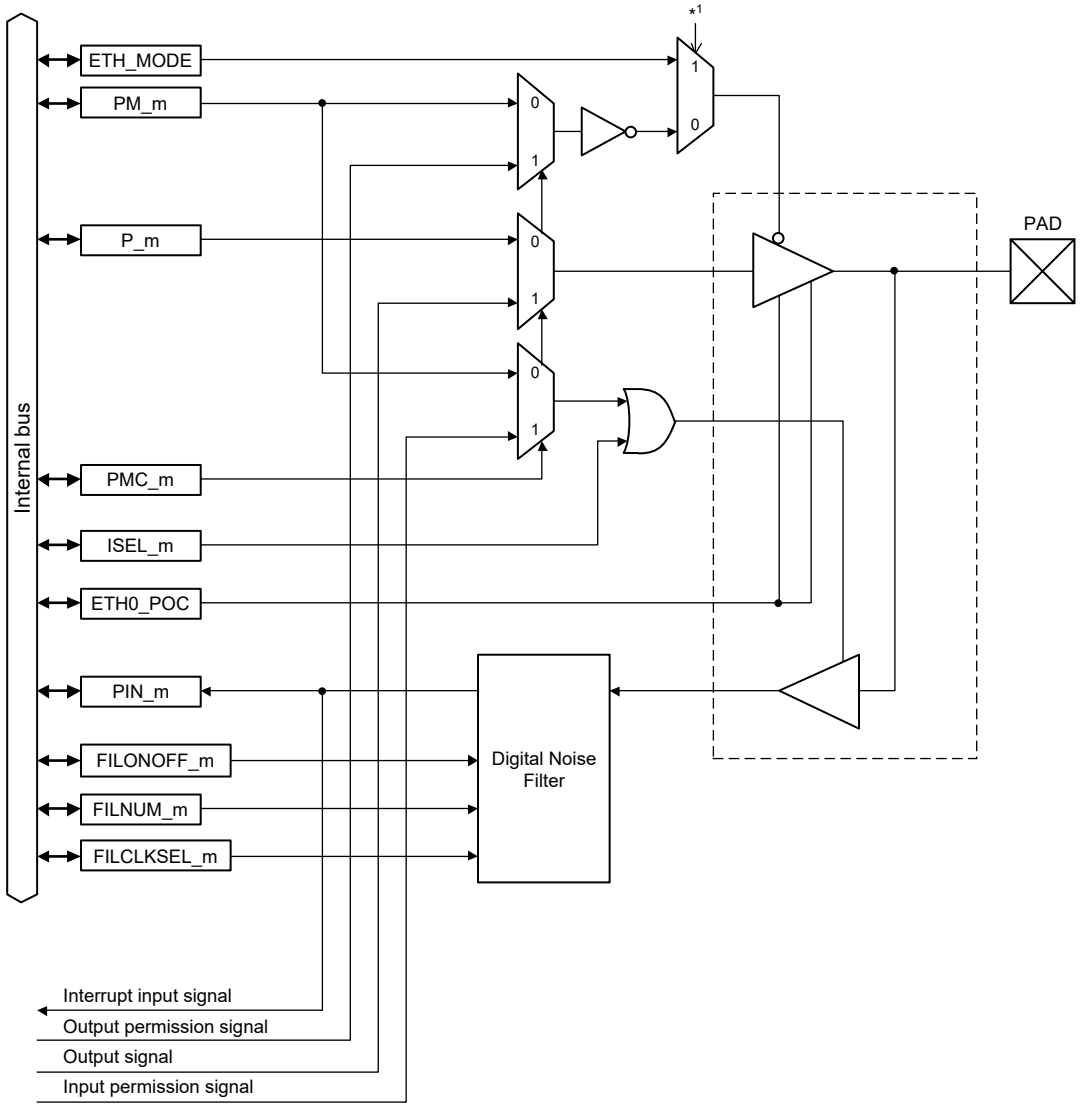
The configuration of the general-purpose I/O port is shown in **Figure 45.1** to **Figure 45.5**.

In addition, the configuration of the multiplexed Ether ch0/ch1 is also included in the configuration diagram of the general-purpose I/O port.



Note 1. When RIIC * is selected as the peripheral module, SMT = 1 and the Schmitt trigger circuit is enabled.

Figure 45.1 P0_0 to P0_3, P5_0 to P6_4, P11_0 to P18_5 (Multiplexed Peripheral Functions)



Note 1. When ETH0 is selected as the peripheral module, the ETH_MODE register output is selected.

Figure 45.2 P1_0 (Multiplexed Ether Ch0)

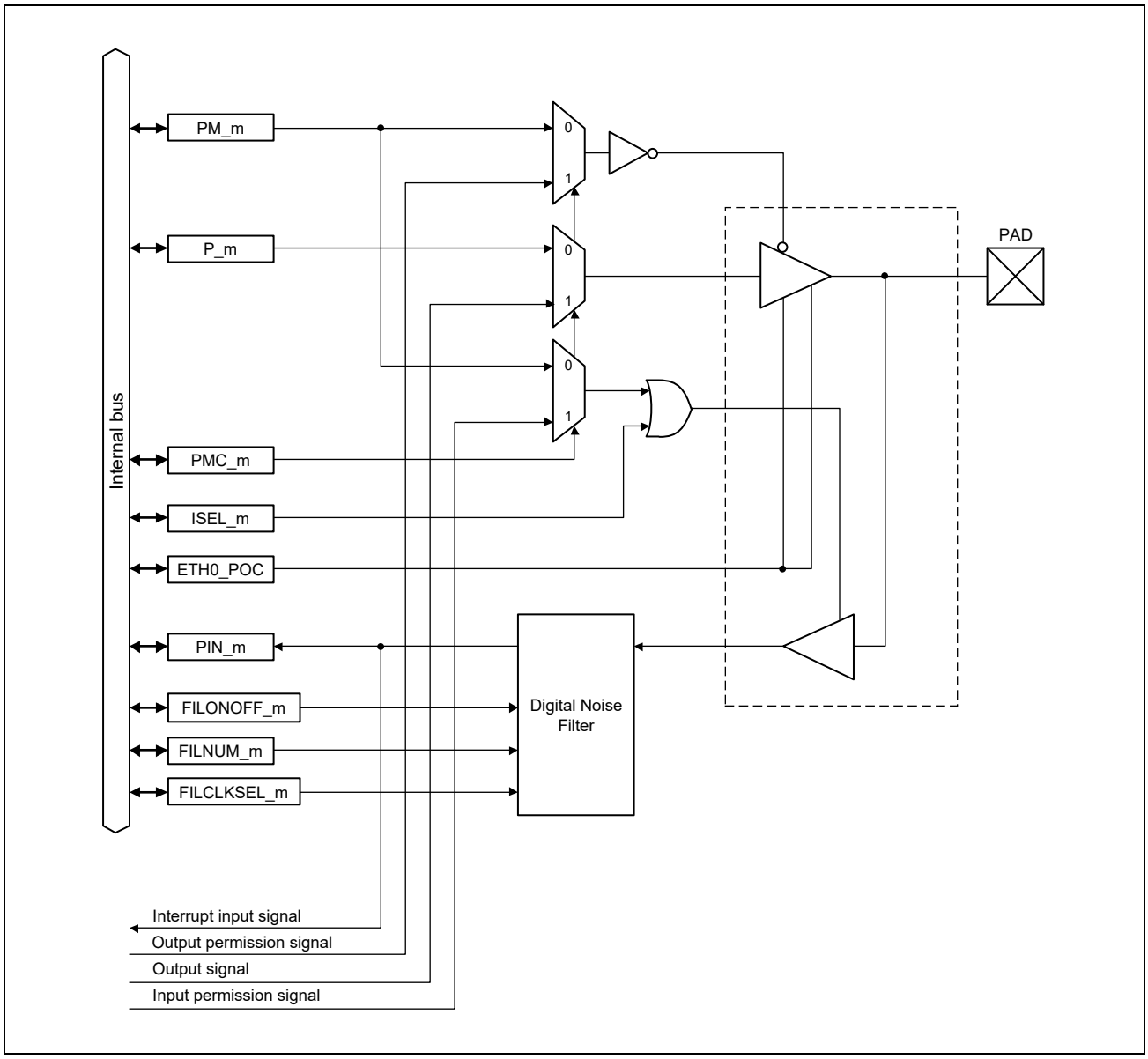
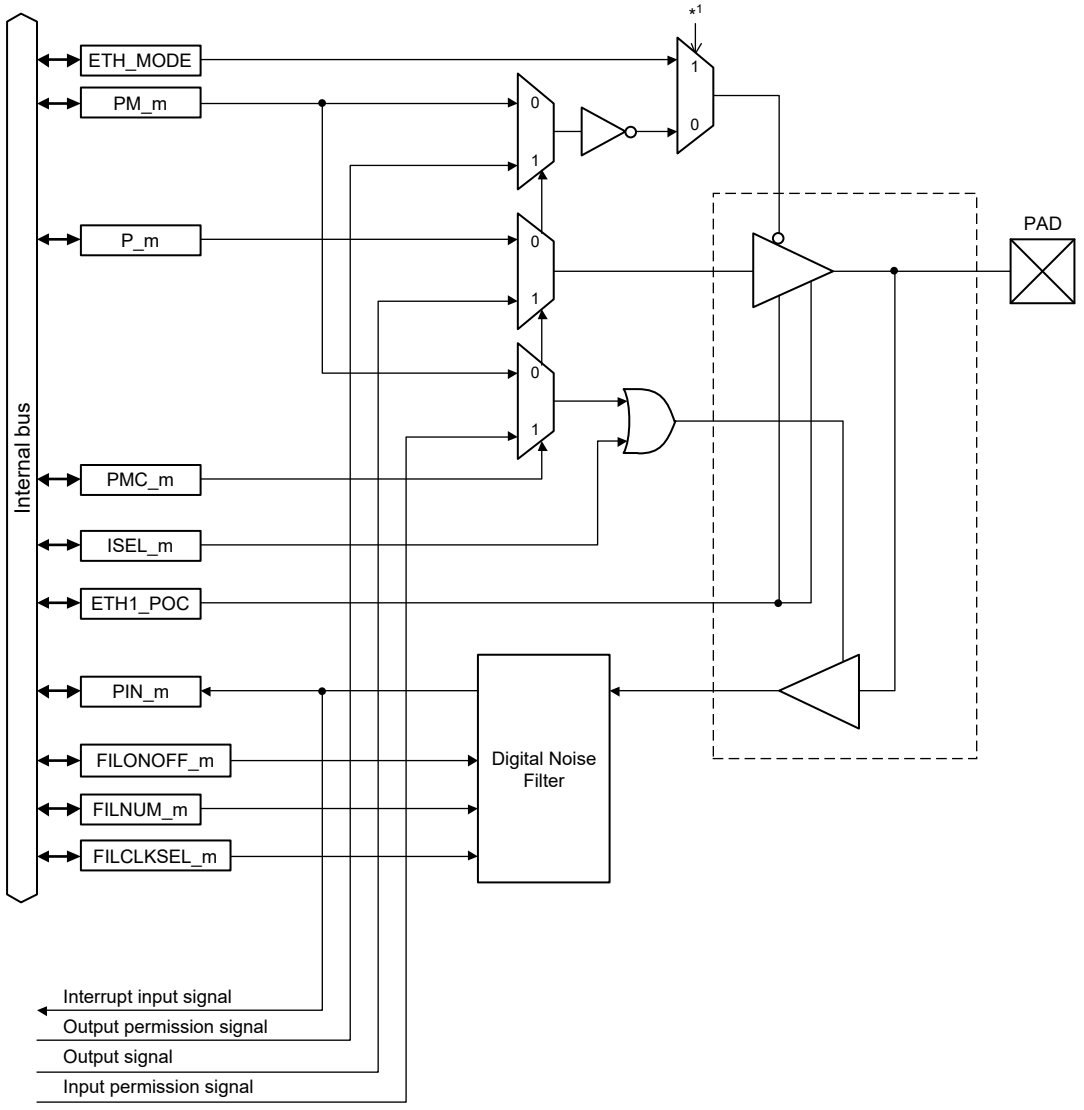


Figure 45.3 P1_1 to P4_5 (Multiplexed Ether Ch0)



Note 1. When ETH1 is selected as the peripheral module, the ETH_MODE register output is selected.

Figure 45.4 P7_0 (Multiplexed Ether Ch1)

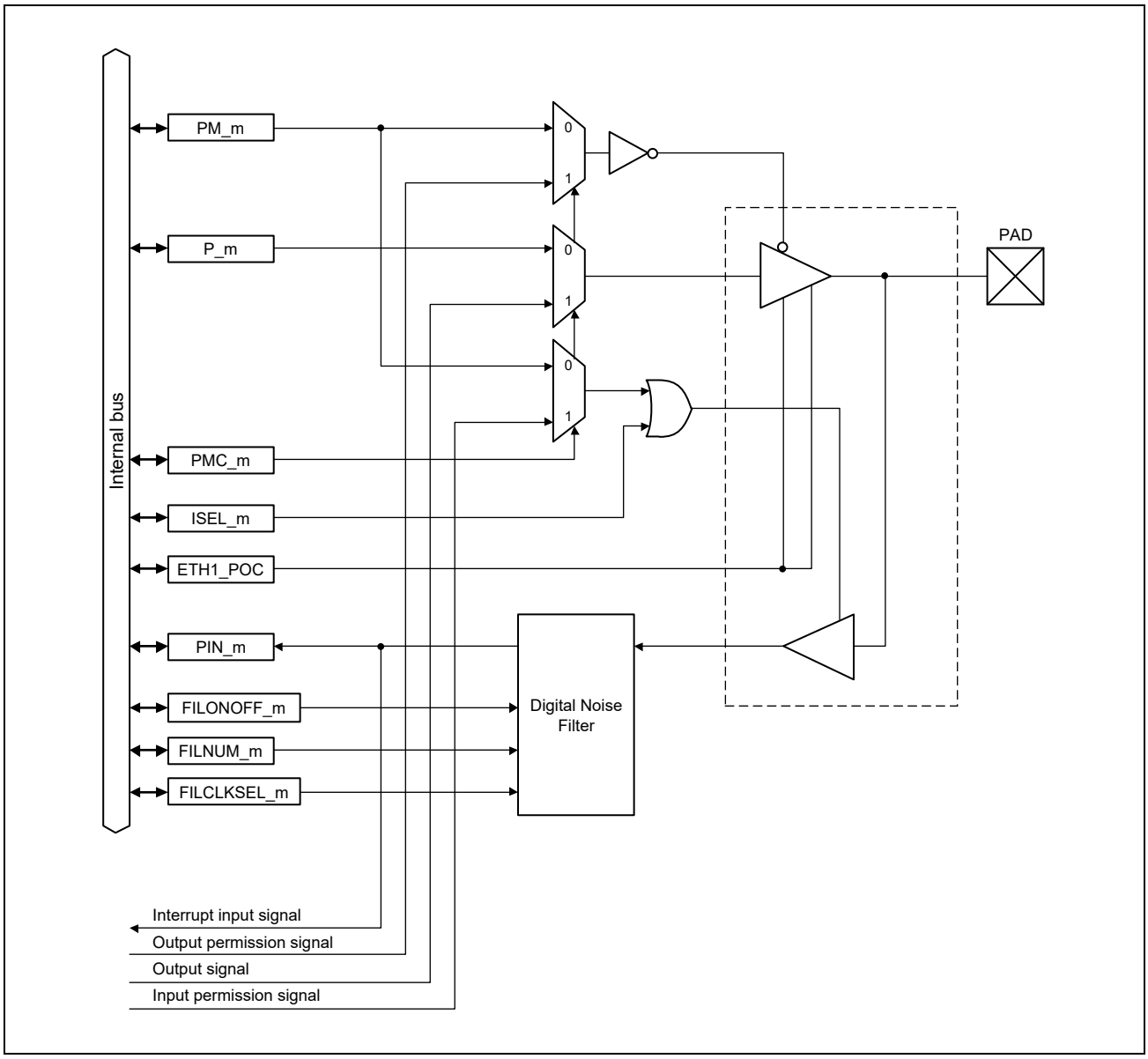
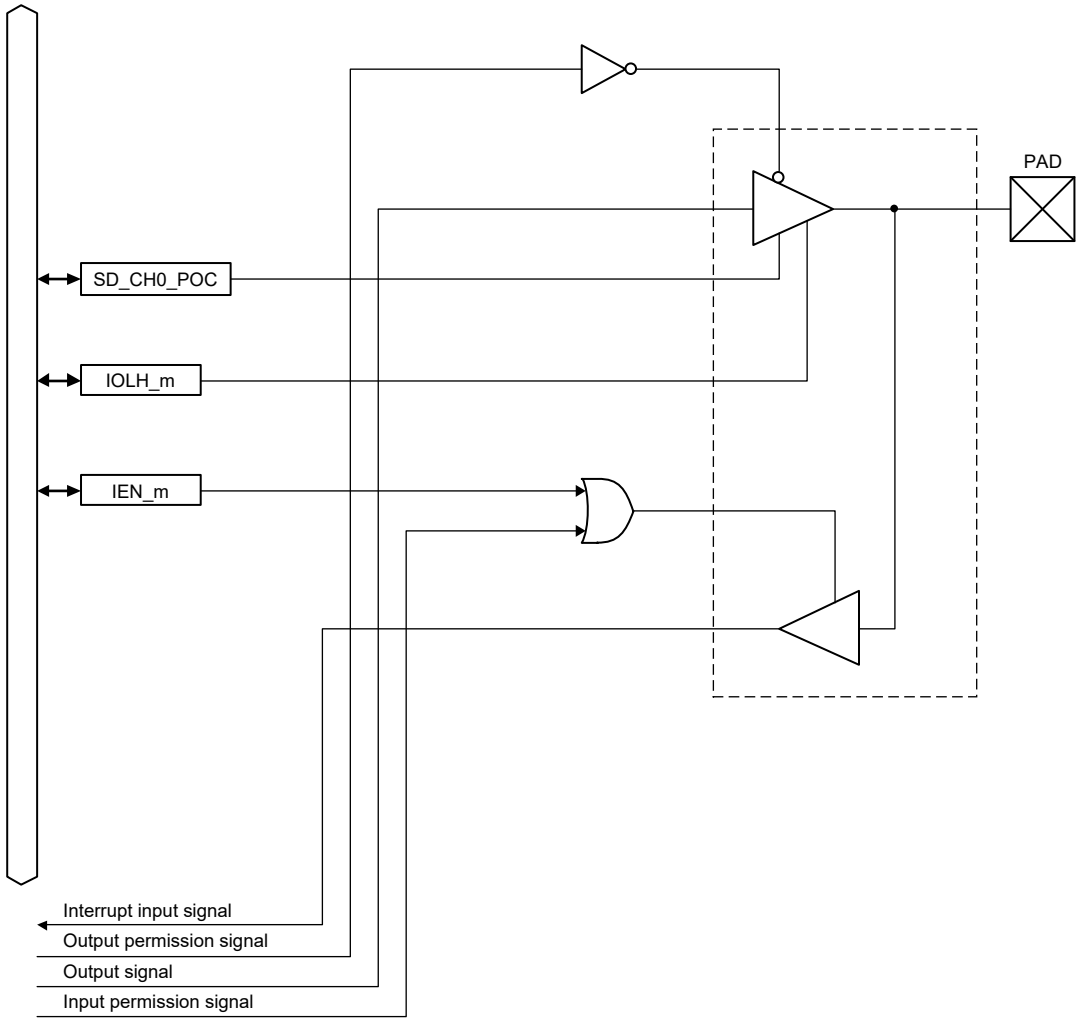


Figure 45.5 P7_1 to P10_4 (Multiplexed Ether Ch1)

45.1.5 Special Purpose Port Configuration

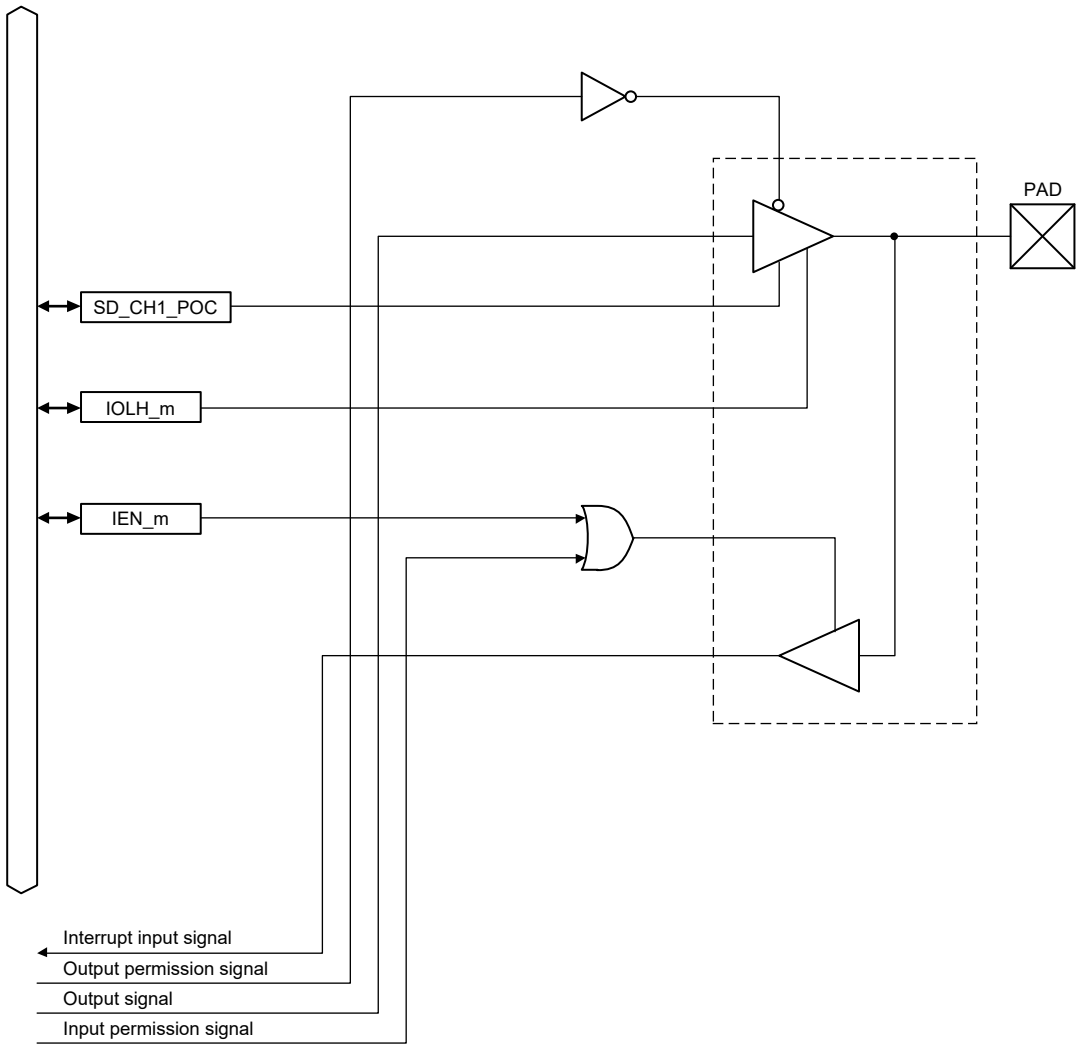
The functions and the signal of configurations shown below are shown in **Figure 45.6** to **Figure 45.12**.

- Target Functions: SDHI ch0, SDHI ch1, QSPI, RIIC
- Target Signals: NMI, TMS/SWDIO, TDO, WDTOVF_PERROUT#, AUDIO_CLK1, AUDIO_CLK2



Note: The IEN_m registers do not exist at the SD0_CLK and SD0_RST# pins.

Figure 45.6 SDHI Ch0 (SD0_CLK, SD0_CMD, SD0_RST#, SD0_DATA[7:0])



Note: The IEN_m registers do not exist at the SD1_CLK pin.

Figure 45.7 SDHI Ch1 (SD1_CLK, SD1_CMD, SD1_DATA[3:0])

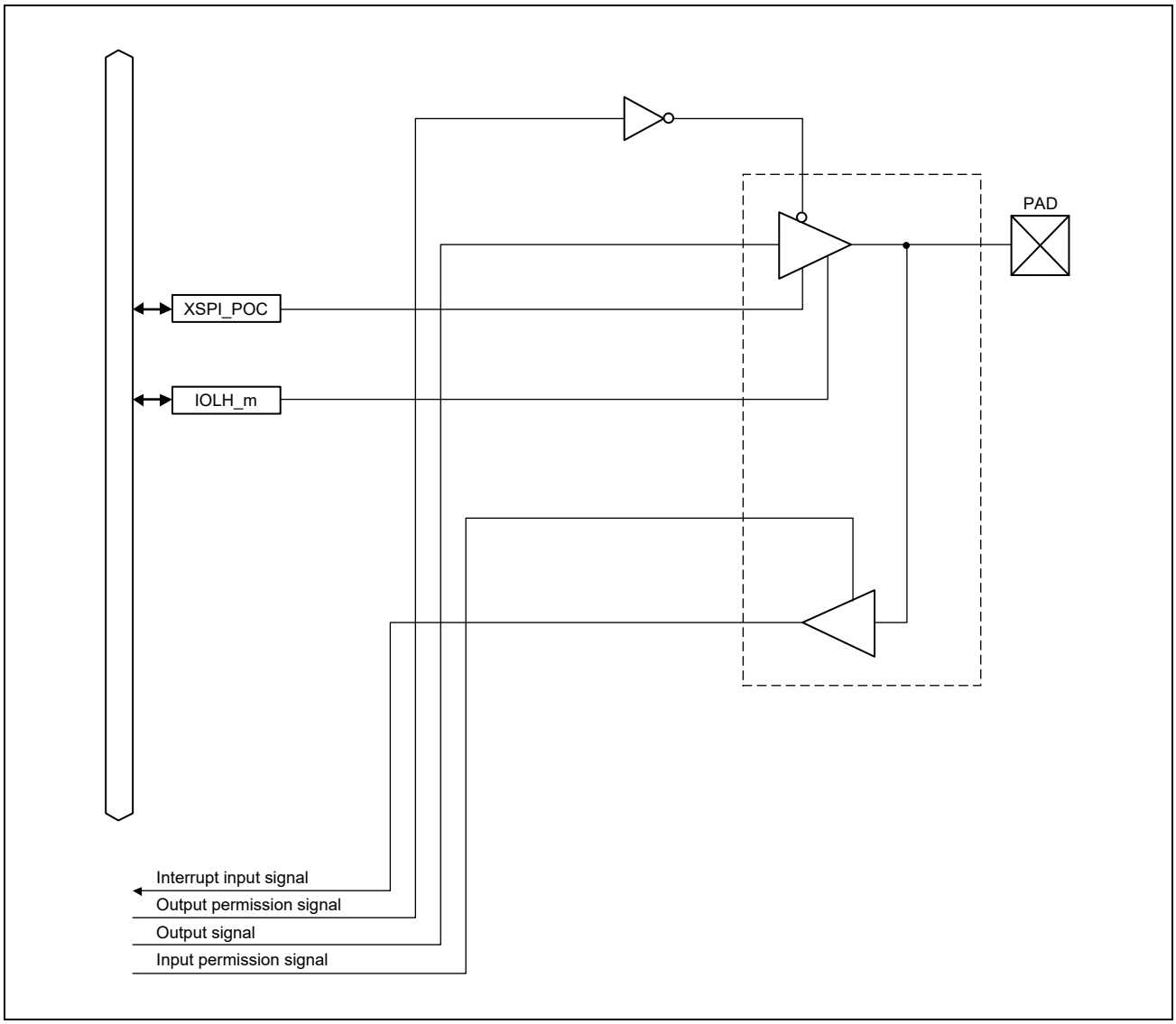


Figure 45.8 XSPI (XSPI_SPCLK, XSPI_IO[7:0], XSPI_CS[1:0]#, XSPI_DS, XSPI_RESET#, XSPI_WP#)

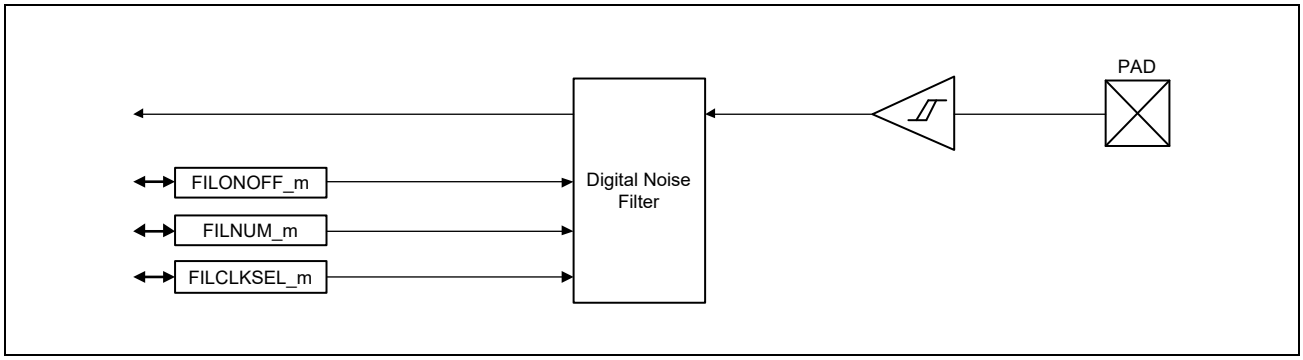


Figure 45.9 NMI

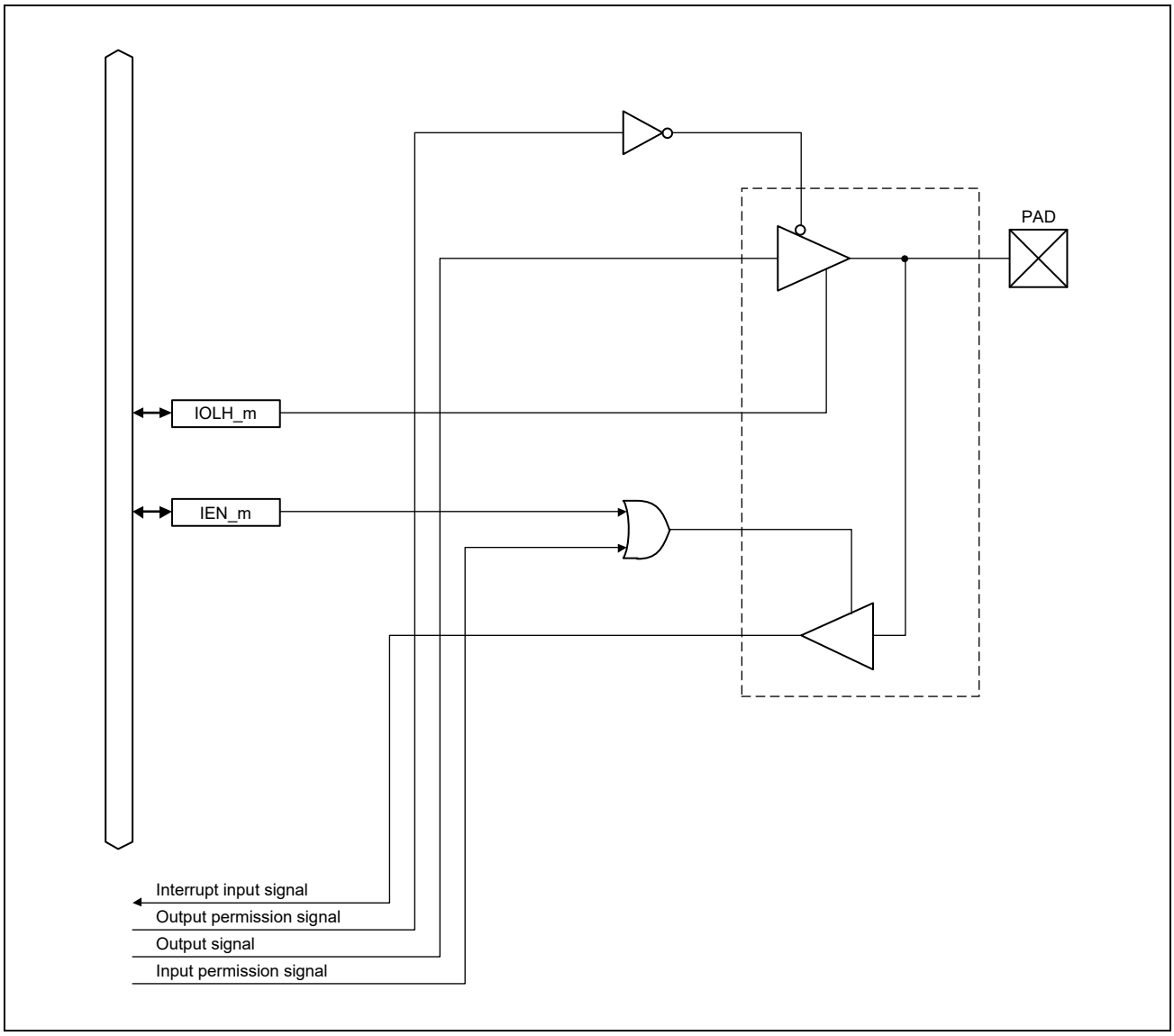


Figure 45.10 TMS/SWDIO

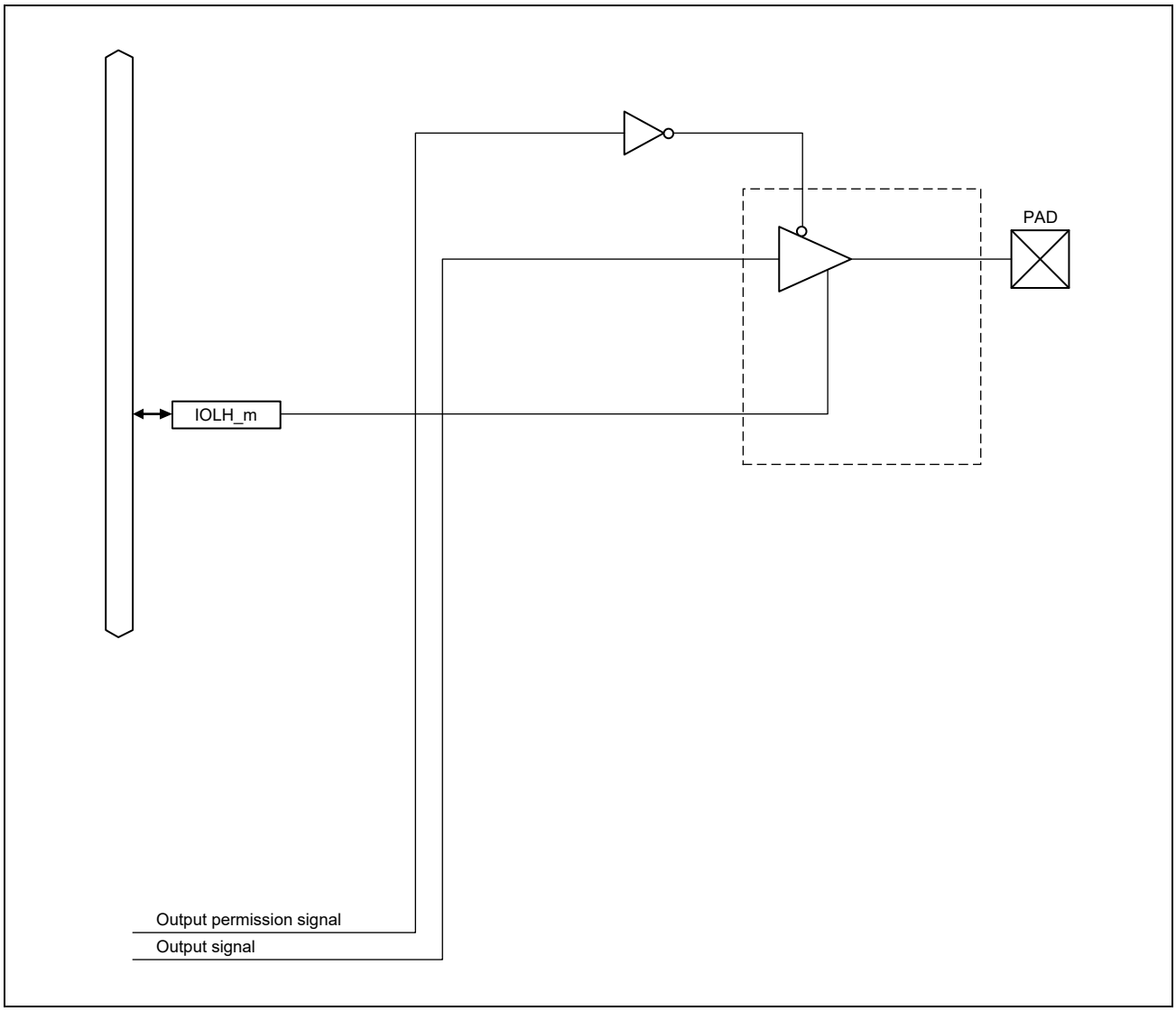


Figure 45.11 TDO, WDTOVF_PERROUT#

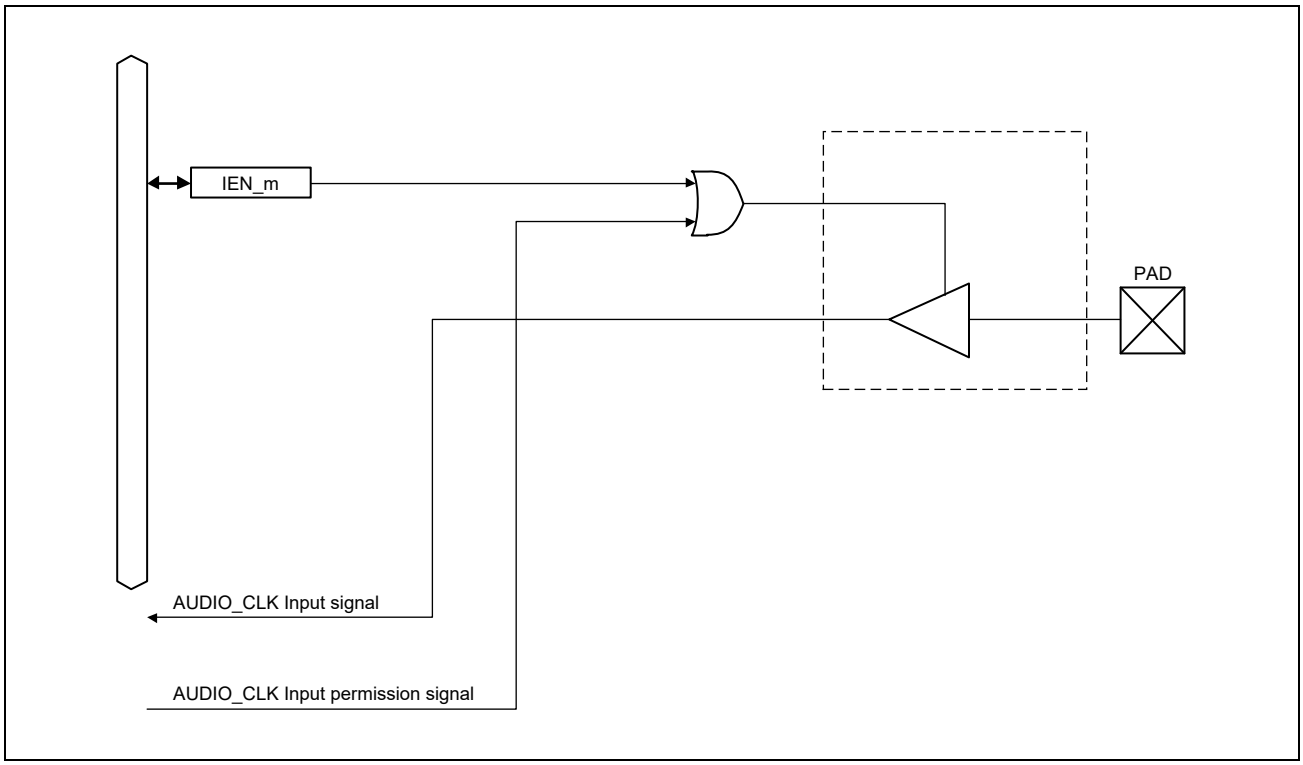


Figure 45.12 AUDIO_CLK1, AUDIO_CLK2

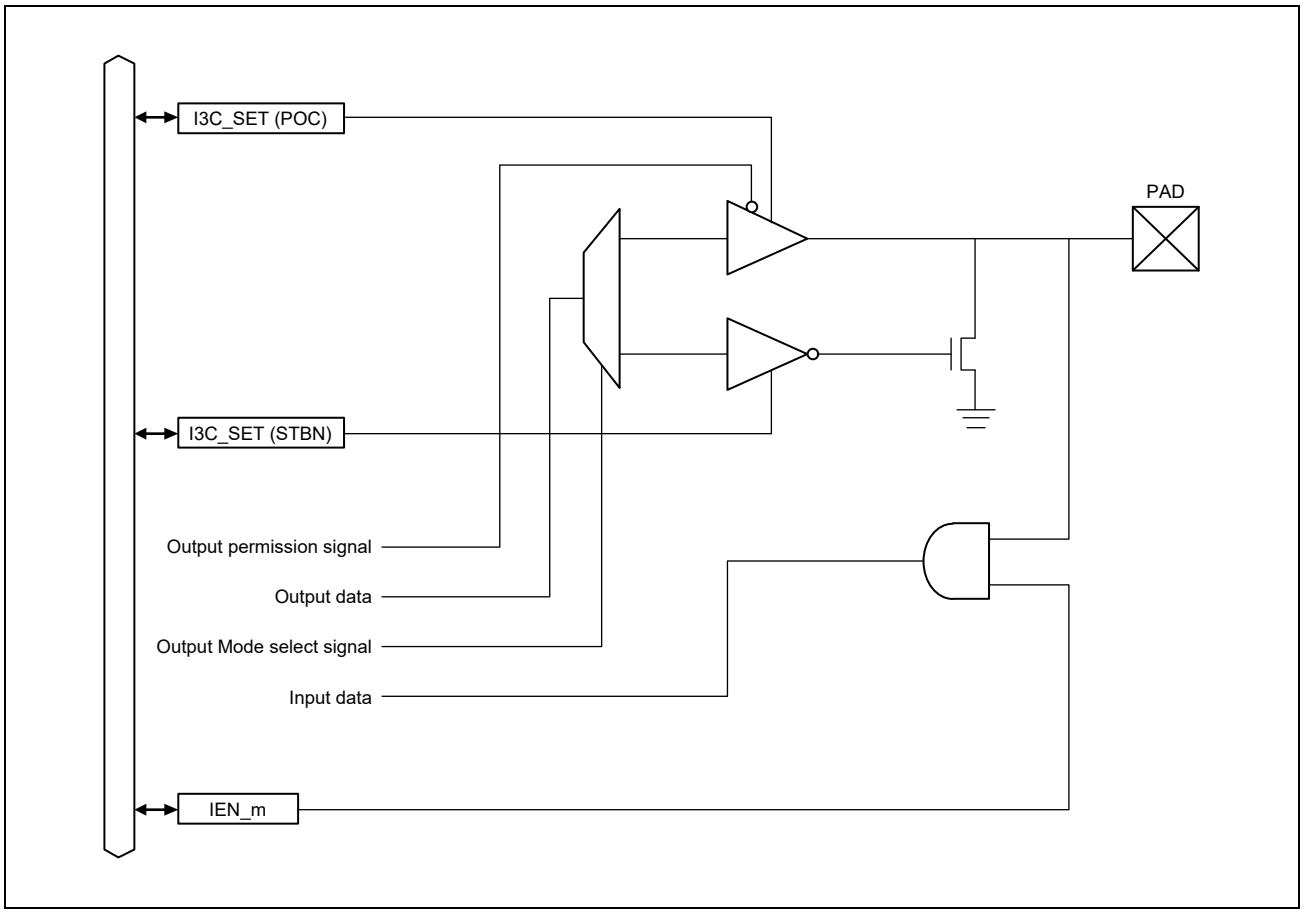


Figure 45.13 I3C (I3C_SDA, I3C_SCL)

45.2 Register Configuration

The base address is shown below.

Base address: H'0_1103_0000 (Cortex-A55 Address Space)
Base address: H'4103_0000 (Cortex-M33 Address Space Non-Secure)
Base address: H'5103_0000 (Cortex-M33 Address Space Secure)

Remark Base address of Non-Secure and Secure are exchangeable by SYS_IPCONT_IDAUZERONS register for Cortex-M33 Address Space, and SYS_IPCONT_IDAUZERONS_FPU register for Cortex-M33_FPU Address Space. The base address above are in case of the default Cortex-M33/Cortex-M33_FPU Address Space definition.

The register addresses shown in the following chapters will be offset addresses from the above base address.

NOTE

Access to areas for which Offset address is not defined in the register map is prohibited.

Please align the addresses according to the access size.

