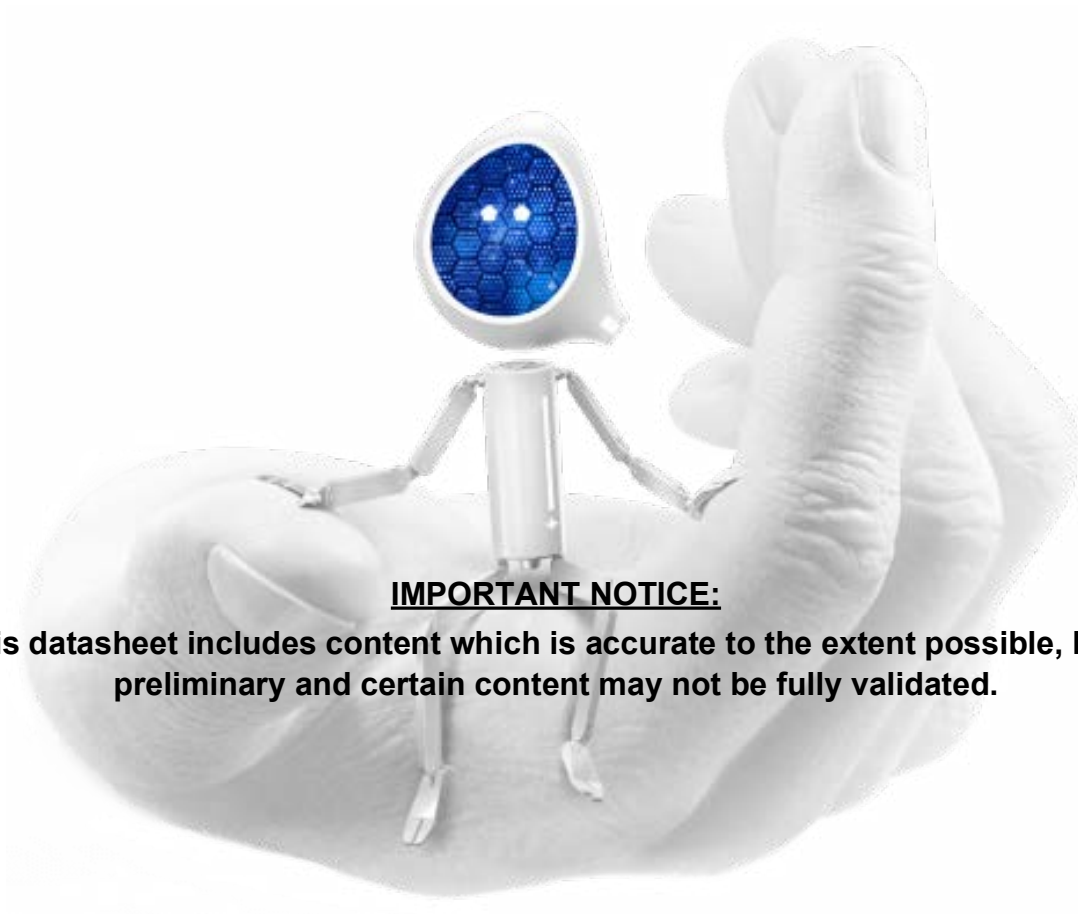**DATASHEET - Preliminary and subject to change**

# Apollo510 SoC

Ultra-low Power Apollo5 SoC Family

Doc. ID: DS-A510-0p9p2

Doc. Revision: 0.9.2, August 2025

<u>IMPORTANT NOTICE:</u>

This datasheet includes content which is accurate to the extent possible, but is preliminary and certain content may not be fully validated.

# Features

## Ultra-Low Supply Current

- 4.5 CM/MHz performance
- > 120 CM/mJ energy efficiency

## Arm® Cortex®-M55 Processor with Helium™ Technology

- Up to 250 MHz clock frequency
- Helium (MVE) vector integer, floating point
- Scalar half, single, and double-precision floating-point
- Supports Arm® TrustZone® security extensions
- Integrated 64 kB Instruction Cache and 64 kB Data Cache
- Integrated 768 kB Instr./Data Tightly Coupled Memory (TCM)
- Memory Protection Unit (MPU)

## secureSPOT 3.0 Security Features

- Arm TrustZone® technology
- Secure boot
- OTP key storage
- PUF-based identity/sign/verify
- Secure over-the-air (OTA) updates
- Key revocation

## Ultra-Low-Power Memory

- Up to 4 MB of non-volatile memory for code/data
- 3.75 MB of TCM and system RAM for code/data

## Ultra-Low-Power Interface for On- and Off-Chip Sensors

- 12-bit ADC, 11 selectable input channels
- Up to 1.7 MS/s sampling rate
- Integrated temperature sensor

## Ultra-Low-Power Flexible Serial Peripherals

- 2x 2/4/8-bit SPI manager interfaces
- 2x 2/4/8/16-bit SPI manager interface supporting 1.2 V
- 8x I²C/SPI managers for peripheral communication
- I²C/SPI subordinate for host communications
- 4x UART modules with FIFOs and flow control
- 1x USB 2.0 HS/FS device controller
- 2x SDIO (SD3.0) / eMMC (v4.51)

## Display

- MIPI DSI 1.2 up to 768 Mbps
- QuadSPI display interface (up to 125 MHz DDR)
- Up to 640 x 480 resolution at 60 FPS
- 4 layers with alpha blending
- Frame buffer decompression

## graphiqSPOT 2.0 Graphics Features

- 2D/2.5D GPU with vector graphics (VG) acceleration
- 96 MHz / 250 MHz operating modes
- Anti-aliasing hardware acceleration
- Rasterizer / full alpha blending / texture mapping
- Texture / frame buffer compression (TSC4, 6, 6A and 12)
- Dithering and radial/conical fill support

## Audio Processing

- 1x Low power audio ADC with PGA
- 1x PDM stereo DMIC interface
- 2x full-duplex multichannel I²S port (1x with ASRC)

## Rich Set of Clock Sources

- PLL for precise clocking applications
- 48 MHz and 32.768 kHz Crystal (XTAL) oscillators
- Low Frequency RC (LFRC) oscillator
- High Frequency RC (HFRC) oscillator

## Power Management

- Operating range: 1.71 - 2.2 V
- Single Inductor Multiple Outputs (SIMO) Buck Converter
- Multiple I/O voltages supported

# Applications

- Smartwatches/bands
- Smart home devices
- Body-worn and ambient AI
- Wireless sensors and industrial edge
- Smart remotes
- Patient health monitoring
- Hearing assist
- Condition monitoring
- Factory predictive maintenance
- Livestock monitoring
- Asset tracking

# Package Options

- 6.6 x 6.6 mm BGA with 183 GPIO
- 4.9 x 4.7 mm WLCSP with 144 GPIO

# Ordering Information

- -20°C to +70°C:
  - AP510NFA-CBR (BGA)
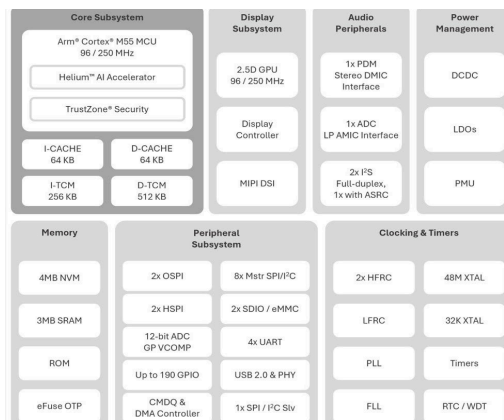  - AP510NFA-CCR (WLCSP)

**Table of Contents**

## List of Figures

## List of Tables

# 1. SoC Product Introduction



**Figure 1. Apollo510 SoC High-Level Block Diagram**

## 1.1 Features

The Apollo510 SoC is an ultra-low power, highly integrated, mixed-signal SoC designed for battery-powered devices. It provides a significant enhancement in processing capability and highly integrated power management and audio capabilities to the Apollo SoC product family. The Apollo510 SoC brings the powerful Arm® Cortex-M55 processor with the Extension Processing Unit which supports M-Profile Vector Extension (MVE) processing and floating point operations, coupled with the world's lowest power audio and communications processing. The SoC takes the Ambiq Micro's patented Subthreshold Power Optimized Technology (SPOT) platform to a whole new level of compute power efficiency, setting new industry benchmarks in low power design and high efficiency portable computing.

> **NOTE**
>
> Any references to the Apollo5 Family herein apply to all existing derivatives of the Apollo5 SoC family. If a feature or function described herein applies to a specific derivative and/or does not apply to all derivatives, the exception(s) will be noted in the applicable context.

The Apollo510 SoC includes features shown in Figure 1 and listed below.

**Arm® Cortex®-M55 Application Processor with Helium™ technology:**

- 96 MHz / 192 MHz / 250 MHz operating modes
- Helium (MVE) AI accelerator, up to 8 MACs per cycle
- Supports Arm® TrustZone®[1] security extensions
- Integer, half-precision and single-precision floating-point M-profile Vector Extensions (MVE) supported
- Integrated 64 kB Instruction Cache and 64 kB Data Cache
- Integrated 256 kB Instruction Tightly Coupled Memory (ITCM) and 512 kB Data TCM (DTCM)
- Memory Protection Unit (MPU)
- Wake-up Interrupt Controller (WIC) with 135 interrupts

**secureSPOT 3.0 Security Features:**

- Arm TrustZone® technology
- Secure boot
- OTP key storage
- PUF-based identity/sign/verify
- Secure over-the-air (OTA) updates
- Key revocation

**Ultra-Low-Power Memory:**

- Up to 4 MB of non-volatile memory for code/data
- Up to 3.75 MB of low-leakage system RAM for code/data

**Ultra-Low-Power Interface for On- and Off-Chip Sensors:**

- 12-bit ADC with 11 selectable input channels
  - 10-bit ENOB
  - Up to 2.0 MS/s (12-bit Mode) / 2.8 MS/s (8-bit Mode) effective continuous, multi-slot sampling rate
  - Integrated temperature sensor
- Voltage comparator

**Ultra-Low-Power Flexible Serial Peripherals:**

- 8x I²C/SPI managers for peripheral communication
- 2x 2/4/8-bit SPI manager interface up to 96 MT/s
- 2x 2/4/8/16-bit SPI manager interface up to 250 MT/s

---

1. Link to the TF-M documentation: https://www.trustedfirmware.org/projects/tf-m.

- 1x Half Duplex I$^2$C or SPI subordinate and 1x Full Duplex or Half Duplex SPI subordinate both with DMA support for host communication
- 4x UART modules with 32-location Tx and Rx FIFOs with flow control and DMA support
- 2x SDIO controllers allowing concurrent SDIO (v3.0) / eMMC (v4.51) interface with DMA
- USB 2.0 HS/FS device controller with DMA

**Display:**

- MIPI DSI 1.2 with two 768 Mbps data lanes (up to 1.536 Gbps total bandwidth)
- QuadSPI display interface supporting up to 125 MT/s
- Up to 1920 x 1080 resolution[1]
- 4 layer with alpha blending
- Frame buffer decompression
- 8080 display support
- 6-bit parallel interface for color Memory-in-Pixel with Fast-Forward support
- LCD controller interface

**graphiqSPOT 2.0 Graphics Features:**

- 2D/2.5D GPU with enhanced Vector Graphics (VG) acceleration
- 96 MHz / 250 MHz operating modes
- Anti-aliasing hardware acceleration
- Rasterizer
- Full alpha blending
- Texture mapping
- Texture and frame buffer compression:
  - 4-bit
  - 6-bit (with/without Alpha)
  - 12-bit (with/without Alpha)
- Dithering support
- Radial/conical fill

**Audio / Communication Processing:**

- Support for one LP analog microphone using two PGA channels (low gain and high gain)
- 2x full-duplex multichannel I$^2$S ports - 1x with Asynchronous Sample Rate Converter (ASRC)
- Ultra low power voice and keyword detect
- PDM interface with support for 1 set of stereo high precision/low power digital microphones

**Rich set of clock sources:**

- High precision / low jitter PLL for high-fidelity audio and high-speed USB
- High frequency XTAL oscillator (19.2 - 52MHz)
  - Integrated within CSEM IP
- 32.768 kHz XTAL oscillator
- 2x high frequency RC oscillators – 192/250 MHz
- Nominal 900 Hz Low Frequency RC (LFRC) oscillator

---

1. Ignoring frame rate, the Display Controller can support up to 1920 x 1080 resolution. The resulting frame rate at any particular resolution depends upon display interface speed, bus-fabric bandwidth and complexity of the graphics assets and related operations. Typically a resolution of 640 x 480 at 60 frames/second can be supported for most applications.

▪ RTC based on Ambiq's AM08X5/18X5 families

**Power Management:**

▪ Operating ranges:
  - Voltage: 1.71 V - 2.2 V
  - Temp Range[1]: -20°C to 70°C
▪ Single Inductor Multiple Outputs (SIMO) Buck Converter
▪ Multiple I/O voltage domains supported (independent voltage rails)

**Package(s):**

▪ 6.6 mm x 6.6 mm, 16 x 16 BGA (225 pins / 183 GPIO)
▪ 4.913 mm x 4.676 mm, 14 x 13 WLCSP (182 pins / 144 GPIO)

---

1. See Ordering Information section for other available temp/package options.

## 1.2 Functional Overview
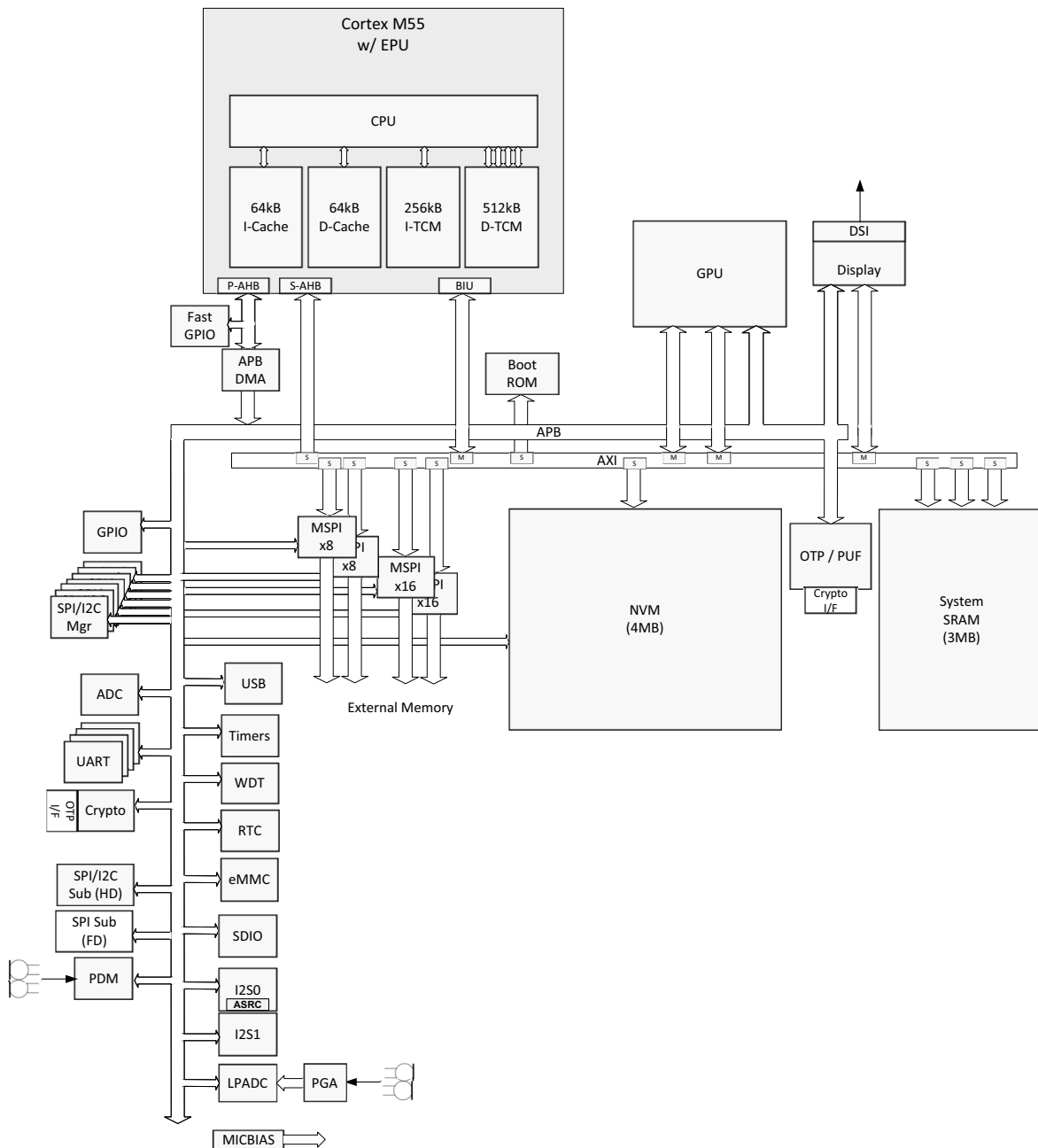


**Figure 2. Apollo510 SoC Detailed Block Diagram**

The ultra-low power Apollo510 SoC, shown in Figure 2, provides an ideal solution for battery-powered applications supporting near- to far-field audio processing. In a typical system, the SoC serves as an applications processor with fully integrated audio subsystem. The Apollo510 SoC includes an extensive set of digital and analog peripheral interfaces with integrated ADCs and digital sensor processing using the integrated serial manager ports. The Cortex-M55 core with Helium[TM] technology, also known as the M-Profile Vector Extension (MVE), integrated in the Apollo510 SoC is capable of running complex data analysis and sensor fusion algorithms to process the sensor data and orchestrate complex audio

processing signal flows. The Cortex-M55 core leverages a broad development and support ecosystem to accelerate time-to-market for application and product deployment.

In other configurations, a host processor can communicate or share data with the Apollo510 SoC over its serial subordinate port using its SPI interface or by using the I$^2$S audio streaming interface module, both of which support full duplex data transfer. With unprecedented energy efficiency for sensor conversion, audio processing and data analysis, the SoC enables months and years of battery life for products only achieving days or months of battery life today. Similarly the device enables the use of significantly complex algorithmic processing due to its industry leading low active mode power. By using the Apollo510 SoC, artificial intelligence at the portable edge is truly brought to life.

The Apollo510 SoC supports various operating modes to maximize energy efficiency depending on the workload demand. For extremely power-sensitive workloads, it supports a low power operating mode leveraging Ambiq Micro's patented SPOT technology to achieve industry leading energy efficiency. For timing critical or higher MIPs workloads, the device supports high performance operating modes through Ambiq Micro's TurboSPOT technology. The TurboSPOT technology enables high performance while still maintaining extremely high energy efficiency operation. The device also supports secure boot using Ambiq's SecureSPOT technology enabling applications to establish and maintain a root of trust from boot to execution.

The Apollo510 SoC brings enhanced display and user interface capabilities with a rich set of display and graphics features. The display controller supports MIPI DSI and other serial/parallel display interfaces. Up to 4 layer composition and blending is supported with full CPU offload. A powerful 2.5D graphics controller enables a rich UI experience without compromising on power. Frame buffer compression is supported to minimize storage and bandwidth requirements.

A rich set of sensor peripherals enable the monitoring of several sensors. An integrated temperature sensor enables the measurement of ambient temperature. A scalable ultra-low power Successive Approximation Register (SAR) Analog-to-Digital Converter (ADC) monitors the temperature sensor, several internal voltages, and up to eight external sensor signals. The General Purpose ADC is uniquely tuned for minimum power with a configurable measurement mode that does not require MCU intervention.

In addition to integrated analog sensor peripherals, I$^2$C/SPI manager ports and/or UART ports enable the MCU to communicate with external sensors that have digital outputs.

The Apollo510 SoC integrates a full audio subsystem supporting one set of stereo PDM microphones, one analog microphone and two full duplex I$^2$S manager/subordinate ports, one with ASRC support. Additionally, ultra low power wake on voice and wake on keyword is supported.

For higher bandwidth peripherals, the Apollo510 SoC supports two Multi-bit SPI (MSPI) controllers for 1-bit, 2-bit, 4-bit and 8-bit data, and two MSPI controllers capable of up to 16-bit data (four MSPI controllers total).

The Apollo510 SoC also includes a set of timing peripherals and an RTC. The general purpose Timer/Counter Module (TIMER), 32-bit System Timer (STIMER), and the RTC may be driven independently by one of four different clock sources:

▪ Low frequency RC oscillator
▪ High frequency RC oscillator
▪ 32.768 kHz crystal oscillator
▪ High frequency crystal oscillator

The Apollo510 SoC supports highly optimized PWM pattern generation for complex, efficient control operation.

To facilitate development and debug, the Apollo510 SoC is supported by a complete suite of standard software development tools. Ambiq Micro provides drivers for all peripherals along with basic application code to shorten development times. Software debug on the Cortex-M55 is facilitated by the addition of an

Instrumentation Trace Macrocell (ITM), Embedded Trace Macrocell (ETM) and a Trace Port Interface Unit (TPIU). The debug functions are accessible via Serial Wire Debugger (SWD).

In addition, the Performance Monitoring Unit (PMU), a powerful tool used for code profiling and optimization, is available to aid developers' debugging and application optimization.

## 1.3    Functional Differences between BGA and WLCSP Packages

This section highlights the differences between the BGA and WLCSP packages of the Apollo510 SoC due to the reduced pinout of the WLCSP package.

### 1.3.1    GPIO

Some GPIO on the BGA are not brought out on the WLCSP. Those GPIO are GPIO58, 89-92, 94, 136, 139-141, 148-155, 186-189, 195 and 208-223, for a total of 39 GPIO not available on the WLCSP. See the Apollo510 SoC Pin Mapping table in the GPIO section for the GPIO pads offered on the two packages and which VDDH power rail that each pad uses. Note that VDDH5 and its associated pads are not available on the WLCSP package of the Apollo510 SoC.

### 1.3.2    Display Subsystem

The MIPI DBI-Type B / Type C Display Bus Interface and the MIPI DPI-2 interface are not offered on the WLCSP package.

# 2. CPU Subsystem
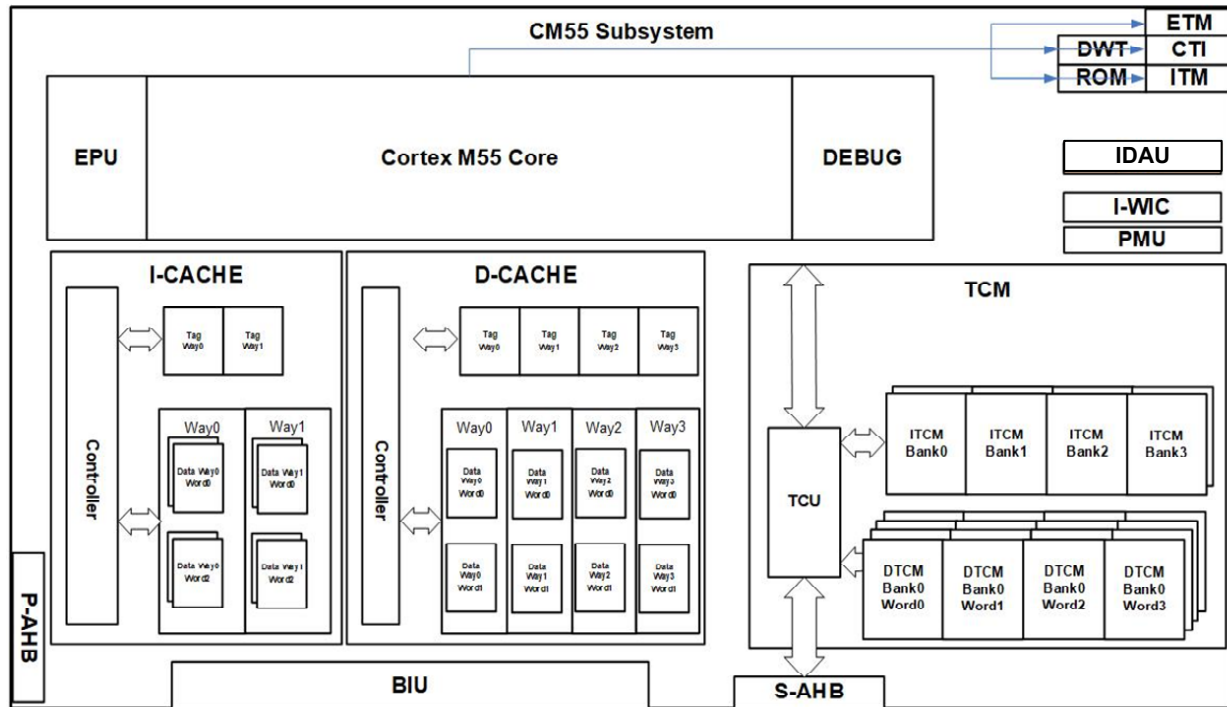


**Figure 3. Apollo5 SoC Family CPU Subsystem Block Diagram**

## 2.1 Features

The CPU subsystem (or CPU complex) includes features as shown in Figure 3 and is composed of an Arm Cortex-M55 CPU with Helium technology, instruction and data caching, instruction and data tightly coupled memory, interrupt and debug logic as well as the associated power management control for the subsystem. The subsystem has the following features:

- Cortex-M55 with Helium technology
  - Half-Precision, Single-Precision and Double Precision Floating Point
  - Vector Processing supporting integer, half-precision and single-precision vector arithmetic
- Arm®v8.1-M ISA with DSP and Security Extensions
- Trustzone® technology support
- Operating Modes
  - 96 MHz Low Power Mode
  - 250 MHz High Performance Mode
- Internal WIC (iWIC) supported
- Memory Protection Unit: 16 non-secure and 16 secure memory regions
- Security attribution Unit: 8 regions
- Debug
  - Embedded Trace Macrocell (ETM)
  - 8x data watchpoint comparators and 8x breakpoint comparators
  - ITM/DWT supported
- CPU Power Management block
- 64 kB Instruction Cache / 64 kB Data Cache
- 256 kB Instruction TCM / 512 kB Data TCM

> **NOTE**
>
> Limitations of DMA operations include the following:
>
> - Transferring between memory/NVM (DTCM, SSRAM, NVM) source and destination is not supported as the APBDMA does not have the capability to transfer from memory to memory. When DTCM is the source of the DMA transfer, the transfer should target a peripheral.
> - A DMA operation cannot be performed to or from DTCM when the CPU is in deep sleep or the DTCM location is powered down.
> - A DMA transfer must not cross a DTCM 4 kB boundary.

## 2.2 Functional Overview

At the center of the Apollo510 SoC is a 32-bit Arm Cortex-M55 core with floating point and vector processing (Helium technology) option. This implementation of the ArmV8.1-M architecture offers highly efficient processing in a very low power design. The Arm M DAP enables debugging access via a Serial Wire Interface from outside of the MCU which allows access to all of the memory, processors and peripheral devices of the SoC. The Apollo510 SoC supports the Cortex-M55 with Helium technology enabling high efficiency compute non-secure processing support. Trustzone ISA extensions are supported to enable secure computing.

The CPU complex has 64 kB of instruction caching and 64 kB of data caching, as well as 256 kB instruction and 512 kB of TCM. In addition, the CPU has access to 3 MB of total shared system SRAM, 4 MB of non-volatile memory and up to 2x 64 MB, 1x 128 MB and 1x 256 MB external memory interfaces. All of the memory is memory mapped and accessible by the CPU. All of the memory accesses are qualified based on the memory protection attributes (enforced within the Cortex-M55) and the system memory protection attributes (enforced within the system memory controllers).

The Cortex-M55 processor supports the Arm®v8.1-M Protected Memory System Architecture (PMSA) that provides programmable support for memory protection using a number of software controllable regions. Memory regions can be programmed to generate faults when accessed inappropriately by unprivileged software reducing the scope of incorrectly written application code. The architecture includes fault status registers to allow an exception handler to determine the source of the fault and to apply corrective action or notify the system.

Reference the "Arm Cortex-M55 Processor Technical Reference Manual" for more details.

## 2.3 Memory Protection Unit (MPU)

The Apollo510 SoC includes an MPU which is a core component for memory protection. The Cortex-M55 processor supports the Arm®v8.1-M Protected Memory System Architecture (PMSA). With the Security Extension, the Cortex-M55 supports both an MPU for the non-secure domain and an S-MPU for the secure domain.

The MPU is an optional component and, when implemented, provides full support for:

- Protection regions
- Access permissions
- Exporting memory attributes to the system

MPU mismatches and permission violations invoke the MemManage handler. See the "ArmV8.1-M Architecture Reference Manual" for more information.

The MPU can be used to enforce privilege rules, separate processes and manage memory attributes. The Apollo510 SoC supports up to 16 memory regions and the S-MCU supports up to 16 *secure* memory regions.

## 2.4    Power Management Overview

Power management is partitioned into several components across the Apollo510 SoC. For the CPU complex, a dedicated finite-state machine controls the transitions of the CPU between power modes. When moving from Active Mode to Deep Sleep Mode, the CPU-PMU manages the state-retention capability of the registers within the Cortex-M55 core and also handshakes with the central power management controller to appropriately handle the voltage rails to the CPU complex. Once in Deep Sleep Mode, the CPU-PMU, in conjunction with the Wake-Up Interrupt Controller, waits for a wakeup event. When the event is observed, the CPU-PMU begins the power restoration process by handshaking with the central power management controller to adjust the voltage rails to the CPU complex and initiate the restoration of the CPU register state. The Cortex-M55 is then returned to active mode once all state is ready.

### 2.4.1    Cortex-M55 Power Modes

The Arm Cortex-M55 supports power modes as listed in Table 1.

**Table 1: Arm Cortex-M55 Power Modes**

| Ambiq M55 Power Mode | Arm M55 Power Mode |
|---|---|
| HP Active | Active |
| Active | |
| Sleep | ON, clock off |
| Deep Sleep | Retention |
| Off | Off |

In addition to the above Arm-defined modes, the Apollo510 SoC supports system level power modes which are defined in the Power Management chapter. See section "Program Control of Power Management" on page 47 which describes programmatic control of power mode transitioning via the MCUPERFREQ register and other module/processor-specific registers in the PWRCTRL register set.

### 2.4.1.1    High Performance Mode

The Apollo510 SoC supports the Ambiq TurboSPOT$^{TM}$ which enables a higher frequency operating mode (HP Mode). In this mode, the Cortex-M55 and all closely coupled memory run at an elevated frequency. All of the non-debug Arm clocks (FCLK, HCLK) also operate at the elevated frequency level. All peripherals are maintained at the nominal frequency level during burst. This mode is entered and exited under software direction but transitions are completely handled in hardware.

This is not a standard Arm-defined power mode. From the Arm core, this mode is treated similarly to "Active Mode".

### 2.4.1.2    Active Mode

In the Active Mode, the Cortex-M55 core is powered up, clocks are active, and instructions are being executed. In this mode, the Cortex-M55 expects all (enabled) devices attached to the interfaces to be powered and clocked for normal access. All of the non-debug Arm clocks (FCLK, HCLK) are active in this state.

To transition from the Active Mode to any of the lower-power modes, a specific sequence of instructions is executed on the Cortex-M55 core. First, specific bits in the Armv8-M System Control Register must be set to determine the mode to enter. See the Armv8-M Architecture Reference Manual for more details.

After the SCR is setup, code can enter the low-power states using one of the three following methods:

▪ Execute a Wait-For-Interrupt (WFI) instruction.
▪ Execute a Wait-For-Event (WFE) instruction.
▪ Set the SLEEPONEXIT bit of the SCR such that the exit from an ISR will automatically return to a sleep state.

The Cortex-M55 will enter a low-power mode after one of these are performed (assuming all conditions are met) and remain there until some event causes the core to return to Active Mode. The possible reasons to return to Active Mode are:

▪ A reset
▪ An enabled Interrupt is received by the NVIC
▪ An event is received by the NVIC
▪ A debug event is received from the DAP

### 2.4.1.3  Sleep Mode

In Sleep Mode, the Cortex-M55 is powered up, but the clocks (HCLK, FCLK) are gated. The power supply is still applied to the Cortex-M55 logic such that it can immediately become active on a wakeup event and begin executing instructions.

### 2.4.1.4  Deep Sleep Mode

In Deep Sleep Mode, the Cortex-M55 enters SRPG mode where the main power is removed, but the flops retain their state. The clocks are not active, and the MCU clock sources for HCLK and FCLK can be deactivated. To facilitate the removal of the source supply and entry into SRPG mode, the Cortex-M55 will handshake with the Wake-up Interrupt Controller and Power Management Unit and set up the possible wakeup conditions.

## 2.5   Debug

The Apollo510 SoC supports several debug features to facilitate software development, profiling and analysis. Standard Serial Wire Debug (SWD) interface is supported along with 1-bit and 4-bit trace data port and optional Serial Wire Output (SWO).

The CPU debug components in the Apollo510 SoC support the following features:

▪ Secure and Non-Secure Debugging
▪ Embedded Trace Macrocell (ETM)
▪ Dedicated 32 kB Embedded Trace Buffer (ETB)
▪ Instrumentation Trace Macrocell (ITM) supporting instrumentation tracing and trace port via Serial Wire Output (SWO)
▪ Trace Port Interface Unit (TPIU)
▪ 8x Data Watchpoint and Trace (DWT)
▪ 8x Breakpoint Unit (BPU)
▪ Hardware fault reporting
▪ Serial Wire Debug (SWD)
▪ Performance Monitoring Unit (PMU)

ERRATUM NOTICE

On the Cortex-M55, self-hosted debug components should be powered on automatically when the Cortex-M55 writes to any of the debug subsystem registers (ITM, WDT, TPIU, etc.). In the Apollo510, writing to the TPIU registers does not automatically power-on the debug subsystem. Therefore software must manually enable the debug power domain, PDDBG, by setting the PWRCTRL_DEVPWREN_PWRENDBG bit, and then can access the debug registers safely. This is required before accessing any self-hosted debug functions, including enabling the SWO for logging or messages. If software writes to a TPIU register when the PDDBG is off, the system hangs. Note that manually enabling PDDBG is not required when an external debugger is attached.

See "ERR003: DEBUG: In self-hosted debug, TPIU register writes when PDDBG is not power up can cause system hang" in the *Apollo510 SoC Errata List*.

ERRATUM NOTICE

The enabled SWO pin goes low during deep sleep mode which violates the SWO specifications. This signal is expected to stay high during deep sleep. The SWO output is in the PDDBG power domain which does remain powered when the Cortex-M55 core goes to sleep. However, these signals are routed through the MCUH power domain which is powered down in deep sleep resulting in the SWO signal being isolated low.

See "ERR032: DEBUG: SWO pin goes low during deep sleep" in the *Apollo510 SoC Errata List*.

### 2.5.1 Embedded Trace Macrocell (ETM)

The Apollo510 SoC supports hardware instruction tracing via an Embedded Trace Macrocell (ETM). The ETM stream is accessible via APB or TPIU. An Embedded Trace Buffer provides 32 kB of trace buffering. Note that while ETM 4.5 is capable of supporting data tracing as well as instruction tracing, data tracing was not implemented in the ETM-M55. However, data-tracing facilities are available through the DWT.

### 2.5.2 Instrumentation Trace Macrocell (ITM)

For system trace the processor integrates an Instrumentation Trace Macrocell (ITM) alongside data watchpoints and a profiling unit. To enable simple and cost-effective profiling of the system events these generate, a Serial Wire Viewer (SWV) can export a stream of software-generated messages, data trace, and profiling information through a single pin.

### 2.5.3 Trace Port Interface Unit (TPIU)

The Apollo510 SoC includes a Cortex-M55 Trace Port Interface Units (TPIU) which can take input from the CPU ITM or ETM. It is an Arm IP component that supports a 1-bit or 4-bit trace data port and Serial Wire Output (SWO). It can act as a bridge between the on-chip trace data from the ITM and the single pin supporting the Serial Wire Viewer Protocol, as well as support the ETM trace output. The TPIU includes a Trace Output Serializer that can format and send the SWV protocol in either a Manchester encoded form or as a standard UART stream.

> **NOTE**
>
> For ETM instruction tracing, Arm recommends the 4-bit trace data port with the TPIU input frequency matching the maximum CPU frequency (250 MHz) for TPIU TRACECLKIN. Note that only GPIO142 to GPIO147 can be used as the TRACE port for clock frequencies above 96 MHz.

### 2.5.4    Serial Wire Debug (SWD)

An external debugger can be connected to the MCU using the Arm Serial Wire Debug (SWD) interface. The SWD interface is a 2-wire interface that is supported by a variety of off-the-shelf commercial debuggers, enabling customers to utilize their development environment of choice.

### 2.5.5    Performance Monitoring Unit (PMU)

The Performance Monitoring Unit (PMU) is a powerful tool used for code profiling and optimization and provides the following capabilities:

- Collect real-time data to fine-tune applications, maximizing efficiency and performance on the Cortex-M55 core.
- Track memory usage and identify system inefficiencies, enhancing responsiveness and minimizing slow memory accesses.
- Optimize AI and digital signal processing by monitoring vector unit utilization and enhancing throughput on complex tasks.
- Gather detailed data on execution patterns, helping identify and resolve bottlenecks faster.

### 2.5.6    Data Watchpoint and Trace (DWT)

The Data Watchpoint and Trace (DWT) of the Cortex-M55 (ArmV8-M) includes watchpoints, data value watchpoints, and trace control signaling which can be used to control the ETM and CTI, performance profiling and more. It does not perform tracing of all data addresses issued by the processor.

See the Arm Cortex-M55 Technical Reference Manual, section 17.1 and ArmV8-M Architecture Reference Manual, section D1.2.63 for more details.

### 2.5.7    Breakpoint Unit (BPU)

The BPU is configured with four instruction address comparators. Each comparator supports breakpoint functionality on all instructions that are fetched across the entire address range in which code is located. If invasive debug is not enabled for a security mode, then debug events associated with breakpoints in that mode are blocked.

### 2.5.8    Faulting Address Trapping Hardware

The Apollo510 SoC offers an optional facility for trapping the address associated with bus faults occurring on any of the three AMBA AHB buses on the chip. This facility must be specifically enabled so that energy is not wasted when one is not actively debugging. The Cortex-M55 core provides bus fault information using the Configurable Fault Status Register that can be configured for usage fault, bus fault and memory management fault reporting. The Auxiliary Fault Status Register provides additional information on the types and causes of faults. The Bus Fault Address Register and the MemManage Fault Address Register capture the memory address that caused one of these faults.

### 2.5.9    PPB/EPPB Interfaces

The ITM, DWT, BPU, CTI and ETM programming interfaces are provided through the Cortex-M55 private peripheral bus (PPB). The system TPIU and the ETB programming interfaces are provided on the external private peripheral bus (EPPB).

## 2.6    Additional Information

Please refer to the MCUCTRL registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 3. Memory Subsystem

## 3.1 Features

The memory for the CPU subsystem consists of the following features.

- 2-way set associative instruction cache
- 4-way set associative data cache
- Tightly-coupled instruction memory
- Tightly-coupled memory
- NVM interface
- System SRAM interface

## 3.2 Functional Overview

The instruction cache for the CPU subsystem caches all instruction accesses to the code region of the memory map. On the Apollo510 SoC, this memory includes internal and external non-volatile memory as well as on-chip and off-chip RAM.

The instruction cache has the following features:

- 64 kB capacity
- 2-way set-associative
- 32-byte line size
- Instruction prefetching

The data cache has the following features:

- 64 kB capacity
- 4-way set-associative
- 32-byte line size
- Write-back and write-through cachable
- Read-allocate and no read-allocate
- Write-allocate and no write-allocate
- Transient and non-transient. Clean cache lines that are associated with transient memory are prioritized for eviction over lines that are associated with non-transient memory
- Data prefetching with programmable look ahead distance

The store buffer has five 64-bit entries and has merging capabilities. If a previous write access has updated an entry, other write accesses on the same double-word can merge into this entry.

The Instruction Tightly Coupled Memory (ITCM) is up to 256 kB in size and provides zero wait-state instruction memory access to the CPU. The ITCM is a low-power/low-latency memory with multiple size configurations.

The TCM is up to 512 kB in size and provides zero wait-state data memory access to the CPU. The DTCM is a low-power/low-latency memory with configurable power settings for multiple size configurations.

---

ERRATUM NOTICE

There are three known restrictions on DMAing to TCM.

1. A DMA transfer cannot cross a DTCM 4 kB boundary.
2. Access to TCM when it is powered down returns an incorrect error condition (Success).
3. Access to TCM while the CPU is in deep sleep with memory retained returns an error. A DMA operation cannot be performed to or from DTCM when the CPU is in deep sleep or the DTCM location is powered down.

See "ERR014: Memory: Restrictions on use of DMA to TCM" in the *Apollo510 SoC Errata List*.

---

The Shared SRAM (SSRAM) is accessible via the SRAM region. The SSRAM is 3072 kB of shared system memory. Access to this memory region incurs 8 or more cycle wait-states, depending on the CPU operating mode.

The CPU sub-system includes a boot ROM which is the initial boot memory for the system. The boot ROM initiates the secure boot flow, if enabled, as well as other primitive/critical helper functions to facilitate accesses such as NVM programming.

---

ERRATUM NOTICE

The option to use data line D4, MSPIn_4, as the clock output line for an MSPI is not supported by the Boot ROM of Apollo510 SoC. The standard clock line, MSPIn_8, must be used by the Boot ROM, i.e., for the purpose of recovering MRAM.

---

## 3.3    Interrupts

Within the SoC, multiple peripherals can generate interrupts. In some cases, a single peripheral may be able to generate multiple different interrupts. Each interrupt signal generated by a peripheral is connected to one of the processing cores. For interrupts that route back to the Cortex-M55 core, these can be routed to two places. First, the interrupts are connected to the Nested Vectored Interrupt Controller, NVIC, in the core. This connection provides the standard changes to program flow associated with interrupt processing. Additionally, they are connected to the WIC outside of the core, allowing the interrupt sources to wake the Cortex-M55 core when it is in a deep sleep (SRPG) mode.

For details on the interrupt model of the Cortex-M55, please see the "Cortex-M55 Devices Generic User Guide."   The VTOR register in the System Control Block (SCB) determines the starting address of the vector table.

In the Apollo510 SoC implementation with the Security Extension, the VTOR is banked so there is a VTOR_S and a VTOR_NS. The initial values of VTOR_S and VTOR_NS are system design specific. The vector table used depends on the target state of the exception. For exceptions targeting the Secure state, VTOR_S is used. For exceptions targeting the Non-secure state, VTOR_NS is used.

Note that for non-secure interrupts, the value in the secure vector table will be an invalid address to generate an exception as this is an error condition (which could be the non-secure address as jumping to it would be a security error). Conversely, for secure-interrupts, the value in the non-secure table will be an

---

invalid address for non-secure code (could be the secure address as this would generate a security error). Banked interrupts means there are secure and non-secure versions of this interrupt. Therefore the vector values in each of the secure and non-secure versions are valid, and should be different (one secure and one non-secure).

### 3.3.1  Vector Table for Apollo510 SoC

In the Apollo510 SoC implementation with the Security Extension, the VTOR is banked so there is a VTOR_S and a VTOR_NS. The initial values of VTOR_S and VTOR_NS are system design specific. The vector table used depends on the target state of the exception. For exceptions targeting the secure state, VTOR_S is used. For exceptions targeting the non-secure state, VTOR_NS is used.

Note that for non-secure interrupts, the value in the secure vector table will be an invalid address to generate an exception as this is an error condition. Conversely, for secure-interrupts, the value in the non-secure table will be an invalid address for non-secure code.   Banked interrupts mean there are secure and non-secure versions of this interrupt. Therefore the vector values in each of the secure and non-secure versions are valid, and should be different (one secure and one non-secure).

Table 2 represents the Cortex-M55 Vector Table for the Apollo510 SoC.

**Table 2: Apollo510 SoC Cortex-M55 Vector Table**

| Exception Number | IRQ | Offset | Vector | Description |
|---|---|---|---|---|
| 150 | 134 | 0x258 | IRQ134 | OTP |
| 149 | 133 | 0x254 | IRQ133 | FP Exception[1] |
| 148 | 132 | 0x250 | IRQ132 | Reserved |
| 141 - 147 | 125 - 131 | 0x234 - 0x24C | IRQ 125 - 131 | GPIO8 - 14: MCUN1INTn (n = 0 to 6) |
| 125 - 140 | 109 - 124 | 0x1F4 - 0x230 | IRQ109 - 124 | Reserved |
| 124 | 108 | 0x1F0 | IRQ108 | Reserved |
| 116 - 123 | 100 - 107 | 0x1D0 - 0x1EC | IRQ100 - 107 | Reserved |
| 115 | 99 | 0x1CC | IRQ99 | IOSFD1 SPI Subordinate Register Access |
| 114 | 98 | 0x1C8 | IRQ98 | IOSFD1 SPI Subordinate |
| 113 | 97 | 0x1C4 | IRQ97 | IOSFD0 SPI Subordinate Register Access |
| 112 | 96 | 0x1C0 | IRQ96 | IOSFD0 SPI Subordinate |
| 108 - 111 | 92 - 95 | 0x1B0 - 0x1BC | IRQ92 - IRQ95 | SW INT[3:0] |
| 107 | 91 | 0x1AC | IRQ91 | Reserved |
| 106 | 90 | 0x1A8 | IRQ90 | Reserved |
| 105 | 89 | 0x1A4 | IRQ89 | Reserved |
| 104 | 88 | 0x1A0 | IRQ88 | Reserved |
| 103 | 87 | 0x19C | IRQ87 | Reserved |
| 102 | 86 | 0x198 | IRQ86 | Reserved |
| 101 | 85 | 0x194 | IRQ85 | Reserved |
| 100 | 84 | 0x190 | IRQ84 | SDIO1 |
| 99 | 83 | 0x18C | IRQ83 | Reserved |

**Table 2: Apollo510 SoC Cortex-M55 Vector Table**

| Exception Number | IRQ | Offset | Vector | Description |
|---|---|---|---|---|
| 98 | 82 | 0x188 | IRQ82 | Timer 15 |
| 97 | 81 | 0x184 | IRQ81 | Timer 14 |
| 96 | 80 | 0x180 | IRQ80 | Timer 13 |
| 95 | 79 | 0x17C | IRQ79 | Timer 12 |
| 94 | 78 | 0x178 | IRQ78 | Timer 11 |
| 93 | 77 | 0x174 | IRQ77 | Timer 10 |
| 92 | 76 | 0x170 | IRQ76 | Timer 9 |
| 91 | 75 | 0x16C | IRQ75 | Timer 8 |
| 90 | 74 | 0x168 | IRQ74 | Timer 7 |
| 89 | 73 | 0x164 | IRQ73 | Timer 6 |
| 88 | 72 | 0x160 | IRQ72 | Timer 5 |
| 87 | 71 | 0x15C | IRQ71 | Timer 4 |
| 86 | 70 | 0x158 | IRQ70 | Timer 3 |
| 85 | 69 | 0x154 | IRQ69 | Timer 2 |
| 84 | 68 | 0x150 | IRQ68 | Timer 1 |
| 83 | 67 | 0x14C | IRQ67 | Timer 0 |
| 82 | 66 | 0x148 | IRQ66 | Reserved |
| 81 | 65 | 0x144 | IRQ65 | Reserved |
| 80 | 64 | 0x140 | IRQ64 | Reserved |
| 79 | 63 | 0x13C | IRQ63 | Reserved |
| 72 - 78 | 56 - 62 | 0x120 - 0x138 | IRQ56 - IRQ62 | GPIO Group 0 - 6: MCUN0INTn (n = 0 to 6) |
| 71 | 55 | 0x11C | IRQ55 | Reserved |
| 70 | 54 | 0x118 | IRQ54 | MSPI3 |
| 69 | 53 | 0x114 | IRQ53 | Reserved |
| 68 | 52 | 0x110 | IRQ52 | Reserved |
| 67 | 51 | 0x10C | IRQ51 | Reserved |
| 66 | 50 | 0x108 | IRQ50 | Reserved |
| 65 | 49 | 0x104 | IRQ49 | Reserved |
| 64 | 48 | 0x100 | IRQ48 | PDM0 |
| 63 | 47 | 0xFC | IRQ47 | Reserved |
| 62 | 46 | 0xF8 | IRQ46 | Reserved |
| 61 | 45 | 0xF4 | IRQ45 | I2S1 |
| 60 | 44 | 0xF0 | IRQ44 | I2S0 |
| 59 | 43 | 0xEC | IRQ43 | Reserved |
| 58 | 42 | 0xE8 | IRQ42 | AUDADC |
| 57 | 41 | 0xE4 | IRQ41 | Reserved |

**Table 2: Apollo510 SoC Cortex-M55 Vector Table**

| Exception Number | IRQ | Offset | Vector | Description |
|---|---|---|---|---|
| 56 | 40 | 0xE0 | IRQ40 | Stimer Capture/Overflow |
| 48 - 55 | 32 - 39 | 0xC0 - 0xDC | IRQ32 - IRQ39 | Stimer Compare[0:7] |
| 47 | 31 | 0xBC | IRQ31 | Reserved |
| 46 | 30 | 0xB8 | IRQ30 | DSI |
| 45 | 29 | 0xB4 | IRQ29 | Display |
| 44 | 28 | 0xB0 | IRQ28 | Graphics |
| 43 | 27 | 0xAC | IRQ27 | USB |
| 42 | 26 | 0xA8 | IRQ26 | SDIO0 |
| 41 | 25 | 0xA4 | IRQ25 | CRYPTO Non-Secure |
| 40 | 24 | 0xA0 | IRQ24 | CRYPTO Secure |
| 39 | 23 | 0x9C | IRQ23 | Clock Control |
| 38 | 22 | 0x98 | IRQ22 | MSPI2 |
| 37 | 21 | 0x94 | IRQ21 | MSPI1 |
| 36 | 20 | 0x90 | IRQ20 | MSPI0 |
| 35 | 19 | 0x8C | IRQ19 | GPADC |
| 34 | 18 | 0x88 | IRQ18 | UART3 |
| 33 | 17 | 0x84 | IRQ17 | UART2 |
| 32 | 16 | 0x80 | IRQ16 | UART1 |
| 31 | 15 | 0x7C | IRQ15 | UART0 |
| 30 | 14 | 0x78 | IRQ14 | Counter/Timers (combined - also see IRQ 65 - 81) |
| 29 | 13 | 0x74 | IRQ13 | $I^2$C/SPI Manager7 |
| 28 | 12 | 0x70 | IRQ12 | $I^2$C/SPI Manager6 |
| 27 | 11 | 0x6C | IRQ11 | $I^2$C/SPI Manager5 |
| 26 | 10 | 0x68 | IRQ10 | $I^2$C/SPI Manager4 |
| 25 | 9 | 0x64 | IRQ9 | $I^2$C/SPI Manager3 |
| 24 | 8 | 0x60 | IRQ8 | $I^2$C/SPI Manager2 |
| 23 | 7 | 0x5C | IRQ7 | $I^2$C/SPI Manager1 |
| 22 | 6 | 0x58 | IRQ6 | $I^2$C/SPI Manager0 |
| 21 | 5 | 0x54 | IRQ5 | IOS $I^2$C/SPI Subordinate Register Access |
| 20 | 4 | 0x50 | IRQ4 | IOS $I^2$C/SPI Subordinate |
| 19 | 3 | 0x4C | IRQ3 | Voltage Comparator |
| 18 | 2 | 0x48 | IRQ2 | RTC |
| 17 | 1 | 0x44 | IRQ1 | Watchdog Timer |
| 16 | 0 | 0x40 | IRQ0 | Brownout Detection |
| 15 | -1 | 0x3C | Systick(_S) | |

**Table 2: Apollo510 SoC Cortex-M55 Vector Table**

| Exception Number | IRQ | Offset | Vector | Description |
|---|---|---|---|---|
| 14 | -2 | 0x38 | PendSV(_S) | |
| 13 | - | 0x34 | Reserved | |
| 12 | -4 | 0x30 | DebugMonitor | |
| 11 | -5 | 0x2C | SVCall(_S) | |
| 10 | - | 0x28 | Reserved | |
| 9 | - | 0x24 | | |
| 8 | - | 0x20 | | |
| 7 | -9 | 0x1C | SecureFault_S | Secure Fault |
| 6 | -10 | 0x18 | UsageFault_S | Usage Fault |
| 5 | -11 | 0x14 | BusFault_S | Bus Fault |
| 4 | -12 | 0x10 | MemoryManage_S | Memory Management Fault |
| 3 | -13 | 0xC | HardFault_S | Hard Fault |
| 2 | -14 | 0x8 | NMI_S | |
| 1 | - | 0x4 | Reset | |
| | | 0x0 | Initial SP | |

1. Floating point exception interrupt is raised when one of six possible FP errors is detected. The FPSCR register must be interrogated to determine the source(s) of the interrupt. Note that the interrupt is not banked (nor is FPSCR), so the interrupt must be reassigned secure if a FP secure context is created. (The FPDSCR is banked).

### 3.3.2   GPIO Extension

The Apollo5 family supports up to 190 GPIOs depending on device and package, so the mapping is extended such that 0-6 are groups of 32 GPIO interrupts, and 8-14 form the second group (GPIO interrupts 7 and 15 are reserved). See "GPIO Interrupts" on page 80 for more information about GPIO interrupt grouping and operation.

## 3.4 Memory Maps

Table 3 shows the memory map for the Arm Cortex-M55 CPU used by the .

**Table 3: Cortex-M55 CPU Memory Map**

| Type | Address Range | Name | Executable | Description |
|---|---|---|---|---|
| Code | 0x00000000 - 0x0003FFFF | ITCM | Y | Instruction TCM (256 kB) |
| | 0x00040000 - 0x003FFFFF | Reserved | X | Reserved |
| | 0x00400000 - 0x0040FFFF | NVM-SBL | Y | NVM reserved for Secure Boot Loader (64 kB) |
| | 0x00410000 - 0x007FFFFF | NVM-APPL | Y | NVM for application (3.94 MB) |
| | 0x00800000 - 0x01FFFFFF | Reserved | X | Reserved |
| | 0x02000000 - 0x0201FFFF | Boot Loader ROM | Y | Boot Loader ROM (128 kB) |
| | 0x02020000 - 0x0FFFFFFF | Reserved | X | Reserved |
| | 0x10000000 - 0x1003FFFF | ITCM | Y | Instruction TCM (256 kB) [Aliased to 0x00000000] |
| | 0x10040000 - 0x1FFFFFFF | Reserved | X | Reserved |
| SRAM | 0x20000000 - 0x2007FFFF | DTCM | Y | Data TCM (512 kB) |
| | 0x20080000 - 0x2037FFFF | System SRAM | Y | System SRAM (3 MB) |
| | 0x20380000 - 0x2FFFFFFF | Reserved | X | Reserved |
| | 0x30000000 - 0x3007FFFF | DTCM | Y | Data TCM (512 kB) [Aliased to 0x20000000] |
| | 0x30080000 - 0x3FFFFFFF | Reserved | X | Reserved |
| Peripheral | 0x40000000 - 0x4FFFFFFF | Peripheral | N | See Peripheral Memory Map |
| | 0x50000000 - 0x5FFFFFFF | Reserved | X | Reserved |
| External RAM | 0x60000000 - 0x6FFFFFFF | MSPI0 | Y | External Memory (256 MB) |
| | 0x70000000 - 0x7FFFFFFF | Reserved | X | Reserved |
| | 0x80000000 - 0x83FFFFFF | MSPI1 | Y | External Memory (64 MB) |
| | 0x84000000 - 0x87FFFFFF | MSPI2 | Y | External Memory (64 MB) |
| | 0x88000000 - 0x8FFFFFFF | MSPI3 | Y | External Memory (128 MB) |
| | 0x90000000 - 0x9FFFFFFF | Reserved | X | Reserved |
| External Device | 0xA0000000 - 0xDFFFFFFF | Reserved | X | Reserved |
| Private Peripheral Device | 0xE0000000 - 0xE00FDFFF | PPB | N | NVIC, System timers, System Control Block, MPU, TPIU, ETB, EWIC |
| | 0xE00FE000 - 0xE00FEFFF | Debug ROM | X | Debug ROM |
| | 0xE00F1000 - 0xE00FFFFF | PPB | X | Reserved |
| Vendor_ SYS | 0xE0100000 - 0xFFFFFFFF | Reserved | X | Reserved |

Peripheral devices within the memory map are allocated on 4 kB boundaries (mostly), allowing each device up to 1024 32-bit control and status registers. Peripherals will return undefined read data when an attempt to access a register which does not exist occurs. Peripherals will always accept any write data sent to their registers without attempting to return an error response. Specifically, a write to a read-only register would just become a don't-care write.