45.2.1 Port Register (P_m)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
PORT REGISTER 20	P_20	R/W	H'00	H'20	8
PORT REGISTER 21	P_21	R/W	H'00	H'21	8
PORT REGISTER 22	P_22	R/W	H'00	H'22	8
PORT REGISTER 23	P_23	R/W	H'00	H'23	8
PORT REGISTER 24	P_24	R/W	H'00	H'24	8
PORT REGISTER 25	P_25	R/W	H'00	H'25	8
PORT REGISTER 26	P_26	R/W	H'00	H'26	8
PORT REGISTER 27	P_27	R/W	H'00	H'27	8
PORT REGISTER 28	P_28	R/W	H'00	H'28	8
PORT REGISTER 29	P_29	R/W	H'00	H'29	8
PORT REGISTER 2A	P_2A	R/W	H'00	H'2A	8
PORT REGISTER 30	P_30	R/W	H'00	H'30	8
PORT REGISTER 31	P_31	R/W	H'00	H'31	8
PORT REGISTER 32	P_32	R/W	H'00	H'32	8
PORT REGISTER 33	P_33	R/W	H'00	H'33	8
PORT REGISTER 34	P_34	R/W	H'00	H'34	8
PORT REGISTER 35	P_35	R/W	H'00	H'35	8
PORT REGISTER 36	P_36	R/W	H'00	H'36	8
PORT REGISTER 37	P_37	R/W	H'00	H'37	8

45.2.2 Port Mode Register (PM_m)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
PORT MODE REGISTER 20	PM_20	R/W	H'0000	H'0140	8/16
PORT MODE REGISTER 21	PM_21	R/W	H'0000	H'0142	8/16
PORT MODE REGISTER 22	PM_22	R/W	H'0000	H'0144	8/16
PORT MODE REGISTER 23	PM_23	R/W	H'0000	H'0146	8/16
PORT MODE REGISTER 24	PM_24	R/W	H'0000	H'0148	8/16
PORT MODE REGISTER 25	PM_25	R/W	H'0000	H'014A	8/16
PORT MODE REGISTER 26	PM_26	R/W	H'0000	H'014C	8/16
PORT MODE REGISTER 27	PM_27	R/W	H'0000	H'014E	8/16
PORT MODE REGISTER 28	PM_28	R/W	H'0000	H'0150	8/16
PORT MODE REGISTER 29	PM_29	R/W	H'0000	H'0152	8/16
PORT MODE REGISTER 2A	PM_2A	R/W	H'0000	H'0154	8/16
PORT MODE REGISTER 30	PM_30	R/W	H'0000	H'0160	8/16
PORT MODE REGISTER 31	PM_31	R/W	H'0000	H'0162	8/16
PORT MODE REGISTER 32	PM_32	R/W	H'0000	H'0164	8/16
PORT MODE REGISTER 33	PM_33	R/W	H'0000	H'0166	8/16
PORT MODE REGISTER 34	PM_34	R/W	H'0000	H'0168	8/16
PORT MODE REGISTER 35	PM_35	R/W	H'0000	H'016A	8/16
PORT MODE REGISTER 36	PM_36	R/W	H'0000	H'016C	8/16
PORT MODE REGISTER 37	PM_37	R/W	H'0000	H'016E	8/16

45.2.3 Port Mode Control Register (PMC_m)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
PORT MODE CONTROL REGISTER 20	PMC_20	R/W	H'00	H'0220	8
PORT MODE CONTROL REGISTER 21	PMC_21	R/W	H'00	H'0221	8
PORT MODE CONTROL REGISTER 22	PMC_22	R/W	H'00	H'0222	8
PORT MODE CONTROL REGISTER 23	PMC_23	R/W	H'00	H'0223	8
PORT MODE CONTROL REGISTER 24	PMC_24	R/W	H'00	H'0224	8
PORT MODE CONTROL REGISTER 25	PMC_25	R/W	H'00	H'0225	8
PORT MODE CONTROL REGISTER 26	PMC_26	R/W	H'00	H'0226	8
PORT MODE CONTROL REGISTER 27	PMC_27	R/W	H'00	H'0227	8
PORT MODE CONTROL REGISTER 28	PMC_28	R/W	H'00	H'0228	8
PORT MODE CONTROL REGISTER 29	PMC_29	R/W	H'00	H'0229	8
PORT MODE CONTROL REGISTER 2A	PMC_2A	R/W	H'00	H'022A	8
PORT MODE CONTROL REGISTER 30	PMC_30	R/W	H'00	H'0230	8
PORT MODE CONTROL REGISTER 31	PMC_31	R/W	H'00	H'0231	8
PORT MODE CONTROL REGISTER 32	PMC_32	R/W	H'00	H'0232	8
PORT MODE CONTROL REGISTER 33	PMC_33	R/W	H'00	H'0233	8
PORT MODE CONTROL REGISTER 34	PMC_34	R/W	H'00	H'0234	8
PORT MODE CONTROL REGISTER 35	PMC_35	R/W	H'00	H'0235	8
PORT MODE CONTROL REGISTER 36	PMC_36	R/W	H'00	H'0236	8
PORT MODE CONTROL REGISTER 37	PMC_37	R/W	H'00	H'0237	8

45.2.4 Port Function Control Register (PFC_m)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
PORT FUNCTION CONTROL REGISTER 20	PFC_20	R/W	H'0000_0000	H'0480	8/16/32
PORT FUNCTION CONTROL REGISTER 21	PFC_21	R/W	H'0000_0000	H'0484	8/16/32
PORT FUNCTION CONTROL REGISTER 22	PFC_22	R/W	H'0000_0000	H'0488	8/16/32
PORT FUNCTION CONTROL REGISTER 23	PFC_23	R/W	H'0000_0000	H'048C	8/16/32
PORT FUNCTION CONTROL REGISTER 24	PFC_24	R/W	H'0000_0000	H'0490	8/16/32
PORT FUNCTION CONTROL REGISTER 25	PFC_25	R/W	H'0000_0000	H'0494	8/16/32
PORT FUNCTION CONTROL REGISTER 26	PFC_26	R/W	H'0000_0000	H'0498	8/16/32
PORT FUNCTION CONTROL REGISTER 27	PFC_27	R/W	H'0000_0000	H'049C	8/16/32
PORT FUNCTION CONTROL REGISTER 28	PFC_28	R/W	H'0000_0000	H'04A0	8/16/32
PORT FUNCTION CONTROL REGISTER 29	PFC_29	R/W	H'0000_0000	H'04A4	8/16/32
PORT FUNCTION CONTROL REGISTER 2A	PFC_2A	R/W	H'0000_0000	H'04A8	8/16/32
PORT FUNCTION CONTROL REGISTER 30	PFC_30	R/W	H'0000_0000	H'04C0	8/16/32
PORT FUNCTION CONTROL REGISTER 31	PFC_31	R/W	H'0000_0000	H'04C4	8/16/32
PORT FUNCTION CONTROL REGISTER 32	PFC_32	R/W	H'0000_0000	H'04C8	8/16/32
PORT FUNCTION CONTROL REGISTER 33	PFC_33	R/W	H'0000_0000	H'04CC	8/16/32
PORT FUNCTION CONTROL REGISTER 34	PFC_34	R/W	H'0000_0000	H'04D0	8/16/32
PORT FUNCTION CONTROL REGISTER 35	PFC_35	R/W	H'0000_0000	H'04D4	8/16/32
PORT FUNCTION CONTROL REGISTER 36	PFC_36	R/W	H'0000_0000	H'04D8	8/16/32
PORT FUNCTION CONTROL REGISTER 37	PFC_37	R/W	H'0000_0000	H'04DC	8/16/32

45.2.5 Port Input Register (PIN_m)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
PORT INPUT REGISTER 20	PIN_20	R	H'xx	H'0820	8
PORT INPUT REGISTER 21	PIN_21	R	H'xx	H'0821	8
PORT INPUT REGISTER 22	PIN_22	R	H'xx	H'0822	8
PORT INPUT REGISTER 23	PIN_23	R	H'xx	H'0823	8
PORT INPUT REGISTER 24	PIN_24	R	H'xx	H'0824	8
PORT INPUT REGISTER 25	PIN_25	R	H'xx	H'0825	8
PORT INPUT REGISTER 26	PIN_26	R	H'xx	H'0826	8
PORT INPUT REGISTER 27	PIN_27	R	H'xx	H'0827	8
PORT INPUT REGISTER 28	PIN_28	R	H'xx	H'0828	8
PORT INPUT REGISTER 29	PIN_29	R	H'xx	H'0829	8
PORT INPUT REGISTER 2A	PIN_2A	R	H'xx	H'082A	8
PORT INPUT REGISTER 30	PIN_30	R	H'xx	H'0830	8
PORT INPUT REGISTER 31	PIN_31	R	H'xx	H'0831	8
PORT INPUT REGISTER 32	PIN_32	R	H'xx	H'0832	8
PORT INPUT REGISTER 33	PIN_33	R	H'xx	H'0833	8
PORT INPUT REGISTER 34	PIN_34	R	H'xx	H'0834	8
PORT INPUT REGISTER 35	PIN_35	R	H'xx	H'0835	8
PORT INPUT REGISTER 36	PIN_36	R	H'xx	H'0836	8
PORT INPUT REGISTER 37	PIN_37	R	H'xx	H'0837	8

45.2.6 Interrupt Enable Control Register (ISEL_m)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
INTERRUPT ENABLE CONTROL REGISTER 20	ISEL_20	R/W	H'0000_0000	H'2D00	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 21_L	ISEL_21_L	R/W	H'0000_0000	H'2D08	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 21_H	ISEL_21_H	R/W	H'0000_0000	H'2D0C	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 22_L	ISEL_22_L	R/W	H'0000_0000	H'2D10	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 22_H	ISEL_22_H	R/W	H'0000_0000	H'2D14	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 23	ISEL_23	R/W	H'0000_0000	H'2D18	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 24	ISEL_24	R/W	H'0000_0000	H'2D20	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 25_L	ISEL_25_L	R/W	H'0000_0000	H'2D28	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 25_H	ISEL_25_H	R/W	H'0000_0000	H'2D2C	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 26	ISEL_26	R/W	H'0000_0000	H'2D30	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 27	ISEL_27	R/W	H'0000_0000	H'2D38	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 28	ISEL_28	R/W	H'0000_0000	H'2D40	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 29	ISEL_29	R/W	H'0000_0000	H'2D48	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 2A_L	ISEL_2A_L	R/W	H'0000_0000	H'2D50	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 2A_H	ISEL_2A_H	R/W	H'0000_0000	H'2D54	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 30_L	ISEL_30_L	R/W	H'0000_0000	H'2D80	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 30_H	ISEL_30_H	R/W	H'0000_0000	H'2D84	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 31	ISEL_31	R/W	H'0000_0000	H'2D88	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 32	ISEL_32	R/W	H'0000_0000	H'2D90	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 33_L	ISEL_33_L	R/W	H'0000_0000	H'2D98	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 33_H	ISEL_33_H	R/W	H'0000_0000	H'2D9C	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 34_L	ISEL_34_L	R/W	H'0000_0000	H'2DA0	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 34_H	ISEL_34_H	R/W	H'0000_0000	H'2DA4	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 35_L	ISEL_35_L	R/W	H'0000_0000	H'2DA8	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 35_H	ISEL_35_H	R/W	H'0000_0000	H'2DAC	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 36	ISEL_36	R/W	H'0000_0000	H'2DB0	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 37_L	ISEL_37_L	R/W	H'0000_0000	H'2DB8	8/16/32
INTERRUPT ENABLE CONTROL REGISTER 37_H	ISEL_37_H	R/W	H'0000_0000	H'2DBC	8/16/32

45.2.7 Driving Ability Control Register (IOLH_m)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
DRIVING ABILITY CONTROL REGISTER 01	IOLH_01	R/W	*1	H'1008	8/16/32
DRIVING ABILITY CONTROL REGISTER 04_L	IOLH_04_L	R/W	*1	H'1020	8/16/32
DRIVING ABILITY CONTROL REGISTER 04_H	IOLH_04_H	R/W	*1	H'1024	8/16/32
DRIVING ABILITY CONTROL REGISTER 05_L	IOLH_05_L	R/W	*1	H'1028	8/16/32
DRIVING ABILITY CONTROL REGISTER 05_H	IOLH_05_H	R/W	*1	H'102C	8/16/32
DRIVING ABILITY CONTROL REGISTER 06	IOLH_06	R/W	*1	H'1030	8/16/32
DRIVING ABILITY CONTROL REGISTER 10	IOLH_10	R/W	*1	H'1080	8/16/32
DRIVING ABILITY CONTROL REGISTER 11_L	IOLH_11_L	R/W	*1	H'1088	8/16/32
DRIVING ABILITY CONTROL REGISTER 11_H	IOLH_11_H	R/W	*1	H'108C	8/16/32
DRIVING ABILITY CONTROL REGISTER 12	IOLH_12	R/W	*1	H'1090	8/16/32
DRIVING ABILITY CONTROL REGISTER 13	IOLH_13	R/W	*1	H'1098	8/16/32
DRIVING ABILITY CONTROL REGISTER 20	IOLH_20	R/W	*1	H'1100	8/16/32
DRIVING ABILITY CONTROL REGISTER 21_L	IOLH_21_L	R/W	*1	H'1108	8/16/32
DRIVING ABILITY CONTROL REGISTER 21_H	IOLH_21_H	R/W	*1	H'110C	8/16/32
DRIVING ABILITY CONTROL REGISTER 22_L	IOLH_22_L	R/W	*1	H'1110	8/16/32
DRIVING ABILITY CONTROL REGISTER 22_H	IOLH_22_H	R/W	*1	H'1114	8/16/32
DRIVING ABILITY CONTROL REGISTER 23	IOLH_23	R/W	*1	H'1118	8/16/32
DRIVING ABILITY CONTROL REGISTER 24	IOLH_24	R/W	*1	H'1120	8/16/32
DRIVING ABILITY CONTROL REGISTER 25_L	IOLH_25_L	R/W	*1	H'1128	8/16/32
DRIVING ABILITY CONTROL REGISTER 25_H	IOLH_25_H	R/W	*1	H'112C	8/16/32
DRIVING ABILITY CONTROL REGISTER 26	IOLH_26	R/W	*1	H'1130	8/16/32
DRIVING ABILITY CONTROL REGISTER 27	IOLH_27	R/W	*1	H'1138	8/16/32
DRIVING ABILITY CONTROL REGISTER 28	IOLH_28	R/W	*1	H'1140	8/16/32
DRIVING ABILITY CONTROL REGISTER 29	IOLH_29	R/W	*1	H'1148	8/16/32
DRIVING ABILITY CONTROL REGISTER 2A_L	IOLH_2A_L	R/W	*1	H'1150	8/16/32
DRIVING ABILITY CONTROL REGISTER 2A_H	IOLH_2A_H	R/W	*1	H'1154	8/16/32
DRIVING ABILITY CONTROL REGISTER 30_L	IOLH_30_L	R/W	*1	H'1180	8/16/32
DRIVING ABILITY CONTROL REGISTER 30_H	IOLH_30_H	R/W	*1	H'1184	8/16/32
DRIVING ABILITY CONTROL REGISTER 31	IOLH_31	R/W	*1	H'1188	8/16/32
DRIVING ABILITY CONTROL REGISTER 32	IOLH_32	R/W	*1	H'1190	8/16/32
DRIVING ABILITY CONTROL REGISTER 33_L	IOLH_33_L	R/W	*1	H'1198	8/16/32
DRIVING ABILITY CONTROL REGISTER 33_H	IOLH_33_H	R/W	*1	H'119C	8/16/32
DRIVING ABILITY CONTROL REGISTER 34_L	IOLH_34_L	R/W	*1	H'11A0	8/16/32
DRIVING ABILITY CONTROL REGISTER 34_H	IOLH_34_H	R/W	*1	H'11A4	8/16/32
DRIVING ABILITY CONTROL REGISTER 35_L	IOLH_35_L	R/W	*1	H'11A8	8/16/32
DRIVING ABILITY CONTROL REGISTER 35_H	IOLH_35_H	R/W	*1	H'11AC	8/16/32
DRIVING ABILITY CONTROL REGISTER 36	IOLH_36	R/W	*1	H'11B0	8/16/32
DRIVING ABILITY CONTROL REGISTER 37_L	IOLH_37_L	R/W	*1	H'11B8	8/16/32
DRIVING ABILITY CONTROL REGISTER 37_H	IOLH_37_H	R/W	*1	H'11BC	8/16/32

Note 1. For the initial value, refer to **Section 45.3.7, Driving Ability Control Register (IOLH_m)**.

45.2.8 Pull-Up/Pull-Down Switching Register (PUPD_m)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
PULL UP/PULL DOWN SWITCHING REGISTER 20	PUPD_20	R/W	H'0000_0000	H'1D00	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 21_L	PUPD_21_L	R/W	H'0000_0000	H'1D08	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 21_H	PUPD_21_H	R/W	H'0000_0000	H'1D0C	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 22_L	PUPD_22_L	R/W	H'0000_0000	H'1D10	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 22_H	PUPD_22_H	R/W	H'0000_0000	H'1D14	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 23	PUPD_23	R/W	H'0000_0000	H'1D18	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 24	PUPD_24	R/W	H'0000_0000	H'1D20	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 25_L	PUPD_25_L	R/W	H'0000_0000	H'1D28	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 25_H	PUPD_25_H	R/W	H'0000_0000	H'1D2C	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 26	PUPD_26	R/W	H'0000_0000	H'1D30	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 27	PUPD_27	R/W	H'0000_0000	H'1D38	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 28	PUPD_28	R/W	H'0000_0000	H'1D40	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 29	PUPD_29	R/W	H'0000_0000	H'1D48	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 2A_L	PUPD_2A_L	R/W	H'0000_0000	H'1D50	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 2A_H	PUPD_2A_H	R/W	H'0000_0000	H'1D54	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 30_L	PUPD_30_L	R/W	H'0000_0000	H'1D80	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 30_H	PUPD_30_H	R/W	H'0000_0000	H'1D84	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 31	PUPD_31	R/W	H'0000_0000	H'1D88	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 32	PUPD_32	R/W	H'0000_0000	H'1D90	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 33_L	PUPD_33_L	R/W	H'0000_0000	H'1D98	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 33_H	PUPD_33_H	R/W	H'0000_0000	H'1D9C	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 34_L	PUPD_34_L	R/W	H'0000_0000	H'1DA0	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 34_H	PUPD_34_H	R/W	H'0000_0000	H'1DA4	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 35_L	PUPD_35_L	R/W	H'0000_0000	H'1DA8	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 35_H	PUPD_35_H	R/W	H'0000_0000	H'1DAC	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 36	PUPD_36	R/W	H'0000_0000	H'1DB0	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 37_L	PUPD_37_L	R/W	H'0000_0000	H'1DB8	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER 37_H	PUPD_37_H	R/W	H'0000_0000	H'1DBC	8/16/32

45.2.9 Write Protected Register (PWPR)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
WRITE PROTECTED REGISTER	PWPR	R/W	H'80	H'3000	8/16/32

45.2.10 Digital Noise Filter Switching Register (FILONOFF_m)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
DIGITAL NOISE FILTER SWITCHING REGISTER 00	FILONOFF_00	R/W	H'0000_0000	H'2000	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 20	FILONOFF_20	R/W	H'0000_0000	H'2100	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 21_L	FILONOFF_21_L	R/W	H'0000_0000	H'2108	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 21_H	FILONOFF_21_H	R/W	H'0000_0000	H'210C	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 22_L	FILONOFF_22_L	R/W	H'0000_0000	H'2110	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 22_H	FILONOFF_22_H	R/W	H'0000_0000	H'2114	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 23	FILONOFF_23	R/W	H'0000_0000	H'2118	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 24	FILONOFF_24	R/W	H'0000_0000	H'2120	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 25_L	FILONOFF_25_L	R/W	H'0000_0000	H'2128	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 25_H	FILONOFF_25_H	R/W	H'0000_0000	H'212C	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 26	FILONOFF_26	R/W	H'0000_0000	H'2130	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 27	FILONOFF_27	R/W	H'0000_0000	H'2138	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 28	FILONOFF_28	R/W	H'0000_0000	H'2140	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 29	FILONOFF_29	R/W	H'0000_0000	H'2148	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 2A_L	FILONOFF_2A_L	R/W	H'0000_0000	H'2150	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 2A_H	FILONOFF_2A_H	R/W	H'0000_0000	H'2154	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 30_L	FILONOFF_30_L	R/W	H'0000_0000	H'2180	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 30_H	FILONOFF_30_H	R/W	H'0000_0000	H'2184	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 31	FILONOFF_31	R/W	H'0000_0000	H'2188	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 32	FILONOFF_32	R/W	H'0000_0000	H'2190	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 33_L	FILONOFF_33_L	R/W	H'0000_0000	H'2198	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 33_H	FILONOFF_33_H	R/W	H'0000_0000	H'219C	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 34_L	FILONOFF_34_L	R/W	H'0000_0000	H'21A0	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 34_H	FILONOFF_34_H	R/W	H'0000_0000	H'21A4	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 35_L	FILONOFF_35_L	R/W	H'0000_0000	H'21A8	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 35_H	FILONOFF_35_H	R/W	H'0000_0000	H'21AC	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 36	FILONOFF_36	R/W	H'0000_0000	H'21B0	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 37_L	FILONOFF_37_L	R/W	H'0000_0000	H'21B8	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 37_H	FILONOFF_37_H	R/W	H'0000_0000	H'21BC	8/16/32

45.2.11 Digital Noise Filter Number Register (FILNUM_m)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
DIGITAL NOISE FILTER NUMBER REGISTER 00	FILNUM_00	R/W	H'0000_0000	H'2400	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 20	FILNUM_20	R/W	H'0000_0000	H'2500	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 21_L	FILNUM_21_L	R/W	H'0000_0000	H'2508	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 21_H	FILNUM_21_H	R/W	H'0000_0000	H'250C	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 22_L	FILNUM_22_L	R/W	H'0000_0000	H'2510	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 22_H	FILNUM_22_H	R/W	H'0000_0000	H'2514	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 23	FILNUM_23	R/W	H'0000_0000	H'2518	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 24	FILNUM_24	R/W	H'0000_0000	H'2520	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 25_L	FILNUM_25_L	R/W	H'0000_0000	H'2528	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 25_H	FILNUM_25_H	R/W	H'0000_0000	H'252C	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 26	FILNUM_26	R/W	H'0000_0000	H'2530	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 27	FILNUM_27	R/W	H'0000_0000	H'2538	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 28	FILNUM_28	R/W	H'0000_0000	H'2540	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 29	FILNUM_29	R/W	H'0000_0000	H'2548	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 2A_L	FILNUM_2A_L	R/W	H'0000_0000	H'2550	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 2A_H	FILNUM_2A_H	R/W	H'0000_0000	H'2554	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 30_L	FILNUM_30_L	R/W	H'0000_0000	H'2580	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 30_H	FILNUM_30_H	R/W	H'0000_0000	H'2584	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 31	FILNUM_31	R/W	H'0000_0000	H'2588	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 32	FILNUM_32	R/W	H'0000_0000	H'2590	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 33_L	FILNUM_33_L	R/W	H'0000_0000	H'2598	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 33_H	FILNUM_33_H	R/W	H'0000_0000	H'259C	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 34_L	FILNUM_34_L	R/W	H'0000_0000	H'25A0	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 34_H	FILNUM_34_H	R/W	H'0000_0000	H'25A4	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 35_L	FILNUM_35_L	R/W	H'0000_0000	H'25A8	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 35_H	FILNUM_35_H	R/W	H'0000_0000	H'25AC	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 36	FILNUM_36	R/W	H'0000_0000	H'25B0	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 37_L	FILNUM_37_L	R/W	H'0000_0000	H'25B8	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 37_H	FILNUM_37_H	R/W	H'0000_0000	H'25BC	8/16/32

45.2.12 Digital Noise Filter Clock Selection Register (FILCLKSEL_m)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 01	FILCLKSEL_00	R/W	H'0000_0000	H'2800	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 20	FILCLKSEL_20	R/W	H'0000_0000	H'2900	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 21_L	FILCLKSEL_21_L	R/W	H'0000_0000	H'2908	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 21_H	FILCLKSEL_21_H	R/W	H'0000_0000	H'290C	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 22_L	FILCLKSEL_22_L	R/W	H'0000_0000	H'2910	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 22_H	FILCLKSEL_22_H	R/W	H'0000_0000	H'2914	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 23	FILCLKSEL_23	R/W	H'0000_0000	H'2918	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 24	FILCLKSEL_24	R/W	H'0000_0000	H'2920	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 25_L	FILCLKSEL_25_L	R/W	H'0000_0000	H'2928	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 25_H	FILCLKSEL_25_H	R/W	H'0000_0000	H'292C	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 26	FILCLKSEL_26	R/W	H'0000_0000	H'2930	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 27	FILCLKSEL_27	R/W	H'0000_0000	H'2938	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 28	FILCLKSEL_28	R/W	H'0000_0000	H'2940	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 29	FILCLKSEL_29	R/W	H'0000_0000	H'2948	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 2A_L	FILCLKSEL_2A_L	R/W	H'0000_0000	H'2950	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 2A_H	FILCLKSEL_2A_H	R/W	H'0000_0000	H'2954	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 30_L	FILCLKSEL_30_L	R/W	H'0000_0000	H'2980	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 30_H	FILCLKSEL_30_H	R/W	H'0000_0000	H'2984	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 31	FILCLKSEL_31	R/W	H'0000_0000	H'2988	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 32	FILCLKSEL_32	R/W	H'0000_0000	H'2990	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 33_L	FILCLKSEL_33_L	R/W	H'0000_0000	H'2998	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 33_H	FILCLKSEL_33_H	R/W	H'0000_0000	H'299C	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 34_L	FILCLKSEL_34_L	R/W	H'0000_0000	H'29A0	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 34_H	FILCLKSEL_34_H	R/W	H'0000_0000	H'29A4	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 35_L	FILCLKSEL_35_L	R/W	H'0000_0000	H'29A8	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 35_H	FILCLKSEL_35_H	R/W	H'0000_0000	H'29AC	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 36	FILCLKSEL_36	R/W	H'0000_0000	H'29B0	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 37_L	FILCLKSEL_37_L	R/W	H'0000_0000	H'29B8	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 37_H	FILCLKSEL_37_H	R/W	H'0000_0000	H'29BC	8/16/32

45.2.13 Ether Ch0 IO Voltage Mode Control Register (ETH0_POC)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
ETHER CH0 IO VOLTAGE MODE CONTROL REGISTER	ETH0_POC	R/W	H'0000_0000	H'3010	8/16/32

45.2.14 Ether Ch1 IO Voltage Mode Control Register (ETH1_POC)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
ETHER CH1 IO VOLTAGE MODE CONTROL REGISTER	ETH1_POC	R/W	H'0000_0000	H'3014	8/16/32

45.2.15 Ether MII/RGMII Mode Control Register (ETH_MODE)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
ETHER MII/RGMII MODE CONTROL REGISTER	ETH_MODE	R/W	H'0000_000F	H'3018	8/16/32

45.2.16 SD Ch0 IO Voltage Mode Control Register (SD_CH0_POC)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
SD CH0 IO VOLTAGE MODE CONTROL REGISTER	SD_CH0_POC	R/W	*1	H'3004	8/16/32

Note 1. For the initial value, refer to **Section 45.3.16, SD Ch0 IO Voltage Mode Control Register (SD_CH0_POC)**.

45.2.17 SD Ch1 IO Voltage Mode Control Register (SD_CH1_POC)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
SD CH1 IO VOLTAGE MODE CONTROL REGISTER	SD_CH1_POC	R/W	H'0000_0000	H'3008	8/16/32

45.2.18 XSPI IO Voltage Mode Control Register (XSPI_POC)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
XSPI IO VOLTAGE MODE CONTROL REGISTER	XSPI_POC	R/W	*1	H'300C	8/16/32

Note 1. For the initial value, refer to **Section 45.3.18, XSPI IO Voltage Mode Control Register (XSPI_POC)**.

45.2.19 Input Enable Control Register (IEN_m)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
INPUT ENABLE CONTROL REGISTER 01	IEN_01	R/W	H'0000_0000	H'1808	8/16/32
INPUT ENABLE CONTROL REGISTER 02	IEN_02	R/W	H'0000_0000	H'1810	8/16/32
INPUT ENABLE CONTROL REGISTER 09	IEN_09	R/W	H'0000_0000	H'1848	8/16/32
INPUT ENABLE CONTROL REGISTER 10	IEN_10	R/W	H'0000_0000	H'1880	8/16/32
INPUT ENABLE CONTROL REGISTER 11_L	IEN_11_L	R/W	H'0000_0000	H'1888	8/16/32
INPUT ENABLE CONTROL REGISTER 11_H	IEN_11_H	R/W	H'0000_0000	H'188C	8/16/32
INPUT ENABLE CONTROL REGISTER 12	IEN_12	R/W	H'0000_0000	H'1890	8/16/32
INPUT ENABLE CONTROL REGISTER 13	IEN_13	R/W	H'0000_0000	H'1898	8/16/32
INPUT ENABLE CONTROL REGISTER 23	IEN_23	R/W	H'0000_0000	H'1918	8/16/32
INPUT ENABLE CONTROL REGISTER 24	IEN_24	R/W	H'0000_0000	H'1920	8/16/32
INPUT ENABLE CONTROL REGISTER 30	IEN_30	R/W	H'0000_0000	H'1980	8/16/32
INPUT ENABLE CONTROL REGISTER 34	IEN_34	R/W	H'0000_0000	H'19A0	8/16/32

45.2.20 I3C Control Register (I3C_SET)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
I3C Control Register	I3C_SET	R/W	H'0000_0001	H'301C	8/16/32

45.2.21 XSPI/OCTA Output Enable Control Register (XSPI/OCTA Hi-Z)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
XSPI / OCTA Output Enable Control Register	XSPI/OCTA Hi-Z	R/W	H'0000_0000	H'3020	8/16/32

45.3 Register Descriptions

45.3.1 Port Register (P_m)

The Pn register sets the GPIO output value.

For the offset address, refer to **Section 45.2.1, Port Register (P_m)**.

Bit	7	6	5	4	3	2	1	0
	P_m7	P_m6	P_m5	P_m4	P_m3	P_m2	P_m1	P_m0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	P_m7	0b	R/W	These bits set the value of the GPIO pin. 0b: Low is output. 1b: High is output. Please refer to the table below for the correspondence of each terminal.
6	P_m6	0b	R/W	
5	P_m5	0b	R/W	
4	P_m4	0b	R/W	
3	P_m3	0b	R/W	
2	P_m2	0b	R/W	
1	P_m1	0b	R/W	
0	P_m0	0b	R/W	

Table 45.1 Correspondence between Register and Each Terminal

Bit Name	bit5	bit4	bit3	bit2	bit1	bit0
P_20	—	—	P0_3	P0_2	P0_1	P0_0
P_21	—	P5_4	P5_3	P5_2	P5_1	P5_0
P_22	—	P6_4	P6_3	P6_2	P6_1	P6_0
P_23	—	—	P11_3	P11_2	P11_1	P11_0
P_24	—	—	—	—	P12_1	P12_0
P_25	—	P13_4	P13_3	P13_2	P13_1	P13_0
P_26	—	—	—	P14_2	P14_1	P14_0
P_27	—	—	P15_3	P15_2	P15_1	P15_0
P_28	—	—	—	—	P16_1	P16_0
P_29	—	—	P17_3	P17_2	P17_1	P17_0
P_2A	P18_5	P18_4	P18_3	P18_2	P18_1	P18_0
P_30	—	P1_4	P1_3	P1_2	P1_1	P1_0
P_31	—	—	P2_3	P2_2	P2_1	P2_0
P_32	—	—	P3_3	P3_2	P3_1	P3_0
P_33	P4_5	P4_4	P4_3	P4_2	P4_1	P4_0
P_34	—	P7_4	P7_3	P7_2	P7_1	P7_0
P_35	—	P8_4	P8_3	P8_2	P8_1	P8_0
P_36	—	—	P9_3	P9_2	P9_1	P9_0
P_37	—	P10_4	P10_3	P10_2	P10_1	P10_0

45.3.2 Port Mode Register (PM_m)

The PMn register sets input/output of GPIO.

For the offset address, refer to **Section 45.2.2, Port Mode Register (PM_m)**.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	PM_m7	00b	R/W	These bits set input/output of the GPIO pin. 00b: Hi-Z (Initial Value) 01b: Input Mode 10b: Output Mode (Input Disable) 11b: Output Mode (Input Enable) Please refer to the table below for the correspondence of each terminal.
13, 12	PM_m6	00b	R/W	
11, 10	PM_m5	00b	R/W	
9, 8	PM_m4	00b	R/W	
7, 6	PM_m3	00b	R/W	
5, 4	PM_m2	00b	R/W	
3, 2	PM_m1	00b	R/W	
1, 0	PM_m0	00b	R/W	

Table 45.2 Correspondence between Register and Each Terminal

Bit Name	bit11-10	bit9-8	bit7-6	bit5-4	bit3-2	bit1-0
PM_20	—	—	P0_3	P0_2	P0_1	P0_0
PM_21	—	P5_4	P5_3	P5_2	P5_1	P5_0
PM_22	—	P6_4	P6_3	P6_2	P6_1	P6_0
PM_23	—	—	P11_3	P11_2	P11_1	P11_0
PM_24	—	—	—	—	P12_1	P12_0
PM_25	—	P13_4	P13_3	P13_2	P13_1	P13_0
PM_26	—	—	—	P14_2	P14_1	P14_0
PM_27	—	—	P15_3	P15_2	P15_1	P15_0
PM_28	—	—	—	—	P16_1	P16_0
PM_29	—	—	P17_3	P17_2	P17_1	P17_0
PM_2A	P18_5	P18_4	P18_3	P18_2	P18_1	P18_0
PM_30	—	P1_4	P1_3	P1_2	P1_1	P1_0
PM_31	—	—	P2_3	P2_2	P2_1	P2_0
PM_32	—	—	P3_3	P3_2	P3_1	P3_0
PM_33	P4_5	P4_4	P4_3	P4_2	P4_1	P4_0
PM_34	—	P7_4	P7_3	P7_2	P7_1	P7_0
PM_35	—	P8_4	P8_3	P8_2	P8_1	P8_0
PM_36	—	—	P9_3	P9_2	P9_1	P9_0
PM_37	—	P10_4	P10_3	P10_2	P10_1	P10_0

45.3.3 Port Mode Control Register (PMC_m)

The PMCn register switches the mode for the multiplexed pins of GPIO.

For the offset address, refer to **Section 45.2.3, Port Mode Control Register (PMC_m)**.

Bit	7	6	5	4	3	2	1	0
	PMC_m7	PMC_m6	PMC_m5	PMC_m4	PMC_m3	PMC_m2	PMC_m1	PMC_m0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	PMC_m7	0b	R/W	These bits switch the mode for the multiplexed pin of the GPIO pin. 0b: Port Mode (GPIO) 1b: Peripheral Function Mode (Peripheral Function) Please refer to the table below for the correspondence of each terminal.
6	PMC_m6	0b	R/W	
5	PMC_m5	0b	R/W	
4	PMC_m4	0b	R/W	
3	PMC_m3	0b	R/W	
2	PMC_m2	0b	R/W	
1	PMC_m1	0b	R/W	
0	PMC_m0	0b	R/W	

Table 45.3 Correspondence between Register and Each Terminal

Bit Name	bit5	bit4	bit3	bit2	bit1	bit0
PMC_20	—	—	P0_3	P0_2	P0_1	P0_0
PMC_21	—	P5_4	P5_3	P5_2	P5_1	P5_0
PMC_22	—	P6_4	P6_3	P6_2	P6_1	P6_0
PMC_23	—	—	P11_3	P11_2	P11_1	P11_0
PMC_24	—	—	—	—	P12_1	P12_0
PMC_25	—	P13_4	P13_3	P13_2	P13_1	P13_0
PMC_26	—	—	—	P14_2	P14_1	P14_0
PMC_27	—	—	P15_3	P15_2	P15_1	P15_0
PMC_28	—	—	—	—	P16_1	P16_0
PMC_29	—	—	P17_3	P17_2	P17_1	P17_0
PMC_2A	P18_5	P18_4	P18_3	P18_2	P18_1	P18_0
PMC_30	—	P1_4	P1_3	P1_2	P1_1	P1_0
PMC_31	—	—	P2_3	P2_2	P2_1	P2_0
PMC_32	—	—	P3_3	P3_2	P3_1	P3_0
PMC_33	P4_5	P4_4	P4_3	P4_2	P4_1	P4_0
PMC_34	—	P7_4	P7_3	P7_2	P7_1	P7_0
PMC_35	—	P8_4	P8_3	P8_2	P8_1	P8_0
PMC_36	—	—	P9_3	P9_2	P9_1	P9_0
PMC_37	—	P10_4	P10_3	P10_2	P10_1	P10_0

45.3.4 Port Function Control Register (PFC_m)

The PFCm register sets multiplexed functions.

This register can be write-protected by the PWPR register.

For the offset address, refer to **Section 45.2.4, Port Function Control Register (PFC_m)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	PFC_m7			—	PFC_m6			—	PFC_m5			—	PFC_m4		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PFC_m3			—	PFC_m2			—	PFC_m1			—	PFC_m0		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 27, 23, 19, 15, 11, 7, 3	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
30 to 28	PFC_m7	000b	R/W	These bits set the multiplexed function of the GPIO pin. 000b: Function1 (Initial value) 001b: Function2 010b: Function3 011b: Function4 100b: Function5 101b: Function6 110b: Function7 111b: Function8 Please refer to the table below for the correspondence of each terminal.
26 to 24	PFC_m6	000b	R/W	
22 to 20	PFC_m5	000b	R/W	
18 to 16	PFC_m4	000b	R/W	
14 to 12	PFC_m3	000b	R/W	
10 to 8	PFC_m2	000b	R/W	
6 to 4	PFC_m1	000b	R/W	
2 to 0	PFC_m0	000b	R/W	

Note: Please refer to “Pin function list” for the function of Function (1 to 8).

Table 45.4 Correspondence between Register and Each Terminal (1/2)

Bit Name	bit22-20	bit18-16	bit14-12	bit10-8	bit6-4	bit2-0
PFC_20	—	—	P0_3	P0_2	P0_1	P0_0
PFC_21	—	P5_4	P5_3	P5_2	P5_1	P5_0
PFC_22	—	P6_4	P6_3	P6_2	P6_1	P6_0
PFC_23	—	—	P11_3	P11_2	P11_1	P11_0
PFC_24	—	—	—	—	P12_1	P12_0
PFC_25	—	P13_4	P13_3	P13_2	P13_1	P13_0
PFC_26	—	—	—	P14_2	P14_1	P14_0
PFC_27	—	—	P15_3	P15_2	P15_1	P15_0
PFC_28	—	—	—	—	P16_1	P16_0
PFC_29	—	—	P17_3	P17_2	P17_1	P17_0
PFC_2A	P18_5	P18_4	P18_3	P18_2	P18_1	P18_0
PFC_30	—	P1_4	P1_3	P1_2	P1_1	P1_0
PFC_31	—	—	P2_3	P2_2	P2_1	P2_0

Table 45.4 Correspondence between Register and Each Terminal (2/2)

Bit Name	bit22-20	bit18-16	bit14-12	bit10-8	bit6-4	bit2-0
PFC_32	—	—	P3_3	P3_2	P3_1	P3_0
PFC_33	P4_5	P4_4	P4_3	P4_2	P4_1	P4_0
PFC_34	—	P7_4	P7_3	P7_2	P7_1	P7_0
PFC_35	—	P8_4	P8_3	P8_2	P8_1	P8_0
PFC_36	—	—	P9_3	P9_2	P9_1	P9_0
PFC_37	—	P10_4	P10_3	P10_2	P10_1	P10_0

45.3.5 Port Input Register (PIN_m)

The PINn register is a read-only register to monitor input values of input pins.

For the offset address, refer to **Section 45.2.5, Port Input Register (PIN_m)**.

Bit	7	6	5	4	3	2	1	0
	PIN_m7	PIN_m6	PIN_m5	PIN_m4	PIN_m3	PIN_m2	PIN_m1	PIN_m0
Initial Value	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	PIN_m7	x	R	Input value of the input pin Please refer to the table below for the correspondence of each terminal.
6	PIN_m6	x	R	
5	PIN_m5	x	R	
4	PIN_m4	x	R	
3	PIN_m3	x	R	
2	PIN_m2	x	R	
1	PIN_m1	x	R	
0	PIN_m0	x	R	

Table 45.5 Correspondence between Register and Each Terminal

Bit Name	bit5	bit4	bit3	bit2	bit1	bit0
PIN_20	—	—	P0_3	P0_2	P0_1	P0_0
PIN_21	—	P5_4	P5_3	P5_2	P5_1	P5_0
PIN_22	—	P6_4	P6_3	P6_2	P6_1	P6_0
PIN_23	—	—	P11_3	P11_2	P11_1	P11_0
PIN_24	—	—	—	—	P12_1	P12_0
PIN_25	—	P13_4	P13_3	P13_2	P13_1	P13_0
PIN_26	—	—	—	P14_2	P14_1	P14_0
PIN_27	—	—	P15_3	P15_2	P15_1	P15_0
PIN_28	—	—	—	—	P16_1	P16_0
PIN_29	—	—	P17_3	P17_2	P17_1	P17_0
PIN_2A	P18_5	P18_4	P18_3	P18_2	P18_1	P18_0
PIN_30	—	P1_4	P1_3	P1_2	P1_1	P1_0
PIN_31	—	—	P2_3	P2_2	P2_1	P2_0
PIN_32	—	—	P3_3	P3_2	P3_1	P3_0
PIN_33	P4_5	P4_4	P4_3	P4_2	P4_1	P4_0
PIN_34	—	P7_4	P7_3	P7_2	P7_1	P7_0
PIN_35	—	P8_4	P8_3	P8_2	P8_1	P8_0
PIN_36	—	—	P9_3	P9_2	P9_1	P9_0
PIN_37	—	P10_4	P10_3	P10_2	P10_1	P10_0

45.3.6 Interrupt Enable Control Register (ISEL_m)

The ISEL register controls whether to enable or disable interrupts.

The pin set as the GPIO input port can be used as an external interrupt input. Controls whether this feature is enabled or disabled. When enabled, this pin can be used as an interrupt by setting the interrupt controller.

For the offset address, refer to **Section 45.2.6, Interrupt Enable Control Register (ISEL_m)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	ISEL_m3	—	—	—	—	—	—	—	ISEL_m2
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ISEL_m1	—	—	—	—	—	—	—	ISEL_m0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25, 23 to 17, 15 to 9, 7 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
24	ISEL_m3	0b	R/W	The ISEL register controls whether to enable or disable interrupts. 0b: Disabled (Initial value) 1b: Enabled Please refer to the table below for the correspondence of each terminal.
16	ISEL_m2	0b	R/W	
8	ISEL_m1	0b	R/W	
0	ISEL_m0	0b	R/W	

Note: Before setting this register, disable interrupt detection in the Interrupt Controller.
If PMCn bit = 1, ISEL bit = 1 is prohibited.

Table 45.6 Correspondence between Register and Each Terminal (1/2)

Bit Name	bit24	bit16	bit8	bit0
ISEL_20	P0_3	P0_2	P0_1	P0_0
ISEL_21_L	P5_3	P5_2	P5_1	P5_0
ISEL_21_H	—	—	—	P5_4
ISEL_22_L	P6_3	P6_2	P6_1	P6_0
ISEL_22_H	—	—	—	P6_4
ISEL_23	P11_3	P11_2	P11_1	P11_0
ISEL_24	—	—	P12_1	P12_0
ISEL_25_L	P13_3	P13_2	P13_1	P13_0
ISEL_25_H	—	—	—	P13_4
ISEL_26	—	P14_2	P14_1	P14_0
ISEL_27	P15_3	P15_2	P15_1	P15_0
ISEL_28	—	—	P16_1	P16_0
ISEL_29	P17_3	P17_2	P17_1	P17_0
ISEL_2A_L	P18_3	P18_2	P18_1	P18_0
ISEL_2A_H	—	—	P18_5	P18_4

Table 45.6 Correspondence between Register and Each Terminal (2/2)

Bit Name	bit24	bit16	bit8	bit0
ISEL_30_L	P1_3	P1_2	P1_1	P1_0
ISEL_30_H	—	—	—	P1_4
ISEL_31	P2_3	P2_2	P2_1	P2_0
ISEL_32	P3_3	P3_2	P3_1	P3_0
ISEL_33_L	P4_3	P4_2	P4_1	P4_0
ISEL_33_H	—	—	P4_5	P4_4
ISEL_34_L	P7_3	P7_2	P7_1	P7_0
ISEL_34_H	—	—	—	P7_4
ISEL_35_L	P8_3	P8_2	P8_1	P8_0
ISEL_35_H	—	—	—	P8_4
ISEL_36	P9_3	P9_2	P9_1	P9_0
ISEL_37_L	P10_3	P10_2	P10_1	P10_0
ISEL_37_H	—	—	—	P10_4

45.3.7 Driving Ability Control Register (IOLH_m)

This register sets the buffer drive ability of GPIO and Special Purpose Port.

For the offset address, refer to **Section 45.2.7, Driving Ability Control Register (IOLH_m)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	IOLH_m3		—	—	—	—	—	—	IOLH_m2	
Initial Value	0	0	0	0	0	0	*1	*1	0	0	0	0	0	0	*1	*1
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	IOLH_m1		—	—	—	—	—	—	IOLH_m0	
Initial Value	0	0	0	0	0	0	*1	*1	0	0	0	0	0	0	*1	*1
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26, 23 to 18, 15 to 10, 7 to 2	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
25, 24	IOLH_m3	*1	R/W	These bits set the drive Ability of each pin.
17, 16	IOLH_m2	*1	R/W	
9, 8	IOLH_m1	*1	R/W	
1, 0	IOLH_m0	*1	R/W	

	Pin Group-A		Pin Group-B		Pin Group-C		
Register Value	3.3 V	1.8 V	3.3 V	1.8 V	3.3 V	2.5 V	1.8 V
00b	1.9 mA	2.2 mA	4 mA	7 mA	4.5 mA	4.7 mA	5.2 mA
01b	4 mA	4.4 mA	6 mA	8 mA	5.2 mA	5.3 mA	6 mA
10b	8 mA	9 mA	8 mA	9 mA	5.7 mA	5.8 mA	6.55 mA
11b	9 mA	10 mA	9 mA	10 mA	6.05 mA	6.1 mA	6.8 mA

Note 1. 10b: applies to terminals of SD0_xxx, SD1_xxx, XSPI_xxx
01b: applies to other terminals except for SDx, XSPI
Please refer to the table below for the correspondence of each pin (including Pin Group).

Table 45.7 Correspondence between Register and Each Terminal (1/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0	Pin Group
IOLH_01	—	—	TDO	TMS_SWDIO	A
IOLH_04_L	XSPI_DS	XSPI_WP_N	XSPI_RESET_N	XSPI_SPCLK	C
IOLH_04_H	—	—	XSPI_CS1_N	XSPI_CS0_N	C
IOLH_05_L	XSPI_IO3	XSPI_IO2	XSPI_IO1	XSPI_IO0	C
IOLH_05_H	XSPI_IO7	XSPI_IO6	XSPI_IO5	XSPI_IO4	C
IOLH_06	—	—	—	WDTOVF_PERROUT	A
IOLH_10	—	SD0_RST_N	SD0_CMD	SD0_CLK	B
IOLH_11_L	SD0_DATA3	SD0_DATA2	SD0_DATA1	SD0_DATA0	B
IOLH_11_H	SD0_DATA7	SD0_DATA6	SD0_DATA5	SD0_DATA4	B
IOLH_12	—	—	SD1_CMD	SD1_CLK	B
IOLH_13	SD1_DATA3	SD1_DATA2	SD1_DATA1	SD1_DATA0	B
IOLH_20	P0_3	P0_2	P0_1	P0_0	A
IOLH_21_L	P5_3	P5_2	P5_1	P5_0	A
IOLH_21_H	—	—	—	P5_4	A

Table 45.7 Correspondence between Register and Each Terminal (2/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0	Pin Group
IOLH_22_L	P6_3	P6_2	P6_1	P6_0	A
IOLH_22_H	—	—	—	P6_4	A
IOLH_23	P11_3	P11_2	P11_1	P11_0	B
IOLH_24	—	—	P12_1	P12_0	B
IOLH_25_L	P13_3	P13_2	P13_1	P13_0	A
IOLH_25_H	—	—	—	P13_4	A
IOLH_26	—	P14_2	P14_1	P14_0	A
IOLH_27	P15_3	P15_2	P15_1	P15_0	A
IOLH_28	—	—	P16_1	P16_0	A
IOLH_29	P17_3	P17_2	P17_1	P17_0	A
IOLH_2A_L	P18_3	P18_2	P18_1	P18_0	A
IOLH_2A_H	—	—	P18_5	P18_4	A
IOLH_30_L	P1_3	P1_2	P1_1	P1_0	C
IOLH_30_H	—	—	—	P1_4	C
IOLH_31	P2_3	P2_2	P2_1	P2_0	C
IOLH_32	P3_3	P3_2	P3_1	P3_0	C
IOLH_33_L	P4_3	P4_2	P4_1	P4_0	C
IOLH_33_H	—	—	P4_5	P4_4	C
IOLH_34_L	P7_3	P7_2	P7_1	P7_0	C
IOLH_34_H	—	—	—	P7_4	C
IOLH_35_L	P8_3	P8_2	P8_1	P8_0	C
IOLH_35_H	—	—	—	P8_4	C
IOLH_36	P9_3	P9_2	P9_1	P9_0	C
IOLH_37_L	P10_3	P10_2	P10_1	P10_0	C
IOLH_37_H	—	—	—	P10_4	C

Note: WDTOVF_PERROUT# bit (H'1030 bit 1-0) in IOLH_m register is reset by the assertion of RST_WDTOVFN signal from CPG.