Table 4 shows the memory map for the peripheral devices. The CPU has access to all peripheral devices. This table illustrates how SAU partitions are used to make certain peripheral regions secure when running with a TF-M secure partition enabled. OTP registers are separated from MRAM control to allow OTP registers to be secure via the SAU while the MRAM registers remain non-secure.

**Table 4: Apollo510 SoC Peripheral Memory Map**

| Address Range | Device Name |
|---|---|
| 0x40000000 - 0x400003FF | Reset/BoD Control (1 kB) |
| 0x40000400 - 0x40003FFF | Reserved |
| 0x40004000 - 0x400041FF | Clock Generator (0.5 kB) |
| 0x40004200 - 0x400047FF | Reserved |
| 0x40004800 - 0x40004BFF | RTC (1 kB) |
| 0x40004C00 - 0x40007FFF | Reserved |
| 0x40008000 - 0x400083FF | Timers (1 kB) |
| 0x40008400 - 0x400087FF | Reserved |
| 0x40008800 - 0x400089FF | STImer (0.5 kB) |
| 0x40008A00 - 0x4000BFFF | Reserved |
| 0x4000C000 - 0x4000C3FF | Voltage Comparator (1 kB) |
| 0x4000C400 - 0x4000FFFF | Reserved |
| 0x40010000 - 0x400107FF | GPIO Control (2 kB) |
| 0x40010800 - 0x40013FFF | Reserved |
| 0x40014000 - 0x40014FFF | MRAM/OTP Control (4 kB) |
| 0x40015000 - 0x4001FFFF | Reserved |
| 0x40020000 - 0x400207FF | Miscellaneous Control (MCUCTRL) (2 kB) |
| 0x40020800 - 0x40020FFF | Reserved |
| 0x40021000 - 0x400213FF | Power Control(1 kB) |
| 0x40021400 - 0x40023FFF | Reserved |
| 0x40024000 - 0x400243FF | Watchdog Timer (1 kB) |
| 0x40024400 - 0x40024FFF | Reserved |
| 0x40025000 - 0x40025FFF | SSC (4 kB) |
| 0x40025400 - 0x4002FFFF | Reserved |
| 0x40030000 - 0x400303FF | Security (1 kB) |
| 0x40030400 - 0x40033FFF | Reserved |
| 0x40034000 - 0x400343FF | I$^2$C/SPI Subordinate (1 kB) |

**Table 4: Apollo510 SoC Peripheral Memory Map**

| Address Range | Device Name |
|---|---|
| 0x40034400 - 0x40034FFF | Reserved |
| 0x40035000 - 0x400353FF | IOSFD0 SPI Subordinate (1 kB) |
| 0x40035400 - 0x40035FFF | Reserved |
| 0x40036000 - 0x400363FF | IOSFD1 SPI Subordinate (1 kB) |
| 0x40036400 - 0x40037FFF | Reserved |
| 0x40038000 - 0x400383FF | GPADC (1 kB) |
| 0x40038400 - 0x40038FFF | Reserved |
| 0x40039000 - 0x400393FF | UART0 (1 kB) |
| 0x40039400 - 0x40039FFF | Reserved |
| 0x4003A000 - 0x4003A3FF | UART1 (1 kB) |
| 0x4003A400 - 0x4003AFFF | Reserved |
| 0x4003B000 - 0x4003B3FF | UART2 (1 kB) |
| 0x4003B400 - 0x4003BFFF | Reserved |
| 0x4003C000 - 0x4003C3FF | UART3 (1 kB) |
| 0x4003C400 - 0x4004FFFF | Reserved |
| 0x40050000 - 0x40050FFF | I$^2$C/SPI Manager0 (4 kB) |
| 0x40051000 - 0x40051FFF | I$^2$C/SPI Manager1 (4 kB) |
| 0x40052000 - 0x40052FFF | I$^2$C/SPI Manager2 (4 kB) |
| 0x40053000 - 0x40053FFF | I$^2$C/SPI Manager3 (4 kB) |
| 0x40054000 - 0x40054FFF | I$^2$C/SPI Manager4 (4 kB) |
| 0x40055000 - 0x40055FFF | I$^2$C/SPI Manager5 (4 kB) |
| 0x40056000 - 0x40056FFF | I$^2$C/SPI Manager6 (4 kB) |
| 0x40057000 - 0x40057FFF | I$^2$C/SPI Manager7 (4 kB) |
| 0x40058000 - 0x4005FFFF | Reserved |
| 0x40060000 - 0x400603FF | MSPI Manager0 (1 kB) |
| 0x40060400 - 0x40060FFF | Reserved |
| 0x40061000 - 0x400613FF | MSPI Manager1 (1 kB) |
| 0x40061400 - 0x40061FFF | Reserved |
| 0x40062000 - 0x400623FF | MSPI Manager2 (1 kB) |
| 0x40062400 - 0x40062FFF | Reserved |
| 0x40063000 - 0x400633FF | MSPI Manager3 (1 kB) |
| 0x40063400 - 0x4006FFFF | Reserved |
| 0x40070000 - 0x400703FF | SDIO0 (1 kB) |
| 0x40070400 - 0x40070FFF | Reserved |
| 0x40071000 - 0x400713FF | SDIO1 (1 kB) |
| 0x40071400 - 0x4008FFFF | Reserved |

**Table 4: Apollo510 SoC Peripheral Memory Map**

| Address Range | Device Name |
|---|---|
| 0x40090000 - 0x4009FFFF | Graphics Subsystem (64 kB) |
| 0x400A0000 - 0x400A7FFF | Display Controller (32 kB) |
| 0x400A8000 - 0x400AFFFF | Display PHY (32 kB) |
| 0x400B0000 - 0x400B3FFF | USB (16 kB) |
| 0x400B4000 - 0x400B7FFF | USB PHY (16 kB) |
| 0x400B8000 - 0x400BFFFF | Reserved |
| 0x400C0000 - 0x400C7FFF | Crypto (32 kB) |
| 0x400C8000 - 0x40200FFF | Reserved |
| 0x40201000 - 0x402013FF | PDM0 (1 kB) |
| 0x40201400 - 0x40207FFF | Reserved |
| 0x40208000 - 0x402083FF | I2S0 (1 kB) |
| 0x40208400 - 0x40208FFF | Reserved |
| 0x40209000 - 0x402093FF | I2S1 (4 kB) |
| 0x40209400 - 0x4020FFFF | Reserved |
| 0x40210000 - 0x402103FF | AUDADC (LPADC) (1 kB) |
| 0x40210400 - 0x41FFFFFF | Reserved |
| 0x42000000 - 0x420007FF | NVM_INFO0 (2 kB) |
| 0x42000800 - 0x42001FFF | Reserved |
| 0x42002000 - 0x420037FF | NVM_INFO1 (6 kB) |
| 0x42003800 - 0x42003FFF | Reserved |
| 0x42004000 - 0x420040FF | OTP2_INFO0 (256 Bytes) |
| 0x42004100 - 0x42005FFF | Reserved |
| 0x42006000 - 0x42006AFF | OTP2_INFO1 (1.375 kB) |
| 0x42006B00 - 0x4FFFFFFF | Reserved |

## 3.5 Static Random Access Memory (SRAM)
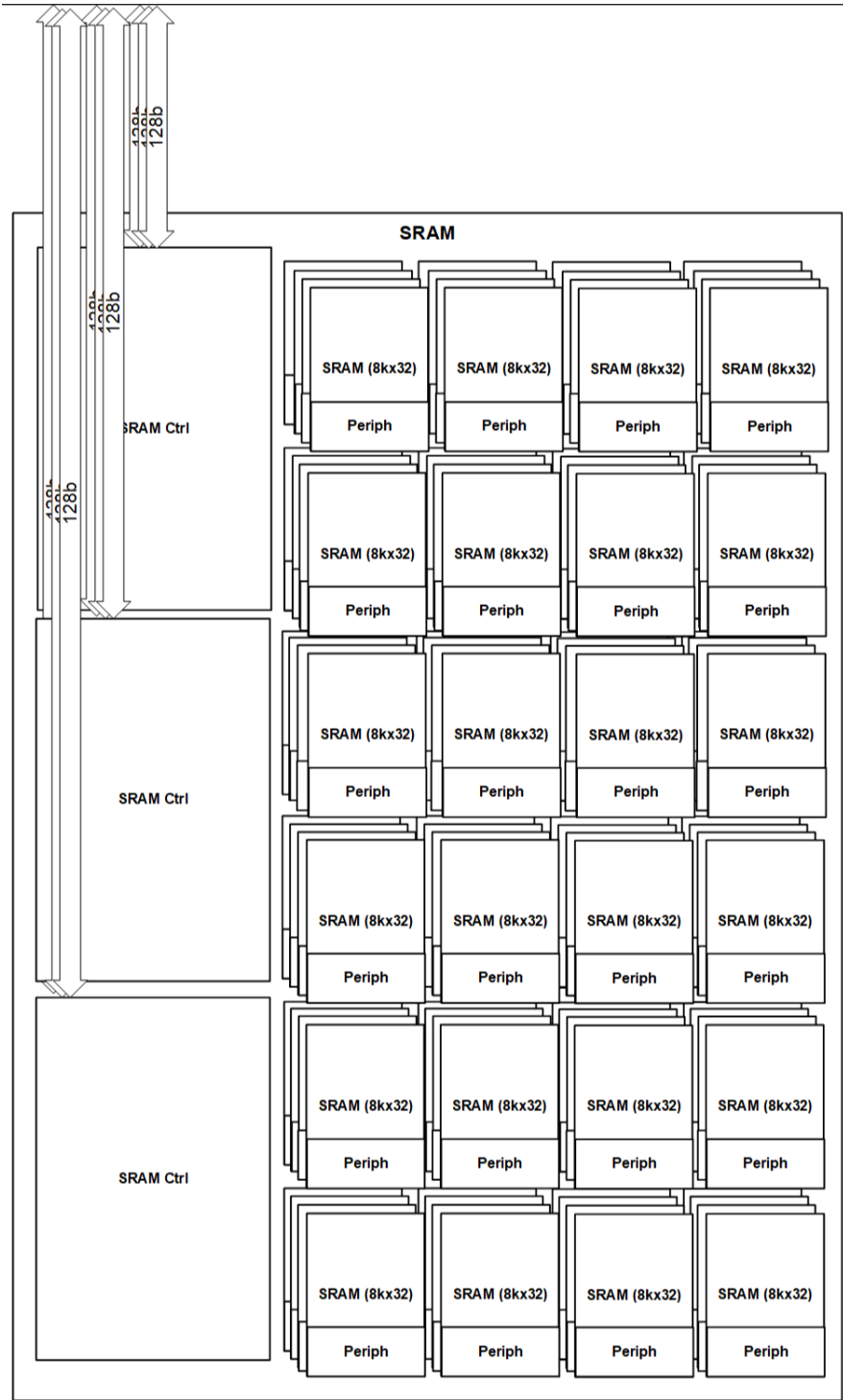


**Figure 4. SRAM Block Diagram**

### 3.5.1 Features

The Static Random Access Memory (SRAM) Controller supports features shown in Figure 4 and listed below.

- Fully synchronous design at 96 MHz (nominal)
- Three (3) 128-bit wide main AXI interfaces
- Internal arrays have 128 kB granularity. Each array can be concurrently accessed unless a physical conflict exists for the same 128 kB region.
- Supports concurrent operation to 3 independent 128 kB banks
- Partitioned into 2 power domains

Please refer to the MCUCTRL registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

### 3.5.2 Functional Overview

The Apollo510 SoC SRAM module provides up to 3 MB of shared internal SRAM for the system. It has three primary 128-bit AXI system interfaces and accommodates up to three simultaneous SRAM operations per cycle as long as the RAM bank doesn't overlap. Below are the port mappings:

- Port 0: DMA
- Port 1: GPU, Display Controller
- Port 2: CPU

## 3.6    OTP
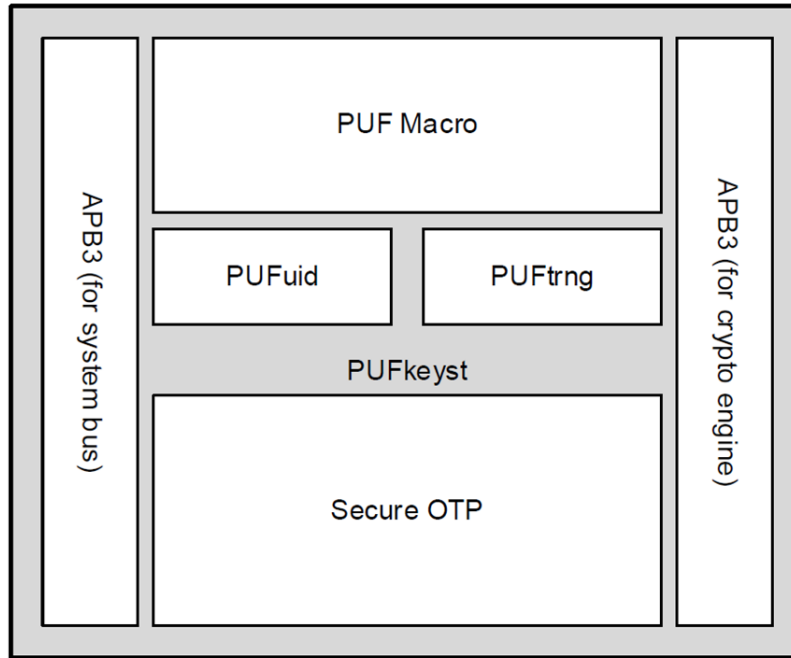


**Figure 5. OTP Block Diagram**

### 3.6.1    Features

The One-Time Programmable (OTP) and Physically Unclonable Function (PUF) include features[1] shown in Figure 5 and listed below.

- OTP anti-fuses
- PUFuid on NeoPUF: 2 kbits on-chip Identification
- PUFtrng: PUF-based True Random Number Generator
  - Conditioning & Entropy Health Test

---

ERRATUM NOTICE

An access to the OTP info space or OTP registers when the OTP is powered down will result in an APB bus hang. The OTP power domain is powered up by setting the PWRCTRL_DEVPWREN_PWRENOTP field to Enable. This field is enabled by default.

See "ERR022: PWRCTRL: APB bus hang when accessing unpowered OTP registers or INFO space" in the *Apollo510 SoC*.

---

1. Some of the features are reserved for Ambiq use.

> **ERRATUM NOTICE**
>
> The SoC design requires that the CPU be powered on when either OTP or ROM is powered on.
>
> See "ERR023: PWRCTRL: CPU cannot go into deep sleep when either OTP or ROM is powered on" in the *Apollo510 SoC Errata List*.

### 3.6.2 Functional Overview

The OTP block on the Apollo510 SoC includes multiple regions of OTP anti-fuse bits, anti-tampering support, read/write locking and dedicated interfaces for system and crypto access. The OTP anti-fuses are divided into the following three regions:

- INFO0: 256 bytes (2048 bits)
- INFO1: 1.375 kB (11 kbits)
- INFOC: 1024 bytes (8 kbits)

#### 3.6.2.1 PUF Features/Functionality

PUF features[1] on the Apollo510 SoC are described below.

- **NeoPUF®** - A weak PUF defined as a PUF with limited CRPs (challenge-response-pairs).
  - Able to be used for generating true random bits which can act as a silicon fingerprint.
  - Has the near ideal PUF characteristics of 50% HW (Hamming-Weight), 50% Inter-HD (Hamming-Distance), 0% Intra-ID HD and 0ppm BER (Bit-Error-Rate).
  - Passes the NIST SP800-22 and NIST SP80090B IID statistical analysis test suites.
- **PUFuid®**
  - Offers 1 kilobit unique keys, which are derived from NeoPUF and programmed into PUF array in OTP by Ambiq. These keys can be used as a source for a unique ID, root key or entropy source, which is different for each chip.
- **PUFtrng®**
  - True random number generator with a PUF-based refinement engine.
  - Leverages 1 kilobits of PUF data for initial seeding.

##### 3.6.2.1.1 PUF System Side Function Descriptions

System side function descriptions are as follows. PUF register addresses listed are offsets from a base address of 0x40014800.

- **Get Random Number Output (0x12A0)**
  - Provides high-quality random number bits (32 bits each time) for security usage.
  - Open to all security levels unless memory range is mapped Secure.
- **Read UID (0x1300 - 0x137C)**
  - Reads from a 1 kilobit PUF Array which contains all the random data that is used for UIDs and the entropy source for the TRNG.
  - 32 UID locations ranging from 0x1300 to 0x137C.
  - Read the UID from PUF cell.
  - These registers are labeled puf_000 - puf_031.

---

1. Certain functions have been protected and are only accessible to secure software if a secure execution environment (SEE) is instantiated.

- Each PUF location provides 32 bits static entropy which can act as a silicon fingerprint for the various security applications of identification, authentication, and local data protection.
- Only accessible to the Secure domain, if SEE is instantiated.

## 3.7    Additional Information

Please refer to the MCUCTRL and OTP registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 4.  System Power Management

System Power Management on the Apollo510 SoC is handled through a combination of hardware and software. The hardware handles the interface and control sequencing between the regulators and the individual power domains within the SoC. The software initiates transitions through power states by processor instructions and system level power control commands.

## 4.1  Functional Overview

The Power Management system is composed of a central power management controller and various power management units (PMUs) for each primary subsystem. The primary PMUs are as follows:

- CPU PMU: responsible for power sequencing for the CPU subsystem
- GPU PMU: responsible for power sequencing for the graphics subsystem
- IO/Peripheral PMU: responsible for power sequencing at each I/O subsystem
- Memory PMU: responsible for power sequencing each memory subsystem/power bank

### 4.1.1  CPU Power Management Unit

When moving from Active Mode to Deep Sleep Mode, the CPU PMU manages the state-retention capability of the registers within the Cortex-M55 core and controls the interface to the voltage regulators as needed to support the various operating modes of the CPU. Once in the Deep Sleep Mode, the CPU PMU, in conjunction with the Wake-Up Interrupt Controller, waits for a wakeup event. When the event is observed, the CPU PMU begins the power restoration process by re-enabling the on-chip voltage regulators and restoring the CPU register state. The Cortex-M55 is then returned to active mode.

The CPU-PMU enables support for the following Arm Cortex-M55 defined power modes:

- OFF
- Deep Sleep
- Sleep
- Active
  - Low Power / High Efficiency: 96 MHz (LP)
  - High Performance: 250 MHz

### 4.1.2  GPU Power Management Unit

When transitioning power states (such as Active LP to Active HP), the system software manages the voltage power mux and clock selection as well as the appropriate isolation and reset controls as needed.

The system software enables support for the following defined power modes:

- OFF
- Sleep/Idle (device is enabled but no active transactions)
- Active
  - Low Power / High Efficiency: 96 MHz
  - High Performance: 250 MHz

### 4.1.3  IO/Peripheral Power Management Unit

The IO/Peripheral PMU's manage power state for I/O subsystems. Each I/O subsystem will support the following power modes. Note that each I/O subsystem may have differences in power state implementation (i.e., what "Active" means). This is implementation specific to each I/O controller. Also, not all power modes may be supported by each IO/Peripheral PMU (typical configuration may support only OFF and Active power modes).

- OFF
- Sleep/Idle: device is enabled but no active transactions

▪ Active

### 4.1.4 Memory Power Management Unit

The Memory PMUs manage power state for the various memory subsystems. Each memory subsystem will support the following power modes.

▪ OFF (no memory retained for volatile RAM, power down mode for non-volatile memory)
▪ Retention (corresponding memory is held at retention level, all memory contents are maintained)
▪ Sleep/Idle (device is enabled but no active transactions)
▪ Active

Note that the primary power state for all memory in the SoC is dependent on the CPU power state. Once CPU goes into Deep Sleep mode, all memories can enter into retention or OFF mode (depending on the software configuration). When in this mode, DMA transactions or accesses from other peripherals (such as Graphics) are on-demand. On access, the PMU ensures the target memory is powered up into Active mode before servicing the transaction.

## 4.2 Power Management Controller

The power management controller provides control functions for each supply regulator as well as the primary power gates under digital logic control. The power management controller receives input from all PMUs indicating requested power levels and also controls from software (via power management control registers). A power management mapping configuration is also provided (sourced from trims shadowed to the power management controller) which dictates the functional operation at the regulator interface based on the input power requests. This mapping configuration allows the power management functionality to be programmatic enabling characterization, tuning and/or bug fixes.

Following are the supply regulator interfaces:

▪ SIMO Buck
▪ Core LDO
▪ Mem LDO
▪ MEM LP LDO

The central power controller is also responsible for controlling power gate enables for all digital power domains. The power gate enables are controlled based on the power level requests. When an "OFF" level is requested from the respective requester PMU or a software override is asserted to force a requester "OFF" or, for I/O requesters, when the corresponding I/O device enable is de-asserted, the respective power gate enable is asserted to power off the domain. For all other power level requests, the power gate is disabled powering up the respective domain. It also controls the power muxes that source the various voltages to each of the corresponding power domains.

SRAM and MRAM power domains are controlled based on the dependent requester domains. For MRAM, if CPU PMU requester is "OFF" and DMA requesters are "OFF" or "Sleep", the MRAM power domain is powered OFF. For SRAM, each SRAM is powered OFF either based on the SKU memory configuration or if CPU requester is "OFF" and DMA requesters are "OFF" or "Sleep" and the SRAM is enabled to power off based on the power control MEMPWDINSLEEP configuration.

### 4.2.1 System Power States

At the SoC level, various power states are supported to enable key workloads and ensure maximum power efficiency. The system power states are defined in the following sub-sections.

#### 4.2.1.1 SYS Active (SACT)

The CPU is in one of the Active Modes and executing instructions. All respective memory and I/O devices are powered on and available as needed.

### 4.2.1.2   SYS Sleep Mode 0 (SS0)

SYS Sleep Mode 0 is a low power state for the SoC and can be entered if all processor cores are in sleep mode or deeper sleep state.

In this mode, the following conditions apply:

- All enabled SRAM memory is retained (up to 3.75 MB).
- NVM memory is in standby.
- HFRC is on.
- Main core clock domain(s) is gated but peripheral clock domains can be on.

This state can be entered if a peripheral device (SPI/UART/I2C/MSPI, etc) is actively (or intermittently) transferring data but the window of acquisition is long enough to allow the processor(s) to go into a deeper low power state.

### 4.2.1.3   SYS Sleep Mode 1 (SS1)

SYS Sleep Mode 1 can be entered if all processor cores are in sleep mode or deeper sleep state, and all peripheral devices are idle and no peripheral device (SPI/UART/I$^2$C/MSPI, etc) is actively transferring data. However, communication may occur within a short time window which prevents the processor(s) from entering Deep Sleep Mode (and subsequently the system from entering a lower power state).

This state is also referred to as "Active Idle" whereby all power domains can be powered on but all clocks are gated. This state is a good power baseline for the system as it represents the active mode DC power level. Typically, the power in this state is dominated by leakage and always-on functional blocks.

In this mode, the following conditions apply:

- All enabled SRAM memory is retained (up to 3.75 MB).
- NVM memory is in standby.
- HFRC is on.
- All functional clocks are gated.

### 4.2.1.4   SYS Deep Sleep Modes

There are 4 levels of SYS Deep Sleep Mode as described in the sub-sections below. Each of the four modes represents a deeper level of sleep and a proportionate reduction in power draw.

In Deep Sleep Mode 0-2, SRAM (SSRAM and TCM) can have a configurable amount of instances in retention depending on the software/system functional and latency requirements. SSRAM retention options, which do not include available selections for size of TCM retention, are the following:

- 3096 kB
- 2048 kB
- 1024 kB
- 0 kB

The ITCM/DTCM retention options in these modes are:

- 256 kB of ITCM and 512 kB of DTCM
- 128 kB of ITCM and 256 kB of DTCM
- 32 kB of ITCM and 128 kB of DTCM
- 0 kB of ITCM and 0 kB of DTCM

#### 4.2.1.4.1   SYS Deep Sleep Mode 0 (SDS0)

SYS Deep Sleep Mode 0 is a deep low power state for the SoC. This state can be entered if a peripheral device (SPI, UART, I$^2$C, MSPI, etc.) is actively or intermittently transferring data but the window of acquisition is long enough to allow the processor(s) to go into a deeper low power state.

In this mode, the following conditions apply:

▪ All processors are in Deep Sleep mode and/or are powered OFF.
▪ Configurable amount of SRAM is in retention (capacity controlled by software).
▪ Cache memory is in retention.
▪ NVM memory is in power down.
▪ HFRC can be on depending on the state of the peripherals needing the HFRC clock.
▪ Main processor power domains are off but peripheral power domains can be on.

### 4.2.1.4.2  SYS Deep Sleep Mode 1 (SDS1)

SYS Deep Sleep Mode 1 is another deep low power state for the SoC. This state can be entered if the latency to warm up the cache can be tolerated. This could be an extended wait for a peripheral communication event.

In this mode, the following conditions apply:

▪ All processors are in Deep Sleep mode or are powered OFF.
▪ Configurable amount of SRAM is in retention (capacity controlled by software).
▪ Cache memory is powered off (no retention).
▪ NVM memory is in power down.
▪ HFRC can be on depending on the state of the peripherals needing the HFRC clock.
▪ Main processor power domains are off but peripheral power domains can be on.

### 4.2.1.4.3  SYS Deep Sleep Mode 2 (SDS2)

SYS Deep Sleep Mode 2 is the minimum power state that the processor(s) can resume normal operation. This state can be entered when all activity has suspended for a duration of time sufficient to sustain the longer exit latencies to resume. This could be a state where periodic data samples are taken and the data is locally processed/accumulated/transferred at long time intervals. This state can only be entered (vs SDS1) if the peripheral devices are either not enabled/active or if the application can afford to save/restore the state of the controller(s) on entry/exit of this mode.

In this mode, the following conditions apply:

▪ All processors are in Deep Sleep or are powered OFF.
▪ Minimal SRAM memory is retained as needed for software to resume.
▪ Cache memory is powered off (no retention).
▪ NVM memory is in power down.
▪ HFRC is off.
▪ XTAL can be on depending on the configuration of timers.
▪ All internal switched power domains are off/gated.
▪ Processor logic state is retained.

### 4.2.1.4.4  SYS Deep Sleep Mode 3 (SDS3)

SYS Deep Sleep Mode 4 is a deep sleep power state that can be entered on long periods of inactivity.

 In this mode, the following conditions apply:

▪ All processors are in Deep Sleep or are powered OFF.
▪ No memory is in retention, all memory is powered down.
▪ LFRC is on (HFRC and XTAL are off)
▪ All internal switched power domains are off/gated.
▪ Processor logic state is retained.

### 4.2.1.5 SYS OFF Mode (SOFF)

In SYS OFF Mode, SoC is completely powered down with no power supplied. This mode is controlled external to the SoC by removing power to the device.

- Processors are in shutdown mode with no state retention.
- Only NVM memory is retained.

## 4.3 Program Control of Power Management

The transition between power modes is largely directed by software and the workload/task requirements.

### 4.3.1 CPU

Transitioning to different power modes is initiated through a couple methods depending on the software intent and performance/power/latency requirements. The different mode transitions are described in the following table.

**Table 5: Transitioning among Power Modes**

|  | Deep Sleep | Sleep | Active - LP | Active - HP |
|---|---|---|---|---|
| **Sleep/ Deep Sleep** | X | X | WAKE+LP | WAKE+HP |
| **Active - LP** | WFI (DS) | WFI (S) | X | HP |
| **Active - HP** | WFI (DS) | WFI (S) | LP | X |

- WFI (DS): Issue WFI instruction Deep Sleep
- WFI (S): Issue WFI instruction Sleep
- LP: Write to CLKGEN_PERFREQ register indicating "LP". Check CLKGEN_PERFSTAT to check if mode switch completed and if performance mode is available. Continue execution.
- HP: Write to CLKGEN_PERFREQ register indicating "HP". Check CLKGEN_PERFSTAT to check if mode switch completed and if performance mode is available. Continue execution.

CPU active state transitions are controlled via the PWRCTRL_MCUPERFREQ register. Software requests the appropriate performance mode (LP or HP mode) by writing to the MCUPERFREQ field as appropriate. Software must check that it received an acknowledge of the performance mode request (by checking the MCUPERFACK and MCUPERFSTATUS fields. It is possible that the PMU will not honor a performance mode change request (i.e. in the case where max power condition might be exceeded). Once acknowledged, the CPU should be operating in the corresponding performance mode until software switches to a different mode.

---

ERRATUM NOTICE

After transitioning from low power (LP) mode to high performance (HP) mode, the MCU still uses the clock for LP (HFRC) after the switch if the clock for HP (HFRC2) is not enabled before switching. For the HP HFRC2 clock to immediately go into effect upon transition from LP mode to HP mode, the HFRC2 clock must be enabled and allowed a settling time for the HFRC2 clock to stabilize before the transition.

See "ERR024: PWRCTRL: MCU continues to use the clock for LP mode (HFRC) after switching from LP to HP mode" in the *Apollo510 SoC Errata List*.

---

### 4.3.2    GPU

GPU active state transitions are controlled via the PWRCTRL_GFXPERFREQ and PWRCTRL_GFXPWRSWSEL registers. Software must sequence the GFXPWRSWSEL and GFXPERFREQ settings properly to ensure proper transition to/from HP and LP mode. Also note that switching of operating mode for GFX must only be done when the GFX device is powered OFF.

For any HP mode transition, software must perform the following sequence:

1.    Set GFXPWRSWSEL_GFXVDDSEL bit to "VDDF" (0x1) level.
2.    Set the desired GFXPERFREQ field setting.

### 4.3.3    I/O

The I/O power modes are determined by how the I/O controller is configured. For example:

- OFF: Controller power domain is OFF, "device enable" bit not set in power controller. This transition can also be made by the I/O PMU if auto-poweroff is supported.
- Sleep: Controller is "device enabled" but the controller interface has not been enabled/activated (AKA IDLE). This transition can also be made by the I/O PMU if auto-poweron is supported.
- LP: This needs to be determined based on the operating mode of the controller. This typically will be for high performance interfaces (such as MSPI).

### 4.3.4    Memory

The memory power modes are determined by PWRCTRL register configuration and the relative state of the various mains in the SoC. The primary state of all memories is dictated by the CPU. Once the CPU goes into Deep Sleep mode, all memories can go into retention or power down mode. If any outstanding transaction exists for a given memory target, the power controller will keep that respective memory device powered in Active mode or will wake the memory from retention or power down mode. Once Active, the outstanding transaction can be completed. There are hysteresis counters to keep memories powered up for longer periods of time to avoid thrashing.

The memory is configured by PWRCTRL_MEMPWREN and PWRCTRL_MEMRETCFG (for MRAM, CPU TCM and cache memories) as well as PWRCTRL_SSRAMPWREN and PWRCTRL_SSRAMRETCFG (for Shared SRAM memories) registers. If a memory is not enabled, the retention configuration is a don't care. Only if enabled, the retention configuration applies and will be enforced whenever the respective memories can enter lower power states.

## 4.4    Additional Information

Please refer to the PWRCTRL registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 5. Security

## 5.1 Features

The Apollo510 SoC provides secure boot and security functions leveraging Ambiq's SecureSPOT$^{TM}$ technology, and additionally supports the Arm Platform Security Architecture (PSA) and is TrustZone$^{®}$ compliant.

Key security features include:

- TrustZone$^{®}$ technology (ArmV8.1-M)
- Physically Unclonable Function (PUF)
- Tamper resistive fuses and key storage bank
- Secure Boot
- Secure Over-the-Air (OTA) Updates
- Secure Wired Updates
- Secure Key Storage
- Key revocation
- Secure debug
- Secure lifecycle management
- Crypto Subsystem (See "Crypto Subsystem" on page 54 for features and details.)

Please refer to the Security and Crypto registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

## 5.2 Functional Overview

The Apollo510 SoC enables the security extension on the Cortex-M55 (TrustZone-M) and protection of a subset of memories and peripherals. This enables the device to incorporate Trusted Execution Environment (TEE) into its software architecture. TrustZone technology creates a new processor execution domain, the secure domain.

In addition, there are Ambiq private resources that are not shared with customers which are held in the Ambiq Private Domain (APD), an extension of the secure domain.

**Table 6: Security Domain Partitioning**

| Mode | Nonsecure | Secure | |
|---|---|---|---|
| | | OEM Accessible | Ambiq Private Domain (APD) |
| Normal (non-privileged) | User Threads, Normal Applications, Normal Middleware | Secure Services, Secure Threads, Secure Applications | |
| Privileged | RTOS, Drivers, Privileged Threads, HAL, Privileged Middleware | Secure Partition Manager | BootRom, Helper Functions, Ambiq-Private Keys, Trims |

TrustZone technology does not provide support for the Ambiq Private Domain, so the Apollo510 SoC employs a variety of mechanisms to isolate the APD resources even from the secure domain.

## 5.3 TrustZone-M Basics

Arm TrustZone technology enables the system and the software to be partitioned into Secure and Normal worlds whereby secure software can access both secure and nonsecure memories and resources, while normal software can only access nonsecure memories and resources. These security states are orthogonal to the existing Thread and Handler modes, enabling both a Thread and a Handler mode in both

secure and nonsecure states.TrustZone-M is memory map-based where the instruction address determines the security state of the processor. Memory and other critical resources designated as secure can only be accessed when the core is executing in secure state. There are two separate MPUs for the secure and nonsecure world.

Memory can have the following security attributes:

- Nonsecure (NS)
- Secure (S)
- Nonsecure Callable (NSC)

The security attributes of memory are configured using the following constructs:

- Implementation Defined Attribution Unit (IDAU) – Fixed in implementation
- Security Attribution Unit (SAU) – Classifies memory regions as S/NS/NSC

Interrupt handling is performed by classifying interrupts as Secure or Nonsecure in the NVIC.

### 5.3.1    *TrustZone-M in the Apollo5 Family SoCs*

TrustZone technology is supported in the Apollo5 family's Cortex-M55 core whereby security attributes of the instruction address determine the security state of the processor. Outside of the core TrustZone support is limited in that all other mains (DMA, GPU, DC) during run-time are treated as nonsecure. Because of this, secure memory cannot be accessed by other mains and must be configured separately. As mentioned above, secure memory is configurable using SAU.

The BootROM helper function region should be classified as secure, while portions of ITCM, DTCM and SSRAM can be classified as secure. Protecting secure memory from external nonsecure access is accomplished by two means:

- TCM Gate Units (TGU) for ITCM and DTCM
- SSRAM Protection (SSRAMPROT) for SSRAM

Secure Peripherals on the APB are configurable using SAU & NVIC. Recommended modules to configure as Secure Peripherals are the following:

- Watchdog
- PWRCTRL
- MCUCTRL
- RESETGEN
- STIMER
- RTC
- CLOCKGEN

System Security Control (SSC) registers are used for global control/locking of peripheral and memory registers.

In the boot flow, Secure Boot (SBR+SBL) always runs in Secure Execution Mode. The SBR/SBL transitions to the Secure Firmware Image, which contains the Customer Firmware Entry. It too is always in Secure Execution Mode. This portion of firmware is responsible for system partitioning into secure/ nonsecure domains by using SAU, TGU, SSRAMPROT and NVIC configurations. The flow eventually transitions to the nonsecure Firmware Image.

## 5.4    Physically Unclonable Function (PUF)

The Apollo510 SoC incorporates a PUF that can be utilized by software and is used internally to secure the OTP memory, providing enhanced protection of assets. See "OTP" on page 40 for more information about PUF features.

## 5.5    Secure Boot

The Secure Boot feature on the Apollo510 SoC provides a secure foundation for customer firmware/ services. The secure boot loader provides authentication, decryption and integrity validation for all firmware upon installation and boot/reset. Secure boot loader provides firmware recovery and OTA update support.

Secure Boot policy can be used to direct the secure boot loader based on the customer security requirements.

A high level flow diagram of the Secure Boot process is illustrated below in Figure 6. See the "Apollo5 Security Guide" [To Be Supplied] for more details.
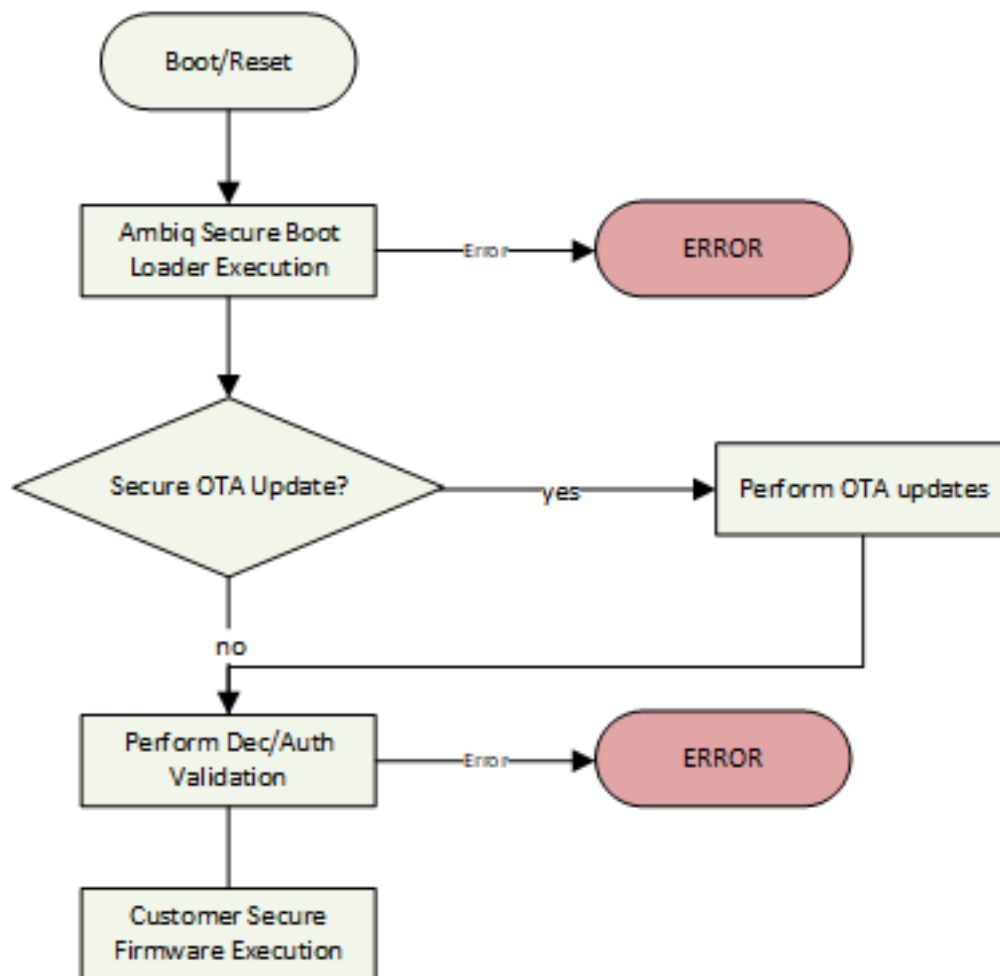


**Figure 6. Secure Boot Process Flow Diagram**

## 5.6    Secure OTA

The Apollo510 SoC supports secure OTA leveraging the Ambiq secure boot loader. Customers can update any firmware component securely as directed via the security policy configuration. The basic Secure OTA flow is shown in Figure 7.
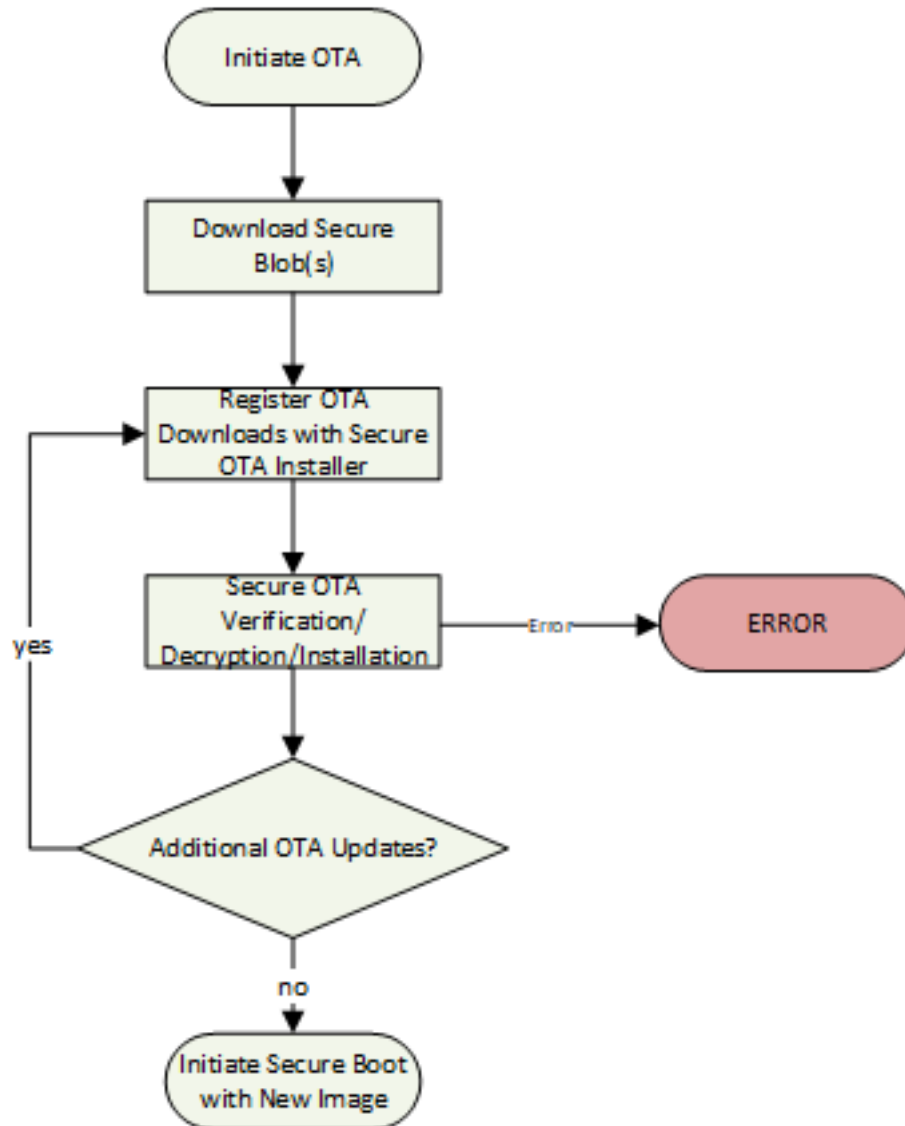
**Figure 7. Secure OTA Flow Diagram**

## 5.7    Secure Key Storage

Key material is managed by hardware and exposed to software via security APIs. The keys are stored securely in OTP memory and are never directly accessible to software. Certain key material is used/ accessible only during certain lifecycle state of the device. These are mainly used for provisioning of the device.

### 5.7.1    Software Keys

Additionally, there are "Software Keys" available to use in OTP (INFOC) that can be provisioned by OEM. These keys can be configured to be accessible only to privileged software (e.g., bootloader).

## 5.8    Secure Lifecycle States

The Apollo510 SoC supports the following lifecycle states:

- Chip Manufacturer (CM)
- Device Manufacturer (DM)
- Secure
- RMA

The lifecycles are managed by hardware and OTP.

## 5.9    Secure Debug

The debug sub-system has been enhanced in the Apollo510 SoC to provide separate capabilities for secure and nonsecure debug.  If only nonsecure debug is enabled, then no secure resources may be accessed, and no secure code may be stepped into, debugged, or traced.

## 5.10 Crypto Subsystem
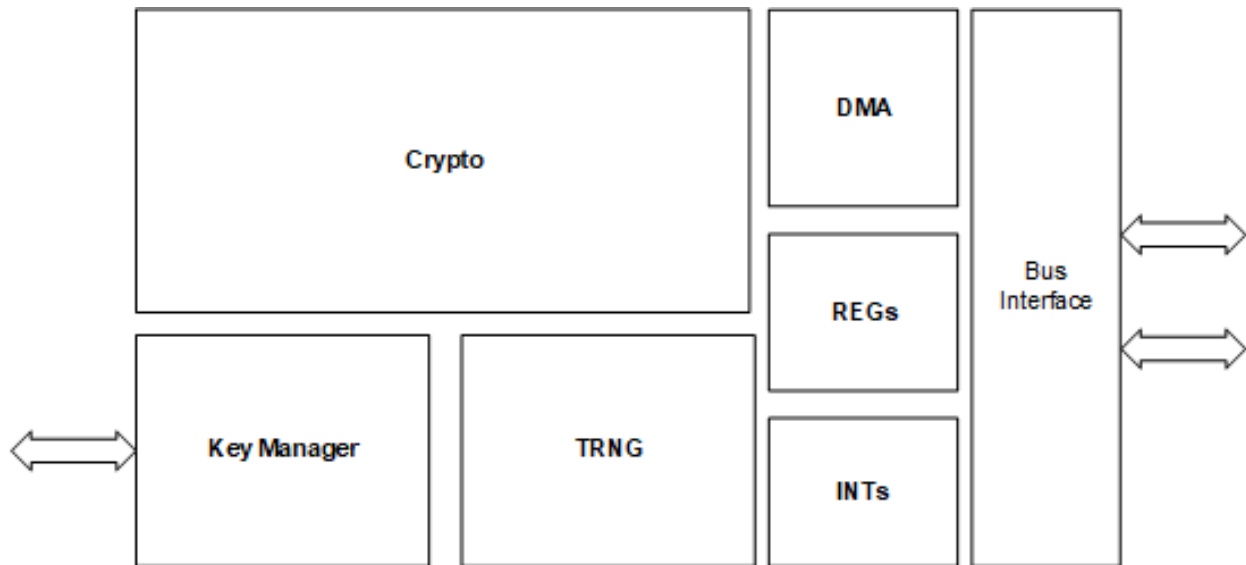


**Figure 8. Crypto Subsystem Block Diagram**

The Apollo510 SoC's Crypto Subsystem provides the cryptographic acceleration and isolation required to support the Apollo5 family security model. These services are managed by software to support private and public-side cryptographic functions, and includes features shown in Figure 8 and listed below.

- Cryptographic acceleration for the protection of data-in-transit and data-at-rest.
- Protection of various assets belonging to the chip manufacturer (ICV) or device manufacturer (OEM), service operators providing services over the target device and the device. These asset protection features include:
  - Image verification at boot/during runtime
  - Authenticated debug
  - Random number generation
  - Security lifecycle state management
  - Asset Provisioning

The following standard specifications are supported:

- FIPS Publication 186-4: Digital Signature Standard (DSS), July 2013, compliant with sections 5.1, 6.2, 6.3, 6.4, B.1.2, B.2.2, B.3.6, B.4.2, C.3.1, C.3.3, C.3.5, C.9, and D.1.2.
- FIPS Publication 197: Advanced Encryption Standard, support only 128-bit and 256-bit keys.
- NIST SP 800-38A: Recommendation for Block Cipher Modes of Operation: Methods and Techniques, compliant with sections 6.1, 6.2, 6.4, and 6.5.
- NIST SP 800-38B: Recommendation for Block Cipher Modes of Operation: the CMAC Mode for Authentication
- NIST SP 800-108: Recommendation for Key Derivation Using Pseudorandom Functions, compliant with section 5.1.
- Standards for Efficient Cryptography Group (SECG): SEC1 Elliptic Curve Cryptography, 2000, compliant with sections 2.1.1, 2.2.1, 3.1.1, 3.2, 3.3.1, 3.6.1, 4, and 6.1.

### 5.10.1   Crypto Acceleration

The CryptoCell-312 crypto accelerator is available to both the secure and the nonsecure domains. The following cryptographic features are supported:

- AES (128, 192, 256 bit)
  - ECB, CBC, CTR, OFB
  - CMAC, CBC-MAC, AES-CCM, AES_GCM
- AES Key Wrapping
- CRC32 with crypto acceleration
- External storage inline encryption/decryption with crypto acceleration
- Diffie-Hellman (1024, 2048, 3072 bit)
  - ANSI X9.42-2003: Public Key Cryptography for the Financial Services Industry: Agreement of Symmetric Keys Using Discrete Logarithm Cryptography.
  - Public-Key Cryptography Standards (PKCS) #3: Diffie-Hellman Key Agreement Standard.
- ECC Key Generation (NIST and 25519 curves)
- ECIES
- ECDSA
- ECDH
- SHA1/SHA224/SHA256
- HKDF
- KDF
  - NIST SP 800-108: Recommendation for Key Derivation Using Pseudorandom Functions
- RSA PKCS#1 (2048, 3072, 4096b)
  - Public-Key Cryptography Standards (PKCS) #1 v2.1: RSA Cryptography Specifications
  - Public-Key Cryptography Standards (PKCS) #1 v1.5: RSA Encryption
- RSA Key Generation
- True Random Number Generator (TRNG)
  - BSI AIS-31: Functionality Classes and Evaluation Methodology for True Random Number Generators.
  - NIST SP 800-90B: Recommendation for the Entropy Sources Used for Random Bit Generation.

See more detailed information at the CryptoCell-312 page of the Arm Developer site.

## 5.11   Additional Information

Please refer to the Security and Crypto registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 6. Reset Generator (RSTGEN)
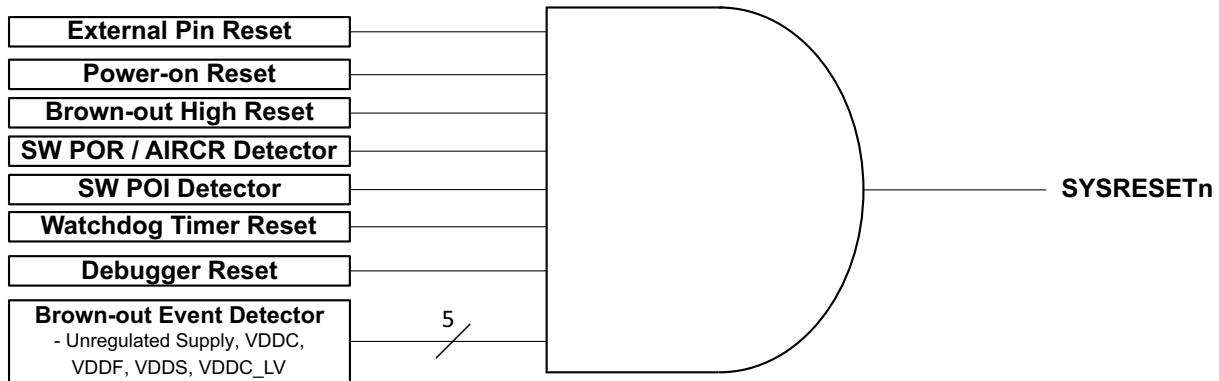


**Figure 9. Reset Generator Module Block Diagram**

## 6.1   Features

There are various resets that can be initiated in the Apollo510 SoC. Depending on the level of reset, certain functions may get cleared or may persist values or executing across specific reset assertions. In all cases, the source of a reset is captured in RSTSTAT register.

The Reset Generator Module (RSTGEN) supports reset sources shown in Figure 9 and listed below.

- External reset pin (RSTn)
- Power-on event (POA)
- Brown-out event (BODL)
- Software requests
- Debugger request
- System request (AIRCR SYSRESETREQ from the Cortex-M55)
- Watchdog expiration

## 6.2  Functional Overview

The Reset Generator Module (RSTGEN) monitors a variety of reset signals and asserts the active low system reset (SYSRESETn) accordingly. A reset causes the entire system to be re-initialized, and the cause of the most recent reset is indicated by the STAT register.

There are four levels of reset on the Apollo510 SoC:

- SYSRESETn System Software Reset
- Software POR Reset
- Software POI Reset
- Power-on Analog (POA) Reset

SYSRESETn is triggered by AIRCR.SYSRESETREQ, a Cortex-M55 warm reset.

Most resets trigger a POR reset, which is a shallow reset. This includes a WatchDog reset, a pin (RSTn) reset and a Cortex-M55 cold reset.

A POI reset is the deepest digital reset affecting hardware subsystems.

POA is the very deepest reset, digital and analog, and is only triggered by voltage going below POA voltage.

The effect of each of these reset levels on SoC modules and functionality is summarized in the following table and described in more detail in the following sections.

**Table 7: Reset Levels and Affected Components**

| Reset Level | CPU, GPIO and Peripherals (Except XT, LFRC, RTC and STIMER) | Debug Subsystem | CLKGEN (XT, LFRC, etc.), Reloads INFO Settings (Trims, Options and Security) | RTC and STIMER |
|---|---|---|---|---|
| SYSRESETn | x | | | |
| POR[1] | x | x | | |
| POI[2] | x | x | x | |
| POA (Power-on Reset) | x | x | x | x |

1. Doesn't affect CLKGEN except for setting the CPU clock to 96 MHz.
2. Reloads CLKGEN_OCTRL_OSEL bit from trim and resets CLKGEN features such as registers, CPU clock, HFADJ and HF2ADJ - everything else is reset only by POA.

---

ERRATUM NOTICE

All the power domains can be trimmed to be powered on after a SWPOI reset. In the failure case, some of the power domains might not power up as expected after a SWPOI reset. This is a hardware startup issue, which could result in non-functional peripherals/memories.

See "ERR026: PWRCTRL: Some power domains might not power up as expected after a SWPOI reset" in the *Apollo510 SoC Errata List*.

---

## 6.3    External Reset Pin

The active-low RSTn pin can be used to generate a reset using an off-chip component (e.g., a push-button). An internal pull-up resistor in the RSTn pad enables optional floating of the RSTn pin, and a de-bounce circuit ensures that bounce glitches on RSTn does not cause unintentional resets. The RSTn pin is

not maskable. An internal pull-down device will be active during a brownout event pulling the RSTn pin low. See Figure 10.
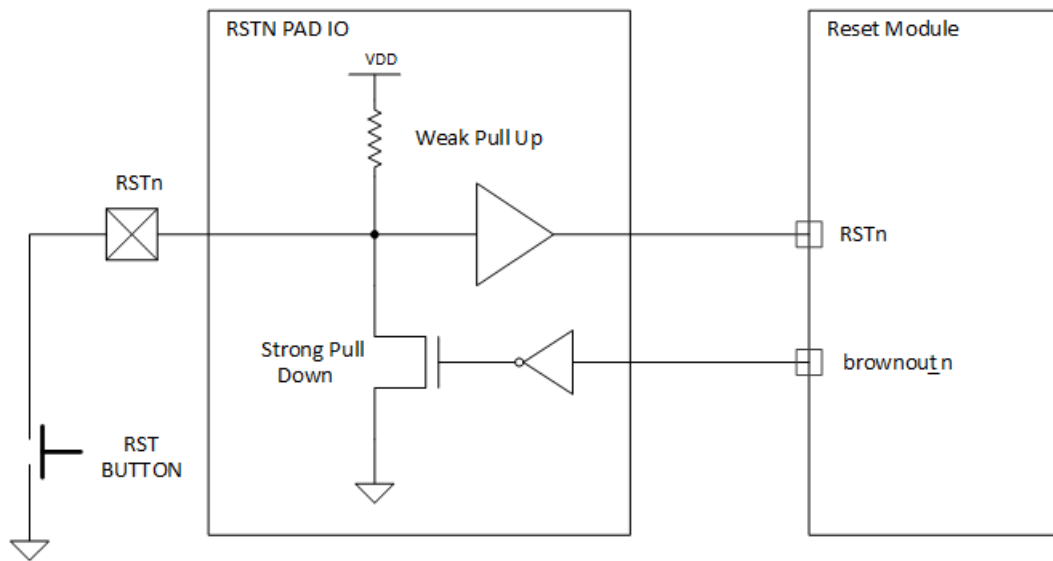


**Figure 10. Block Diagram of Reset Pin Circuitry**

## 6.4 Power-on Event

An integrated power-on detector monitors the supply voltage and keeps SYSRESETn asserted while VDD is below the rising power-on voltage, $V_{POR+}$. When VDD rises above $V_{POR}$ at initial power on, the reset module will initialize the low power analog circuitry followed by de-assertion of SYSRESETn, and normal operation proceeds. SYSRESETn is re-asserted as soon as VDD falls below the falling power-on voltage, $V_{POR-}$. The power-on reset signal, PORn, is not maskable.

## 6.5 Brown-out Events

An integrated brown-out detector monitors the primary supply voltage and causes an automatic and non-configurable reset when the voltage has fallen below the low brownout threshold (BODL).

In addition, there are individual brownout detector monitors integrated within the core/memory which cause separate/maskable reset assertions when the voltage falls below a critical level for the respective voltage rails - VDDC, VDDC_LV, VDDS or VDDF. In the event any of the supply voltages fall below its corresponding core/memory threshold if enabled, the reset module will initiate a system reset, enabling the RSTn pull-down and driving the reset pin low. The occurrence of a brownout reset will be reflected by the setting of the associated bit in the RSTGEN's INTSTAT Register after reset

In the event of a brownout detection, the following functionality is maintained until a power down detection occurs.

- All RTC registers retain state.
- RTC and STIMER counters continue operation from 32 kHz XTAL or from LFRC (if below BODL). If clock sources stop oscillating at very low voltage, the RTC and STIMER will continue to maintain state.
- Clock configuration registers retain state.

ERRATUM NOTICE

The brown-out reset status bit, RSTGEN_STAT_BORSTAT, may inadvertently get set due to the brown-out signals powering up inconsistently during system power-up. Whether the status bit gets set is influenced by the voltage levels of the power supplies (VDDH/VDDP/VDDA) and the ramp rate of these supplies.

See "ERR027: RSTGEN: Brown-out reset status bit may get set inadvertently during power-up" in the *Apollo510 SoC Errata List*.

## 6.6    Software Reset

The SYSRESETn reset may be generated via software using the Application Interrupt and Reset Control Register (AIRCR) defined in the Cortex-M55. For additional information on the AIRCR, see the Arm document titled "Cortex-M55 Devices Generic User Guide." The software reset request is non-maskable.

A second source for the identical software reset functionality is made available through the SWPOR register in the RSTGEN peripheral module. This reset causes a reset to all blocks except for registers in CLKGEN, RTC, STIMER and the power management unit. Most resets trigger a POR reset, which is a shallow reset. POR doesn't affect CLKGEN except that it sets the CPU clock to 96 MHz.

A third source of reset, which is the deepest digital reset affecting hardware subsystems, is made available through the SWPOI register. Writing the key value (0x1B) to this register triggers a Software POI reset, which causes:

▪ INFO space settings to be reloaded from Flash.
▪ a reset of all blocks except for registers in the RTC and the STIMER.
▪ the reloading from trim of the OSEL bit to the CLKGEN_OCTRL register, which selects either the LFRC or the XTAL to be used as the RTC clock source.
▪ reset of the CLKGEN features such as registers, CPU clock, HFADJ and HF2ADJ - everything else is reset only by POA.

## 6.7    Watchdog Reset

The Watchdog Timer sub-module generates an interrupt if it has not been properly managed by software within a pre-defined time. The watchdog reset is maskable.

## 6.8    Additional Information

Please refer to the RSTGEN registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 7. Clock Generator (CLKGEN)
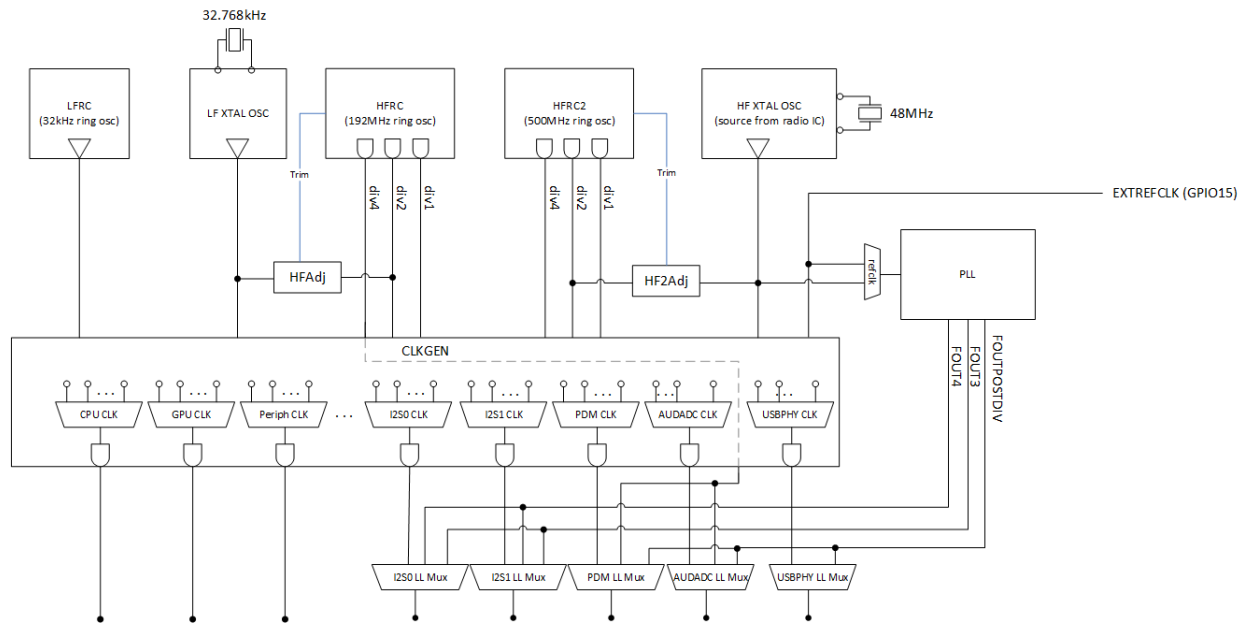


**Figure 11. Block Diagram for Clock Generator Module**

## 7.1 Features

The Apollo510 SoCs clock generation subsystem is responsible for generating all of the primary and derived clocks in the SoC.

- Independent frequency scaling for various SoC subsystems
- Ultra low power, low frequency clock generation with XTAL calibration
- Programmable I/O clock dividers
- High precision audio clock generation
- High precision / low jitter PLL for high-fidelity audio and high-speed USB

<div style="border:1px solid">

NOTE

When enabling a module which automatically starts clocking with a default clock source, or when changing the clock source for any enabled module, there is a required 30 µs settling time for the selected clock.

</div>

## 7.2 Functional Overview

A high-level view of the Clock Generator Module, which supplies all clocks required by the Apollo510 SoC, is shown in Figure 11. Note that the output clock frequencies from the clock sources are nominal values. Consult the Electrical Characteristics section for specified values.

### 7.2.1 Clock Generation Subsystem

The clock generation subsystem consists of the following sub-modules:

- High frequency crystal oscillator circuit (XTALHS)

- 2x High frequency RC oscillators (HFRC and HFRC2)
- Low frequency crystal oscillator circuit (XTAL)
- Low frequency RC oscillator (LFRC)
- High frequency PLL circuits
- SoC clock generation logic
- Audio subsystem clock generation logic

### 7.2.1.1 Low-Frequency Crystal Oscillator (XTAL)

The high accuracy XTAL Oscillator is tuned to an external 32.768 kHz crystal, and has a nominal frequency of 32.768 kHz. It is used when frequency accuracy is critically important. To minimize power consumption, the XTAL is only enabled when an internal module is using it.

It should be noted that the XTAL oscillator is also optional if the requirements of the design can tolerate the internal LFRC/HFRC oscillator specifications.

External load capacitors are required to tune the XTAL oscillator circuit. Refer to "Clocks/Oscillators" on page 243 for more guidance and specifications.

| NOTE |
| --- |
| The XTAL is highly sensitive to external leakage on the XI pin. Therefore it is recommended to minimize the components on XI and to use extremely low leakage load capacitors. |

The RTC clock source, either the LFRC Oscillator or the XTAL Oscillator, is selected via the CLKGEN_OCTRL_OSEL bit. If the XTAL Oscillator experiences a temporary failure and subsequently restarts, the Apollo510 SoC will switch back to the XTAL Oscillator.

A 32 kHz reference clock is output from the XTAL circuit to several GPIO pins on all packages.

### 7.2.1.2 High Speed Crystal Oscillator (XTALHS)

The high speed crystal sets the primary clock input for the on-chip audio subsystem. XO32M bypass accepts 0.9 Vpp clipped sine wave (e.g., TCXO output).

XTALHS can be used directly to clock audio (PDM, $I^2S$, AUDADC) or USB. In this case a 24.576 MHz or 22.579 MHz XTAL can be used for audio only, or similarly, a 24 or 48 MHz XTAL can be used directly for USB only. In addition, a bypass option allows XTALHS or a clock input on EXTREFCLK to be used as the clock for the SYSPLL which in turn can be used as the clock source for audio (PDM, $I^2S$, AUDADC) or USB.

A clock is generated from the XTALHS circuit to support external system clocking if needed via GPIO46. This output clock can be used in systems where an external radio or audio codec is needed and requires high quality clock source. This option is available in configurations where the Apollo510 SoC has a high frequency XTAL source attached. XTALHS support is dependent on product SKU. The XTALHS external clock source supports two output modes:

1. Attenuated output (0.9 V via resistive attenuation)
2. Square wave output at 1.8 V with maximum added phase noise of -150 dBc/Hz at an offset of 100 kHz.

### 7.2.1.3 Low-Frequency RC Oscillator (LFRC)

The low-power low-frequency RC block, LFRC, with a nominal frequency of 900 Hz, provides the low-frequency clocks for timers and other logic within the SoC. It is used when short-term frequency accuracy

is not important. It also supplies clocks for the SIMO buck regulator in low power mode as well as some basic state machines and is always enabled.

### 7.2.1.4   High Frequency RC Oscillator (HFRC)

The high-frequency RC oscillator (HFRC) provides all the primary clocks for the high-frequency digital processing blocks in the SoC except for audio, radio and HP mode clocks. These clocks are gated/ selected based on performance requirements. The digital clocks are isolated to avoid noise injection into the critical clocks for audio and radio communications. Additionally, the high-frequency digital clock is programmatically divided to generate the various I/O clocks in the system. All high-frequency clocks can be gated if not needed. The HFRC also supports a frequency-locked loop (FLL) circuit to ensure the HFRC oscillator locks to a specific frequency range to ensure high quality/low ppm output reference as needed to meet audio clock quality requirements. Although the HFRC can be calibrated to a variety of input reference clocks, the primary clock reference is the 32 kHz XTAL input.

The high-frequency HFRC Oscillator, with a nominal frequency of 96 MHz, is used to supply all high-frequency clocks in the Apollo510 SoC such as the processor clock for the Arm core, memories and many peripheral modules. Digital calibration is not supported for the HFRC, but its frequency may be automatically adjusted by the auto-adjustment function which is a combination of analog and digital operations.

The HFRC is enabled only when it is required by an internal module. When the Arm core goes into a sleep mode, the HFRC will be disabled unless another module is using it. If the Arm core goes into deep sleep mode, the HFRC will be powered down when it is not needed. When the HFRC is powered up, it will take a few microseconds for it to begin oscillating, and a few more microseconds before the output is completely stable. In order to prevent erroneous internal clocks from occurring, the internal clocks are gated until the HFRC is stable.

### 7.2.2   Secondary High Frequency RC Oscillator (HFRC2)

A secondary high-frequency RC block, HFRC2, provides the optional high-frequency clocks needed for the CPU and GPU in HP mode and optionally for MSPI high-frequency mode. The HFRC2 also supports an FLL circuit (via HFAdj2) to ensure the HFRC2 oscillator locks to a specific frequency range to ensure high quality/low ppm output reference as required for audio features. Although the HFRC2 can be calibrated to a variety of input reference clocks, the primary clock reference is the high-frequency XTAL input.

### 7.2.2.1   System PLL

The system PLL is used to generate the high-precision clocks needed for high-fidelity audio and to create the clocks needed for the USB PHY reference clock and the DSI. The System PLL has the following features:

▪ Wide output frequency range
▪ 24-bit fractional accuracy
▪ 2 post-dividers each with a selectable setting of 1 to 7
▪ 1 additional divide-by-6 or divide-by-8 post-divider option for audio clocks

---

ERRATUM NOTICE

The SYSPLL can be clocked by either the high-speed crystal oscillator (XTALHS) or an external reference clock (EXTREFCLK) through a glitch-less mux within the SYSPLL. If XTALHS had been enabled at any time after reset and then disabled, then the selection of the EXTREFCLK as the SYSPLL clock source cannot be made. If the XTALHS had been enabled and then disabled, it must be re-enabled to be able to transition the SYSPLL clock source from XTALHS (default setting) to the EXTREFCLK.

If XTALHS had not been enabled prior to selecting the clock source for the SYSPLL or is not present in the system, then EXTREFCLK can be selected as the SYSPLL clock source directly (by setting the MCUCTRL_PLLCTL0_FREFSEL bit to EXTREFCLK).

See "ERR030: SYSPLL: Special case where the SYSPLL cannot select the EXTREFCLK as its clock source" in the *Apollo510 SoC Errata List*.

---

### 7.2.2.2  External Reference Clock Input

An external reference clock input, EXTREFCLK, can be used to provide an alternate reference clock source as needed to synchronize various clock generators within the SoC (such as the system PLL, DSI and/or USB PLL) to avoid interference or spur injection when in the Bluetooth enabled configurations.

---

## 7.3  Clock Sources for Apollo510 SoC

Clock sources used within the SoC are as shown in Table 8. All clock sources except for the external reference clock (EXTREFCLK) can be gated.

**Table 8: Requirements for Clock Sources**

| Clock Name | Frequency | Tolerance | Duty Cycle (% High/ Low) | Additional Requirements / Comments |
|---|---|---|---|---|
| HFRC_96MHz | 96 MHz | ±5% | 50/50 | |
| HFRC_192MHz | 192 MHz | ±5% | 30/70 | |
| HFRC48 | 48 MHz | | | HFRC48 is a dedicated clock from the HFRC block (HFRC192_DIV4); selected by MCUCTRL_PLLMUXC-TRL field for the PDM or AUDADC modules only. |
| LFRC | Nominal 900 Hz | | 50/50 | |
| HFRC2_125MHz | 125 MHz | ±5% | 50/50 | |
| HFRC2_250MHz | 250 MHz | ±5% | 45/55 | |
| XTALHS[1] | 32 MHz, 24 MHz and other frequencies | | 50/50 | Startup time < 1 ms maximum, < 0.5 ms ideal. |
| XT_32KHz[2] | 32.768 kHz | ±50 ppm[3] | 50/50 | |
| EXTREFCLK | Variable depending on the external clock source | | | |
| PLLCLK | 1.22 MHz to 960 MHz | | 50/50 | PLLCLK output offered as clock for $I^2S$ and other audio modules in HP mode, and for the USBPHY. |

1. See Clocks/Oscillator section of Electrical Characteristics for the specifications and range of frequencies supported by the high frequency XTAL.
2. See Clocks/Oscillator section of Electrical Characteristics for the specifications of the 32 kHz XTAL.
3. Depends on the accuracy of the external 32 kHz crystal. See Clocks/Oscillator section of Electrical Characteristics for crystal requirements.

Clocks used by the CPU, SRAM and MRAM are as listed in Table 9.

**Table 9: CPU, Memory and Storage Clocks**

| Block | Clock | LP | | HP | | Additional Requirements / Comments |
|---|---|---|---|---|---|---|
| | | Frequency | Source | Frequency | Source | |
| CPU | CLKIN | 96 MHz | HFRC | 250 MHz | HFRC2 | |
| | IWICCLK | 96 MHz | HFRC | 250 MHz | HFRC2 | Synchronous with CLKIN, same frequency when in use.<br>Gated during sleep. |
| | PMUCLK | 96 MHz | HFRC | 250 MHz | HFRC2 | Synchronous with CLKIN, same frequency when in use.<br>Gated when not in use. |
| | STCLK | EXT: 3 MHz | HFRC_DIV32 | EXT: 3 MHz | HFRC_DIV32 | Asynchronous clock.<br>STCLK has 2 clock sources - EXT and INT. |
| | | INT: 96 MHz | HFRC | INT: 250 MHz | HFRC2 | |
| | DBGCLK | See Debug section in Table 10 | | | | Synchronous with CLKIN,<br>Gated when not in use. |
| | TRACECLKIN | See Debug section in Table 10 | | | | Asynchronous clock.<br>Gated.<br>Arm recommends that this match the maximum frequency of the CPU clock. |
| | SWCLKTCK | See Debug section in Table 10 | | | | Input clock on SWJ DAP interface. |
| Shared SRAM/ Fabric | AXICLK | 96 MHz | HFRC | | | |
| MRAM | Bus Clock | 96 MHz | HFRC | | | |

Clock sources and divider options used by the modules and sub-systems on the SoC are as listed in Table 10.

**Table 10: Module Clocks and Dividers**

| Module / Subsystem | Clock Node | Source / Frequency Selections | Additional Requirements / Comments |
|---|---|---|---|
| USBPHY | Ref Clock 24 MHz | • HFRC_48MHz with HFAdj<br>• HFRC_24MHz with HFAdj<br>• EXTREFCLK<br>• EXTREFCLK_DIV2<br>• XTALHS<br>• XTALHS_DIV2<br>• PLLCLK | PLLCLK (FOUTPOSTDIV only) selected by MCUCTRL_PLLMUXCTRL register. All others selected by USB_CLKCTRL_PHYREFCLKSEL field.<br>Quiesce clock option available. |
| | Bus Clock | • HFRC_96MHz | |
| USB Controller | Bus Clock | • HFRC_96MHz | |
| DSIPHY | Ref Clock | • HFRC_24MHz<br>• HFRC_12MHz<br>• XTALHS<br>• XTALHS_DIV2<br>• EXTREFCLK<br>• EXTREFCLK_DIV2 | |
| | Bus Clock | • HFRC_96MHz | |
| MSPI | MSPI0, MSPI3: HFRC, HFRC2<br><br>MSPI1, MSPI2: HFRC | MSPI0, MSPI3 Sources:<br>• HFRC_96MHz<br>• HFRC_192MHz<br>• HFRC2_125MHz<br>• HFRC2_250MHz<br><br>MSPI1, MSPI2 Sources:<br>• HFRC_96MHz<br>• HFRC_192MHz | Clock source selected by CLKGEN_MSPIIOCLKCTRL register.<br><br>Each MSPI further divided by 1, 2, 3, 4, 6, 8, 12, 16, 24 or 32 by MSPI_DEV0CFG_CLKDIV0. |
| | Bus Clock | • HFRC_96MHz | |
| SDIO0/1 | Ref Clock | • HFRC (up to 96MHz) | Source synchronous.<br>xin_clk is fixed to HFRC_96MHz in the SDK software. HFRC_48MHz and HFRC_24MHz are in Apollo510 MCUCTRL register as options. |
| | Bus Clock | • HFRC_96MHz | |
| Crypto | Core Clock | • HFRC_96MHz | |
| GFX | Core Clock | CLKCTRL_GFXCORECLKSEL:<br>• HFRC_96MHz (LP)<br>• HFRC2_250MHz (HP) | 250 MHz HP mode is primary/default frequency and recommended as GPU HP clock. |
| | Bus Clock | • HFRC_96MHz (LP)<br>• HFRC2_250MHz (HP) | |

**Table 10: Module Clocks and Dividers**

| Module / Subsystem | Clock Node | Source / Frequency Selections | Additional Requirements / Comments |
|---|---|---|---|
| Display Controller | Bus Clock | • HFRC_96MHz | |
| | Pixel Clock | • HFRC (12, 24, 48, 96, 192 MHz)<br>• DPHY (TxByteClkHS, 96 MHz max) | In DC+DSI 1-lane mode, DPHYPLL is primary DC clock source.<br>In DC+DSI 2-lane mode, HFRC_96MHz is primary DC clock source.<br>In DC only mode, HFRC_96MHz and HFRC_48MHz are primary DC clock sources. |
| Audio Subsystem | PDM I/O CLK 400-3200 kHz | • PDM MCLK | |
| | PDM MCLK | • HFRC2 31.25MHz (250MHz_DIV8) with HFAdj2<br>• XTALHS<br>• XTALHS_DIV2<br>• XTALHS_DIV4<br>• HFRC_96MHz_DIV4 with HFAdj<br>• EXTREFCLK<br>• PLLCLK<br>• HFRC48 | HFRC is the primary clock source. HFRC48 is a dedicated clock from the HFRC (HFRC192_DIV4).<br>PLLCLK (FOUTPOSTDIV only)<br>PLLCLK and HFRC48 selected by MCUCTRL_PLLMUXCTRL register.<br>All others selected by PDM_CTRL_CLKSEL field.<br>Quiesce clock option available. |
| | PDM Bus Clock | • HFRC_96MHz | |
| | I$^2$S I/O SCLK: 128-3027 kHz | • I$^2$S SCLK | |
| | I$^2$S MCLK | • HFRC DIV1 to DIV128 in octave steps<br>• HFRC2 DIV8 to DIV1024 in octave steps<br>• EXTREFCLK DIV1 to DIV64 in octave steps<br>• PLLCLK: PLLFOUT3 (nominal 8 MHz) or PLLFOUT4 (nominal 6 MHz) | PLL output frequency is selected by MCUCTRL_PLLMUXCTRL register:<br>• PLLFOUT3 is FOUTPOSTDIV divided by 6, or nominal 8 MHz based on 48 MHz PLL clock ref.<br>• PLLFOUT4 is FOUTPOSTDIV divided by 8, or nominal 6 MHz based on 48 MHz PLL clock ref.<br>All others selected by I2S_CLKCFG_FSEL field. HFRC is the primary clock source.<br>Quiesce clock option available. |
| | I$^2$S NCO Clock | • HFRC2_31.25MHz with HFAdj2<br>• HFRC_48MHz with HFAdj<br>• XTALHS<br>• EXTREFCLK | |
| | I$^2$S Bus Clock | • HFRC_96MHz | |
| | AUDADC Clock | • HFRC_48MHz with HFAdj<br>• HFRC2_31.25MHz with HFAdj2<br>• XTALHS<br>• EXTREFCLK<br>• HFRC48<br>• PLLCLK | HFRC is the primary clock source. HFRC_48MHz is HFRC96_DIV2 and HFRC48 is a dedicated clock from the HFRC (HFRC192_DIV4).<br>PLLCLK option is FOUTPOSTDIV only.<br>PLLCLK and HFRC48 are selected by MCUCTRL_PLLMUXCTRL register.<br>All others selected by AUDADC_CFG_CLKSEL field.<br>Quiesce clock option available. |
| | AUDADC Bus Clock | Same as AUDADC. | |

**Table 10: Module Clocks and Dividers**

| Module / Subsystem | Clock Node | Source / Frequency Selections | Additional Requirements / Comments |
|---|---|---|---|
| WDT | Ref Clock | • LFRC/8, LFRC/64, LFRC/1024, LFRC/16384<br>• XTALHS, XTALHS/2 | LFRC source clock is nominally 900 Hz. |
| | Bus Clock | • HFRC_48MHz | |
| Timers | Various Source Clocks | • HFRC_DIV4, HFRC_DIV16, HFRC_DIV64, HFRC_DIV256, HFRC_DIV1024, HFRC_DIV4K<br>• LFRC, LFRC_DIV2, LFRC_DIV32, LFRC_DIV1K<br>• 32 kHz XT: XT (uncalibrated), XT_DIV2, XT_DIV4, XT_DIV8, XT_DIV16, XT_DIV32, XT_DIV64, XT_DIV128<br>• RTC_100HZ<br>• TMR00, TMR01 - TMR150, TMR151<br>• GPIO00 - GPIO127<br>• HFRC2: 16/8/4/2/1 MHz, 512 kHz<br>• XTALHS, XTALHS_DIV2, XTALHS_DIV4 | LFRC source clock is nominally 900 Hz. |
| | Bus Clock | • HFRC_48MHz | |
| IOS | IOS I/O Clock 0.75 - 48 MHz | • External | Sourced from external manager |
| | Bus Clock | • HFRC_96MHz | |
| IOM | IOM I/O Clock | • HFRC_48MHZ<br>• HFRC_24MHZ<br>• HFRC_12MHZ<br>• HFRC_6MHZ<br>• HFRC_3MHZ<br>• HFRC_1p5MHZ<br>• HFRC_750KHz<br>• HFRC_375KHz | Quiesce clock option available. |
| | Bus Clock | • HFRC_96MHz | |
| UART | UART HF Clock | With HFAdj:<br>• HFRC_48MHZ<br>• HFRC_24MHZ<br>• HFRC_12MHZ<br>• HFRC_6MHZ<br>• HFRC_3MHZ<br><br>• PLLCLK | |
| | Bus Clock | • HFRC_96MHz | |
| GPIO | Bus Clock | • HFRC_48MHz | |
| PWRCTRL | Bus Clock | • HFRC_48MHz | |

**Table 10: Module Clocks and Dividers**

| Module / Subsystem | Clock Node | Source / Frequency Selections | Additional Requirements / Comments |
|---|---|---|---|
| Debug | TPIU Clock | • HFRC2_250MHz<br>• HFRC2_125MHz (DIV2)<br>• HFRC_96MHz | Gated when not in use. |
| | Debug Subsystem Clock | • HFRC2_250MHz<br>• HFRC_96MHz | Gated when not in use.<br>DBGCLK is synchronous with the CPU clock. |
| | SWD Clock 100 MHz | • External (Pad Input) | SWJ input clock on the DAP. Tools support up to 100 MHz interface. |
| ADC | ADC Clock | • HFRC 48 MHz (HFRC_96MHz_DIV2)<br>• HFRC 24 MHz (HFRC_96MHz_DIV4)<br>• HFRC2 31.25 MHz (HFRC2_250MHz_DIV8) | |
| MCUCTRL | Bus Clock | • HFRC_24MHz | |
| RSTGEN | Bus Clock | • HFRC_24MHz | |

## 7.4    Additional Information

Please refer to the CLKGEN registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 8. Real Time Clock (RTC)
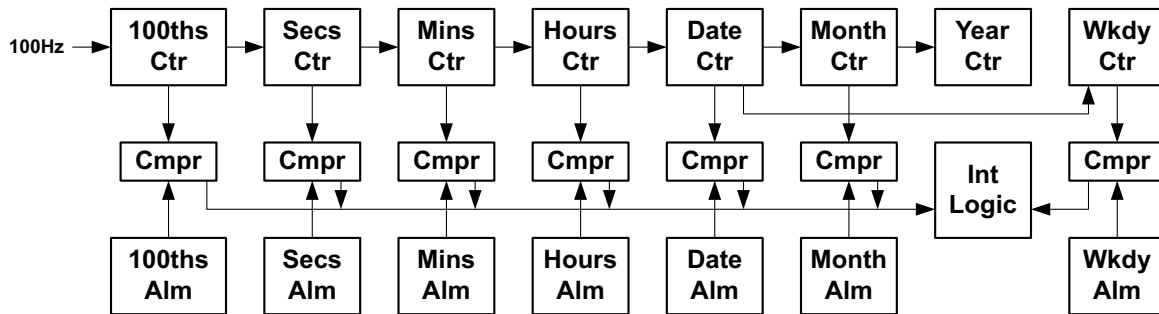


**Figure 12. Block diagram for the Real Time Clock Module**

## 8.1 Features

The Real Time Clock (RTC) Module, shown in Figure 12, provides an accurate means of maintaining real time. Key features are:

- 100$^{th}$ of a second resolution
- Time is measured for the years between 1900 and 2199
- Automatic leap year calculation
- Hours are specified in 24 hour mode
- Alarm precise to 1/100 second
- Alarm interval every 100$^{th}$ second, second, minute, hour, date, day of the week or month
- 100 Hz input clock taken from either the high accuracy XT Oscillator or the low power LFRC Oscillator

## 8.2 Functional Overview

Real time is held in a set of eight Calendar Counters, which hold the following counts of current units in BCD format:

- 1/100$^{th}$ of a second (RTC_CTRLOW_CTR100)
- Second (RTC_CTRLOW_CTRSEC)
- Minute (RTC_CTRLOW_CTRMIN)
- Hour (RTC_CTRLOW_CTRHR)
- Date (RTC_CTRUP_CTRDATE)
- Day of the week (RTC_CTRUP_CTRWKDY)
- Month (RTC_CTRUP_CTRMO)
- Year (RTC_CTRUP_CTRYR)

The timer chain which generates the 100 Hz clock is reset to 0 whenever any of the Calendar Counter Registers is written. Since unintentional modification of the calendar counters is a serious problem, the RTC_RTCCTL_WRTC bit must be set in order to write any of the counters. Software may stop the clock to the calendar counters by setting the RTC_RTCCTL_RSTOP bit.

The RTC includes special logic to help ensure that the calendar counters may be read reliably, i.e., that no rollover has occurred. Two 32-bit reads are required to read the complete set of counters.

There are seven alarm register fields in the ALMLOW and ALMUP registers which may be used to generate an alarm interrupt at a specific time. These fields correspond to the above listed time calendar counters except for year which does not have an associated alarm. When all selected counters match their corresponding alarm register, the ALM interrupt flag is set.

By setting the Century Enable Bit (RTC_CTRUP_CEB), the Century Bit (RTC_CTRUP_CB) indicates the current century. A value of 0 indicates the 21$^{st}$ century, and a value of 1 indicates the 20$^{th}$ or 22$^{nd}$ century.

The Weekday Counter is a 3-bit counter which counts up to 6 and then resets to 0. It is the responsibility of software to assign particular days of the week to each counter value.

---

ERRATUM NOTICE

When the RTC_CTRUP_CEB bit is set to enable the Century bit (CB) to change, the CB toggles whenever and as long as the Year field (CTRYR) is 99. This happens not just on rollover from 99 to 00 but toggles randomly, which is (essentially) every RTC clock.

See "ERR028: RTC: CB field value is unpredictable when year = 99 and CEB = 1" in the *Apollo510 SoC Errata List*.

---

## 8.3    Additional Information

Please refer to the RTC registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 9.    Counter/Timer (TIMER)
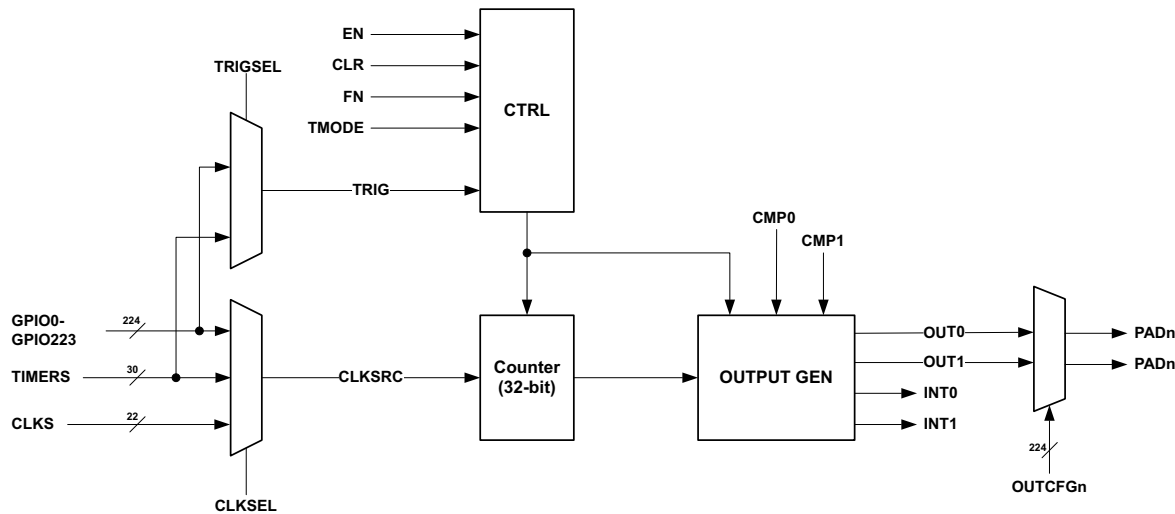


**Figure 13. TIMER Block Diagram**

## 9.1    Features

The Counter/Timer Module (TIMER) includes features shown in Figure 13 and listed below.

- Sixteen 32-bit binary up-counters used for simple waveform generation, interrupt sources, and counting applications.
- Each timer core has 2 interrupts, TMRnINT0/TMRnINT1 (Only for EDGE, UPCNT and PWM).
  - TMRnINT0 is generated when the value of the TIMER matches the TMRnCMP0 value.
  - TMRnINT1 is generated when the value of the TIMER matches the TMRnCMP1 value.
- For all modes CMP0 defines the end of a counter cycle and the timer will either stop or repeat. CMP1 is a secondary comparator.
- Each timer has two outputs which are controlled from CMP0 and CMP1 registers based on timer mode.
- Each timer is fully independent but can be linked by clocking one timer from another's output.
- Each timer offers several clock options including CLKGEN clocks, another timer output, or any GPIO input.
- Each timer has an interface to the GPIO module which allows any GPIO to be driven by any timer output and any GPIO to be used as a timer input.
- The start of a timer can optionally be triggered by a rising, falling or either edge of another timer output, an STIMER capture/compare event, or any GPIO input.
- Counter register value may be written directly.
- Either or both OUT0/OUT1 outputs of any timer can be inverted by using the POL0/POL1 bit.
- TMRnLMT register can be used to generate 1-255 repetitions of a waveform (0=unlimited).

## 9.2    Functional Overview

The Apollo510 SoC's Timer/Counter module includes sixteen Timer/Counters, one of which is shown in the Figure 13. This is in addition to a system timer as described in the System Timer chapter. Each Timer/Counter includes a very low power asynchronous 32-bit counter. Each Timer/Counter has external pin connections using any GPIO pads as outputs for each of the two comparators.

Timer functions include those listed in Table 11.

**Table 11: Timer Modes**

| Mode | Outputs | Description/Uses |
|------|---------|------------------|
| EDGE | OUT0 transitions when TIMERn = CMP0.<br>OUT1 transitions when TIMERn = CMP1 (if CMP1 < CMP0). | Edge generation (0x1):<br>TIMERn counts up from 0 to CMP0 and stops. A single edge is generated on OUT0 when TIMERn reaches CMP0.<br><br>If CMP1 < CMP0, then a single edge is also generated on OUT1 when TIMERn reaches CMP1.<br><br>If CMP0 < CMP1, then no edge is generated on OUT1 (because TIMERn never reaches CMP1).<br><br>TMRnLMT: Has no effect in this mode.<br><br>Trigger:  Timer does not start until trigger occurs (if enabled).  Subsequent triggers ignored. |
| UPCOUNT | OUT0 pulses for 1 clock cycle when TIMERn = CMP0; counter resets to 0; repeats per TMRnLMT setting.<br><br>OUT1 pulses for 1 clock cycle when TIMERn = CMP1 (if CMP1 < CMP0). | Repeatable Up-counter (0x2):<br>Counts up from zero to CMP0 and stops (TMRnLMT = 1), repeats N times (TMRnLMT = 2 - 255), or repeats indefinitely until TMRnEN is cleared (TMRnLMT = 0).<br><br>Timer outputs will be a pulse of one source clock period on the output when the TIMER reaches the associated CMP value.<br><br>If CMP1 < CMP0, then a single pulse of one source clock period is also generated on OUT1 when TIMERn reaches CMP1.<br><br>If CMP0 < CMP1, then no pulse is generated on OUT1 (because TIMERn never reaches CMP1).<br><br>Trigger: Timer does not start until trigger occurs (if enabled).  Subsequent triggers ignored. |
| PWM | OUT0 transitions when TIMERn = CMP1, then transitions again when TIMERn = CMP0; repeats per TMRnLMT setting.<br><br>OUT1 is the complement of OUT0. | Repeatable PWM (0x4):<br>Counts up from zero to CMP0 and stops (TMRnLMT=1), repeats N times (TMRnLMT = 2 - 255), or repeats indefinitely (TMRnLMT = 0).<br><br>CMP0 specifies the period and CMP1 specifies number of TIMERn counts for the initial phase of the output waveform.<br><br>Trigger: Timer does not start until trigger occurs (if enabled).  Subsequent triggers ignored. |

**Table 11: Timer Modes**

| Mode | Outputs | Description/Uses |
|---|---|---|
| SINGLEPATTERN | For TMRnLMT < 32 and for TIMERn count = 0 to TMRnLMT:<br>OUT0 = CMP0[TIMERn].<br>OUT1 = CMP1[TIMERn].<br><br>For TMRnLMT between 32 and 63 (maximum 64-bit pattern consisting of CMP1:CMP0) and for TIMERn count = 0 to TMRnLMT:<br>OUT0 = OUT1 = CMP1:CMP0[TIMERn].<br>OUT1 can be configured to be the complement of OUT0. | Single-run Pattern Generation (0xC):<br>CMP0 and CMP1 are bit-shifted to form the output pattern on OUT0 and OUT1, or concatenation of CMP1:CMP0 is bit shifted to form the output pattern on both OUT0 and OUT1.<br><br>The TIMER count is an up-counter that indexes into the bits of CMP0 (TIMER values from 1-31) and CMP1 (TIMER values 32-63). TMRnLMT value defines the length of the pattern minus 1 (0 = 1-bit pattern, 63 = 64-bit pattern).<br><br>OUT1 is the same as OUT0, but can be inverted for motor control applications.<br><br>INT0 is triggered when all TMRnLMT+1 bits have been streamed out.<br><br>INT1 is triggered only when TMRnLMT is set to 31 and all 32 bits of CMP0 have been streamed out.<br><br>Trigger: Timer does not start until trigger occurs (if enabled). After clearing and reconfiguring the timer for this mode, a subsequent trigger restarts the pattern replay. |
| REPEATPATTERN | For TMRnLMT < 32 and for TIMERn count = 0 to TMRnLMT:<br>OUT0 = CMP0[TIMERn].<br>OUT1 = CMP1[TIMERn].<br><br>For TMRnLMT between 32 and 63 (maximum 64-bit pattern consisting of CMP1:CMP0) and for TIMERn count = 0 to TMRnLMT:<br>OUT0 = OUT1 = CMP1:CMP0[TIMERn].<br>OUT1 can be configured to be the complement of OUT0. | Repeated Pattern Generation (0xD):<br>CMP0 and CMP1 are bit-shifted to form the output pattern on OUT0 and OUT1 as described in SINGLEPATTERN mode with the exception that the timer repeats the pattern after TMRnLMT+1 bits have been streamed out, resetting TMRnLMT back to 0 after each iteration.<br><br>Since TMRnLMT is used for the repeat pattern length, there is no option to repeat the pattern N times. The iterations repeat indefinitely until TIMERn is disabled. |

---

**NOTE**

CMP0/CMP1 values should not be changed when the TIMER is running or else it will corrupt the counter.

Software should assert TMRnCLR each time before asserting TMRnEN. if this is not done, stale values will remain.

---

## 9.3   Additional Information

Please refer to the TIMER registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

---

# 10. System Timer (STIMER)
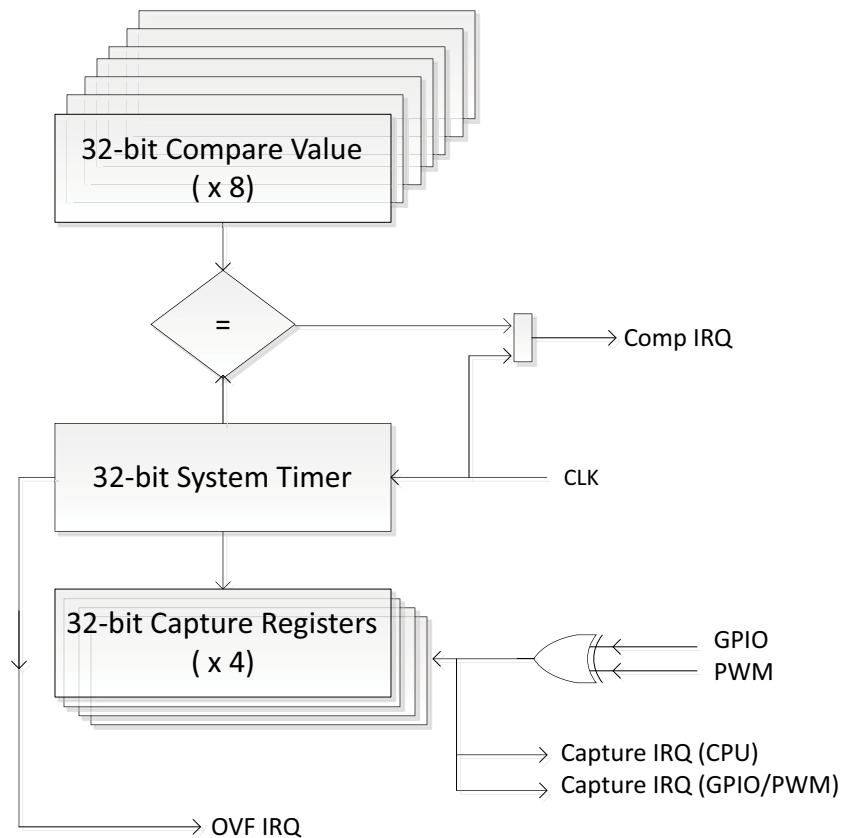


**Figure 14. STIMER Block Diagram**

## 10.1 Features

The System Timer (STIMER) includes features shown in Figure 14 and listed below.

- 32-bit binary counter used for RTOS scheduling decisions
- Eight 32-bit compare and interrupt registers to facilitate light weight scheduling (designs without RTOS)
- Accurate scheduling of comparator interrupts
- Only offsets from "NOW" written to comparator registers
- Maintains real time epoch for applications
- Overflow interrupt to allow firmware to keep the extended part (more than 32-bits) of real time epoch
- Time stamping hardware for multiple sensor streams (4 capture registers)
- Firmware handling of odd calculations such as Leap Second and events like surprise/legislated changes to the daylight savings time transition dates
- Firmware handling of 1024 versus 1000 scaling of real time conversions
- Only reset by POA (Power On Analog - system cold reset) so that it retains time across all POI and POR (system warm reset) events except full power cycles
- Contains three 32-bit NVRAM registers that are only reset by POA to maintain real time offset from epoch
- Programmable external GPIO trigger and/or PWM trigger on capture (required for sensor synchronization)

## 10.2 Functional Overview

The System Timer tracks the global synchronized counter. It can be used for RTOS scheduling and real-time system tracking. This timer is provided in addition to the other timer peripherals to enable software/firmware to have a simple, globally synchronized timer source.

The System Timer Module provides real time measurement for all task scheduling, sensor sample rate calibration, and tracking of real time and calendar maintenance.

---

ERRATUM NOTICE

Compare interrupts are delayed by one STIMER clock. Additionally, it takes two STIMER clock cycles for the write to an SCMPRn register (where n is 0 to 7 representing one of the STIMER Compare registers) to get operated on. These timing issues put constraints on the minimum value of delta that can be applied to SCMPRn, which is 4 STIMER clock cycles.

In addition, back-to-back writes to SCMPRn may not work reliably (i.e., take the last value) unless the application ensures not to write within two STIMER clock cycles of the previous one. As well, after writing to SCMPRn, the application needs to wait for at least three STIMER clock cycles before reading it back for the new value to be reflected.

It takes two STIMER clock cycles for the write to STCFG to take effect. This caused 2 extra cycles to add to the minimum delta.

See "ERR029: STIMER: Constraints on writing to SCMPRn registers and handling Compare interrupts" in the *Apollo510 SoC Errata List*.

---

## 10.3 Additional Information

Please refer to the STIMER registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 11. Watchdog Timer (WDT)
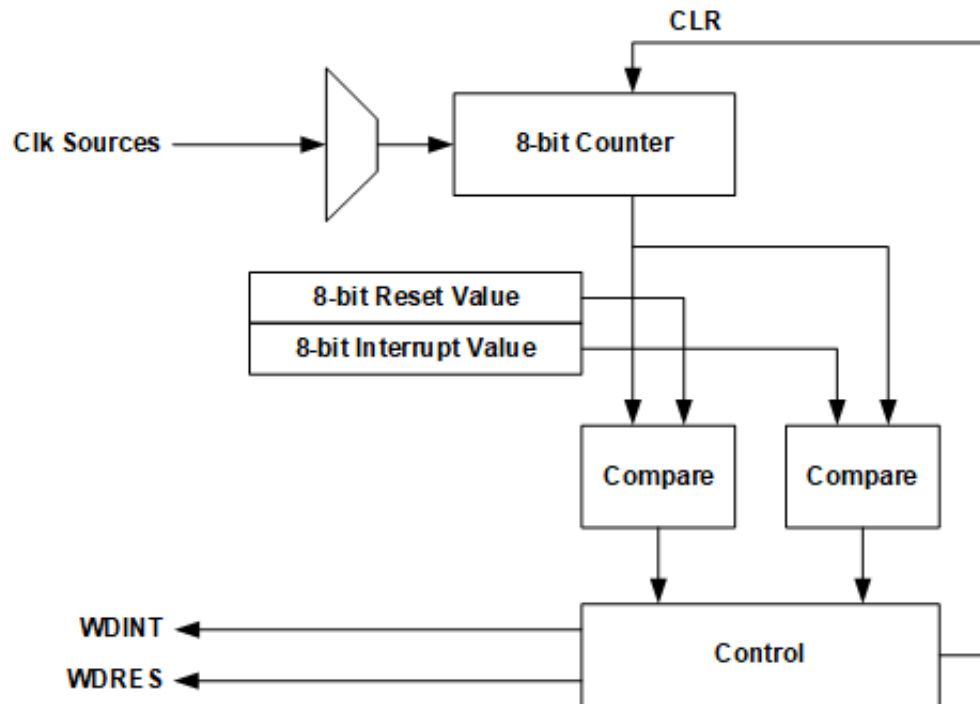


**Figure 15. Watchdog Timer Block Diagram**

## 11.1  Features

The Watchdog Timer (WDT) includes features shown in Figure 15 and listed below.

- Ensures software remains operational by initiating a reset if WDT times out
- May be clocked by one of four selectable prescalers of the LFRC clock
- May be locked to ensure software cannot disable its functionality
- Early warning may be implement via an interrupt

## 11.2   Functional Overview

The Watchdog Timer (WDT) is used to ensure that software is operational, by resetting the Apollo510 SoC if the WDT reaches a configurable value before being cleared by software. The WDT can be clocked by one of four selectable prescalers of the always active low-power LFRC clock, but is nominally clocked at 128 Hz. The WDT may be locked to ensure that software cannot disable its functionality, in which case the WDTCFG register cannot be accidentally reprogrammed. An interrupt can also be generated at a different counter value to implement an early warning function. Note: The RESEN bit in the WDTCFG register must be set and the WDREN bit in the RSTCFG register must be set to enable a watchdog timer reset condition.

## 11.3   Additional Information

Please refer to the WDT registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 12. General Purpose Input/Output (GPIO)
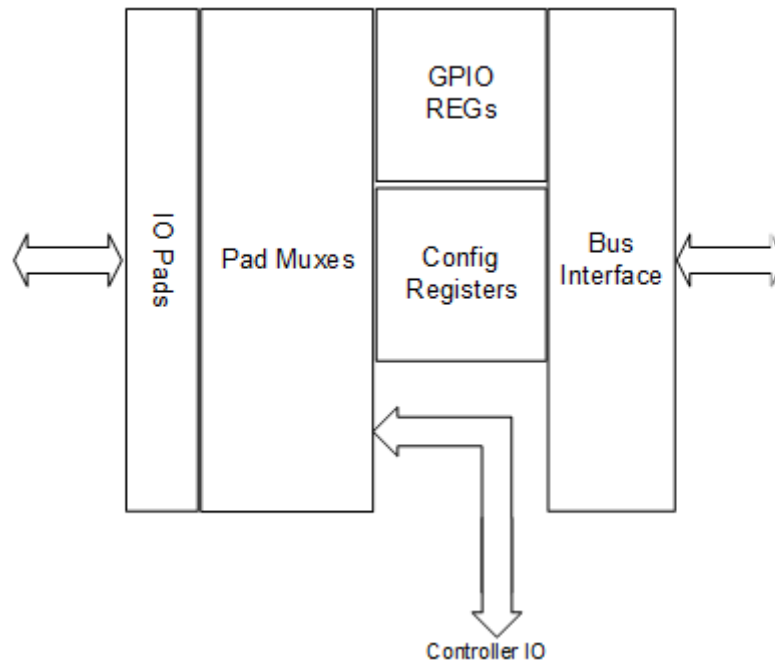


**Figure 16. Block Diagram for the General Purpose Input/Output (GPIO) Module**

## 12.1 Features

The GPIO module can be represented by the block diagram shown in Figure 16.

Key features include the following:

- Up to 190 GPIOs depending on derivative and package
- Six selectable I/O voltage domains depending on derivative and package (applies to set of GPIOs)
- 14 programmable GPIO interrupt groups
- Flexible GPIO Controls
  - 50 kΩ pull-down and 1.5 kΩ to 100 kΩ pull-up resistors
  - Multiple drive strengths
  - Multi-edge interrupts
  - Up to 11 function selects per GPIO
  - Grouped or masked interrupts
  - Configurable output drive mode (push-pull, open drain, tri-stated push-pull modes)

## 12.2 Functional Overview

The Apollo510 SoC's General Purpose I/O and Pad Configuration (GPIO) Module controls connections to up to 190 digital/analog pads. Each pad may be connected to a variety of module interface signals, with all pad input and output selection and control managed by the GPIO module. In addition, any pad may function as a general purpose input and/or output pad to be configured for a variety of external functions. Each GPIO may be configured to generate an interrupt when a transition occurs on the input.

For the Apollo510 SoC, any GPIO pad brought out to an external pin may be configured as a chip enable for any IOM or for the Display Controller. Pins available as chip enables for MSPI instances are as specified starting in Table 13. The pins available for specific MSPI chip enables are shown as function

selections with a "MNCEx_y" entry (orange), where x is the MSPI instance and y is the chip select number for that instance.

---

NOTE

For high-speed pads (GPIO37-GPIO45, 64-73, 105-113, 115-124, 142-146), there are 50 kΩ pull-up/pull-down options and 8 drive strength options (no slew rate).

---

NOTE

High-speed pads, those with MSPI0, MSPI3 or SWTRACE interface functionality, do not share the ability for the interface pin to be read by setting the pad's INPEN field when configured as an output, as other GPIO pads do.

---

## 12.3 GPIO Interrupts

The Apollo510 architecture has GPIO interrupts in two groups: 0-7 and 8-15.  0-6 are groups of 32 GPIO interrupts, and 8-14 form the second group (GPIO interrupts 7 and 15 are reserved). Each interrupt is enabled, disabled, cleared or set with a standard set of interrupt registers GPIO_MCUN0INTnEN, GPIO_MCUN0INTnSTAT, GPIO_MCUN0INTnCLR and GPIO_MCUN0INTnSET, where n = 0 to 6 for GPIO pads 0 to 31, 32 to 63, 64 to 95, 96 to 127, 128 to 159, 160 to 191, and 192 to 223, respectively[1]. The N0 designation in these registers indicates that these are interrupt register set 0. A duplicate set of registers, with a designation N1, is available and these registers are similarly named GPIO_MCUN1INTnEN, GPIO_MCUN1INTnSTAT, GPIO_MCUN1INTnCLR and GPIO_MCUN1INTnSET. The purpose of this dual set of registers is to enable segregation of an interrupt, or a small set of interrupts, which may have higher importance and thus should be handled at a higher priority level and/or with minimal latency.

## 12.4   Pad Configuration Functions

Each GPIO on a Apollo510 SoC package can be configured as one of several functions according to the Pin Mapping tables starting with Table 13. Functions are grouped by module per the color coding shown in Table 16.

The I/O voltage source reference for each pad as listed in the "I/O Voltage" column of the Pin Mapping tables equates to the corresponding voltage supply shown in Table 12. See the Electricals "External Voltage Supplies" section for specified voltage range for each supply.

---

1. Bits used in the GPIO_MCUNmINTn registers depend on the pads used by the specific SoC derivative and package. See the available pads for each package in the Pin Mapping table later in this chapter.

---

**Table 12: I/O Pin Voltage Source**

| I/O Voltage Reference | Voltage Supply | GPIO Powered | Modules Powered |
|---|---|---|---|
| 0 | VDDH | GPIO0-36, 58-63, 74-92, 138-139, 156-160, 165, 208 | ADC, DC, I2S, IOM, IOS, MSPI2, SDIO, SWTRACE, VCOMP |
| 1 | VDDH1 | GPIO142-147 | SWTRACE |
| 2 | VDDH2 | GPIO46, 50-57, 93, 95-104, 125-134, 136, 140-141, 148-155, 186-189 | DBIB, DC, I2S, MSPI1, PDM, SDIO, SWTRACE, VCOMP |
| 3 | VDDH3 | GPIO37-45, 64-73, 199-200 | MSPI0, SWTRACE |
| 4 | VDDH4 | GPIO105-124, 195 | MSPI3, I2S, SDIO |
| 5 | VDDH5 | GPIO209-223 | DBIB, DC |

---

NOTE

The Secure Bootloader (SBL) uses GPIO28 as the SWO output (1 Mbaud) to make bootup status/information available to the user. This pin is configured as an output and will toggle during bootup. Care should be taken by the user when connecting this pin to an external peripheral.