45.3.8 Pull-Up/Pull-Down Switching Register (PUPD_m)

The PUPD register sets pull-up and pull-down of the GPIO pins and the special purpose pins.

For the offset address, refer to **Section 45.2.8, Pull-Up/Pull-Down Switching Register (PUPD_m)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PUPD_m3		—	—	—	—	—	—	PUPD_m2	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PUPD_m1		—	—	—	—	—	—	PUPD_m0	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26, 23 to 18, 15 to 10, 7 to 2	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
25, 24	PUPD_m3	00b	R/W	These bits set pull-up and pull-down of each pin 00b: Neither pull-up nor pull-down is set. 01b: Pull-up is selected. 10b: Pull-down is selected. 11b: Setting prohibited Please refer to the table below for the correspondence of each terminal.
17, 16	PUPD_m2	00b	R/W	
9, 8	PUPD_m1	00b	R/W	
1, 0	PUPD_m0	00b	R/W	

Table 45.8 Correspondence between Register and Each Terminal (1/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0
PUPD_20	P0_3	P0_2	P0_1	P0_0
PUPD_21_L	P5_3	P5_2	P5_1	P5_0
PUPD_21_H	—	—	—	P5_4
PUPD_22_L	P6_3	P6_2	P6_1	P6_0
PUPD_22_H	—	—	—	P6_4
PUPD_23	P11_3	P11_2	P11_1	P11_0
PUPD_24	—	—	P12_1	P12_0
PUPD_25_L	P13_3	P13_2	P13_1	P13_0
PUPD_25_H	—	—	—	P13_4
PUPD_26	—	P14_2	P14_1	P14_0
PUPD_27	P15_3	P15_2	P15_1	P15_0
PUPD_28	—	—	P16_1	P16_0
PUPD_29	P17_3	P17_2	P17_1	P17_0
PUPD_2A_L	P18_3	P18_2	P18_1	P18_0
PUPD_2A_H	—	—	P18_5	P18_4
PUPD_30_L	P1_3	P1_2	P1_1	P1_0
PUPD_30_H	—	—	—	P1_4
PUPD_31	P2_3	P2_2	P2_1	P2_0

Table 45.8 Correspondence between Register and Each Terminal (2/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0
PUPD_32	P3_3	P3_2	P3_1	P3_0
PUPD_33_L	P4_3	P4_2	P4_1	P4_0
PUPD_33_H	—	—	P4_5	P4_4
PUPD_34_L	P7_3	P7_2	P7_1	P7_0
PUPD_34_H	—	—	—	P7_4
PUPD_35_L	P8_3	P8_2	P8_1	P8_0
PUPD_35_H	—	—	—	P8_4
PUPD_36	P9_3	P9_2	P9_1	P9_0
PUPD_37_L	P10_3	P10_2	P10_1	P10_0
PUPD_37_H	—	—	—	P10_4

45.3.9 Write Protected Register (PWPR)

The PWPR register sets write permission/prohibition for the PFC register and the PFCWE bit of this register.

For the offset address, refer to **Section 45.2.9, Write Protected Register (PWPR)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	B0WI	PFCWE	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7	B0WI	1b	R/W	1b: Writing a value to the PFCWE bit disabled 0b: Writing a value to the PFCWE bit enabled
6	PFCWE	0b	R/W	1b: Writing a value to the PFC register enabled 0b: Writing a value to the PFC register disabled
5 to 0	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

45.3.10 Digital Noise Filter Switching Register (FILONOFF_m)

This register controls whether the digital noise filter are enabled/disabled for GPIO input pin and NMI pin.

For the offset address, refer to **Section 45.2.10, Digital Noise Filter Switching Register (FILONOFF_m)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	FILONOFF_m3	—	—	—	—	—	—	—	FILONOFF_m2
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	FILONOFF_m1	—	—	—	—	—	—	—	FILONOFF_m0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25, 23 to 17, 15 to 9, 7 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
24	FILONOFF_m3	0b	R/W	These bits set the enable or the disable of Digital noise filter of each pin. 0b: No filter is used. 1b: A filter is used. Please refer to the table below for the correspondence of each terminal.
16	FILONOFF_m2	0b	R/W	
8	FILONOFF_m1	0b	R/W	
0	FILONOFF_m0	0b	R/W	

Note: Before setting this register, disable interrupt detection of the Interrupt controller.

Table 45.9 Correspondence between Register and Each Terminal (1/2)

Bit Name	bit24	bit16	bit8	bit0
FILONOFF_00	—	—	—	NMI
FILONOFF_20	P0_3	P0_2	P0_1	P0_0
FILONOFF_21_L	P5_3	P5_2	P5_1	P5_0
FILONOFF_21_H	—	—	—	P5_4
FILONOFF_22_L	P6_3	P6_2	P6_1	P6_0
FILONOFF_22_H	—	—	—	P6_4
FILONOFF_23	P11_3	P11_2	P11_1	P11_0
FILONOFF_24	—	—	P12_1	P12_0
FILONOFF_25_L	P13_3	P13_2	P13_1	P13_0
FILONOFF_25_H	—	—	—	P13_4
FILONOFF_26	-	P14_2	P14_1	P14_0
FILONOFF_27	P15_3	P15_2	P15_1	P15_0
FILONOFF_28	—	—	P16_1	P16_0
FILONOFF_29	P17_3	P17_2	P17_1	P17_0
FILONOFF_2A_L	P18_3	P18_2	P18_1	P18_0
FILONOFF_2A_H	—	—	P18_5	P18_4
FILONOFF_30_L	P1_3	P1_2	P1_1	P1_0
FILONOFF_30_H	—	—	—	P1_4
FILONOFF_31	P2_3	P2_2	P2_1	P2_0

Table 45.9 Correspondence between Register and Each Terminal (2/2)

Bit Name	bit24	bit16	bit8	bit0
FILONOFF_32	P3_3	P3_2	P3_1	P3_0
FILONOFF_33_L	P4_3	P4_2	P4_1	P4_0
FILONOFF_33_H	—	—	P4_5	P4_4
FILONOFF_34_L	P7_3	P7_2	P7_1	P7_0
FILONOFF_34_H	—	—	—	P7_4
FILONOFF_35_L	P8_3	P8_2	P8_1	P8_0
FILONOFF_35_H	—	—	—	P8_4
FILONOFF_36	P9_3	P9_2	P9_1	P9_0
FILONOFF_37_L	P10_3	P10_2	P10_1	P10_0
FILONOFF_37_H	—	—	—	P10_4

45.3.11 Digital Noise Filter Number Register (FILNUM_m)

This register sets the number of FF stages of digital noise filter for GPIO pin and NMI pin.

For the offset address, refer to **Section 45.2.11, Digital Noise Filter Number Register (FILNUM_m)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	FILNUM_m3		—	—	—	—	—	—	FILNUM_m2	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FILNUM_m1		—	—	—	—	—	—	FILNUM_m0	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26, 23 to 18, 15 to 10, 7 to 2	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
25, 24	FILNUM_m3	00b	R/W	00b: 4-stage filter (41.666 ns × 4 = 166.666 ns) (initial value) 01b: 8-stage filter (41.666 ns × 8 = 333.333 ns) 10b: 12-stage filter (41.666 ns × 12 = 500 ns) 11b: 16-stage filter (41.666 ns × 16 = 666.666 ns) *Note*: This value is the value when the external clock is 24 MHz. Please refer to the table below for the correspondence of each terminal.
17, 16	FILNUM_m2	00b	R/W	
9, 8	FILNUM_m1	00b	R/W	
1, 0	FILNUM_m0	00b	R/W	

Note: Before setting this register, disable interrupt detection of the Interrupt controller.

Table 45.10 Correspondence between Register and Each Terminal (1/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0
FILNUM_00	—	—	—	NMI
FILNUM_20	P0_3	P0_2	P0_1	P0_0
FILNUM_21_L	P5_3	P5_2	P5_1	P5_0
FILNUM_21_H	—	—	—	P5_4
FILNUM_22_L	P6_3	P6_2	P6_1	P6_0
FILNUM_22_H	—	—	—	P6_4
FILNUM_23	P11_3	P11_2	P11_1	P11_0
FILNUM_24	—	—	P12_1	P12_0
FILNUM_25_L	P13_3	P13_2	P13_1	P13_0
FILNUM_25_H	—	—	—	P13_4
FILNUM_26	-	P14_2	P14_1	P14_0
FILNUM_27	P15_3	P15_2	P15_1	P15_0
FILNUM_28	—	—	P16_1	P16_0
FILNUM_29	P17_3	P17_2	P17_1	P17_0
FILNUM_2A_L	P18_3	P18_2	P18_1	P18_0
FILNUM_2A_H	—	—	P18_5	P18_4

Table 45.10 Correspondence between Register and Each Terminal (2/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0
FILNUM_30_L	P1_3	P1_2	P1_1	P1_0
FILNUM_30_H	—	—	—	P1_4
FILNUM_31	P2_3	P2_2	P2_1	P2_0
FILNUM_32	P3_3	P3_2	P3_1	P3_0
FILNUM_33_L	P4_3	P4_2	P4_1	P4_0
FILNUM_33_H	—	—	P4_5	P4_4
FILNUM_34_L	P7_3	P7_2	P7_1	P7_0
FILNUM_34_H	—	—	—	P7_4
FILNUM_35_L	P8_3	P8_2	P8_1	P8_0
FILNUM_35_H	—	—	—	P8_4
FILNUM_36	P9_3	P9_2	P9_1	P9_0
FILNUM_37_L	P10_3	P10_2	P10_1	P10_0
FILNUM_37_H	—	—	—	P10_4

45.3.12 Digital Noise Filter Clock Selection Register (FILCLKSEL_m)

The FILCLKSEL register selects the divided clock to be input to digital noise filters.

For the offset address, refer to **Section 45.2.12, Digital Noise Filter Clock Selection Register (FILCLKSEL_m)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	FILCLKSEL_m3		—	—	—	—	—	—	FILCLKSEL_m2	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FILCLKSEL_m1		—	—	—	—	—	—	FILCLKSEL_m0	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26, 23 to 18, 15 to 10, 7 to 2	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
25, 24	FILCLKSEL_m3	00b	R/W	00b: Not divided (initial value) 01b: Divided by 9000 (41.666 ns × 9000 = 375,000 ns) 10b: Divided by 18000 (41.666 ns × 18000 = 750,000 ns) 11b: Divided by 36000 (41.666 ns × 36000 = 1,500,000 ns) *Note*: This value is the value when the external clock is 24 MHz. Please refer to the table below for the correspondence of each terminal.
17, 16	FILCLKSEL_m2	00b	R/W	
9, 8	FILCLKSEL_m1	00b	R/W	
1, 0	FILCLKSEL_m0	00b	R/W	

Note: Before setting this register, disable interrupt detection of the Interrupt controller.

Table 45.11 Correspondence between Register and Each Terminal (1/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0
FILCLKSEL_00	—	—	—	NMI
FILCLKSEL_20	P0_3	P0_2	P0_1	P0_0
FILCLKSEL_21_L	P5_3	P5_2	P5_1	P5_0
FILCLKSEL_21_H	—	—	—	P5_4
FILCLKSEL_22_L	P6_3	P6_2	P6_1	P6_0
FILCLKSEL_22_H	—	—	—	P6_4
FILCLKSEL_23	P11_3	P11_2	P11_1	P11_0
FILCLKSEL_24	—	—	P12_1	P12_0
FILCLKSEL_25_L	P13_3	P13_2	P13_1	P13_0
FILCLKSEL_25_H	—	—	—	P13_4
FILCLKSEL_26	—	P14_2	P14_1	P14_0
FILCLKSEL_27	P15_3	P15_2	P15_1	P15_0
FILCLKSEL_28	—	—	P16_1	P16_0
FILCLKSEL_29	P17_3	P17_2	P17_1	P17_0
FILCLKSEL_2A_L	P18_3	P18_2	P18_1	P18_0
FILCLKSEL_2A_H	—	—	P18_5	P18_4

Table 45.11 Correspondence between Register and Each Terminal (2/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0
FILCLKSEL_30_L	P1_3	P1_2	P1_1	P1_0
FILCLKSEL_30_H	—	—	—	P1_4
FILCLKSEL_31	P2_3	P2_2	P2_1	P2_0
FILCLKSEL_32	P3_3	P3_2	P3_1	P3_0
FILCLKSEL_33_L	P4_3	P4_2	P4_1	P4_0
FILCLKSEL_33_H	—	—	P4_5	P4_4
FILCLKSEL_34_L	P7_3	P7_2	P7_1	P7_0
FILCLKSEL_34_H	—	—	—	P7_4
FILCLKSEL_35_L	P8_3	P8_2	P8_1	P8_0
FILCLKSEL_35_H	—	—	—	P8_4
FILCLKSEL_36	P9_3	P9_2	P9_1	P9_0
FILCLKSEL_37_L	P10_3	P10_2	P10_1	P10_0
FILCLKSEL_37_H	—	—	—	P10_4

The operation of the digital noise filter will be explained using a timing chart using the external terminal P0_0 as an example.

(1) When the digital noise filter is disabled (FILONOFF = 0)

The timing chart of FILONOFF = 0 (OFF), FILNUM [1: 0] = “00” (FF4 stage), FILCLKSEL [1: 0] = “00” (without frequency division) is shown below.

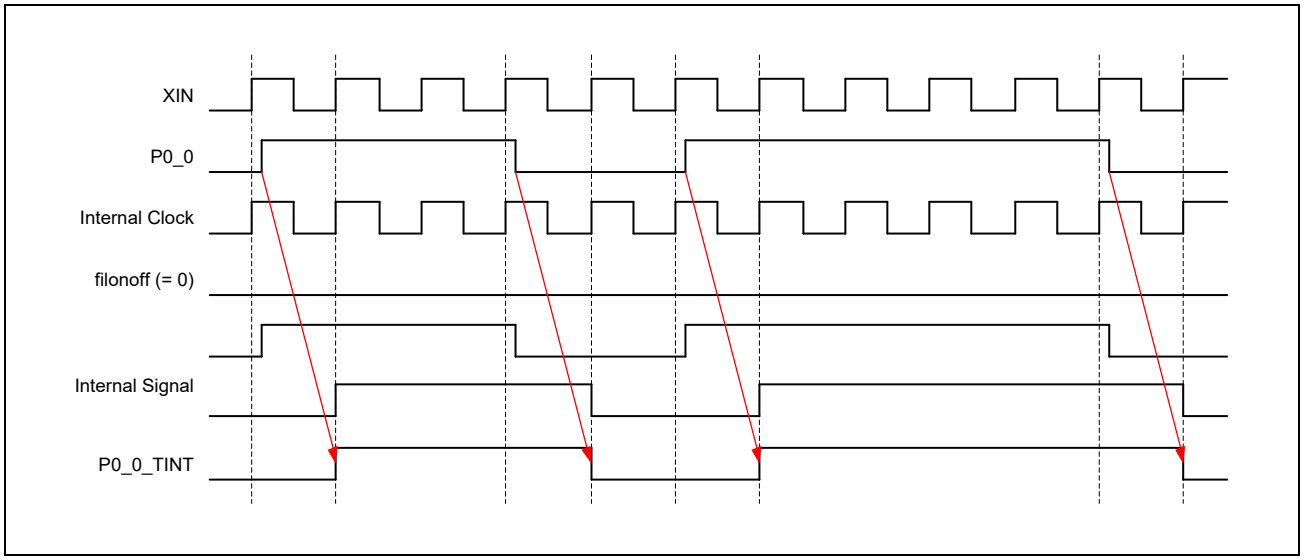


Figure 45.14 Digital Noise Filter (FILONOFF = 0) Timing Chart

The signal at the external terminal (P0_0) is not filtered regardless of the number of stages set in the FILNUM register.

The digital noise filter circuit inserts one stage of FF in the output of a digital noise filter.

Therefore, the external terminal (P0_0) signal is propagated internally with a delay of one cycle. (XIN).

(2) When the digital noise filter is enabled (FILONOFF = 1)

The timing chart of FILONOFF = 1 (ON), FILNUM [1: 0] = “00” (FF4 stage), FILCLKSEL [1: 0] = “00” (without frequency division) is shown below.

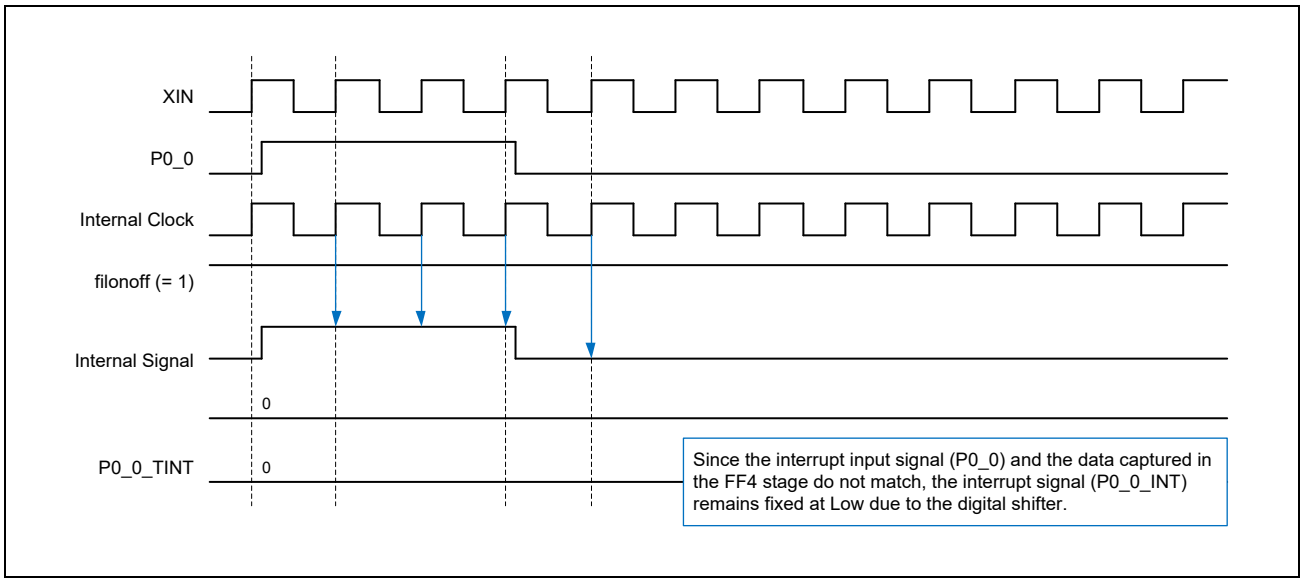


Figure 45.15 Digital Noise Filter (FILONOFF = 1) Timing Chart (1)

The signal of the external terminal (P0_0) is filtered, if even one signal captured by each FF for the number of FF stages set in the FILNUM register does not match.

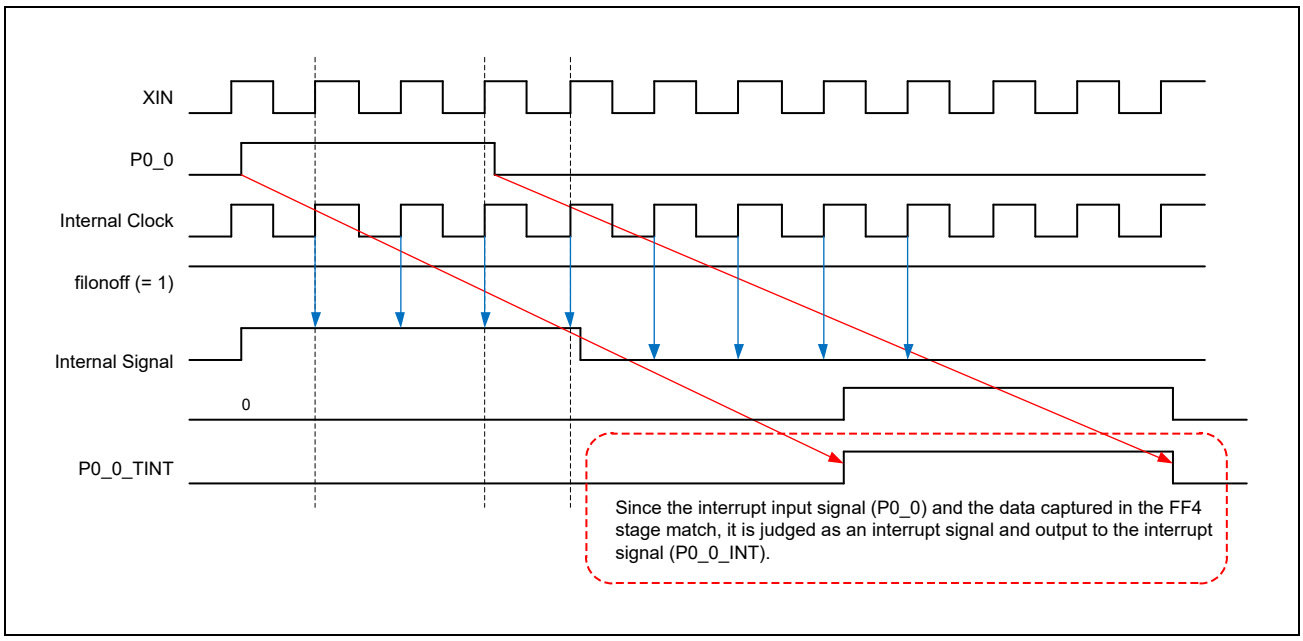


Figure 45.16 Digital Noise Filter (FILONOFF = 1) Timing Chart (2)

The external terminal (P0_0) signal is the signal captured by each FF for the number of FF stages set in the FILNUM register. If they all match, no filtering is done.

The digital noise filter circuit outputs to the interrupt signal with a delay of FILTER_CYCLE.

[FILTER_CYCLE Calculation]

FILTER_CYCLE = FILNUM_FF + $FILTER_FF

FILNUM_FF: Number of FF stages set in the FILNUM register @XIN
FILTER_FF: FF 2 stage in digital noise filter @XIN + Digital noise filter output FF 1 stage @XIN

Note: One stage of FF is inserted in the final stage of the digital noise filter circuit. Therefore, interrupt signals less than 1CLK @ XIN may be removed regardless of whether the digital noise filter is enabled or disabled.

45.3.13 Ether Ch0 Voltage Mode Control Register (ETH0_POC)

This register is sets the IO voltage mode control of Ether ch0.

For the offset address, refer to **Section 45.2.13, Ether Ch0 IO Voltage Mode Control Register (ETH0_POC)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ETH0_POC[1]	ETH0_POC[0]
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	ETH0_POC[1]	0b	R/W	These bits set the IO voltage mode.
0	ETH0_POC[0]	0b	R/W	

Bit 1	Bit 0	IO Voltage
0	0	3.3 V (Initial value)
0	1	1.8 V
1	0	2.5 V
1	1	Setting prohibited

- For the IO voltage mode of P1_0 to P4_5, set the above register according to the voltage mode of the external device to be connected.
- At Function1, the Ether ch0 function is selected, so set the value of this register according to the power supply voltage (PVDD182533_0) to be used.

  When Function2 to Function8, a function other than Ether ch0 is selected, so set the value of this register to IO voltage mode (3.3v).

  For each peripheral function, refer to Function1 to Function8 in the Pin function List.

  When using function0 (GPIO function is selected), this register can be set 3.3 V, 2.5 V or 1.8 V according to power supply voltage of VDD182533_0.

Note: When setting this register, do not set the IO voltage mode lower than the applied voltage. For example, if the applied voltage is set to 3.3 V and the register is set to 1.8 V, the reliability of this LSI will be affected.

45.3.14 Ether Ch1 Voltage Mode Control Register (ETH1_POC)

This register sets the IO voltage mode control for Etherch1.

For the offset address, refer to **Section 45.2.14, Ether Ch1 IO Voltage Mode Control Register (ETH1_POC)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ETH1_POC[1]	ETH1_POC[0]
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	ETH1_POC[1]	0b	R/W	These bits set the IO voltage mode.
0	ETH1_POC[0]	0b	R/W	

Bit 1	Bit 0	IO Voltage
0	0	3.3 V (Initial value)
0	1	1.8 V
1	0	2.5 V
1	1	Setting prohibited

- For the IO voltage mode of P7_0 to P10_4, set the above register according to the voltage mode of the external device to be connected.
- At Function1, the Ether ch1 function is selected, so set the value of this register according to the power supply voltage (PVDD182533_1) to be used.

  When Function2 to Function8, a function other than Ether ch1 is selected, so set the value of this register to IO voltage mode (3.3v).

  For each peripheral function, refer to Function1 to Function8 in the Pin function List.

  When using function0 (GPIO function is selected), this register can be set 3.3 V, 2.5 V or 1.8 V according to power supply voltage of VDD182533_1.

Note: When setting this register, do not set the IO voltage mode lower than the applied voltage. For example, if the applied voltage is set to 3.3 V and the register is set to 1.8 V, the reliability of this LSI will be affected.

45.3.15 Ether MII/RGMII Mode Control Register (ETH_MODE)

This register controls the direction (input/output) of IO blocks related to Ether MII/RGMII mode.

This register is valid only for Function 1.

For the offset address, refer to **Section 45.2.15, Ether MII/RGMII Mode Control Register (ETH_MODE)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	OEN3	OEN2	OEN1	OEN0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	OEN3	1b	R/W	0b = The Direction of the IO buffer is Output. 1b = The Direction of the IO buffer is Input. (Initial value)
2	OEN2	1b	R/W	<Bit> <Terminal name (Terminal name when using Function1)> OEN3: P7_1 (ET1_TX_CTL/TX_EN)*1 OEN2: P1_1 (ET0_TX_CTL/TX_EN)*1 OEN1: P7_0 (ET1_TXC/TX_CLK)*2 OEN0: P1_0 (ET0_TXC/TX_CLK)*2
1	OEN1	1b	R/W	
0	OEN0	1b	R/W	

Note 1. When using the terminal as output (in MII or RGMII mode), it is necessary to set OEN bit to 0 (direction is output).

Note 2. If not using the terminal (e.g. RGMII mode), it is necessary to set OEN bit to 0 (direction is output)

Note: By setting this register, the direction of some pins when using Ether's MII or RGMII can be selected.
Refer to **Section 31, Gigabit Ethernet Interface** for switching between MII and RGMII of Ether.

Note: In RGMII mode, set the external terminals shown below to port mode (PMCn = 0) and I/O disabled (PMn = 00b).

Ether Ch0		Ether Ch1	
Terminal Name Function 0	Terminal Name Function 1	Terminal Name Function 0	Terminal Name Function 1
P2_2	ET0_TX_COL	P8_2	ET1_TX_COL
P2_3	ET0_TX_CRS	P8_3	ET1_TX_CRS
P4_2	ET0_RX_ERR	P10_1	ET1_RX_ERR

45.3.16 SD Ch0 IO Voltage Mode Control Register (SD_CH0_POC)

This register sets the IO voltage mode control of SD ch0.

For the offset address, refer to **Section 45.2.16, SD Ch0 IO Voltage Mode Control Register (SD_CH0_POC)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SD0_POC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Note 1. In Boot Mode 1 and Boot Mode 2, R/W is R (Read Only).

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	SD0_POC	0b*1	R/W	This bit sets the IO voltage mode. 1b: I/O domain voltage ≤ 1.8 V 0b: I/O domain voltage ≥ 3.3 V (Initial value)*1

Note 1. The initial value of this register

- The IO voltage mode of SD0_CLK, SD0_CMD, SD0_DATA[7-0], SD0_RST# is set by selecting each mode of this register and MD_BOOT[2: 0].
  - When Boot Modes 1 is not selected: This register can be Read/Write (R/W). The initial value is the above value.
  - When Boot Modes 1 is selected: This register is Read Only (R), and the voltage mode of the selected device is as follows.

MD_BOOT[2:0]			Mode	Selection Function	Selection Device	Control Method
0	0	1	Boot Mode 1	SDHI0	eMMC (3.3 V)	By Hardware
1	0	1	Boot Mode 1	SDHI0	eMMC (1.8 V)	By Hardware
Other			SD_ch0 Control by Register			

Note: When setting this register, do not set the IO voltage mode lower than the applied voltage. For example, if the applied voltage is set to 3.3 V and the register is set to 1.8 V, the reliability of this LSI will be affected.

45.3.17 SD Ch1 IO Voltage Mode Control Register (SD_CH1_POC)

This register sets the IO voltage mode control of SD ch1.

For the offset address, refer to **Section 45.2.17, SD Ch1 IO Voltage Mode Control Register (SD_CH1_POC)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SD1_POC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	SD1_POC	0b	R/W	This bit sets the IO voltage mode. 1b: I/O domain voltage ≤ 1.8 V 0b: I/O domain voltage ≥ 3.3 V (Initial value)

Note: The IO voltage mode of SD1_CLK, SD1_CMD, SD1_DATA [3-0] is set by this register.

Note: When setting this register, do not set the IO voltage mode lower than the applied voltage. For example, if the applied voltage is set to 3.3 V and the register is set to 1.8 V, the reliability of this LSI will be affected.

45.3.18 XSPI IO Voltage Mode Control Register (XSPI_POC)

This register sets the IO voltage mode control of XSPI.

For the offset address, refer to **Section 45.2.18, XSPI IO Voltage Mode Control Register (XSPI_POC)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	XSPI_POC[1]	XSPI_POC[0]
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Note 1. In Boot Mode 2, R/W is R (Read Only).

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1,0	XSPI_POC [1:0]	00b*1	R/W	This bit sets the IO voltage mode. 00b: I/O domain voltage 3.3 V (Initial value)*1 01b: I/O domain voltage 1.8 V 10b: I/O domain voltage 2.5 V 11b: Setting prohibitted

Note 1. The initial value of this register

- The IO voltage of QSPI0_SPCLK, QSPI0_IO[3-0], QSPI0_SSL, QSPI_RESET#, QSPI_WP# is set by each mode of this register and MD_BOOT[2: 0].
  - When Boot Mode 2 is not selected: This register can be Read/Write (R/W). The initial value is the above value.
  - When Boot Mode 2 is selected: This register is Read Only (R), and the voltage mode of the selected device is as follows.

MD_BOOT[2:0]			Mode	Selection Function	Selection Device	Control method
0	1	0	Boot Mode 2	XSPI I/O Bus Controller	Single/Quad (3.3 V)	By Hardware
1	1	0	Boot Mode 2		Single/Quad/Octal (1.8 V)	By Hardware
Other			XSPI Control by register			

Note: When setting this register, do not set the IO voltage mode lower than the applied voltage. For example, if the applied voltage is set to 1.8 V and the register is set to 3.3 V, the reliability of this LSI will be affected.

45.3.19 Input Enable Control Register (IEN_m)

This register is a register that controls the input of the special purpose port.

For the offset address, refer to **Section 45.2.19, Input Enable Control Register (IEN_m)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	IEN_m3	—	—	—	—	—	—	—	IEN_m2
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	IEN_m1	—	—	—	—	—	—	—	IEN_m0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25, 23 to 17, 15 to 9, 7 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
24	IEN_m3	*1	R/W	These bits control input mode of the terminal when it is used for a dedicated pin or a peripheral function other than GPIO. 0b: Input disabled (Initial value) 1b: Input enabled Please refer to the table below for the correspondence of each terminal.
16	IEN_m2	*1	R/W	
8	IEN_m1	*1	R/W	
0	IEN_m0	*1	R/W	

Note 1. 1b: applies to terminals of TMS_SWDIO, I3C_SCL, I3C_SDA, P1_0, P7_0
0b: applies to other terminals except for TMS_SWDIO, I3C_SCL, I3C_SDA, P1_0, P7_0

Table 45.12 Correspondence between Register and Each Terminal

Bit Name	bit24	bit16	bit8	bit0
IEN_01	—	—	—	TMS_SWDIO
IEN_02	—	—	AUDIO_CLK2	AUDIO_CLK1
IEN_09	—	—	I3C_SCL	I3C_SDA
IEN_10	—	—	SD0_CMD	—
IEN_11_L	SD0_DATA3	SD0_DATA2	SD0_DATA1	SD0_DATA0
IEN_11_H	SD0_DATA7	SD0_DATA6	SD0_DATA5	SD0_DATA4
IEN_12	—	—	SD1_CMD	—
IEN_13	SD1_DATA3	SD1_DATA2	SD1_DATA1	SD1_DATA0
IEN_23	P11_3 (SD2_DATA1)*1	P11_2 (SD2_DATA0)*1	P11_1 (SD2_CMD)*1	—
IEN_24	—	—	P12_1 (SD2_DATA3)*1	P12_0 (SD2_DATA2)*1
IEN_30	—	—	—	P1_0 (ET0_TXC/TX_CLK)*1
IEN_34	—	—	—	P7_0 (ET1_TXC/TX_CLK)*1

Note 1. Regarding these terminals, this register is available when using a peripheral function described in parentheses.

45.3.20 I3C Control Register (I3C_SET)

This register is a register that controls I3C Interface.

For the offset address, refer to **Section 45.2.20, I3C Control Register (I3C_SET)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	POC	—	STBN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3, 1	—	All 0	R/W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2	POC	0	R/W	Set the voltage mode of I3C Interface. 0b: I/O domain voltage 1.8 V (Initial value) 1b: I/O domain voltage 1.2 V
0	STBN	1	R/W	Control the standby mode of I3C Interface. 0b: Standby mode*1 1b: Normal mode (Initial value)

Note 1. The output is fixed at Hi-Z and no data is transmitted to the inside even if data is input from outside. “Standby mode” is available when using I^2C mode only. (Not available when using I3C mode)

45.3.21 XSPI/OCTA Output Enable Control Register (XSPI/OCTA Hi-Z)

This register is a register that controls the output enable of XSPI /OctaRAM I/O buffer.

For the offset address, refer to **Section 45.2.21, XSPI/OCTA Output Enable Control Register (XSPI/OCTA Hi-Z)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SORST_N
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	SORST_N	0	R/W	Control the output enable of XSPI /OctaRAM Interface (XSPI_xxx pins). 0b: Control to Hi-Z (output disable) 1b: Unlock Hi-Z*1

Note 1. In this case, status of output enable (Active Low) is Low or depends on status of XSPI/OCTA controller.

45.4 Operation

The operating procedure for the general-purpose I/O port function and the operating procedure for peripheral functions are shown below.

45.4.1 Operation for GPIO Function

(1) Set the PMC_m register to “0” (initial value = 0).

(2) Set the GPIO input/output with the PM_m register.

(3) Set the GPIO output value in the P_m register.

(4) The input value of the input terminal can be detected by the PIN_m register.

(5) When setting the interrupt input using the GPIO input pin, set the ISEL_m register to “1”.

- Set the FILONOFF_m, IOLH_m and PUPD_m registers between (1) and (2) as necessary.
- Set the FILONOFF_m register when using GPIO to set a digital noise filter for interrupt input and GPIO input.
- Set the FILNUM_m and FILCLKSEL_m registers according to the purpose of use.

45.4.2 Operation for Peripheral Function

(1) Set the PWPR register to allow writing to the PFC_m register.

After setting the PWPR.B0WI bit to “0” (initial value = 1), set the PWPR.PFCWE bit to “1” (initial value = 0).

Select the required function from Functions 1-8. (Hereafter, Function1 setting example)

(2) Set PFC_m = 000b and switch to Function1.

(3) Set the PMC_m register to “1” (initial value = 0).

(4) Set the PFC_m register to write-protected. After setting the PWPR.PFCWE bit to “0”, set the PWPR.B0WI bit to “1”.

- Set the FILONOFF_m, IOLH_m and PUPD_m registers according to the purpose of use.
- Set the FILONOFF_m register when using GPIO to set a digital noise filter for the IRQ[0-7].
- Set the FILNUM_m and FILCLKSEL_m registers according to the purpose of use. And also, the NMI input can set the noise filter by the same registers.

46. Debug Interface

46.1 Overview

This LSI has a debug interface for boundary scan function and debug support for Cortex-A55, Cortex-M33 and Cortex-M33_FPU.

46.1.1 Features

The functions of the debug interface of this LSI are shown below.

- Debug Interface
  - Support JTAG and SWD

- Debugger control function (DAP function)
  - Direct control of IP without going through CPU by switching Access Port (AP)

- Trace data support
  - Cortex-A55 trace output
  - Cortex-M33 / Cortex-M33_FPU trace output
  - Buffer function of trace data by ETF
  - Trace data output to system bus by using ETR

- Boundary Scan
  - Connection test between this LSI and other LSIs on the user board

- Other functions
  - Interlocking operation of (Cortex-A55, Cortex-M33 and Cortex-M33_FPU), system counter (SYC), watchdog timer, general timer (GTM), and debug component by cross trigger
  - Support WDT counter stop control function
  - Add time information to trace data by Time Stamp (24 MHz operation)

46.1.2 Block Diagram

Figure 46.1 shows the block diagram of the debug interface, and **Figure 46.2** shows the block diagram of the Core Sight System (CST).

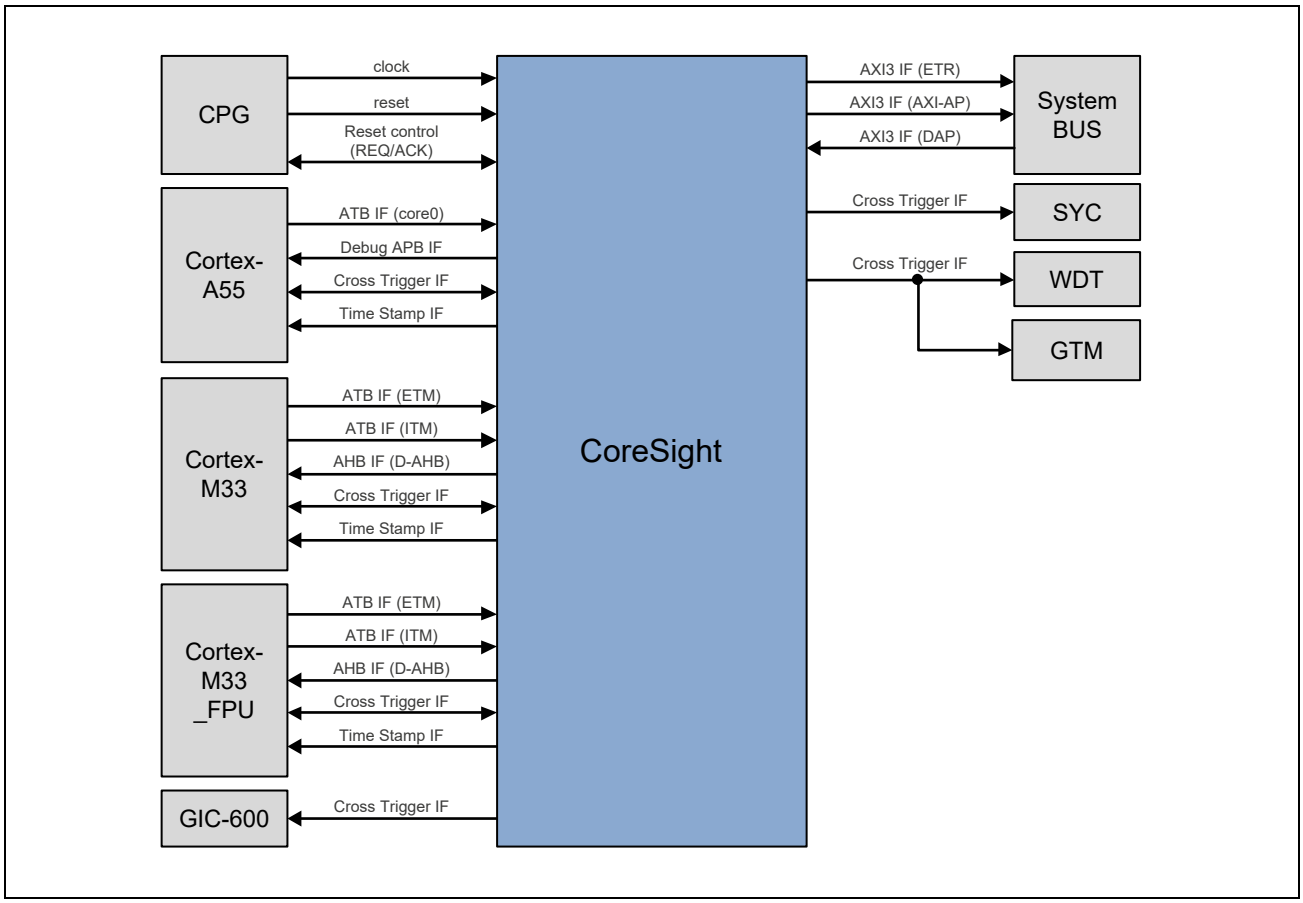


Figure 46.1 Debug System Block Diagram

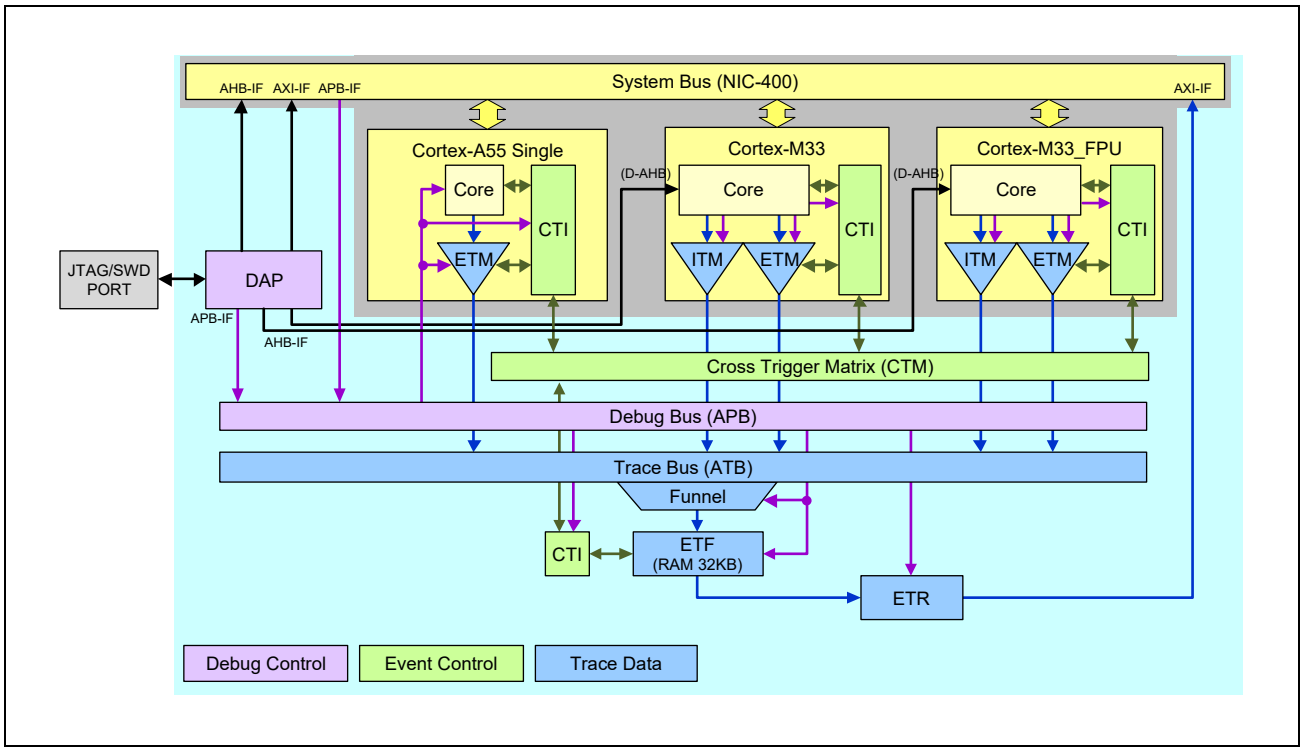


Figure 46.2 Core Sight System Block Diagram

- Debug Bus (APB)

  Bus used for controlling debug functions and acquiring debug data (register values and trace data in On-Chip Buffer)

  - DAP: Debug Access Port

- Cross Trigger Matrix (CTM)

  Functions required for co-debugging triggered by mutual events between CPU and Component

  - CTI: Cross Trigger Interface

- Trace Bus (ATB)

  A data bus which the trace data output by ETM flows

  - ETM: Embedder Trace Macrocell
  - ETR: Embedder Trace Router
  - ETF: Embedder Trace FIFO
  - Funnel: Trace data merge function

For details on ETM, ETR, and ETF, refer to “CoreSight Trace Memory Controller Technical Reference Manual”.

46.1.3 External Pins

- JTAG pins

The debug interface supports JTAG IF and SWD IF. **Table 46.1** shows the JTAG pins and their functions.

Table 46.1 JTAG Pins

Pin Name	I/O	Function		Name
		JTAG IF	SWD IF	
TCK/SWDCLK	Input	TCK	SWCLK	Test clock, Serial wire clock
TMS/SWDIO	Input, Input/Output	TMS	SWDIO	Test mode select, Serial wire data input/output
TDI	Input	TDI	—	Test data input
TDO	Output	TDO	—	Test data output
TRST#	Input	TRST	—	Test reset

- DEBUGEN pins

Table 46.2 shows the functions of the DEBUGEN pin.

Table 46.2 DEBUGEN Pins

Pin Name	I/O	Function
DEBUGEN	Input	Switches between normal operation and debug operation. 0: Normal operation 1: Debug operation

The DEBUGEN value must be fixed before releasing the power-on reset (PRST#). **Table 46.3** shows the difference in operation between normal operation and debug operation.

Table 46.3 Difference in Operation by DEBUGEN

	Normal Operation (DEBUGEN=0)	Debug Operation (DEBUGEN=1)	Related Module
Debug function module (CoreSight (other than DAP))	Boot impossible (Reset state)	Boot (Refer **Section 46.6.1, Debug Related Reset** for reset condition and range)	CPG (Reset control)

- BSCANP pins

Table 46.4 shows the functions of the BSCANP pin.

Table 46.4 BSCANP Pins

Pin Name	I/O	Function
BSCANP	Input	Switches between normal operation and boundary scan test mode. 0: Normal operation 1: Boundary scan test mode

Boundary scan test mode is a mode in which a connection test is performed between the terminals of this LSI and other ICs connected on the customer's board and is performed when BSCANP = 1.

The actual access is controlled by the 5 terminals of JTAG when BSCANP = 1.

Table 46.5 shows the operation modes of this LSI according to the values of the DEBUGEN pin and BSCANP pin.

Table 46.5 Operating Mode

Pin Name		
BSCANP	DEBUGEN	Operating Mode
0	0	Normal operation
0	1	Debug operation
1	*	Boundary scan test mode

46.1.4 Treatment of Unused Pins

When the external terminal of the debug interface is not used, handle it as shown in **Table 46.6**.

Table 46.6 Handling of Unused Pins

Pin Name	Handle*1
TCK/SWDCLK	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level)
TMS/SWDIO	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level)
TDI	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level)
TDO	Open-circuit
TRST#*2	Fix this pin at a low level (pull down or connect to the ground level)
DEBUGEN	Fix this pin at a low level (pull down or connect to the ground level)
BSCANP	Fix this pin at a low level (pull down or connect to the ground level)

Note 1. Mount the pull-up/pull-down resistors on the board side.

Note 2. TRST# should be fixed at low level in non-debug mode, and TRST# should be changed from low level to high level in debug mode.