---

**Table 13: Apollo510 SoC Production Version Pin Mapping (Pg 1)**

| Pad | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 9 | 10 | 12 | BGA Pkg | CSP Pkg | I/O Voltage |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | SWTRACECLK | SLFDSCK | - | GPIO0 | UART0TX | UART1TX | CT0 | NCE0 | VCMPO | - | - | X | X | 0 |
| 1 | SWTRACE0 | SLFDMOSI | SLFDWIR3 | GPIO1 | UART2TX | UART3TX | CT1 | NCE1 | VCMPO | SLFD1WIR3 | - | X | X | 0 |
| 2 | SWTRACE1 | SLFDMISO | TRIG1 | GPIO2 | UART0RX | UART1RX | CT2 | NCE2 | VCMPO | SLFD1MISO | - | X | X | 0 |
| 3 | SWTRACE2 | SLFDnCE | SWO | GPIO3 | UART2RX | UART3RX | CT3 | NCE3 | - | I2S1_SDIN | - | X | X | 0 |
| 4 | SWTRACE3 | SLFDINT | 32KHzXT | GPIO4 | UART0RTS | UART1RTS | CT4 | NCE4 | I2S0_SDIN | SLFD1INT | - | X | X | 0 |
| 5 | M0SCL | M0SCK | I2S0_CLK | GPIO5 | UART2RTS | UART3RTS | CT5 | NCE5 | - | - | - | X | X | 0 |
| 6 | M0SDAWIR3 | M0MOSI | I2S0_DATA | GPIO6 | UART0CTS | UART1CTS | CT6 | NCE6 | I2S0_SDOUT | - | - | X | X | 0 |
| 7 | M0MISO | TRIG0 | I2S0_WS | GPIO7 | UART2CTS | UART3CTS | CT7 | NCE7 | MNCE2_0 | MNCE3_1 | - | X | X | 0 |
| 8 | CMPRF1 | TRIG1 | - | GPIO8 | M1SCL | M1SCK | CT8 | NCE8 | - | I2S1_CLK | - | X | X | 0 |
| 9 | CMPRF0 | TRIG2 | - | GPIO9 | M1SDAWIR3 | M1MOSI | CT9 | NCE9 | I2S1_DATA | I2S1_SDOUT | - | X | X | 0 |
| 10 | CMPIN0 | TRIG3 | MNCE0_0 | GPIO10 | M1MISO | MNCE2_0 | CT10 | NCE10 | DISP_TE | I2S1_WS | MNCE2_1 | X | X | 0 |
| 11 | CMPIN1 | TRIG0 | I2S0_CLK | GPIO11 | UART2RX | UART3RX | CT11 | NCE11 | SLSCL | SLSCK | - | X | X | 0 |
| 12 | ADCSE7 | TRIG1 | I2S0_DATA | GPIO12 | UART0TX | UART1TX | CT12 | NCE12 | CMPRF2 | I2S0_SDOUT | - | X | X | 0 |
| 13 | ADCSE6 | TRIG2 | I2S0_WS | GPIO13 | UART2TX | UART3TX | CT13 | NCE13 | SLnCE | - | - | X | X | 0 |
| 14 | ADCSE5 | TRIG3 | - | GPIO14 | - | UART1RX | CT14 | NCE14 | - | I2S0_SDIN | - | X | X | 0 |
| 15 | ADCSE4 | TRIG0 | - | GPIO15 | - | UART3RX | CT15 | NCE15 | - | REFCLK_EXT | - | X | X | 0 |
| 16 | ADCSE3 | TRIG1 | I2S1_CLK | GPIO16 | - | UART1RTS | CT16 | NCE16 | - | - | - | X | X | 0 |
| 17 | ADCSE2 | TRIG2 | I2S1_DATA | GPIO17 | - | UART3RTS | CT17 | NCE17 | I2S1_SDOUT | - | - | X | X | 0 |
| 18 | ADCSE1 | - | I2S1_WS | GPIO18 | UART0CTS | UART1CTS | CT18 | NCE18 | - | - | - | X | X | 0 |
| 19 | ADCSE0 | - | - | GPIO19 | UART2CTS | UART3CTS | CT19 | NCE19 | I2S1_SDIN | - | - | X | X | 0 |
| 20 | SWDCK | TRIG1 | - | GPIO20 | UART0TX | UART1TX | CT20 | NCE20 | - | - | - | X | X | 0 |
| 21 | SWDIO | TRIG2 | - | GPIO21 | UART0RX | UART1RX | CT21 | NCE21 | - | - | - | X | X | 0 |
| 22 | M7SCL | M7SCK | SWO | GPIO22 | UART2TX | UART3TX | CT22 | NCE22 | VCMPO | - | - | X | X | 0 |
| 23 | M7SDAWIR3 | M7MOSI | SWO | GPIO23 | UART2RX | UART3RX | CT23 | NCE23 | VCMPO | - | - | X | X | 0 |
| 24 | M7MISO | TRIG3 | SWO | GPIO24 | UART0RTS | UART1RTS | CT24 | NCE24 | MNCE0_0 | MNCE0_1 | - | X | X | 0 |
| 25 | M2SCL | M2SCK | - | GPIO25 | - | UART1TX | CT25 | NCE25 | - | - | - | X | X | 0 |
| 26 | M2SDAWIR3 | M2MOSI | - | GPIO26 | - | UART1RX | CT26 | NCE26 | VCMPO | - | - | X | X | 0 |
| 27 | M2MISO | TRIG0 | MNCE3_0 | GPIO27 | - | UART1CTS | CT27 | NCE27 | - | - | - | X | X | 0 |
| 28 | SWO | VCMPO | - | GPIO28 | UART2CTS | - | CT28 | NCE28 | - | - | - | X | X | 0 |
| 29 | TRIG0 | VCMPO | - | GPIO29 | UART1CTS | - | CT29 | NCE29 | - | - | - | X | X | 0 |
| 30 | TRIG1 | VCMPO | - | GPIO30 | UART0TX | - | CT30 | NCE30 | - | - | - | X | X | 0 |
| 31 | M3SCL | M3SCK | I2S0_CLK | GPIO31 | UART2TX | UART2CTS | CT31 | NCE31 | VCMPO | - | - | X | X | 0 |
| 32 | M3SDAWIR3 | M3MOSI | I2S0_DATA | GPIO32 | UART0RX | UART3CTS | CT32 | NCE32 | I2S0_SDOUT | - | - | X | X | 0 |
| 33 | M3MISO | CLKOUT | I2S0_WS | GPIO33 | UART2RX | UART2RTS | CT33 | NCE33 | DISP_TE | MNCE1_0 | - | X | X | 0 |
| 34 | M4SCL | M4SCK | SWO | GPIO34 | UART0TX | UART2RX | CT34 | NCE34 | VCMPO | I2S1_CLK | - | X | X | 0 |
| 35 | M4SDAWIR3 | M4MOSI | SWO | GPIO35 | UART2TX | UART3TX | CT35 | NCE35 | I2S1_SDOUT | I2S1_DATA | - | X | X | 0 |
| 36 | M4MISO | TRIG0 | MNCE3_0 | GPIO36 | UART0RX | UART1RX | CT36 | NCE36 | MNCE1_0 | I2S1_WS | - | X | X | 0 |
| 37 | MSPI0_10 | TRIG1 | 32KHzXT | GPIO37 | UART2RX | UART3RX | CT37 | NCE37 | - | - | - | X | X | 3 |
| 38 | MSPI0_11 | TRIG2 | SWTRACECLK | GPIO38 | UART0RTS | UART2RTS | CT38 | NCE38 | - | - | - | X | X | 3 |
| 39 | MSPI0_12 | TRIG3 | SWTRACE0 | GPIO39 | UART2RTS | UART3RTS | CT39 | NCE39 | - | - | - | X | X | 3 |
| 40 | MSPI0_13 | TRIG1 | SWTRACE1 | GPIO40 | UART0CTS | UART1CTS | CT40 | NCE40 | - | - | - | X | X | 3 |
| 41 | MSPI0_14 | TRIG0 | SWTRACE2 | GPIO41 | UART0TX | UART1TX | CT41 | NCE41 | SWO | - | - | X | X | 3 |
| 42 | MSPI0_15 | TRIG2 | SWTRACE3 | GPIO42 | UART2TX | UART3TX | CT42 | NCE42 | - | - | - | X | X | 3 |
| 43 | MSPI0_16 | TRIG3 | SWTRACECTL | GPIO43 | UART0RX | UART1RX | CT43 | NCE43 | - | - | - | X | X | 3 |
| 44 | MSPI0_17 | TRIG1 | SWO | GPIO44 | UART2RX | UART3RX | CT44 | NCE44 | VCMPO | - | - | X | X | 3 |
| 45 | MSPI0_18 | TRIG2 | 32KHzXT | GPIO45 | UART0TX | UART1TX | CT45 | NCE45 | - | - | - | X | X | 3 |
| 46 | - | TRIG3 | CLKOUT_32M | GPIO46 | UART2TX | UART3TX | CT46 | NCE46 | - | - | - | X | X | 2 |
| 47 | M5SCL | M5SCK | - | GPIO47 | UART0RX | UART1RX | CT47 | NCE47 | - | I2S0_CLK | - | X | X | 0 |
| 48 | M5SDAWIR3 | M5MOSI | - | GPIO48 | UART2RX | UART3RX | CT48 | NCE48 | - | I2S0_WS | - | X | X | 0 |
| 49 | M5MISO | TRIG0 | MNCE1_0 | GPIO49 | UART0RTS | UART1RTS | CT49 | NCE49 | I2S0_DATA | I2S0_SDOUT | MNCE1_1 | X | X | 0 |
| 50 | PDM0_CLK | TRIG0 | SWTRACECLK | GPIO50 | UART2RTS | UART3RTS | CT50 | NCE50 | DISP_TE | - | - | X | X | 2 |
| 51 | PDM0_DATA | TRIG1 | SWTRACE0 | GPIO51 | UART0CTS | UART1CTS | CT51 | NCE51 | - | - | - | X | X | 2 |
| 52 | MNCE3_0 | TRIG2 | SWTRACE1 | GPIO52 | UART2CTS | UART3CTS | CT52 | NCE52 | VCMPO | I2S0_CLK | SLSDAWIR3 | X | X | 2 |
| 53 | MNCE1_0 | TRIG3 | SWTRACE2 | GPIO53 | UART0TX | UART1TX | CT53 | NCE53 | - | I2S0_WS | - | X | X | 2 |
| 54 | MNCE1_1 | TRIG0 | SWTRACE3 | GPIO54 | UART2TX | UART3TX | CT54 | NCE54 | I2S0_DATA | I2S0_SDOUT | - | X | X | 2 |
| 55 | MNCE3_1 | TRIG1 | SWTRACECTL | GPIO55 | UART0RX | UART1RX | CT55 | NCE55 | - | I2S1_CLK | - | X | X | 2 |
| 56 | MNCE0_0 | TRIG2 | SWO | GPIO56 | UART2RX | UART3RX | CT56 | NCE56 | VCMPO | I2S1_WS | - | X | X | 2 |
| 57 | MNCE0_1 | TRIG3 | SWO | GPIO57 | UART0RTS | UART1RTS | CT57 | NCE57 | I2S1_DATA | I2S1_SDOUT | - | X | X | 2 |
| 58 | - | - | - | GPIO58 | UART0RTS | UART3RTS | CT58 | NCE58 | - | - | - | X | - | 0 |
| 59 | - | TRIG0 | - | GPIO59 | UART0CTS | UART1CTS | CT59 | NCE59 | - | - | - | X | X | 0 |
| 60 | - | TRIG1 | - | GPIO60 | UART0TX | UART3CTS | CT60 | NCE60 | - | - | - | X | X | 0 |

**Table 14: Apollo510 SoC Production Version Pin Mapping (Pg 2)**

| Pad | PADnFNCSEL 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 9 | 10 | 12 | BGA Pkg | CSP Pkg | I/O Voltage |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 61 | M6SCL | M6SCK | I2S1_CLK | GPIO61 | UART2TX | UART3TX | CT61 | NCE61 | - | - | - | X | X | 0 |
| 62 | M6SDAWIR3 | M6MOSI | I2S1_DATA | GPIO62 | UART0RX | UART1RX | CT62 | NCE62 | I2S1_SDOUT | - | - | X | X | 0 |
| 63 | M6MISO | CLKOUT | I2S1_WS | GPIO63 | UART2RX | UART3RX | CT63 | NCE63 | DISP_TE | MNCE2_0 | MNCE2_1 | X | X | 0 |
| 64 | MSPI0_0 | 32KHzXT | SWO | GPIO64 | UART0RTS | UART2CTS | CT64 | NCE64 | I2S1_SDIN | - | - | X | X | 3 |
| 65 | MSPI0_1 | 32KHzXT | SWO | GPIO65 | UART0CTS | UART1CTS | CT65 | NCE65 | - | - | - | X | X | 3 |
| 66 | MSPI0_2 | CLKOUT | SWO | GPIO66 | UART0TX | UART1TX | CT66 | NCE66 | - | - | - | X | X | 3 |
| 67 | MSPI0_3 | CLKOUT | SWO | GPIO67 | UART2TX | UART3TX | CT67 | NCE67 | - | - | - | X | X | 3 |
| 68 | MSPI0_4 | SWO | - | GPIO68 | UART0RX | UART1RX | CT68 | NCE68 | - | - | - | X | X | 3 |
| 69 | MSPI0_5 | 32KHzXT | SWO | GPIO69 | UART2RX | UART3RX | CT69 | NCE69 | - | - | - | X | X | 3 |
| 70 | MSPI0_6 | 32KHzXT | SWTRACE0 | GPIO70 | UART0RTS | UART1RTS | CT70 | NCE70 | - | - | - | X | X | 3 |
| 71 | MSPI0_7 | CLKOUT | SWTRACE1 | GPIO71 | UART0CTS | UART3RTS | CT71 | NCE71 | - | - | - | X | X | 3 |
| 72 | MSPI0_8 | CLKOUT | SWTRACE2 | GPIO72 | UART0TX | UART1TX | CT72 | NCE72 | VCMPO | - | - | X | X | 3 |
| 73 | MSPI0_9 | - | SWTRACE3 | GPIO73 | UART2TX | UART3TX | CT73 | NCE73 | - | - | - | X | X | 3 |
| 74 | MSPI2_0 | - | - | GPIO74 | UART0RX | UART3CTS | CT74 | NCE74 | - | - | - | X | X | 0 |
| 75 | MSPI2_1 | 32KHzXT | - | GPIO75 | UART2RX | UART3RX | CT75 | NCE75 | - | - | - | X | X | 0 |
| 76 | MSPI2_2 | 32KHzXT | - | GPIO76 | UART0RTS | UART1RTS | CT76 | NCE76 | - | - | - | X | X | 0 |
| 77 | MSPI2_3 | - | - | GPIO77 | UART0CTS | UART1CTS | CT77 | NCE77 | - | - | - | X | X | 0 |
| 78 | MSPI2_4 | - | - | GPIO78 | UART0TX | UART1TX | CT78 | NCE78 | - | - | - | X | X | 0 |
| 79 | MSPI2_5 | DISP_QSPI_D0_OUT | DISP_QSPI_D0 | GPIO79 | SWO | UART1RTS | CT79 | NCE79 | DISP_SPI_SD | DISP_SPI_SDO | - | X | X | 0 |
| 80 | MSPI2_6 | CLKOUT | DISP_QSPI_D1 | GPIO80 | SWTRACE0 | UART2CTS | CT80 | NCE80 | DISP_SPI_DCX | - | - | X | X | 0 |
| 81 | MSPI2_7 | CLKOUT | DISP_QSPI_SCK | GPIO81 | SWTRACE1 | UART2RTS | CT81 | NCE81 | DISP_SPI_SCK | - | - | X | X | 0 |
| 82 | MSPI2_8 | 32KHzXT | DISP_QSPI_D2 | GPIO82 | SWTRACE2 | UART3CTS | CT82 | NCE82 | DISP_SPI_SDI | - | - | X | X | 0 |
| 83 | MSPI2_9 | 32KHzXT | DISP_QSPI_D3 | GPIO83 | SWTRACE3 | UART3RTS | CT83 | NCE83 | DISP_SPI_RST | SLMISO | - | X | X | 0 |
| 84 | - | - | SDIF0_DAT0 | GPIO84 | - | - | CT84 | NCE84 | - | - | - | X | X | 0 |
| 85 | - | - | SDIF0_DAT1 | GPIO85 | - | - | CT85 | NCE85 | - | - | - | X | X | 0 |
| 86 | - | - | SDIF0_DAT2 | GPIO86 | - | - | CT86 | NCE86 | - | - | - | X | X | 0 |
| 87 | - | - | SDIF0_DAT3 | GPIO87 | - | - | CT87 | NCE87 | DISP_TE | - | - | X | X | 0 |
| 88 | - | - | SDIF0_CLKOUT | GPIO88 | - | - | CT88 | NCE88 | - | - | - | X | X | 0 |
| 89 | - | - | - | GPIO89 | - | - | CT89 | NCE89 | DISP_SPI_RST | - | - | X | - | 0 |
| 90 | - | - | - | GPIO90 | - | - | CT90 | NCE90 | VCMPO | - | - | X | - | 0 |
| 91 | - | - | - | GPIO91 | - | - | CT91 | NCE91 | VCMPO | - | - | X | - | 0 |
| 92 | - | - | - | GPIO92 | - | - | CT92 | NCE92 | VCMPO | - | - | X | - | 0 |
| 93 | MNCE1_0 | - | - | GPIO93 | - | - | CT93 | NCE93 | VCMPO | - | - | X | X | 2 |
| 94 | - | - | - | GPIO94 | - | - | CT94 | NCE94 | VCMPO | - | - | X | - | 0 |
| 95 | MSPI1_0 | - | - | GPIO95 | - | - | CT95 | NCE95 | I2S0_SDIN | - | - | X | X | 2 |
| 96 | MSPI1_1 | - | - | GPIO96 | - | - | CT96 | NCE96 | - | - | - | X | X | 2 |
| 97 | MSPI1_2 | - | - | GPIO97 | - | - | CT97 | NCE97 | - | - | - | X | X | 2 |
| 98 | MSPI1_3 | - | - | GPIO98 | - | - | CT98 | NCE98 | - | - | - | X | X | 2 |
| 99 | MSPI1_4 | - | - | GPIO99 | - | - | CT99 | NCE99 | - | - | - | X | X | 2 |
| 100 | MSPI1_5 | DISP_QSPI_D0_OUT | DISP_QSPI_D0 | GPIO100 | DISP_SPI_SD | DISP_SPI_SDO | CT100 | NCE100 | I2S0_CLK | - | - | X | X | 2 |
| 101 | MSPI1_6 | - | DISP_QSPI_D1 | GPIO101 | DISP_SPI_DCX | - | CT101 | NCE101 | I2S0_DATA | I2S0_SDOUT | - | X | X | 2 |
| 102 | MSPI1_7 | - | DISP_QSPI_SCK | GPIO102 | DISP_SPI_SCK | - | CT102 | NCE102 | I2S0_WS | - | - | X | X | 2 |
| 103 | MSPI1_8 | - | DISP_QSPI_D2 | GPIO103 | DISP_SPI_SDI | - | CT103 | NCE103 | - | - | - | X | X | 2 |
| 104 | MSPI1_9 | - | DISP_QSPI_D3 | GPIO104 | DISP_SPI_RST | - | CT104 | NCE104 | - | - | - | X | X | 2 |
| 105 | MSPI3_10 | - | - | GPIO105 | - | - | CT105 | NCE105 | - | - | - | X | X | 4 |
| 106 | MSPI3_11 | - | - | GPIO106 | - | - | CT106 | NCE106 | - | - | - | X | X | 4 |
| 107 | MSPI3_12 | - | - | GPIO107 | - | - | CT107 | NCE107 | - | - | - | X | X | 4 |
| 108 | MSPI3_13 | - | - | GPIO108 | - | - | CT108 | NCE108 | - | - | - | X | X | 4 |
| 109 | MSPI3_14 | - | - | GPIO109 | - | - | CT109 | NCE109 | - | - | - | X | X | 4 |
| 110 | MSPI3_15 | - | - | GPIO110 | - | - | CT110 | NCE110 | - | - | - | X | X | 4 |
| 111 | MSPI3_16 | - | - | GPIO111 | - | - | CT111 | NCE111 | - | - | - | X | X | 4 |
| 112 | MSPI3_17 | - | - | GPIO112 | - | - | CT112 | NCE112 | - | - | - | X | X | 4 |
| 113 | MSPI3_18 | - | - | GPIO113 | - | - | CT113 | NCE113 | - | - | - | X | X | 4 |
| 114 | - | MNCE3_0 | - | GPIO114 | - | - | CT114 | NCE114 | - | - | - | X | X | 4 |
| 115 | MSPI3_0 | - | - | GPIO115 | - | - | CT115 | NCE115 | - | - | - | X | X | 4 |
| 116 | MSPI3_1 | - | - | GPIO116 | - | - | CT116 | NCE116 | - | - | - | X | X | 4 |
| 117 | MSPI3_2 | - | - | GPIO117 | - | - | CT117 | NCE117 | - | - | - | X | X | 4 |
| 118 | MSPI3_3 | - | - | GPIO118 | - | - | CT118 | NCE118 | - | - | - | X | X | 4 |
| 119 | MSPI3_4 | - | - | GPIO119 | - | - | CT119 | NCE119 | - | - | - | X | X | 4 |
| 120 | MSPI3_5 | - | - | GPIO120 | - | - | CT120 | NCE120 | - | - | - | X | X | 4 |
| 121 | MSPI3_6 | - | - | GPIO121 | - | - | CT121 | NCE121 | - | - | - | X | X | 4 |

**Table 15: Apollo510 SoC Production Version SoC Pin Mapping (Pg 3)**

| Pad | PADnFNCSEL 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 9 | 10 | 12 | BGA Pkg | CSP Pkg | I/O Voltage |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 122 | MSPI3_7 | - | - | GPIO122 | - | - | CT122 | NCE122 | - | - | - | X | X | 4 |
| 123 | MSPI3_8 | - | - | GPIO123 | - | - | CT123 | NCE123 | - | - | - | X | X | 4 |
| 124 | MSPI3_9 | - | - | GPIO124 | - | - | CT124 | NCE124 | - | - | - | X | X | 4 |
| 125 | SDIF1_DAT0 | - | - | GPIO125 | - | - | CT125 | NCE125 | - | - | - | X | X | 2 |
| 126 | SDIF1_DAT1 | - | - | GPIO126 | - | - | CT126 | NCE126 | - | - | - | X | X | 2 |
| 127 | SDIF1_DAT2 | - | - | GPIO127 | - | - | CT127 | NCE127 | - | - | - | X | X | 2 |
| 128 | SDIF1_DAT3 | - | - | GPIO128 | - | - | CT128 | NCE128 | - | - | - | X | X | 2 |
| 129 | SDIF1_CLKOUT | - | - | GPIO129 | - | - | CT129 | NCE129 | - | - | - | X | X | 2 |
| 130 | SDIF1_DAT4 | MNCE0_0 | - | GPIO130 | - | - | CT130 | NCE130 | - | I2S1_CLK | - | X | X | 2 |
| 131 | SDIF1_DAT5 | MNCE3_0 | - | GPIO131 | - | - | CT131 | NCE131 | I2S1_DATA | I2S1_SDOUT | - | X | X | 2 |
| 132 | SDIF1_DAT6 | - | - | GPIO132 | - | - | CT132 | NCE132 | - | I2S1_WS | - | X | X | 2 |
| 133 | SDIF1_DAT7 | - | - | GPIO133 | - | - | CT133 | NCE133 | - | I2S1_SDIN | - | X | X | 2 |
| 134 | SDIF1_CMD | - | - | GPIO134 | - | - | CT134 | NCE134 | - | - | - | X | X | 2 |
| 136 | - | DISP_D15 | - | GPIO136 | - | - | CT136 | NCE136 | - | - | - | X | - | 2 |
| 138 | MNCE1_1 | 32KHzXT | - | GPIO138 | - | - | CT138 | NCE138 | I2S0_SDIN | - | - | X | X | 0 |
| 139 | - | - | - | GPIO139 | - | - | CT139 | NCE139 | - | I2S0_SDIN | - | X | - | 0 |
| 140 | - | DISP_D16 | - | GPIO140 | - | - | CT140 | NCE140 | - | - | - | X | - | 2 |
| 141 | - | DISP_D17 | - | GPIO141 | - | - | CT141 | NCE141 | - | - | - | X | - | 2 |
| 142 | SWTRACECLK | - | - | GPIO142 | - | - | CT142 | NCE142 | - | - | - | X | X | 1 |
| 143 | SWTRACE0 | - | - | GPIO143 | - | - | CT143 | NCE143 | - | - | - | X | X | 1 |
| 144 | SWTRACE1 | - | - | GPIO144 | - | - | CT144 | NCE144 | - | - | - | X | X | 1 |
| 145 | SWTRACE2 | - | - | GPIO145 | - | - | CT145 | NCE145 | - | - | - | X | X | 1 |
| 146 | SWTRACE3 | - | - | GPIO146 | - | - | CT146 | NCE146 | - | - | - | X | X | 1 |
| 147 | SWTRACECTL | - | - | GPIO147 | - | - | CT147 | NCE147 | MNCE0_0 | MNCE3_0 | - | X | X | 1 |
| 148 | - | DISP_D5 | DBIB_D8 | GPIO148 | - | - | CT148 | NCE148 | - | - | - | X | - | 2 |
| 149 | - | DISP_D6 | DBIB_D9 | GPIO149 | - | - | CT149 | NCE149 | - | - | - | X | - | 2 |
| 150 | - | DISP_D7 | DBIB_D10 | GPIO150 | - | - | CT150 | NCE150 | - | - | - | X | - | 2 |
| 151 | - | DISP_D8 | DBIB_D11 | GPIO151 | - | - | CT151 | NCE151 | - | - | - | X | - | 2 |
| 152 | - | DISP_D9 | DBIB_D12 | GPIO152 | - | - | CT152 | NCE152 | - | - | - | X | - | 2 |
| 153 | - | DISP_D10 | DBIB_D13 | GPIO153 | - | - | CT153 | NCE153 | - | - | - | X | - | 2 |
| 154 | - | DISP_D11 | DBIB_D14 | GPIO154 | - | - | CT154 | NCE154 | - | - | - | X | - | 2 |
| 155 | - | DISP_D12 | DBIB_D15 | GPIO155 | - | - | CT155 | NCE155 | - | - | - | X | - | 2 |
| 156 | SDIF0_DAT4 | MNCE2_1 | - | GPIO156 | - | - | CT156 | NCE156 | - | - | - | X | X | 0 |
| 157 | SDIF0_DAT5 | MNCE2_0 | - | GPIO157 | - | - | CT157 | NCE157 | - | - | - | X | X | 0 |
| 158 | SDIF0_DAT6 | - | - | GPIO158 | - | - | CT158 | NCE158 | - | - | - | X | X | 0 |
| 159 | SDIF0_DAT7 | - | - | GPIO159 | - | - | CT159 | NCE159 | - | - | - | X | X | 0 |
| 160 | SDIF0_CMD | - | - | GPIO160 | - | - | CT160 | NCE160 | - | - | - | X | X | 0 |
| 165 | - | - | - | GPIO165 | - | - | CT165 | NCE165 | - | - | - | X | X | 0 |
| 186 | - | DISP_SD | - | GPIO186 | - | - | CT186 | NCE186 | - | - | - | X | - | 2 |
| 187 | - | DISP_DE | - | GPIO187 | - | - | CT187 | NCE187 | - | - | - | X | - | 2 |
| 188 | - | DISP_CM | - | GPIO188 | - | - | CT188 | NCE188 | - | - | - | X | - | 2 |
| 189 | - | DISP_PCLK | - | GPIO189 | - | - | CT189 | NCE189 | - | - | - | X | - | 2 |
| 195 | - | MNCE3_1 | MNCE3_0 | GPIO195 | - | - | CT195 | NCE195 | - | - | - | X | - | 4 |
| 199 | - | MNCE0_0 | - | GPIO199 | - | - | CT199 | NCE199 | - | - | - | X | X | 3 |
| 200 | - | MNCE0_1 | MNCE0_0 | GPIO200 | - | - | CT200 | NCE200 | - | - | - | X | X | 3 |
| 208 | - | 32KHzXT | - | GPIO208 | - | - | CT208 | NCE208 | - | - | - | X | - | 0 |
| 209 | DISP_ENB | DISP_D1 | DBIB_CSX | GPIO209 | - | - | CT209 | NCE209 | - | - | - | X | - | 5 |
| 210 | DISP_XRST | DISP_D4 | DBIB_DCX | GPIO210 | - | - | CT210 | NCE210 | - | - | - | X | - | 5 |
| 211 | DISP_R1 | DISP_D23 | DBIB_WRX | GPIO211 | - | - | CT211 | NCE211 | - | - | - | X | - | 5 |
| 212 | DISP_R2 | DISP_D22 | DBIB_RDX | GPIO212 | - | - | CT212 | NCE212 | - | - | - | X | - | 5 |
| 213 | DISP_G1 | DISP_D21 | DBIB_D0 | GPIO213 | - | - | CT213 | NCE213 | - | - | - | X | - | 5 |
| 214 | DISP_G2 | DISP_D20 | DBIB_D1 | GPIO214 | - | - | CT214 | NCE214 | - | - | - | X | - | 5 |
| 215 | DISP_B1 | DISP_D19 | DBIB_D2 | GPIO215 | - | - | CT215 | NCE215 | - | - | - | X | - | 5 |
| 216 | DISP_B2 | DISP_D18 | DBIB_D3 | GPIO216 | - | - | CT216 | NCE216 | - | - | - | X | - | 5 |
| 217 | DISP_HST | DISP_HS | DBIB_D4 | GPIO217 | - | - | CT217 | NCE217 | - | - | - | X | - | 5 |
| 218 | DISP_VST | DISP_VS | DBIB_D5 | GPIO218 | - | - | CT218 | NCE218 | - | - | - | X | - | 5 |
| 219 | DISP_HCK | DISP_D3 | DBIB_D6 | GPIO219 | - | - | CT219 | NCE219 | - | - | - | X | - | 5 |
| 220 | DISP_VCK | DISP_D0 | DBIB_D7 | GPIO220 | - | - | CT220 | NCE220 | - | - | - | X | - | 5 |
| 221 | - | DISP_D2 | - | GPIO221 | - | - | CT221 | NCE221 | - | - | - | X | - | 5 |
| 222 | - | DISP_D13 | - | GPIO222 | - | - | CT222 | NCE222 | MNCE0_0 | - | - | X | - | 5 |
| 223 | - | DISP_D14 | - | GPIO223 | - | - | CT223 | NCE223 | MNCE3_0 | - | - | X | - | 5 |

**Table 16: Pad Function Color Code**

| Color/Symbol | Module |
|---|---|
| Analog | Analog Modules (ADC, VCOMP) |
| CLKOUT | Clock Output |
| CT | Counter/Timer |
| DBIB | MIPI DBI Type B Display Interface |
| Debug | Debug/Special |
| DISP | Display Interface |
| GPIO | General Purpose Input/Output |
| I2S0 | Inter-IC Sound 0 |
| I2S1 | Inter-IC Sound 1 |
| IOM0 | I2C/SPI Manager 0 |
| IOM1 | I2C/SPI Manager 1 |
| IOM2 | I2C/SPI Manager 2 |
| IOM3 | I2C/SPI Manager 3 |
| IOM4 | I2C/SPI Manager 4 |
| IOM5 | I2C/SPI Manager 5 |
| IOM6 | I2C/SPI Manager 6 |
| IOM7 | I2C/SPI Manager 7 |
| IOS | I2C/SPI Subordinate |
| MSPI0 | Multi-bit Serial Peripheral Interface 0 |
| MSPI1 | Multi-bit Serial Peripheral Interface 1 |
| MSPI2 | Multi-bit Serial Peripheral Interface 2 |
| MSPI3 | Multi-bit Serial Peripheral Interface 3 |
| NCE/MNCE | IOM/MSPI Chip Enables |
| PDM0 | PDM-to-PCM Converter |
| SDIF0 | Secure Digital Input Output Interface 0 |
| SDIF1 | Secure Digital Input Output Interface 1 |
| UART0 | Universal Asynchronous Receiver/Transmitter 0 |
| UART1 | Universal Asynchronous Receiver/Transmitter 1 |
| UART2 | Universal Asynchronous Receiver/Transmitter 2 |
| UART3 | Universal Asynchronous Receiver/Transmitter 3 |
| USB | Universal Serial Bus |

## 12.5 Additional Information

Please refer to the GPIO registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 13. General Purpose ADC and Temperature Sensor Module
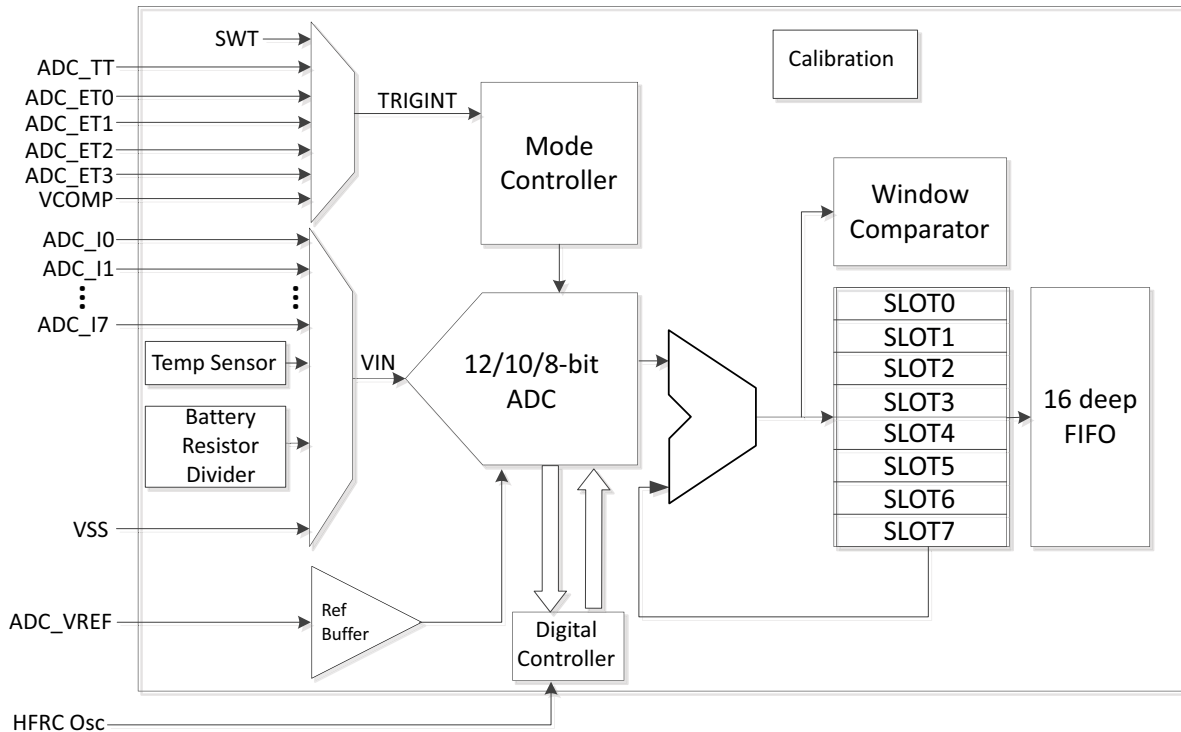


**Figure 17. Block Diagram for ADC and Temperature Sensor**

## 13.1 Features

The general purpose Analog-to-Digital Converter (ADC) and Temperature Sensor Module include a single-ended 12-bit multichannel Successive Approximation Register (SAR) ADC as shown in Figure 17.

Key features include:

- 11 user-selectable channels with sources including:
  - 8 single ended external pins
  - Internal voltage (VSS)
  - Voltage divider (battery)
  - Temperature sensor
- Configurable automatic low power control between scans
- Optional Battery load enable for voltage divider measurement
- Single shot, repeating single shot, scan, and repeating scan modes
- Variable sample tracking time, configurable on per-slot basis
- User-selectable clock source for variable sampling rates
- Automatically accumulate and scale module for hardware averaging of samples
- A 16-entry FIFO and DMA capability for storing measurement results and maximizing MCU sleep time
- Supports ping-pong DMA jobs
- Window comparator for monitoring excursions of voltage into or out of user-selectable thresholds
- Up to 2.8 MS/s effective continuous, multi-slot sampling rate (at 8-bit resolution)
- Interrupts for FIFO full, FIFO 75% full, Scan Complete, Conversion Complete, Window Incursion, Window Excursion, and various DMA-related notifications

## 13.2  Functional Overview

The Apollo510 SoC integrates a 12-bit successive approximation Analog to Digital Converter (ADC) block for sensing both internal and external voltages. The block provides eight separately managed conversion requests, called slots which are serially sequenced. The result of each conversion requests is delivered to a 16 deep FIFO. Firmware can utilize various interrupt notifications to determine when to collect the sampled data from the FIFO or from a buffer written by DMA. This block is extremely effective at automatically managing its power states and its clock sources.

The ADC supports one internal reference source used for the analog to digital conversion step. The reference voltage is 1.19 V and is not user settable. ADC input voltages > 1.19 V exceed the ADC range and return full scale code, but will not damage ADC inputs.

DMA jobs may be ping-pong processed to allow software to pre-process and/or post-process one DMA job while hardware is processing another DMA job.

## 13.3  Voltage Divider and Switchable Battery Load

The Apollo510 SoC's ADC includes a switchable voltage divider that enables the ADC to measure the input voltage to the VDD rail. In most systems this will be the battery voltage applied to the SoC. The voltage divider is only switched on when one of the active slots is selecting analog mux channel 9. That is only when the mode controller is ultimately triggered and powers up the ADC block for a conversion scan of all active slots. Otherwise, the voltage divider is turned off.
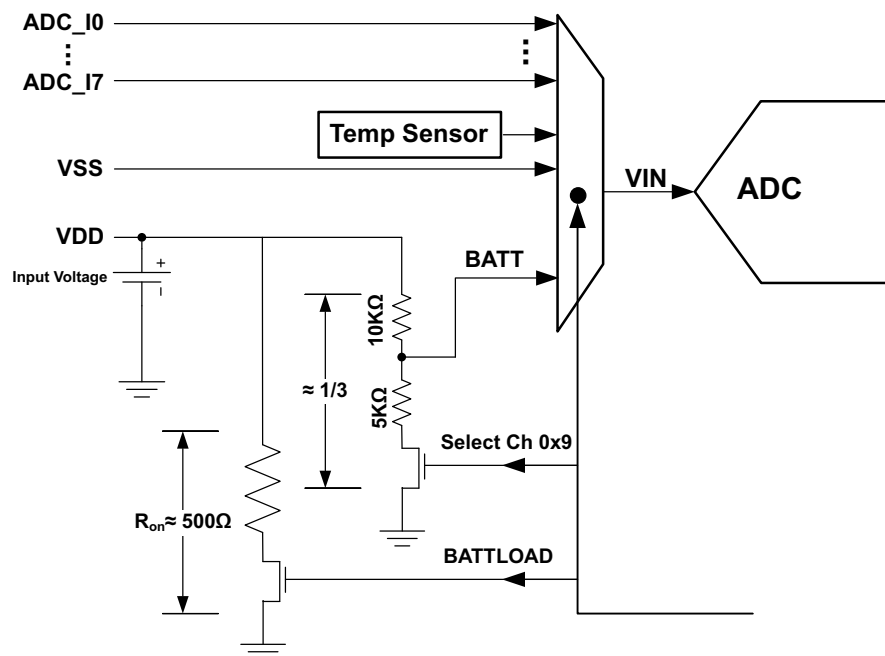


**Figure 18. Switchable Battery Load**

The switchable load resistor is enabled by the BATTLOAD bit as shown in the ADCBATTLOAD Register of the MCUCTRL Registers.This feature is used to help estimate the health of the battery chemistry by estimating the internal resistance of the battery.

## 13.4  Additional Information

Please refer to the ADC registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 14. Voltage Comparator (VCOMP)
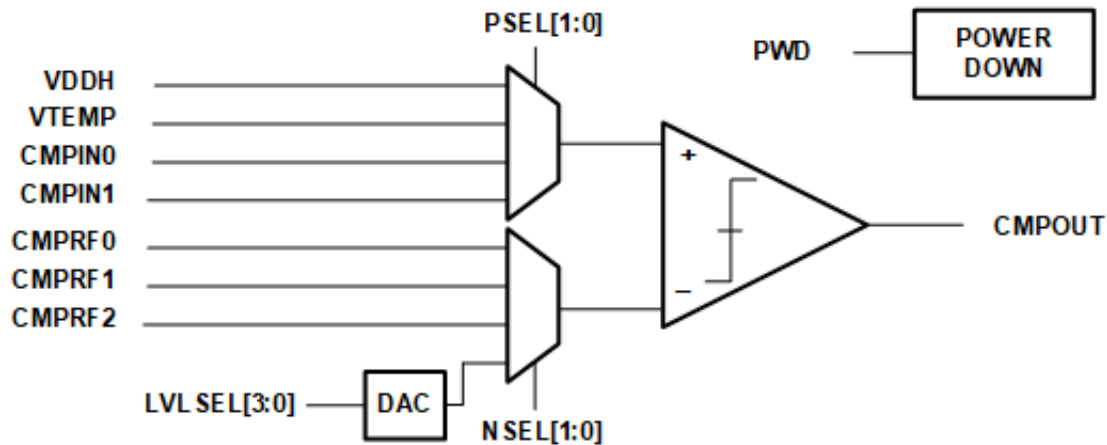


**Figure 19. Voltage Comparator Block Diagram**

## 14.1 Features

The Voltage Comparator (VCOMP) includes features shown in Figure 19 and listed below.

- Measures a user-selectable voltage
- Multiple options for input and reference voltages
- Provides interrupt and software access to comparator output
- Can generate an interrupt when monitored voltage rises above or drops below a user-configurable threshold
- Monitored voltage may be any of:
  - Supply voltage (VDDH)
  - The PTAT voltage from the temperature sensor (VTEMP)
  - One of two external voltage channels - CMPIN0, CMPIN1
- Reference voltage may be any of:
  - Three external voltage channels - CMPRF0, CMPRF1, CMPRF2
  - Internally generated reference voltage (VREFINT) tunable using on-chip DAC with level select signal LVLSEL[3:0].
- VCOMPOUT output remains high while the positive input is above reference input and transitions low when the positive input falls below the reference input
- CMPOUT output directly accessible via register read
- Two settable interrupts
  - OUTHI can trigger if VCOMPOUT transitions high
  - OUTLOW can trigger if VCOMPOUT transitions low

## 14.2 Functional Overview

The Voltage Comparator Module in the Apollo510 SoC measures a user-selectable voltage at all times. It provides interrupt and software access to the comparator output with multiple options for input and reference voltages. It can be configured to generate an interrupt when the monitored voltage rises above a user-configurable threshold or when the monitored voltage drops below a user-configurable threshold.

The voltage to be monitored is selected by programming the comparator's positive terminal signal, CFG_PSEL[1:0] and may be any of:

1. The supply voltage (VDDH)

2.	The PTAT voltage from the temperature sensor (VTEMP)

3.	Two external voltage channels selected by a GPIO function (CMPIN0 and CMPIN1)

The reference voltage is selected by programming the comparator's negative terminal, CFG_NSEL[1:0] and may be either of:

1.	Three external voltage channels selected by a GPIO function (CMPRF0, CMPRF1, CMPRF2)

2.	The internally generated reference voltage selected by CFG_LVLSEL

The internal reference voltage is tuned using an on-chip DAC with level select signal LVLSEL[3:0]. When using external inputs or reference inputs, the associated pads must be configured using the GPIO function selects explained in the GPIO document section.

The Voltage Comparator CMPOUT output will remain high while the voltage at the positive input is above the voltage at reference input. The CMPOUT output will transition low when the voltage at the positive input to the comparator falls below the reference input taking into account hysteresis. The CMPOUT output is directly accessible by software by reading the CMPOUT field in the status register. The OUTHI interrupt will be set if enabled and the CMPOUT transitions high or if it is high at the time the interrupt is enabled. Similarly, the OUTLOW interrupt will be set if enabled and the CMPOUT output transitions low or if it is low at the time the interrupt is enabled.

## 14.3	Additional Information

Please refer to the VCOMP registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 15. Multi-bit Serial Peripheral Interface (MSPI)
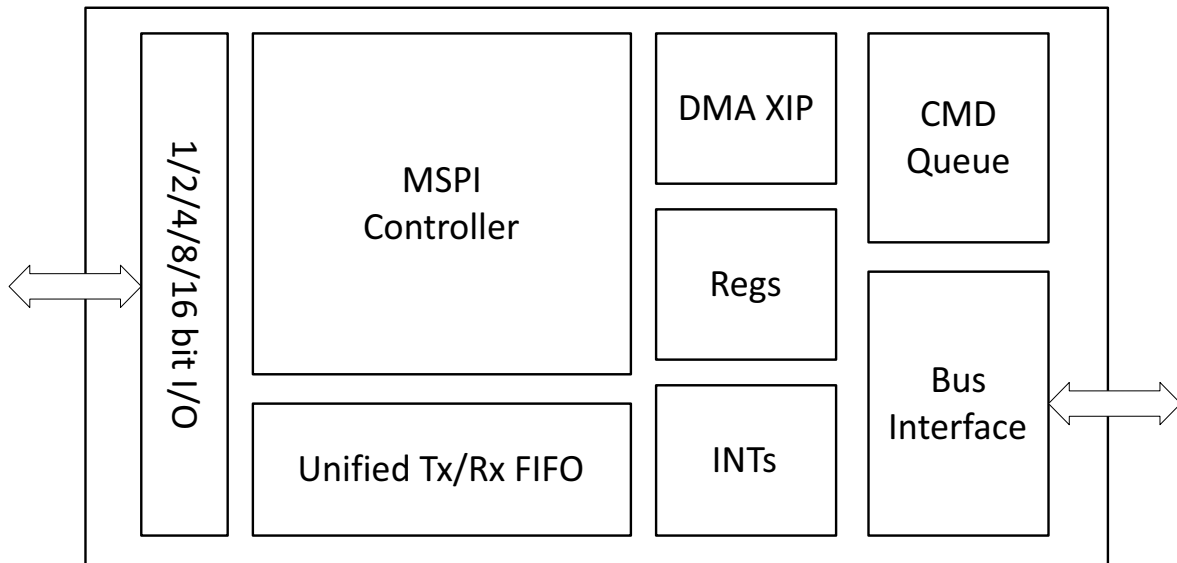


**Figure 20. MSPI Module Block Diagram**

## 15.1  Features

Each of the Multi-bit Serial Peripheral Interface (MSPI) modules includes the major functional blocks as shown in Figure 20.

Key MSPI features include:

- 2x HexSPI controllers
  - 1/2/4/8/16-bit SPI interface
  - SDR/DDR modes
  - Up to 125 MT/s SDR and 250 MT/s DDR throughput on MSPI0 and MSPI3
  - Supports up to 256 MB devices on MSPI0 and 128 MB devices on MSPI3
  - Interoperability with JESD-251 rev C-compliant devices
- 2x OctalSPI controllers
  - 1/2/4/8-bit SPI interface
  - SDR/DDR modes
  - Up to 96 MT/s SDR and DDR throughput on MSPI1 and MSPI2
  - Supports up to 64 MB devices
- 9-bit command mode support
- XiP support
- DMA with peripheral-to-memory and peripheral-to-peripheral support
- Command Queue support
- All four SPI CPOL/CPHA modes supported

| NOTE |
| --- |
| There is no full-duplex operation in 4-wire SPI mode of an MSPI on the Apollo510 SoC. As well, MSPI 4-wire mode does not support reads of an arbitrary length, instead it always reads multiples of 4. |

Please refer to the MSPI registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

---

ERRATUM NOTICE

An explicit memory access (read, write and instruction fetch) or a data and instruction pre-fetch to a memory that is in a powered down or unclocked state may cause a system hang. If there is dirty data in the data cache, a cacheable access to another data region may cause a cache-line eviction when attempting to write back the cache line to memory. If that memory is powered down, then a system hang may occur.

See "ERR009: MSPI: Accessing some powered-down/unclocked memory interfaces can cause system hangs" in the *Apollo510 SoC Errata List*.

---

ERRATUM NOTICE

When an MSPI clock source lower than or equal to 48 MHz is selected, not all MSPI peripheral accesses can be guaranteed. Thus, for MSPI0 and MSPI3, only HFRC_96MHz, HRFC2_125MHz, HFRC_192MHz or HRFC2_250MHz may be selected in the CLKGEN_MSPIIOCLKCTRL_MSPInIOCLKSEL register fields. For MSPI1 and MSPI2, only HFRC_96MHz or HFRC_192MHz may be selected.

See "ERR016: MSPI: Peripheral accesses not guaranteed when the clock source <= 48 MHz" in the *Apollo510 SoC Errata List*.

---

ERRATUM NOTICE

A DMA transfer which has a start address near the upper boundary of a memory block and the transfer extends into the next memory block, a device hang occurs. The root cause is that a DMA burst is fixed to 0, so the DMA address will not change when it crosses the memory boundary.

This condition applies for several boundaries, which are the DTCM - SRAM0, SRAM0 - SRAM1 and SRAM1 - SRAM2 boundaries. The address ranges for the affected memory spaces are the following for the Apollo510/Apollo510B devices:

- DTCM:    0x20000000 - 0x2007FFFF
- SSRAM0: 0x20080000 - 0x2017FFFF
- SSRAM1: 0x20180000 - 0x2027FFFF
- SSRAM2: 0x20280000 - 0x2037FFFF

In addition, this susceptibility to device hang occurs when a DMA transfer crosses a 4 kB boundary in DTCM.

See "ERR018: MSPI: Crossing a memory boundary during DMA transfer causes device hang" in the *Apollo510 SoC Errata List*.

---

ERRATUM NOTICE

Behavior of the upper data lines (IO2 and IO3) during the instruction phase of mixed-mode (1-1-4 or 1-4-4) transfers is preventing IO3 to function properly as the nHOLD signal, as is done on certain NAND devices. When nHOLD is pulled low at the start of the transaction and despite pull-ups present, it causes the chip to ignore all other signals. The SoC should leave the upper data lines floating during the instruction phase of mixed-mode transfers.

See "ERR019: MSPI: Upper data lines are pulled low instead of staying in high impedance mode" in the *Apollo510 SoC Errata List*.

ERRATUM NOTICE

A glitch occurs on the D0 line between address and data in mixed mode D4, or 1-1-4. The glitch is observed only during write operation in 1-1-4 mode and its occurrence may cause data write failures. Read operation is not affected and therefore it does not pose a problem for XIP/XIPMM reads

See "ERR020: MSPI: Mixed Mode 1-1-4 does not work as expected" in the *Apollo510 SoC Errata List*.

## 15.2  Functional Overview

The Apollo510 SoC integrates four MSPI modules which can be used to connect to external memory mapped devices. Two MSPI controllers support DDR operation up to 125 MHz and can transfer in serial, dual, quad, octal and hex (x16) modes. The other two controllers support DDR operation up to 48 MHz (96 MT/s) and can transfer in serial, dual, quad, and octal modes.

Each MSPI module has a unified 32-entry FIFO (32 bits wide) that is used for both transmit and receive data. To ensure that transactions are not dropped because of system or software latency, the MSPI controller pauses the clock (and thus the transfer on the bus) if the TX FIFO empties or the RX FIFO fills during an operation. It automatically resumes once the FIFO condition has cleared.

MSPI transfers generally consist of transmitting a 1-byte instruction, a 1-byte to 4-byte address (optional), and 1 byte to 16 MB of write or read data (with an optional number of turnaround clock cycles between address0 and RX data, as well as an optional number of turnaround clock cycles between address and RX data).

Most devices use the same number of pins to transmit instruction, address, and data (for example, all are quad or all are serial). However, some devices utilize mixed transfer modes to implement parallel data transfer on top of an inherently serial command structure. These devices are supported by the MSPI by utilizing the XIPMIXEDn configuration, which forces the MSPI to switch into dual or quad modes of operation for a portion of the transfer.

To utilize mixed mode transfers, the MSPI's normal configuration should be set to match the device's transfer characteristics for commands (usually serial), which allows the MSPI to communicate with the device in its native mode. The XIPMIXEDn field in the DEVnXIP register should then be programmed to indicate whether the data phase (and optionally address phase) of the command should be performed in dual or quad mode. The MSPI will automatically switch to the new mode after transmitting the command to the device for all DMA and XIP operations.

The MSPIn modules of the Apollo510 SoC are directly attached to the system AXI bus and are memory mapped. For example, data read/write accesses and instruction accesses (referred to as XIP for eXecute In Place) for MSPI0 are via the 0x60000000 - 0x6FFFFFFF (non-secure) address range. Each MSPIn device register space has provisions to support two independent devices, however this is not currently supported in the Apollo510 SoC. Only the DEV0* registers which configure the first device are valid. The device selected for the transaction can be specified in the PIODEV field (for PIO operations), DMADEV field (for DMA operations), or by the address range for memory-mapped operations. Each MSPIn device supports the configuration of two external devices which can be mapped within the MSPIn's memory region using the DEVnAXI configuration registers.

The MSPI modules of the Apollo510 SoC are memory mapped as follow:

- MSPI0: 0x60000000 - 0x6FFFFFFF
- MSPI1: 0x80000000 - 0x83FFFFFF
- MSPI2: 0x84000000 - 0x87FFFFFF
- MSPI3: 0x88000000 - 0x8FFFFFFF

Access to the MSPI devices is as follows:

- Cortex-M55 instruction accesses to XIP space are read-only and handled through the Cortex-M55 I-cache (which must be enabled).
- Cortex-M55 data accesses to XIP space are read/write and handled through the Cortex-M55 D-cache (which must be enabled). The GFX and the display controller access data directly through the XIP interface.
- PIO: The SoC can initiate PIO-based operations to manage basic device configuration and other low-level manual operations.
- DMA: MSPI module can autonomously transfer data between the external device and internal memory or NVM.

Note that XIP and DMA do not enforce hardware coherency, so the cache must be managed by software when performing DMA or XIP operations to regions that contain code which may be cached. In each of these modes, the MSPI module also supports data scrambling on accesses within a programmable address range having boundaries aligned to 64 kB address boundaries. Once the external devices are configured, the MSPI supports a simple DMA model, where software can program the internal (SRAM or flash) address and external device address, transfer direction, and transfer size. Once enabled, the MSPI DMA interface will move data between the system and external flash and interrupt when complete. The MSPI also supports a higher-level command queuing (CQ) protocol, where software can construct a buffer of operations in SRAM (or internal flash memory) and the MSPI will execute the series of operations autonomously. The MSPI can also power itself down at the end of DMA or CQ operations.

While each MSPI module can be used as a generic SPI device (with two chip enables), in addition to supporting serial, dual, and quad displays, it is primarily designed to support serial NAND/NOR flash memory or PSRAM memory. It is intended to be used to initialize the external memory devices and then be configured with the parameters matching the flash access characteristics. Devices can then be accessed through DMA or XIP operations with minimal software overhead.

The DMA address range has been expanded to support the larger flash and SRAM sizes, and the MSPI DMA/transfer length has been expanded to 24 bits to allow burst transactions of more than 64 kB.

The MSPI module also contains:

- A DEVnBOUNDARYn register which can be programmed to break a single long MSPI DMA into smaller transfers at periodic intervals (DMATIMELIMITn bit field).
- Address boundaries (DMABOUNDn bit field) to provide breaks in DMA for XIP traffic and satisfy the page crossing and maximum refresh times of external PSRAM devices.

> **NOTE**
>
> The DMATIMELIMITn is approximate since the MSPI will continue transmitting to the next 32-bit word boundary before disengaging on the bus. For this reason, a device requiring an 8 µs maximum transmission time should be set to have about a 7.5 µs time limit.

> **NOTE**
>
> For DMABOUNDn to properly break at a page crossing, the DMADEVADDR for the transfer must be 4-byte aligned. If a non-aligned starting edge of the transfer is required, software should manually break the transaction into two parts, with the first transaction ending on the page boundary. Failure to observe this limitation will result in data loss as the MSPIn may write 1-3 additional bytes past the boundary which will either wrap within the device's page or be discarded by the device.

### 15.2.1 Configuring MSPI as a DMA Target and a DMA Client Concurrently

A DMA deadlock may occur when there is heavy traffic of concurrent DMA accesses such as when the MSPI is used as both a "DMA client", where MSPI is sourcing or sinking data through the DMA, and a "DMA target", where MSPI is a memory-mapped source or destination for other peripheral DMA. For example, a situation may exist when the ADC is targeting a memory device through the MSPI XIPMM aperture as a the ADC sample "DMA target" at the same time that the MSPI is using DMA itself to target SSRAM or TCM. This condition may result in a DMA deadlock due to a circular dependency when the APBDMA-AXI, MSPI-AXI, MSPI-XIPDMA and APBDMA-ARBITOR states are blocking or waiting for DMA resources.

To avoid this potential problem, software should control the DMA's configuration to alternate between MSPI as a DMA client and as a DMA target so as not to allow overlap of these DMA accesses. Note that there should not be a threat of this deadlock situation when short CPU accesses such as XIP or memory mapped MSPI are occurring, due to the location of arbitration against DMA traffic.

## 15.3  Pad Configuration and Enables

For the Apollo510 SoC all four MSPI modules, MSPI0-MSPI3, support serial, dual, quad or octal mode, while MSPI0 and MSPI3 additionally support hex mode. Each MSPI module supports the external connections shown in the following tables. The columns to the right indicate which bits are used in each configuration (S=serial, D=dual, Q=quad, O=octal with CE#). Within the table, O=output pin, I=input pin, and X=bidirectional.

---

NOTE

Maximum clock speed of a specific SPI interface is dependent on the timing characterization of the chip enable (CEn) for each MSPI instance. Those speeds are specified in the Electricals chapter.

---

**Table 17: MSPI0 Pin Muxing (Serial, Dual, Quad, Octal, Hex)**

| Pin Name | Direction | GPIO | Description | S0 | S1 | D0 | D1 | Q0 | Q1 | O0 | O1 | H0 | H1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MSPI0.0 | Output | 10, 24, 56, 130, 147, 199. 200, 222* | MSPI0 CE0 | O | | O | | O | | O | | O | |
| MSPI0.1 | Output | 24, 57, 200 | MSPI0 CE1 | | O | | O | | O | | O | | O |
| MSPI0_18 | Input/Output | 45 | MSPI0 DM1/DQS1 (Hex) | | | | | | | | | X | X |
| MSPI0_17 | Input/Output | 44 | MSPI0 Data Bit 15 | | | | | | | | | X | X |
| MSPI0_16 | Input/Output | 43 | MSPI0 Data Bit 14 | | | | | | | | | X | X |
| MSPI0_15 | Input/Output | 42 | MSPI0 Data Bit 13 | | | | | | | | | X | X |
| MSPI0_14 | Input/Output | 41 | MSPI0 Data Bit 12 | | | | | | | | | X | X |
| MSPI0_13 | Input/Output | 40 | MSPI0 Data Bit 11 | | | | | | | | | X | X |
| MSPI0_12 | Input/Output | 39 | MSPI0 Data Bit 10 | | | | | | | | | X | X |
| MSPI0_11 | Input/Output | 38 | MSPI0 Data Bit 9 | | | | | | | | | X | X |
| MSPI0_10 | Input/Output | 37 | MSPI0 Data Bit 8 | | | | | | | | | X | X |
| MSPI0_9 | Input/Output | 73 | MSPI0 DM0/DQS0 (Octal/Hex) | | | | | | | X | X | X | X |
| MSPI0_8 | Output | 72 | MSPI0 CLK | O | O | O | O | O | O | O | O | O | O |
| MSPI0_7 | Input/Output | 71 | MSPI0 Data Bit 7 | | | | | | X | X | X | X | X |
| MSPI0_6 | Input/Output | 70 | MSPI0 Data Bit 6 | | | | | | X | X | X | X | X |
| MSPI0_5 | Input/Output | 69 | MSPI0 Data Bit 5 | | I | | X | | X | X | X | X | X |
| MSPI0_4 | Input/Output | 68 | MSPI0 Data Bit 4 | | O | | X | | X | X | X | X | X |
| MSPI0_3 | Input/Output | 67 | MSPI0 Data Bit 3 | | | | | X | | X | X | X | X |
| MSPI0_2 | Input/Output | 66 | MSPI0 Data Bit 2 | | | | | X | | X | X | X | X |
| MSPI0_1 | Input/Output | 65 | MSPI0 Data Bit 1 | I | | X | | X | | X | X | X | X |
| MSPI0_0 | Input/Output | 64 | MSPI0 Data Bit 0 | O | | X | | X | | X | X | X | X |

*Not pinned out on the WLCSP package

---

**Table 18: MSPI1 Pin Muxing (Serial, Dual, Quad, Octal)**

| Pin Name | Direction | GPIO | Description | S0 | S1 | D0 | D1 | Q0 | Q1 | O0 | O1 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| MSPI1.0 | Output | 33, 36, 49, 53, 93 | MSPI1 CE0 | O | | O | | O | | O | |
| MSPI1.1 | Output | 49, 54, 138 | MSPI1 CE1 | | O | | O | | O | | O |
| MSPI1_9 | Input/Output | 104 | MSPI1 DM0/DQS0 (Octal) | | | | | | | X | X |
| MSPI1_8 | Output | 103 | MSPI1 CLK | O | O | O | O | O | O | O | O |
| MSPI1_7 | Input/Output | 102 | MSPI1 Data Bit 7 | | | | | | X | X | X |
| MSPI1_6 | Input/Output | 101 | MSPI1 Data Bit 6 | | | | | | X | X | X |
| MSPI1_5 | Input/Output | 100 | MSPI1 Data Bit 5 | | I | | X | | X | X | X |
| MSPI1_4 | Input/Output | 99 | MSPI1 Data Bit 4 | | O | | X | | X | X | X |
| MSPI1_3 | Input/Output | 98 | MSPI1 Data Bit 3 | | | | | X | | X | X |
| MSPI1_2 | Input/Output | 97 | MSPI1 Data Bit 2 | | | | | X | | X | X |
| MSPI1_1 | Input/Output | 96 | MSPI1 Data Bit 1 | I | | X | | X | | X | X |
| MSPI1_0 | Input/Output | 95 | MSPI1 Data Bit 0 | O | | X | | X | | X | X |

**Table 19: MSPI2 Pin Muxing (Serial, Dual, Quad, Octal)**

| Pin Name | Direction | GPIO | Description | S0 | S1 | D0 | D1 | Q0 | Q1 | O0 | O1 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| MSPI2.0 | Output | 7, 10, 63, 157, 192 | MSPI2 CE0 | O | | O | | O | | O | |
| MSPI2.1 | Output | 10, 63, 156 | MSPI2 CE1 | | O | | O | | O | | O |
| MSPI2_9 | Input/Output | 83 | MSPI2 DM0/DQS0 (Octal) | | | | | | | X | X |
| MSPI2_8 | Output | 82 | MSPI2 CLK | O | O | O | O | O | O | O | O |
| MSPI2_7 | Input/Output | 81 | MSPI2 Data Bit 7 | | | | | | X | X | X |
| MSPI2_6 | Input/Output | 80 | MSPI2 Data Bit 6 | | | | | | X | X | X |
| MSPI2_5 | Input/Output | 79 | MSPI2 Data Bit 5 | | I | | X | | X | X | X |
| MSPI2_4 | Input/Output | 78 | MSPI2 Data Bit 4 | | O | | X | | X | X | X |
| MSPI2_3 | Input/Output | 77 | MSPI2 Data Bit 3 | | | | | X | | X | X |
| MSPI2_2 | Input/Output | 76 | MSPI2 Data Bit 2 | | | | | X | | X | X |
| MSPI2_1 | Input/Output | 75 | MSPI2 Data Bit 1 | I | | X | | X | | X | X |
| MSPI2_0 | Input/Output | 74 | MSPI2 Data Bit 0 | O | | X | | X | | X | X |

**Table 20: MSPI3 Pin Muxing (Serial, Dual, Quad, Octal, Hex)**

| Pin Name | Direction | GPIO | Description | S0 | S1 | D0 | D1 | Q0 | Q1 | O0 | O1 | H0 | H1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MSPI3.0 | Output | 27, 36, 52, 114, 131, 147, 195*, 223* | MSPI3 CE0 | O | | O | | O | | O | | O | |
| MSPI3.1 | Output | 7, 55, 195* | MSPI3 CE1 | | O | | O | O | | | O | | O |
| MSPI3_18 | Input/Output | 113 | MSPI3 DM1/DQS1 (Hex) | | | | | | | | | X | X |
| MSPI3_17 | Input/Output | 112 | MSPI3 Data Bit 15 | | | | | | | | | X | X |
| MSPI3_16 | Input/Output | 111 | MSPI3 Data Bit 14 | | | | | | | | | X | X |
| MSPI3_15 | Input/Output | 110 | MSPI3 Data Bit 13 | | | | | | | | | X | X |
| MSPI3_14 | Input/Output | 109 | MSPI3 Data Bit 12 | | | | | | | | | X | X |
| MSPI3_13 | Input/Output | 108 | MSPI3 Data Bit 11 | | | | | | | | | X | X |
| MSPI3_12 | Input/Output | 107 | MSPI3 Data Bit 10 | | | | | | | | | X | X |
| MSPI3_11 | Input/Output | 106 | MSPI3 Data Bit 9 | | | | | | | | | X | X |
| MSPI3_10 | Input/Output | 105 | MSPI3 Data Bit 8 | | | | | | | | | X | X |
| MSPI3_9 | Input/Output | 124 | MSPI3 DM0/DQS0 (Octal/Hex) | | | | | | | X | X | X | X |
| MSPI3_8 | Output | 123 | MSPI3 CLK | O | O | O | O | O | O | O | O | O | O |
| MSPI3_7 | Input/Output | 122 | MSPI3 Data Bit 7 | | | | | | X | X | X | X | X |
| MSPI3_6 | Input/Output | 121 | MSPI3 Data Bit 6 | | | | | | X | X | X | X | X |
| MSPI3_5 | Input/Output | 120 | MSPI3 Data Bit 5 | | I | | X | X | X | X | X | X | X |
| MSPI3_4 | Input/Output | 119 | MSPI3 Data Bit 4 | | O | | X | X | X | X | X | X | X |
| MSPI3_3 | Input/Output | 118 | MSPI3 Data Bit 3 | | | | | X | | X | X | X | X |
| MSPI3_2 | Input/Output | 117 | MSPI3 Data Bit 2 | | | | | X | | X | X | X | X |
| MSPI3_1 | Input/Output | 116 | MSPI3 Data Bit 1 | I | | X | | X | | X | X | X | X |
| MSPI3_0 | Input/Output | 115 | MSPI3 Data Bit 0 | O | | X | | X | | X | X | X | X |

*Not pinned out on the WLCSP package

The PADOUTEN register should be programmed to enable the proper pins for the selected mode. If using any mode with a data width less than Octal, the output clock can selectively be switched from the standard clock pin, MSPIn_8, to data bit 4 by setting the CLKOND4 bit. Typically, most serial SPI devices use a separate MOSI and MISO when operating in serial mode. The SEPIO0 or SEPIO1 bits in the DEV0CFG or DEV1CFG registers, respectively, should be set when software needs to read data from devices in serial mode, since it redirects the MISO input from pin 1 down to input data pin 0 of the MSPI's RX logic.

Table 21 below shows the required field configurations for typical MSPI operating modes. It should be noted that if the PADOUTEN_CLKOND4 is set to move the CLK pin from MSPIn_8 to MSPIn_4 (serial, dual or quad mode), then the PADOUTEN_OUTEN field must be adjusted accordingly, e.g., 0x101 becomes 0x011.

The I/O widths for data, address and instruction phases do not need to be the same in either PIO mode or XIP/DMA mode. 1-4-4 and other such mixed modes refer to the width of the instruction, address and data phases of the MSPI bus, respectively. The width(s) of the data phase, or both the address and data phases, of a command are selected by the PIOMIXED field of the MSPI_CTRLn register or by the

XIPMIXEDn field of the MSPI_DEVnXIP register, where n is device0 or device1 for the particular MSPI instance. Depending on the I/O widths supported by the MSPI instance, the data width or both data and address widths can be specified as dual, quad or octal width.

**Table 21: Required Settings for Typical MSPI Configurations**

| Mode (Data Lines and CE) | | | | | DEV0CFG_ DEVCFG0 | DEV0CFG_ SEPIO0 | DEV0XIP_ XIPMIXED0 | PADOUTEN_ OUTEN |
|---|---|---|---|---|---|---|---|---|
| Instruction | Address | Data | Separate IO | Chip Enable (CE) | | | | |
| Serial | Serial | Serial | Yes | 0 | SERIAL0 (1) | 1 | NORMAL (0) | 0x101 |
| Serial | Serial | Serial | No | 0 | SERIAL0 (1) | 0 | NORMAL (0) | 0x101 |
| Serial | Serial | Serial | Yes | 1 | SERIAL1 (2) | 1 | NORMAL (0) | 0x110 |
| Serial | Serial | Serial | No | 1 | SERIAL1 (2) | 0 | NORMAL (0) | 0x110 |
| Serial | Serial | Dual | No | 0 | SERIAL0 (1) | 0 | D2 (1) | 0x103 |
| Serial | Serial | Dual | No | 1 | SERIAL1 (2) | 0 | D2 (1) | 0x130 |
| Serial | Dual | Dual | No | 0 | SERIAL0 (1) | 0 | AD2 (3) | 0x103 |
| Serial | Dual | Dual | No | 1 | SERIAL1 (2) | 0 | AD2 (3) | 0x130 |
| Serial | Serial | Quad | No | 0 | SERIAL0 (1) | 0 | D4 (5) | 0x10F |
| Serial | Serial | Quad | No | 1 | SERIAL1 (2) | 0 | D4 (5) | 0x1F0 |
| Serial | Quad | Quad | No | 0 | SERIAL0 (1) | 0 | AD4 (7) | 0x10F |
| Serial | Quad | Quad | No | 1 | SERIAL1 (2) | 0 | AD4 (7) | 0x1F0 |
| Dual | Dual | Dual | No | 0 | DUAL0 (5) | 0 | NORMAL (0) | 0x103 |
| Dual | Dual | Dual | No | 1 | DUAL1 (6) | 0 | NORMAL (0) | 0x130 |
| Quad | Quad | Quad | No | 0 | QUAD0 (9) | 0 | NORMAL (0) | 0x10F |
| Quad | Quad | Quad | No | 1 | QUAD1 (0xA) | 0 | NORMAL (0) | 0x1F0 |
| Octal | Octal | Octal | No | 0 | OCTAL0 (0xD) | 0 | NORMAL (0) | 0x3FF |
| Octal | Octal | Octal | No | 1 | OCTAL1 (0xE) | 0 | NORMAL (0) | 0x3FF |
| Hex | Hex | Hex | No | 0 | HEX0 (0x11) | 0 | NORMAL (0) | 0x7FFFF |
| Hex | Hex | Hex | No | 1 | HEX1 (0x12) | 0 | NORMAL (0) | 0x7FFFF |

## 15.4 Board/Package Considerations for MSPI Pin Timing

### 15.4.1 Delay Step Size

Each delay step discussed in this section represents a delay of 80 to 200 ps / LSB per the DDR Delay Step Size provided in the MSPI section of the Electrical Characteristics.

### 15.4.2 SDR Mode with non-DQS

The timing parameters specified in the datasheet are based on 0-tap delay on both TX and RX delay lines, regardless of the actual number of taps being programmed. TX delay taps will delay the SCLK as illustrated by the red dotted line in Figure 21.
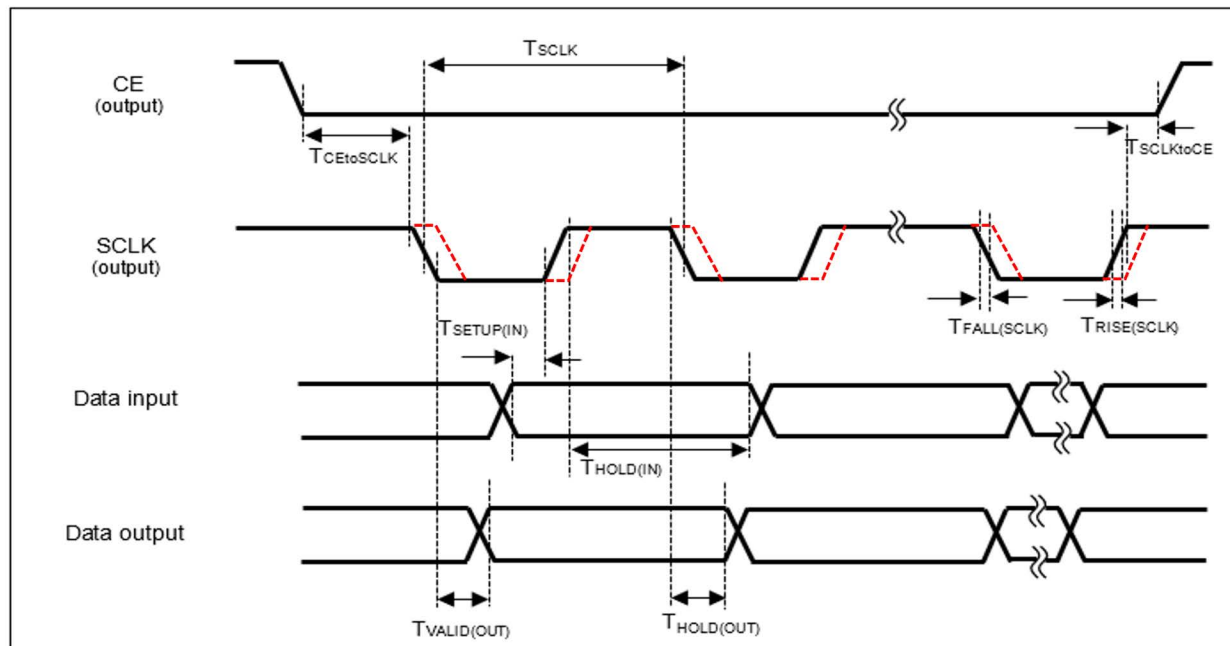


**Figure 21. SDR Mode with Non-DQS**

With more TX delay taps, Tvalid(out) will be sooner with respect to the delayed SCLK, but Thold(out) will be sooner as well.

The RX delay tap will delay the internal sampling clock (not shown in the timing diagram). With more RX delay taps, it allows a longer Tsetup(in) for external device, but a shorter Thold(in).

### 15.4.3 DDR Mode with DQS

The timing parameters specified in the MSPI section of the Electricals chapter are based on 0-tap delay on both TX and RX delay lines, regardless of the actual number of taps being programmed.

TX delay taps (TXDQSDELAY0) will delay the SCLK, as illustrated by the red dashed lines in the SCLK (output) waveform of Figure 22. With more TX delay taps, Tvalid(out) will be sooner with respect to the delayed SCLK, but Thold(out) will be sooner as well.

RX delay tap (RXDQSDELAY*0) will delay the DQS input in the chip, as illustrated by the red dashed lines in the DQS (input) waveform of Figure 22. With more RX delay taps, it allows a longer Tsetup(in) for external device, but a shorter Thold(in).
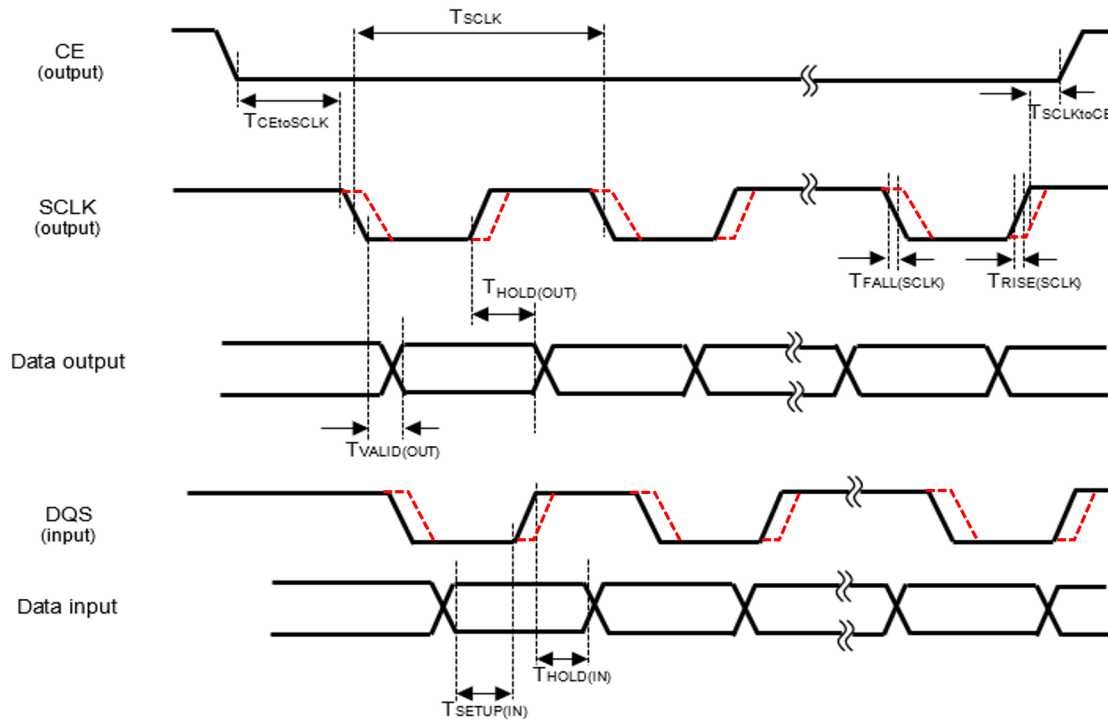
**Figure 22. DDR Mode with DQS**

Figure 23 shows DQS0 delay parameters in Octal/Hex DDR Mode with DQS, with respect to the input (RX) data byte (lower data byte for Hex) when RXDQSDELAYNEGEN = 0 and when RXDQSDELAYNEGEN = 1. See the fields in the DEVnDDR register which set these parameters.
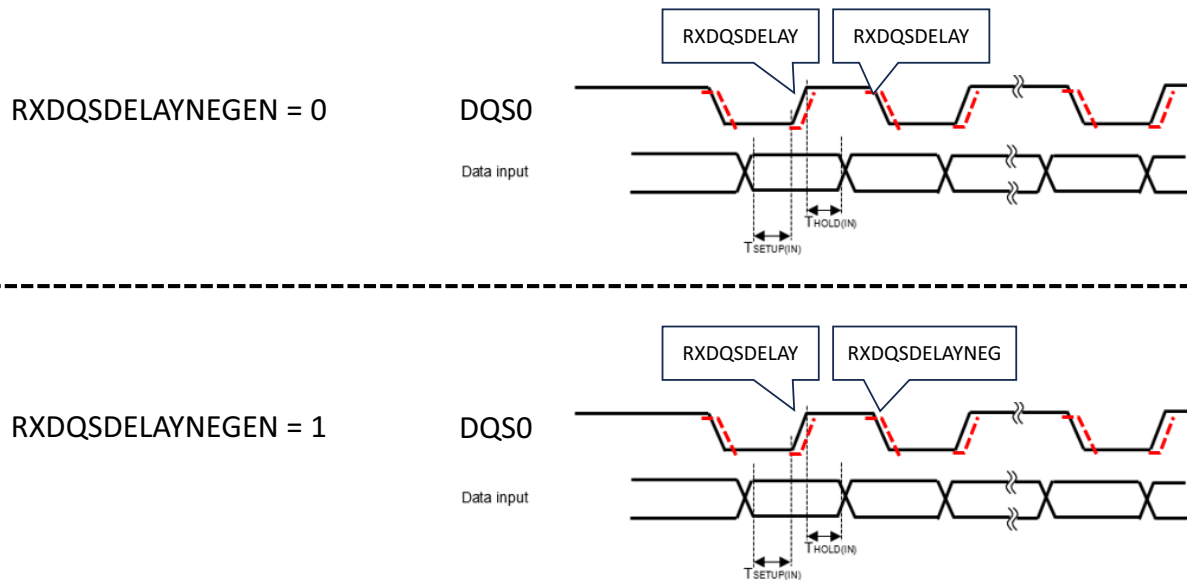


**Figure 23. Octal/Hex DDR Mode with DQS - Read Delay Lines - Lower Data Byte DQS**

Figure 24 shows DQS1 delay parameters in Hex DDR Mode with DQS, with respect to the upper input (RX) data byte under the following conditions:

1. RXDQSDELAYHIEN = 0 and RXDQSDELAYNEGEN = 0
2. RXDQSDELAYHIEN = 0 and RXDQSDELAYNEGEN = 1
3. RXDQSDELAYHIEN = 1 and RXDQSDELAYNEGEN = 0
4. RXDQSDELAYHIEN = 1 and RXDQSDELAYNEGEN = 1
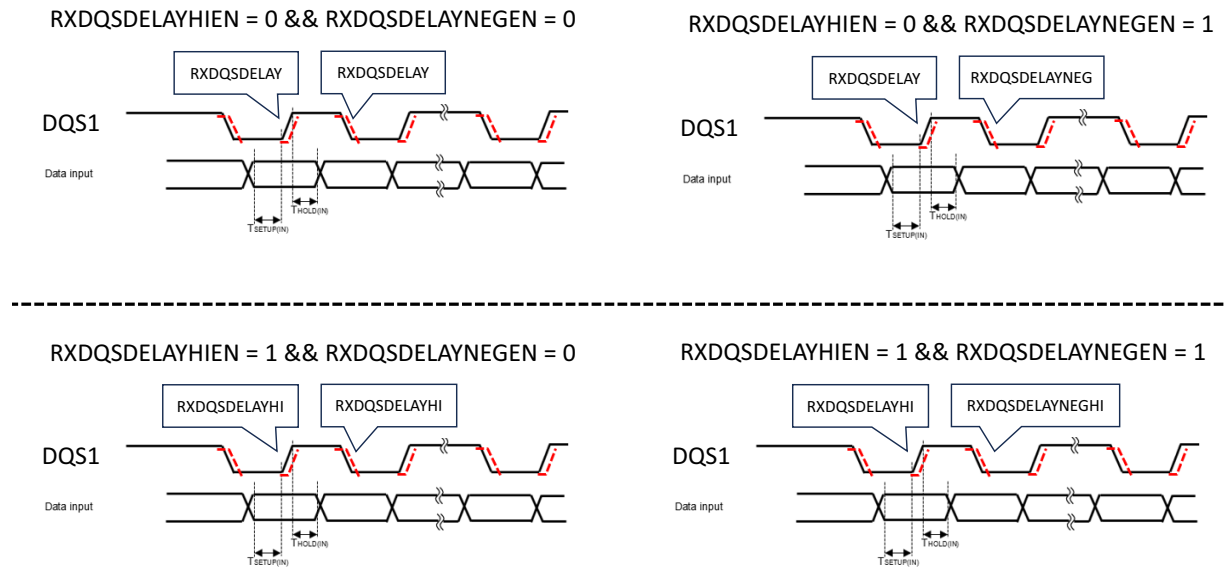


**Figure 24. Hex DDR Mode with DQS - Read Delay Lines - Upper Data Byte DQS**

## 15.5 Additional Information

Please refer to the MSPI registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 16. I²C/SPI Manager (IOM)
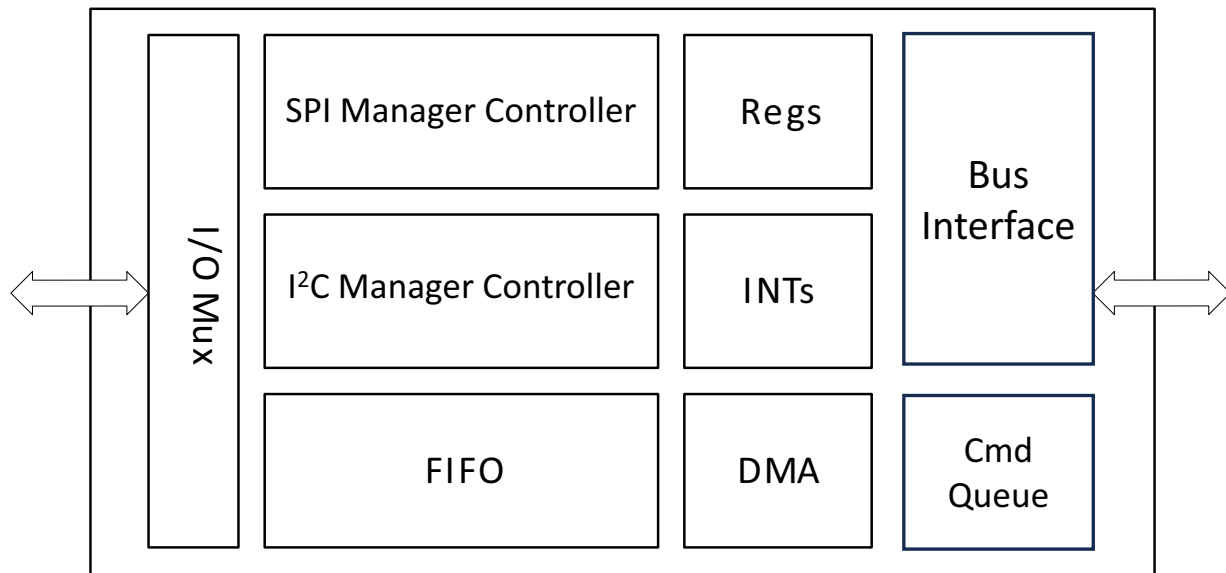


**Figure 25. IOM Block Diagram**

## 16.1 Features

The I²C/SPI Manager Module includes features shown in Figure 25 and listed below.

- Eight (8) controller instances supporting up to 4 chip selects per instance
- Up to 48 MHz clock for SPI interface
- I²C and SPI modes
- DMA with peripheral-to-memory and peripheral-to-peripheral supported
- Command Queue support

---

NOTE

There is no full-duplex transfer operation for IOM DMA on the Apollo510 SoC.

---

## 16.2 Functional Overview

The Apollo510 SoC includes 8 I²C/SPI high-speed manager modules, each of which functions as the manager of an I²C or SPI interface as selected by the IOMm_SUBMODCTRL_SMODnEN bit (m = 0 to 7, n=0 or 1) register field. A 64-byte bidirectional FIFO and a sophisticated command mechanism allow simple initiation of I/O operations without requiring software interaction.

In I²C mode the I²C/SPI Manager supports 7- and 10-bit addressing, multi-manager arbitration, interface frequencies from 1.2 kHz to 1.0 MHz and up to 4095-byte burst operations. In SPI mode the I²C/SPI Manager supports up to 4 subordinates with automatic nCE selection, 3- and 4-wire implementation, all SPI polarity/phase combinations and up to 4095-byte burst operations, with both standard embedded address operations and raw read/write transfers. Interface timing limits are as specified in the Serial Peripheral Interface (SPI) Manager Interface table of the Electrical Characteristics chapter.

---

---

**NOTE**

I$^2$C clock stretching operation is not guaranteed on this SoC. If an I$^2$C peripheral device that performs clock stretching is used, the recommendation is to perform compatibility testing with the Apollo510 I$^2$C interface.

---

ERRATUM NOTICE

In normal read operation, the FIFO threshold interrupt (THR) and associated register bit (INTSTAT_THR) are asserted when the number of valid bytes in the read FIFO (FIFOPTR_FIFOnSIZ) equals or exceeds the value set in the read threshold field (FIFOTHR_FIFORTHR), and similarly for write operation.

When doing a FIFO read transaction (DMA or non-DMA), the write threshold check is not gated off. This could trigger an incorrect/invalid write interrupt which should be ignored. If the application uses the wrong interrupt to check the read data, incorrect data processing could result.

See "ERR008: IOM: FIFO threshold interrupt incorrectly triggered" in the *Apollo510 SoC Errata List*.

---

## 16.3   Data Alignment

All data accesses between the MCU and the IOM interface are word aligned. Since the transfer size is specified in bytes, unused bytes within the word will either be discarded (for write operations) or filled with zero (read operations) to align to the next word boundary. DMA operations support a byte starting address, and the programmed DMA address does not have to be word aligned. Direct mode write operations will start transferring the least significant byte of the word (little endian style) at the current write FIFO pointer. If any remaining bytes are unused in a word at the end of the write operation, they will be discarded, and the write pointer will be set to the next word location. Direct mode read operations will store the first received byte into the least significant byte of location specified by the read FIFO pointer, and will fill any unused byte locations with zero if the transaction size is not a word multiple. The FIFO read pointer will point to the next FIFO location in the read FIFO, which will be word aligned.

### 16.3.1  Direct Mode Data Transfers

Direct mode data is enabled when DMA is disabled via the IOMn_DMACFG_DMAEN and the data transfer size (TSIZE) is greater than 0. In this mode, the MCU transfers data via direct writes or reads to registers in the IOM. The IOM maintains separate FIFO pointers for the read and write FIFOs, and updates these when a PUSH or POP register is accessed. Writing to the IOMn_FIFOPUSH register will perform a push event of the word into the FIFO and update the write pointer by 4 bytes. Only word accesses are supported to the IOM, and any unused bytes within a word will be discarded. An example of a 5 byte write transfer is shown below.

---
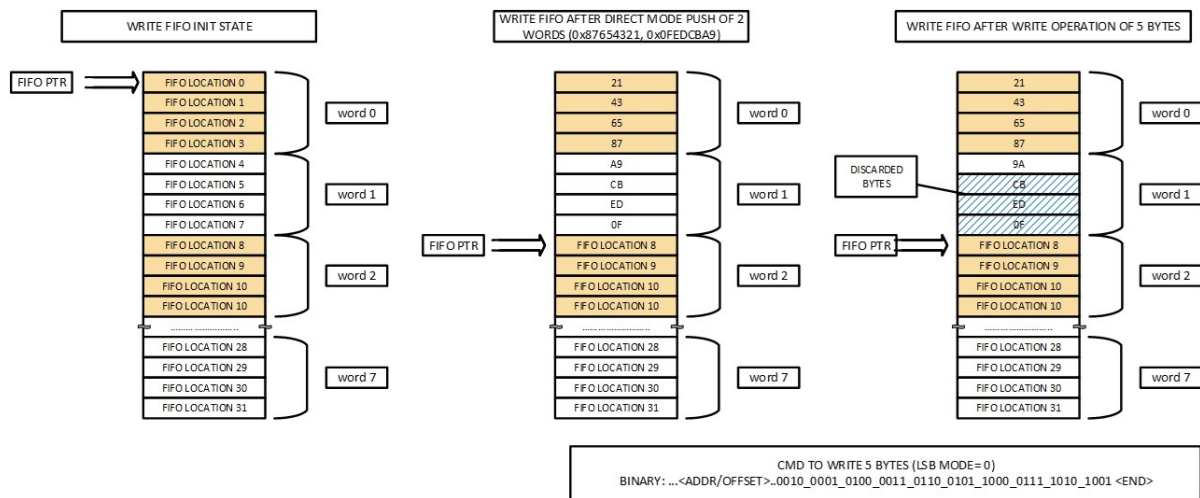
**Figure 26. Direct Mode 5-byte Write Transfer**

Reading from the IOMn_FIFOPOP register will perform a POP operation, return 4 bytes of data and advance the internal read FIFO pointer by 4 bytes. Any unused bytes within the read data will be filled with zeros and aligned to a word boundary at the end of the transaction. An example of a 5 byte read operation is shown below.
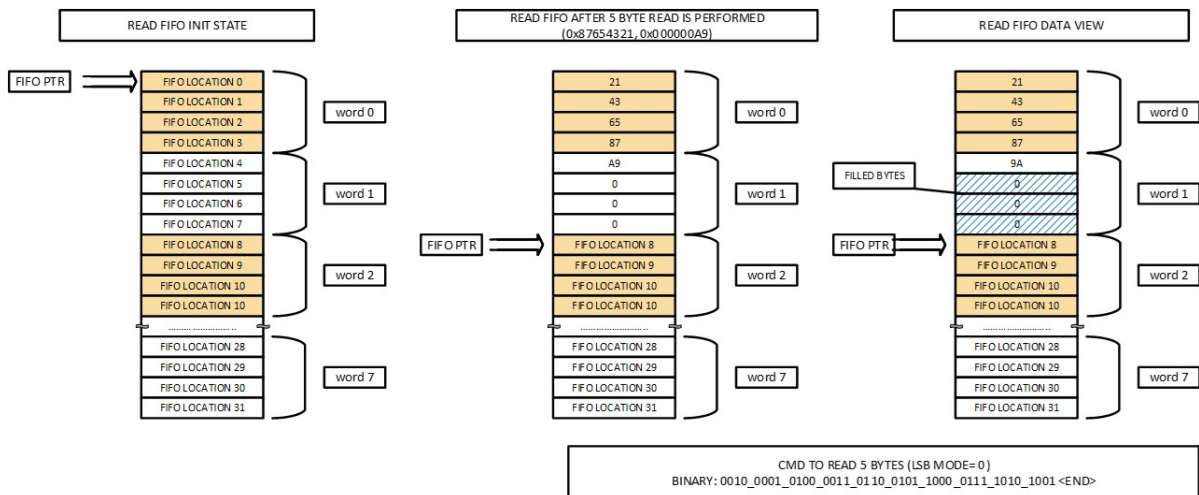


**Figure 27. Direct Mode 5-byte Read**

The IOM also supports a non-destructive POP mechanism to prevent unintended POP events from occurring. If the IOMn_FIFOCTRL_POPWR field is active (1), a write to the IOMn_FIFOPOP register will be required in order to complete the POP event. Reads will return the current data.

An active transaction will be paced by data availability and will hold the clock low if there is not enough data to continue write operations, or if the read FIFO is full during read operations. This wait condition is indicated when the IOMn_CMDSTAT_CMDSTAT field is 0x6. Once new data or FIFO locations are present, the command will continue operation automatically.

### 16.3.2 DMA Data transfers

DMA transfers are enabled by configuring the DMA related registers, enabling the DMA channel, and then issuing the command. The command will automatically fetch and store the data associated with the command without MCU intervention. The DMA channel is enabled via the IOMn_DMACFG_DMAEN field. P2M DMA operations transfer data from peripheral to memory and are used in IOM READ operations. M2P DMA operations transfer data from memory to peripheral and are used in IOM write operations. DMA transfer size is programmed into the IOMn_DMATOTCOUNT register and supports up to 4095 bytes of data transfer. The DMA transfer size is independent from the transaction size, and allows a single DMA setting to be used across multiple commands. The direction of DMA data transfer must match the command. The IOMn_DMACFG_DMAEN field enables/disables the DMA transfer capability and must be set last when configuring the DMA, generally prior to sending the command.

The DMA engine within the module will initiate a transfer of data when a trigger event occurs. There are 2 types of triggers available, threshold (THR) and command completion (CMDCMP). The THR trigger will activate when the threshold programmed into the FIFOWTHR or FIFORTHR in the IOMn_FIFOTHR register meets the data criteria. Because the MCU access to the interface is 32 bits wide, only the word count of the selected THR is used, and the low order bits of the FIFOWTHR or FIFORTHR are ignored.

During the transfer, the TOTCOUNT register is decremented to reflect the number of bytes transferred.

For IOM write operations (data written from IOM out to an external device), the THR trigger will activate when the write FIFO contains FIFOWTHR[5:2] free words. If the remaining DMA transfer size is less than this, only the needed number of words are transferred.

For IOM read operations (data read from external device), the THR trigger will activate when the read FIFO contains FIFORTHR[5:2] words of valid data. If the remaining DMA transfer size is less than the RTHR words, then the CMDCMP trigger can be enabled to transfer the remaining data. If the CMDCMP trigger is disabled, and the number of bytes in the read FIFO is greater to or equal to the current TOTCOUNT, a DMA transfer of TOTCOUNT will be done to complete the DMA operation. This mode requires that the THR trigger be enabled as well.

The CMDCMP trigger activates when the command is complete and will transfer the lesser of the TOTCOUNT or the number of bytes in the read FIFO. Note that this trigger is not needed for write operations, and the THR trigger should be used in this case. If a read operation is done, and the THR trigger is disabled, and only the CMDCMP trigger is enabled, and the transaction size is greater than the FIFO size (32 bytes), the module will hang, as there is no trigger to cause a DMA operation, and the logic will pause the interface until there is room within the read FIFO to store data.

If DMA transfer size is matched to the IOM transaction size, it is recommended to program both the FIFORTHR and FIFOWTHR to 0x10 (16 bytes) and only enable the THR trigger.

## 16.4   Transaction Initiation

To start a transaction, the IOM module must be powered up and the target external pins enabled via the GPIO module. For SPI transactions, this will generally require 4 pins to be enabled via the function select field of the PADREG registers in the GPIO module. The CEN pin for SPI transaction requires setting of the FNCSEL field of the appropriate pin, as well as the CFGREG of the corresponding pin. This also includes the setting of the default value of the CEN. This is needed to allow the IOM module to power down and not activate the CEN signal.

Once the IOM module is powered on, and the external pins configured, the IOM submodule must be enabled via the IOMn_SUBMODCTRL register. This will activate either the SPI or I$^2$C interface. Once this is complete, the submodule specific registers should be configured to set the desired mode and features. If DMA is desired, the DMA registers should also be set, with the IOMn_DMACTRL_DMAEN field set last. The registers relating to DMA operations are as follows:

- IOMn_DMATRIGEN – Sets the trigger source for starting a DMA transfer

- IOMn_DMACFG – Sets the DMA direction and enable for DMA
- IOMn_DMATOTCOUNT – Sets the total count of bytes to be transferred via the DMA operation. Recommended to match the IOMn_CMD.TSIZE field for simplicity.
- IOMn_DMATARGADDR – The source or destination address of the DMA data. Sources can be either SRAM or storage. Destination address can only be SRAM. This is the memory mapped address of the DMA data as accessed by the MCU.

 After the module setup is complete, the command register is written. This will start the IO transfer. The IOMn_CMD register contains the command itself, along with other fields used in the command, such as channel number, offset counts and transfer size. The IOM supports 2 main commands, read and write. A read command will write user selectable number of offset bytes (0 to 3), and then read IOMn_CMD_TSIZE bytes, storing the data into the read FIFO. A write command will write the user selectable number of offset bytes (0 to 3), followed by a write of IOMn_CMD_TSIZE bytes sourced from the write FIFO. Transfer sizes can be 0-4095 bytes for SPI operations and 0-512 bytes for $I^2C$ operations. The number of offset bytes for each command is specified in the IOMn_CMD_OFFSETCNT field.

## 16.5  Command Queue

The IOM module can also fetch register write data from SRAM or storage, and update the registers as if the write was performed via the MCU. Register data is stored as a doublet of 2 words. The first word is the module register address offset, word aligned. The second word is the write data value. Once enabled, the command queue (CQ) will fetch the address and perform a write to the register. If no command is started by the register write, the next doublet will be fetched by the CQ. If a command is started (write to IOMn_CMD register is done), the CQ processing will wait until the transaction is complete before fetching the next register write doublet. This is shown in the diagram below. No prefetching is done via the CQ, and the register write operations are performed in series with the transactions. This allows a predictable path for execution of commands. DMA enabled commands should be used during CQ operation, as there is no support to perform a direct mode read operation via the CQ.
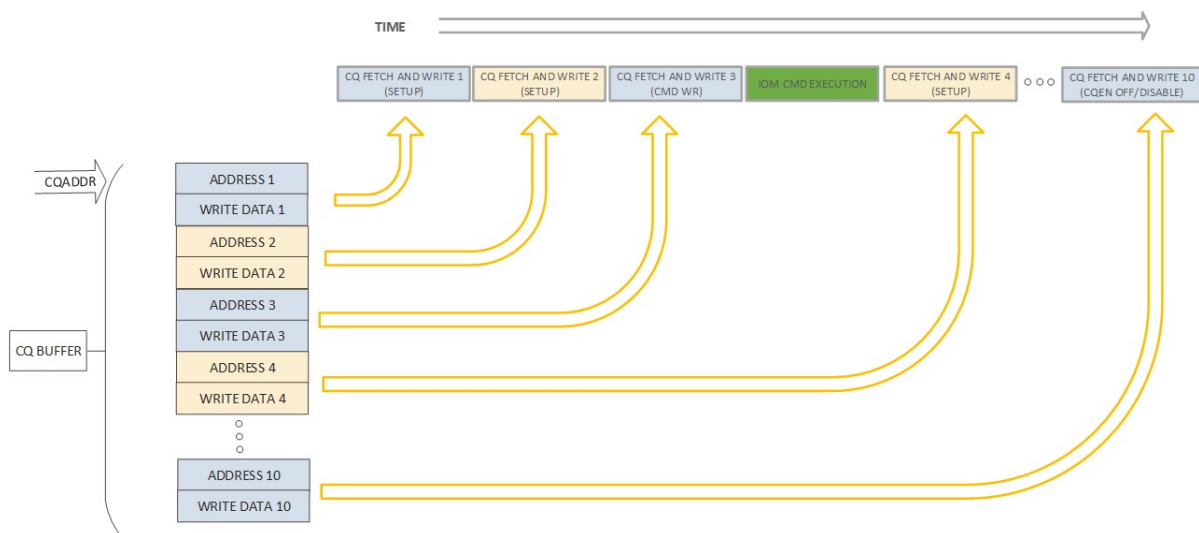


**Figure 28. Register Write Data Fetches**

The CQ starting fetch address is specified in the IOMn_CQADDR register. The CQ operation will start to fetch when the IOMn_CQCFG.CQEN field is set. This field should only be set when the IOM is idle and the FIFOs are empty. Once enabled, the CQ will continue to fetch sequentially until it encounters a pause event. A pause event can be caused by a CQ register write operation, or from external signals. This is

shown in the sequence below.
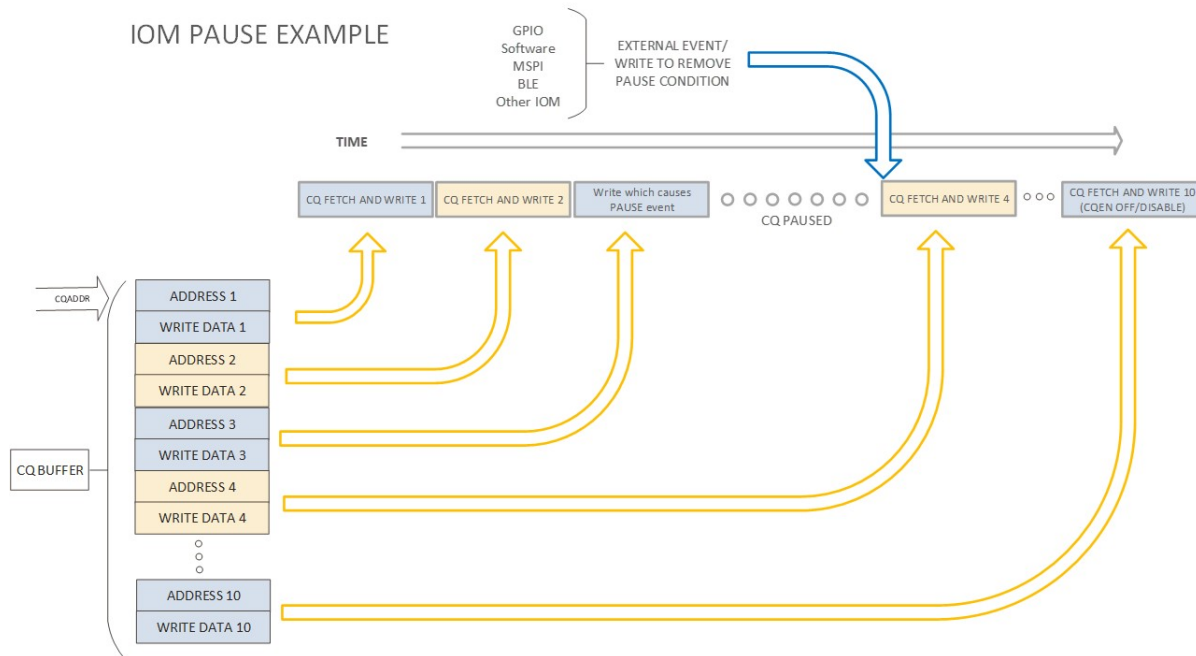


**Figure 29. IOM Pause Example**

Each pause source is independently enabled via the IOMn_PAUSEEN register. In addition to independent enable of the pause bits, there is also independent control of which pause event will signal a CQPAUSE interrupt. This is controlled through the IOMn_CQFLAGS.CQIRQMASK field.

There are 16 possible pause sources. When the value of the pause source is set, and the pause is enabled in the IOMn_PAUSEEN register, the CQ will stop fetching. The IOMn_CQADDR is updated after each fetch, and when paused, will point to the next doublet to be fetched when the pause condition is removed. The connection of the pause bits are shown below. The SW Flags are accessed via the IOMn_CQSETCLEAR register.
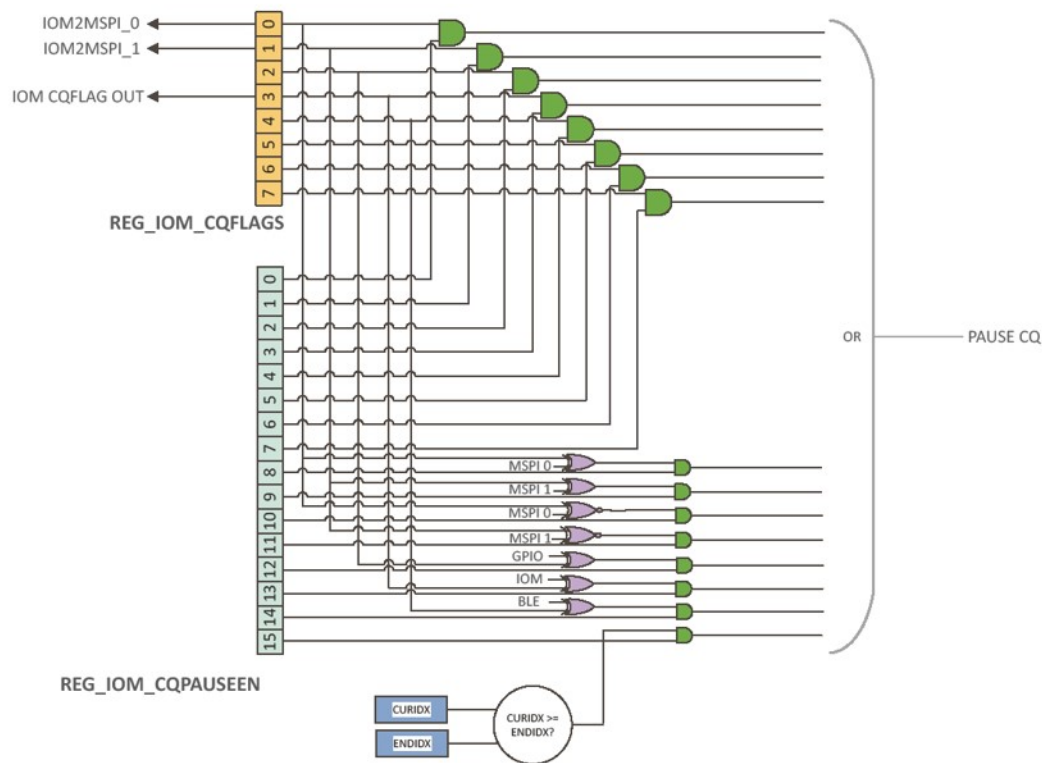
**Figure 30. CQ Pause Bit Fetching**

The first 8 pause sources (bits 7:0) are register bits which are directly writable via the MCU or through the CQ. These first 8 locations are called SW Flags. Because the CQ does not support a read-modify-write operation, special facilities are available to set, reset or toggle the SW Flags. This is accessed through the IOMn_CQSETCLEAR register. The 3 fields in this register allow a per bit set, reset or toggle of the SW Flag bits.

The next 7 pause sources (bits 14:8) use the SW Flags along with an external signal to set the pause event. The external signals are from the GPIO module, the MSPI module, or other IOM modules. On some cases, such as the MSPI interface, 4 of the SW Flags are used and combined with 2 similar signals from the MSPI module to facilitate a ping pong method of sharing 2 buffers and preventing overruns without MCU intervention.

The last pause source (bit 15) is used for index pausing. If this pause bit is enabled, the CQ will pause when the value of the IOMn_CURIDX matches the IOMn_ENDIDX. This is useful for software to be able to update the CQ buffer without causing a race condition between the CQ data buffer writes and the CQ fetches.

### 16.5.1 CQ Programming Notes

Following are some points to consider when programming command queue operation.

- Additional restrictions when using the CQ function is that the DMA must be disabled prior to writing the IOMn_CQADDR register, either from the MCU or from the CQ itself.
- For multiple commands using DMA, the DMAEN must be reset after the command is done and before the DMA registers are set for the next transaction.
- It is possible for the CQ to write the IOMn_CQADDR register during the CQ operation. The new address will take effect on the next fetch and allows the CQ to be relocated or looped.

- When starting the CQ operation, 1 doublet will be fetched regardless of the state of the pause status and bits. If any pause is active, it will take effect after the first fetch. For this reason, it is generally advisable to have a dummy register write as the first CQ doublet.

CQ write operations to SW flags used in combination with pause events 15:8 must first disable the pause enable, perform the SW flag write, then re-enable the pause enable register. SW flags 7:0 can be written without this restriction and will cause a pause immediately if activated.

## 16.6  Additional Information

Please refer to the IOM registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 17. I²C/SPI Subordinate (IOS)
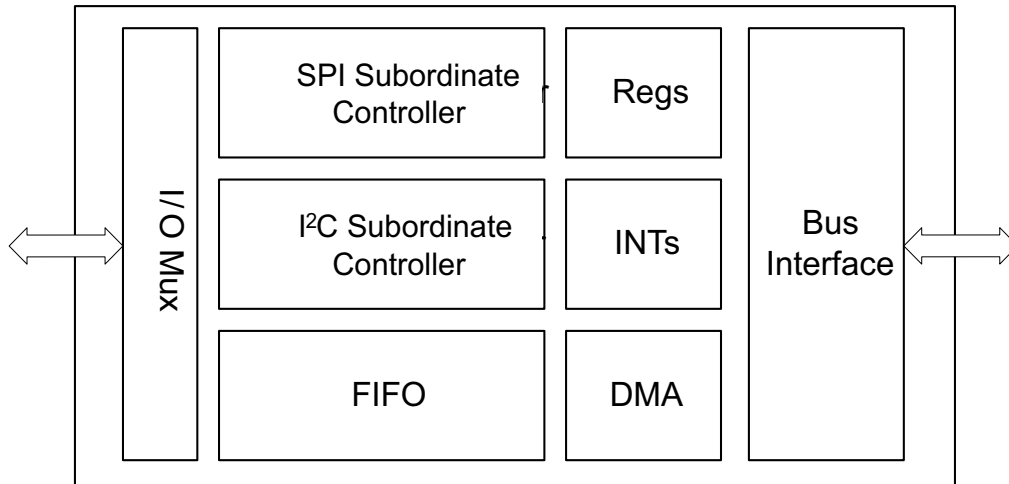


**Figure 31. Block Diagram for IOS**

## 17.1 Features

The I2C/SPI Subordinate (IOS) includes features shown in Figure 31 and listed below.

- Functions as a synchronous communications subordinate in an I²C or SPI configuration
- Supports DMA mode for either SPI or I²C interface
  - DMA total transfer count support for 64 kB data
  - Supports DMA complete and DMA error interrupts
- Supports half duplex operation
- May be placed in a sleep mode and still operate over the I/O interface
- Contains 256 bytes of Local RAM (LRAM) maintained in deep sleep mode for data/parameter storage, which can be configured into three blocks:
  - a block directly accessible via the I/O interface
  - a block which functions as a FIFO for read operations on the interface
  - a block of generally accessible RAM
- Extensive set of interrupts to control data flow and command processing, and provide alerts for writes to various locations

---

NOTE

Restrictions on IOS operation include the following:

1. Simultaneous DMA read and write operations are not supported. The DMA transfer direction should be configured properly.
2. Command Queue is not supported for IOS.
3. DMA mode is supported only for IOS FIFO mode (Host read/write to IO address 0x7F).

---

ERRATUM NOTICE

If the SPI manager pauses the SPI SCK, an Apollo510 subordinate CPU may get stuck waiting for the next SCK from the SPI manager when accessing IOS registers. The control state machine of the IOS assumes that once the interface starts an operation (read or write), it finishes it and the bus is held off until that happens because only one operation can take place with the LRAM at a time. The read or write request is asserted for one interface clock cycle, so if the clock stops the request will be held and the IOS (and MCU) will be stalled.

See "ERR010: IOS: FIFO read gets stuck/stalled" in the *Apollo510 SoCErrata List*.

ERRATUM NOTICE

When configured as a SPI subordinate using the IOS module, the Apollo510 does not tri-state the MISO pin when CE is de-asserted. Instead, the MISO pin is driven static low when CE is driven high. This condition exists for the half-duplex as well as the full-duplex IOS instances.

See "ERR011: IOS: MISO line is not tri-stated when CE is de-asserted" in the *Apollo510 SoC Errata List*.

ERRATUM NOTICE

There are two conditions which may cause FIFO mode to fail because of a conflict with the host-side register space:

- FIFO_BASE is set to 0x10 (halfway into the LRAM) when FIFO reads are to be executed.
- FIFOPTR is set within the range 0x78 to 0x7F.

See "ERR012: IOS: FIFO mode failures with certain FIFO_BASE and FIFOPTR settings" in the *Apollo510 SoC Errata List*.

ERRATUM NOTICE

The half-duplex IOS0 Subordinate module's interrupt (SLINT) is a function select for GPIO162. However, this GPIO pad is not bonded out to an external pin on the Apollo510 or the Apollo510B BGA. Therefore, the IOS0's supported and built-in interrupt mechanism is not operational.

See "ERR013: IOS: Subordinate interrupt (SLINT) is not bonded out to external pin" in the *Apollo510 SoC Errata List*.

## 17.2  Functional Overview

The Apollo510 SoC includes one instance of the I2C/SPI Subordinate module referred to as IOS. In I2C mode the subordinate supports fully configurable 7- and 10-bit addressing with interface timing limits as specified in the Inter-Integrated Circuit (I2C) Interface section of the Electricals chapter. In SPI mode, the Subordinate supports all polarity/phase combinations and interface frequencies as specified in the Serial Peripheral Interface (SPI) Subordinate Interface section of the Electricals. The IOS operates in an independent fashion, so that the device may be placed in a sleep mode and still receive operations over the I/O interface. The subordinate may be configured to generate an interrupt on specific references.