46.2 Address Space

The CoreSight of this LSI has a 4 MB debug system address space. The address space of the debug system is shown in **Table 46.7**.

Table 46.7 Address Space of the Debug System

Address	Component*1
CoreSight Debug Components (Overall Address Space of this LSI)	
H'0_10C0_0000 ~ H'0_10C0_FFFF	CoreSight: ROM table
H'0_10C1_0000 ~ H'0_10C1_FFFF	CoreSight: Timestamp generator
H'0_10C2_0000 ~ H'0_10C2_FFFF	CoreSight: ETF
H'0_10C3_0000 ~ H'0_10C3_FFFF	CoreSight: ETR
H'0_10C4_0000 ~ H'0_10C4_FFFF	CoreSight: Trace Funnel
H'0_10C5_0000 ~ H'0_10C5_FFFF	CoreSight: CTI0
H'0_10C6_0000 ~ H'0_10C6_FFFF	CoreSight: CTI1
H'0_10C7_0000 ~ H'0_10CE_FFFF	Reserved
H'0_10CF_0000 ~ H'0_10CF_FFFF	Reserved
Reserved	
H'0_10D0_0000 ~ H'0_10DF_FFFF	Reserved
Cortex-A55 Debug Components (Overall Address Space of this LSI)	
H'0_10E0_0000 ~ H'0_10E0_FFFF	Cortex-A55: ROM table
H'0_10E1_0000 ~ H'0_10E1_FFFF	Cortex-A55: Core0 Debug
H'0_10E2_0000 ~ H'0_10E2_FFFF	Cortex-A55: Core0 CTI
H'0_10E3_0000 ~ H'0_10E3_FFFF	Cortex-A55: Core0 PMU
H'0_10E4_0000 ~ H'0_10E4_FFFF	Cortex-A55: Core0 ETM
H'0_10E5_0000 ~ H'0_10ED_FFFF	Reserved
H'0_10EE_0000 ~ H'0_10EE_FFFF	Cortex-A55: Core0 DSU CTI
H'0_10EF_0000 ~ H'0_10FF_FFFF	Reserved
Cortex-M33/Cortex-M33_FPU Debug Components	
H'E000_0000 ~ H'E000_0FFF	Cortex-M33/Cortex-M33_FPU: ITM
H'E000_1000 ~ H'E000_1FFF	Cortex-M33/Cortex-M33_FPU: DWT
H'E000_2000 ~ H'E000_2FFF	Cortex-M33/Cortex-M33_FPU: FPB
H'E000_3000 ~ H'E000_DFFF	Reserved
H'E000_E000 ~ H'E000_EFFF	Cortex-M33/Cortex-M33_FPU: Secure SCS
H'E000_F000 ~ H'E002_DFFF	Reserved
H'E002_E000 ~ H'E002_EFFF	Cortex-M33/Cortex-M33_FPU: Non-Secure SCS
H'E002_F000 ~ H'E004_0FFF	Reserved
H'E004_1000 ~ H'E004_1FFF	Cortex-M33/Cortex-M33_FPU: ETM
H'E004_2000 ~ H'E004_2FFF	Cortex-M33/Cortex-M33_FPU: CTI
H'E004_3000 ~ H'E00F_EFFF	Reserved
H'E00F_F000 ~ H'E00F_FFFF	Cortex-M33/Cortex-M33_FPU: ROM Table

Note 1. Access to the Reserved area is prohibited. Operation is not guaranteed when accessed.

NOTE

- The CoreSight Debug Components area and Cortex-A55 Debug Components area can be accessed via the system bus and APB-AP.
  APB-AP is connected to port-0 of DAP. For the port configuration of DAP, refer to **Section 46.6.2, DAP**.
- Each component is placed on a 64KB address boundary according to the ARMv8 Debug memory map. As an entity, the first 4 KB is valid.

46.3 Register Descriptions

46.3.1 ID Register (BSID)

The BSID register is a 32-bit register that cannot be accessed from the CPU. When the BSCANP pin is 1, it can be read from the TDO pin by setting the IDCODE command on the TAP controller. It cannot be written.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BSID[31:16]															
Initial Value	0	0	0	0	1	0	0	0	0	1	0	1	1	1	1	0
R/W	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSID[15:0]															
Initial Value	0	0	0	0	0	1	0	0	0	1	0	0	0	1	1	1
R/W	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BSID[31:0]	H'085E_0447	—	This is the ID register specified in JTAG. The upper 4 bits (bit31-28) may change depending on the chip version.

46.3.2 Other Registers

For details on each register other than the BSID register, refer to "ARM CoreSight SoC-400 Technical Reference Manual" and "CoreSight Trace Memory Controller Technical Reference Manual".

46.4 Debug Connection

46.4.1 Debug Connection Mode

There are the following four modes for the debug connection method of this LSI.

Authentication Mode	Method
No authentication	Allow access from the debugger without authentication (initial setting)
SEC authentication*1	Allow access from the debugger after authentication is OK. (authentication method is security authentication by security IP)
Authentication refusal*1	Block access from the debugger

Note 1. SEC authentication and Authentication refusal can only be authenticated with secure products.

For SEC authentication modes, enter the authentication code from the debugger is needed.

For information on the authentication function of secure products, please contact our sales.

46.5 Debug Access Range

The debug access range by CoreSight is for all slaves as well as Cortex-A55. For details, refer to **Section 5.3, Accessible Areas**.

46.6 Operation

46.6.1 Debug Related Reset

Please refer to **Section 7.3.2.1, Range of Reset Application** for debug-related reset and reset range.

46.6.2 DAP

DAP is equipped with SWJ-DP as Debug Port (DP) and APB-AP, AXI-AP, AHB-AP as Access Port (AP).

The DAP port numbers are shown in **Table 46.8**. Indicates the correspondence between the port number specified in the DP SELECT register and the AP, and its usage.

Table 46.8 DAP Port Number

Port number	AP	Usage
Port-0	APB-AP	Debug Component recognition and settings
Port-1	AXI-AP	System IP control (access to system bus)
Port-3	AHB-AP	Cortex-M33 settings
Port-4	AHB-AP	Cortex-M33_FPU settings

46.6.3 Trace

46.6.3.1 Data Flow

In CoreSight, all trace data is aggregated once by Trace Funnel, and stored in ETF, or output to the system bus via ETR in the subsequent stage. Setting this port number to the Trace Funnel according to the route of the trace data is needed. The Trace Funnel port numbers are shown in **Table 46.9**.

Table 46.9 Trace Funnel Port Number

Port Number	Source Components
Port-0	Cortex-A55: Core0 ETM
Port-1	Cortex-M33: ITM
Port-2	Cortex-M33: ETM
Port-3	Cortex-M33_FPU: ITM
Port-4	Cortex-M33_FPU: ETM

46.6.3.2 Timestamp

If time information want to be included in the trace data, the Timestamp Generator can be used. The timestamp replicator port numbers are shown in **Table 46.10**. The Timestamp Generator runs on a 24-MHz clock.

Table 46.10 Timestamp Replicator Port Number

Port Number	Target Components
Port-0	Cortex-A55: Core0 ETM
Port-1	Cortex-M33
Port-2	Cortex-M33_FPU

46.6.4 Cross Trigger

The CTI (Cross Trigger Interface) implemented in CoreSight, Cortex-A55, Cortex-M33 and Cotex-M33_FPU is used to communicate debug events.

46.6.4.1 CoreSight Cross Trigger Connection

CoreSight has two CTIs. The connection specifications are shown in **Table 46.11** to **Table 46.14**.

Table 46.11 CoreSight CTI0 Trigger Input

Trigger Input Bit	Source Components	Description
[7]	(Unused)	—
[6]	(Unused)	—
[5]	(Unused)	—
[4]	(Unused)	—
[3]	ETR	ETR Event Output 1 (FULL)
[2]	ETR	ETR Event Output 0 (ACQCOMP)
[1]	ETF	ETF Event Output 1 (FULL)
[0]	ETF	ETF Event Output 0 (ACQCOMP)

Table 46.12 CoreSight CTI0 Trigger Output

Trigger Output Bit	Destination Components	Description
[7]	SYC	SYC Event input1 (RESTART REQ/ACK)
[6]	SYC	SYC Event input0 (HALT REQ)
[5]	WDT/GTM	WDT/GTM Event input1 (RESTART REQ/ACK)
[4]	WDT/GTM	WDT/GTM Event input0 (HALT REQ)
[3]	ETR	ETR Event Input 1 (TRIGIN)
[2]	ETR	ETR Event Input 0 (FLUSHIN)
[1]	ETF	ETF Event Input 1 (TRIGIN)
[0]	ETF	ETF Event Input 0 (FLUSHIN)

Table 46.13 CoreSight CTI1 Trigger Input

Trigger Input Bit	Source Components	Description
[7]	(Unused)	—
[6]	(Unused)	—
[5]	(Unused)	—
[4]	(Unused)	—
[3]	(Unused)	—
[2]	(Unused)	—
[1]	(Unused)	—
[0]	(Unused)	—

Table 46.14 CoreSight CTI1 trigger output

Trigger Output Bit	Destination Components	Description
[7]	(Unused)	—
[6]	(Unused)	—
[5]	(Unused)	—
[4]	(Unused)	—
[3]	(Unused)	—
[2]	(Unused)	—
[1]	Cortex-A55	Cortex-A55 Event Input (PMUSNAPSHOTREQ/PMUSNAPSHOTACK)
[0]	GIC-600	GIC-600 Event Input (sample_req/sample_ack)

46.6.4.2 Cortex-A55 Cross Trigger Connection

The Cortex-A55 has one CTI (Cortex-A55 Core0 CTI). The connection specifications for each are shown in **Table 46.15** and **Table 46.16**.

Table 46.15 Cortex-A55 CTI0 trigger input

Trigger Input Bit	Source Components	Description
[7]	Cortex-A55: Core0 ETM	ETM Trace Output 3 trigger event
[6]	Cortex-A55: Core0 ETM	ETM Trace Output 2 trigger event
[5]	Cortex-A55: Core0 ETM	ETM Trace Output 1 trigger event
[4]	Cortex-A55: Core0 ETM	ETM Trace Output 0 trigger event
[3]	(Unused)	—
[2]	Cortex-A55: Core0 PE	Profiling sample trigger event
[1]	Cortex-A55: Core0 PE	Performance Monitors Overflow trigger event
[0]	Cortex-A55: Core0 PE	Cross-halt trigger event

Table 46.16 Cortex-A55 CTI0 Trigger Output

Trigger Output Bit	Destination Components	Description
[7]	Cortex-A55: Core0 ETM	Generic Trace External Input 3 trigger event
[6]	Cortex-A55: Core0 ETM	Generic Trace External Input 2 trigger event
[5]	Cortex-A55: Core0 ETM	Generic Trace External Input 1 trigger event
[4]	Cortex-A55: Core0 ETM	Generic Trace External Input 0 trigger event
[3]	(Unused)	—
[2]	External (GIC-600)	Generic CTI Interrupt trigger event
[1]	Cortex-A55: Core0 PE	Restart Request trigger event
[0]	Cortex-A55: Core0 PE	Debug Request trigger event

46.6.4.3 Cortex-M33 Cross Trigger Connection

The Cortex-M33 and Cortex-M33_FPU have one CTI. The connection specifications for each are shown in **Table 46.17** and **Table 46.18**.

Table 46.17 Cortex-M33 / Cortex-M33_FPU CTI0 trigger input

Trigger Input Bit	Source Components	Description
[7]	(Unused)	—
[6]	(Unused)	—
[5]	Cortex-M33:ETM	ETM Event Output 1
[4]	Cortex-M33:ETM/Processor	ETM Event Output 0 or Comparator Output 3
[3]	Cortex-M33:Processor	DWT Comparator Output 2
[2]	Cortex-M33:Processor	DWT Comparator Output 1
[1]	Cortex-M33:Processor	DWT Comparator Output 0
[0]	Cortex-M33:Processor	Processor Halted

Table 46.18 Cortex-M33 / Cortex-M33_FPU CTI0 Trigger Output

Trigger Output Bit	Destination Components	Description
[7]	Cortex-M33:ETM	ETM Event Input 3
[6]	Cortex-M33:ETM	ETM Event Input 2
[5]	Cortex-M33:ETM	ETM Event Input 1
[4]	Cortex-M33:ETM	ETM Event Input 0
[3]	Cortex-M33:system	Interrupt request 1
[2]	Cortex-M33:system	Interrupt request 0
[1]	Cortex-M33:Processor	Processor Restart
[0]	Cortex-M33:Processor	Processor debug request

46.6.5 WDT Counter Stop Control

This LSI has a WDT counter stop control function linked with a debug event to prevent an unintended reset by WDT during debug operation. When a debug event occurs, CoreSight asserts the WDT counter stop request signal, controls the WDT counter stop signal (CNTSTOP), and stops counting. The target WDT is all WDT. For details on the WDT count stop function, refer to **Section 6.4.2, WDT Stop Control Function**.

46.6.6 GTM Counter Stop Control

This LSI has a GTM counter stop control function linked with debug events to prevent unintended GTM counts during debug operation. When a debug event occurs, CoreSight asserts a GTM counter stop request signal to stop the GTM count.

46.6.7 SYC Counter Stop Control

This LSI has a SYC counter stop control function linked with debug events to prevent unintended SYC counts during debug operation. When a debug event occurs, CoreSight asserts a SYC counter stop request signal to stop the SYC count.

47. Electrical Characteristics

47.1 Absolute Maximum Ratings

Table 47.1 Absolute Maximum Ratings

Item		Symbol	Value	Unit
Power Supply voltage (0.94 V)		V_{DD}	−0.3 to +1.26	V
		PLL16_AV_{DD}		
		PLL23_AV_{DD}		
		PLL4_AV_{DD}		
		V_{DD_ISO}		
		PCIE_V_{DD09}		
Power Supply voltage (1.8 V)		VBATT_V_{DD}	−0.3 to +2.5	V
		ADC_AV_{DD18}		
		OTP_AV_{DD18}		
		USB_AV_{DD18}		
		USB_V_{DD18}		
		PCIE_V_{DD18}		
		DDR_V_{AA}		
Power Supply voltage (1.8 V)		PV_{DD18}	−0.3 to +2.45	V
		JTAG_PV_{DD}		
Power Supply voltage (3.3 V)		PV_{DD33}	−0.3 to +4.1	V
		USB_V_{DD33}		
Power Supply Voltage (1.8 V/3.3 V switchable)		XSPI_PV_{DD}	−0.3 to +2.45 (1.8 V mode)	V
		SDn_PV_{DD} (n = 0, 1)	−0.3 to +4.1 (3.3 V mode)	V
Power Supply Voltage (1.2 V/1.8 V switchable)		I3C_PV_{DD}	−0.3 to +2.45 (1.8 V mode)	V
			−0.3 to +1.8 (1.2 V mode)	V
Power Supply Voltage (1.8 V/2.5 V/3.3 V switchable)		$PV_{DD182533\_n}$ (n = 0, 1)	−0.3 to +2.45 (1.8 V mode)	V
			−0.3 to +3.2 (2.5 V mode)	V
			−0.3 to +4.1 (3.3 V mode)	V
Power Supply Voltage (DDR)		DDR_V_{DDQ}	−0.3 to +1.68	V
Input voltage	3.3 V I/O input signal	—	−0.5 to 3.3 V power supply + 0.5	V
	2.5 V I/O input signal	—	−0.5 to 2.5 V power supply + 0.5	V
	1.8 V I/O input signal	—	−0.5 to 1.8 V power supply + 0.5	V
Operating temperature	Ambient temperature	T_a	−40 to +85*1	°C
	Junction temperature	T_j	−40 to +125	°C
Storage temperature	Ambient temperature	T_{stg}	−40 to +150	°C

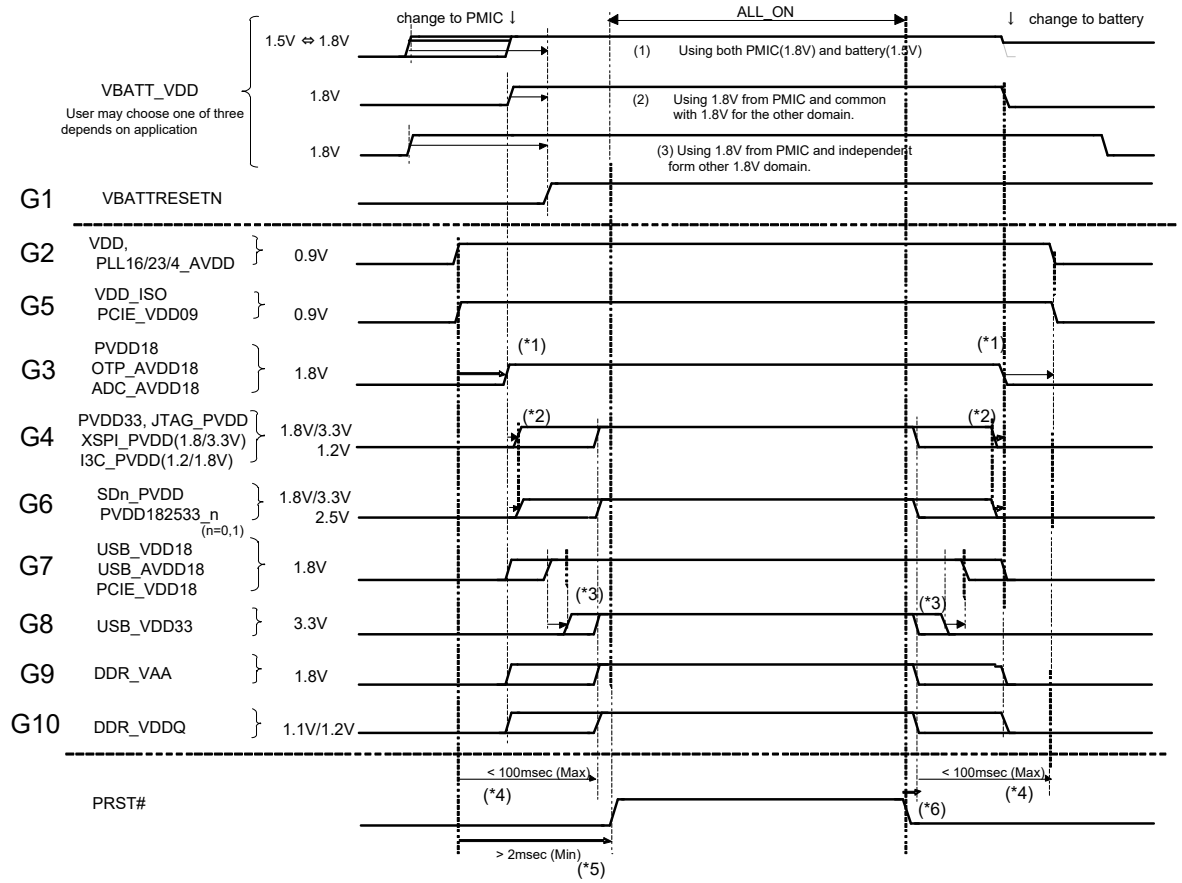
Note 1. If wider temp is required than this range, use case has to be investigated.

47.2 Power Supply

Table 47.2 Power Supply

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
Power Supply Voltage (VBATT)	VBATT_V_{DD}	1.50	1.8	1.95	V	
Power Supply Voltage (Core)	V_{DD}	0.905	0.94	0.99	V	
Power Supply Voltage (PLL)	PLL16_AV_{DD}	0.905	0.94	0.99	V	
	PLL23_AV_{DD}	0.905	0.94	0.99	V	
	PLL4_AV_{DD}	0.905	0.94	0.99	V	
Power Supply Voltage (I/O)	PV_{DD33}	3.0	3.3	3.6	V	
	PV_{DD18}	1.65	1.8	1.95	V	
Power Supply Voltage (ADC/TSU)	ADC_AV_{DD18}	1.65	1.8	1.95	V	
Power Supply Voltage (OTP)	OTP_AV_{DD18}	1.65	1.8	1.95	V	
Power Supply Voltage (JTAG)	JTAG_PV_{DD}	1.65	1.8	1.95	V	
Power Supply Voltage (XSPI)	XSPI_PV_{DD}	1.65	1.8	1.95	V	1.8 V mode
		3.0	3.3	3.6	V	3.3 V mode
Power Supply Voltage (I3C)	I3C_PV_{DD}	1.65	1.8	1.95	V	1.8 V mode
		1.1	1.2	1.3	V	1.2 V mode
Power Supply Voltage (ISO)	V_{DD_ISO}	0.905	0.94	0.99	V	
Power Supply Voltage (SD)	SDn_PV_{DD}	1.65	1.8	1.95	V	1.8 V mode
	(n = 0, 1)	3.0	3.3	3.6	V	3.3 V mode
Power Supply Voltage (Ether)	$PV_{DD182533\_n}$	1.65	1.8	1.95	V	1.8 V mode
	(n = 0, 1)	2.3	2.5	2.7	V	2.5 V mode
		3.0	3.3	3.6	V	3.3 V mode
Power Supply Voltage (USB)	USB_V_{DD33}	3.0	3.3	3.6	V	
	USB_AV_{DD18}	1.65	1.8	1.95	V	
	USB_V_{DD18}	1.65	1.8	1.95	V	
Power Supply Voltage (PCIe)	PCIE_V_{DD18}	1.65	1.8	1.95	V	
	PCIE_V_{DD09}	0.905	0.94	0.99	V	
Power Supply Voltage (DDR)	DDR_V_{AA}	1.65	1.8	1.95	V	
	DDR_V_{DDQ}	1.14	1.2	1.26	V	DDR4 mode
		1.06	1.1	1.17	V	LPDDR4 mode

47.3 Power-On/Power-Off Sequence



Note 1. G3 must be powered on after G2, G5 and powered off before G2, G5. G3 and G7 may be powered on / off at the same time.

Note 2. G4, G6 must be powered on after G3 and powered off before G3.

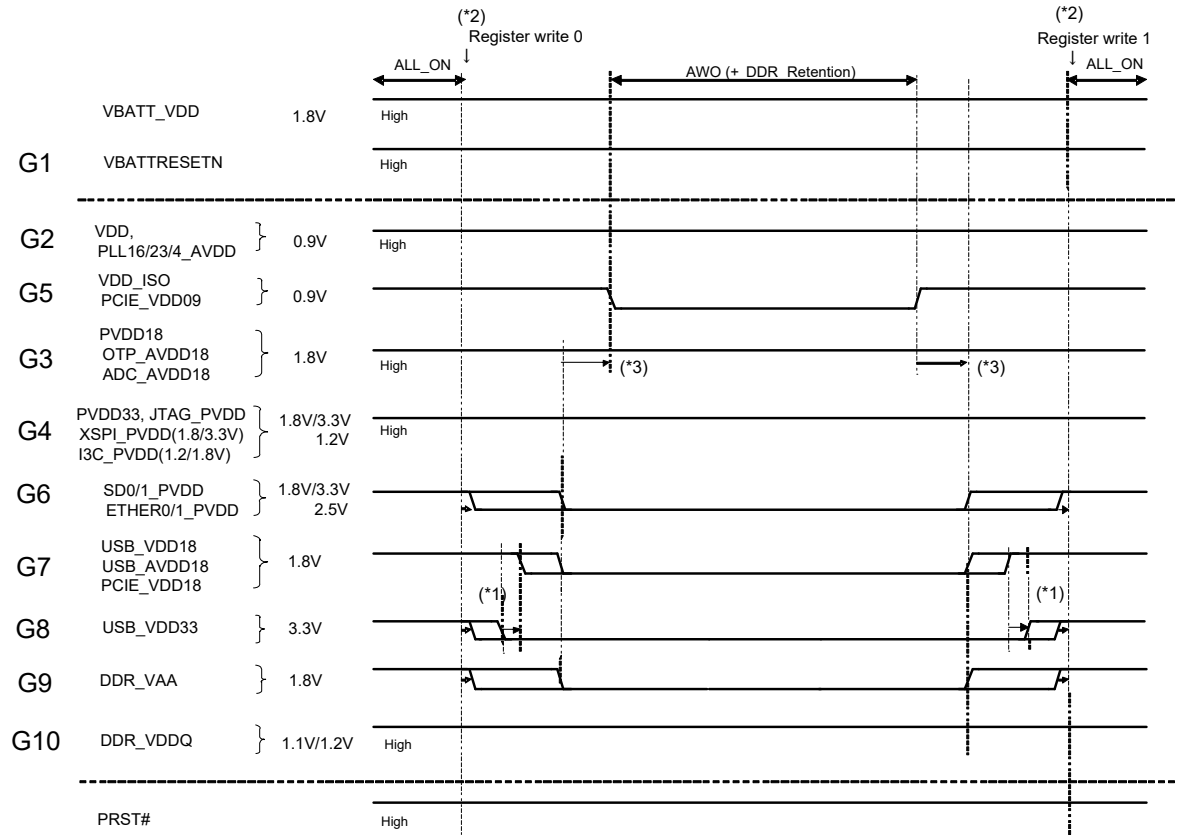
Note 3. G8 must be powered on after G7 and powered off before G7.

Note 4. It is recommended that all the power domains are powered on or off within approximately 100 msec.

Note 5. PRST# must be High after 2 msec from VDD (G2) power on.

Note 6. PRST# must be changed to Low before powered off.

Figure 47.1 Power-On/Power-Off Sequence 1 (All_OFF to ALL_ON, ALL_ON to ALL_OFF)

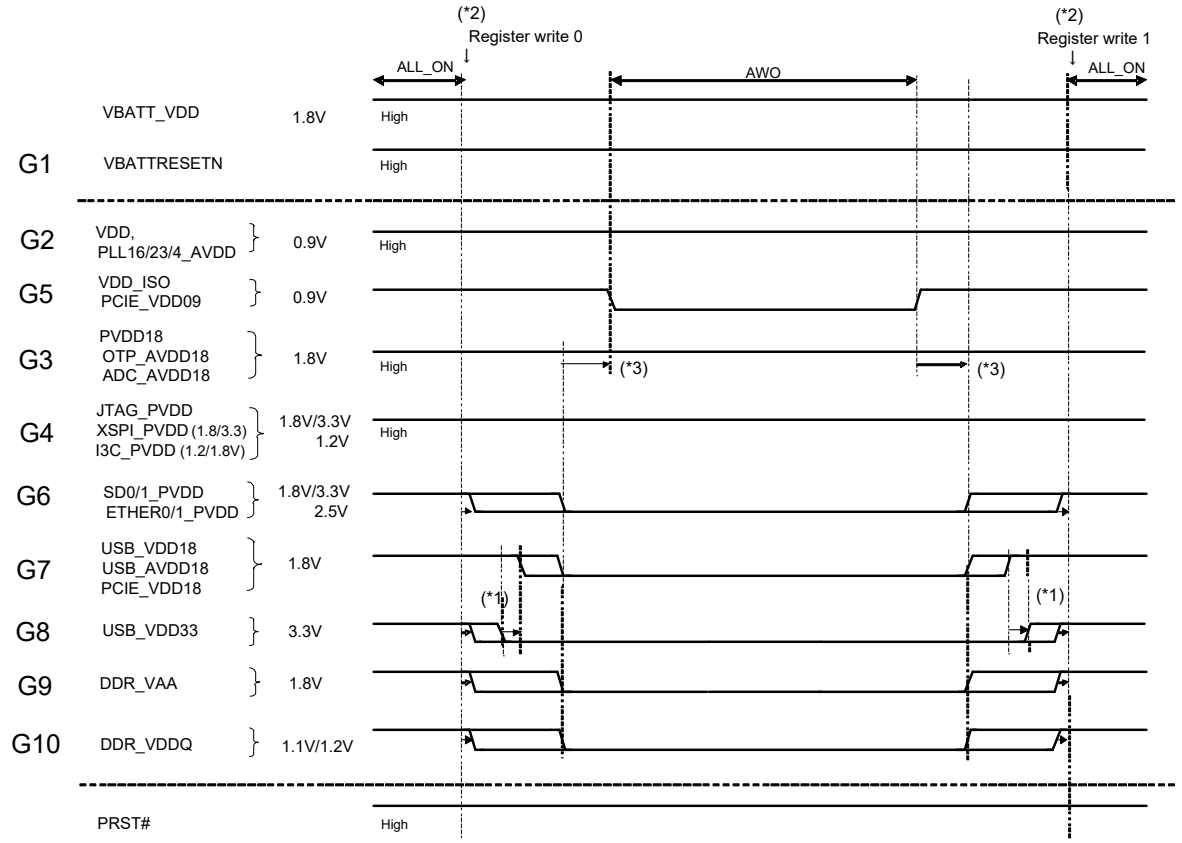


Note 1. G8 must be powered off before G7 and powered on after G7.

Note 2. The following registers must be written before G6, G7, G8, G9 are powered off or after G6, G7, G8, G9 are powered on.
- ISO_IOBUF_SE18_CTRL
- SYS_USB_PWRRDY
- CPG_RST_PCI (in CPG) or SYS_PCIE_RST_RSM_B (in SYSC)

Note 3. G5 must be powered off after G6 to G9 are powered off and powered on before G6 to G9 are powered on.

Figure 47.2 Power-On/Power-Off Sequence 2 (ALL_ON to AWO (+DDR Retention), AWO (+DDR Retention) to ALL_ON)

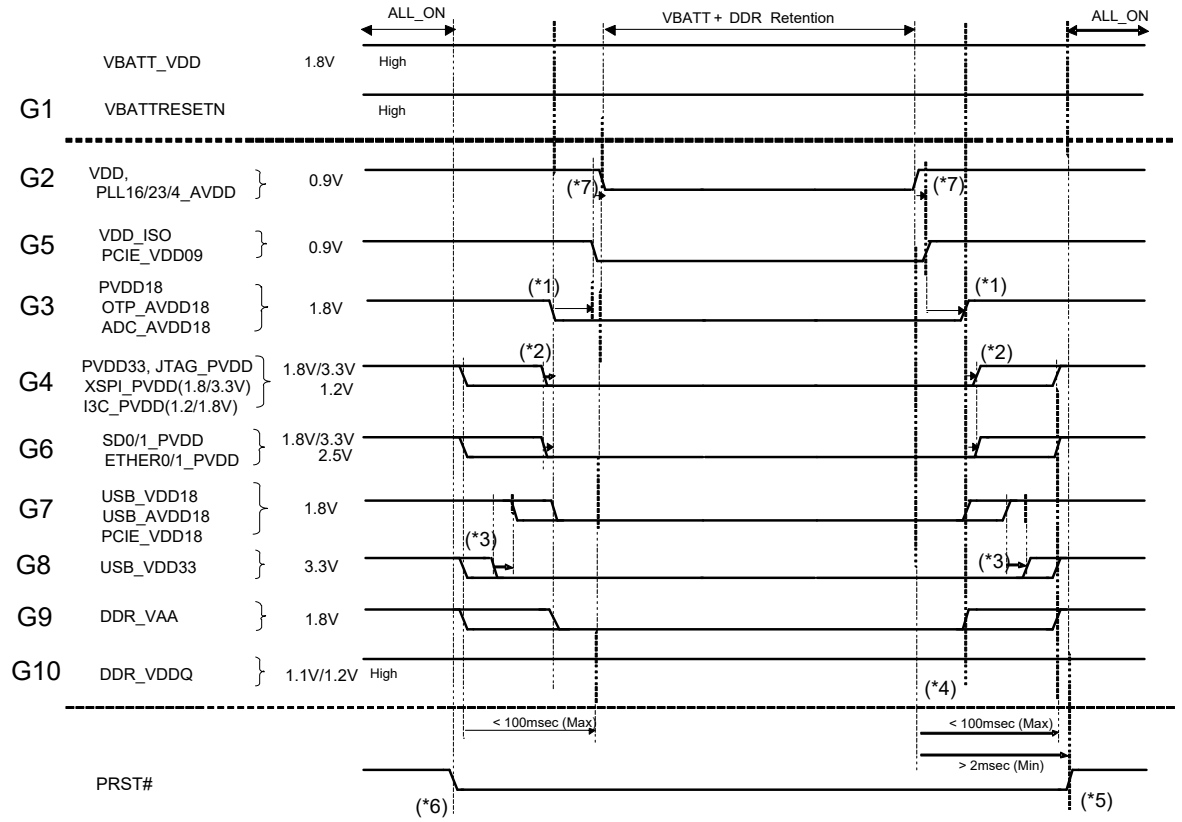


Note 1. G8 must be powered off before G7 and powered on after G7.

Note 2. The following registers must be written before G6, G7, G8, G9, G10 are powered off or after G6, G7, G8, G9, G10 are powered on.
- ISO_IOBUF_SE18_CTRL
- SYS_USB_PWRRDY
- CPG_RST_PCI (in CPG) or SYS_PCIE_RST_RSM_B (in SYSC)

Note 3. G5 must be powered off after G6 to G10 are powered off and powered on before G6 to G10 are powered on.

Figure 47.3 Power-On/Power-Off Sequence 3 (ALL_ON to AWO, AWO to ALL_ON)



Note 1. G3 must be powered off before G2, G5 and powered on after G2, G5. G3 and G7 can be powered on/off at the same time.

Note 2. G4, G6 must be powered off before G3 and powered on after G3.

Note 3. G8 must be powered off before G7 and powered on after G7.

Note 4. It is recommended that all the power domains are powered on or off within approximately 100 msec.

Note 5. PRST# must be High after 2 msec from VDD (G2) power on.

Note 6. PRST# must be changed to Low before powered off.

Note 7. G5 must be powered off before G2 and powered on after G2.

Figure 47.4 Power-On/Power-Off Sequence 3 (ALL_ON to VBATT + DDR Retention)

47.4 DC Characteristics

Table 47.3 DC Characteristics (1) [GP I/O (3.3 V)]

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
High-level input voltage		V_{IH}	2	—	—	V	
Low-level input voltage		V_{IL}	—	—	0.8	V	
Hysteresis threshold ↑		V_T+	2.1	—	—	V	
Hysteresis threshold ↓		V_T-	—	—	0.7	V	
Output logic high voltage	(I_{OH} = –1.9 mA)	V_{OH}	V_{CCQ} – 0.4	—	—	V	
	(I_{OH} = –4 mA)						
	(I_{OH} = –8 mA)						
	(I_{OH} = –9 mA)						
Output logic low voltage	(I_{OL} = 1.9 mA)	V_{OL}	—	—	0.4	V	
	(I_{OL} = 4 mA)						
	(I_{OL} = 8 mA)						
	(I_{OL} = 9 mA)						
Weak pull-up resistor (input mode)		R_{UP}	10 K	50 K	90 K	Ω	
Weak pull-down resistor (input mode)		R_{DN}	10 K	50 K	90 K	Ω	

Table 47.4 DC Characteristics (2) [SD I/O (1.8/3.3 V)]

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
High-level input voltage		V_{IH}	V_{DDQ} × 0.625	—	—	V	3.3 V mode
Low-level input voltage		V_{IL}	—	—	V_{DDQ} × 0.25	V	
Output logic high voltage	(I_{OH} = –4 mA)	V_{OH}	V_{CCQ} – 0.4	—	—	V	
	(I_{OH} = –6 mA)						
	(I_{OH} = –8 mA)						
	(I_{OH} = –9 mA)						
Output logic low voltage	(I_{OL} = 4 mA)	V_{OL}	—	—	0.4	V	
	(I_{OL} = 6 mA)						
	(I_{OL} = 8 mA)						
	(I_{OL} = 9 mA)						
High-level input voltage		V_{IH}	1.27	—	—	V	1.8 V mode
Low-level input voltage		V_{IL}	—	—	0.58	V	
Output logic high voltage	(I_{OH} = –7 mA)	V_{OH}	V_{CCQ} – 0.45	—	—	V	
	(I_{OH} = –8 mA)						
	(I_{OH} = –9 mA)						
	(I_{OH} = –10 mA)						
Output logic low voltage	(I_{OL} = 7 mA)	V_{OL}	—	—	0.45	V	
	(I_{OL} = 8 mA)						
	(I_{OL} = 9 mA)						
	(I_{OL} = 10 mA)						
Weak pull-up resistor (input mode)		R_{UP}	10 K	50 K	90 K	Ω	3.3 V/1.8 V mode
Weak pull-down resistor (input mode)		R_{DN}	10 K	50 K	90 K	Ω	

Table 47.5 DC Characteristics (3) [RGMII (1.8/2.5/3.3 V)]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
High-level input voltage	V_{IH}	V_{CCQ} × 0.65	—	—	V	3.3 V mode
Low-level input voltage	V_{IL}	—	—	V_{CCQ} × 0.35	V	
Output logic high voltage	V_{OH}	V_{CCQ} – 0.4	—	—	V	
Output logic low voltage	V_{OL}	—	—	0.4	V	
High-level input voltage	V_{IH}	1.7	—	—	V	2.5 V mode
Low-level input voltage	V_{IL}	—	—	0.7	V	
Output logic high voltage	V_{OH}	V_{CCQ} – 0.4	—	—	V	
Output logic low voltage	V_{OL}	—	—	0.4	V	
High-level input voltage	V_{IH}	V_{CCQ} × 0.65	—	—	V	1.8 V mode
Low-level input voltage	V_{IL}	—	—	V_{CCQ} × 0.35	V	
Output logic high voltage	V_{OH}	V_{CCQ} – 0.45	—	—	V	
Output logic low voltage	V_{OL}	—	—	0.45	V	

Table 47.6 DC Characteristics (4) [I3C (1.2/1.8 V)]

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
High-level input voltage		V_{IH}	V_{CCQ} × 0.7	—	—	V	1.8 V
Low-level input voltage		V_{IL}	—	—	V_{CCQ} × 0.3	V	I3C Open Drain mode
Output logic low voltage	IOL = 3 mA	V_{OL}	—	—	0.27	V	
High-level input voltage		V_{IH}	V_{CCQ} × 0.7	—	—	V	1.8 V
Low-level input voltage		V_{IL}	—	—	V_{CCQ} × 0.3	V	I2C Open Drain mode*1
Output logic low voltage	IOL = 20 mA	V_{OL}	—	—	0.4	V	
High-level input voltage		V_{IH}	V_{CCQ} × 0.7	—	—	V	1.8 V
Low-level input voltage		V_{IL}	—	—	V_{CCQ} × 0.3	V	I3C CMOS mode
Output logic high voltage	IOL= −3 mA	V_{OH}	V_{CCQ} − 0.27	—		V	
Output logic low voltage	IOL= 3 mA	V_{OL}	—	—	0.27	V	
High-level input voltage		V_{IH}	V_{CCQ} × 0.7	—	—	V	1.2 V
Low-level input voltage		V_{IL}	—	—	V_{CCQ} × 0.3	V	I3C CMOS mode
Output logic high voltage	IOL = −2 mA	V_{OH}	V_{CCQ} − 0.18	—		V	
Output logic low voltage	IOL = 2 mA	V_{OL}	—	—	0.18	V	

Note 1. fast-mode, fast-mode plus

Table 47.7 DC Characteristics (5) [I2C (1.8 V)]

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
High-level input voltage		V_{IH}	V_{CCQ} × 0.7	—	—	V	
Low-level input voltage		V_{IL}	—	—	V_{CCQ} × 0.3	V	
Output logic low voltage	IOL = 20 mA	V_{OL}	—	—	0.4	V	

Table 47.8 DC Characteristics (6) [Input (Schmitt/1.8 V)]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
High-level input voltage	V_{IH}	V_{CCQ} × 0.75	—	—	V	
Low-level input voltage	V_{IL}	—	—	V_{CCQ} × 0.25	V	

Table 47.9 DC Characteristics (7) [Input (3.3 V tolerant/Schmitt)]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
High-level input voltage	V_{IH}	V_{CCQ} × 0.75	—	—	V	
Low-level input voltage	V_{IL}	—	—	V_{CCQ} × 0.25	V	

Table 47.10 DC Characteristics (8) [USB 2.0]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
Input levels for low/full speed						
High (driven)	V_{IH}	2.0	—	—	V	
Low	V_{IL}	—	—	0.8	V	
Differential input sensitivity	V_{DI}	0.2	—	—	V	
Differential common mode range	V_{CM}	0.8	—	2.5	V	
Input levels for high speed						
High-speed squelch detection threshold (differential signal amplitude)	V_{HSSQ}	100	—	150	mV	
High-speed data signaling common mode voltage range (guideline for receiver)	V_{HSCM}	−50	—	500	mV	
Output levels for low/full speed						
Low	V_{OL}	0.0	—	0.3	V	
High (driven)	V_{OH}	2.8	—	3.6	V	
Output signal crossover voltage	V_{CRS}	1.3	—	2.0	V	
Output levels for high-speed						
High-speed idle level	V_{HSOI}	−10.0	—	10.0	mV	
High-speed data signaling high	V_{HSOH}	360	—	440	mV	
High-speed data signaling low	V_{HSOL}	−10.0	—	10.0	mV	
Chirp J level (differential voltage)	V_{CHIRPJ}	700	—	1100	mV	
Chirp K level (differential voltage)	V_{CHIRPK}	−900	—	−500	mV	

Table 47.11 DC Characteristics (9) [ADC]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
Resolution	—	—	12	—	Bit	
Analog input channel	—	—	—	8	Channel	
Analog input range	AIN	V_{SS}	—	ADC_AV_{DD18}	V	
Differential non-linearity	DNL	—	—	±3.0	LSB	
Integral non-linearity	INL	—	—	±6.0	LSB	

Table 47.12 DC Characteristics (10) [Current Consumption]

Item	Power Rail Symbol	Max Current	Unit	Remarks
Power Supply Voltage (VBATT)	VBATT_V_{DD}	10	mA	
Power Supply Voltage (Core)	V_{DD}	450	mA	
Power Supply Voltage (ISO)	V_{DD_ISO}	2350	mA	Condition: Cortex-A55, Dhrystone
Power Supply Voltage (PLL)	PLL16_AV_{DD}	10	mA	
	PLL23_AV_{DD}	10	mA	
	PLL4_AV_{DD}	10	mA	
Power Supply Voltage (I/O)	PV_{DD33}	1100	mA	
	PV_{DD18}	20	mA	
Power Supply Voltage (ADC/TSU)	ADC_AV_{DD18}	10	mA	
Power Supply Voltage (OTP)	OTP_AV_{DD18}	10	mA	
Power Supply Voltage (JTAG)	JTAG_PV_{DD}	10	mA	
Power Supply Voltage (XSPI) (1.8 V)	XSPI_PV_{DD}	400	mA	
Power Supply Voltage (XSPI) (3.3 V)	XSPI_PV_{DD}	600	mA	
Power Supply Voltage (I3C) (1.8 V)	I3C_PV_{DD}	10	mA	
Power Supply Voltage (I3C) (1.2 V)	I3C_PV_{DD}	10	mA	
Power Supply Voltage (SD) (1.8 V)	SD0_PV_{DD}	400	mA	
	SD1_PV_{DD}	300	mA	
Power Supply Voltage (SD) (3.3 V)	SD0_PV_{DD}	700	mA	
	SD1_PV_{DD}	500	mA	
Power Supply Voltage (Ether) (1.8 V)	$PV_{DD182533\_0}$	110	mA	
	$PV_{DD182533\_1}$	110	mA	
Power Supply Voltage (Ether) (2.5 V)	$PV_{DD182533\_0}$	200	mA	
	$PV_{DD182533\_1}$	200	mA	
Power Supply Voltage (Ether) (3.3 V)	$PV_{DD182533\_0}$	200	mA	
	$PV_{DD182533\_1}$	200	mA	
Power Supply Voltage (USB)	USB_V_{DD33}	100	mA	
	USB_AV_{DD18}	200	mA	
	USB_V_{DD18}	200	mA	
Power Supply Voltage (PCIe)	PCIE_V_{DD18}	100	mA	
	PCIE_V_{DD09}	100	mA	
Power Supply Voltage (DDR)	DDR_V_{AA}	10	mA	
Power Supply Voltage (DDR) (DDR4)	DDR_V_{DDQ}	300	mA	
Power Supply Voltage (DDR) (LPDDR4)	DDR_V_{DDQ}	400	mA	

Note: T_j = 125°C, Power supply voltage = Max.

47.5 AC Characteristics

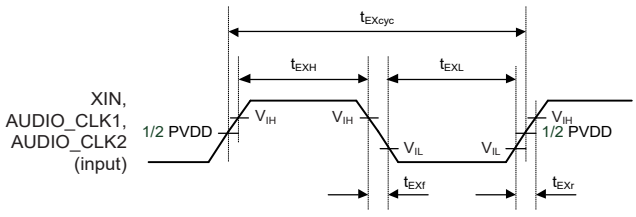
Conditions: VBATT_V_{DD} = 1.50 to 1.95 V, V_{DD} = PLL16/23/4_AV_{DD} = 0.905 to 0.99 V,
PV_{DD33} = 3.0 to 3.6 V, PV_{DD18} = ADC_AV_{DD18} = OTP_AV_{DD18} = 1.65 to 1.95 V,
JTAG_PV_{DD} = 1.65 to 1.95 V, XSPI_PV_{DD} = 1.65 to 1.95 V / 3.0 to 3.6 V,
I3C_PV_{DD} = 1.1 to 1.3 V / 1.65 to 1.95 V, V_{DD_ISO} = PCIE_V_{DD09} = 0.905 to 0.99 V,
SDn_PV_{DD} (n = 0, 1) = 1.65 to 1.95 V / 3.0 to 3.6 V,
$PV_{DD182533\_n}$ (n = 0, 1) = 1.65 to 1.95 V / 2.3 to 2.7 V / 3.0 to 3.6 V,
USB_V_{DD33} = 3.0 to 3.6 V, USB_AV_{DD18} = USB_V_{DD18} = PCIE_V_{DD18} = 1.65 to 1.95 V,
DDR_V_{AA} = 1.65 to 1.95 V, DDR_V_{DDQ} = 1.06 to 1.17 V (LPDDR4) / 1.14 to 1.26 V (DDR4),
USB_V_{SS} = 0 V, V_{SS} = 0 V, T_a = −40 to +85°C, T_j = −40 to +125°C

47.5.1 Clock Timing

Table 47.13 Clock Timing Table

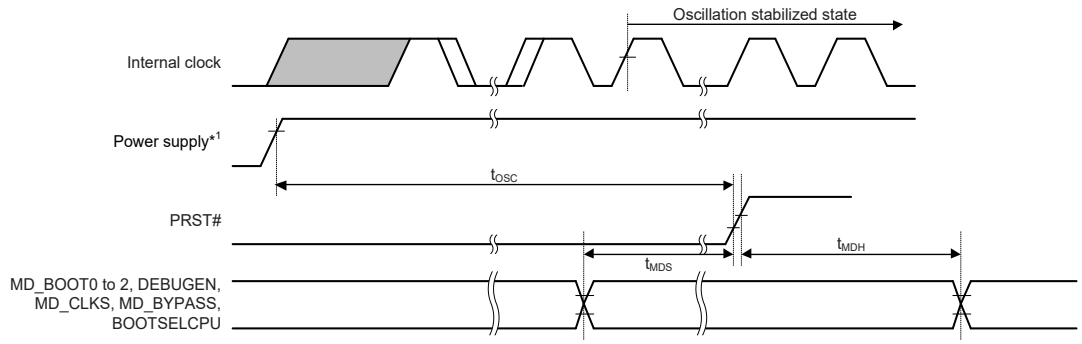
Item	Symbol	Min.	Max.	Unit	Figures
XIN clock input frequency	f_{EX}	24 – 50ppm*1	24 + 50ppm*1	MHz	**Figure 47.5**
XIN clock input cycle time	t_{EXcyc}	41.67	41.67	ns	
AUDIO_CLK1, AUDIO_CLK2 clock input frequency (external clock is input)	f_{EX}	10	50	MHz	
AUDIO_CLK1, AUDIO_CLK2 clock input cycle time (external clock is input)	t_{EXcyc}	20	100	ns	
XIN, AUDIO_CLK1, AUDIO_CLK2 clock input low level pulse width	t_{EXL}	0.4	0.6	t_{EXcyc}	
XIN, AUDIO_CLK1, AUDIO_CLK2 clock input high level pulse width	t_{EXH}	0.4	0.6	t_{EXcyc}	
XIN, AUDIO_CLK1, AUDIO_CLK2 clock input rise time	t_{EXr}	—	4	ns	
XIN, AUDIO_CLK1, AUDIO_CLK2 clock input fall time	t_{EXf}	—	4	ns	
Oscillator stabilization time	t_{OSC}	—	2	ms	**Figure 47.6**, **Figure 47.7**
Mode hold time	t_{MDH}	—	12	µs	
Mode setup time	t_{MDS}	—	0	µs	

Note 1. When using RGMII interface. If not using RGMII mode, this spec is ±100 ppm.