The IOS in the Apollo510 SoC includes DMA features and supports half duplex transfers in either direction. It contains 256 bytes of RAM which is only accessible when the module is enabled. This RAM may be flexibly configured into three spaces:

▪ A block directly accessible via the I/O interface
▪ A block which functions as a FIFO for read operations on the interface
▪ A block of generally accessible RAM used to store parameters during deep sleep mode

## 17.3  Half Duplex DMA Features

Figure 32 shows the basic environment of the half duplex configuration with read and write paths shown.
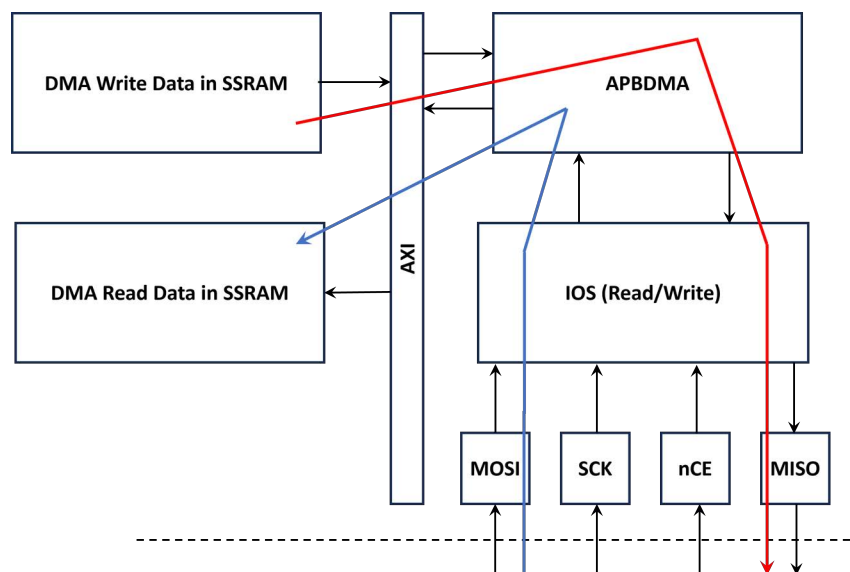


**Figure 32. IOS DMA Transfers**

### 17.3.1  Half Duplex DMA Reads

Half duplex DMA reads use the data flow path shown by the blue arrows in Figure 32. Once the DMA transfer is configured in the IOS module (shown as IOSFD1), the Host initiates a write transfer, generating the clock SCK and the chip enable nCE with data coming in on the MOSI pad. The transfer address must be the FIFO address 0x7F. The data flows through the IOS FIFO in the LRAM, and the APBDMA Controller reads that data over the APB and writes it into the target memory (such as SSRAM0) via the AXI bus.

### 17.3.2  Half Duplex DMA Writes

Half duplex DMA writes use the data path shown by the red arrows in Figure 32. In the figure IOSFD0 is shown executing the DMA transfer, but the IOS can be configured to do so as well. IOS makes DMA

requests and the APBDMA controller transfers data from the source memory (such as SSRAM1) via the AXI bus to the IOS FIFO in the LRAM. The Host initiates a read transfer, generating the clock SCK and the chip enable nCE with data flowing out of IOS on the MISO pad. The transfer address must be the FIFO address 0x7F.

## 17.4   IOS Pin Connections

The pads of IOS which supports half duplex SPI and I2C on the Apollo510 are as follows:

- SCK (FNCSEL = 10); SCL (FNCSEL = 9) - GPIO11
- MOSI (FNCSEL = 13); SDAWIR3 (FNCSEL = 12) - GPIO52
- MISO (FNCSEL = 10) - GPIO83
- nCE (FNCSEL = 9) - GPIO13
- SLINT (FNCSEL = 0) - GPIO162

## 17.5   IOS DMA Registers

Several new registers have been added to IOS to support DMA operations, while existing IOS registers are unchanged. Please refer to the IOS registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

## 17.6   IOS LRAM Configuration for DMA

The first step in configuring the IOS is to allocate the LRAM memory regions. Since we often want the Host to be able to transfer commands into the IOS, we typically assign a small part of memory, i.e., 8 or 16 bytes, as the Direct region by setting FIFOBASE to 1 or 2. We can then use the REGACC interrupts to signal the CPU that there is a command available. If we don't need any generally accessible LRAM area, which is by far the most common case, the remaining LRAM is typically assigned to the FIFO area by setting FIFOMAX to 32. All DMA transfers will go through the FIFO area. If the Direct Area of memory is not used, FIFOBASE can be set to 0 to maximize the size of the FIFO.

The IOS contains 256 bytes of LRAM. FIFOMAX must be 32 or less, and FIFOBASE must be small enough to configure at least 48 bytes of FIFO space.

## 17.7   Executing a DMA Operation

An operation is initiated by the Host signaling to the MCU that a transfer will be executed, i.e., sending a command. This requires an interrupt to be generated to the CPU, waking it up if necessary, which can be done in several ways:

- Generating a hardware interrupt to the SLINT input pad from a Host pin.
- Generating a hardware interrupt to a GPIO input pad from a Host pin.
- Generating a software interrupt by writing to the HOST_WCS register in the IOS at address offset 0x7B.
- Generating a software interrupt by writing to the Direct memory area with the appropriate REGACC interrupt(s) configured. This is the recommended method.

Once the command has been detected, software on the CPU determines what the desired DMA parameters are - transfer direction, length and target address. Software then configures the DMA transfer in APB_DMA and IOS and starts the DMA operation. At this point the CPU must send a handshake signal to the Host indicating that the IOS is ready for the transfer. This handshake can be through a GPIO output pad to a Host pin.

When the Host detects that the IOS is ready for a transfer, it initiates a burst transfer (read or write) to the first address in the FIFO area, which is typically address offset 8 or 16. The Host then streams as much data as specified. The IOS transfers this data to or from the Host, and initiates DMA transfers as necessary to empty/refill the FIFO in the LRAM. The addresses in the LRAM rolls over from FIFOMAX-9 to

FIFOBASE. When the DMATOTCOUNT_TOTCOUNT reaches 0, the operation terminates and a DMASTAT_DMACPL interrupt is generated.

After the operation is complete, the system waits for the receipt of another command from the Host and the process continues.

## 17.8 Interrupt Processing

Two new interrupts, DCMP and DERR, are added to the existing IOSLAVEINT interrupt register. DCMP indicates the end of a DMA transfer, and the XCMPRR/XCMPRF and XCMPWR/XCMPWF interrupts are used as the Transfer Complete interrupts for read and write bursts respectively.

The DCMP interrupt (and the DMACPL status bit) is set when the DMA transfer completes, and the XCMPRF/XCMPWF interrupts (and the DMASTAT_XFRCMP status bit) is set when the IO transfer completes on the interface. In general software should enable the DCMP interrupt on P2M transfers (Peripheral-to-Manager, or DMA reads), and should enable the XCMPRF interrupt on M2P transfers (Manager-to-Peripheral, or DMA writes). The expected behavior is described in Table 22:

**Table 22: DMA Interrupt Processing**

| DMADIR | Host-TOT | Interrupt | DMACPL | UNDFL | TOTCOUNT | DMAFSIZE | XCMPWF |
|--------|----------|-----------|--------|-------|----------|----------|--------|
| 0:P2M | == | DCMP | 1 | 0 | 0 | 0 | 1 |
| 0:P2M | < | DCMP | 1 | 0 | Note 1 | 0 | 1 |
| 0:P2M | > | DCMP | 1 | 0 | 0 | Note 2 | Note 2 |
| 1:M2P | == | XCMPRF | 1 | 0 | 0 | 0 | 0 |
| 1:M2P | < | XCMPRF | Note 3 | 0 | Note 4 | Note 5 | 0 |
| 1:M2P | > | XCMPRF | 1 | 1 | 0 | Note 5 | 0 |

Table Notes:

Note 1: (Initial TOTCOUNT - Final TOTCOUNT) = number of bytes written to memory.

Note 2: If XCMPRF = 1, if DMAFOVF = 0, DMAFSIZE contains the number of valid bytes in the FIFO which were received from the Host but not written to memory. If DMAFOVF = 1, the number of valid bytes in the FIFO is unknown because the FIFO has been overwritten. It is possible that for a very slow interface (like I2C) executing a very long transfer, XCMPF will still be 0 when the DCMP interrupt is serviced. If software needs the remaining FIFO information it must wait until XCMPRF becomes 1 to read DMAFSIZE and DMAFOVF.

Note 3: If the Host interface is very fast, the internal busses are heavily utilized and the CPU responds very quickly to interrupts. There is a very small probability that DMACPL is not asserted when the XCMPRF interrupt is received. But in any case, it will be set soon after that.

Note 4: (Initial TOTCOUNT - Final TOTCOUNT) = number of bytes read from memory. This will be more data than was actually transferred on the Host interface and is not a useful value.

Note 5: DMAFSIZE is not valid.

The XFRCOUNT register field contains the number of bytes actually transferred on the interface on a DMA read, and the number of bytes actually transferred + 1 on the interface on a DMA write. This is very useful in determining any "extra" data in the FIFO at the completion of an operation.

## 17.9   Additional Information

Please refer to the IOS registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 18. Full Duplex SPI Subordinate (IOSFD)
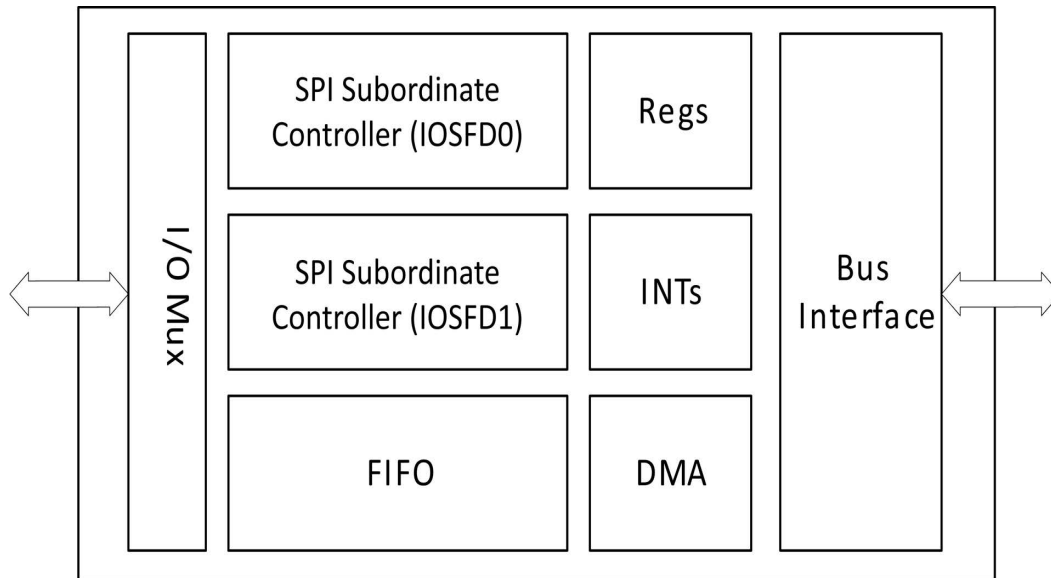


**Figure 33. Full Duplex IOS SPI Subordinate Controller Pair (IOSFD0 / IOSFD1)**

## 18.1 Features

The Apollo510 SoC includes two instances of the Full DuplexSPI Subordinate and both instances are designed to work in tandem to enable full duplex (FD) operation. Features of the controller pair are shown in Figure 33 and listed below.

- Functions as a synchronous communications subordinate in a SPI configuration
- Supports DMA mode
  - DMA total transfer count support for 64 kB data
  - Supports DMA complete and DMA error interrupts
- Supports full duplex operation when both instances are used in combination
- Each instance can work individually to provide half duplex operation
- May be placed in a sleep mode and still operate over the I/O interface
- Each controller contains 64 bytes of Local RAM (LRAM) maintained in deep sleep mode for data/parameter storage, which can be configured flexibly as three operational blocks
- Extensive set of interrupts to control data flow and command processing, and provide alerts for writes to various locations

> **NOTE**
>
> Restrictions on IOSFD operation:
>
> 1. Only one IOSFD can be operational at a time when operating in half duplex mode due to interface signals of both instances using the same set of IO pads.
> 2. Simultaneous DMA read and write operations are not supported for an IOSFD configured for half duplex. The DMA transfer direction should be configured properly.
> 3. Command Queue is not supported for IOSFD.
> 4. DMA mode is supported only for IOSFD FIFO mode (Host read/write to IO address 0x7F).

> **ERRATUM NOTICE**
>
> When configured as a SPI subordinate using the IOS module, the Apollo510 does not tri-state the MISO pin when CE is de-asserted. Instead, the MISO pin is driven static low when CE is driven high. This condition exists for the half-duplex as well as the full-duplex IOS instances.
>
> See "ERR011: IOS: MISO line is not tri-stated when CE is de-asserted" in the *Apollo510 SoC Errata List*.

## 18.2  Functional Overview

The Apollo510 SoC includes two instances of the Full Duplex SPI Subordinate module, IOSFD0 and IOSFD1, which in combination support full duplex operation. Only SPI mode is supported in the IOSFD module, and it SPI mode supports all polarity/phase combinations and interface frequencies as specified in the Serial Peripheral Interface (SPI) Subordinate Interface sub-section of the Electricals. Each IOSFD module can function as a half duplex SPI interface, but in this case only one of them (either IOSFD0 or IOSFD1) can be used at any one time. The device may be placed in a sleep mode and still receive operations over the I/O interface. The subordinate may be configured to generate an interrupt on specific references.

Each IOSFD instance contains 64 bytes of LRAM which is only accessible when the module is enabled. This RAM may be flexibly configured into three spaces:

- A block directly accessible via the I/O interface.
- A block which functions as a FIFO for read operations on the interface. This area is required because DMA transfers must utilize the FIFO.
- A block of generally accessible RAM used to store parameters during deep sleep mode. Although this can be configured for use in IOSFD, it is not recommended.

## 18.3  Full Duplex DMA Features

The IOSFDn interface connection options are configured on the following pads with the functions selected with the Function Select (FNCSEL) values as shown:

- GPIO0 (FNCSEL = 1) - SLFDSCK - the input clock which is connected to both IOSFD0 and IOSFD1.
- GPIO1 (FNCSEL = 1) - SLFDMOSI - the input data which is connected to both IOSFD0 and IOSFD1.

- GPIO1 (FNCSEL = 2) - SLFDSDAWIR3 - the bidirectional data in/out which is connected to IOSFD0.
- GPIO1 (FNCSEL = 10) - SLFD1WIR3 - the bidirectional data in/out which is connected to IOSFD1.
- GPIO2 (FNCSEL = 1) - SLFDMISO - the output data which is connected to IOSFD0.
- GPIO2 (FNCSEL = 10) - SLFD1MISO - the output data which is connected to IOSFD1.
- GPIO3 (FNCSEL = 1) - SLFDnCE - the input nCE which is connected to both IOSFD0 and IOSFD1.
- GPIO4 (FNCSEL = 1) - SLFDINT - the interrupt out which is connected to IOSFD0.
- GPIO4 (FNCSEL = 10) - SLFD1INT - the interrupt out which is connected to IOSFD1.

The output data from IOSFD1 instead of IOSFD0 may be connected to GPIO2 by selecting SLFD1MISO. The interrupt output from IOSFD1 instead of IOSFD0 may be connected to GPIO4 (by selecting SLFD1INT). This allows half duplex testing of both IOSFD modules.

In normal operation, GPIO2 is connected to IOSFD0 and IOSFD0 is configured to execute a DMA write transaction (DMACFGn_DMADIR = 0). IOSFD1 is configured to execute a DMA read transaction (DMACFGn_DMADIR = 1). Separate memory addresses (in DMATARGADDR_TARGADDR) must be configured in the two IOSFD modules, but the transfer length DMATOTCOUNT_TOTCOUNT may or may not be the same in both modules. Once the modules are enabled, they both respond independently to a full duplex transfer on the interface, with the read and write DMA operations proceeding in parallel. DMA completion and error interrupts will typically occur in both modules simultaneously.

Configuring a full duplex DMA environment involves several steps:

1. Configure the pinmux to connect the four IOSFD pads to both IOSFD0 and IOSFD1, with IOSFD0 supplying the MISO signal and executing the DMA write and IOSFD1 executing the DMA read.

   [Note that the DMA direction is the opposite of the interface direction, which may be confusing. On a DMA write, data is read over the interface (using MISO) and written to memory. On a DMA read, data is read from memory and written over the interface (using MOSI).]

2. Set DMADIR to 0 in IOSFD0 and to 1 in IOSFD1.
3. Set the DMACFG_FRCDMA bit to 1 in both modules. Set the DMACFG_FRCRDWRT bit to 1 in IOSFD0 and to 0 in IOSFD1.
4. Set the DMACFG_PADBYTEEN bit to 1 in IOSFD0 and configure the desired DMACFG_PADBYTE value. These fields can also be set in IOSFD1 but have no effect on DMA reads.

The Host must send an address byte at the beginning of the transfer, but the R/W bit is ignored and each IOSFD module executes the desired transfer.

Figure 34 shows the basic environment and test structures. A full duplex configuration is shown.

## 18.4 Full Duplex DMA Transfers in SPI Mode

Full duplex transfers are implemented by configuring a DMA read in one IOSFD (IOSFD1 in Figure 34) and a DMA write in the other IOSFD module (IOSFD0 in Figure 34). The configurations force IOSFD1 to treat the transfer as a read and IOSFD0 to treat the transfer as a write, independent of the R/W bit. Both modules are connected to SCK and nCE so they execute a synchronized transfer. The Host sends the DMA read data on MOSI (to IOSFD1) and receives the DMA write data on MISO (from IOSFD0).
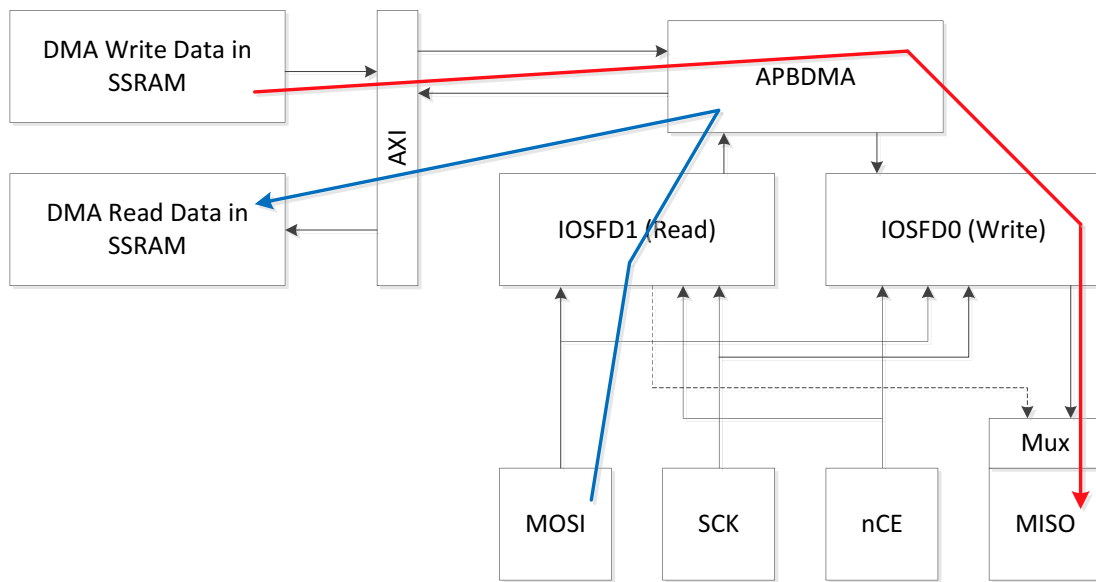
**Figure 34. Full Duplex IOS DMA Transfers in SPI Mode**

## 18.5  Half Duplex DMA Reads in SPI Mode

Half duplex DMA reads use the data flow path shown by the blue arrow in Figure 35.
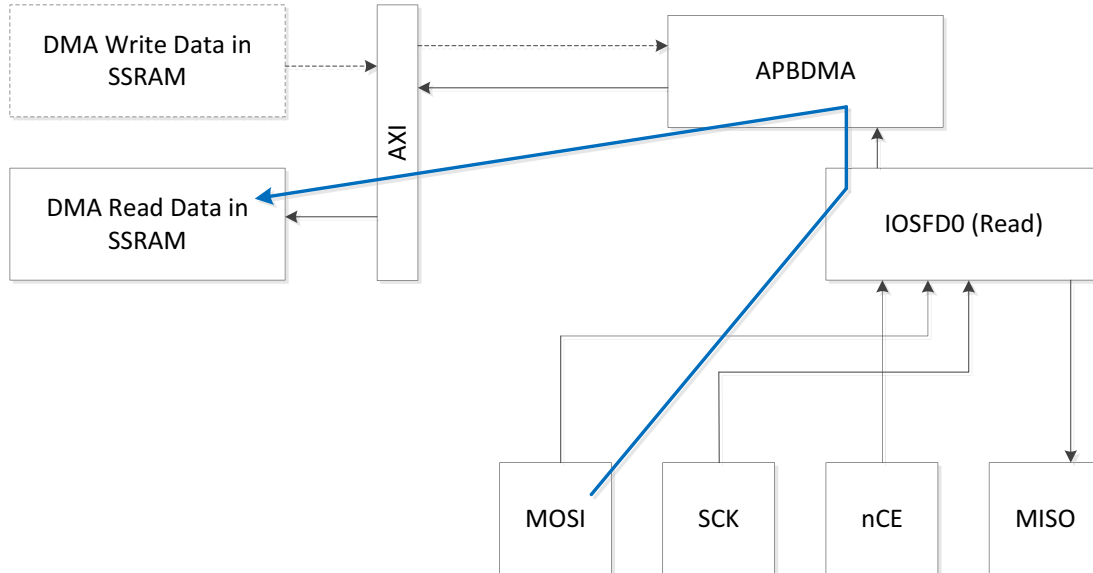


**Figure 35. Half Duplex IOS DMA Reads in SPI Mode**

Once the DMA transfer is configured in the IOSFDn module (either can be used although IOSFD0 is shown in the diagram), the Host initiates a write transfer, generating the clock SCK and the chip enable nCE with data coming in on the MOSI pad. The transfer address must be the FIFO address 0x7F. The data flows through the IOSFD FIFO in the LRAM, and the APBDMA Controller reads that data over the APB and writes it into the target memory (such as SSRAM0) via the AXI bus.

## 18.6  Half Duplex DMA Writes in SPI Mode

Half duplex DMA writes use the data path shown by the red arrow in Figure 36. In this example IOSFD0 is executing the DMA transfer, but either IOSFDn can be configured to do so.
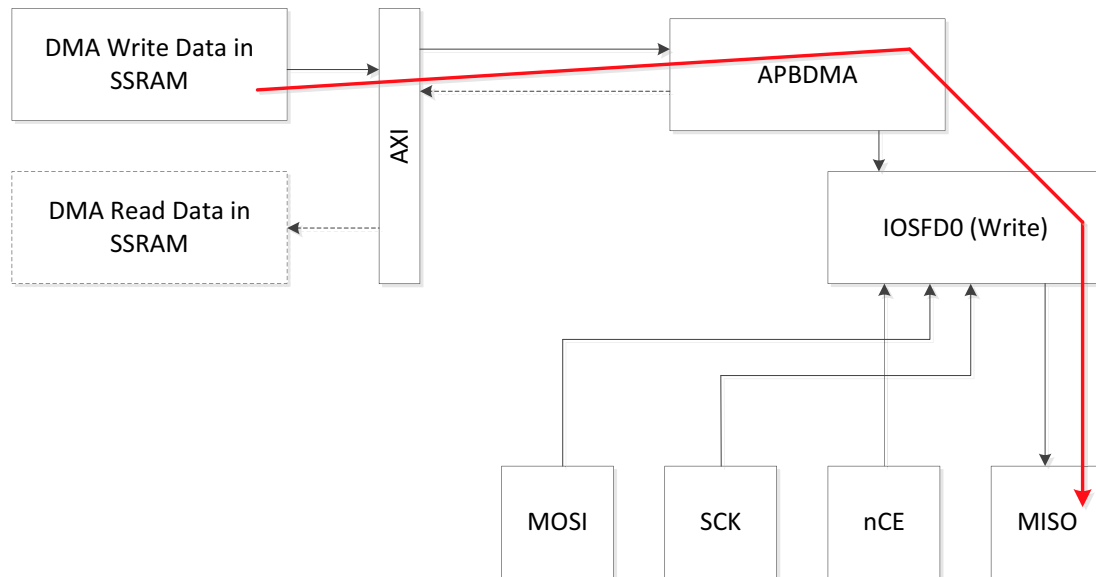


**Figure 36. Half Duplex IOS DMA Writes in SPI Mode**

IOSFD0 makes DMA requests and the APBDMA controller transfers data from the source memory (such as SSRAM1) via the AXI bus to the IOSFD0 FIFO in the LRAM. The Host initiates a read transfer, generating the clock SCK and the chip enable nCE with data flowing out of IOSFD0 on the MISO pad. The transfer address must be the FIFO address 0x7F.

## 18.7  IOSFD DMA Registers

Several registers are included in the IOSFDn register set to support DMA operations. Please refer to the IOSFD registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

## 18.8  IOSFD LRAM Configuration for DMA

The first step in configuring the IOSFD is to allocate the LRAM memory regions. Since we often want the Host to be able to transfer commands into the IOSFD, we typically assign a small part of memory, i.e., 8 or 16 bytes, as the Direct region by setting FIFOBASE to 1 or 2. We can then use the REGACC interrupts to signal the CPU that there is a command available. If we don't need any LRAM area, which is by far the most common case, the remaining LRAM is typically assigned to the FIFO area by setting FIFOMAX to 32. All DMA transfers will go through the FIFO area. If the Direct Area of memory is not used, FIFOBASE can be set to 0 to maximize the size of the FIFO.

The IOSFD modules, IOSFD0 and IOSFD1, contain 64 bytes of LRAM each. FIFOMAX must be 8 or less, and FIFOBASE must be small enough to configure at least 48 bytes of FIFO space.

## 18.9   Executing a DMA Operation

An operation is initiated by the Host signaling to the MCU that a transfer will be executed, i.e., sending a command. This requires an interrupt to be generated to the CPU, waking it up if necessary, which can be done in several ways:

- Generating a hardware interrupt to the SLINT input pad from a Host pin.
- Generating a hardware interrupt to a GPIO input pad from a Host pin.
- Generating a software interrupt by writing to the HOST_WCS register in the IOSFD at address offset 0x7B.
- Generating a software interrupt by writing to the Direct memory area with the appropriate REGACC interrupt(s) configured. This is the recommended method.

Once the command has been detected, software on the CPU determines what the desired DMA parameters are - transfer direction, length and target address. Software then configures the DMA transfer in APB_DMA and IOSFD and starts the DMA operation. At this point the CPU must send a handshake signal to the Host indicating that the IOSFD is ready for the transfer. This handshake can be through a GPIO output pad to a Host pin.

When the Host detects that the IOSFD is ready for a transfer, it initiates a burst transfer (read or write) to the first address in the FIFO area, which is typically address offset 8 or 16. The Host then streams as much data as specified. The IOSFD transfers this data to or from the Host, and initiates DMA transfers as necessary to empty/refill the FIFO in the LRAM. The addresses in the LRAM rolls over from FIFOMAX-9 to FIFOBASE. When the DMATOTCOUNT_TOTCOUNT reaches 0, the operation terminates and a DMASTAT_DMACPL interrupt is generated.

After the operation is complete, the system waits for the receipt of another command from the Host, and the process continues.

## 18.10   Interrupt Processing

Two new interrupts, DCMP and DERR, are included in the IOSLAVEINT interrupt register. DCMP indicates the end of a DMA transfer, and the XCMPRR/XCMPRF and XCMPWR/XCMPWF interrupts are used as the Transfer Complete interrupts for read and write bursts respectively.

The DCMP interrupt (and the DMACPL status bit) is set when the DMA transfer completes, and the XCMPRF/XCMPWF interrupts (and the DMASTAT_XFRCMP status bit) is set when the IO transfer completes on the interface. In general software should enable the DCMP interrupt on P2M transfers (Peripheral-to-Manager, or DMA reads), and should enable the XCMPRF interrupt on M2P transfers (Manager-to-Peripheral, or DMA writes). The expected behavior is described in Table 23:

**Table 23: DMA Interrupt Processing**

| DMADIR | Host-TOT | Interrupt | DMACPL | UNDFL | TOTCOUNT | DMAFSIZE | XCMPWF |
|--------|----------|-----------|--------|-------|----------|----------|--------|
| 0:P2M  | ==       | DCMP      | 1      | 0     | 0        | 0        | 1      |
| 0:P2M  | <        | DCMP      | 1      | 0     | Note 1   | 0        | 1      |
| 0:P2M  | >        | DCMP      | 1      | 0     | 0        | Note 2   | Note 2 |
| 1:M2P  | ==       | XCMPRF    | 1      | 0     | 0        | 0        | 0      |
| 1:M2P  | <        | XCMPRF    | Note 3 | 0     | Note 4   | Note 5   | 0      |
| 1:M2P  | >        | XCMPRF    | 1      | 1     | 0        | Note 5   | 0      |

Table Notes:

Note 1: (Initial TOTCOUNT - Final TOTCOUNT) = number of bytes written to memory.

Note 2: If XCMPRF = 1, if DMAFOVF = 0, DMAFSIZE contains the number of valid bytes in the FIFO which were received from the Host but not written to memory. If DMAFOVF = 1, the number of valid bytes in the FIFO is unknown because the FIFO has been overwritten. It is possible that for a very slow interface executing a very long transfer, XCMPF will still be 0 when the DCMP interrupt is serviced. If software needs the remaining FIFO information it must wait until XCMPRF becomes 1 to read DMAFSIZE and DMAFOVF.

Note 3: If the Host interface is very fast, the internal busses are heavily utilized and the CPU responds very quickly to interrupts. There is a very small probability that DMACPL is not asserted when the XCMPRF interrupt is received. But in any case, it will be set soon after that.

Note 4: (Initial TOTCOUNT - Final TOTCOUNT) = number of bytes read from memory. This will be more data than was actually transferred on the Host interface and is not a useful value.

Note 5: DMAFSIZE is not valid.

The XFRCOUNT register field contains the number of bytes actually transferred on the interface on a DMA read, and the number of bytes actually transferred + 1 on the interface on a DMA write or in the write IOSFD in a full duplex transfer. This is very useful in determining any "extra" data in the FIFO at the completion of an operation.

For a full duplex transfer, the terminating interrupt should be DCMP on the IOSFD that is implementing the DMA read, which is typically iOSFD1. The status values in each IOSFD is a function of the configured DMA length, which may be different in each IOSFD, and the Host transfer length, which is the same in both IOSFDs.

## 18.11  Full Duplex DMA Processing

At the completion of each DMA operation, after reading all relevant status information, DMAEN should be set low. This will reset many of the status register bits and prepare the module for the next operation. All configuration fields should be written and all status and interrupt bits cleared *before* DMAEN is set to 1.

For full duplex transfers, the following sequence should be used. This assumes IOSFD0 is implementing the DMA write and IOSFD1 is implementing the DMA read.

1. Make sure DMAEN is clear in IOSFD0 and IOSFD1.
2. Configure each IOSFD instances for its respective transfer. Clear any writable status bits like DMAUNDFL and DMAERR.
3. Enable the DCMP interrupt on IOSFD1. This will be the last interrupt status set on a correct transfer.
4. Set DMAEN for both IOSFD instances.
5. Wait for the DCMP interrupt from IOSFD1 (not from IOSFD0). At this point the Host should have read the correct data (not all PADBYTES).
6. The XCMPRF and DCMP interrupts should be set for IOSFD0, and the XCMPWF interrupt should be set for IOSFD1. These should be checked.
7. Read any status (TOTCOUNT, XFRCOUNT, etc.) to determine the transfer result.
8. Clear DMAEN for both IOSFD instances.

After DMAEN is set on the DMA read side (IOSFD1 in the typical case), the first transfer executed by the Host MUST be the actual full duplex transfer. The Host must not attempt to access any other interface address, such as the Direct Address space or the IOINT module.

Once the interface transfer is complete, the Host may then access other offsets within the IOSFD. These accesses (reads and writes) will all go to the IOSFD configured for the DMA write transfer (IOSFD0 in the typical case). No interface accesses will occur to the other IOSFD module.

## 18.12  Additional Information

Please refer to the IOSFD registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 19. Universal Asynchronous Receiver/Transmitter (UART)
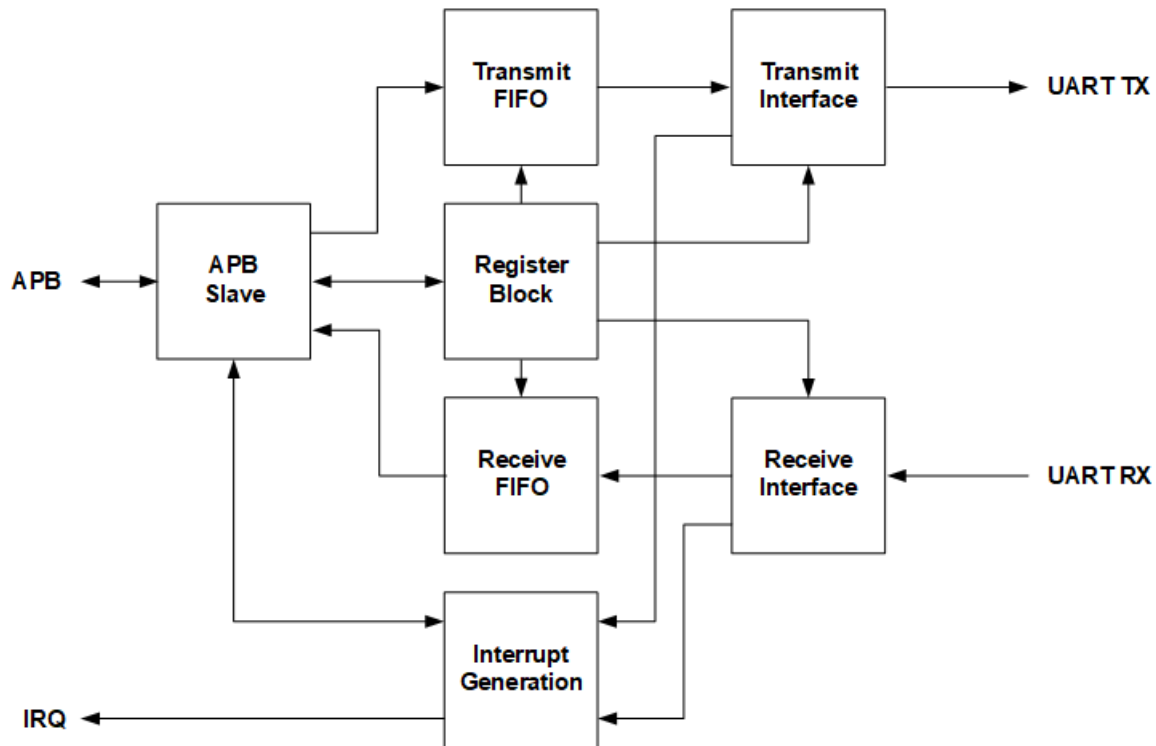


**Figure 37. UART Block Diagram**

## 19.1 Features

The Universal Asynchronous Receiver/Transmitter (UART) includes features shown in Figure 37 and listed below.

There are four (4) Universal Asynchronous Receiver/Transmitter (UART) instances on the Apollo510 SoC. The UART Module includes the following key features:

- Operates independently, allowing the MCU to enter a low power sleep mode during communication
- 32 x 8 transmit FIFO and 32 x 12 receive FIFO to reduce MCU computational load
- Programmable baud rate generator capable of achieving a maximum rate of 3 Mbps
- Fully programmable data size, parity, and stop bit length
- Programmable hardware flow control
- Support for full-duplex and half-duplex communication
- Supports DMA (half-duplex mode only)
- Loop-back functionality for diagnostics and testing

---

NOTE

To be able to receive data reliably at 3 Mbps baud rate on UART1, UART2 or UART3, the UART clock source must be set to SYSPLL by setting the UARTn_CR_CLKSEL = PLL_CLK. Also, the PCLK for the specific UART in use must be set to always on by setting the MCUCTRL_D2ASPARE_UARTnALWAYSON bit. When bit 22 to 25 of this register is set, it will force the PCLK clock enable for UART0 to UART3, respectively, to be always on. UART0 is not affected by this limitation.

To receive at baud rates lower than 3 Mbps, it is not necessary to set this bit or to use the SYSPLL as the UART's clock source.

---

NOTE

Since UART DMA is only supported in half-duplex mode, two UART instances are required (one for TX and one for RX) to support full-duplex DMA. In this full-duplex mode, the TX instance should be configured to use 2 stop bits and the one for RX should use 1 stop bit setting. The external UART TX needs to use 2 stop bits setting.

---

## 19.2   Functional Overview

Each UART converts parallel data written through the APB Subordinate port into serial data which is transmitted to an external device. It also receives serial data from an external device and converts it to parallel data, which is then stored in a buffer until the CPU reads the data.

The UART Module includes a programmable baud rate generator which is capable of operating at high baud rates. An interrupt generator optionally sends interrupts to the CPU core for transmit, receive and error events.

Internally, the UART Module maintains two FIFOs. The transmit FIFO is 1-byte wide with 32 locations. The receive FIFO is 12-bits wide with 32 locations. The extra four bits in the receive FIFO are used to capture any error status information that the Apollo510 SoC needs to analyze.

---

NOTE

In accordance with the PC16550 UART specifications, the CTS interrupt triggers (when enabled) if the state of the CTS input changes from low to high or from high to low. Additionally, if the CTS input is high during the initialization and enabling of the UART, the CTS interrupt will be triggered. Software should handle such interrupt events appropriately.

---

## 19.3   Additional Information

Please refer to the UART registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

---

# 20. Universal Serial Bus (USB)
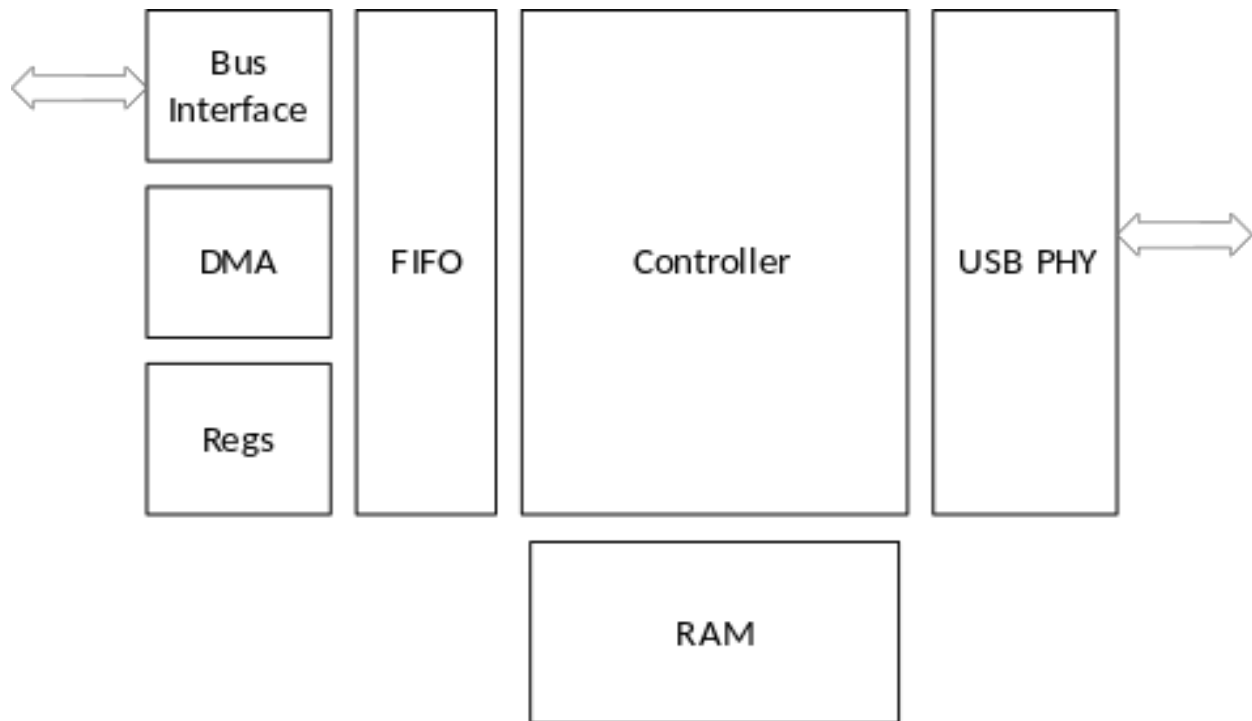


**Figure 38. USB Block Diagram**

## 20.1 Features

The Universal Serial Bus (USB) Subsystem includes features shown in Figure 38 and listed below.

- USB 2.0 FS/HS Device Mode
- Dynamic FIFO sizing: 4 kB total FIFO
- IN endpoints: 5
- OUT endpoints: 5
- IN bulk packet splitting supported
- OUT bulk packet combining supported
- Concurrent DMA supported for all IN/OUT BULK endpoints
- Soft connect/disconnect supported
- Suspend mode supported
- Supported classes
  - Communication Device Class (CDC)
  - Mass Storage Class (MSC)
  - Device Firmware Upgrade (DFU) Class
  - Human Interface Device (HID) Class
- Charging Support
  - Battery Charging 1.2 (BC1.2) supported for battery charger detection
  - Selected 3rd-party proprietary charger detection supported

---

## 20.2  Functional Overview

The USB subsystem provides support for USB full-speed (12 Mbps) and high-speed (480 Mbps) interface. This interface is primarily used for bulk data transfer, firmware updates and charging detect.

The USB controller supports up to 5 IN / 5 OUT endpoints plus 1 control. The FIFO sizing for each endpoint is dynamically configurable up to 4 kB, with the restriction that IN and OUT port sizing must be up to half the total FIFO size. The controller also supports DMA transfers to/from system memory.

The Apollo510 SoC has an integrated USB 2.0 PHY with support for Suspend Mode operation. Battery charger detection is supported within the PHY to enable battery charge algorithm execution and control of the external battery charge / power management IC. The charger detection supports Battery Charging Specification 1.2 (BC1.2) as well as the majority of proprietary (non-BC1.2 standard) chargers.

---

ERRATUM NOTICE

An output pulse on the D+ line while VDDUSB0P9 and/or VDDUSB33 are/is rising may be interpreted by a host as a connect event immediately followed by a disconnect one, causing the host USB SW to report about the unexpected disconnect from Apollo510. Such a report/message could be safely ignored for the USB-compliant hosts which would attempt USB device re-enumeration. An enumeration attempt performed when all USB power rails are stable succeeds.

See "ERR031: USB: Induced D+ output pulse may cause unintended disconnect" in the *Apollo510 SoC Errata List*.

---

## 20.3  Additional Information

Please refer to the USB and USBPHY registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 21. Secure Digital Input Output (SDIO)
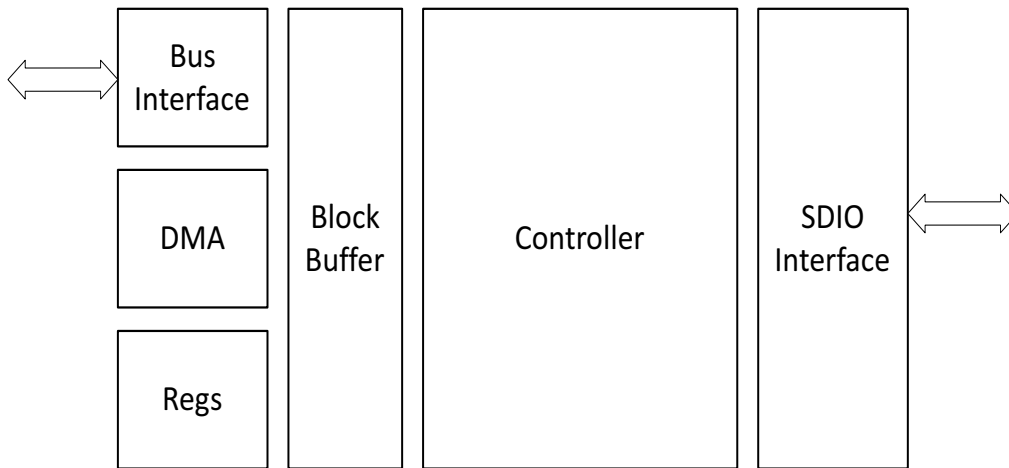


**Figure 39. SDIO Block Diagram**

## 21.1 Features

The Secure Digital Input Output (SDIO) Module includes features shown in Figure 39 and listed below.

- 2x SDIO Controller instances allowing for concurrent eMMC and SDIO interface support
- SDIO card specification Version 3.0
- Host clock rate variable between 0 and 96 MHz
- Up to 48 MB/s data rate using 4 parallel data lines (SDR50/DDR50 mode)
- Transfers data in 1-bit and 4-bit SD modes
- Transfers data in SDR50 or DDR50 modes
- SDIO0 and SDIO1 transfer data in 8-bit eMMC mode for 96 MB/s maximum transfer rate
- Cyclic Redundancy Check CRC7 for command and CRC16 for data integrity
- Variable-length data transfers
- Performs Read Wait Control, Suspend/Resume operation

## 21.2  Functional Overview

The SDIO host controller provides support for higher bandwidth device transfer. A typical application is for WiFi IC connectivity. The SDIO controller supports up to 2 kB block buffering as well as dedicated DMA controller support to provide maximum host offload. The DMA algorithm supported is the Advanced DMA version 2 (ADMA2) which allows for flexibility in memory allocation. The controller interface supports a programmable DLL to allow for timing tuning for optimal windowing.

---

NOTE

The default value for the Card Detect (CD) and the Write Protect (WP) pins for both SDIO channels is GPIO127, which is also the (only) GPIO option for the SDIO1 DATA2 line. The SDIFnWP and SDIFnCD fields in the GPIO_SDIF0CDWP and GPIO_SDIF1CDWP registers must be set (before SDIO channel initialization) so as not to cause any pin conflict.

---

## 21.3  Additional Information

Please refer to the SDIO registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 22. Display Subsystem



**Figure 40. Block Diagram of the Display Subsystem**

## 22.1 Features

The Apollo510 SoC's integrated Display Subsystem contains several smart tools and functions to compose multiple graphics, improving image quality and contributing significantly to the reduction of the SoC power consumption. The Display Controller supports composition features, a wide range of display interfaces and advanced proprietary framebuffer compression technology. The core is designed to lessen the workload of the Graphics Processing Unit (GPU) and the host processor (CPU), and minimize the memory bandwidth in GPU-less systems. Multiple layers can be scaled, clipped, positioned and composed on the final display by overlaying graphics, cursors or application windows, with or without transparency.

The Display Subsystem is represented by the block diagram shown in Figure 40. Its key features include the following:

- Supports MIPI DSI 1.2[1]
  - Up to 2 lanes (D0N/D0P, D1N/D1P)
  - 768 Mbps / lane, 1.536 Gbps total
  - Low power mode
  - Command mode
- Supports up to 1920 x 1080[2] resolution
- Up to 24-bit color
- Supports Memory-in-Pixel displays with optional fast-forward support
- Configurable stride/pitch enabling panning and clipping
- Up to 4 layers with alpha blending and scaling
- Multiple Input Surface formats
  - RGBX8888, XRGB8888 (32-bit)
  - RGB888 (24-bit mode)
  - RGBX5551, RGB565 (16-bit)

---

1. The DPI-2 and DBI-Type B interfaces are not offered on the WLCSP package.
2. Ignoring frame-rate, the Display Controller can support up to 1920 x 1080 resolution. The resulting frame rate at any particular resolution depends upon display interface speed, bus-fabric bandwidth and complexity of the graphics assets and related operations. Typically a resolution of 640 x 480 at 60 frames per second can be supported for most applications.

---

- YUYV (32-bit, 2 pixels / word)
- RGB232, LUT8, Grayscale (8-bit)
▪ Display Interfaces
- Parallel RGB
- MIPI DBI-Type B and DBI-Type C (Display Bus Interface)
- MIPI DPI-2
- Serial formats 3-,4-beat and two phase serial 12-bit RGB
- Programmable HSYNC, VSYNC, DE, pixel clock polarity
- 3-4 Wire SPI (Serial Peripheral Interface) bus
- DualSPI (Dual Serial Peripheral Interface) bus
- QuadSPI (Quad Serial Peripheral Interface) bus
- Parallel MIP display
▪ Compressed frame buffer and frame buffer decompression support:
- 4-bit
- 6-bit (with/without Alpha)
- 12-bit (with/without Alpha)
▪ Adaptive sync
▪ Adaptive brightness
▪ Powerful composition
▪ Programmable size, offset and format per layer
▪ Programmable stride/pitch enabling panning and clipping
▪ Per layer palette
▪ Global or per layer gamma correction
▪ Dithering for better results on 18-bit displays
▪ Programmable event interrupt
▪ Smart DMA support

---

NOTE

The DSI's D1N/D1P data lane are not offered on the WLCSP package.

---

NOTE

24-bit, 16-bit and 8-bit color formats work only when the frame buffer resolution is set to a multiple of 4, as is the case for the startX and startY coordinates for all TSC color formats. If the panel resolution is not a multiple of 4, then the frame buffer should be sized slightly larger than the panel resolution to make it a multiple of 4.

Since the frame buffer size is just the input of each layer, the start position of the layer as the output to the panel can be any desired value. And since it is just the frame buffer that must be larger, the desired output resolution is not restricted (i.e., can be a partial frame output or full frame of resolution that is less than the frame buffer size).

---

ERRATUM NOTICE

Failures in the form of incorrect data being read or the AXI bus hanging occur if the Display Controller (DC) is fetching data from multiple memories for its multiple layers and one of the memories is MRAM.

See "ERR002: DC: Limitations with MRAM AXI bus read transactions" in the *Apollo510 SoC Errata List*.

ERRATUM NOTICE

Cannot free the DBI when reading a register from the DSI display and the display is not connected. When executing a DSI read command, the correct sequence is:

1.  DBI sends read command.
2.  DSI IP translates this command to DSI and sends this command to display.
3.  When DSI gets returned data from display, it sets DBI_DataValid and returns data to DBI.
4.  DBI receives data and enters idle state.

But when the display is not connected or is damaged, the DSI cannot get returned data and ACK from the display. It will enter an error state, and will not set DBI_DataValid. Then the DBI cannot be freed and it will stay in the busy state. The DC status register will indicate "DBI/SPI CS is busy" (bit 14 at 0x400A00FC).

See "ERR005: DSI: Cannot free DBI when reading DSI register" in the *Apollo510 SoC Errata List*.

ERRATUM NOTICE

When the DSI PHY is powered up, a VSS power switch produces an additional 8 mA of current leakage. The VSS power switch should be opened when DSI PHY is powered up but there is a PN diode in the PMOS switch that creates a forward bias current from VDDF to VSS. Note that this current leakage issue is fixed in silicon revision B2.

See "ERR006: DSI: DSI PHY draws 8 mA when powered up" in the *Apollo510 SoC Errata List*.

NOTE

The DPI-2, Parallel RGB and DBI-Type B display interfaces are not offered on the WLCSP package.

## 22.2  Functional Overview

The Display Subsystem offers flexibility for system designers to select the number of layers, functionality of composition modules and processing methods tailored to their requirements. This section provides a high-level description of the main functions, the proprietary TSC™ compression/decompression, the supported color formats, and the display format interfaces.

### 22.2.1  Display Controller Functionality

The Display Controller is a dedicated hardware block allowing the offload of display interface and control functions. The configuration register file controls the operation of the display controller. Timing parameters are programmable and data are fetched for each layer by a dedicated DMA engine. Additional functions include:

▪ Gamma adjustment
▪ Dithering application
▪ RGB conversion to YCbCr/YUV format

#### 22.2.1.1 Display Controller Clocking
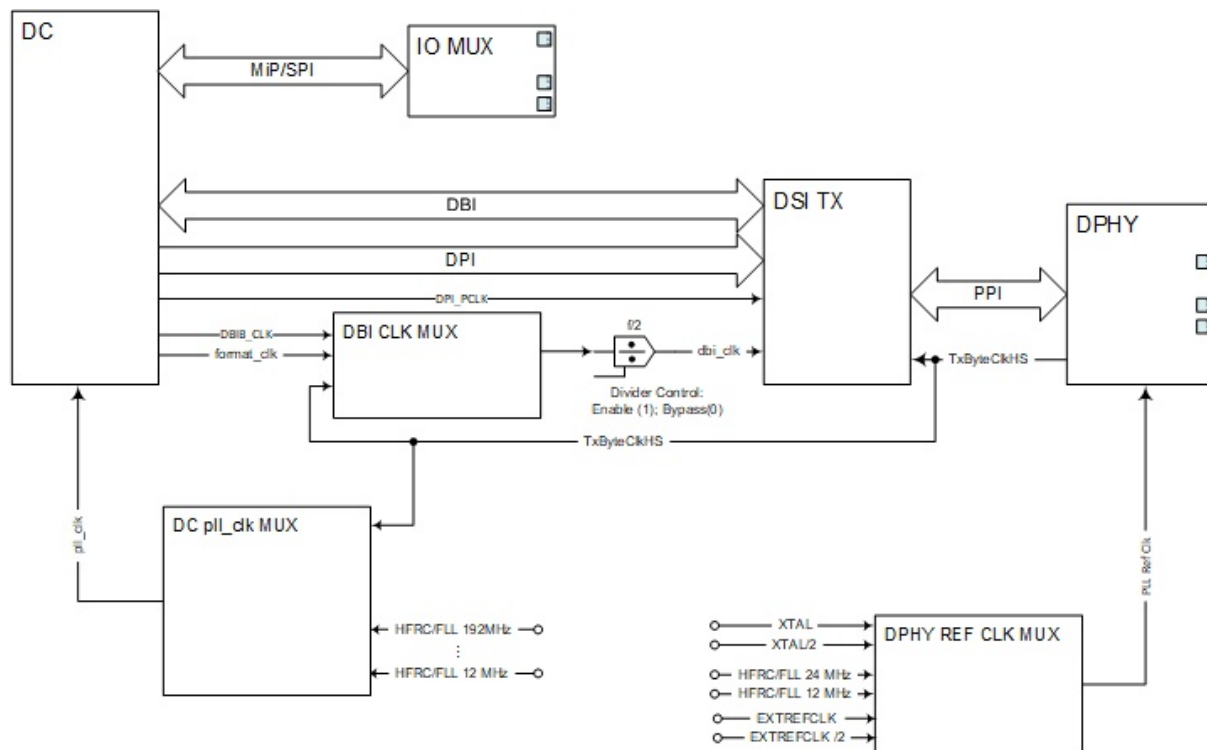


**Figure 41. Display Controller Clocking Block Diagram**

#### 22.2.1.2 Layer Overlays

The Display Controller supports up to four layers sourced from different memory regions. Each layer can have separate color modes, alpha blending, and filtering attributes. The main control registers for each layer is the LAYERn_MODE (n = 0-3) registers. Optionally, the layer can support scaling.

### 22.2.1.3 Blending Modes

During blending process, a translucent foreground color (current layer) with a background color (previous layer) are combined and a new blended color is produced. Foreground color's translucency may range from completely transparent to completely opaque. If the foreground color is completely transparent, the blended color will be the background color and if the foreground color is completely opaque, the blended color will be the foreground color. When the translucency ranges in between, the blended color is computed as a weighted average of the foreground and background colors.

### 22.2.1.4 Palette/Gamma Correction

The Display Controller supports palette color mapping and gamma correction per layer.

### 22.2.1.5 Dithering

Dithering is the process of degrading the color image with a method that tries to produce better results than information truncation. Dithering is used to create the illusion of "color depth" in images with a limited color palette. In a dithered image, colors that are not available in the palette are approximated by a diffusion of colored pixels from within the available palette. The human eye perceives the diffusion as a mixture of the colors within it.

## 22.2.2 Display Serial Interface (DSI) Controller
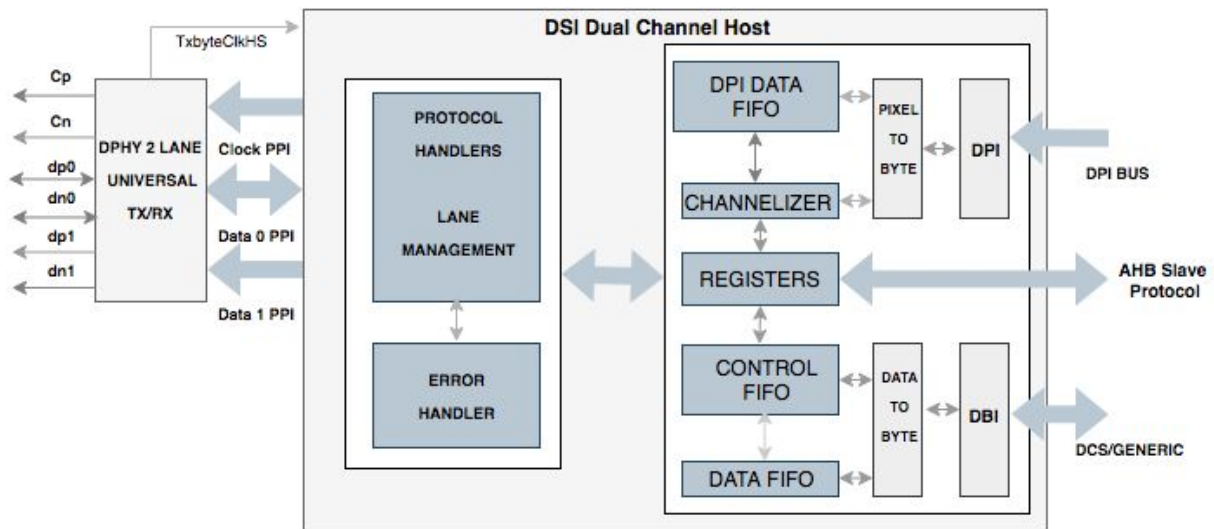


**Figure 42. Block Diagram for the DSI Controller**

The Display Serial Interface (DSI) Controller includes features shown in Figure 40 and listed below.

- Two MIPI lanes
- 768 Mbps per MIPI lane support

The DSI bus on the Apollo510 SoC is a type of serial bus that enables transfer of data between a transmitter device and a receiver device. The DSI device has a point-to-point connection with DSI devices via D-PHYs as shown in Figure 43.
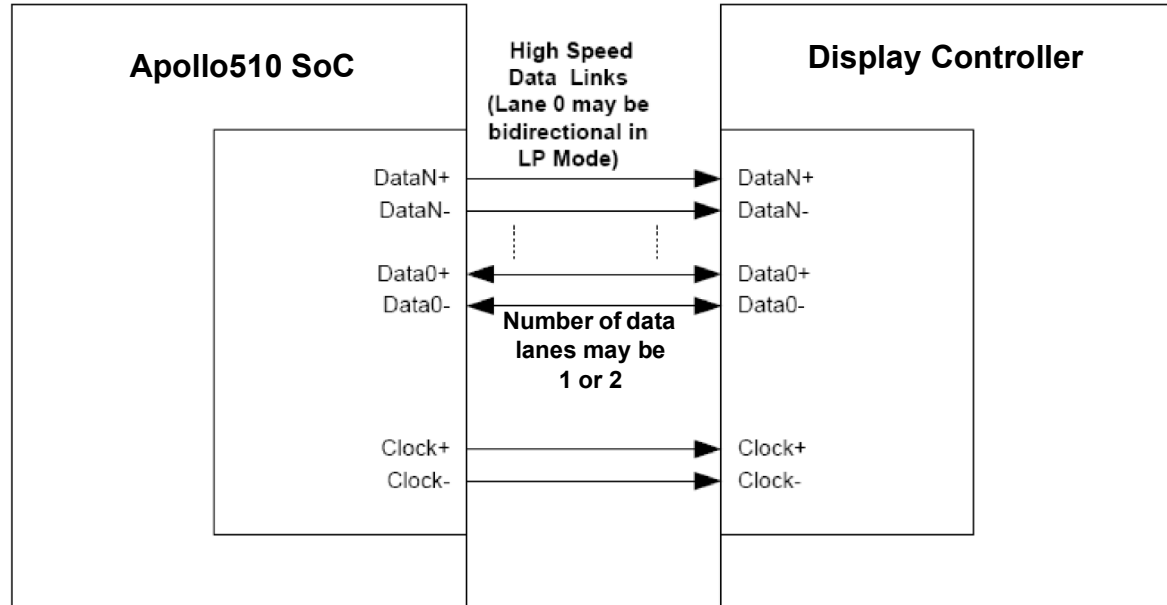
**Figure 43. Display Serial Interface Bus with DSI Devices**

The DSI module is configured to specify the interface and provide a connection between the MCU and a peripheral such as a display module. It is built on existing MIPI Alliance standards by adopting pixel formats, controlling pins and a command set specified in DPI-2, DBI-Type B and DCS standards.

NOTE

The DPI-2 and DBI-Type B interfaces are not offered on the WLCSP package.

The D-PHY's data lane signals are transferred point-to-point as differential signals using two signal lines and a clock lane. There are two signaling modes: high speed mode that operates at a rate of 768 Mbps per lane (1.536 Gbps total) and a low power mode (LP) that operates at a lower transfer rate of 10 Mbps. The mode is set to a low power mode and a stop state at start up / power up. Depending on the desired data transfer type, the lanes switch between high and low power modes. High speed data transfer is unidirectional and data transfer at low speed can be unidirectional or bidirectional.

## 22.3   Additional Information

Please refer to the Display Controller and DSI registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 23. Graphics Processing Unit (GPU)
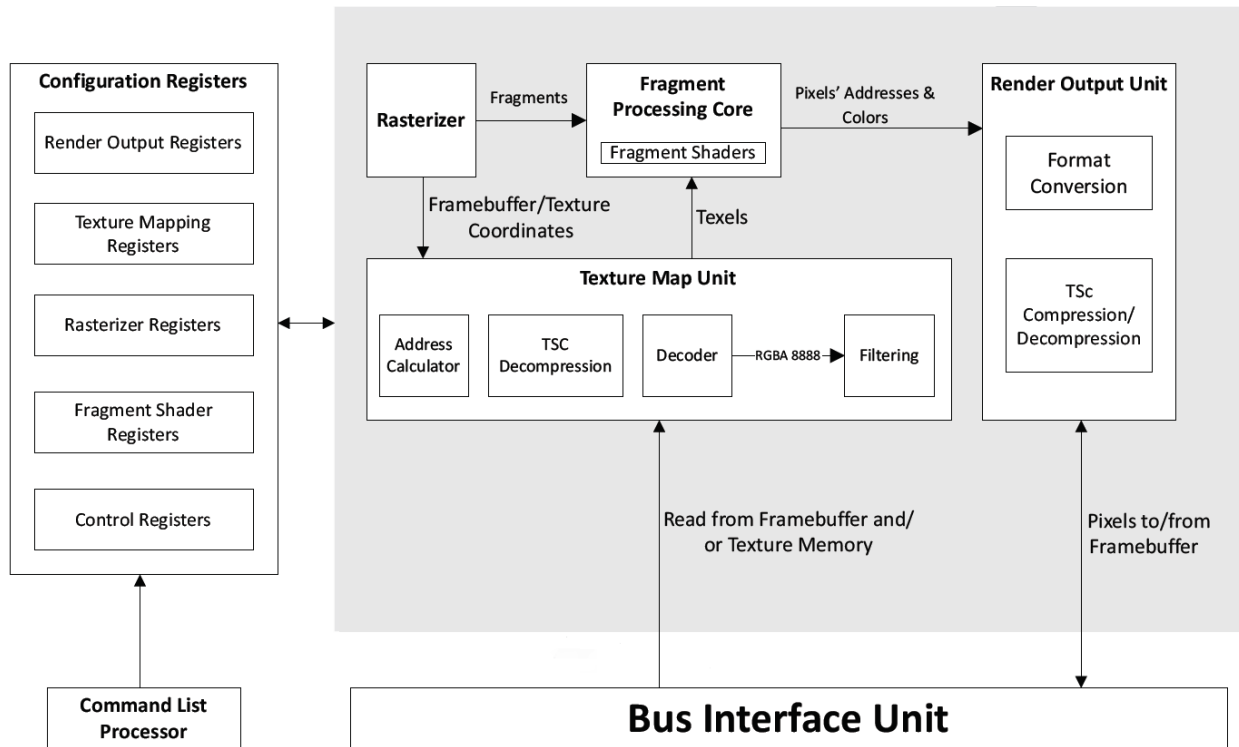


**Figure 44. GPU Block Diagram**

## 23.1  Features

The GPU Module is depicted in Figure 44. The Graphics Subsystem provides the following features:

- Fully programmable VLIW-based engine
- Fixed point functional units
- Command list based DMAs to minimize CPU overhead
- Compression schemes:
  - 4-bit
  - 6-bit (with/without Alpha)
  - 12-bit (with/without Alpha)
- 2.5D drawing
  - Pixel / Line drawing
  - Filled rectangles
  - Triangles (Gouraud Shaded)
  - Quadrilateral
- Blit support
  - Rotation
  - Mirroring
  - Stretch (independently on x and y axis)
  - Source and/or destination color keying
  - Format conversions
- Color formats
  - 32/24/16/8 bit with or without Alpha
  - Grayscale

- RGB
▪ Full Alpha blending
  - Programmable blending modes
  - Source/Destination color keying
▪ Anti-Aliasing support
▪ Dithering support
▪ Image transformation
  - 2.5D Perspective Correct Projections
  - Texture mapping
    - Point sampling
    - Bilinear filtering
▪ Vector Graphics (VG) supported by dedicated hardware accelerator
▪ Configurable burst length
▪ Radial/Conical fill
▪ TrueType Font (TTF)
▪ Dedicated hardware to support the following:
  - Vertex Transformation
  - Bezier Tessellation
  - Bezier Draw

## 23.2   Functional Overview

The GPU on the Apollo510 SoC brings high quality graphics for user interfaces in a very small power budget. The GPU supports entry level IoT platforms, wearable and embedded devices with low cost and ultra-low power requirements and provides fluid graphics experience for a wide range of applications. Developers are able to create compelling Graphical User Interfaces (GUIs) and software applications with ultra-long battery life at a significantly lower cost for power-memory-area constrained IoT devices.

## 23.3   Additional Information

Please refer to the GPU registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 24. PDM-to-PCM Converter Module (PDM)
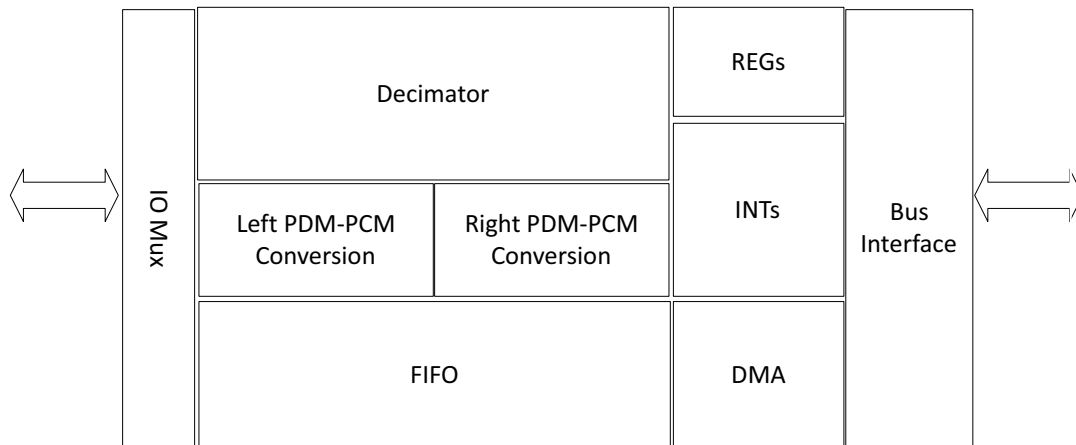


**Figure 45. PDM Block Diagram**

## 24.1 Features

The PDM-to-PCM Converter Module (PDM) includes features shown in Figure 45 and listed below.

- Support for up to 2 digital microphones
- Stereo/Mono Dual Mode PDM-to-PCM conversion
- 1-bit PDM (pulse-density modulated) input
- 24-bit PCM digital data output
- Programmable performance modes
- Supports ping-pong DMA jobs
- Supports digital microphone clock rate from 400 kHz to 3.072 MHz
- Supports granular decimation rates from 1x to 254x (64x typ)
- Supports PCM sampling rates of 8, 16, 24, 32 and 48 kHz
- PGA gain: -12 dB to +34.5 dB in 1.5 dB/step

---

ERRATUM NOTICE

In sampling configurations/frequencies with high Oversampling Ratio (> 128), the audio from the digital mic has distortion during speech. Most of the distortion comes from aliasing which is occurring during the PDM-to-PCM conversion. With a DMIC clock of 3.072 MHz and an OSR of 192, there is distortion from 1.6 kHz to 16 kHz. When OSR = 192, the MIC input signal bandwidth is greater than Fs/2 (= 16/2 = 8 kHz). If the mic input signal spectrum is wider than 8 kHz, a significant amount of aliasing occurs. Therefore, a maximum OSR of 128 is supported.

See "ERR021: PDM: High OSR causes aliasing and PDM mic distortion to occur during the PDM-to-PCM conversion" in the *Apollo510 SoC Errata List*.

---

## 24.2 Functional Overview

The Apollo510 SoC supports 1x stereo PDM controller. The PDM controller features low power stereo/ mono PDM-to-PCM converter with register programming.

Each controller operates in dual mode (stereo or mono).

In stereo mode, the controller converts 1-bit stereo pulse-density modulated (PDM) bit stream data from external digital microphones into 24-bit pulse-code modulated (PCM) data for base-band processing. In default operation, the PDM data sampled on the rising-edge of digital microphone clock is assumed to be left channel input, while data on the falling-edge is assumed to be right channel input. Optional channel swap is available through register setting. In mono mode, only the left channel PCM output put is valid while the right channel output is zero (no toggling).

The PDM-to-PCM converter typically supports a data sampling rate of 16 kHz for voice applications. It is capable of supporting output sampling rates (Fs) of 8, 16, 24, 32 and 48 kHz at different manager clock conditions. After input sampling, the PDM data bits are fed into digital filters for data conversion and gain amplification.

DMA jobs may be ping-pong processed to allow software to pre-process and/or post-process one DMA job while hardware is processing another DMA job.

## 24.3 Additional Information

Please refer to the PDM registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 25. Low Power Analog Audio Interface
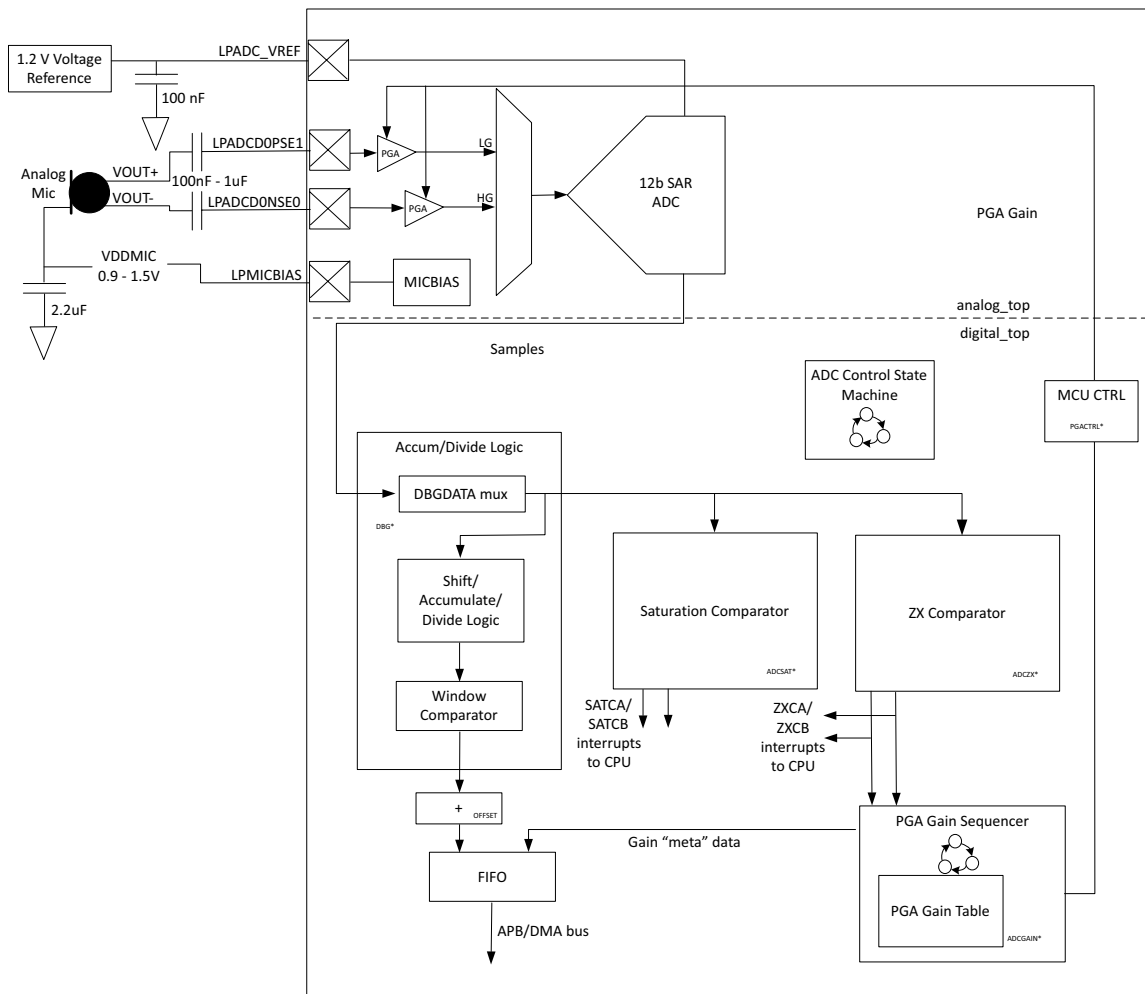


**Figure 46. Low Power Analog Audio Interface Block Diagram**

## 25.1  Features

The Low Power Analog Audio Interface is comprised of 2 channels of Programmable Gain Amplifiers (PGAs), 12-bit 2-channel Audio Analog-to-Digital Converter (AUDADC), and 1 low power microphone bias (MICBIAS) as shown in Figure 46.

Key features of the PGAs include:

- Programmable gain for AC-coupled audio inputs (20 Hz - 20 kHz) to drive AUDADC
- Audio inputs may be microphone or line inputs
  - Single Ended (SE)
  - Pseudo Differential (PD)
  - Fully Differential (FD)
- Full Scale Voltage
  - SE/PD: 0.5 Vrms
  - FD: 1 Vrms
- Gain steps supported: 0-24 dB in 0.5 dB increments

---

- Set input common-mode for active and sleep mode operation
- Implicit 2/3 attenuation to fit 1.2 V ADC full scale


Key features of the AUDADC include:

- Always-on operation
- Reconfigurable Successive Approximation Register (SAR) ADC
- 2 dedicated single-ended input channels (D0N/SE0, D0P/SE1) from two PGA sources[1]
- Input Range: 0 V to 1.2 V
- Configurable automatic low power control between scans
- Configurable for 12-bit, 10-bit and 8-bit ADC Precision Modes
- Sampling rate up to 2.0 MS/s (12-bit Mode) & 2.8 MS/s (8-bit Mode)
- Configurable sampling time
- Uses 1.2 V external reference with internal buffer
- Single shot, repeating single shot, scan, and repeating scan modes
- Variable sample tracking time, configurable on per-slot basis
- User-selectable clock source for variable sampling rates
- Automatically accumulate and scale module for hardware averaging of samples
- 16-entry FIFO and DMA capability for storing measurement results and maximizing SoC sleep time
- Multiple Interrupt Support:
  - FIFO full
  - FIFO 75% full
  - Scan Complete
  - Conversion Complete
  - Window Incursion
  - Window Excursion
  - Various DMA-related notifications
- Window comparator for monitoring voltage excursions into or out of user-selectable thresholds
  - Unsigned mode support ONLY
- Supports signed data mode by way of AUDADC_ADCCFG_DATAFMT
- Settable sampling/tracking time per-slot
- ADC-internal trigger timer providing low-jitter periodic repeated triggers
- Additional delays configurable via AUDADC registers

---

1. LPADC inputs D1N/SE2 and D1P/SE3 are not offered on either BGA or the WLCSP package.

Key features of MICBIAS include:

▪ MICBIAS provides user-programmable regulated (0.9 V to 1.5 V) supply to analog MEMS microphones
▪ Performance Summary:
  - 200 μA max load current with 2.2 μF capacitor
  - 560 nA quiescent current
  - Typical PSR (from VDDAUD)
    - 34 dB @ 1 kHz
    - 15 dB @ 20 kHz
    - Startup < 1 ms

## 25.2  Functional Overview

The always-on Audio ADC on the Apollo510 SoC is connected to 2 Programmable Gain Amplifier (PGA) channels. These gains are programmable gains and may be controlled by the ADC. Generally, the low-gain and high-gain path will maintain a fixed dB gain delta between them. The low gain parameter is adjusted depending on the amplitude of the incoming audio signal and this, in turn, also adjusts the high gain parameter in the same fashion.

When the Audio ADC takes samples, they can be analyzed for a specific number of saturation events wherein the amplitude of the signal exceeds a programmable threshold. This can notify the CPU via interrupts such that it may take action to reduce the PGA gain. Saturation may be monitored on either low gain or high gain path, but not a mix of the two.

Additionally, these samples can be monitored for zero-crossings (ZX) such that when the signal crosses this configurable sample region, gain parameters may be automatically updated. This is done in order to reduce the likelihood of pops or clicks as gains are adjusted. There is also an "immediate" gain update mode in which the gains will be updated immediately following the upcoming scan (a full set of conversions). As with saturation detection, ZX may be monitored on either both low gain or both high gain paths, but not a mix of the two. The ZX monitor may also provide per-channel interrupts. This can be useful to let the CPU know when a ZX gain update is about to occur.

To help avoid the potential of a varying gain making it difficult to process the sampled audio, the low-gain parameter which was used for the specific sample is encoded and packed along with high gain/low gain pairs of up to 12-bit samples in each 32-bit word. Initially, samples may be taken using only the low-gain path, reducing power consumption ("LP" mode). Then, once a keyword or voice is detected, the audio ADC may be switched into "MED" mode, which enables sampling on the high-gain paths as well.

Otherwise, the AUDADC is an instance of the general purpose ADC.

---

NOTE

The AUDADC can be configured for High Performance (Telco) mode or Low Power (Always On) mode. In High Performance mode, the PLL should be configured to provide a 48 MHz source clock by setting the MCUCTRL_PLL-MUXCTL_AUDADCPLLCLKSEL register to PLL, with AUDADC_CFG_CLKSEL set to OFF.

In Low Power mode, the low power HFRC_48MHz clock source should be used by setting the AUDADC_CFG_CLKSEL to HFRC_48MHz.

The recommended sampling rate for both modes is 16 kHz for best performance.

---

> **ERRATUM NOTICE**
>
> After a SW POR, if the AUDADC is attempted to be re-enabled using a prior configuration of using the XTALHS clock option without first enabling the high-speed crystal again after the reset, the MCU hangs.
>
> See "ERR001: AUDADC: MCU hangs when attempting to configure the AUDADC with disabled clock" in the *Apollo510 SoC Errata List*.

## 25.3  Additional Information

Please refer to the AUDADC registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 26. Inter-IC Sound (I$^2$S)
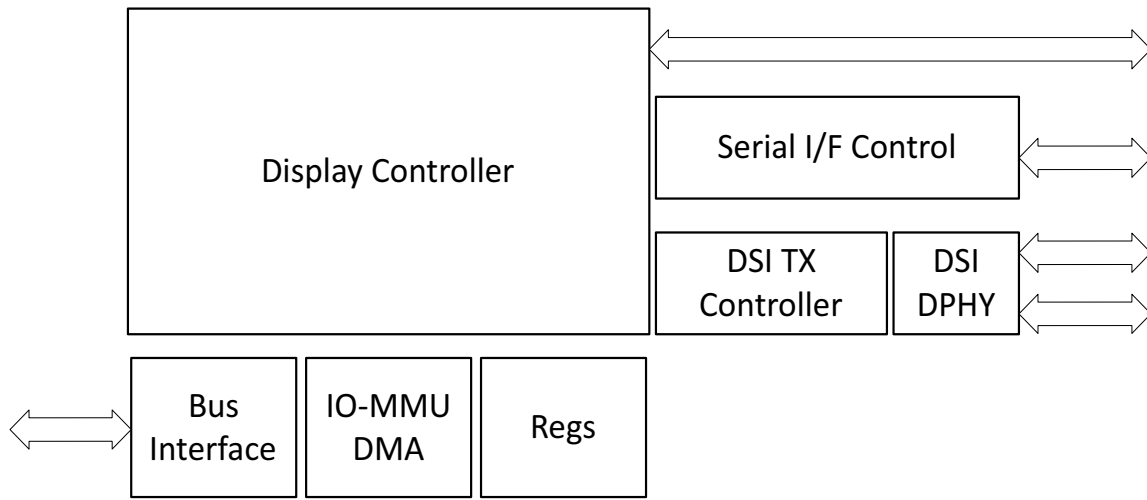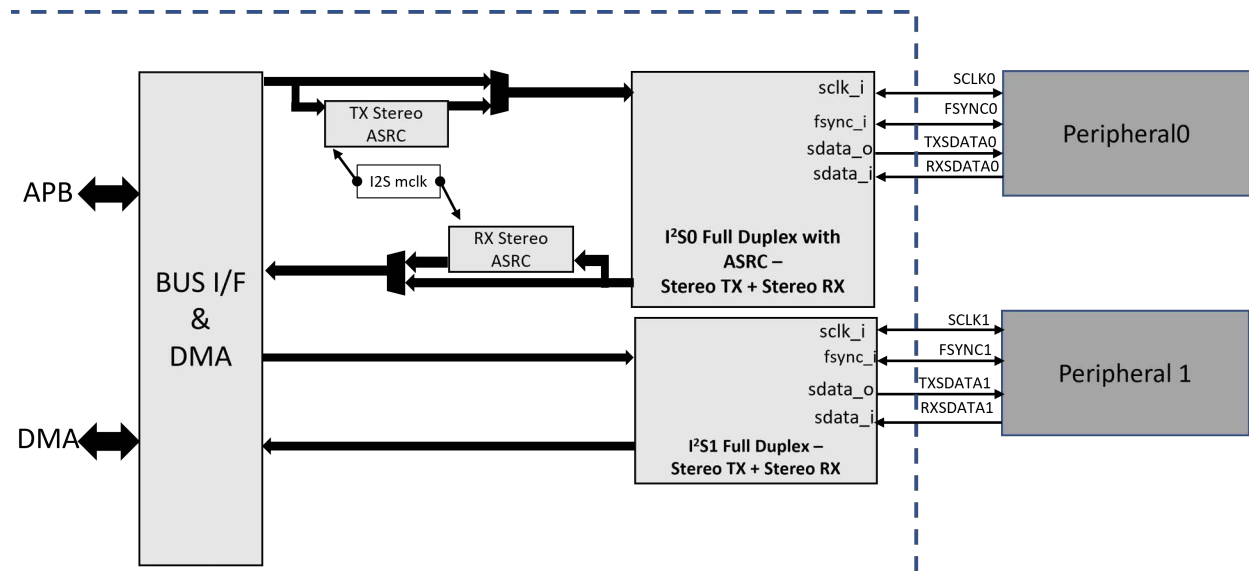


**Figure 47. I$^2$S Block Diagram**

## 26.1 Features

The Inter-IC Sound (I$^2$S) module includes features shown in Figure 47 and listed below.

- Inter-IC audio streaming interface
- Modes
  - I$^2$S Philips mode
  - I$^2$S right-justified and left-justified serial audio format modes
  - Time Division Multiplexing (TDM) mode
- Audio sample sizes of 8, 16, 24 and 32 bit
  - TDM has tremendous flexibility for framing and bit width
- 2x instances of full-duplex I$^2$S (stereo TX + stereo RX) using shared CLK & FS
- Manager and subordinate
- Supports ping-pong DMA jobs
- Optional Asynchronous Sample Rate Conversion (ASRC) on subordinate I$^2$S channels (I2S0 only)
  - Sample size: 24 bits (for internal processing, but accepts 8, 16, 24 or 32 bits)
  - Supported sampling rates: 8, 16, 24, 32, 44.1 and 48 kHz
  - Maximum down conversion of 1.95.1
  - Maximum up conversion of 1:7
  - Fixed FSYNC:SCLK ratio of 64:1 required
  - Lower than -130 dB THD+N for common conversion ratios (when using 24-bit samples)
- 1 to 8 Channel TDM interface

## 26.2 Functional Overview

The Apollo510 SoC supports two I$^2$S controllers. I2S0 supports manager or subordinate and I2S1 supports full duplex manager mode. Various modes and sample rates are supported to provide flexible audio data processing. DMA is supported to enable efficient transfer of data to/from SRAM. I2S0 also supports a configurable Asynchronous Sample Rate Converters (ASRC) capable of supporting stereo transmit and/or receive when configured as a subordinate device. DMA jobs can be ping-pong processed to allow software to pre-process and/or post-process one DMA job while the hardware is processing another DMA job.

## 26.3 Additional Information

Please refer to the I$^2$S registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 27. Voltage Regulator Module
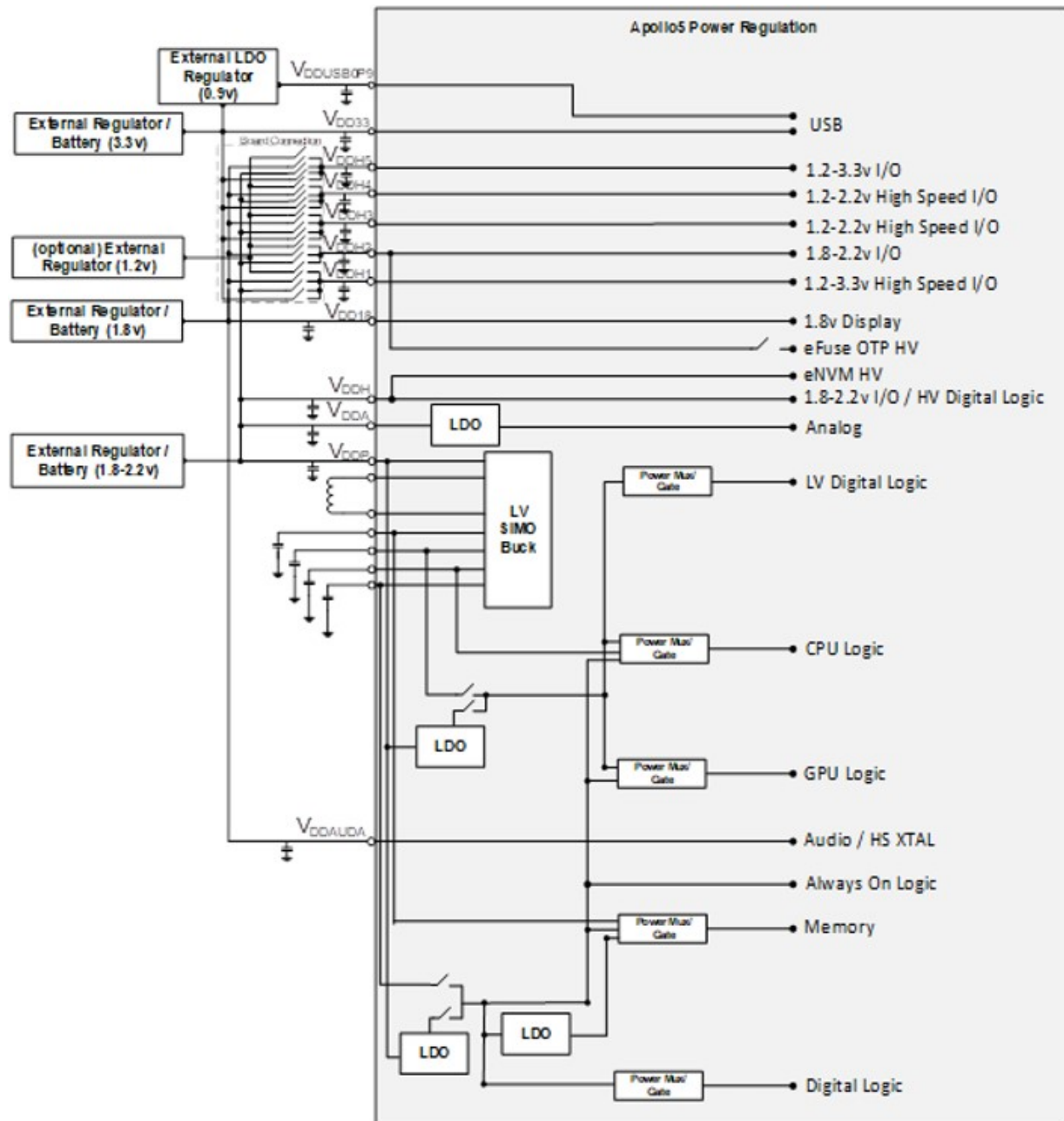


**Figure 48. Block Diagram for Voltage Supplies and Regulation on Apollo510 SoC**

## 27.1 Features

The voltage supplies and regulation subsystem includes features shown in Figure 48 and listed below.

- Down-converts and efficiently regulates the VDD supply voltage
- Optimized for low power environments
- Includes single-inductor/multiple-output SIMO Buck with ultra-low quiescent current

---

- SIMO Buck sources primary supplies for the core and memory domains
- Prevents drop out of regulated voltages from the SIMO Buck
- Enters efficient ultra-low power mode automatically based on active system load current
- Low-dropout (LDO) linear regulator available for very low-power modes
- LDO offers a lower cost system solution

## 27.2  Functional Overview

The Voltage Regulator Module on the Apollo510 SoC down-converts and regulates the supply voltage, VDD, with extremely high efficiency. The SIMO Buck, which is a single-inductor/multiple-output design enabled via software, sources the primary supplies for the core and memory domains. It enables down-conversion from the power supply input (e.g., a battery or external regulator) at more than 80% efficiency. With ultra-low quiescent current, the SIMO Buck Converter is optimized for low power environments.

Upon enabling the SIMO Buck, it will be powered up and stabilized through hardware control. The SIMO Buck has an efficient ultra-low power mode that is entered automatically via hardware control based on active load current of the system.

For cost/area constrained designs, the SIMO Buck can be disabled and a low-dropout (LDO) linear regulator can be used in very low-power modes. In this configuration, the SIMO Buck will remain powered down. There is also a zero-length detect circuit to ensure the regulated voltages from the SIMO Buck do not drop out.

The LDO regulator can also be utilized to provide a lower cost system solution by eliminating the need for the external inductor required in buck mode. The VDDC and VDDF capacitors are still required for the internal LDO.

The SIMO Buck Converter and LDO of the Voltage Regulator Module are tightly coupled to the various low power modes in the Apollo510 SoC. When the device enters deep sleep mode, the Buck Converter switches into a low power mode to provide very high efficiency at low quiescent current.

---

NOTE

The falling slew rate of a supply cannot exceed 2 kV/s. Doing so will cause indeterminate device behavior. In addition, I/O on rails should not exceed VDDHn+0.3V before the rail in question exceeds VDDHn_min, as it may cause indeterminate behavior.

---

ERRATUM NOTICE

The SIMO Buck cannot be enabled automatically via INFO0.

See "ERR015: Memory: SIMO Buck cannot be enabled via INFO0 setting" in the *Apollo510 SoC Errata List*.

---

<div style="border:1px solid black; padding:10px;">

ERRATUM NOTICE

An automatic SIMO Buck transition mode, which is enabled by setting both PWRCTRL_TONCNTRCTRL register fields LPMODESWOVR and ENABLELPOVR to 0x1, causes the SIMO Buck to transition to active mode if there is an increase in load current, but otherwise stays in Low Power (LP) Mode. Once the SIMO Buck transitions into Active Mode in deep sleep mode, it periodically retries to go into LP Mode. The SIMO Buck can transition from LP to Active if the temperature change and load change are gradual. However, it cannot handle the transition if there is a rapid change in temperature or load current. Because of this limitation, production trims do not make use of SIMO Buck automatic mode transitioning, and it is recommended that these PWRCTRL bits remain cleared.

See "ERR025: PWRCTRL: SIMO Buck fails to transition from Active Mode to LP Mode" in the *Apollo510 SoC Errata List*.

</div>

## 27.3   Additional Information

Please refer to the voltage regulation and control registers in the MCUCTRL and PWRCTRL registers of the Apollo510 SoC register set. The register set is delivered as part of the AmbiqSuite SDK.

# 28. Apollo510 SoC Package Pins

## 28.1 Pin Configuration
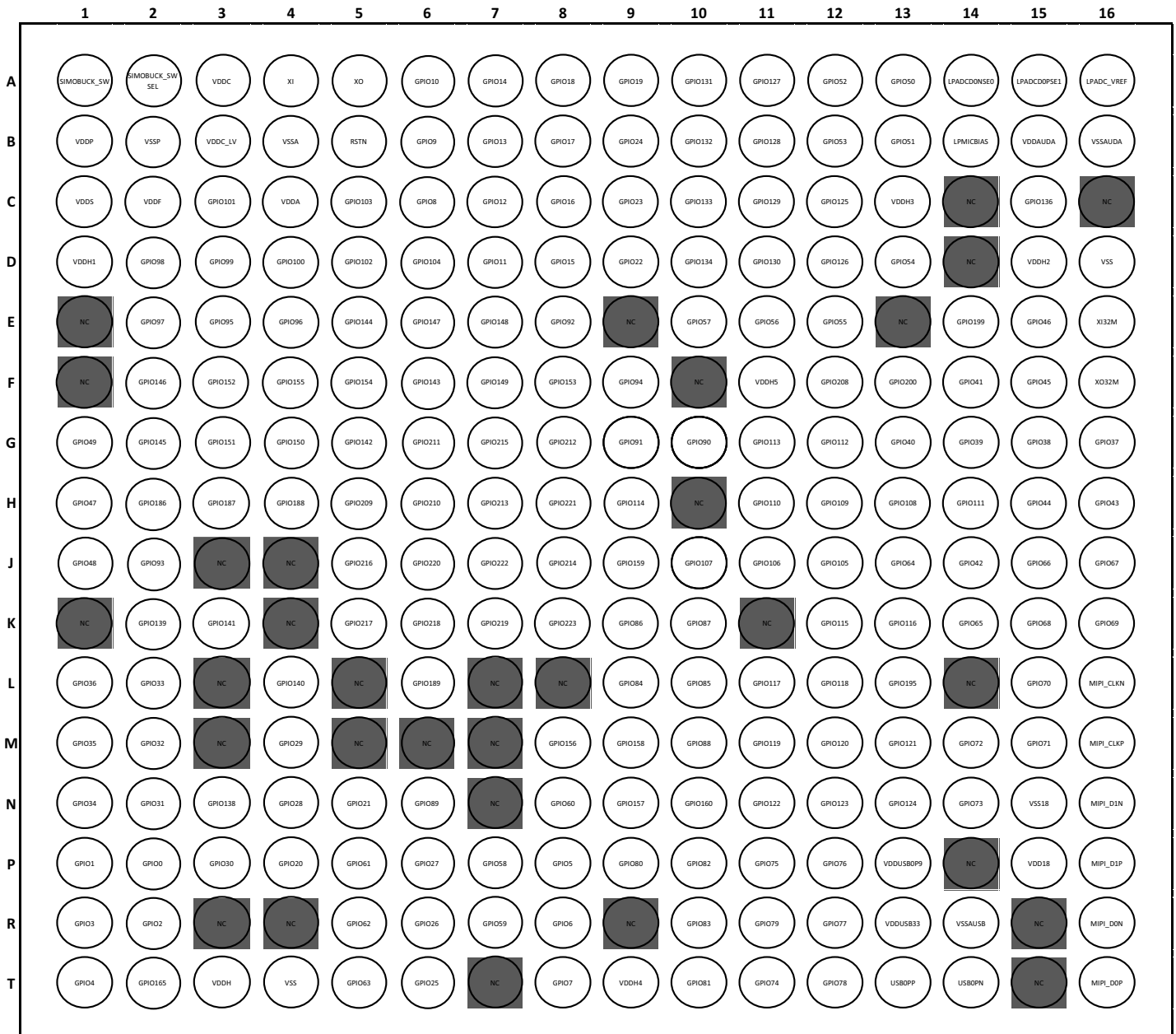


**Figure 49. Apollo510 SoC BGA Pin Configuration Diagram - Top View**

NOTE

All balls shaded gray are "no connects" and are not populated on the package.
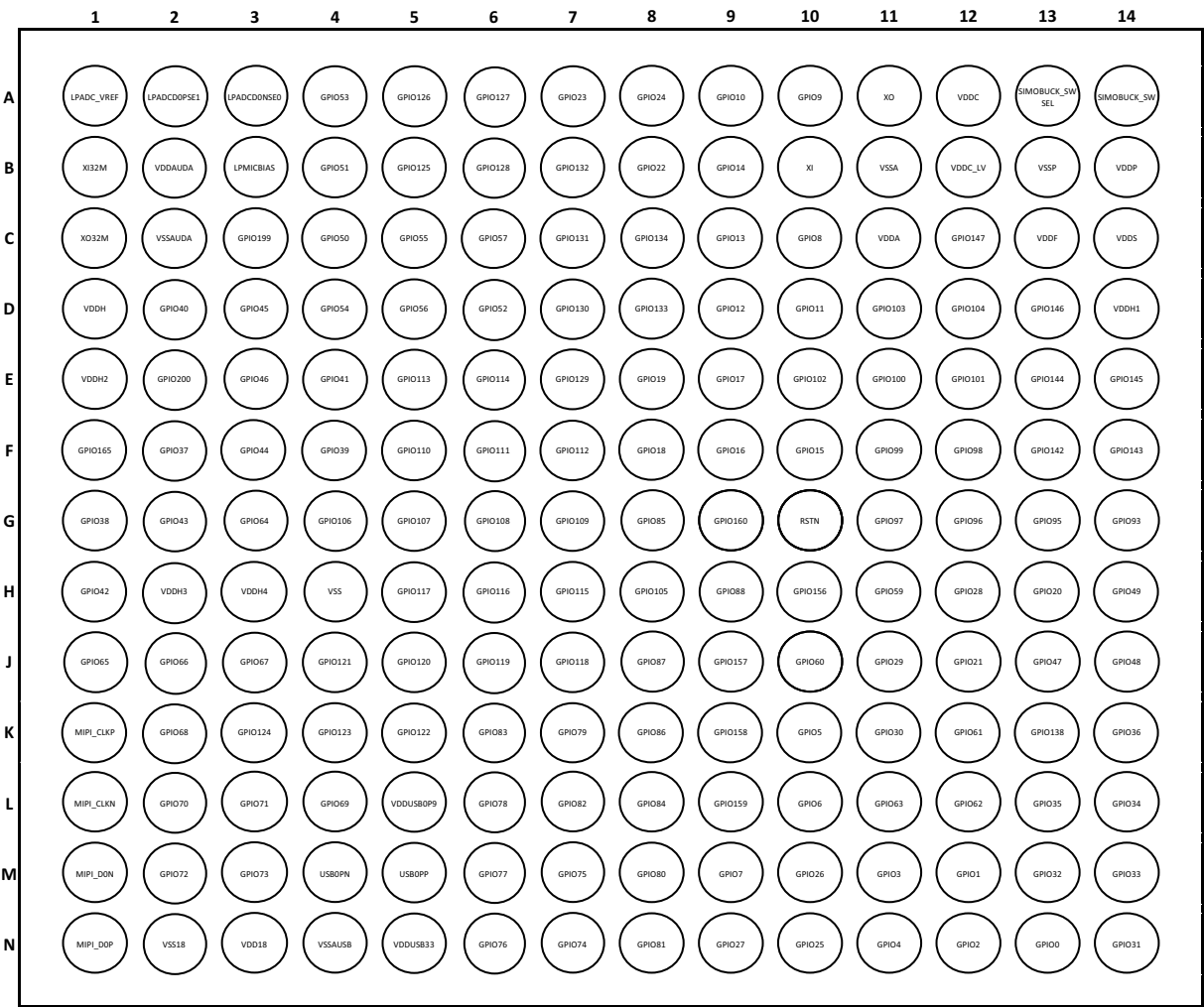
**Figure 50. Apollo510 SoC WLCSP Pin Configuration Diagram - Top View**

## 28.2 Pin Connections

The following table lists the external pins of the Apollo510 SoC and their available functions.

> **NOTE**
>
> For the GPIO pads in the Pin List and Function Table, only those GPIO offered on the BGA package and WLCSP package are listed in the table.

> **NOTE**
>
> Ball D16 is a ground on the BGA package and must be grounded to ensure proper operation of the Apollo510 SoC.

### Table 24: Apollo510 SoC Pin List and Function Table

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| P15 | N3 | - | - | VDD18 | VDD supply for MIPI PHY | Power |
| C4 | C11 | - | - | VDDA | Analog voltage supply | Power |
| B15 | B2 | - | - | VDDAUDA | Analog Audio Voltage supply | Power |
| A3 | A12 | - | - | VDDC | Core Buck converter VOUT | Power |
| B3 | B12 | - | - | VDDC_LV | Core_LV Buck converter VOUT | Power |
| C2 | C13 | - | - | VDDF | Mem Buck converter  VOUT | Power |
| T3 | D1 | - | - | VDDH | High voltage domain power supply | Power |
| D1 | D14 | - | - | VDDH1 | High voltage domain1 power supply | Power |
| D15 | E1 | - | - | VDDH2 | High voltage domain2 power supply | Power |
| C13 | H2 | - | - | VDDH3 | High voltage domain3 power supply | Power |
| T9 | H3 | - | - | VDDH4 | High voltage domain4 power supply | Power |
| F11 | - | - | - | VDDH5 | High voltage domain5 power supply | Power |
| B1 | B14 | - | - | VDDP | VDD supply to I/O pads (Core) | Power |
| C1 | C14 | - | - | VDDS | SRAM high voltage supply | Power |
| P13 | L5 | - | - | VDDUSB0P9 | USB 0.9v analog voltage supply | Power |
| R13 | N5 | - | - | VDDUSB33 | USB 3.3v voltage supply | Power |
| D16 | H4 | - | - | VSS | Digital Ground for VDDF and PADS (Noisy) - (Previously called GNDD) | Ground |
| N15 | N2 | - | - | VSS18 | MIPI PHY Analog Ground | Ground |
| B4 | B11 | - | - | VSSA | Analog Ground - Same as GNDA | Ground |
| B16 | C2 | - | - | VSSAUDA | Analog Audio Ground | Ground |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| R14 | N4 | - | - | VSSAUSB | USB PHY Analog Ground | Ground |
| B2 | B13 | - | - | VSSP | Ground Connection for buck regs - Same as GNDP | Ground |
| A16 | A1 | - | - | LPADC_VREF | LP ADC Reference Decap | Analog |
| A14 | A3 | - | - | LPADCD0NSE0 | LP Analog to Digital Converter SE0/DiffN IN0 | Input |
| A15 | A2 | - | - | LPADCD0PSE1 | LP Analog to Digital Converter SE1/DiffP IN0 | Input |
| B14 | B3 | - | - | LPMICBIAS | LP Microphone Bias | Output |
| L16 | L1 | - | - | MIPI_CLKN | MIPI DPHY Clock Lane N | I/O |
| M16 | K1 | - | - | MIPI_CLKP | MIPI DPHY Clock Lane P | I/O |
| R16 | M1 | - | - | MIPI_D0N | MIPI DPHY Data Lane 0N | I/O |
| T16 | N1 | - | - | MIPI_D0P | MIPI DPHY Data Lane 0P | I/O |
| N16 | - | - | - | MIPI_D1N | MIPI DPHY Data Lane 1N | I/O |
| P16 | - | - | - | MIPI_D1P | MIPI DPHY Data Lane 1P | I/O |
| B5 | G10 | - | - | RSTN | External reset input (aka nRST) | Input |
| A1 | A14 | - | - | SIMOBUCK_SW | SIMO Buck converter inductor switch output | Power |
| A2 | A13 | - | - | SIMOBUCK_SWSEL | SIMO Buck converter inductor switch input | Power |
| T14 | M4 | - | - | USB0PN | The differential input/output signals of the PHY that support multiple modes. Depending on mode of operation they are either signaling 3.3V or 800mV differential. | Power |
| T13 | M5 | - | - | USB0PP | The differential input/output signals of the PHY that support multiple modes. Depending on mode of operation they are either signaling 3.3V or 800mV differential. | Power |
| A4 | B10 | - | - | XI | 32.768kHz crystal input | XT |
| E16 | B1 | - | - | XI32M | 32MHz crystal input | XT24M |
| A5 | A11 | - | - | XO | 32.768kHz crystal output | XT |
| F16 | C1 | - | - | XO32M | 32MHz crystal output | XT24M |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| P2 | N13 | 0 | 0 | SWTRACECLK | Serial Wire Debug Trace Clock | Output |
| | | | 1 | SLFDSCK | SPI Subordinate Full Duplex clock | Input |
| | | | 2 | - | - | - |
| | | | 3 | GPIO0 | General purpose I/O | I/O |
| | | | 4 | UART0TX | UART0 transmit output | Output |
| | | | 5 | UART1TX | UART1 transmit output | Output |
| | | | 6 | CT0 | Timer/counter 0 | Output |
| | | | 7 | NCE0 | IOMSTR N Chip Select 0 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| P1 | M12 | 1 | 0 | SWTRACE0 | Serial Wire Debug Trace Output 0 | Output |
| | | | 1 | SLFDMOSI | SPI Subordinate Full Duplex input data | Input |
| | | | 2 | SLFDWIR3 | SPI Subordinate Full Duplex I/O pin for 3-wire | Bidirectional 3-state |
| | | | 3 | GPIO1 | General purpose I/O | I/O |
| | | | 4 | UART2TX | UART2 transmit output | Output |
| | | | 5 | UART3TX | UART3 transmit output | Output |
| | | | 6 | CT1 | Timer/counter 1 | Output |
| | | | 7 | NCE1 | IOMSTR N Chip Select 1 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | SLFD1WIR3 | SPI Subordinate1 Full Duplex I/O pin for 3-wire | Output |
| | | | 12 | - | - | - |
| R2 | N12 | 2 | 0 | SWTRACE1 | Serial Wire Debug Trace Output 1 | Output |
| | | | 1 | SLFDMISO | SPI Subordinate Full Duplex output data | Output |
| | | | 2 | TRIG1 | ADC trigger input | Input |
| | | | 3 | GPIO2 | General purpose I/O | I/O |
| | | | 4 | UART0RX | UART0 receive input | Input |
| | | | 5 | UART1RX | UART1 receive input | Input |
| | | | 6 | CT2 | Timer/counter 2 | Output |
| | | | 7 | NCE2 | IOMSTR N Chip Select 2 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | SLFD1MISO | SPI Subordinate1 Full Duplex output data | Output |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| R1 | M11 | 3 | 0 | SWTRACE2 | Serial Wire Debug Trace Output 2 | Output |
| | | | 1 | SLFDnCE | SPI Subordinate Full Duplex chip enable | Input |
| | | | 2 | SWO | Serial Wire Debug | Output |
| | | | 3 | GPIO3 | General purpose I/O | I/O |
| | | | 4 | UART2RX | UART2 receive input | Input |
| | | | 5 | UART3RX | UART3 receive input | Input |
| | | | 6 | CT3 | Timer/counter 3 | Output |
| | | | 7 | NCE3 | IOMSTR N Chip Select 3 | Output |
| | | | 9 | - | - | - |
| | | | 10 | I2S1_SDIN | I2S1 Data input | Input |
| | | | 12 | - | - | - |
| T1 | N11 | 4 | 0 | SWTRACE3 | Serial Wire Debug Trace Output 3 | Output |
| | | | 1 | SLFDINT | Configurable Subordinate Interrupt | Output |
| | | | 2 | 32KHzXT | 32kHZ from analog | Output |
| | | | 3 | GPIO4 | General purpose I/O | I/O |
| | | | 4 | UART0RTS | UART0 Request to Send (RTS) | Output |
| | | | 5 | UART1RTS | UART1 Request to Send (RTS) | Output |
| | | | 6 | CT4 | Timer/counter 4 | Output |
| | | | 7 | NCE4 | IOMSTR N Chip Select 4 | Output |
| | | | 9 | I2S0_SDIN | I2S0 Data input | Input |
| | | | 10 | SLFD1INT | Configurable Subordinate Interrupt1 | Output |
| | | | 12 | - | - | - |
| P8 | K10 | 5 | 0 | M0SCL | I2C Manager 0 clock | Open Drain Output |
| | | | 1 | M0SCK | SPI Manager 0 clock | Output |
| | | | 2 | I2S0_CLK | I2S0 Bit clock | Bidirectional |
| | | | 3 | GPIO5 | General purpose I/O | I/O |
| | | | 4 | UART2RTS | UART2 Request to Send (RTS) | Output |
| | | | 5 | UART3RTS | UART3 Request to Send (RTS) | Output |
| | | | 6 | CT5 | Timer/counter 5 | Output |
| | | | 7 | NCE5 | IOMSTR N Chip Select 5 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| R8 | L10 | 6 | 0 | M0SDAWIR3 | I2C Manager 0 I/O data  (I2C) 3 Wire data (SPI) | Bidirectional Open Drain |
| | | | 1 | M0MOSI | SPI Manager 0 output data | Output |
| | | | 2 | I2S0_DATA | I2S0 Data | Bidirectional |
| | | | 3 | GPIO6 | General purpose I/O | I/O |
| | | | 4 | UART0CTS | UART0 Clear to Send (CTS) | Input |
| | | | 5 | UART1CTS | UART1 Clear to Send (CTS) input | Input |
| | | | 6 | CT6 | Timer/counter 6 | Output |
| | | | 7 | NCE6 | IOMSTR N Chip Select 6 | Output |
| | | | 9 | I2S0_SDOUT | I2S0 Data output | Output |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| T8 | M9 | 7 | 0 | M0MISO | SPI Manager 0 input data | Input |
| | | | 1 | TRIG0 | ADC trigger input | Input |
| | | | 2 | I2S0_WS | I2S0 L/R clock | Bidirectional |
| | | | 3 | GPIO7 | General purpose I/O | I/O |
| | | | 4 | UART2CTS | UART2 Clear to Send (CTS) input | Input |
| | | | 5 | UART3CTS | UART3 Clear to Send (CTS) input | Input |
| | | | 6 | CT7 | Timer/counter 7 | Output |
| | | | 7 | NCE7 | IOMSTR N Chip Select 7 | Output |
| | | | 9 | MNCE2_0 | MSPI Manager 2 nCE 0 Signal | - |
| | | | 10 | MNCE3_1 | MSPI Manager 3 nCE 1 Signal | - |
| | | | 12 | - | - | - |
| C6 | C10 | 8 | 0 | CMPRF1 | Comparator reference 1 | Input |
| | | | 1 | TRIG1 | ADC trigger input | Input |
| | | | 2 | - | - | - |
| | | | 3 | GPIO8 | General purpose I/O | I/O |
| | | | 4 | M1SCL | I2C Manager 1 clock | Open Drain Output |
| | | | 5 | M1SCK | SPI Manager 1 clock | Output |
| | | | 6 | CT8 | Timer/counter 8 | Output |
| | | | 7 | NCE8 | IOMSTR N Chip Select 8 | Output |
| | | | 9 | - | - | - |
| | | | 10 | I2S1_CLK | I2S1 Bit clock | Bidirectional |
| | | | 12 | - | - | - |