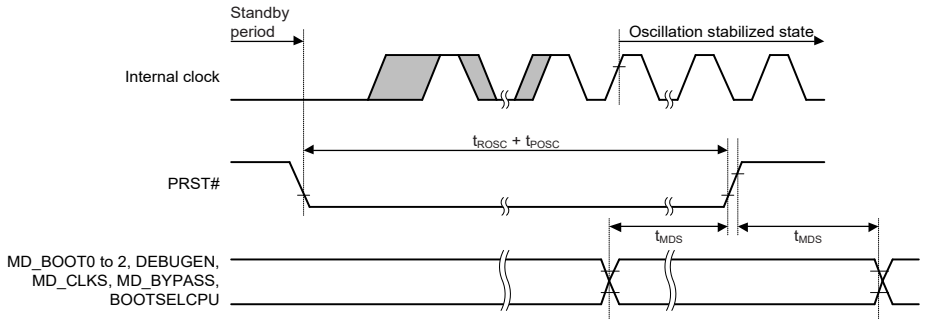
Note: When the clock is input on the EXCLK, AUDIO_CLK1 or AUDIO_CLK2

Figure 47.5 EXCLK, AUDIO_CLK1 and AUDIO_CLK2 Clock Input Timing



Note 1. V_{DD}, PLLn_DV_{DD11}, PLLn_AV_{DD18}, ADC_AV_{DD18}, V_{DD18}, OTP_V_{DD18}, USB_V_{DD18}, CSI_V_{DD18}, PV_{DD},USB_V_{DD33}, DDR_V_{DDQ}, SPI_PV_{DD}, SDn_PV_{DD}, $PV_{DD182533\_0}$, $PV_{DD182533\_1}$, OTP_V_{DD18}

Figure 47.6 Power-On Oscillation Settling Time



Note: Oscillation setting time when the internal oscillator is used.

Figure 47.7 Oscillation Settling Time on Return from Standby (Return by Reset)

Table 47.14 32-kHz Clock Oscillator Timing Table

Item	Symbol	Min.	Max.	Unit	Figures
32-kHz clock oscillator stabilization time	t_{ROSC}	—	2.5	sec	

Note: Oscillation stabilization time after enabling the oscillation of the 32-kHz OSC implemented in the VBATTB region

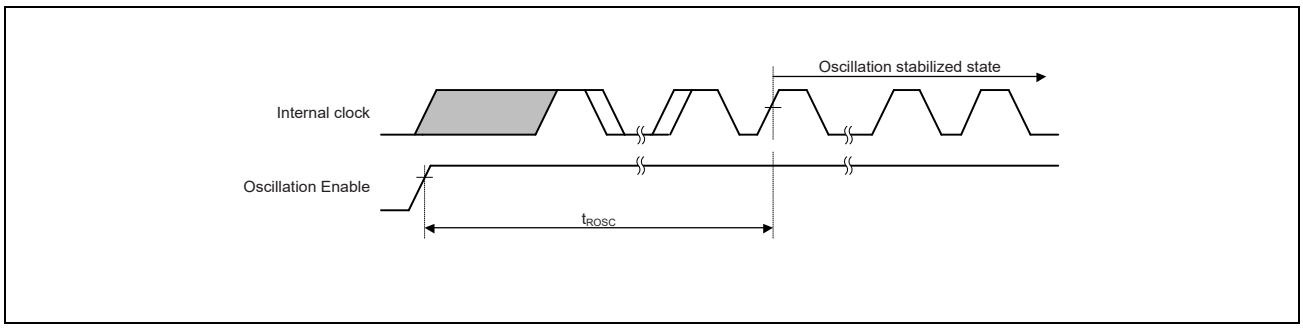


Figure 47.8 32-kHz Clock Oscillator Stabilization Time

47.5.2 Control Signal Access Timing

Table 47.15 Control Signal Timing

Item	Symbol	Min.	Max.	Unit	Figures
PRST# pulse width	t_{RESW}	20	—	t_{cyc}*1	**Figure 47.9**
TRST# pulse width	t_{TRSW}	20	—	t_{cyc}*1	
NMI pulse width	t_{NMIW}	20	—	t_{cyc}*1	**Figure 47.11**
IRQ pulse width	t_{IRQW}	20	—	t_{cyc}*1	
TINT pulse width	t_{TINTW}	20	—	t_{cyc}*1	
PRST# input rise time	t_{RSr}	—	500	µs	**Figure 47.10**

Note 1. t_{cyc} = 41.666 ns (24 MHz)

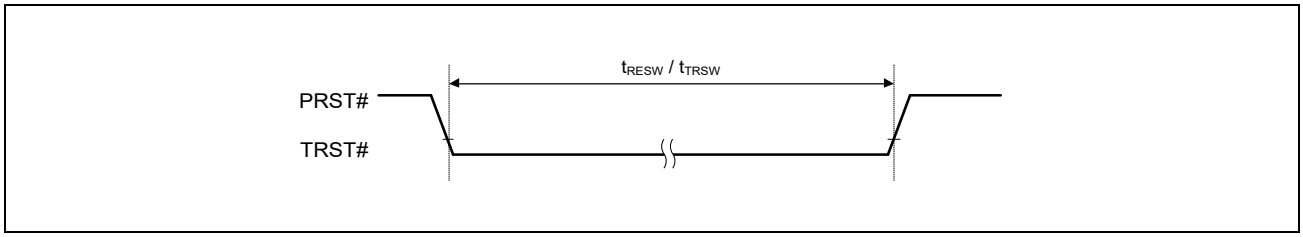


Figure 47.9 Reset Input Timing 1

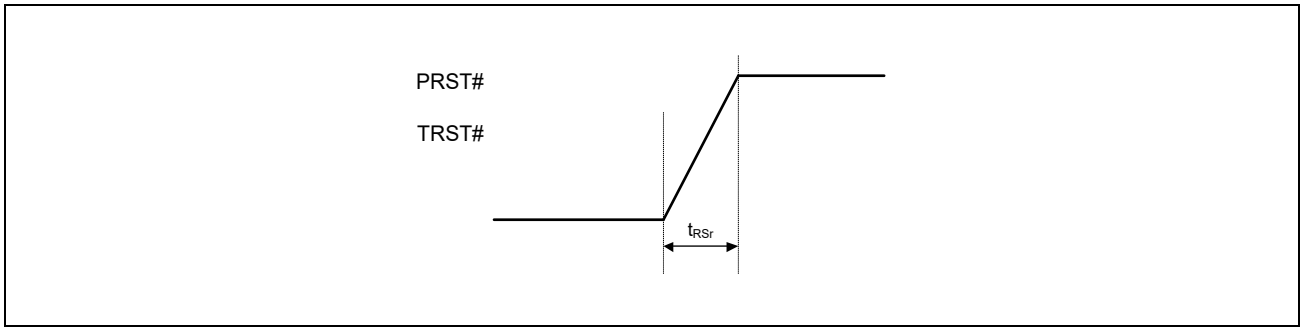
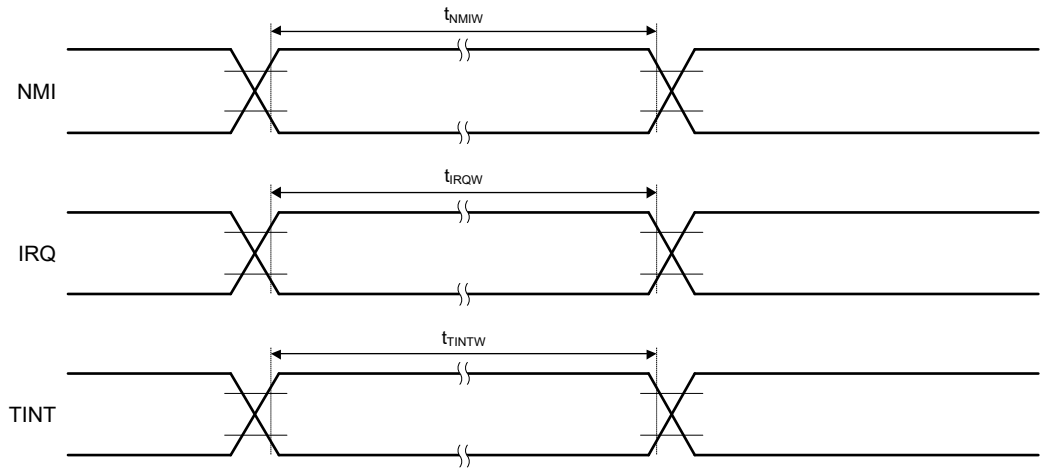


Figure 47.10 Reset Input Timing 2



Note: This specification “(Min. 20 t_{cycle})” is the min. pulse width the case that the digital noise filter is off. Refer to **Section 45, General Purpose Input Output Port (GPIO)** for Digital Noise Filter Settings in detail.

Figure 47.11 Interrupt Signal Input Timing

47.5.3 SDHI Access Timing

47.5.3.1 SDHI Access Timing (SDR 3.3-V)

Table 47.16 SDHC AC Access Timing (SDR at 3.3-V Operation)

Item	Symbol	Default Speed Mode (15.625 MHz)		High Speed Mode (31.25 MHz)		Unit	Figures
		Min.	Max.	Min.	Max.		
SD_CLK clock cycle	t_{SDCYC}	64.00	—	32.0	—	ns	**Figure 47.12**
SD_CLK clock high level width	t_{SDWH}	10	—	7	—	ns	
SD_CLK clock low level width	t_{SDWL}	10	—	7	—	ns	
SD_CLK clock rise time	t_{SDLH}	—	10	—	3	ns	
SD_CLK clock fall time	t_{SDHL}	—	10	—	3	ns	
SD_CMD,SD_DATA output delay	t_{SDODLY}	−9.0	4.0	−8.0	4.0	ns	
SD_CMD,SD_DATA input set up time	t_{SDIS}	5.5	—	5.5	—	ns	
SD_CMD,SD_DATA input hold time	t_{SDIH}	2.0	—	2.0	—	ns	
SD_CMD,SD_DATA input data width	t_{SDIDW}	—	—	—	—	ns	

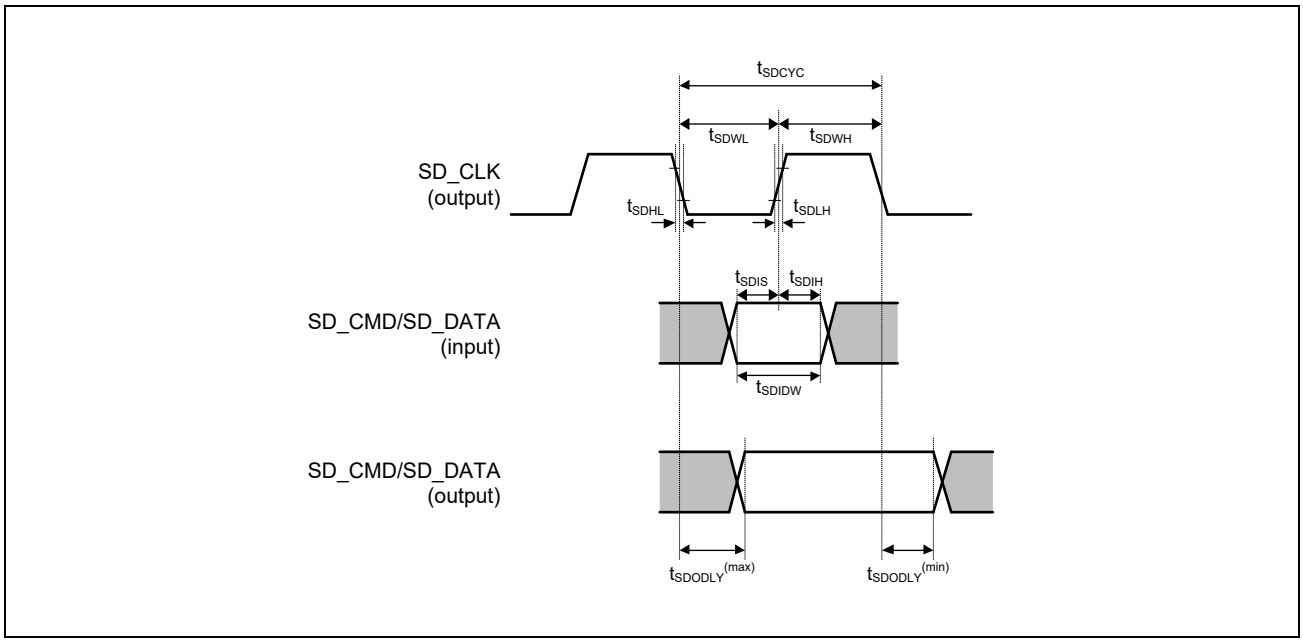


Figure 47.12 SDHC Interface Timing (SDR 3.3-V Power Supply)

NOTE

The disclosure of other characteristics of the SD interface needs the conclusion of the following agreement.

- SD Host/Ancillary Product License Agreement (SD HALA)

For details, contact your local sales representatives.

47.5.4 eMMC Access Timing

47.5.4.1 eMMC Host Interface Timing (Default)

Table 47.17 eMMC Host Interface Timing (MMC Default 3.3-V Power Supply)

Item	Symbol	Min.	Max.	Unit	Figures
SD0_CLK clock cycle	t_{MMCPP}	32.00	—	ns	**Figure 47.13**
SD0_CLK clock high level width	t_{MMCWH}	7	—	ns	
SD0_CLK clock low level width	t_{MMCWL}	7	—	ns	
SD0_CLK clock rise time	t_{MMCLH}	—	3	ns	
SD0_CLK clock fall time	t_{MMCHL}	—	3	ns	
SD0_CMD/SDDAT output delay	$t_{MMCODLY}$	−8.0	4.0	ns	
SD0_CMD/SDDAT input set up time	t_{MMCISU}	5.5	—	ns	
SD0_CMD/SDDAT input hold time	t_{MMCIH}	2.0	—	ns	
SD0_CMD/SDDAT input data width	t_{MMCIDW}	—	—	ns	

Table 47.18 eMMC Host Interface Timing (MMC Default 1.8-V Power Supply)

Item	Symbol	Min.	Max.	Unit	Figures
SD0_CLK clock cycle	t_{MMCPP}	16.00	—	ns	**Figure 47.13**
SD0_CLK clock high level width	t_{MMCWH}	4.5	—	ns	
SD0_CLK clock low level width	t_{MMCWL}	4.5	—	ns	
SD0_CLK clock rise time	t_{MMCLH}	—	2.45	ns	
SD0_CLK clock fall time	t_{MMCHL}	—	2.45	ns	
SD0_CMD/SDDAT output delay	$t_{MMCODLY}$	−7.00	4.00	ns	
SD0_CMD/SDDAT input set up time	t_{MMCISU}	5.00	—	ns	
SD0_CMD/SDDAT input hold time	t_{MMCIH}	1.40	—	ns	
SD0_CMD/SDDAT input data width	t_{MMCIDW}	—	—	ns	

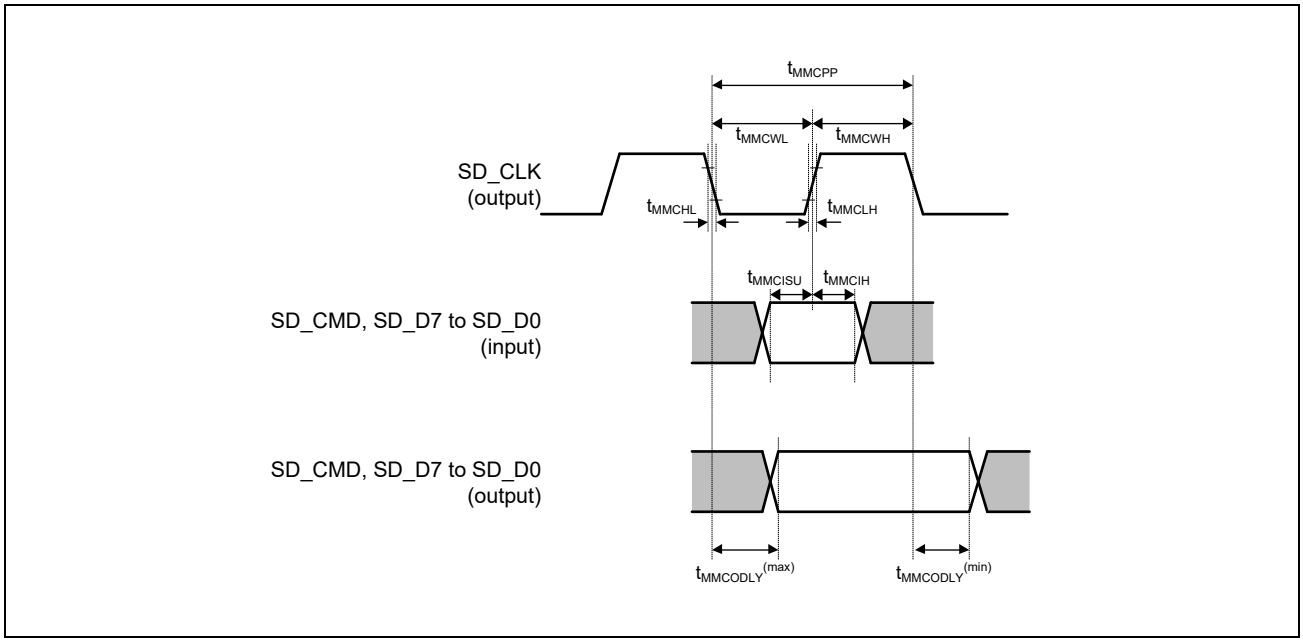


Figure 47.13 eMMC Host Interface Timing (MMC Default 1.8-V/3.3-V Power Supply)

47.5.4.2 eMMC host interface timing (HS-SDR)

NOTES

1. The spec of eMMC host interface timing (HS-SDR 3.3-V power supply) is the same as **Table 47.17** eMMC host interface timing (MMC default 3.3-V power supply).
2. The spec of eMMC host interface timing (HS-SDR 1.8V power supply) is the same as **Table 47.18** eMMC host interface timing (MMC default 1.8-V power supply)

47.5.4.3 eMMC host interface timing (HS200)

Table 47.19 eMMC Host Interface Timing (HS200 1.8-V Power Supply Operation, Output Load 15 pF)

Item	Symbol	Min.	Max.	Unit	Figures
SD0_CLK clock cycle	t_{MMCPP}	8.00	16.00	ns	**Figure 47.14**
SD0_CLK clock high level width	t_{MMCWH}	2.25	—	ns	
SD0_CLK clock low level width	t_{MMCWL}	2.25	—	ns	
SD0_CLK clock rise time	t_{MMCLH}	—	1.22	ns	
SD0_CLK clock fall time	t_{MMCHL}	—	1.22	ns	
SD0_CMD/SDDAT output delay	$t_{MMCODLY}$	−3.00	2.00	ns	
SD0_CMD/SDDAT input set up time	t_{MMCISU}	—	—	ns	
SD0_CMD/SDDAT input hold time	t_{MMCIH}	—	—	ns	
SD0_CMD/SDDAT input data width	t_{MMCIDW}	4.31	—	ns	

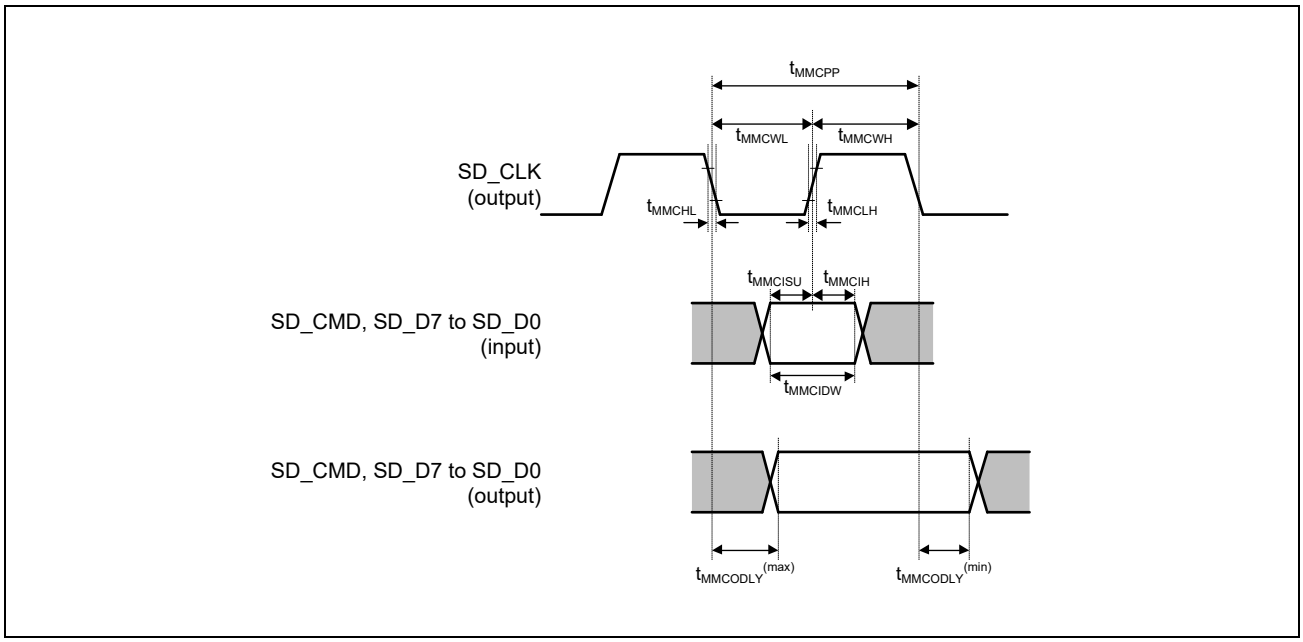


Figure 47.14 eMMC Host Interface (MMC Interface HS200 Mode 1.8-V Power Supply Selection)

47.5.5 USB 2.0 Host/Function Module Access Timing

47.5.5.1 USB 2.0 Low-Speed Access Timing

Table 47.20 USB Transceiver Timing (Low-Speed)

Item	Symbol	Min.	Max.	Unit	Figures
Rise time	t_{LR}	75	300	ns	**Figure 47.15**
Fall time	t_{LF}	75	300	ns	
Rise/fall time lag	t_{LR}/t_{LF}	80	125	%	

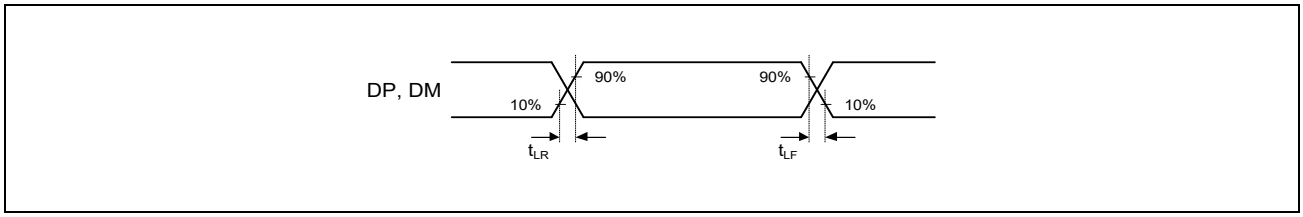


Figure 47.15 USB0_DP, USB1_DP, USB0_DM, and USB1_DM Output Timing (Low-Speed)

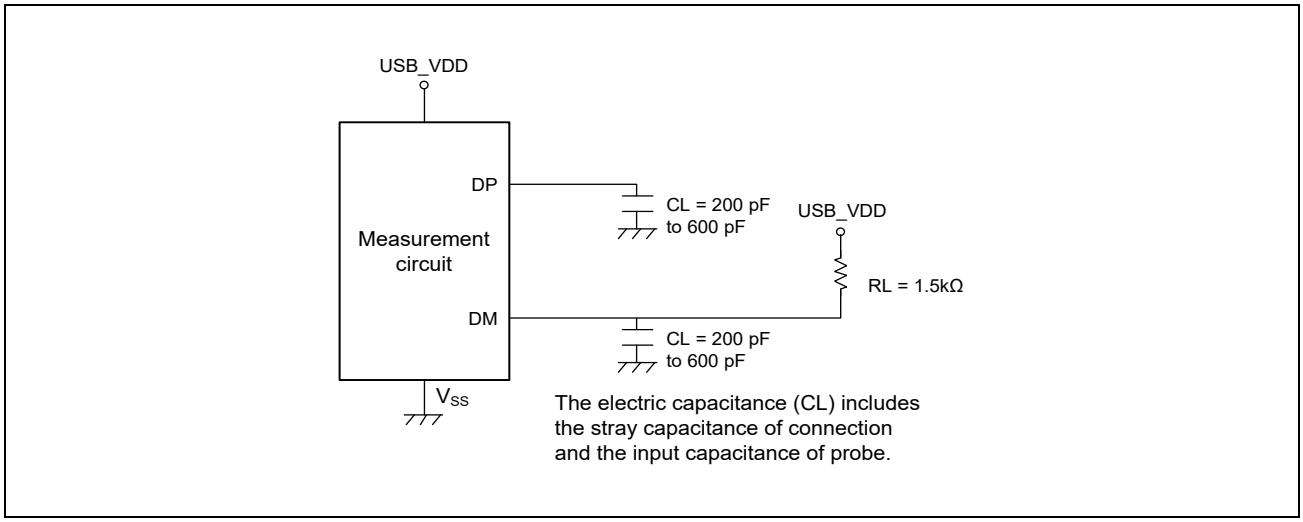


Figure 47.16 Measurement Circuit (Low-Speed)

47.5.5.2 USB 2.0Full-Speed Access Timing

Table 47.21 USB Transceiver Timing (Full-Speed)

Item	Symbol	Min.	Max.	Unit	Figures
Rise time	t_{FR}	4	20	ns	**Figure 47.17**
Fall time	t_{FF}	4	20	ns	
Rise/fall time lag	t_{FR}/t_{FF}	90	111.11	%	

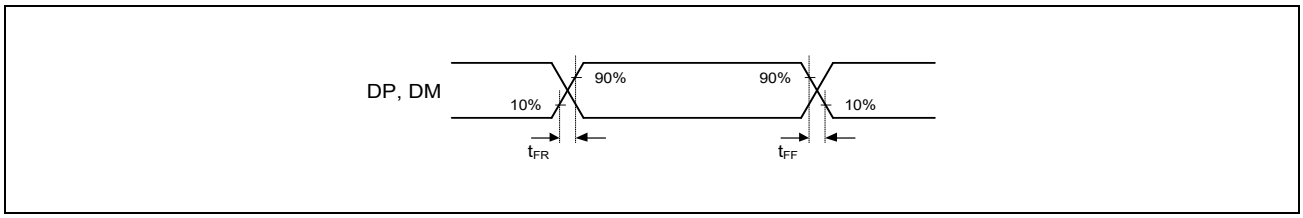


Figure 47.17 USB0_DP, USB1_DP, USB0_DM, and USB1_DM Output Timing (Full-Speed)

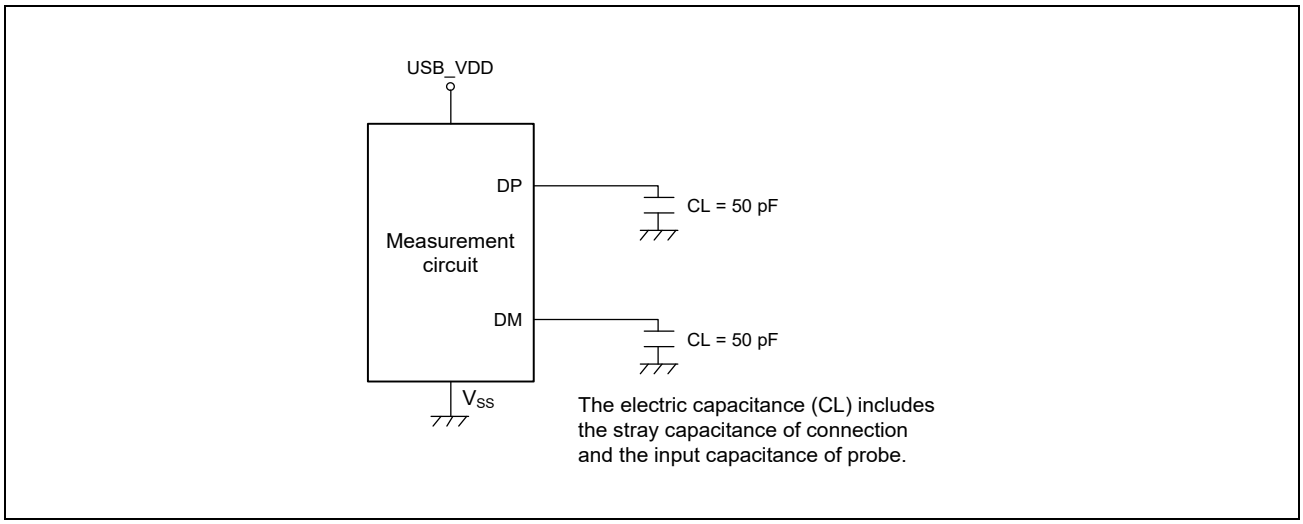


Figure 47.18 Measurement Circuit (Full-Speed)

47.5.5.3 USB 2.0 Hi-Speed Access Timing

Table 47.22 USB Transceiver Timing (Hi-Speed)

Item	Symbol	Min.	Max.	Unit	Figures
Rise edge rate	t_{HSR}	—	2133	V/µs	**Figure 47.19**
Fall edge rate	t_{HSF}	—	2133	V/µs	
Output driver resistance	Z_{HSDRV}	40.5	49.5	Ω	

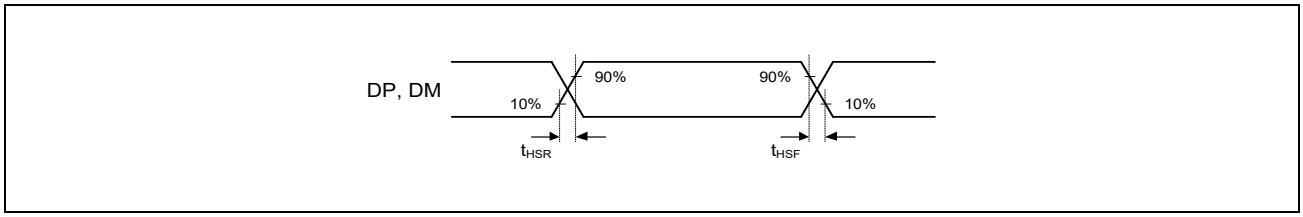


Figure 47.19 USB0_DP, USB1_DP, USB0_DM, and USB1_DM Output Timing (Hi-Speed)

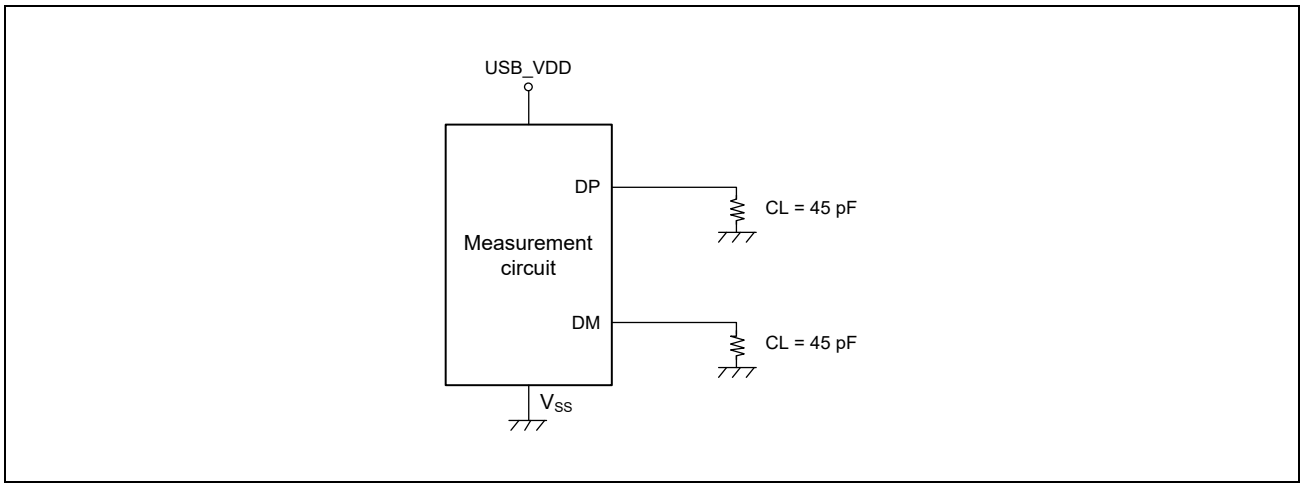
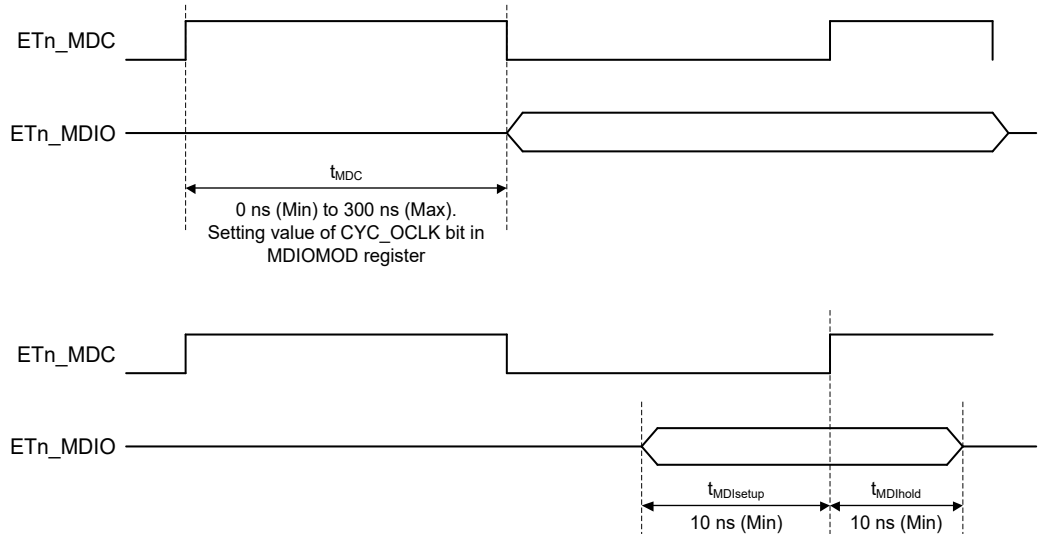


Figure 47.20 Measurement Circuit (Hi-Speed)

47.5.6 Ethernet Interface Access Timing

Table 47.23 Ethernet Interface Access Timing

Item	Symbol	Min.	Max.	Unit	Figures
MDC half cycle	t_{MDC}	0	300	ns	**Figure 47.21**
MDI setup time	$t_{MDIsetup}$	10	—	ns	
MDI hold time	$t_{MDIhold}$	10	—	ns	



Note: (n = 0, 1)

Figure 47.21 Management Interface

47.5.6.1 Ethernet-IF (Ether MII)

Table 47.24 Ethernet-IF Access Timing (Ether MII)

Item		Symbol	Min.	Max.	Unit	Figures
Ether MII	ETH_GTXTXC_TXC period	t_{Tcyc}	40	—	ns	**Figure 47.22**
	ETH_TXCTL output delay	t_{TENd}	0	25	ns	
	ETH_TXD3-0 output delay	t_{MTDd}	0	25	ns	
	ETH_RXC period	t_{Rcyc}	40	—	ns	
	ETH_RXDV setup time	t_{RDVs}	10	—	ns	
	ETH_RXDV hold time	t_{RDVh}	10	—	ns	
	ETH_RXD3-0 setup time	t_{MRDs}	10	—	ns	
	ETH_RXD3-0 hold time	t_{MRDh}	10	—	ns	
	ETH_RXER setup time	t_{RERs}	10	—	ns	
	ETH_RXER hold time	t_{RERh}	10	—	ns	

Note: I/O driving ability: 12 mA
CL = 8 pF

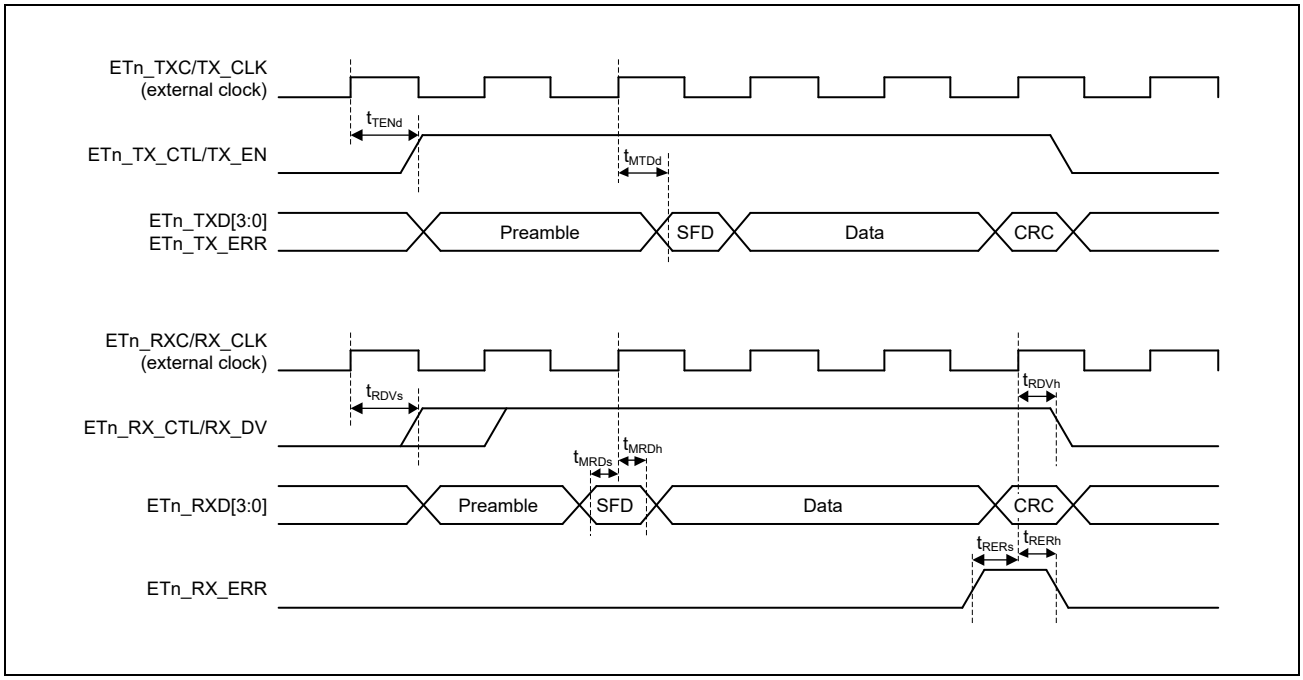


Figure 47.22 MII Transmission Timing

Validated with IEEE802.3 regulation.

The propagation delay for each twisted pair, measured from the MII connector to the PHY, shall not exceed 2.5 ns.

The variation in the propagation delay of the twisted pairs in a given cable bundle, measured from the MII connector to the PHY, shall not exceed 0.1 ns.

47.5.6.2 Ethernet-IF (Ether RGMII)

Table 47.25 Ethernet-IF Access Timing (Ether RGMII)

Item		Symbol	Min.	Typ.	Max.	Unit	capaci-tance	Remarks	Figures
Ether RGMII	Data to clock output skew @ transmitter	T_{skewT}	−500	0	500	ps	8 pF	Tx RGMII	**Figure 47.23**
	Data to clock input skew @ receiver	T_{skewR}	1	1.8	2.6	ns	8 pF		
	Data to clock output setup @ transmitter integrated delay	T_{setupT}	1.2	2.0	—	ns	8 pF	Rx RGMII-ID	**Figure 47.24**
	Clock to data output hold @ transmitter integrated delay	T_{holdT}	1.2	2.0	—	ns	8 pF		
	Data to clock input setup setup @ receiver integrated delay	T_{setupR}	1.0	2.0	—	ns	8 pF	Rx RGMII-ID	**Figure 47.24**
	Data to clock input setup hold @ receiver integrated delay	T_{holdR}	1.0	2.0	—	ns	8 pF		
	Clock cycle duration*2	T_{cyc}	7.2	8	8.8	ns	8 pF	—	—
	Duty cycle for gigabit	Duty_G	40*1	50	60*1	%	8 pF		
	Duty cycle for 10/100T	Duty_T	40*1	50	60*1	%	8 pF		
	Rise/fall time (20 - 80%)	T_r/T_f	—	—	0.75	ns	8 pF		

Note 1. Relaxed from regulation of RGMII.

Note 2. For 10 Mbps and 100 Mbps, T_{cyc} will scale to 400 ns ± 40 ns and 40 ns ± 4 ns respectively.

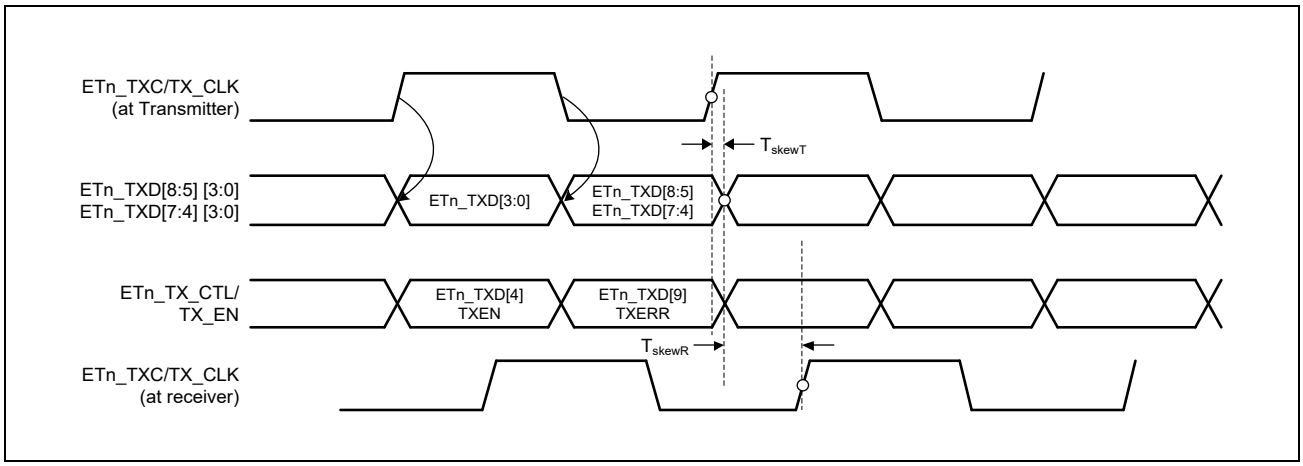


Figure 47.23 Multiplexing & Timing Diagram — RGMII (Transmitter)

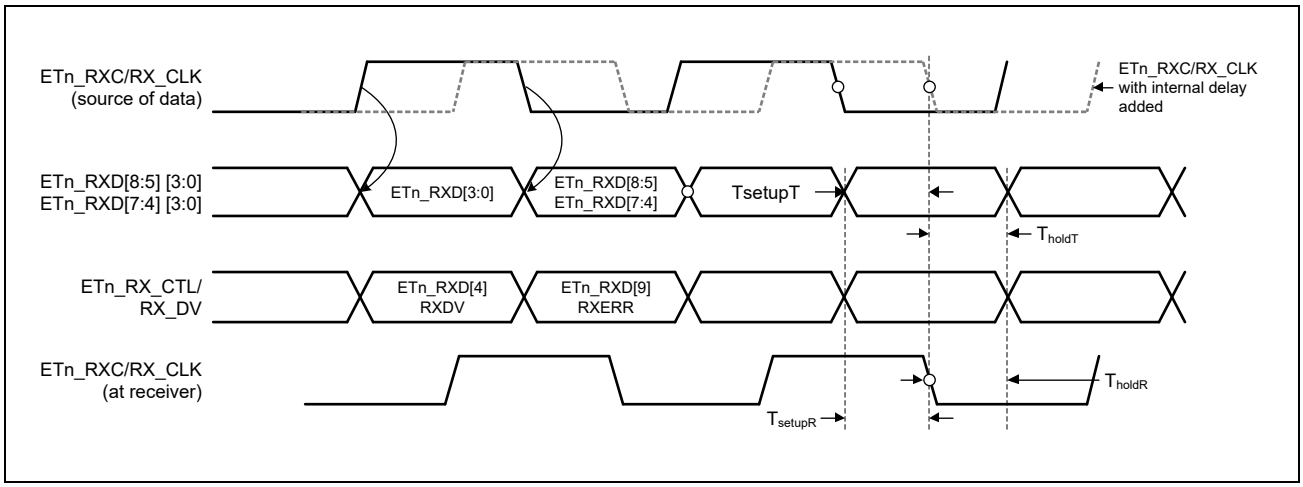


Figure 47.24 Multiplexing & Timing Diagram — RGMII-ID (Receiver)

47.5.7 JTAG Debugger Interface Access Timing

Table 47.26 Debugger IF Timing

Item	Symbol	Min.	Max.	Unit	Figures
TCK_SWCLK cycle time	t_{TCKcyc}	50	—	ns	**Figure 47.25**
TCK_SWCLK high pulse width	t_{TCKH}	20	—	ns	**Figure 47.26**
TCK_SWCLK low pulse width	t_{TCKL}	20	—	ns	
TDI setup time	t_{TDIS}	15	—	ns	
TDI hold time	t_{TDIH}	15	—	ns	
TMS_SWDIO setup time	t_{TMSS}	15	—	ns	
TMS_SWDIO hold time	t_{TMSH}	15	—	ns	
SWDIO delay time	t_{SWDO}	—	14	ns	
TDO delay time	t_{TDOD}	—	14	ns	
Capture register setup time	t_{CAPTS}	10	—	ns	**Figure 47.27**
Capture register hold time	t_{CAPTH}	10	—	ns	
Update register delay time	$t_{UPDATED}$	—	20	ns	

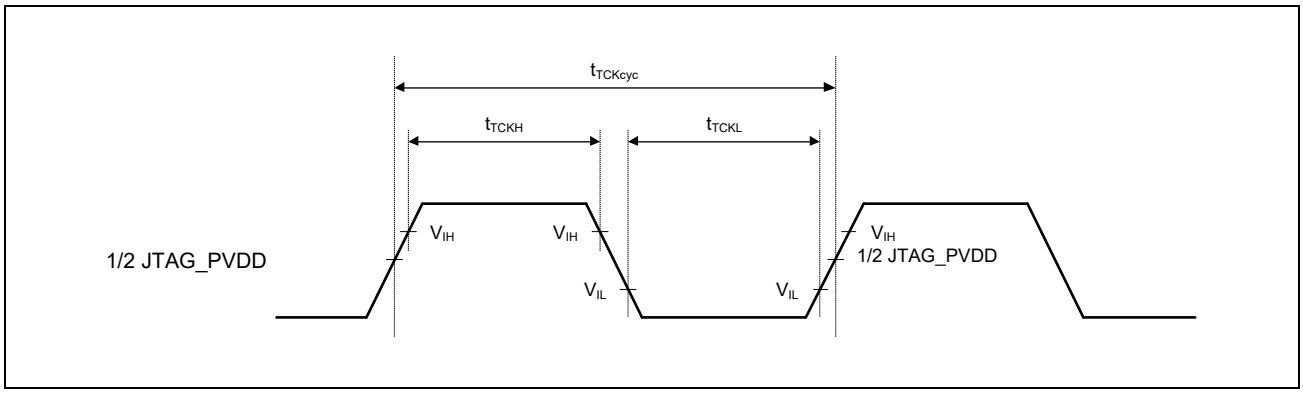


Figure 47.25 TCK_SWDCLK Input Timing

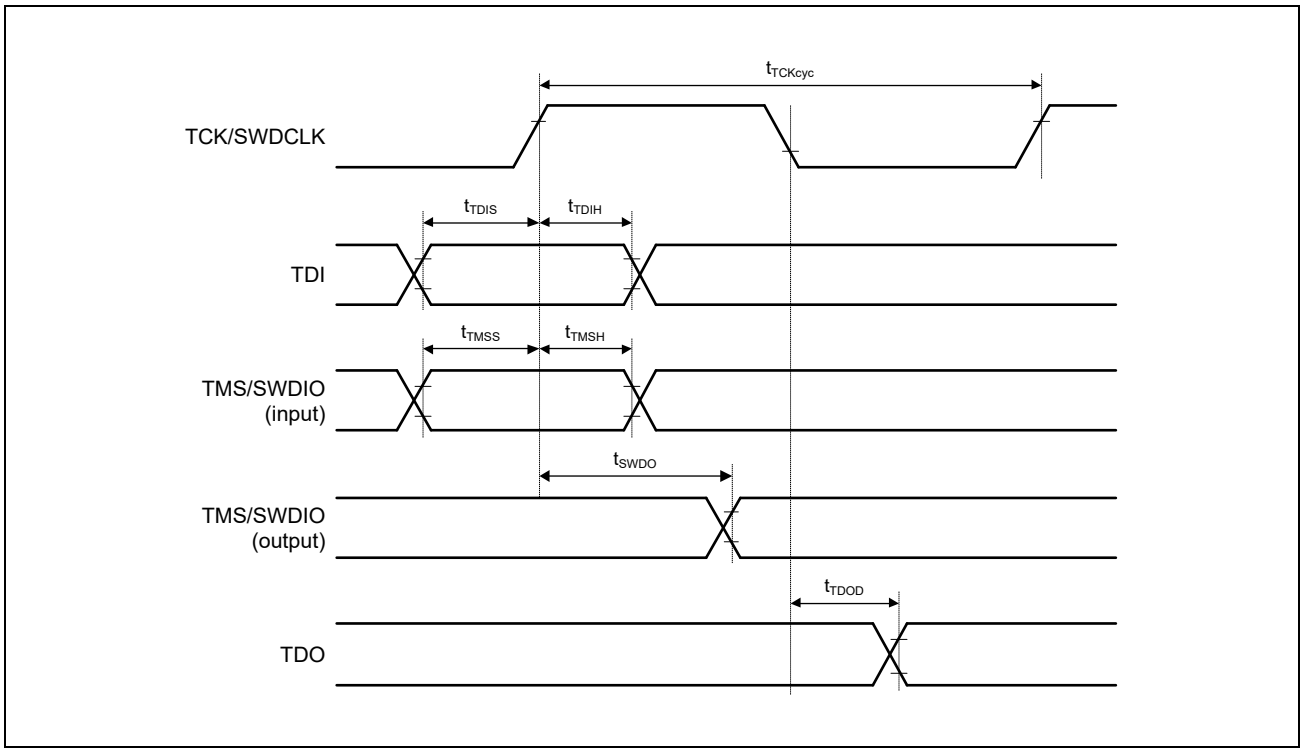


Figure 47.26 Data Transfer Timing

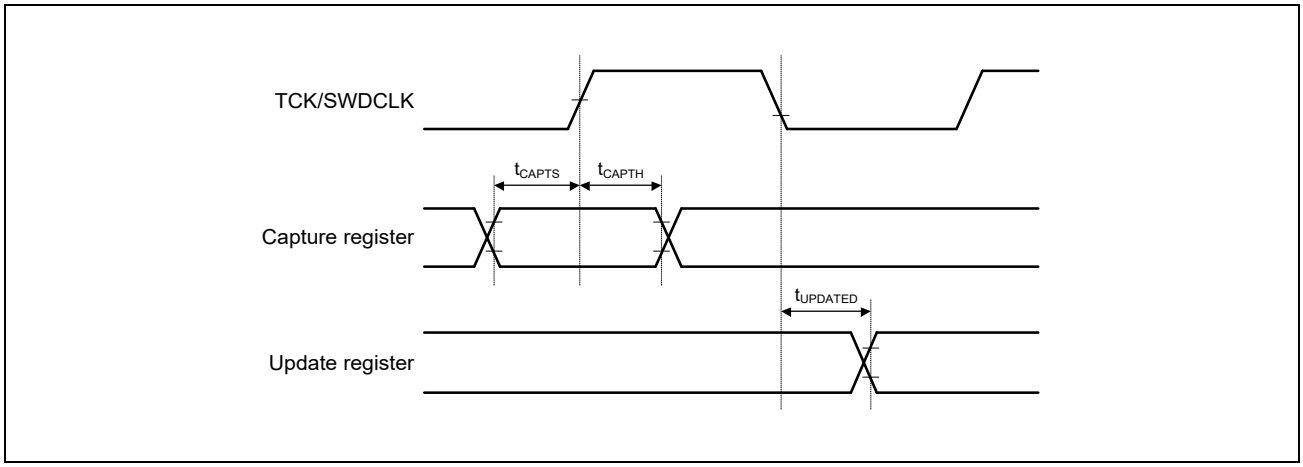


Figure 47.27 Boundary Scan Input/Output I/O Timing

47.5.8 xSPI Timing

Table 47.27 xSPI Timing

Item		Symbol	1.8 V		3.3 V		Unit	Figures
			Min.	Max.	Min.	Max.		
Clock cycle	SDR	t_{SPBcyc}	15	—	15	—	ns	**Figure 47.28**
	DDR		7.5	—	—	—	ns	
Clock output slew rate		t_{SRck}	0.75/0.56*2	—	1.03*6	—	V/ns	
Clock minimum pulse width		t_{CKMPW}	t_{PERIOD} × 0.45	—	t_{PERIOD} × 0.45	—	ns	
DS duty cycle distortion		t_{DSDCD}	0.0	t_{PERIOD} × 0.04	0.0	t_{PERIOD} × 0.04	ns	
DS minimum pulse width		t_{DSMPW}	t_{PERIOD} × 0.41	—	t_{PERIOD} × 0.41	—	ns	
Data input/output slew rate		t_{SR}	0.75/0.56*2	—	1.03*6	—	V/ns	
Data input setup time (to CK)	SDR	t_{SU}	5.0	—	5.0	—	ns	**Figure 47.29**
Data input hold time (to CK)		t_H	1.0	—	1.0	—	ns	
Data output delay time	SDR	t_{OD}	—	8.32*3	—	8.32*3	ns	
	DDR		—	1	—	—	ns	
Data output hold time	SDR	t_{OH}	4.8	—	4.8	—	ns	
	DDR		−1	—	—	—	ns	
Data output buffer off time	SDR	t_{BOFF}	4.8	—	4.8	—	ns	
	DDR		−1	—	—	—	ns	
Data input setup time (to DS)	DDR*1,*3	t_{SU}	−0.6/−0.8	—	−0.6/−0.8	—	ns	**Figure 47.30**
Data input hold time (to DS)		t_H	t_{PERIOD} × 0.41 − 0.6/0.8	—	t_{PERIOD} × 0.41 – 0.6/0.8	—	ns	
Data output setup time (to CK)		t_{SUO}	0.6*5	—	1.0	—	ns	
Data output hold time (to CK)		t_{HO}	0.6*5	—	1.0	—	ns	
CS low to clock high		t_{CSLCKH}	6.0/8.0*2,*4	—	8.0*4	—	ns	**Figure 47.29**, **Figure 47.30**
Clock low to CS high		t_{CKLCSH}	6.0/8.0*2	—	8.0	—	ns	
CS high time		t_{CSTD}	1	16	1	16	t_{PERIOD}	
DS low to CS high		t_{DSLCSH}	6.0/8.0*2	—	10.6	—	ns	**Figure 47.31**
CS high to DS Tri-state		t_{CSHDST}	0.0	t_{PERIOD}	0.0	t_{PERIOD}	ns	
CS low to DS low		t_{CSLDSL}	0.0	—	0.0	—	ns	
DS Tri-state to CS low		t_{DSTCSL}	0.0	—	0.0	—	ns	

Remarks: CK: XSPI_SPCLK
DS: XSPI_DS
CS: XSPI_CS0#, XSPI_CS1#

Note 1. The DS shift setting (WRAPCFG.DSSFTCSx[4:0]) is 0_1000b for xSPI200.

Note 2. Specification at 133 MHz / Specification at 100 MHz

Note 3. These are values when the OEN assertion is extended in the Output Enable Asserting extension bit (COMCFG.OEASTEX = 1b).

Note 4. These are the values when the CS assertion is extended in the CS asserting extension bit (LIOCFGCSn.CSASTEX = 1b) and the CS negation is extended in CS negating extension bit.

Note 5. The standard value for xSPI266 is 0.8 ns.

Note 6. When IOLH register is 10b or more.

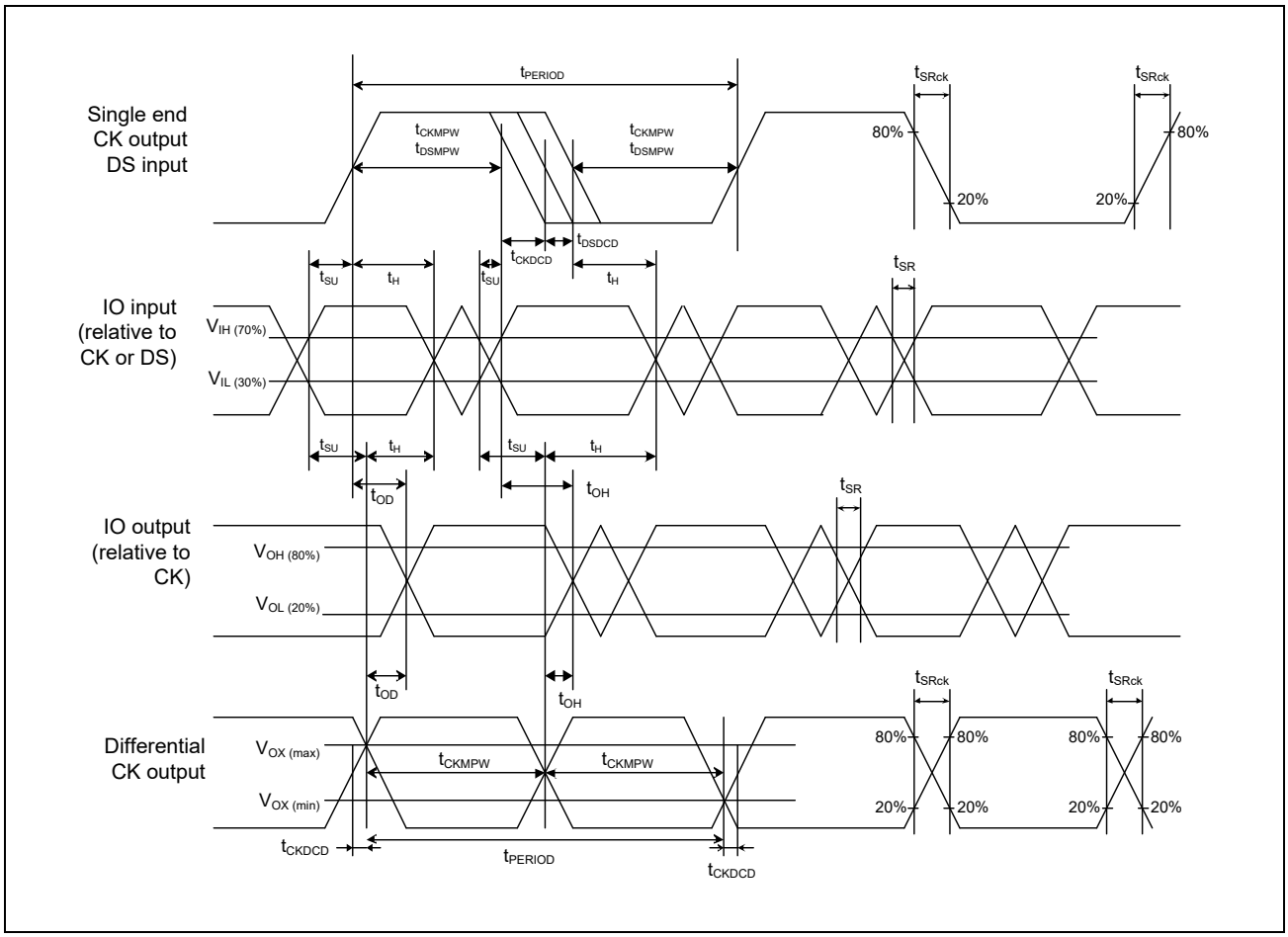


Figure 47.28 xSPI Clock / DS Timing

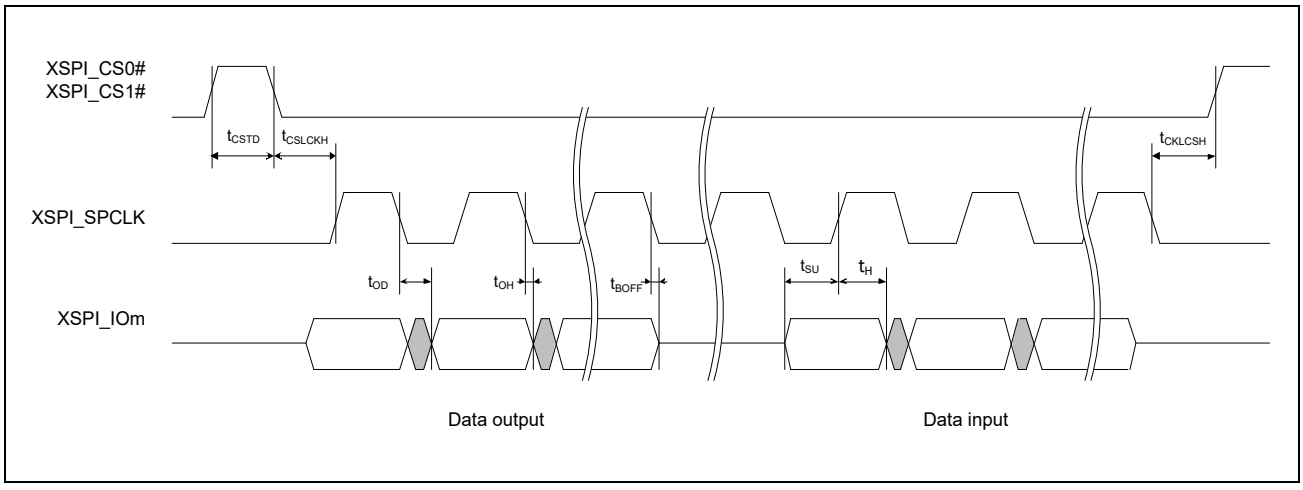


Figure 47.29 SDR Transmit/Receive Timing (1S-1S-1S, 1S-2S-2S, 2S-2S-2S, 1S-4S-4S, 4S-4S-4S)

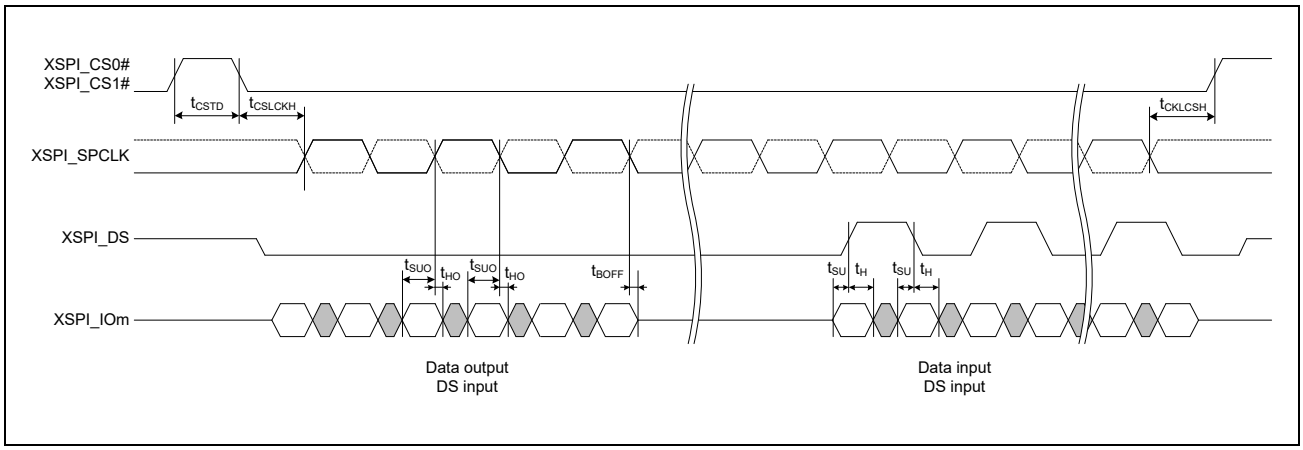


Figure 47.30 DDR Transmit/Receive Timing (8D-8D-8D)

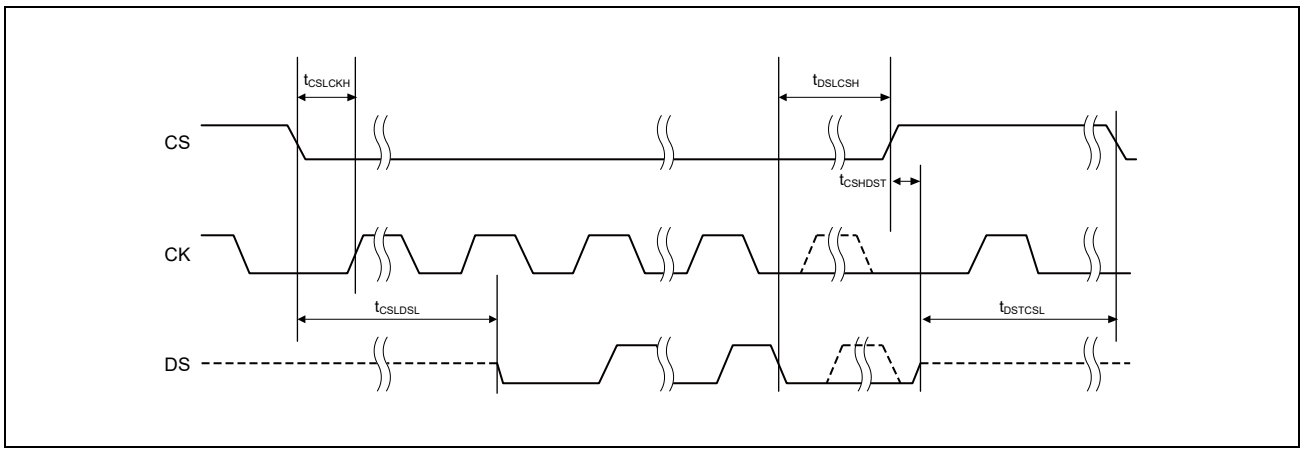


Figure 47.31 DS to CS Signal Timing

47.5.9 Serial Sound Interface (SSIF-2) Access Timing

Table 47.28 SSIF-2 Timing

Item	I/O	Symbol	Min.	Max.	Unit	Figures
Output clock cycle	Output	t_O	80	64000	ns	**Figure 47.32**
Input clock cycle	Input	t_I	80	—	ns	
Clock high	Bidirectional	t_{HC}	32	—	ns	
Clock low		t_{LC}	32	—	ns	
Clock rise time/clock fall time	Output	t_{RC}/t_{FC}	—	25	ns	
Setup time	Input	t_{SR}	25	—	ns	**Figure 47.33**, **Figure 47.34**, **Figure 47.35**
Hold time		t_{HR}	5	—	ns	
SILRCK output delay time	Output	t_{DTR}	−5	25	ns	
Data output delay time (Noise canceler not in use)		t_{DTR}	−5	25	ns	
Data output delay time (Noise canceler in use)		t_{DTR}	10	50	ns	

Note: AC access timing condition: drive ability 12 mA, output load 30 pF

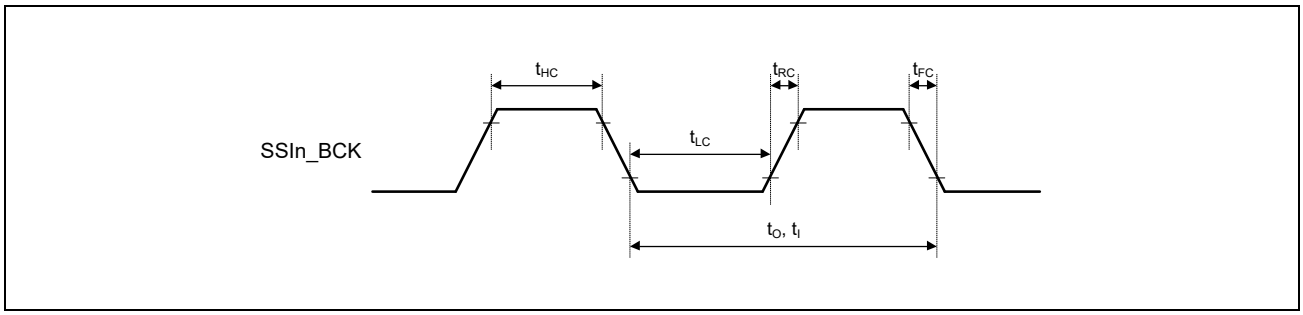


Figure 47.32 Bit Clock Input/Output Timing

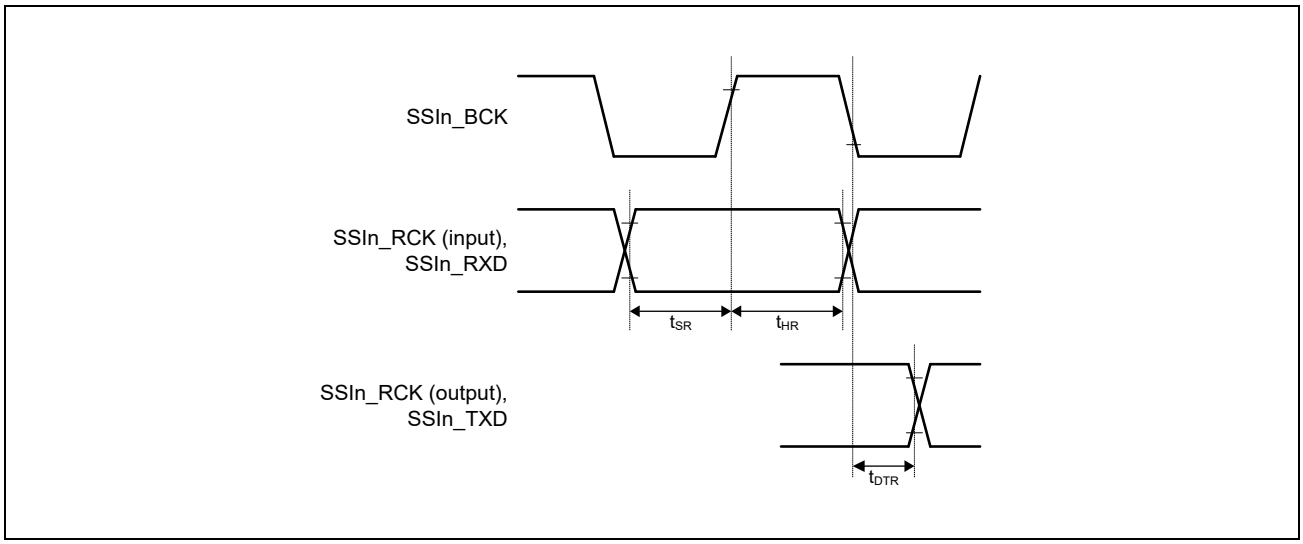


Figure 47.33 Transmission and Reception Timing (SSIBCK Falling Output)

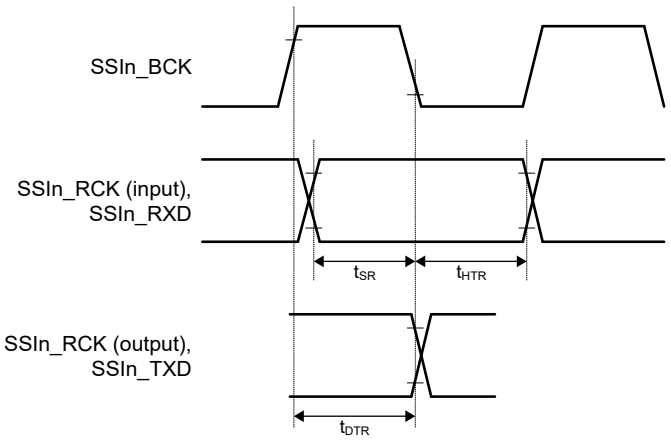
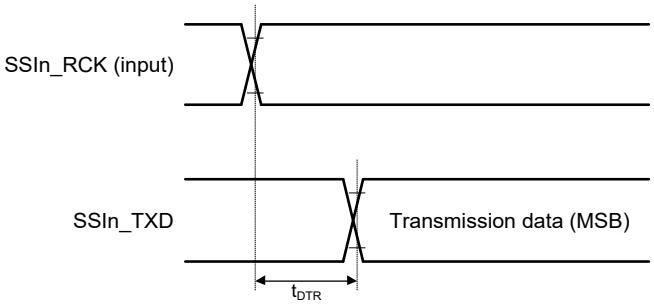


Figure 47.34 Transmission and Reception Timing (SSIBCK Rising Output)



Note: Output timing of the MSB when slave transmission satisfies the following condition a) or b)
a) SSICR.DEL = 1, SSICR.SDTA = 0
b) SSICR.DEL = 1, SSICR.SDTA = 1, without padding

Figure 47.35 Transmission Timing (Slave, in Synchronization with SSILRCK)

47.5.10 CAN-FD Interface Access Timing

Table 47.29 CAN-FD Interface Timing

Item	Symbol	CAN		CAN-FD		Unit	Figures
		Min.	Max.	Min.	Max.		
Internal delay time	t_{node}*1	—	100	—	50	ns	**Figure 47.36**
Transmission rate	—	—	1	—	8	Mbps	

Note: AC access timing condition: drive ability 8 mA, output load 30 pF

Note 1. Internal delay time (t_{node}) = Internal transfer delay time (t_{output}) + Internal receive delay time (t_{input})

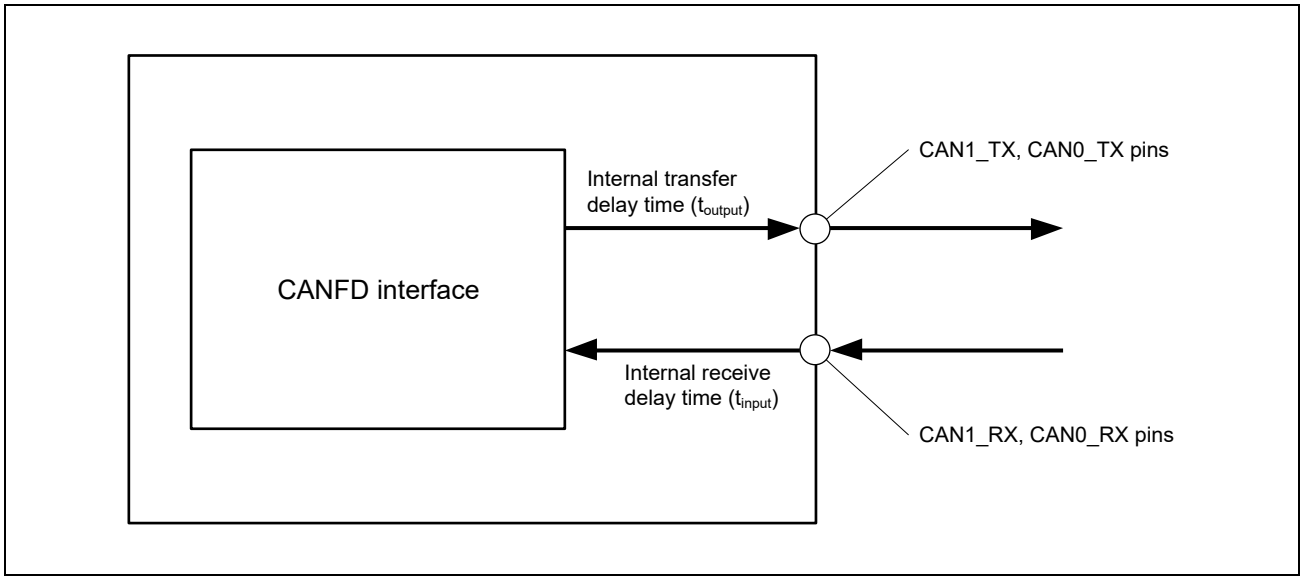


Figure 47.36 CAN-FD Interface Condition

47.5.11 Multi-Function Timer Pulse Unit 3 (MTU3a) Access Timing

Table 47.30 MTU3a Timing

Item			Symbol	Min.	Max.	Unit*1	Figures
MTU3a	Input capture input pulse width	Single-edge setting	t_{MTICW}	1.5	—	t_{p1cyc}*1	**Figure 47.37**
		Both-edge setting		2.5	—		
	Timer clock pulse width	Single-edge setting	t_{MTCKWH}, t_{MTCKWL}	1.5	—	t_{p1cyc}*1	**Figure 47.38**
		Both-edge setting		2.5	—		
		Phase counting mode		2.5	—		

Note: AC access timing condition: drive ability 8 mA, output load 30 pF

Note 1. t_{p1cyc} indicates peripheral clock means MTU_X_MCLK_MTU3 (P0ϕ).

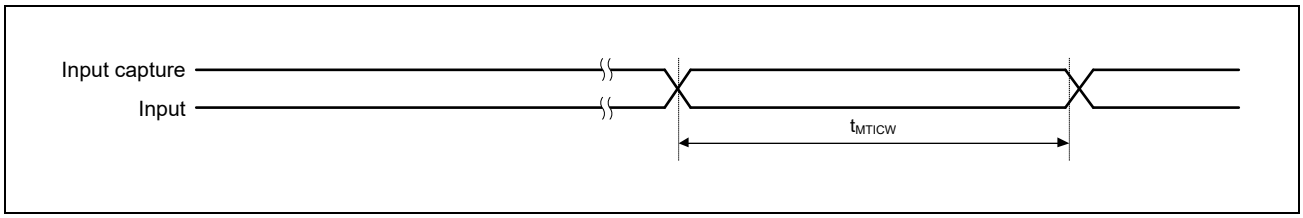


Figure 47.37 MTU3a Input Capture Input Timing

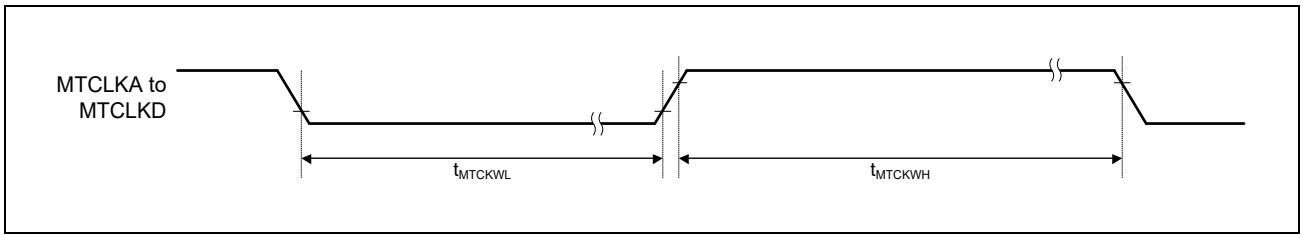


Figure 47.38 MTU3a Clock Input Timing

47.5.12 Port Output Enable 3 (POE3) Access Timing

Table 47.31 POE3 Timing

Item		Symbol	Min.	Max.	Unit	Figures
POE3	POEn# input pulse width	t_{POE3W}	1.5	—	t_{p1cy}*1	**Figure 47.39**

Note 1. t_{p1cyc} indicates peripheral clock means POE3_CLKM_POE (P0ϕ).

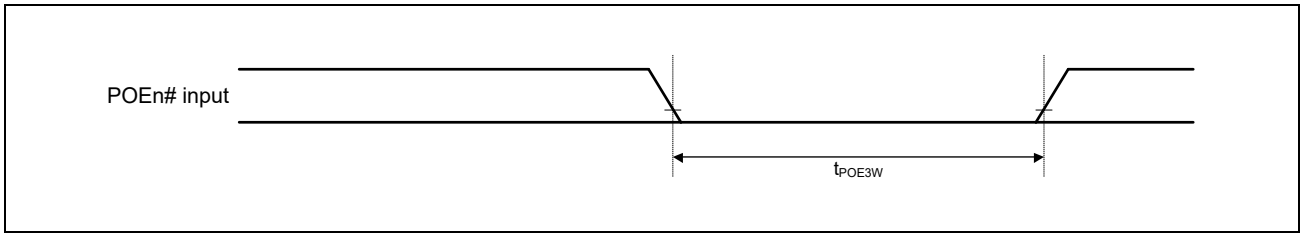


Figure 47.39 POEn# Input Pulse Timing

47.5.13 General PWM Timer (GPT) Access Timing

Table 47.32 GPT Timing

Item			Symbol	Min.	Max.	Unit	Figures
GPT	Input capture input pulse width	Single-edge setting	t_{GTICW}	1.5	—	t_{p1cyc}*1	**Figure 47.40**
		Both-edge setting		2.5	—		

Note: AC access timing condition: drive ability 8 mA, output load 30 pF

Note 1. t_{p1cyc} indicates peripheral clock means GPT_PCLK (P0ϕ).

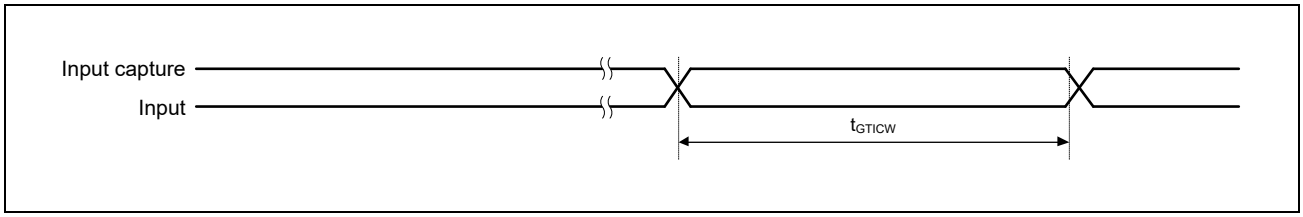


Figure 47.40 GPT Input Capture Input Timing

47.5.14 Port Output Enable for GPT (POEG) Access Timing

Table 47.33 POEG Timing

Item		Symbol	Min.	Max.	Unit	Figures
POEG	POEG input pulse width	t_{POEGW}	3	—	t_{p1cyc}*1	**Figure 47.41**

Note 1. t_{p1cyc} indicates peripheral clock means POEG_x_CLKP (P0ϕ) (x = A, B, C, D).

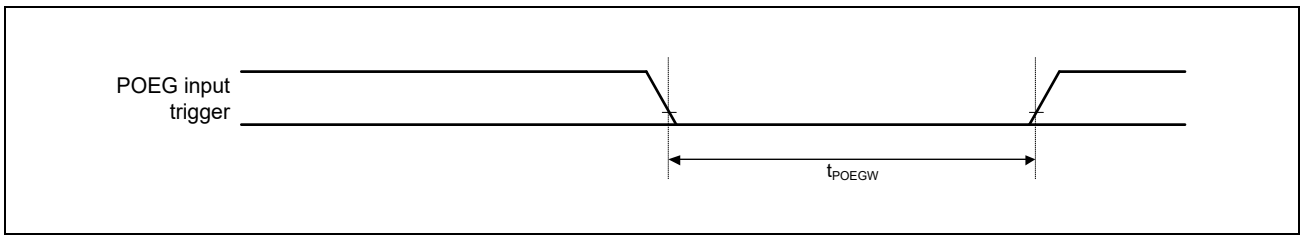


Figure 47.41 POEG Input Trigger Timing

47.5.15 I²C Bus Interface Access Timing

Table 47.34 I²C Bus Interface Timing

Item	Symbol	I/O	Standard Mode (Sm)		Fast Mode (Fm)		Fast Mode Plus (Fm+)		Unit
			Min.	Max.	Min.	Max.	Min.	Max.	
SCL clock frequency	f_{CLK}	I/O	0	100	0	400	0	1000	kHz
Bus free time (between stop and start condition)	t_{BUF}	I/O	4.7	—	1.3	—	0.5	—	µs
Hold time*1	$t_{HD:STA}$	I/O	4.0	—	0.6	—	0.26	—	µs
Low period of SCL clock	t_{LOW}	I/O	4.7	—	1.3	—	0.5	—	µs
High period of SCL clock	t_{HIGH}	I/O	4.0	—	0.6	—	0.26	—	µs
Setup time for start / restart condition	$t_{SU:STA}$	I/O	4.7	—	0.6	—	0.26	—	µs
Data hold time (I²C bus device)	$t_{HD:DAT}$	I/O	0*2	—	0*2	—	0	—	µs
Data setup time	$t_{SU:DAT}$	I/O	250	—	100*3	—	50	—	ns
SDA and SCL signal rise time	t_R	Input	—	1000	20	300	—	120	ns
SDA and SCL signal fall time*3	t_F	Input	—	300	20 × (PV_{DD18}/ 5.5 V)	300	20 × (PV_{DD18}/ 5.5 V)	120	ns
		Output	—	300	20 × (PV_{DD18}/ 5.5 V)*6	300*6	20 × (PV_{DD18}/ 5.5 V)*7	120*7	ns
Setup time for STOP condition	$t_{SU:STO}$	I/O	4.0	—	0.6	—	0.26	—	µs
Capacitive load for each bus line	C_b	—	—	400*4	—	400*4	—	550*4	pF
Pulse width of spikes that must be suppressed by the input filter	t_{SP}	Input	—	—	0	50*5	0	50*5	ns

Note: In the above table and subsequently, SCL and SDA refer to the RIICnSCL and RIICnSDA signals, respectively.

Note: AC access timing condition: drive ability 4 mA, output load 400 pF

Note 1. The first clock pulse is generated on the SCL line after the start condition has been issued and the hold time has elapsed.

Note 2. This module requires a minimum of 300 ns hold time internally for the SDA signal to handle the period over which the falling edge of SCL has not reached a defined level (time until the CnSCL signal reaches V_{IL} (max.) from V_{IH} (min.)).

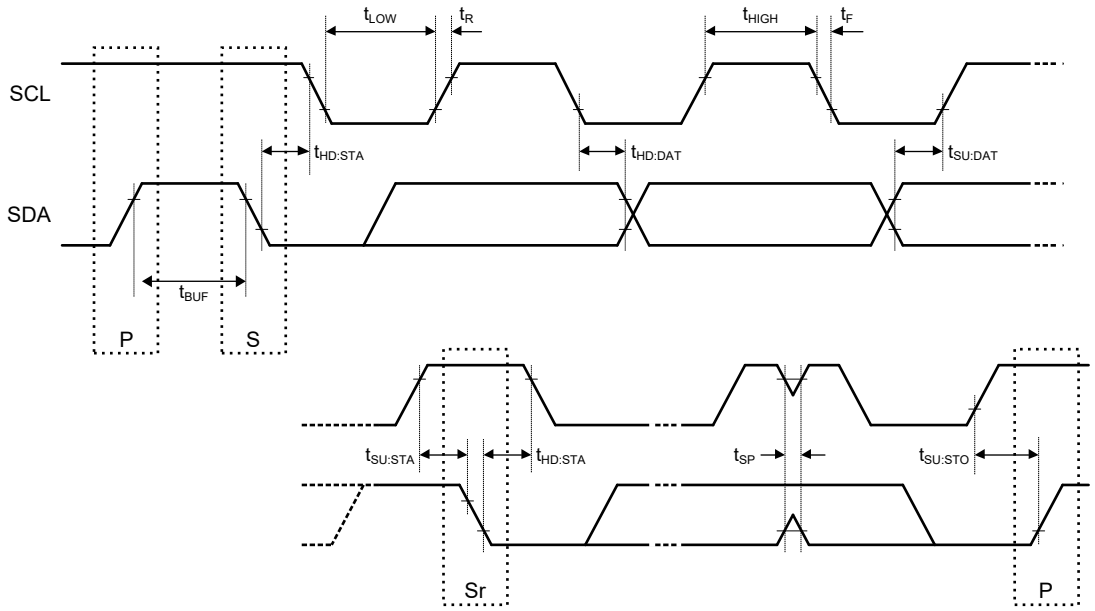
Note 3. The fast-mode I²C bus device can be used in the standard mode I²C bus system. In this case, the minimum value of the data setup time ($t_{SU:DAT}$ (min.) 250 [ns]) must be satisfied.
If the system does not extend the low period of SCL clock (t_{LOW}), this condition is automatically satisfied. If the system extends the low period of SCL clock (t_{LOW}), transmit the subsequent data bit to the SDA line before the SCL line is released (t_R (max.) + $t_{SU:DAT}$ (min.) = 1000 + 250 = 1250 [ns]: (standard mode I²C bus specification)).

Note 4. Total capacitance of one bus line. The allowable maximum bus capacitance may differ from this specification, depending on the actual operating voltage and frequency of an application. For techniques to cope with a large bus capacitance, see the I²C bus specification provided by NXP Semiconductors.

Note 5. Noise is removed by the analog and digital input filters. The level of noise reduction of the digital input filter is determined by the period of internal reference clock (IICϕ) and the NF[1:0] bits in RIICnMR3. For details, refer to **Section 25, I²C Bus Interface**.

Note 6. When using RIIC ch2 or RIIC ch3, the following limitations exists.
- Driving ability: IOLH_xx[1:0] = 01b
- Tfmin cannot meet the specification when the load of bus line is less than 200 pF.
- The range of pull-up resistance (Rp) (in the case of 200/300/400 pF)
  Cb is 200 pF: 1076.7 to 1770Ω, Cb is 300 pF: 966.7 to 1180Ω, Cb is 400 pF: 856.7 to 885Ω

Note 7. When using RIIC ch2 or RIIC ch3, the following limitations exists.
- Driving ability: IOLH_xx[1:0] = 11b
- The range of pull-up resistance (Rp) (in the case of 10/150/275/400/550 pF)
  Cb is 10 pF: 300 to 14163Ω, Cb is 150 pF: 300 to 944Ω, Cb is 275 pF: 300 to 515Ω, Cb is 400 pF: 300 to 354Ω, Cb is 550 pF: 300Ω (fixed)
- Tfmin cannot meet the specification in fast-mode plus.



Note: P: Stop condition
S: Start condition
Sr: Start condition for retransmission

Figure 47.42 Input/Output Timing

47.5.16 I3C Bus Interface Access Timing

Table 47.35 Characteristics of the SDA and SCL Bus Lines for Standard, Fast, and Fast-Mode Plus I²C-Bus Devices*1

Item	Symbol	Conditions	Standard-Mode		Fast-Mode		Fast-Mode Plus		Unit
			Min.	Max.	Min.	Max.	Min.	Max.	
SCL clock frequency	f_{SCL}		0	100	0	400	0	1000	kHz
Hold time (repeated) START condition	$t_{HD:STA}$	After this period,the first clock pulse is generated	4.0	—	0.6	—	0.26	—	µs
LOW period of the SCL clock	t_{LOW}		4.7	—	1.3	—	0.5	—	µs
HIGH period of the SCL clock	t_{HIGH}		4.0	—	0.6	—	0.26	—	µs
Setup time for a repeated START condition	$t_{SU:STA}$		4.7	—	0.6	—	0.26	—	µs
Data hold time*2	$t_{HD:DAT}$	I²C-bus device	0 *3	—*4	0*3	—*4	0	—	µs
Data setup time	$t_{SU:DAT}$		250	—	100	—	50	—	ns
Rise time of both SDA and SCL signals	t_r		—	1000	20	300	—	120	ns
Fall time of both SDA and SCL signals*3,*6,*7	t_f		—	300	20 × (I3C_PV_{DD}/ 5.5 V)	300	20 × (I3C_PV_{DD}/ 5.5 V)*8	120*7	ns
Setup time for STOP condition	$t_{SU:STO}$		4.0	—	0.6	—	0.26	—	µs
Bus free time between a STOP and START condition	t_{BUF}		4.7	—	1.3	—	0.5	—	µs

Note 1. All values referred to V_{IH}(min) (0.3 × I3C_PV_{DD}) and V_{IL}(max) (0.7 × I3C_PV_{DD}) levels, refer to **Table 47.6**

Note 2. $t_{HD:DAT}$ is the data hold time that is measured from the falling edge of SCL, applies to data in transmission and the acknowledge.

Note 3. A device must internally provide a hold time of at least 300 ns for the SDA signal (with respect to the V_{IH}(min) of the SCL signal) to bridge the undefined region of the falling edge of SCL.

Note 4. The maximum $V_{HD:DAT}$ could be 3.45 µs and 0.9 µs for Standard mode and Fast mode, but must be less than the maximum of $t_{VD:DAT}$ or $t_{VD:ACK}$ by a transition time. This maximum must only be met if the device does not stretch the LOW period () of the SCL signal. If the clock stretches the SCL, the data must be valid by the setup time before it releases the clock.

Note 5. A Fast-mode I²C-bus device can be used in a Standard-mode I²C-bus system, but the requirement $t_{SU:DAT}$ 250 ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line t_r(max) + $t_{SU:DAT}$ = 1000 + 250 = 1250 ns (according to the Standard-mode I²C-bus specification) before the SCL line is released. Also the acknowledge timing must meet this set-up time.

Note 6. The maximum t_f for the SDA and SCL bus lines is specified at 300 ns. The maximum fall time for the SDA output stage t_f is specified at 250 ns. This allows series protection resistors to be connected in between the SDA and the SCL pins and the SDA/SCL bus lines without exceeding the maximum specified t_f.

Note 7. In Fast-mode Plus, fall time is specified the same for both output stage and bus timing. If series resistors are used, designers should allow for this when considering bus timing.

Note 8. Necessary to be backwards compatible to Fast-mode.

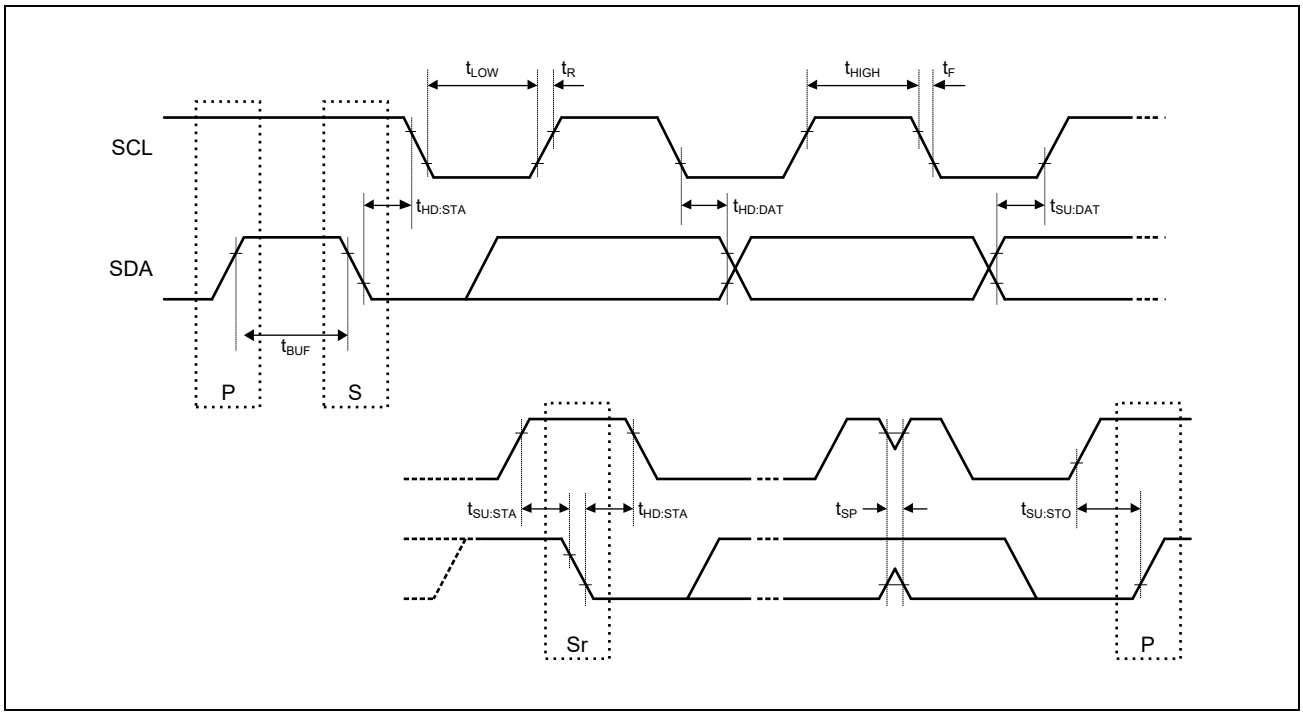


Figure 47.43 Input/Output Timing

Table 47.36 I3C Open Drain Timing Parameters

Item	Symbol	I3C Open Drain Mode			Figures	Notes
		Min.	Max.	Unit		
Low Period of SCL Clock	t_{LOW_OD}	200	—	ns	**Figure 47.47**	*1, *2
	$t_{DIG\_OD\_L}$	t_{LOW_ODmin} + t_{fDA_ODmin}	—	ns	**Figure 47.47**	
High Period of SCL Clock	t_{HIGH}	—	41	ns	**Figure 47.45**	*3, *4
	t_{DIG_H}	—	t_{HIGH} + t_{CF}	ns	**Figure 47.45**, **Figure 47.54**	
Fall Time of SDA Signal	t_{fDA_OD}	t_{CF}	12	ns	**Figure 47.47**	
SDA Data Setup Time During Open Drain Mode	t_{SU_OD}	3	—	ns	**Figure 47.46**, **Figure 47.47**	*1
Clock After START(S) Condition	t_{CAS}	38.4 nano	For ENTAS0: 1 µ For ENTAS1: 100 µ For ENTAS2: 2 milli For ENTAS3: 50 milli	seconds	**Figure 47.47**	*5, *6
Clock Before STOP(P) Condition	t_{CBP}	t_{CASmin}/2	—	seconds	**Figure 47.48**	
Current Master to secondary Master Overlap time during handoff	$t_{MMOverlap}$	$t_{DIG\_OD\_Lmin}$	—	ns	**Figure 47.53**	
Bus Available Condition	t_{AVAL}	1	—	µs	—	*7
Bus Idle Condition	t_{IDLE}	200	—	µs	—	
Time Internal Where New Master Not Driving SDA Low	t_{MMLock}	$t_{AVALmin}$	—	µs	**Figure 47.53**	

Note 1. This is approximately equal to t_{LOWmin} + t_{DS_ODmin} + t_{fDA_ODtyp} + t_{SU_ODmin}.

Note 2. The Master may use a shorter low period if it knows that this is safe, i.e., that SDA is already above V_{IH}.

Note 3. Based on t_{SPIKE}, rise and fall times, and interconnect.

Note 4. This maximum High period may be exceeded when the signals can be safely seen by Legacy I^2C Devices, and/or in consideration of the interconnect (e.g., a short bus).

Note 5. On a Legacy Bus where I^2C Devices need to see Start, the t_{CAS}(min) value is further constrained.

Note 6. Slaves that do not support the optional ENTASx CCCs shall use the t_{CAS}(max) value shown for ENTAS3.

Note 7. On a Mixed Bus with Fm Legacy I^2C Devices, t_{AVAL} is 300 ns shorter than the Fm Bus Free Condition time (t_{BUF}).

Table 47.37 I3C Push-Pull Timing Parameters for SDR Mode

Item		Symbol	Min.	Typ.	Max.	Unit	Figures	Notes
SCL Clock Frequency		f_{SCL}	0.01	12.5	12.9	MHz	—	*1
SCL Clock Low Period		t_{LOW}	24	—	—	ns	**Figure 47.44**	—
		t_{DIG_L}	32	—	—	ns	**Figure 47.45**	*2, *4
SCL Clock High Period for Mixed Bus		t_{HIGH_MIXED}	24	—	—	ns	**Figure 47.45**	—
		$t_{DIG\_H\_MIXED}$	32	—	45	ns	**Figure 47.45**	*2, *3
SCL Clock High Period		t_{HIGH}	24	—	—	ns	**Figure 47.44**	—
		t_{DIG_H}	32	—	—	ns	**Figure 47.45** **Figure 47.44**	*2
Clock in to Data Out for Slave		t_{SCO}	—	—	12	ns	**Figure 47.50**	*7, *8
SCL Clock Rise Time		t_{CR}	—	—	150e06 × 1 / f_{SCL} (capped at 60)	ns	**Figure 47.44**	*5
SCL Clock Fall Time		t_{CF}	—	—	150e06 × 1 / f_{SCL} (capped at 60)	ns	**Figure 47.44**	*5
SDA Signal Data Hold in Push-Pull Mode	Master	t_{HD_PP}	t_{CR} + 3 and t_{CF} +3	—	—	—	**Figure 47.49**	*4, *6
	Slave	t_{HD_PP}	0	—	—	—	**Figure 47.51**	*6
SDA Signal Data Setup in Push-Pull Mode		t_{SU_PP}	3	—	N/A	ns	**Figure 47.49**, **Figure 47.50**	—
Clock After Repeated START (Sr)		t_{CASr}	t_{CASmin}	—	N/A	ns	**Figure 47.52**	—
Clock Before Repeated START (Sr)		t_{CBSr}	t_{CASmin} / 2	—	N/A	ns	**Figure 47.52**	—
Capacitive Load per Bus Line (SDA/SDL)		C_b	—	—	50	pF	—	—

Note 1. FSCL = 1 / (t_{DIG_L} + t_{DIG_H})

Note 2. t_{DIG_L} and t_{DIG_H} are the clock low and high periods as seen at the receiver end of the I3C bus using V_{IL} and V_{IH} (see **Figure 47.44**).

Note 3. When communicating with an I3C Device on a mixed Bus, the $t_{DIG\_H\_MIXED}$ period must be constrained in order to make sure that I^2C Devices do not interpret I3C signaling as valid I^2C signaling.

Note 4. As both edges are used, the hold time needs to be satisfied for the respective edges; i.e., t_{CF} + 3 for falling edge clocks, and t_{CR} + 3 for rising edge clocks.

Note 5. The clock maximum rise/fall time is capped at 60 ns. For lower frequency rise and fall the maximum value is limited at 60 ns, and is not dependent upon the clock frequency.

Note 6. t_{HD_PP} is a Hold time parameter for Push-Pull Mode that has a different value for Master mode vs. Slave mode. In SDR Mode the Hold time parameter is referred to as t_{HD_SDR}.

Note 7. Devices with more than 12 ns of t_{SCO} delay shall set the limitation bit in the BCR, and shall support the GETMXDS CCC to allow the Master to read this value and adjust computations accordingly. For purposes of system design and test conformance, this parameter should be considered together with pad delay, bus capacitance, propagation delay, and clock triggering points.

Note 8. Pad delay based on 90Ω/4 mA driver and 50 pF load. Note that Master may be a Slave in a multi-Master system, and thus shall also adhere to this requirement.

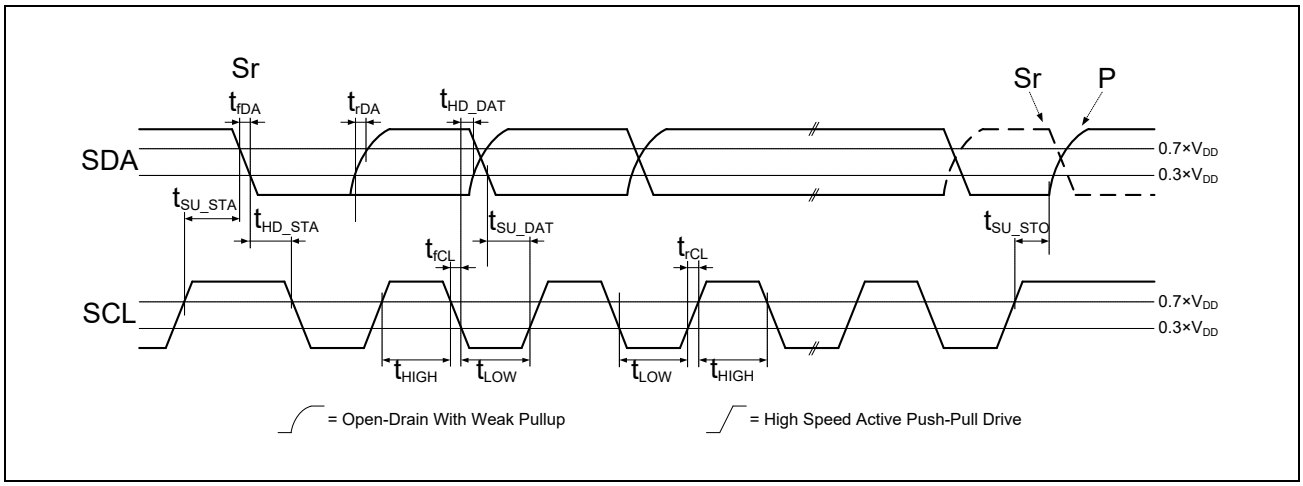


Figure 47.44 I3C Legacy Mode Timing

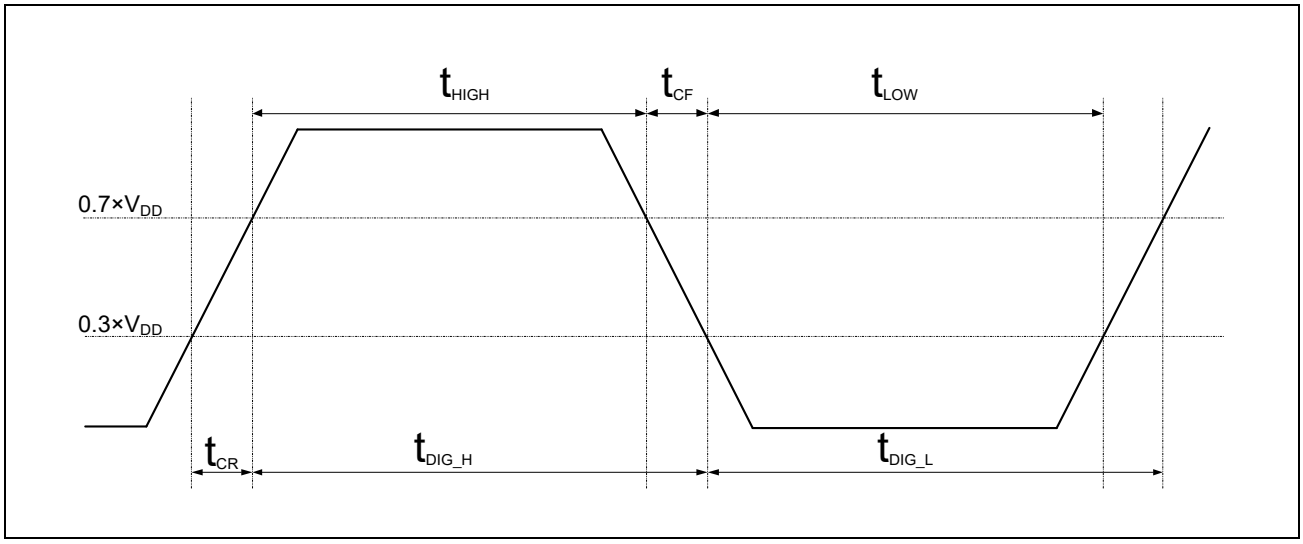


Figure 47.45 t_{DIG_H} and t_{DIG_L}

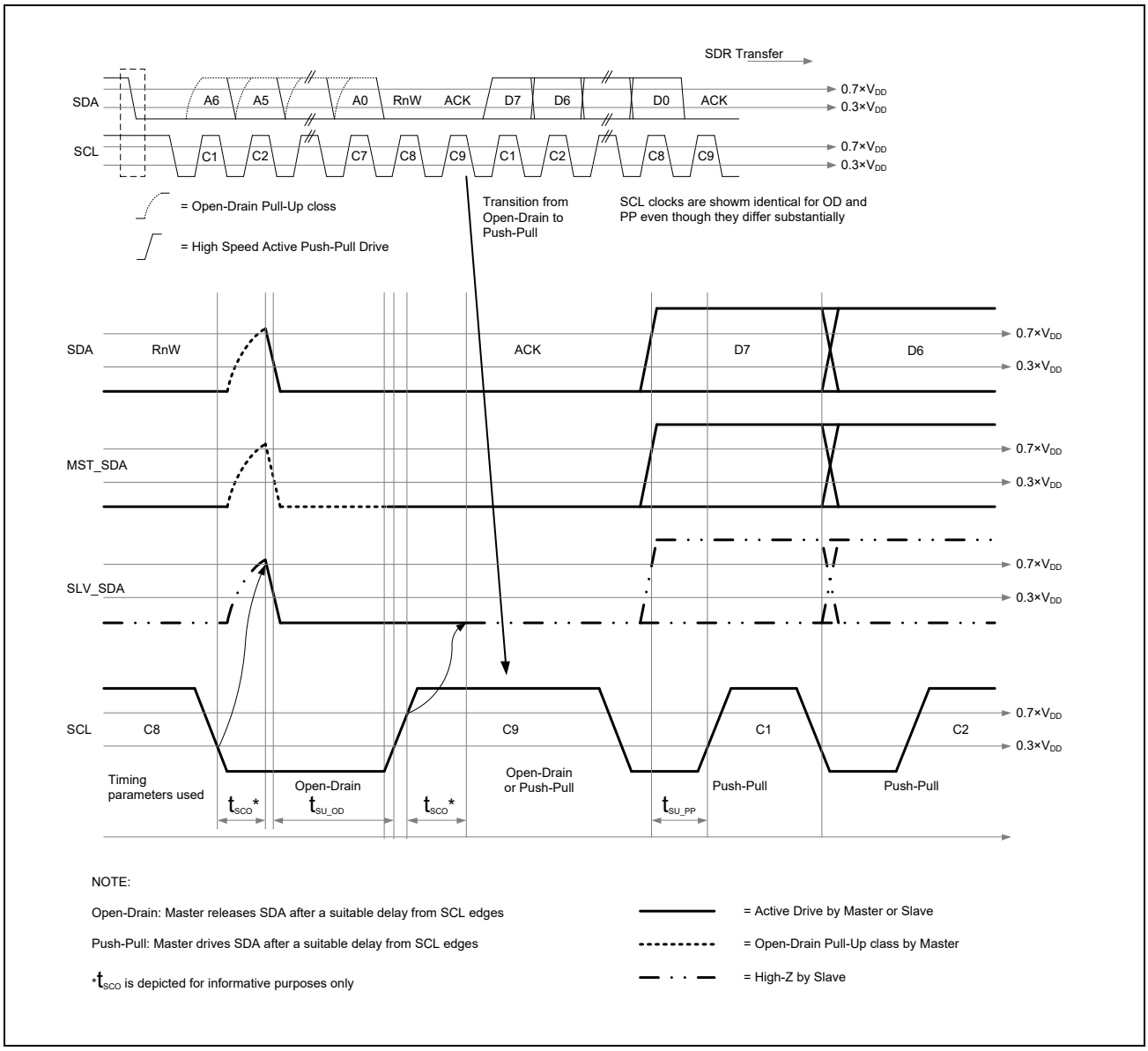


Figure 47.46 I3C Data Transfer – ACK by Slave

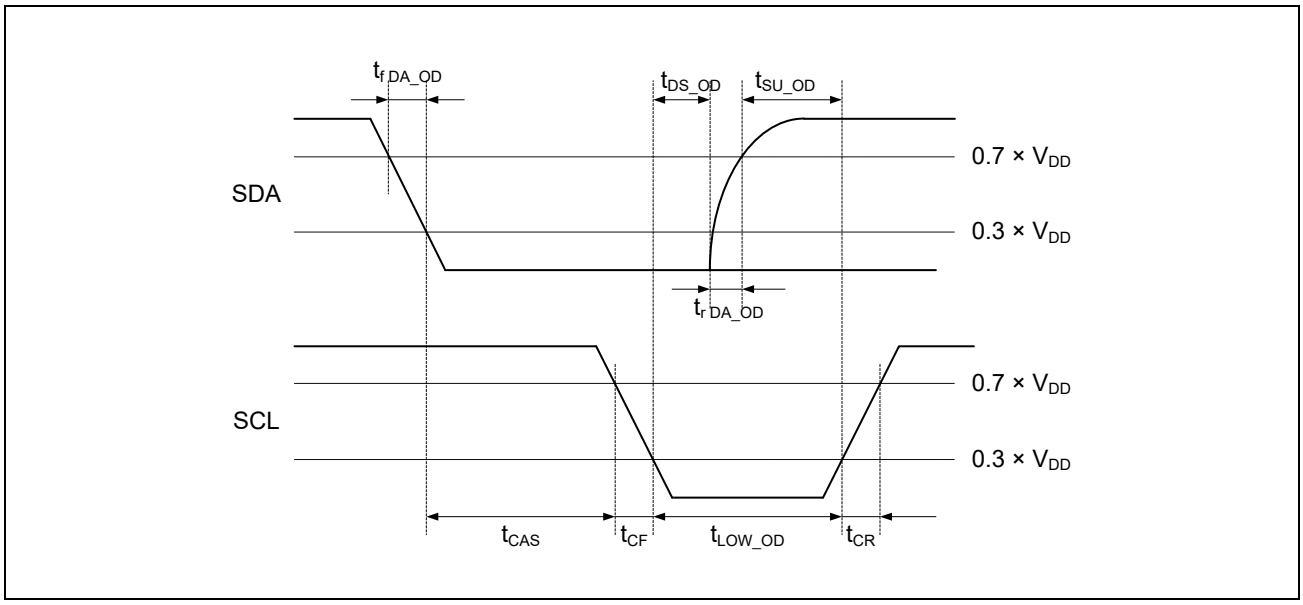


Figure 47.47 I3C START Condition Timing

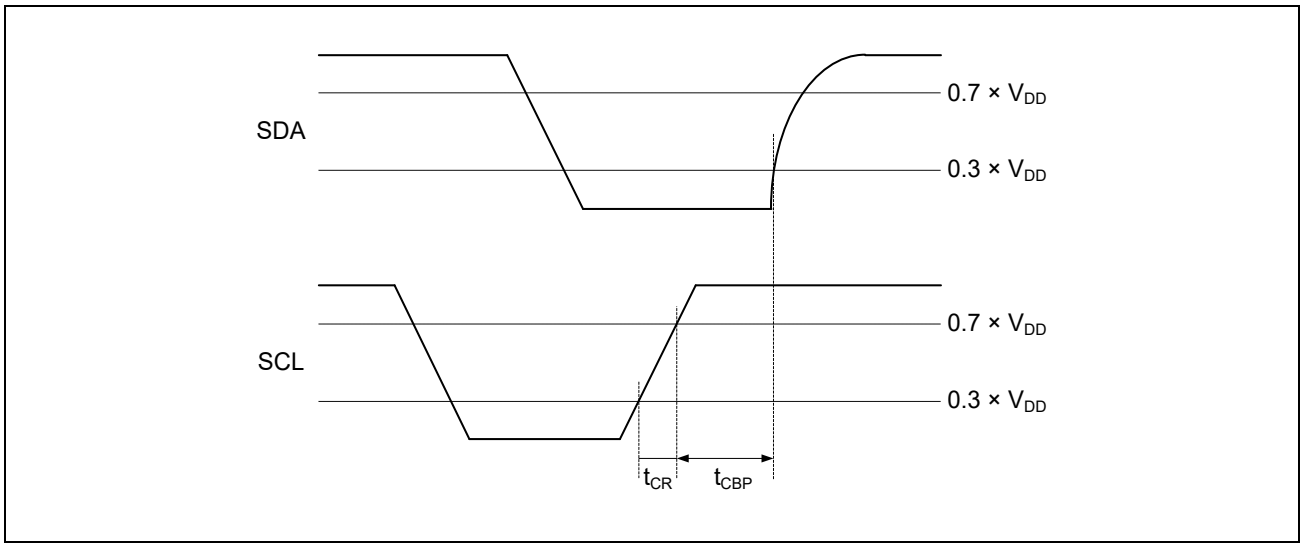


Figure 47.48 I3C STOP Condition Timing

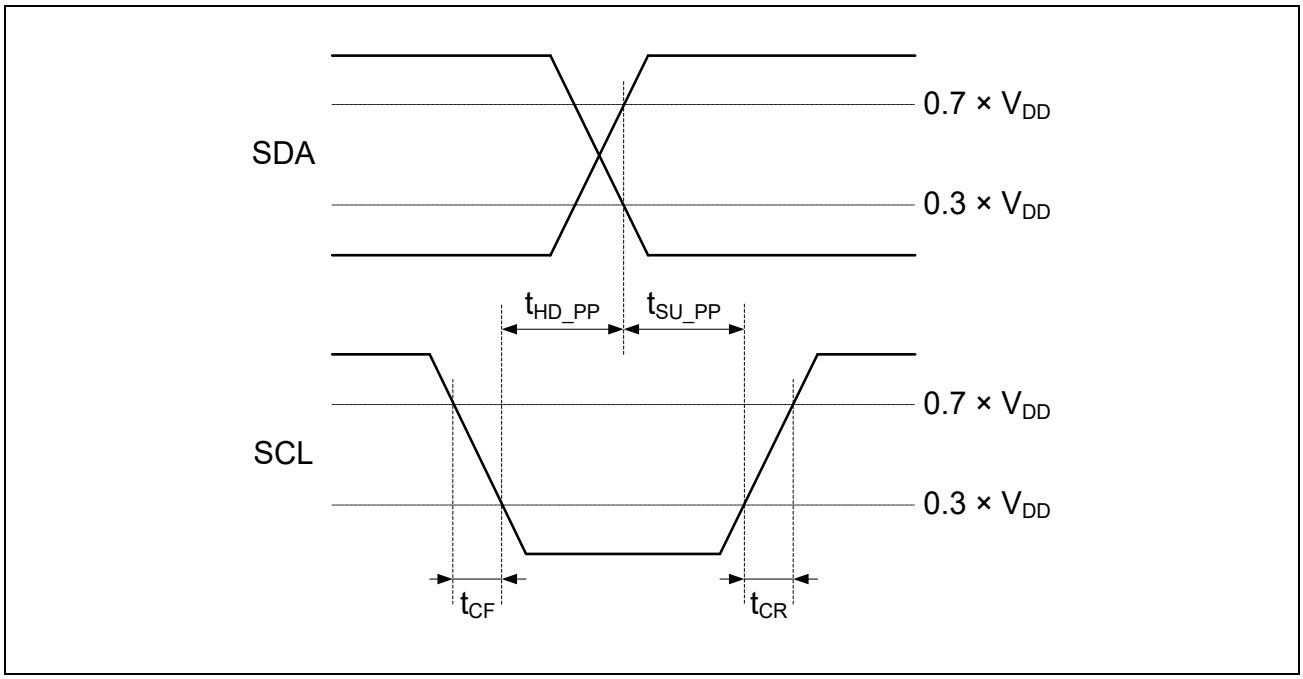


Figure 47.49 I3C Master Out Timing

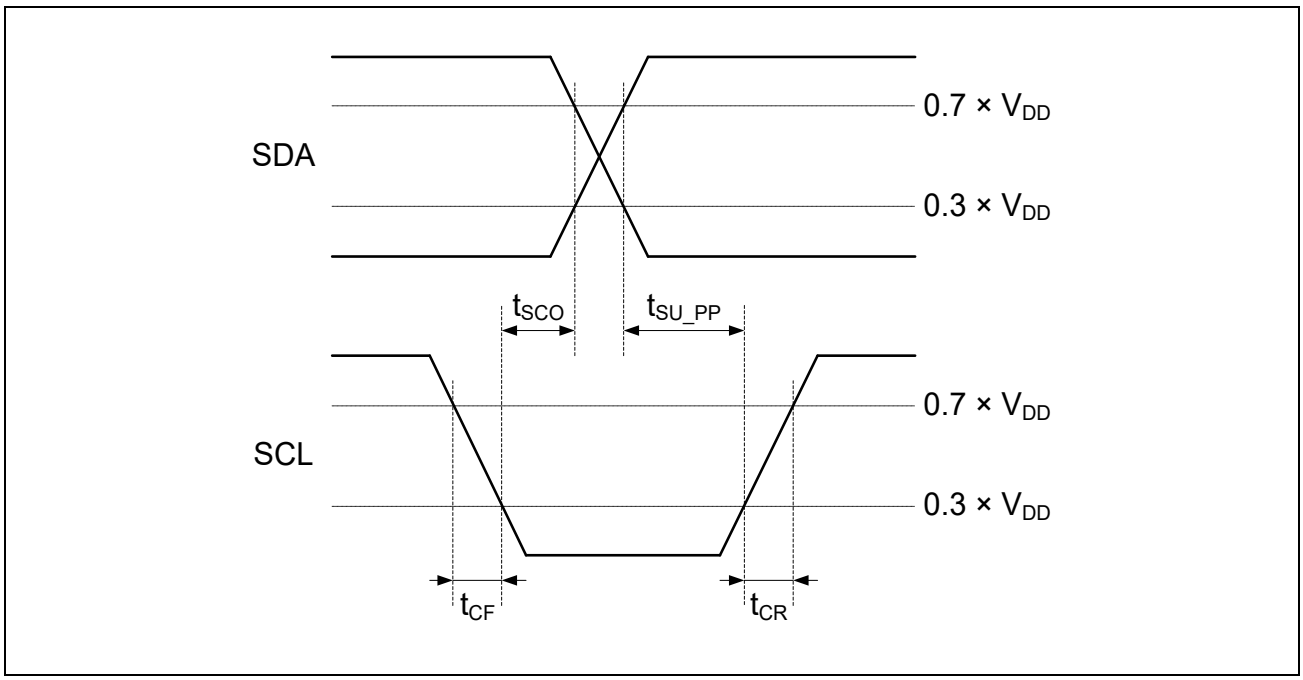


Figure 47.50 I3C Slave Out Timing

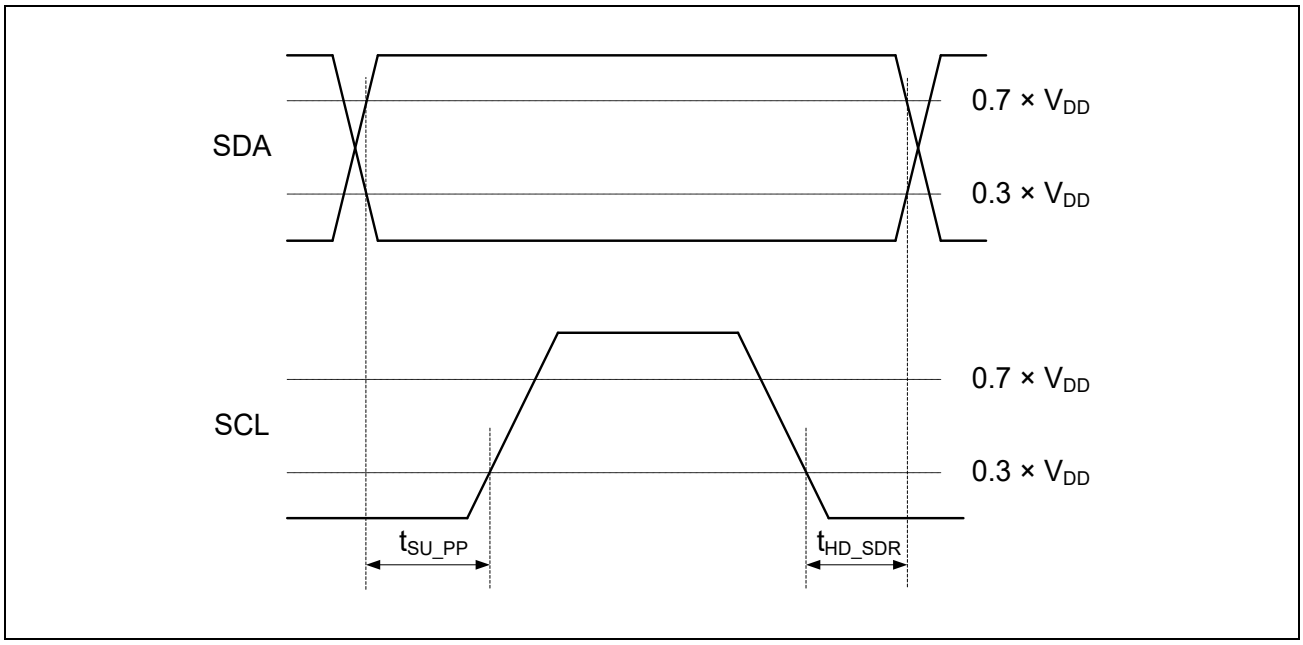


Figure 47.51 Master SDR Timing

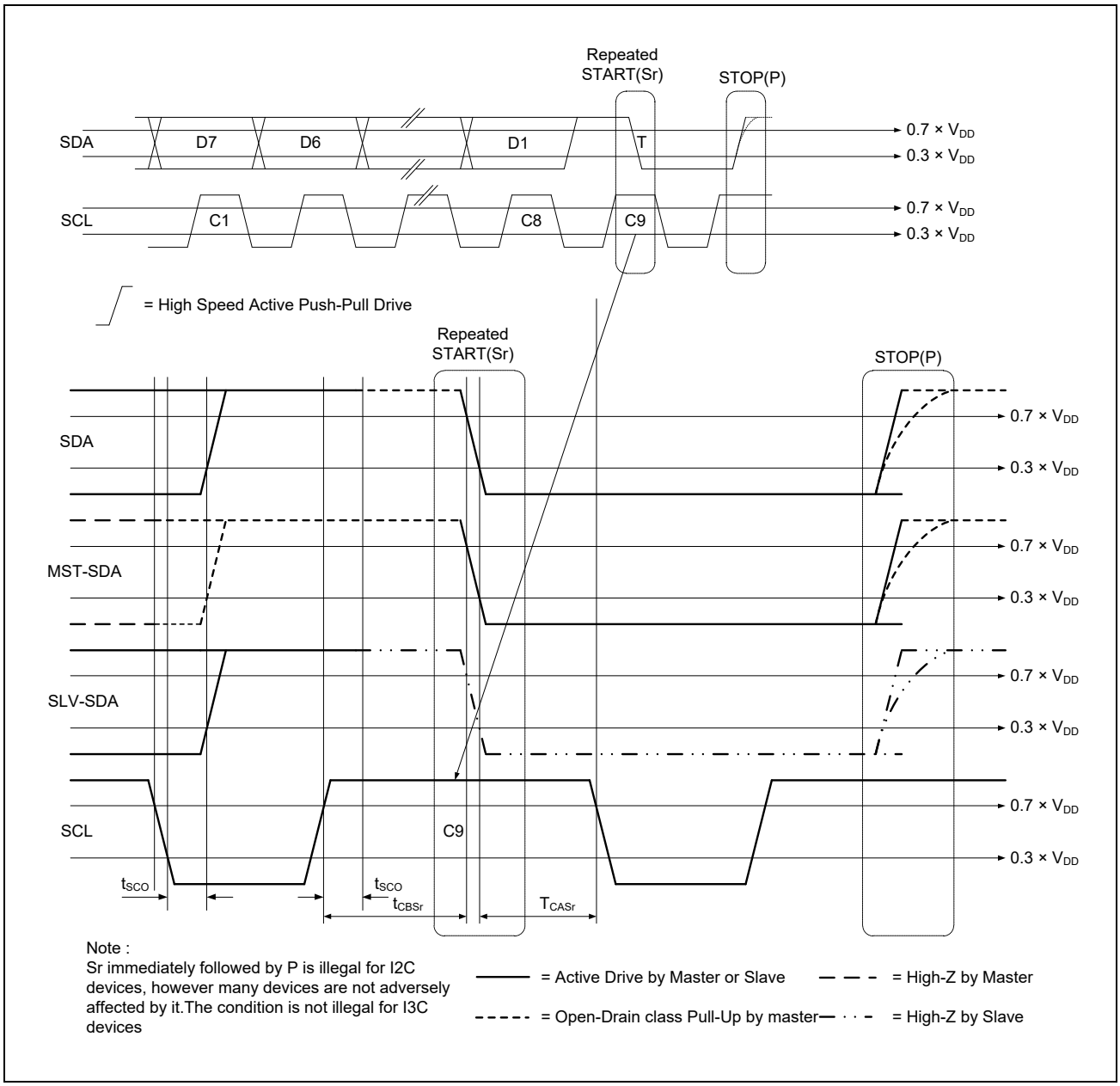


Figure 47.52 T-Bit When Master Ends Read with Repeated START and STOP

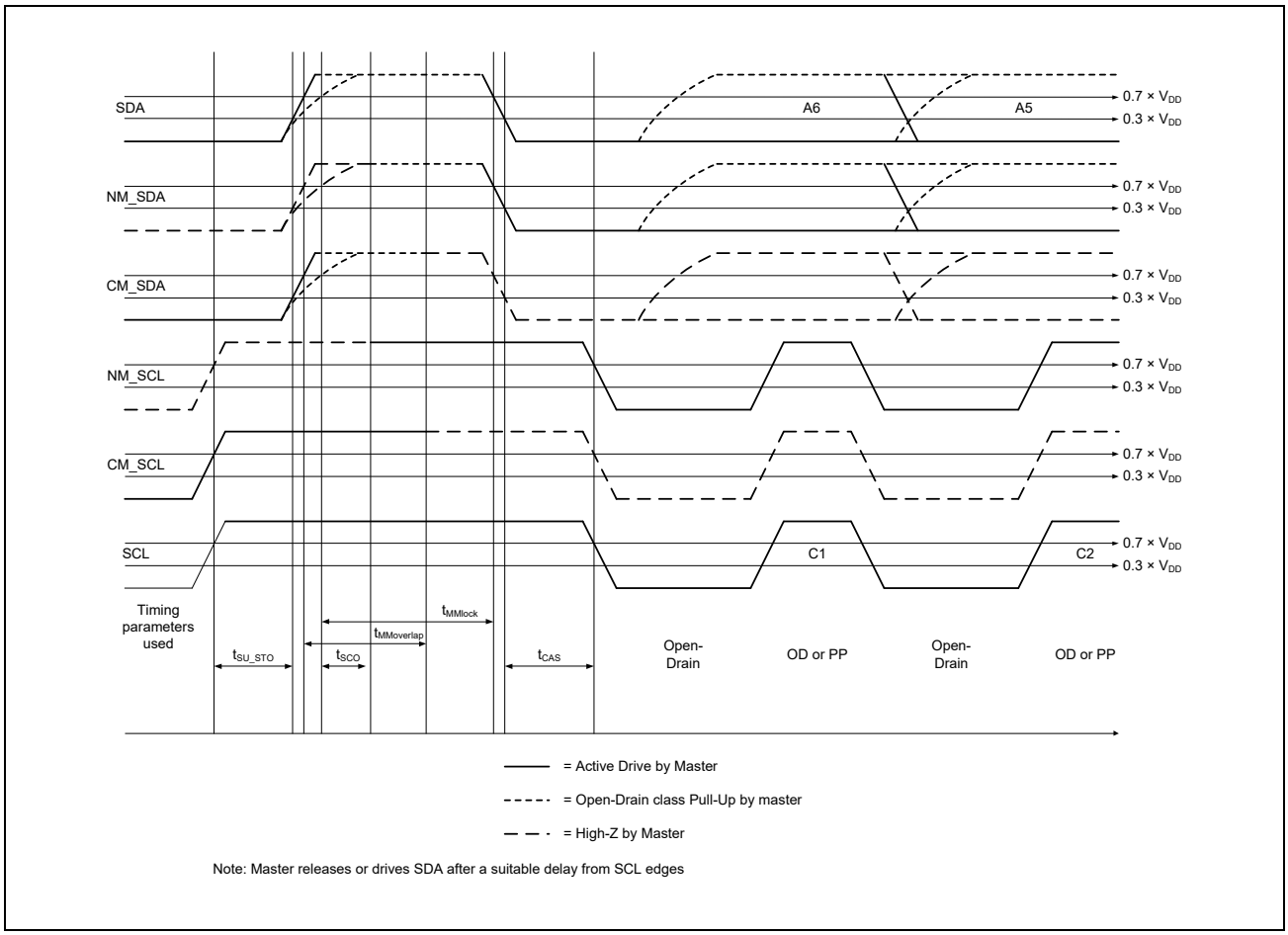


Figure 47.53 I3C Timing

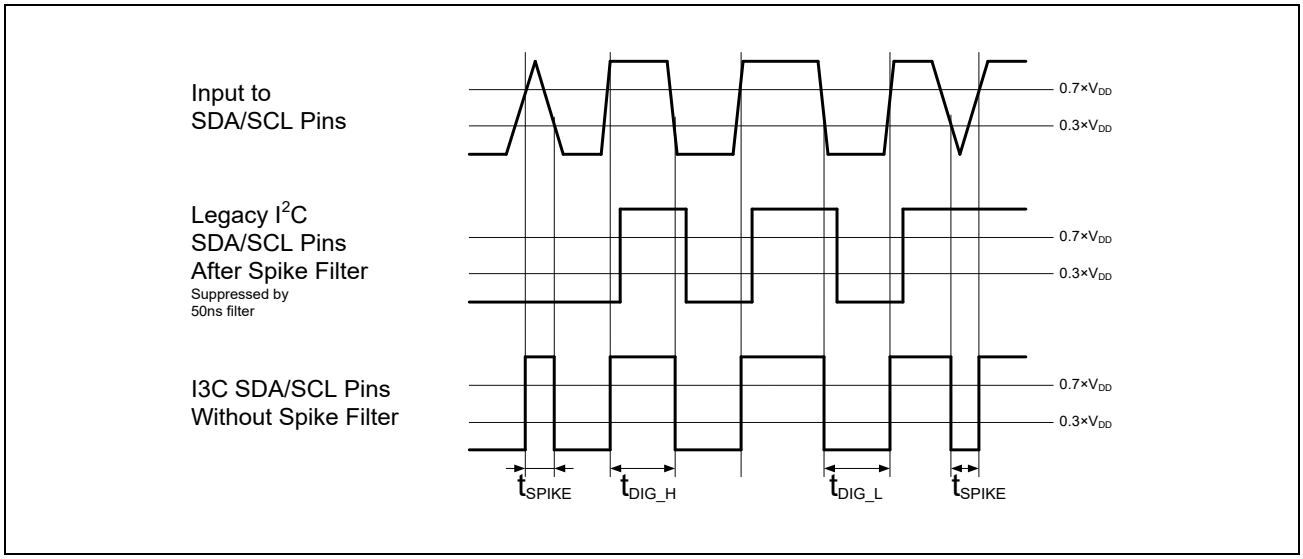


Figure 47.54 I2C Spike Filter Behavior

47.5.17 Serial Communications Interface with FIFO (SCIFA) Access Timing

Table 47.38 SCIFA Timing

Item			Symbol	Min.	Max.	Unit	Figures
SCIFA	Input clock cycle	Asynchronous	t_{Scyc}	4	—	t_{p1cyc}*1	**Figure 47.55**
		Clocked synchronous		12	—		
	Input clock pulse width		t_{SCKW}	0.4	0.6	t_{p1cyc}*1	
	Input clock rise time		t_{SCKr}	—	5	ns	
	Input clock fall time		t_{SCKf}	—	5	ns	
	Output clock cycle	Asynchronous*2	t_{Scyc}	8	—	t_{p1cyc}*1	
		Clocked synchronous		4	—		
	Output clock pulse width		t_{SCKW}	0.4	0.6	t_{p1cyc}*1	
	Output clock rise time		t_{SCKr}	—	9	ns	
	Output clock fall time		t_{SCKf}	—	9	ns	
	Transmit data delay time	Internal clock	t_{TXD}	−10	10	ns	**Figure 47.56**
		External clock		3 × t_{p1cyc}*1	4 × t_{p1cyc}*1 + 20		
	Receive data setup time	Internal clock	t_{RXS}	3 × t_{p1cyc}*1 + 20	—	ns	
		External clock		t_{p1cyc}*1 + 10	—		
	Receive data hold time	Internal clock	t_{RXH}	−3 × t_{p1cyc}*1	—	ns	
		External clock		2 × t_{p1cyc}*1 + 10	—		

Note: AC access timing condition: drive ability 12 mA, output load 30 pF, slew

Note 1. t_{p1cyc} indicates peripheral clock means SCIFn_CLK_PCK (P0ϕ) (n = 0 to 5).

Note 2. When the SEMR.ABCS0 and SEMR.BGDM bits are set to 1.

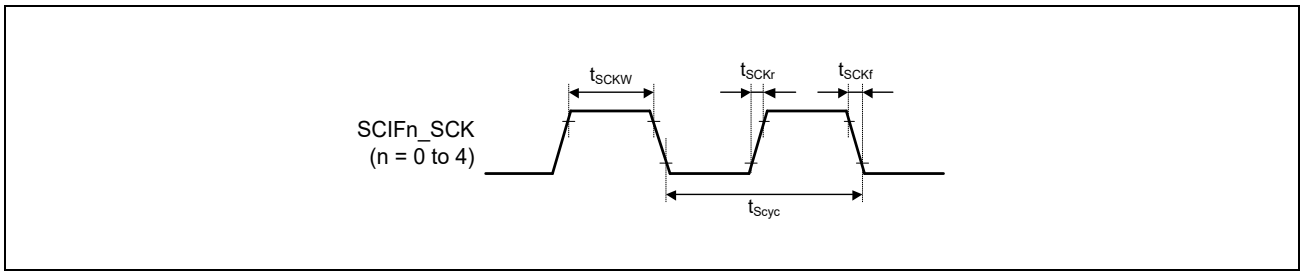
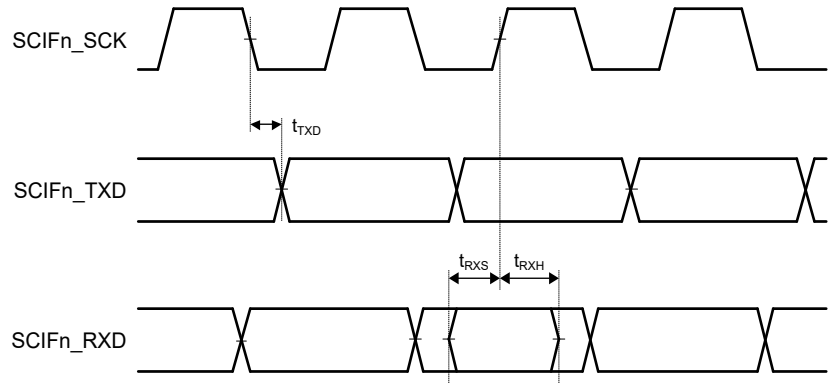


Figure 47.55 SCK Input Clock Timing



Note: n = 0 to 4

Figure 47.56 SCIFA Input/Output Timing in Clocked Synchronous Mode

47.5.18 Serial Communications Interface (SCIg) Access Timing

Table 47.39 SCIg Timing

Item			Symbol	Min.	Max.	Unit	Figures
SCIg	Input clock cycle	Asynchronous	t_{Scyc}	4	—	t_{p1cyc}*1	**Figure 47.57**
		Clocked synchronous		6	—		
	Input clock pulse width		t_{SCKW}	0.4	0.6	t_{p1cyc}*1	
	Input clock rise time		t_{SCKr}	—	5	ns	
	Input clock fall time		t_{SCKf}	—	5	ns	
	Output clock cycle	Asynchronous*2	t_{Scyc}	8	—	t_{p1cyc}*1	
		Clocked synchronous		4	—		
	Output clock pulse width		t_{SCKW}	0.4	0.6	t_{p1cyc}*1	
	Output clock rise time		t_{SCKr}	—	5	ns	
	Output clock fall time		t_{SCKf}	—	5	ns	
	Transmit data delay time	Clocked synchronous	t_{TXD}	—	28	ns	**Figure 47.58**
	Receive data setup time	Clocked synchronous	t_{RXS}	15	—	ns	
	Receive data hold time	Clocked synchronous	t_{RXH}	5	—	ns	

Note: AC access timing condition: drive ability 12 mA, output load 30 pF

Note 1. t_{p1cyc} indicates peripheral clock means SCIn_CLKP (P0ϕ) (n = 0 to 1).

Note 2. When the SEMR.ABCS0 and SEMR.BGDM bits are set to 1.

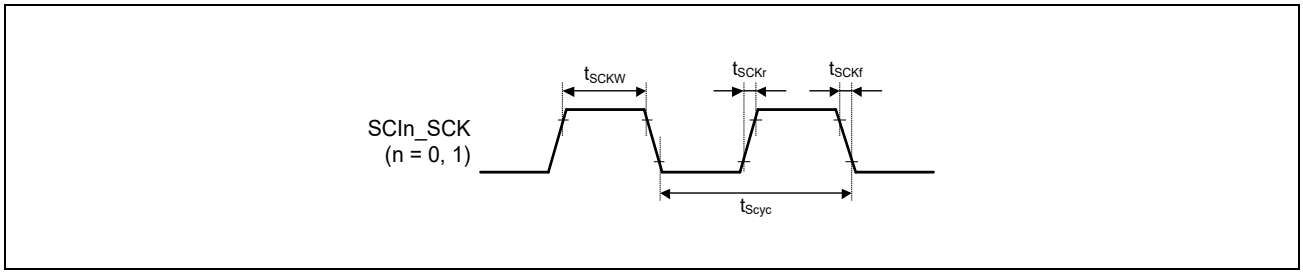
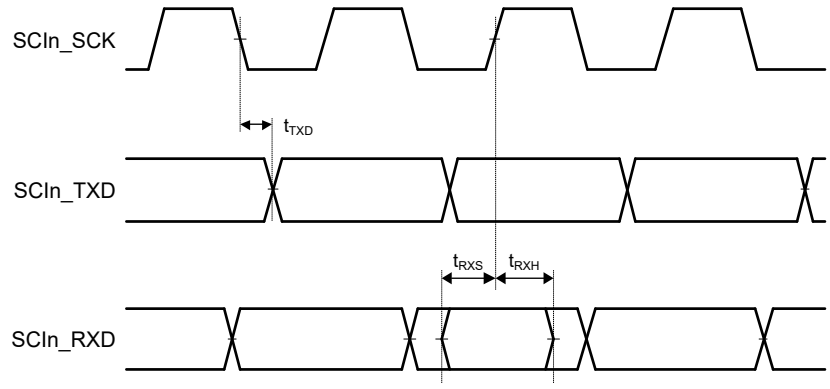


Figure 47.57 SCK Input Clock Timing



Note: n = 0, 1

Figure 47.58 SCIFA Input/Output Timing in Clocked Synchronous Mode

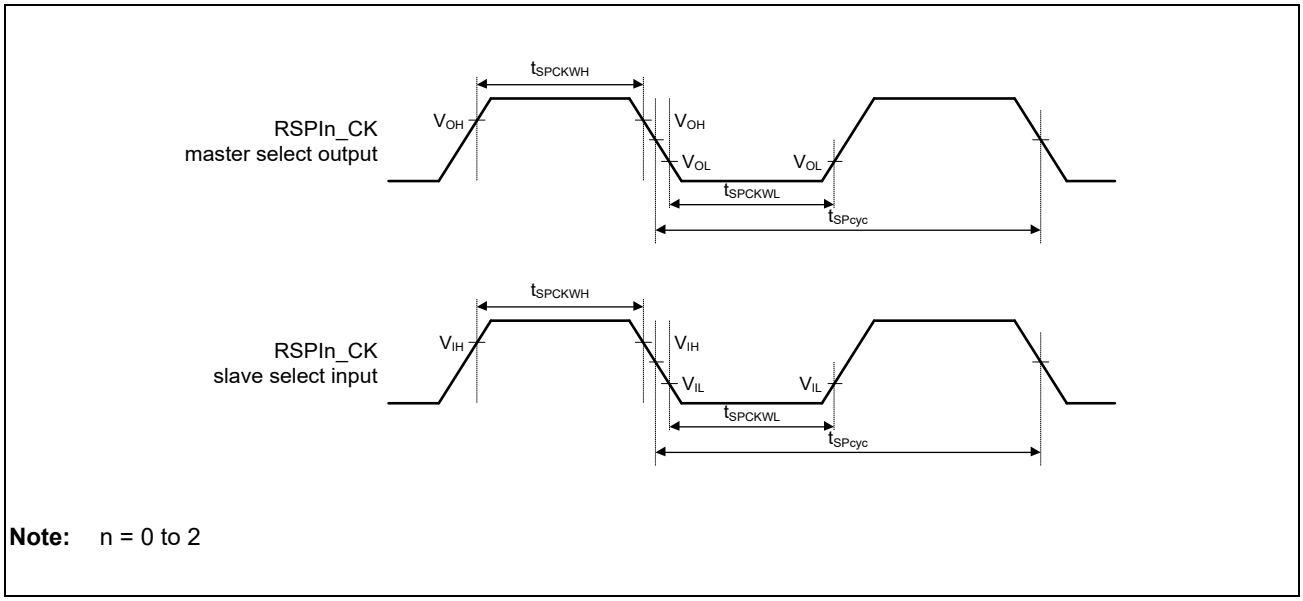
47.5.19 Renesas Serial Peripheral Interface (RSPI) Access Timing

Table 47.40 Renesas Serial Peripheral Interface Timing

Item		Symbol	Min.	Max.	Unit	Figure
RSPCK clock cycle	Master	t_{SPcyc}	2	4096	t_{p1cyc}*1	**Figure 47.59**
	Slave		8	4096		
RSPCK clock high pulse width	Master	t_{SPCKWH}	0.4	—	t_{SPcyc}*1	
	Slave		0.4	—		
RSPCK clock low pulse width	Master	t_{SPCKWL}	0.4	—	t_{SPcyc}*1	
	Slave		0.4	—		
Data input setup time	Master	t_{SU}	10	—	ns	**Figure 47.60** to **Figure 47.63**
	Slave		0	—	t_{p1cyc}*1	
Data input hold time	Master	t_H	0	—	ns	
	Slave		4	—	t_{p1cyc}*1	
SSL setup time	Master	t_{LEAD}	1 × t_{SPcyc} − 20	8 × t_{SPcyc}	ns	
	Slave		4	—	t_{p1cyc}*1	
SSL hold time	Master	t_{LAG}	1 × t_{SPcyc}	8 × t_{SPcyc} + 20	ns	
	Slave		4	—	t_{p1cyc}*1	
Data output delay time	Master	t_{OD}	—	19	ns	
	Slave		—	4	t_{p1cyc}*1	
Data output hold time	Master	t_{OH}	5	—	ns	
	Slave		2	—	t_{p1cyc}*1	
Continuous transmission delay time	Master	t_{TD}	1 × t_{SPcyc} + 2 × t_{cyc}	8 × t_{SPcyc} + 2 × t_{cyc}	ns	
	Slave		4 × t_{cyc}	—		
Slave access time		t_{SA}	—	4	t_{p1cyc}*1	**Figure 47.62**, **Figure 47.63**
Slave out release time		t_{REL}	—	3	t_{p1cyc}*1	

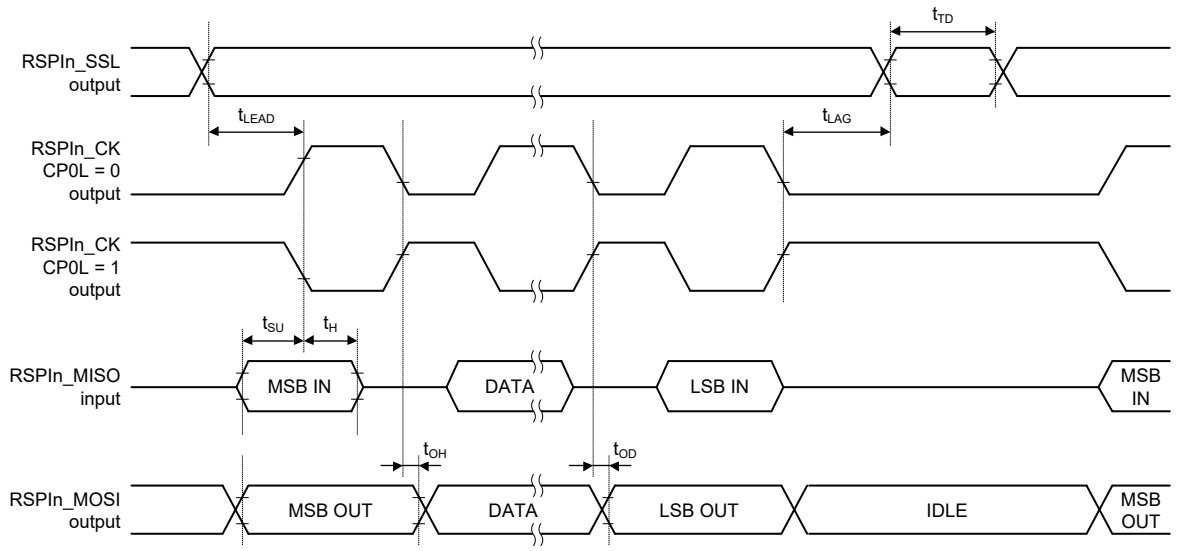
Note: AC access timing condition: drive ability 12 mA, output load 30 pF

Note 1. t_{p1cyc} indicates peripheral clock means RSPIn_CLKB (P0ϕ) (0 to 4).



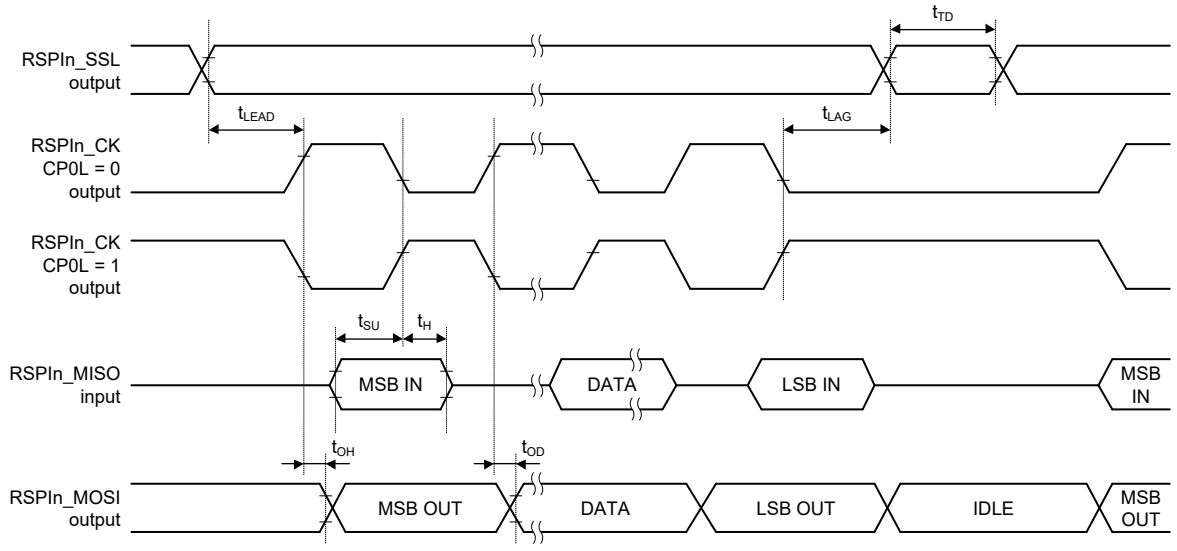
Note: n = 0 to 2

Figure 47.59 Clock Timing



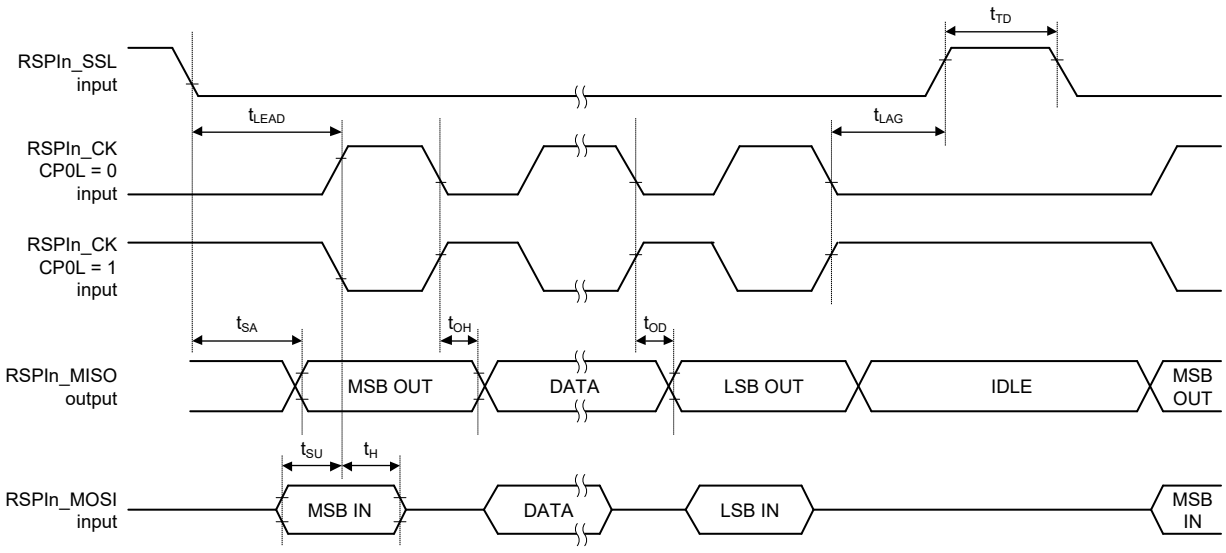
Note: n = 0 to 2

Figure 47.60 Transmission and Reception Timing (Master, CPHA = 0)



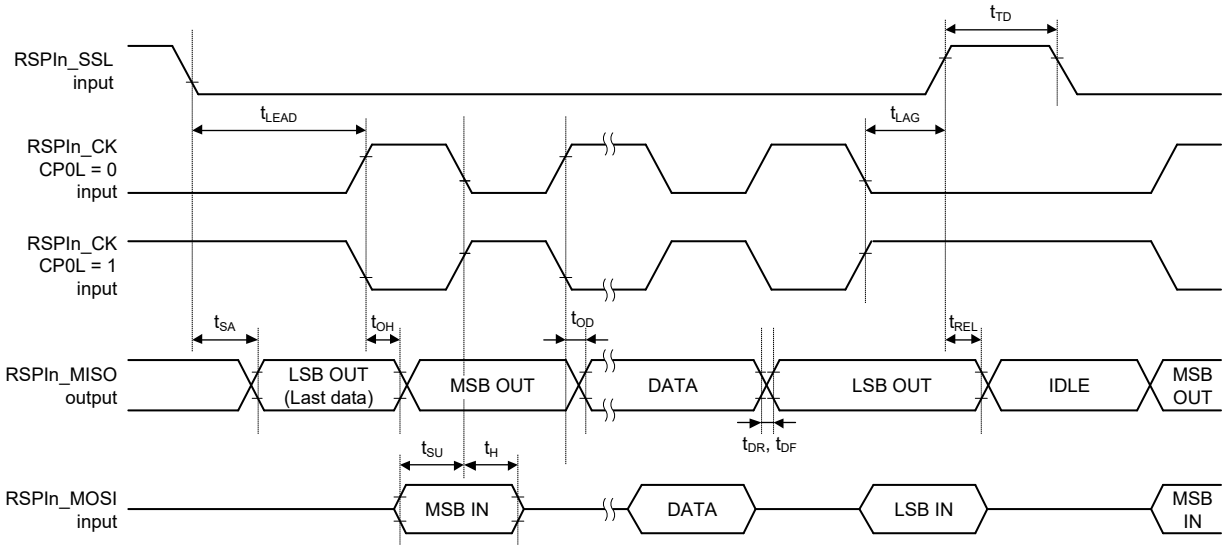
Note: n = 0 to 2

Figure 47.61 Transmission and Reception Timing (Master, CPHA = 1)



Note: n = 0 to 2

Figure 47.62 Transmission and Reception Timing (Slave, CPHA = 0)



Note: n = 0 to 2

Figure 47.63 Transmission and Reception Timing (Slave, CPHA = 1)

47.5.20 A/D Converter Access Timing

Table 47.41 A/D Converter Timing

Item	Symbol	Min.	Max.	Unit	Figures
ADC Trigger Input Pulse Width	t_{TRGW}	1.5*2		t_{P1cyc}*1	**Figure 47.64**

Note 1. t_{p1cyc} indicates peripheral clock means ADC_ADCLK (TSUϕ).

Note 2. When a noise filter in ADC is off.

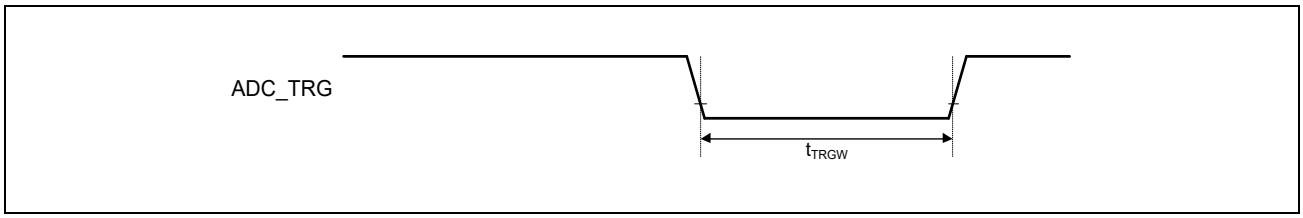


Figure 47.64 ADC Trigger Input Timing

47.5.21 Watchdog Timer Access Timing

Table 47.42 Watchdog Timer Timing

Item	Symbol	Min.	Max.	Unit	Figures
WDTOVF_PERROUT# Output Time	t_L	64	64	t_{P1cyc}*1	**Figure 47.65**

Note 1. t_{P1cyc} indicates peripheral clock means WDTn_CLK (OSCCLK) (n = 0 to 2).

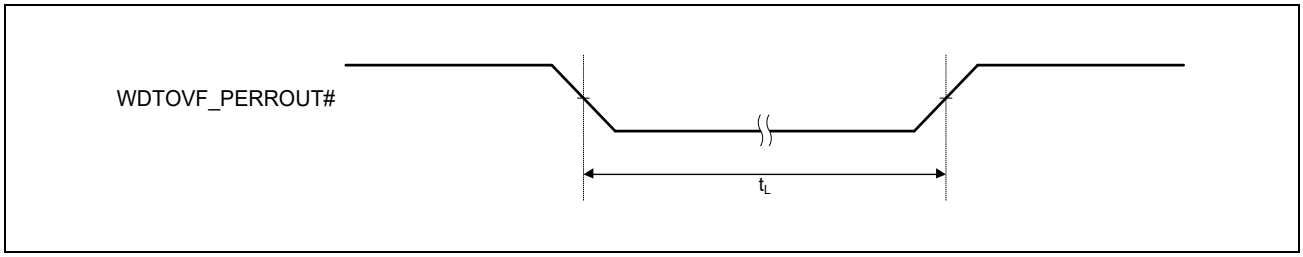


Figure 47.65 Watchdog Timer Output Timing

47.5.22 PDM Interface Access Timing

Table 47.43 PDM Interface Timing

Item	Symbol	Min.	Max.	Unit	Remarks
Clock period	t_{PSYNC}	2	32	t_{CCcyc} = 125 ns (8 MHz)	
Clock High level period	t_{PDCKWH}	125	2000	ns	
Clock Low level period	t_{PDCKWL}	125	2000	ns	

Note: I/O driving ability: 4 mA, CL = 30 pF

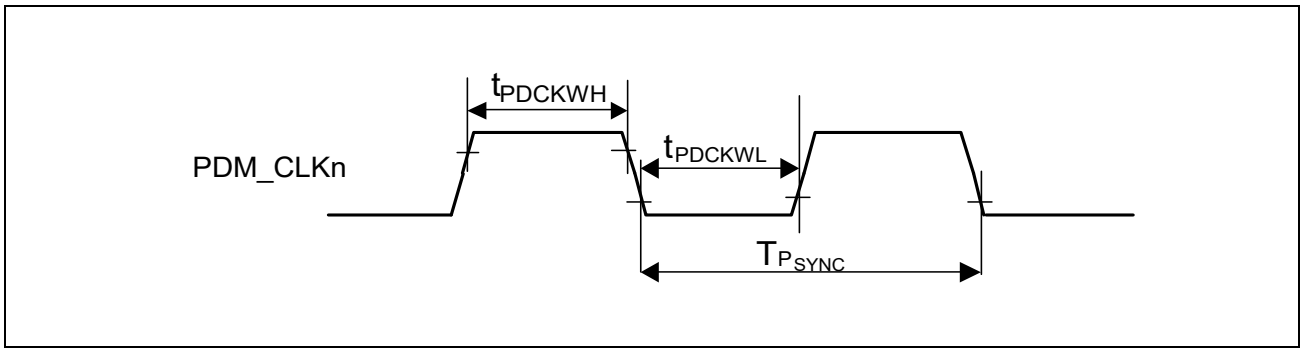


Figure 47.66 Timing of Clock Output (PDMn_CLK)

Table 47.44 PDM Interface Timing

Item	Symbol	Min.	Max.	Unit	Remarks
PDMn_DAT input setup time	t_{SU}	12.5	—	ns	
PDMn_DAT input hold time	t_H	0.0	—	ns	

Note: I/O driving ability: 4 mA, CL = 30 pF

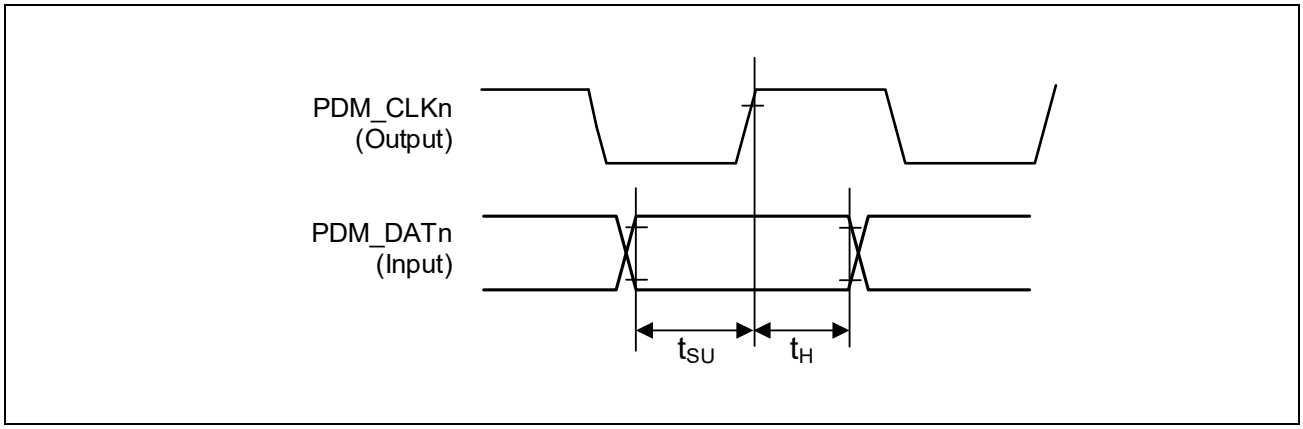


Figure 47.67 Receive Timing (Synchronized with the Rise of PDMn_CLK)

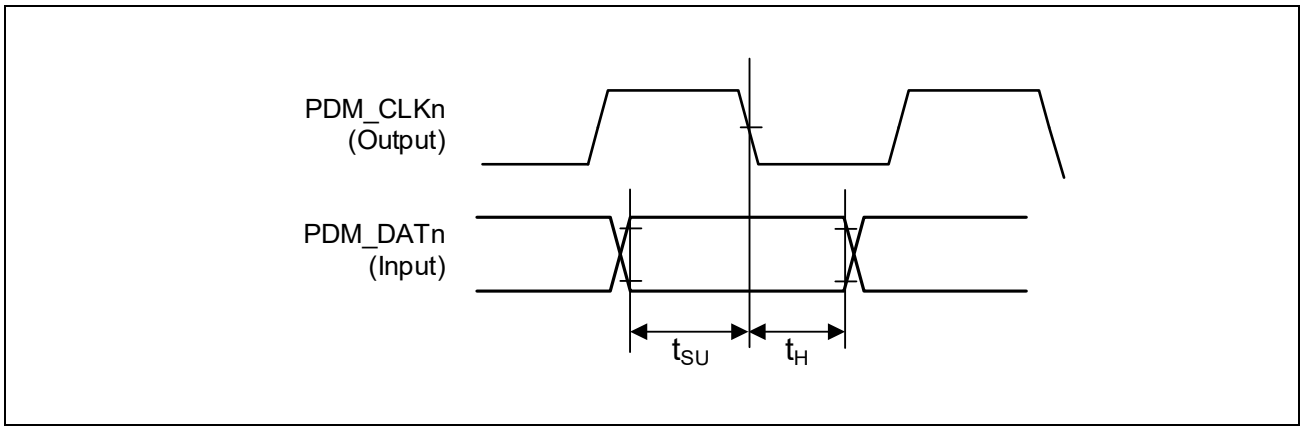


Figure 47.68 Receive Timing (Synchronized with the Fall of PDMn_CLK)

47.5.23 Octa Memory Controller Timing

Table 47.45 Octa Memory Controller Timing*1,*2

Item		Symbol	Min.	Max.	Unit	Figures
QSPI0_SPCLK clock frequency		f_{OCcyc}	—	100	MHz	**Figure 47.69**
QSPI0_SPCLK high pulse width		t_{OCwh}	0.45	0.55	t_{OCcyc}	
QSPI0_SPCLK low pulse width		t_{OCwl}	0.45	0.55	t_{OCcyc}	
QSPI0_SPCLK rise time		t_{OCr}	—	1	ns	
QSPI0_SPCLK fall time		t_{OCf}	—	1	ns	
OM_CS1#, QSPI0_SSL setup time	SPI/SOPI	t_{OCLEAD}	$1 \times t_{OCcyc} - 3$ (Minimum register settings)	$2.5 \times t_{OCcyc} + 3$ (Maximum register settings)	ns	**Figure 47.70**, **Figure 47.71**
	DOPI	t_{OCLEAD}	$0.75 \times t_{OCcyc} - 3$ (Minimum register settings)	$2.25 \times t_{OCcyc} + 3$ (Maximum register settings)	ns	**Figure 47.72**
OM_CS1#, QSPI0_SSL hold time	SPI/SOPI	t_{OCLAG}	$1 \times t_{OCcyc} - 3$ (Minimum register settings)	$4.5 \times t_{OCcyc} + 3$ (Maximum register settings)	ns	**Figure 47.70**, **Figure 47.71**
	DOPI read	t_{OCLAG}	$3.25 \times t_{OCcyc} - 3$ (Minimum register settings)	$4.25 \times t_{OCcyc} + 3$ (Maximum register settings)	ns	**Figure 47.72**
	DOPI write	t_{OCLAG}	$0.75 \times t_{OCcyc} - 3$ (Minimum register settings)	$4.25 \times t_{OCcyc} + 3$ (Maximum register settings)	ns	
Continuous transfer delay time		t_{OCTD}	$1 \times t_{OCcyc} - 3$ (Minimum register settings)	$8.5 \times t_{OCcyc} + 3$ (Maximum register settings)	ns	**Figure 47.70**, **Figure 47.71**, **Figure 47.72**
Data input setup time	QSPI0_SPCLK base point	t_{SU}	8.5	—	ns	**Figure 47.70**
Data input hold time		t_H	0.5	—	ns	
Data input setup time	SOPI/DOPI DQS base point*3	t_{SU}	−0.7	—	ns	**Figure 47.71**, **Figure 47.72**
Data input hold time		t_H	2.775	—	ns	
Skew of Clock to Data Strobe		t_{CKDS}	—	20	ns	
Data output delay time	SPI/SOPI	t_{OD}	—	1.4	ns	**Figure 47.70**, **Figure 47.71**
Data output hold time		t_{OH}	−1.4	—	ns	
Data output buffer off time	SOPI	t_{BOFF}	2	—	ns	**Figure 47.71**
Data output delay time	DOPI*3	t_{OD}	2.775	—	ns	**Figure 47.72**, **Figure 47.73**
Data output hold time		t_{OH}	0.975	—	ns	
Data output buffer off time	DOPI	t_{BOFF}	0.9	—	ns	**Figure 47.72**
DQS refresh input setup time		t_{DQSS}	12	—	ns	**Figure 47.74**
DQS refresh input hold time		t_{DQSH}	$0.5 \times t_{OCcyc}$	—	ns	

Note 1. t_{OCcyc} indicates the QSPI0_SPCLK cycle.

Note 2. Maximum load capacitance: 15 pF

Note 3. QSPI0_SPCLK frequency: 100 MHz

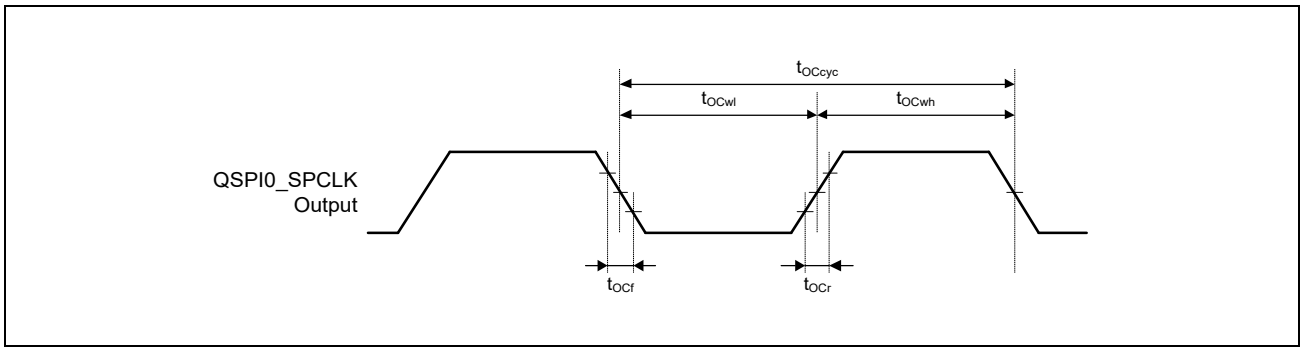


Figure 47.69 Clock Timing

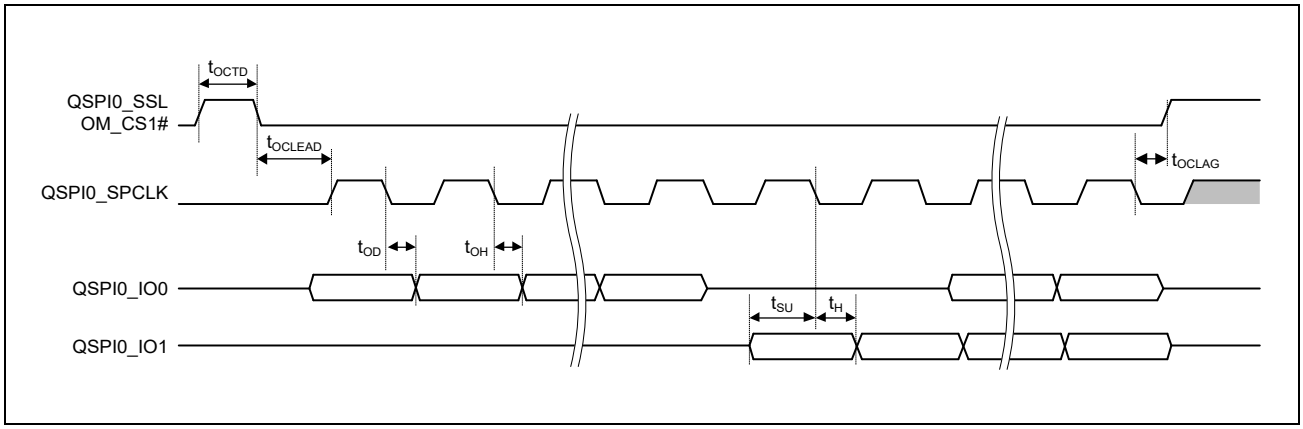


Figure 47.70 SPI Transferring Format Transmission and Receiving Timing

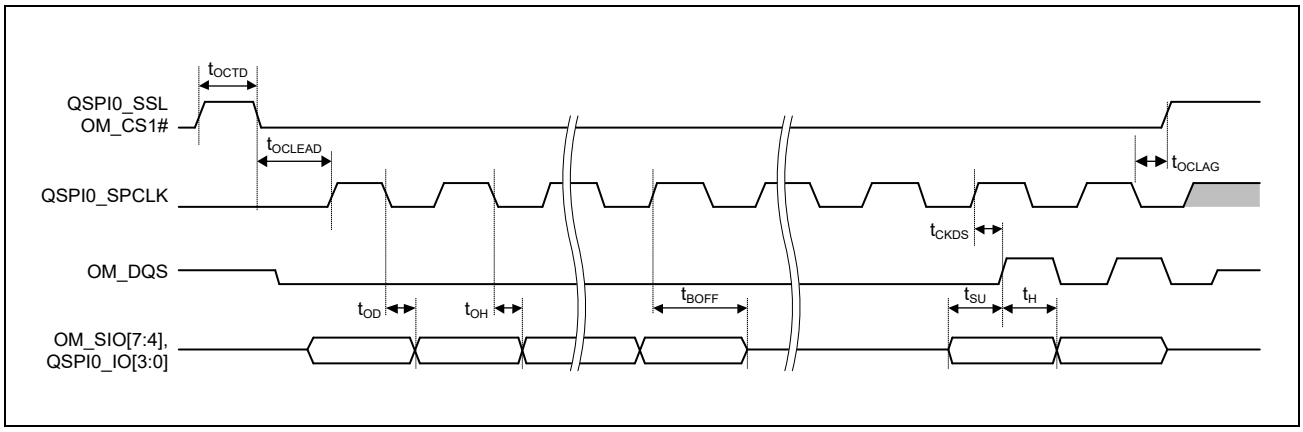


Figure 47.71 SOPI Transferring Format Transmission and Receiving Timing

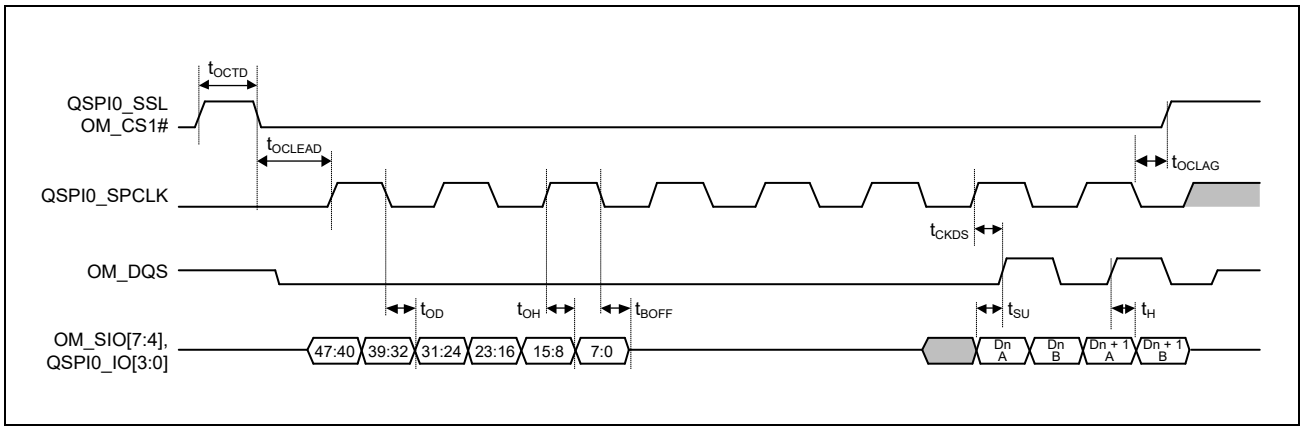


Figure 47.72 DOPI Transferring Format Transmission and Receiving Timing

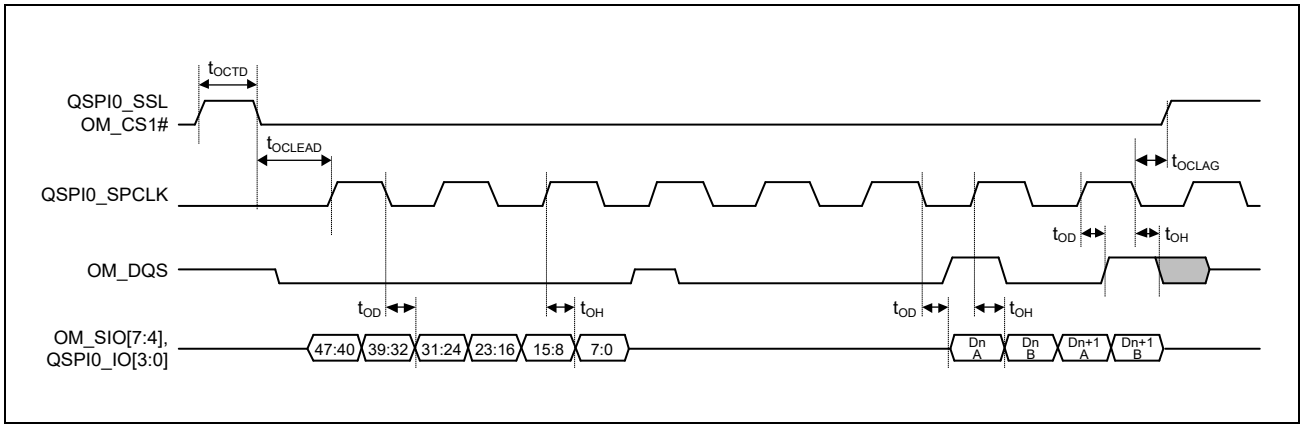


Figure 47.73 DOPI Transferring Format Transmission Timing

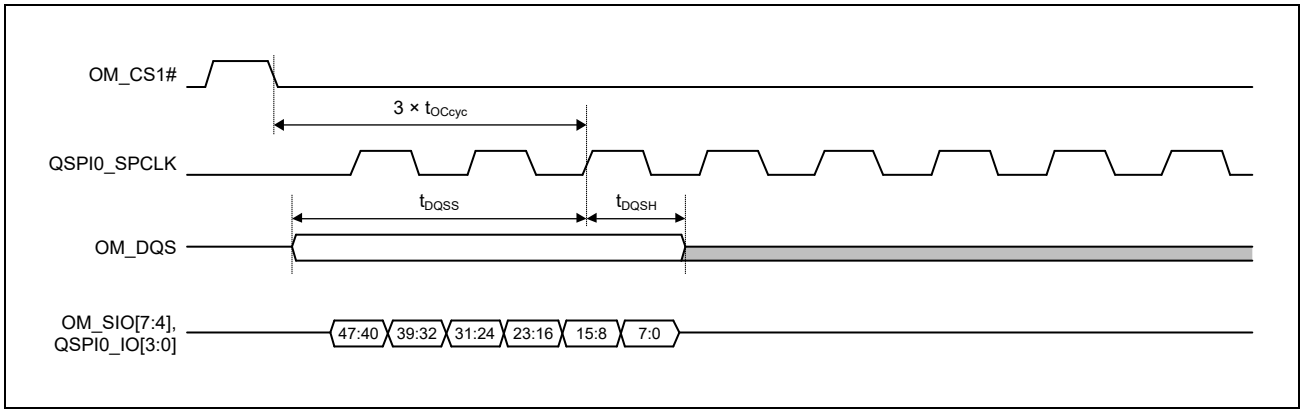


Figure 47.74 DQS Refresh Input Timing (OctaRAM Read and Write)

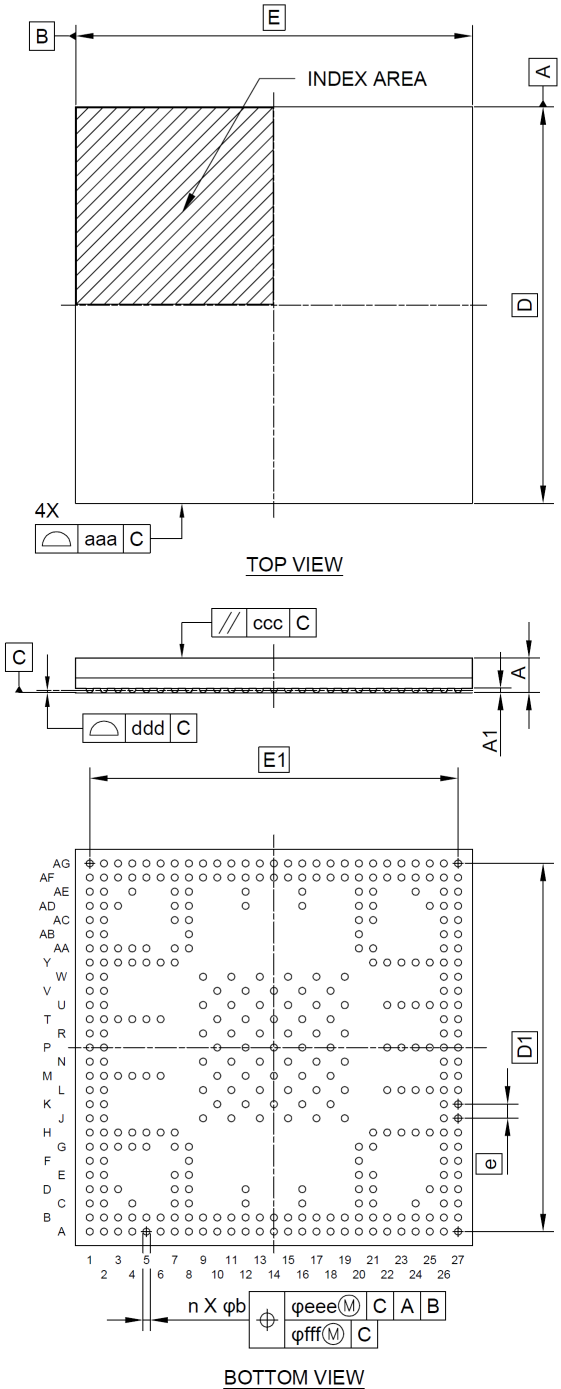
47.5.24 PCI Express PHY Characteristics

The PCI Express PHY of this LSI is compliant with the following PCIe standard:

Revision 4.0 of the PCI Express® Base Specification for Gen1/Gen 2/ Gen 3

Appendix A Package Dimensions

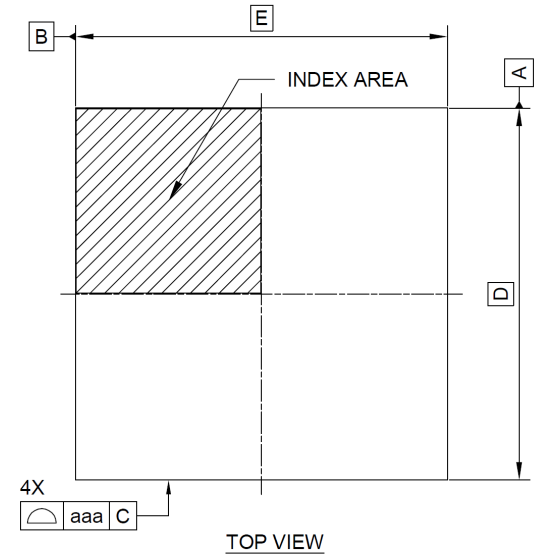
JEITA Package code	RENESAS code	MASS(TYP.)[g]
P-LFBGA359-14x14-0.50	PLBG0359KA-A	0.45



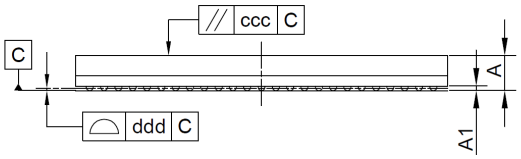
Reference Symbol	Dimension in Millimeters		
	Min.	Nom.	Max.
D	—	14.00	—
E	—	14.00	—
D1	—	13.00	—
E1	—	13.00	—
A	—	—	1.40
A1	0.11	—	—
b	0.20	0.25	0.30
e	—	0.50	—
aaa	—	—	0.15
ccc	—	—	0.20
ddd	—	—	0.08
eee	—	—	0.15
fff	—	—	0.08
n	—	359	—

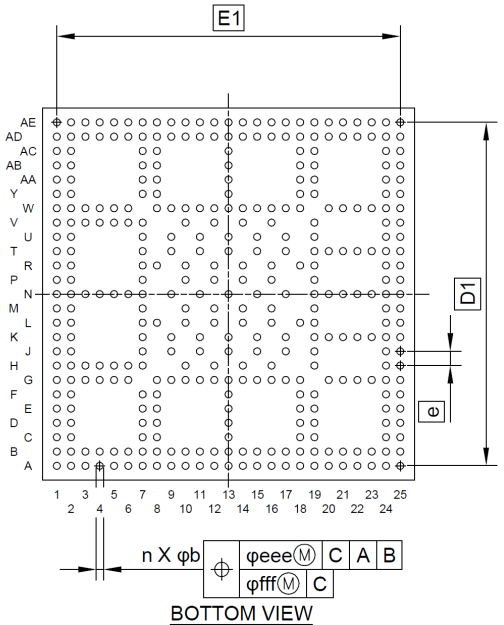
Appendix A.1 Package Dimensions (359Pin BGA)

JEITA Package code	RENESAS code	MASS(TYP.)[g]
P-LFBGA361-13x13-0.50	PLBG0361KD-A	0.40



TOP VIEW





BOTTOM VIEW

Reference Symbol	Dimension in Millimeters		
	Min.	Nom.	Max.
D	—	13.00	—
E	—	13.00	—
D1	—	12.00	—
E1	—	12.00	—
A	—	—	1.40
A1	0.11	—	—
b	0.20	0.25	0.30
e	—	0.50	—
aaa	—	—	0.15
ccc	—	—	0.20
ddd	—	—	0.08
eee	—	—	0.15
fff	—	—	0.08
n	—	361	—

Appendix A.2 Package Dimensions (361Pin BGA)

REVISION HISTORY	RZ/G3S Group User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
1.00	Aug 24, 2023	—	First edition issued
1.10	Nov 30, 2023	1. Overview	
		82	1.2.12 Power Supply Voltage: The value of VBATT_VDD, changed (1.65 to 1.95 V → 1.50 to 1.95 V)
		83	Table 1.1 Product Lineup: The CPU items, added; four products (R9A08G045S17GBG, R9A08G045S13GBG, R9A08G045S15GBG, R9A08G045S11GBG), added
		7. Clock Pulse Generator (CPG)	
		269	Table 7.5 Register list: Note 1, modified
		9. DDR4/LPDDR4 SDRAM Memory Controller (MEMC)	
		658	Table 9.1 DDR4/LPDDR4 SDRAM Memory Controller (MEMC) Features: The density of the DRAM IF, changed
		26. I3C Bus Interface (I3C)	
		1651	26. I3C Bus Interface (I3C): NOTE, added
		1651	Table 26.1 I2C Specifications: The entries related to Hs-mode for "Communication protocol" and "Address detection", deleted
		41. Low Power Consumption	
		3593	Figure 41.3 Power Mode Transition: A transition from AWO to VBATT, deleted; Note, deleted
		—	41.6.5 AWO to VBATT, deleted
		47. Electrical Characteristics	
		3710	Table 47.2 Power Supply: The min. value of VBATT_VDD, changed (1.65 V → 1.50 V)
		—	Figure 47.5 Power-On/Power-Off Sequence 4 (AWO to VBATT + DDR Retention), deleted
		—	Figure 47.6 Power-On/Power-Off Sequence 4 (AWO to VBATT), deleted
		3719	47.5 AC Characteristics: The value of VBATT_VDD, changed (1.65 to 1.95 V → 1.50 to 1.95 V)
		3747	Table 47.35 Characteristics of the SDA and SCL Bus Lines for Standard, Fast, and Fast-Mode Plus I^2C-Bus Devices: Note 6, deleted

RZ/G3S Group User's Manual: Hardware

Publication Date: Rev.1.00 Aug 24, 2023
Rev.1.10 Nov 30, 2023

Published by: Renesas Electronics Corporation

RZ/G3S Group

Renesas Electronics Corporation

R01UH1014EJ0110