| |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| B6 | A10 | 9 | 0 | CMPRF0 | Comparator reference 0 | Input |
| | | | 1 | TRIG2 | ADC trigger input | Input |
| | | | 2 | - | - | - |
| | | | 3 | GPIO9 | General purpose I/O | I/O |
| | | | 4 | M1SDAWIR3 | I2C Manager 1 I/O data  (I2C) 3 Wire data (SPI) | Bidirectional Open Drain |
| | | | 5 | M1MOSI | SPI Manager 1 output data | Output |
| | | | 6 | CT9 | Timer/counter 9 | Output |
| | | | 7 | NCE9 | IOMSTR N Chip Select 9 | Output |
| | | | 9 | I2S1_DATA | I2S1 Data | Bidirectional |
| | | | 10 | I2S1_SDOUT | I2S1 Data output | Output |
| | | | 12 | - | - | - |
| A6 | A9 | 10 | 0 | CMPIN0 | Voltage comparator input 0 | Input |
| | | | 1 | TRIG3 | ADC trigger input | Input |
| | | | 2 | MNCE0_0 | MSPI Manager 0 nCE 0 Signal | - |
| | | | 3 | GPIO10 | General purpose I/O | I/O |
| | | | 4 | M1MISO | SPI Manager 1 input data | Input |
| | | | 5 | MNCE2_0 | MSPI Manager 2 nCE 0 Signal | - |
| | | | 6 | CT10 | Timer/counter 10 | Output |
| | | | 7 | NCE10 | IOMSTR N Chip Select 10 | Output |
| | | | 9 | DISP_TE | Display TE input | Input |
| | | | 10 | I2S1_WS | I2S1 L/R clock | Bidirectional |
| | | | 12 | MNCE2_1 | MSPI Manager 2 nCE 1 Signal | - |
| D7 | D10 | 11 | 0 | CMPIN1 | Voltage comparator input 1 | Input |
| | | | 1 | TRIG0 | ADC trigger input | Input |
| | | | 2 | I2S0_CLK | I2S0 Bit clock | Bidirectional |
| | | | 3 | GPIO11 | General purpose I/O | I/O |
| | | | 4 | UART2RX | UART2 receive input | Input |
| | | | 5 | UART3RX | UART3 receive input | Input |
| | | | 6 | CT11 | Timer/counter 11 | Output |
| | | | 7 | NCE11 | IOMSTR N Chip Select 11 | Output |
| | | | 9 | SLSCL | I2C Subordinate Half Duplex clock | Input |
| | | | 10 | SLSCK | SPI Subordinate Half Duplex clock | Input |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| C7 | D9 | 12 | 0 | ADCSE7 | Analog to Digital Converter SE IN7 | Input |
| | | | 1 | TRIG1 | ADC trigger input | Input |
| | | | 2 | I2S0_DATA | I2S0 Data | Bidirectional |
| | | | 3 | GPIO12 | General purpose I/O | I/O |
| | | | 4 | UART0TX | UART0 transmit output | Output |
| | | | 5 | UART1TX | UART1 transmit output | Output |
| | | | 6 | CT12 | Timer/counter 12 | Output |
| | | | 7 | NCE12 | IOMSTR N Chip Select 12 | Output |
| | | | 9 | CMPRF2 | Comparator reference 2 | Input |
| | | | 10 | I2S0_SDOUT | I2S0 Data output | Output |
| | | | 12 | - | - | - |
| B7 | C9 | 13 | 0 | ADCSE6 | Analog to Digital Converter SE IN6 | Input |
| | | | 1 | TRIG2 | ADC trigger input | Input |
| | | | 2 | I2S0_WS | I2S0 L/R clock | Bidirectional |
| | | | 3 | GPIO13 | General purpose I/O | I/O |
| | | | 4 | UART2TX | UART2 transmit output | Output |
| | | | 5 | UART3TX | UART3 transmit output | Output |
| | | | 6 | CT13 | Timer/counter 13 | Output |
| | | | 7 | NCE13 | IOMSTR N Chip Select 13 | Output |
| | | | 9 | SLnCE | SPI Subordinate Half Duplex chip enable | Input |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| A7 | B9 | 14 | 0 | ADCSE5 | Analog to Digital Converter SE IN5 | Input |
| | | | 1 | TRIG3 | ADC trigger input | Input |
| | | | 2 | - | - | - |
| | | | 3 | GPIO14 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | UART1RX | UART1 receive input | Input |
| | | | 6 | CT14 | Timer/counter 14 | Output |
| | | | 7 | NCE14 | IOMSTR N Chip Select 14 | Output |
| | | | 9 | - | - | - |
| | | | 10 | I2S0_SDIN | I2S0 Data input | Input |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| D8 | F10 | 15 | 0 | ADCSE4 | Analog to Digital Converter SE IN4 | Input |
| | | | 1 | TRIG0 | ADC trigger input | Input |
| | | | 2 | - | - | - |
| | | | 3 | GPIO15 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | UART3RX | UART3 receive input | Input |
| | | | 6 | CT15 | Timer/counter 15 | Output |
| | | | 7 | NCE15 | IOMSTR N Chip Select 15 | Output |
| | | | 9 | - | - | - |
| | | | 10 | REFCLK_EXT | External Reference Clock | Input |
| | | | 12 | - | - | - |
| C8 | F9 | 16 | 0 | ADCSE3 | Analog to Digital Converter SE IN3 | Input |
| | | | 1 | TRIG1 | ADC trigger input | Input |
| | | | 2 | I2S1_CLK | I2S1 Bit clock | Bidirectional |
| | | | 3 | GPIO16 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | UART1RTS | UART1 Request to Send (RTS) | Output |
| | | | 6 | CT16 | Timer/counter 16 | Output |
| | | | 7 | NCE16 | IOMSTR N Chip Select 16 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| B8 | E9 | 17 | 0 | ADCSE2 | Analog to Digital Converter SE IN2 | Input |
| | | | 1 | TRIG2 | ADC trigger input | Input |
| | | | 2 | I2S1_DATA | I2S1 Data | Bidirectional |
| | | | 3 | GPIO17 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | UART3RTS | UART3 Request to Send (RTS) | Output |
| | | | 6 | CT17 | Timer/counter 17 | Output |
| | | | 7 | NCE17 | IOMSTR N Chip Select 17 | Output |
| | | | 9 | I2S1_SDOUT | I2S1 Data output | Output |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| A8 | F8 | 18 | 0 | ADCSE1 | Analog to Digital Converter SE IN1 | Input |
| | | | 1 | - | - | - |
| | | | 2 | I2S1_WS | I2S1 L/R clock | Bidirectional |
| | | | 3 | GPIO18 | General purpose I/O | I/O |
| | | | 4 | UART0CTS | UART0 Clear to Send (CTS) | Input |
| | | | 5 | UART1CTS | UART1 Clear to Send (CTS) input | Input |
| | | | 6 | CT18 | Timer/counter 18 | Output |
| | | | 7 | NCE18 | IOMSTR N Chip Select 18 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| A9 | E8 | 19 | 0 | ADCSE0 | Analog to Digital Converter SE IN0 | Input |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO19 | General purpose I/O | I/O |
| | | | 4 | UART2CTS | UART2 Clear to Send (CTS) input | Input |
| | | | 5 | UART3CTS | UART3 Clear to Send (CTS) input | Input |
| | | | 6 | CT19 | Timer/counter 19 | Output |
| | | | 7 | NCE19 | IOMSTR N Chip Select 19 | Output |
| | | | 9 | I2S1_SDIN | I2S1 Data input | Input |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| P4 | H13 | 20 | 0 | SWDCK | Software debug clock Input | Input |
| | | | 1 | TRIG1 | ADC trigger input | Input |
| | | | 2 | - | - | - |
| | | | 3 | GPIO20 | General purpose I/O | I/O |
| | | | 4 | UART0TX | UART0 transmit output | Output |
| | | | 5 | UART1TX | UART1 transmit output | Output |
| | | | 6 | CT20 | Timer/counter 20 | Output |
| | | | 7 | NCE20 | IOMSTR N Chip Select 20 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| N5 | J12 | 21 | 0 | SWDIO | Software data I/O | Bidirectional 3-state |
| | | | 1 | TRIG2 | ADC trigger input | Input |
| | | | 2 | - | - | - |
| | | | 3 | GPIO21 | General purpose I/O | I/O |
| | | | 4 | UART0RX | UART0 receive input | Input |
| | | | 5 | UART1RX | UART1 receive input | Input |
| | | | 6 | CT21 | Timer/counter 21 | Output |
| | | | 7 | NCE21 | IOMSTR N Chip Select 21 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| D9 | B8 | 22 | 0 | M7SCL | I2C Manager 7 Clk | Bidirectional Open Drain |
| | | | 1 | M7SCK | SPI Manager 7 Clk | Output |
| | | | 2 | SWO | Serial Wire Debug | Output |
| | | | 3 | GPIO22 | General purpose I/O | I/O |
| | | | 4 | UART2TX | UART2 transmit output | Output |
| | | | 5 | UART3TX | UART3 transmit output | Output |
| | | | 6 | CT22 | Timer/counter 22 | Output |
| | | | 7 | NCE22 | IOMSTR N Chip Select 22 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| C9 | A7 | 23 | 0 | M7SDAWIR3 | I2C Manager 7 I/O data  (I2C) 3 Wire data (SPI) | Bidirectional Open Drain |
| | | | 1 | M7MOSI | SPI Manager 7 data out | Output |
| | | | 2 | SWO | Serial Wire Debug | Output |
| | | | 3 | GPIO23 | General purpose I/O | I/O |
| | | | 4 | UART2RX | UART2 receive input | Input |
| | | | 5 | UART3RX | UART3 receive input | Input |
| | | | 6 | CT23 | Timer/counter 23 | Output |
| | | | 7 | NCE23 | IOMSTR N Chip Select 23 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| B9 | A8 | 24 | 0 | M7MISO | SPI Manager 7 data in | Input |
| | | | 1 | TRIG3 | ADC trigger input | Input |
| | | | 2 | SWO | Serial Wire Debug | Output |
| | | | 3 | GPIO24 | General purpose I/O | I/O |
| | | | 4 | UART0RTS | UART0 Request to Send (RTS) | Output |
| | | | 5 | UART1RTS | UART1 Request to Send (RTS) | Output |
| | | | 6 | CT24 | Timer/counter 24 | Output |
| | | | 7 | NCE24 | IOMSTR N Chip Select 24 | Output |
| | | | 9 | MNCE0_0 | MSPI Manager 0 nCE 0 Signal | - |
| | | | 10 | MNCE0_1 | MSPI Manager 0 nCE 1 Signal | - |
| | | | 12 | - | - | - |
| T6 | N10 | 25 | 0 | M2SCL | I2C Manager 2 clock | Open Drain Output |
| | | | 1 | M2SCK | SPI Manager 2 clock | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO25 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | UART1TX | UART1 transmit output | Output |
| | | | 6 | CT25 | Timer/counter 25 | Output |
| | | | 7 | NCE25 | IOMSTR N Chip Select 25 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| R6 | M10 | 26 | 0 | M2SDAWIR3 | I2C Manager 2 I/O data  (I2C) 3 Wire data (SPI) | Bidirectional Open Drain |
| | | | 1 | M2MOSI | SPI Manager 2 output data | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO26 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | UART1RX | UART1 receive input | Input |
| | | | 6 | CT26 | Timer/counter 26 | Output |
| | | | 7 | NCE26 | IOMSTR N Chip Select 26 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| P6 | N9 | 27 | 0 | M2MISO | SPI Manager 2 input data | Input |
| | | | 1 | TRIG0 | ADC trigger input | Input |
| | | | 2 | MNCE3_0 | MSPI Manager 3 nCE 0 Signal | - |
| | | | 3 | GPIO27 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | UART1CTS | UART1 Clear to Send (CTS) input | Input |
| | | | 6 | CT27 | Timer/counter 27 | Output |
| | | | 7 | NCE27 | IOMSTR N Chip Select 27 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| N4 | H12 | 28 | 0 | SWO | Serial Wire Debug | Output |
| | | | 1 | VCMPO | Output of the voltage comparator signal | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO28 | General purpose I/O | I/O |
| | | | 4 | UART2CTS | UART2 Clear to Send (CTS) input | Input |
| | | | 5 | - | - | - |
| | | | 6 | CT28 | Timer/counter 28 | Output |
| | | | 7 | NCE28 | IOMSTR N Chip Select 28 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| M4 | J11 | 29 | 0 | TRIG0 | ADC trigger input | Input |
| | | | 1 | VCMPO | Output of the voltage comparator signal | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO29 | General purpose I/O | I/O |
| | | | 4 | UART1CTS | UART1 Clear to Send (CTS) input | Input |
| | | | 5 | - | - | - |
| | | | 6 | CT29 | Timer/counter 29 | Output |
| | | | 7 | NCE29 | IOMSTR N Chip Select 29 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| P3 | K11 | 30 | 0 | TRIG1 | ADC trigger input | Input |
| | | | 1 | VCMPO | Output of the voltage comparator signal | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO30 | General purpose I/O | I/O |
| | | | 4 | UART0TX | UART0 transmit output | Output |
| | | | 5 | - | - | - |
| | | | 6 | CT30 | Timer/counter 30 | Output |
| | | | 7 | NCE30 | IOMSTR N Chip Select 30 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| N2 | N14 | 31 | 0 | M3SCL | I2C Manager 3 clock | Open Drain Output |
| | | | 1 | M3SCK | SPI Manager 3 clock | Output |
| | | | 2 | I2S0_CLK | I2S0 Bit clock | Bidirectional |
| | | | 3 | GPIO31 | General purpose I/O | I/O |
| | | | 4 | UART2TX | UART2 transmit output | Output |
| | | | 5 | UART2CTS | UART2 Clear to Send (CTS) input | Input |
| | | | 6 | CT31 | Timer/counter 31 | Output |
| | | | 7 | NCE31 | IOMSTR N Chip Select 31 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| M2 | M13 | 32 | 0 | M3SDAWIR3 | I2C Manager 3 I/O data  (I2C) 3 Wire data (SPI) | Bidirectional Open Drain |
| | | | 1 | M3MOSI | SPI Manager 3 output data | Output |
| | | | 2 | I2S0_DATA | I2S0 Data | Bidirectional |
| | | | 3 | GPIO32 | General purpose I/O | I/O |
| | | | 4 | UART0RX | UART0 receive input | Input |
| | | | 5 | UART3CTS | UART3 Clear to Send (CTS) input | Input |
| | | | 6 | CT32 | Timer/counter 32 | Output |
| | | | 7 | NCE32 | IOMSTR N Chip Select 32 | Output |
| | | | 9 | I2S0_SDOUT | I2S0 Data output | Output |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| L2 | M14 | 33 | 0 | M3MISO | SPI Manager 3 input data | Input |
| | | | 1 | CLKOUT | Oscillator output clock | Output |
| | | | 2 | I2S0_WS | I2S0 L/R clock | Bidirectional |
| | | | 3 | GPIO33 | General purpose I/O | I/O |
| | | | 4 | UART2RX | UART2 receive input | Input |
| | | | 5 | UART2RTS | UART2 Request to Send (RTS) | Output |
| | | | 6 | CT33 | Timer/counter 33 | Output |
| | | | 7 | NCE33 | IOMSTR N Chip Select 33 | Output |
| | | | 9 | DISP_TE | Display TE input | Input |
| | | | 10 | MNCE1_0 | MSPI Manager 1 nCE 0 Signal | - |
| | | | 12 | - | - | - |
| N1 | L14 | 34 | 0 | M4SCL | I2C Manager 4 Clk | Output |
| | | | 1 | M4SCK | SPI Manager 4 Clk | Output |
| | | | 2 | SWO | Serial Wire Debug | Output |
| | | | 3 | GPIO34 | General purpose I/O | I/O |
| | | | 4 | UART0TX | UART0 transmit output | Output |
| | | | 5 | UART2RX | UART2 receive input | Input |
| | | | 6 | CT34 | Timer/counter 34 | Output |
| | | | 7 | NCE34 | IOMSTR N Chip Select 34 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | I2S1_CLK | I2S1 Bit clock | Bidirectional |
| | | | 12 | - | - | - |
| M1 | L13 | 35 | 0 | M4SDAWIR3 | I2C Manager 4 I/O data  (I2C) 3 Wire data (SPI) | Bidirectional Open Drain |
| | | | 1 | M4MOSI | SPI Manager 4 data out | Input |
| | | | 2 | SWO | Serial Wire Debug | Output |
| | | | 3 | GPIO35 | General purpose I/O | I/O |
| | | | 4 | UART2TX | UART2 transmit output | Output |
| | | | 5 | UART3TX | UART3 transmit output | Output |
| | | | 6 | CT35 | Timer/counter 35 | Output |
| | | | 7 | NCE35 | IOMSTR N Chip Select 35 | Output |
| | | | 9 | I2S1_SDOUT | I2S1 Data output | Output |
| | | | 10 | I2S1_DATA | I2S1 Data | Bidirectional |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| L1 | K14 | 36 | 0 | M4MISO | SPI Manager 4 data in | Input |
| | | | 1 | TRIG0 | ADC trigger input | Input |
| | | | 2 | MNCE3_0 | MSPI Manager 3 nCE 0 Signal | - |
| | | | 3 | GPIO36 | General purpose I/O | I/O |
| | | | 4 | UART0RX | UART0 receive input | Input |
| | | | 5 | UART1RX | UART1 receive input | Input |
| | | | 6 | CT36 | Timer/counter 36 | Output |
| | | | 7 | NCE36 | IOMSTR N Chip Select 36 | Output |
| | | | 9 | MNCE1_0 | MSPI Manager 1 nCE 0 Signal | - |
| | | | 10 | I2S1_WS | I2S1 L/R clock | Bidirectional |
| | | | 12 | - | - | - |
| G16 | F2 | 37 | 0 | MSPI0_10 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | TRIG1 | ADC trigger input | Input |
| | | | 2 | 32KHzXT | 32kHZ from analog | Output |
| | | | 3 | GPIO37 | General purpose I/O | I/O |
| | | | 4 | UART2RX | UART2 receive input | Input |
| | | | 5 | UART3RX | UART3 receive input | Input |
| | | | 6 | CT37 | Timer/counter 37 | Output |
| | | | 7 | NCE37 | IOMSTR N Chip Select 37 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| G15 | G1 | 38 | 0 | MSPI0_11 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | TRIG2 | ADC trigger input | Input |
| | | | 2 | SWTRACECLK | Serial Wire Debug Trace Clock | Output |
| | | | 3 | GPIO38 | General purpose I/O | I/O |
| | | | 4 | UART0RTS | UART0 Request to Send (RTS) | Output |
| | | | 5 | UART2RTS | UART2 Request to Send (RTS) | Output |
| | | | 6 | CT38 | Timer/counter 38 | Output |
| | | | 7 | NCE38 | IOMSTR N Chip Select 38 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| G14 | F4 | 39 | 0 | MSPI0_12 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | TRIG3 | ADC trigger input | Input |
| | | | 2 | SWTRACE0 | Serial Wire Debug Trace Output 0 | Output |
| | | | 3 | GPIO39 | General purpose I/O | I/O |
| | | | 4 | UART2RTS | UART2 Request to Send (RTS) | Output |
| | | | 5 | UART3RTS | UART3 Request to Send (RTS) | Output |
| | | | 6 | CT39 | Timer/counter 39 | Output |
| | | | 7 | NCE39 | IOMSTR N Chip Select 39 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| G13 | D2 | 40 | 0 | MSPI0_13 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | TRIG1 | ADC trigger input | Input |
| | | | 2 | SWTRACE1 | Serial Wire Debug Trace Output 1 | Output |
| | | | 3 | GPIO40 | General purpose I/O | I/O |
| | | | 4 | UART0CTS | UART0 Clear to Send (CTS) | Input |
| | | | 5 | UART1CTS | UART1 Clear to Send (CTS) input | Input |
| | | | 6 | CT40 | Timer/counter 40 | Output |
| | | | 7 | NCE40 | IOMSTR N Chip Select 40 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| F14 | E4 | 41 | 0 | MSPI0_14 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | TRIG0 | ADC trigger input | Input |
| | | | 2 | SWTRACE2 | Serial Wire Debug Trace Output 2 | Output |
| | | | 3 | GPIO41 | General purpose I/O | I/O |
| | | | 4 | UART0TX | UART0 transmit output | Output |
| | | | 5 | UART1TX | UART1 transmit output | Output |
| | | | 6 | CT41 | Timer/counter 41 | Output |
| | | | 7 | NCE41 | IOMSTR N Chip Select 41 | Output |
| | | | 9 | SWO | Serial Wire Debug | Output |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| J14 | H1 | 42 | 0 | MSPI0_15 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | TRIG2 | ADC trigger input | Input |
| | | | 2 | SWTRACE3 | Serial Wire Debug Trace Output 3 | Output |
| | | | 3 | GPIO42 | General purpose I/O | I/O |
| | | | 4 | UART2TX | UART2 transmit output | Output |
| | | | 5 | UART3TX | UART3 transmit output | Output |
| | | | 6 | CT42 | Timer/counter 42 | Output |
| | | | 7 | NCE42 | IOMSTR N Chip Select 42 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| H16 | G2 | 43 | 0 | MSPI0_16 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | TRIG3 | ADC trigger input | Input |
| | | | 2 | SWTRACECTL | Serial Wire Debug Trace Control | Output |
| | | | 3 | GPIO43 | General purpose I/O | I/O |
| | | | 4 | UART0RX | UART0 receive input | Input |
| | | | 5 | UART1RX | UART1 receive input | Input |
| | | | 6 | CT43 | Timer/counter 43 | Output |
| | | | 7 | NCE43 | IOMSTR N Chip Select 43 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| H15 | F3 | 44 | 0 | MSPI0_17 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | TRIG1 | ADC trigger input | Input |
| | | | 2 | SWO | Serial Wire Debug | Output |
| | | | 3 | GPIO44 | General purpose I/O | I/O |
| | | | 4 | UART2RX | UART2 receive input | Input |
| | | | 5 | UART3RX | UART3 receive input | Input |
| | | | 6 | CT44 | Timer/counter 44 | Output |
| | | | 7 | NCE44 | IOMSTR N Chip Select 44 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| F15 | D3 | 45 | 0 | MSPI0_18 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | TRIG2 | ADC trigger input | Input |
| | | | 2 | 32KHzXT | 32kHZ from analog | Output |
| | | | 3 | GPIO45 | General purpose I/O | I/O |
| | | | 4 | UART0TX | UART0 transmit output | Output |
| | | | 5 | UART1TX | UART1 transmit output | Output |
| | | | 6 | CT45 | Timer/counter 45 | Output |
| | | | 7 | NCE45 | IOMSTR N Chip Select 45 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| E15 | E3 | 46 | 0 | - | - | - |
| | | | 1 | TRIG3 | ADC trigger input | Input |
| | | | 2 | CLKOUT_32M | 32MHz Oscillator output clock | Output |
| | | | 3 | GPIO46 | General purpose I/O | I/O |
| | | | 4 | UART2TX | UART2 transmit output | Output |
| | | | 5 | UART3TX | UART3 transmit output | Output |
| | | | 6 | CT46 | Timer/counter 46 | Output |
| | | | 7 | NCE46 | IOMSTR/MSPI N Chip Select 46 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| H1 | J13 | 47 | 0 | M5SCL | I2C Manager 5 Clk | Bidirectional Open Drain |
| | | | 1 | M5SCK | SPI Manager 5 Clk | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO47 | General purpose I/O | I/O |
| | | | 4 | UART0RX | UART0 receive input | Input |
| | | | 5 | UART1RX | UART1 receive input | Input |
| | | | 6 | CT47 | Timer/counter 47 | Output |
| | | | 7 | NCE47 | IOMSTR N Chip Select 47 | Output |
| | | | 9 | - | - | - |
| | | | 10 | I2S0_CLK | I2S0 Bit clock | Bidirectional |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| J1 | J14 | 48 | 0 | M5SDAWIR3 | I2C Manager 5 I/O data  (I2C) 3 Wire data (SPI) | Bidirectional Open Drain |
| | | | 1 | M5MOSI | SPI Manager 5 data out | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO48 | General purpose I/O | I/O |
| | | | 4 | UART2RX | UART2 receive input | Input |
| | | | 5 | UART3RX | UART3 receive input | Input |
| | | | 6 | CT48 | Timer/counter 48 | Output |
| | | | 7 | NCE48 | IOMSTR N Chip Select 48 | Output |
| | | | 9 | - | - | - |
| | | | 10 | I2S0_WS | I2S0 L/R clock | Bidirectional |
| | | | 12 | - | - | - |
| G1 | H14 | 49 | 0 | M5MISO | SPI Manager 5 data in | Input |
| | | | 1 | TRIG0 | ADC trigger input | Input |
| | | | 2 | MNCE1_0 | MSPI Manager 1 nCE 0 Signal | - |
| | | | 3 | GPIO49 | General purpose I/O | I/O |
| | | | 4 | UART0RTS | UART0 Request to Send (RTS) | Output |
| | | | 5 | UART1RTS | UART1 Request to Send (RTS) | Output |
| | | | 6 | CT49 | Timer/counter 49 | Output |
| | | | 7 | NCE49 | IOMSTR N Chip Select 49 | Output |
| | | | 9 | I2S0_DATA | I2S0 Data | Bidirectional |
| | | | 10 | I2S0_SDOUT | I2S0 Data output | Output |
| | | | 12 | MNCE1_1 | MSPI Manager 1 nCE 1 Signal | - |
| A13 | C4 | 50 | 0 | PDM0_CLK | PDM0 Clock output | Output |
| | | | 1 | TRIG0 | ADC trigger input | Input |
| | | | 2 | SWTRACECLK | Serial Wire Debug Trace Clock | Output |
| | | | 3 | GPIO50 | General purpose I/O | I/O |
| | | | 4 | UART2RTS | UART2 Request to Send (RTS) | Output |
| | | | 5 | UART3RTS | UART3 Request to Send (RTS) | Output |
| | | | 6 | CT50 | Timer/counter 50 | Output |
| | | | 7 | NCE50 | IOMSTR N Chip Select 50 | Output |
| | | | 9 | DISP_TE | Display TE input | Input |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| B13 | B4 | 51 | 0 | PDM0_DATA | PDM0 audio data input to chip | Input |
| | | | 1 | TRIG1 | ADC trigger input | Input |
| | | | 2 | SWTRACE0 | Serial Wire Debug Trace Output 0 | Output |
| | | | 3 | GPIO51 | General purpose I/O | I/O |
| | | | 4 | UART0CTS | UART0 Clear to Send (CTS) | Input |
| | | | 5 | UART1CTS | UART1 Clear to Send (CTS) input | Input |
| | | | 6 | CT51 | Timer/counter 51 | Output |
| | | | 7 | NCE51 | IOMSTR N Chip Select 51 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| A12 | D6 | 52 | 0 | MNCE3_0 | MSPI Manager 3 nCE 0 Signal | - |
| | | | 1 | TRIG2 | ADC trigger input | Input |
| | | | 2 | SWTRACE1 | Serial Wire Debug Trace Output 1 | Output |
| | | | 3 | GPIO52 | General purpose I/O | I/O |
| | | | 4 | UART2CTS | UART2 Clear to Send (CTS) input | Input |
| | | | 5 | UART3CTS | UART3 Clear to Send (CTS) input | Input |
| | | | 6 | CT52 | Timer/counter 52 | Output |
| | | | 7 | NCE52 | IOMSTR/MSPI N Chip Select 52 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | I2S0_CLK | I2S0 Bit clock | Bidirectional |
| | | | 12 | SLSDAWIR3 | I2C Sub Half Duplex I/O data (I2C) 3 Wire Data (SPI) | Bidirectional Open Drain |
| B12 | A4 | 53 | 0 | MNCE1_0 | MSPI Manager 1 nCE 0 Signal | - |
| | | | 1 | TRIG3 | ADC trigger input | Input |
| | | | 2 | SWTRACE2 | Serial Wire Debug Trace Output 2 | Output |
| | | | 3 | GPIO53 | General purpose I/O | I/O |
| | | | 4 | UART0TX | UART0 transmit output | Output |
| | | | 5 | UART1TX | UART1 transmit output | Output |
| | | | 6 | CT53 | Timer/counter 53 | Output |
| | | | 7 | NCE53 | IOMSTR N Chip Select 53 | Output |
| | | | 9 | - | - | - |
| | | | 10 | I2S0_WS | I2S0 L/R clock | Bidirectional |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| D13 | D4 | 54 | 0 | MNCE1_1 | MSPI Manager 1 nCE 1 Signal | - |
| | | | 1 | TRIG0 | ADC trigger input | Input |
| | | | 2 | SWTRACE3 | Serial Wire Debug Trace Output 3 | Output |
| | | | 3 | GPIO54 | General purpose I/O | I/O |
| | | | 4 | UART2TX | UART2 transmit output | Output |
| | | | 5 | UART3TX | UART3 transmit output | Output |
| | | | 6 | CT54 | Timer/counter 54 | Output |
| | | | 7 | NCE54 | IOMSTR N Chip Select 54 | Output |
| | | | 9 | I2S0_DATA | I2S0 Data | Bidirectional |
| | | | 10 | I2S0_SDOUT | I2S0 Data output | Output |
| | | | 12 | - | - | - |
| E12 | C5 | 55 | 0 | MNCE3_1 | MSPI Manager 3 nCE 1 Signal | - |
| | | | 1 | TRIG1 | ADC trigger input | Input |
| | | | 2 | SWTRACECTL | Serial Wire Debug Trace Control | Output |
| | | | 3 | GPIO55 | General purpose I/O | I/O |
| | | | 4 | UART0RX | UART0 receive input | Input |
| | | | 5 | UART1RX | UART1 receive input | Input |
| | | | 6 | CT55 | Timer/counter 55 | Output |
| | | | 7 | NCE55 | IOMSTR N Chip Select 55 | Output |
| | | | 9 | - | - | - |
| | | | 10 | I2S1_CLK | I2S1 Bit clock | Bidirectional |
| | | | 12 | - | - | - |
| E11 | D5 | 56 | 0 | MNCE0_0 | MSPI Manager 0 nCE 0 Signal | - |
| | | | 1 | TRIG2 | ADC trigger input | Input |
| | | | 2 | SWO | Serial Wire Debug | Output |
| | | | 3 | GPIO56 | General purpose I/O | I/O |
| | | | 4 | UART2RX | UART2 receive input | Input |
| | | | 5 | UART3RX | UART3 receive input | Input |
| | | | 6 | CT56 | Timer/counter 56 | Output |
| | | | 7 | NCE56 | IOMSTR/MSPI N Chip Select 56 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | I2S1_WS | I2S1 L/R clock | Bidirectional |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| E10 | C6 | 57 | 0 | MNCE0_1 | MSPI Manager 0 nCE 1 Signal | - |
| | | | 1 | TRIG3 | ADC trigger input | Input |
| | | | 2 | SWO | Serial Wire Debug | Output |
| | | | 3 | GPIO57 | General purpose I/O | I/O |
| | | | 4 | UART0RTS | UART0 Request to Send (RTS) | Output |
| | | | 5 | UART1RTS | UART1 Request to Send (RTS) | Output |
| | | | 6 | CT57 | Timer/counter 57 | Output |
| | | | 7 | NCE57 | IOMSTR/MSPI N Chip Select 57 | Output |
| | | | 9 | I2S1_DATA | I2S1 Data | Bidirectional |
| | | | 10 | I2S1_SDOUT | I2S1 Data output | Output |
| | | | 12 | - | - | - |
| P7 | NC | 58 | 0 | - | - | - |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO58 | General purpose I/O | I/O |
| | | | 4 | UART0RTS | UART0 Request to Send (RTS) | Output |
| | | | 5 | UART3RTS | UART3 Request to Send (RTS) | Output |
| | | | 6 | CT58 | Timer/counter 58 | Output |
| | | | 7 | NCE58 | IOMSTR N Chip Select 58 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| R7 | H11 | 59 | 0 | - | - | - |
| | | | 1 | TRIG0 | ADC trigger input | Input |
| | | | 2 | - | - | - |
| | | | 3 | GPIO59 | General purpose I/O | I/O |
| | | | 4 | UART0CTS | UART0 Clear to Send (CTS) | Input |
| | | | 5 | UART1CTS | UART1 Clear to Send (CTS) input | Input |
| | | | 6 | CT59 | Timer/counter 59 | Output |
| | | | 7 | NCE59 | IOMSTR N Chip Select 59 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| N8 | J10 | 60 | 0 | - | - | - |
| | | | 1 | TRIG1 | ADC trigger input | Input |
| | | | 2 | - | - | - |
| | | | 3 | GPIO60 | General purpose I/O | I/O |
| | | | 4 | UART0TX | UART0 transmit output | Output |
| | | | 5 | UART3CTS | UART3 Clear to Send (CTS) input | Input |
| | | | 6 | CT60 | Timer/counter 60 | Output |
| | | | 7 | NCE60 | IOMSTR/MSPI N Chip Select 60 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| P5 | K12 | 61 | 0 | M6SCL | I2C Manager 6 Clk | Bidirectional Open Drain |
| | | | 1 | M6SCK | SPI Manager 6 Clk | Output |
| | | | 2 | I2S1_CLK | I2S1 Bit clock | Bidirectional |
| | | | 3 | GPIO61 | General purpose I/O | I/O |
| | | | 4 | UART2TX | UART2 transmit output | Output |
| | | | 5 | UART3TX | UART3 transmit output | Output |
| | | | 6 | CT61 | Timer/counter 61 | Output |
| | | | 7 | NCE61 | IOMSTR N Chip Select 61 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| R5 | L12 | 62 | 0 | M6SDAWIR3 | I2C Manager 6 I/O data (I2C) 3 Wire data (SPI) | Bidirectional Open Drain |
| | | | 1 | M6MOSI | SPI Manager 6 data out | Output |
| | | | 2 | I2S1_DATA | I2S1 Data | Bidirectional |
| | | | 3 | GPIO62 | General purpose I/O | I/O |
| | | | 4 | UART0RX | UART0 receive input | Input |
| | | | 5 | UART1RX | UART1 receive input | Input |
| | | | 6 | CT62 | Timer/counter 62 | Output |
| | | | 7 | NCE62 | IOMSTR N Chip Select 62 | Output |
| | | | 9 | I2S1_SDOUT | I2S1 Data output | Output |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| T5 | L11 | 63 | 0 | M6MISO | SPI Manager 6 data in | Input |
| | | | 1 | CLKOUT | Oscillator output clock | Output |
| | | | 2 | I2S1_WS | I2S1 L/R clock | Bidirectional |
| | | | 3 | GPIO63 | General purpose I/O | I/O |
| | | | 4 | UART2RX | UART2 receive input | Input |
| | | | 5 | UART3RX | UART3 receive input | Input |
| | | | 6 | CT63 | Timer/counter 63 | Output |
| | | | 7 | NCE63 | IOMSTR N Chip Select 63 | Output |
| | | | 9 | DISP_TE | Display TE input | Input |
| | | | 10 | MNCE2_0 | MSPI Manager 2 nCE 0 Signal | - |
| | | | 12 | MNCE2_1 | MSPI Manager 2 nCE 1 Signal | - |
| J13 | G3 | 64 | 0 | MSPI0_0 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | 32KHzXT | 32kHZ from analog | Output |
| | | | 2 | SWO | Serial Wire Debug | Output |
| | | | 3 | GPIO64 | General purpose I/O | I/O |
| | | | 4 | UART0RTS | UART0 Request to Send (RTS) | Output |
| | | | 5 | UART2CTS | UART2 Clear to Send (CTS) input | Input |
| | | | 6 | CT64 | Timer/counter 64 | Output |
| | | | 7 | NCE64 | IOMSTR N Chip Select 64 | Output |
| | | | 9 | I2S1_SDIN | I2S1 Data input | Input |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| K14 | J1 | 65 | 0 | MSPI0_1 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | 32KHzXT | 32kHZ from analog | Output |
| | | | 2 | SWO | Serial Wire Debug | Output |
| | | | 3 | GPIO65 | General purpose I/O | I/O |
| | | | 4 | UART0CTS | UART0 Clear to Send (CTS) | Input |
| | | | 5 | UART1CTS | UART1 Clear to Send (CTS) input | Input |
| | | | 6 | CT65 | Timer/counter 65 | Output |
| | | | 7 | NCE65 | IOMSTR N Chip Select 65 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| J15 | J2 | 66 | 0 | MSPI0_2 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | CLKOUT | Oscillator output clock | Output |
| | | | 2 | SWO | Serial Wire Debug | Output |
| | | | 3 | GPIO66 | General purpose I/O | I/O |
| | | | 4 | UART0TX | UART0 transmit output | Output |
| | | | 5 | UART1TX | UART1 transmit output | Output |
| | | | 6 | CT66 | Timer/counter 66 | Output |
| | | | 7 | NCE66 | IOMSTR N Chip Select 66 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| J16 | J3 | 67 | 0 | MSPI0_3 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | CLKOUT | Oscillator output clock | Output |
| | | | 2 | SWO | Serial Wire Debug | Output |
| | | | 3 | GPIO67 | General purpose I/O | I/O |
| | | | 4 | UART2TX | UART2 transmit output | Output |
| | | | 5 | UART3TX | UART3 transmit output | Output |
| | | | 6 | CT67 | Timer/counter 67 | Output |
| | | | 7 | NCE67 | IOMSTR N Chip Select 67 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| K15 | K2 | 68 | 0 | MSPI0_4 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | SWO | Serial Wire Debug | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO68 | General purpose I/O | I/O |
| | | | 4 | UART0RX | UART0 receive input | Input |
| | | | 5 | UART1RX | UART1 receive input | Input |
| | | | 6 | CT68 | Timer/counter 68 | Output |
| | | | 7 | NCE68 | IOMSTR N Chip Select 68 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| K16 | L4 | 69 | 0 | MSPI0_5 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | 32KHzXT | 32kHZ from analog | Output |
| | | | 2 | SWO | Serial Wire Debug | Output |
| | | | 3 | GPIO69 | General purpose I/O | I/O |
| | | | 4 | UART2RX | UART2 receive input | Input |
| | | | 5 | UART3RX | UART3 receive input | Input |
| | | | 6 | CT69 | Timer/counter 69 | Output |
| | | | 7 | NCE69 | IOMSTR N Chip Select 69 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| L15 | L2 | 70 | 0 | MSPI0_6 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | 32KHzXT | 32kHZ from analog | Output |
| | | | 2 | SWTRACE0 | Serial Wire Debug Trace Output 0 | Output |
| | | | 3 | GPIO70 | General purpose I/O | I/O |
| | | | 4 | UART0RTS | UART0 Request to Send (RTS) | Output |
| | | | 5 | UART1RTS | UART1 Request to Send (RTS) | Output |
| | | | 6 | CT70 | Timer/counter 70 | Output |
| | | | 7 | NCE70 | IOMSTR N Chip Select 70 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| M15 | L3 | 71 | 0 | MSPI0_7 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | CLKOUT | Oscillator output clock | Output |
| | | | 2 | SWTRACE1 | Serial Wire Debug Trace Output 1 | Output |
| | | | 3 | GPIO71 | General purpose I/O | I/O |
| | | | 4 | UART0CTS | UART0 Clear to Send (CTS) | Input |
| | | | 5 | UART3RTS | UART3 Request to Send (RTS) | Output |
| | | | 6 | CT71 | Timer/counter 71 | Output |
| | | | 7 | NCE71 | IOMSTR N Chip Select 71 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| M14 | M2 | 72 | 0 | MSPI0_8 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | CLKOUT | Oscillator output clock | Output |
| | | | 2 | SWTRACE2 | Serial Wire Debug Trace Output 2 | Output |
| | | | 3 | GPIO72 | General purpose I/O | I/O |
| | | | 4 | UART0TX | UART0 transmit output | Output |
| | | | 5 | UART1TX | UART1 transmit output | Output |
| | | | 6 | CT72 | Timer/counter 72 | Output |
| | | | 7 | NCE72 | IOMSTR N Chip Select 72 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| N14 | M3 | 73 | 0 | MSPI0_9 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | SWTRACE3 | Serial Wire Debug Trace Output 3 | Output |
| | | | 3 | GPIO73 | General purpose I/O | I/O |
| | | | 4 | UART2TX | UART2 transmit output | Output |
| | | | 5 | UART3TX | UART3 transmit output | Output |
| | | | 6 | CT73 | Timer/counter 73 | Output |
| | | | 7 | NCE73 | IOMSTR/MSPI N Chip Select 73 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| T11 | N7 | 74 | 0 | MSPI2_0 | MSPI Manager 2 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO74 | General purpose I/O | I/O |
| | | | 4 | UART0RX | UART0 receive input | Input |
| | | | 5 | UART3CTS | UART3 Clear to Send (CTS) input | Input |
| | | | 6 | CT74 | Timer/counter 74 | Output |
| | | | 7 | NCE74 | IOMSTR N Chip Select 74 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| P11 | M7 | 75 | 0 | MSPI2_1 | MSPI Manager 2 Interface Signal | I/O |
| | | | 1 | 32KHzXT | 32kHZ from analog | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO75 | General purpose I/O | I/O |
| | | | 4 | UART2RX | UART2 receive input | Input |
| | | | 5 | UART3RX | UART3 receive input | Input |
| | | | 6 | CT75 | Timer/counter 75 | Output |
| | | | 7 | NCE75 | IOMSTR N Chip Select 75 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| P12 | N6 | 76 | 0 | MSPI2_2 | MSPI Manager 2 Interface Signal | I/O |
| | | | 1 | 32KHzXT | 32kHZ from analog | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO76 | General purpose I/O | I/O |
| | | | 4 | UART0RTS | UART0 Request to Send (RTS) | Output |
| | | | 5 | UART1RTS | UART1 Request to Send (RTS) | Output |
| | | | 6 | CT76 | Timer/counter 76 | Output |
| | | | 7 | NCE76 | IOMSTR N Chip Select 76 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| R12 | M6 | 77 | 0 | MSPI2_3 | MSPI Manager 2 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO77 | General purpose I/O | I/O |
| | | | 4 | UART0CTS | UART0 Clear to Send (CTS) | Input |
| | | | 5 | UART1CTS | UART1 Clear to Send (CTS) input | Input |
| | | | 6 | CT77 | Timer/counter 77 | Output |
| | | | 7 | NCE77 | IOMSTR N Chip Select 77 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| T12 | L6 | 78 | 0 | MSPI2_4 | MSPI Manager 2 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO78 | General purpose I/O | I/O |
| | | | 4 | UART0TX | UART0 transmit output | Output |
| | | | 5 | UART1TX | UART1 transmit output | Output |
| | | | 6 | CT78 | Timer/counter 78 | Output |
| | | | 7 | NCE78 | IOMSTR N Chip Select 78 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| R11 | K7 | 79 | 0 | MSPI2_5 | MSPI Manager 2 Interface Signal | I/O |
| | | | 1 | DISP_QSPI_D0_OUT | Display SPI Data0 | Output |
| | | | 2 | DISP_QSPI_D0 | Display SPI Data0 | Bidirectional |
| | | | 3 | GPIO79 | General purpose I/O | I/O |
| | | | 4 | SWO | Serial Wire Debug | Output |
| | | | 5 | UART1RTS | UART1 Request to Send (RTS) | Output |
| | | | 6 | CT79 | Timer/counter 79 | Output |
| | | | 7 | NCE79 | IOMSTR N Chip Select 79 | Output |
| | | | 9 | DISP_SPI_SD | Display SPI Data Out | Bidirectional |
| | | | 10 | DISP_SPI_SDO | Display SPI Data Out | Output |
| | | | 12 | - | - | - |
| P9 | M8 | 80 | 0 | MSPI2_6 | MSPI Manager 2 Interface Signal | I/O |
| | | | 1 | CLKOUT | Oscillator output clock | Output |
| | | | 2 | DISP_QSPI_D1 | Display SPI Data1 | Output |
| | | | 3 | GPIO80 | General purpose I/O | I/O |
| | | | 4 | SWTRACE0 | Serial Wire Debug Trace Output 0 | Output |
| | | | 5 | UART2CTS | UART2 Clear to Send (CTS) input | Input |
| | | | 6 | CT80 | Timer/counter 80 | Output |
| | | | 7 | NCE80 | IOMSTR N Chip Select 80 | Output |
| | | | 9 | DISP_SPI_DCX | Display SPI DCx | Output |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| T10 | N8 | 81 | 0 | MSPI2_7 | MSPI Manager 2 Interface Signal | I/O |
| | | | 1 | CLKOUT | Oscillator output clock | Output |
| | | | 2 | DISP_QSPI_SCK | Display SPI CLK | Output |
| | | | 3 | GPIO81 | General purpose I/O | I/O |
| | | | 4 | SWTRACE1 | Serial Wire Debug Trace Output 1 | Output |
| | | | 5 | UART2RTS | UART2 Request to Send (RTS) | Output |
| | | | 6 | CT81 | Timer/counter 81 | Output |
| | | | 7 | NCE81 | IOMSTR N Chip Select 81 | Output |
| | | | 9 | DISP_SPI_SCK | Display SPI Clock | Output |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| P10 | L7 | 82 | 0 | MSPI2_8 | MSPI Manager 2 Interface Signal | I/O |
| | | | 1 | 32KHzXT | 32kHZ from analog | Output |
| | | | 2 | DISP_QSPI_D2 | Display SPI Data2 | Output |
| | | | 3 | GPIO82 | General purpose I/O | I/O |
| | | | 4 | SWTRACE2 | Serial Wire Debug Trace Output 2 | Output |
| | | | 5 | UART3CTS | UART3 Clear to Send (CTS) input | Input |
| | | | 6 | CT82 | Timer/counter 82 | Output |
| | | | 7 | NCE82 | IOMSTR N Chip Select 82 | Output |
| | | | 9 | DISP_SPI_SDI | Display SPI Data IN | input |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| R10 | K6 | 83 | 0 | MSPI2_9 | MSPI Manager 2 Interface Signal | I/O |
| | | | 1 | 32KHzXT | 32kHZ from analog | Output |
| | | | 2 | DISP_QSPI_D3 | Display SPI Data3 | Output |
| | | | 3 | GPIO83 | General purpose I/O | I/O |
| | | | 4 | SWTRACE3 | Serial Wire Debug Trace Output 3 | Output |
| | | | 5 | UART3RTS | UART3 Request to Send (RTS) | Output |
| | | | 6 | CT83 | Timer/counter 83 | Output |
| | | | 7 | NCE83 | IOMSTR/MSPI N Chip Select 83 | Output |
| | | | 9 | DISP_SPI_RST | Display SPI Reset | Output |
| | | | 10 | SLMISO | SPI Subordinate Half Duplex output data | Output |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| L9 | L8 | 84 | 0 | - | - | - |
| | | | 1 | - | - | - |
| | | | 2 | SDIF0_DAT0 | SD/SDIO/MMC Data0 pin | I/O |
| | | | 3 | GPIO84 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT84 | Timer/counter 84 | Output |
| | | | 7 | NCE84 | IOMSTR/MSPI N Chip Select 84 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| L10 | G8 | 85 | 0 | - | - | - |
| | | | 1 | - | - | - |
| | | | 2 | SDIF0_DAT1 | SD/SDIO/MMC Data1 pin | I/O |
| | | | 3 | GPIO85 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT85 | Timer/counter 85 | Output |
| | | | 7 | NCE85 | IOMSTR/MSPI N Chip Select 85 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| K9 | K8 | 86 | 0 | - | - | - |
| | | | 1 | - | - | - |
| | | | 2 | SDIF0_DAT2 | SD/SDIO/MMC Data2 pin | I/O |
| | | | 3 | GPIO86 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT86 | Timer/counter 86 | Output |
| | | | 7 | NCE86 | IOMSTR/MSPI N Chip Select 86 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| K10 | J8 | 87 | 0 | - | - | - |
| | | | 1 | - | - | - |
| | | | 2 | SDIF0_DAT3 | SD/SDIO/MMC Data3 pin | I/O |
| | | | 3 | GPIO87 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT87 | Timer/counter 87 | Output |
| | | | 7 | NCE87 | IOMSTR N Chip Select 87 | Output |
| | | | 9 | DISP_TE | Display TE input | Input |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| M10 | H9 | 88 | 0 | - | - | - |
| | | | 1 | - | - | - |
| | | | 2 | SDIF0_CLKOUT | SD/SDIO/MMC Clock to Card (CLK) | Output |
| | | | 3 | GPIO88 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT88 | Timer/counter 88 | Output |
| | | | 7 | NCE88 | IOMSTR N Chip Select 88 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| N6 | NC | 89 | 0 | - | - | - |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO89 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT89 | Timer/counter 89 | Output |
| | | | 7 | NCE89 | IOMSTR/MSPI N Chip Select 89 | Output |
| | | | 9 | DISP_SPI_RST | Display SPI Reset | Output |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| G10 | NC | 90 | 0 | - | - | - |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO90 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT90 | Timer/counter 89 | Output |
| | | | 7 | NCE90 | - | - |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| G9 | NC | 91 | 0 | - | - | - |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO91 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT91 | Timer/counter 89 | Output |
| | | | 7 | NCE91 | IOMSTR/MSPI N Chip Select 91 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| E8 | NC | 92 | 0 | - | - | - |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO92 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT92 | Timer/counter 89 | Output |
| | | | 7 | NCE92 | IOMSTR N Chip Select 92 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| J2 | G14 | 93 | 0 | MNCE1_0 | MSPI Manager 1 nCE 0 Signal | - |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO93 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT93 | Timer/counter 89 | Output |
| | | | 7 | NCE93 | IOMSTR/MSPI N Chip Select 93 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| F9 | NC | 94 | 0 | - | - | - |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO94 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT94 | Timer/counter 89 | Output |
| | | | 7 | NCE94 | IOMSTR N Chip Select 94 | Output |
| | | | 9 | VCMPO | Output of the voltage comparator signal | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| E3 | G13 | 95 | 0 | MSPI1_0 | MSPI Manager 1 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO95 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT95 | Timer/counter 89 | Output |
| | | | 7 | NCE95 | IOMSTR N Chip Select 95 | Output |
| | | | 9 | I2S0_SDIN | I2S0 Data input | Input |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| E4 | G12 | 96 | 0 | MSPI1_1 | MSPI Manager 1 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO96 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT96 | Timer/counter 89 | Output |
| | | | 7 | NCE96 | IOMSTR N Chip Select 96 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| E2 | G11 | 97 | 0 | MSPI1_2 | MSPI Manager 1 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO97 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT97 | Timer/counter 89 | Output |
| | | | 7 | NCE97 | IOMSTR N Chip Select 97 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| D2 | F12 | 98 | 0 | MSPI1_3 | MSPI Manager 1 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO98 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT98 | Timer/counter 89 | Output |
| | | | 7 | NCE98 | IOMSTR N Chip Select 98 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| D3 | F11 | 99 | 0 | MSPI1_4 | MSPI Manager 1 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO99 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT99 | Timer/counter 89 | Output |
| | | | 7 | NCE99 | IOMSTR N Chip Select 99 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| D4 | E11 | 100 | 0 | MSPI1_5 | MSPI Manager 1 Interface Signal | I/O |
| | | | 1 | DISP_QSPI_D0_OUT | Display SPI Data0 | Output |
| | | | 2 | DISP_QSPI_D0 | Display SPI Data0 | Bidirectional |
| | | | 3 | GPIO100 | General purpose I/O | I/O |
| | | | 4 | DISP_SPI_SD | Display SPI Data Out | Bidirectional |
| | | | 5 | DISP_SPI_SDO | Display SPI Data Out | Output |
| | | | 6 | CT100 | Timer/counter 100 | Output |
| | | | 7 | NCE100 | IOMSTR N Chip Select 100 | Output |
| | | | 9 | I2S0_CLK | I2S0 Bit clock | Bidirectional |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| C3 | E12 | 101 | 0 | MSPI1_6 | MSPI Manager 1 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | DISP_QSPI_D1 | Display SPI Data1 | Output |
| | | | 3 | GPIO101 | General purpose I/O | I/O |
| | | | 4 | DISP_SPI_DCX | Display SPI DCx | Output |
| | | | 5 | - | - | - |
| | | | 6 | CT101 | Timer/counter 101 | Output |
| | | | 7 | NCE101 | IOMSTR N Chip Select 101 | Output |
| | | | 9 | I2S0_DATA | I2S0 Data | Bidirectional |
| | | | 10 | I2S0_SDOUT | I2S0 Data output | Output |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| D5 | E10 | 102 | 0 | MSPI1_7 | MSPI Manager 1 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | DISP_QSPI_SCK | Display SPI CLK | Output |
| | | | 3 | GPIO102 | General purpose I/O | I/O |
| | | | 4 | DISP_SPI_SCK | Display SPI Clock | Output |
| | | | 5 | - | - | - |
| | | | 6 | CT102 | Timer/counter 102 | Output |
| | | | 7 | NCE102 | IOMSTR N Chip Select 102 | Output |
| | | | 9 | I2S0_WS | I2S0 L/R clock | Bidirectional |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| C5 | D11 | 103 | 0 | MSPI1_8 | MSPI Manager 1 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | DISP_QSPI_D2 | Display SPI Data2 | Output |
| | | | 3 | GPIO103 | General purpose I/O | I/O |
| | | | 4 | DISP_SPI_SDI | Display SPI Data IN | input |
| | | | 5 | - | - | - |
| | | | 6 | CT103 | Timer/counter 103 | Output |
| | | | 7 | NCE103 | IOMSTR N Chip Select 103 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| D6 | D12 | 104 | 0 | MSPI1_9 | MSPI Manager 1 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | DISP_QSPI_D3 | Display SPI Data3 | Output |
| | | | 3 | GPIO104 | General purpose I/O | I/O |
| | | | 4 | DISP_SPI_RST | Display SPI Reset | Output |
| | | | 5 | - | - | - |
| | | | 6 | CT104 | Timer/counter 104 | Output |
| | | | 7 | NCE104 | IOMSTR N Chip Select 104 | Output |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| J12 | H8 | 105 | 0 | MSPI3_10 | MSPI Manager 0 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO105 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT105 | Timer/counter 105 | Output |
| | | | 7 | NCE105 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| J11 | G4 | 106 | 0 | MSPI3_11 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO106 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT106 | Timer/counter 106 | Output |
| | | | 7 | NCE106 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| J10 | G5 | 107 | 0 | MSPI3_12 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO107 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT107 | Timer/counter 107 | Output |
| | | | 7 | NCE107 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| H13 | G6 | 108 | 0 | MSPI3_13 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO108 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT108 | Timer/counter 108 | Output |
| | | | 7 | NCE108 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| H12 | G7 | 109 | 0 | MSPI3_14 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO109 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT109 | Timer/counter 109 | Output |
| | | | 7 | NCE109 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| H11 | F5 | 110 | 0 | MSPI3_15 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO110 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT110 | Timer/counter 110 | Output |
| | | | 7 | NCE110 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| H14 | F6 | 111 | 0 | MSPI3_16 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO111 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT111 | Timer/counter 111 | Output |
| | | | 7 | NCE111 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| G12 | F7 | 112 | 0 | MSPI3_17 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO112 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT112 | Timer/counter 112 | Output |
| | | | 7 | NCE112 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| G11 | E5 | 113 | 0 | MSPI3_18 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO113 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT113 | Timer/counter 113 | Output |
| | | | 7 | NCE113 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| H9 | E6 | 114 | 0 | - | - | - |
| | | | 1 | MNCE3_0 | MSPI Manager 3 nCE 0 Signal | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO114 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT114 | Timer/counter 114 | Output |
| | | | 7 | NCE114 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| K12 | H7 | 115 | 0 | MSPI3_0 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO115 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT115 | Timer/counter 115 | Output |
| | | | 7 | NCE115 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| K13 | H6 | 116 | 0 | MSPI3_1 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO116 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT116 | Timer/counter 116 | Output |
| | | | 7 | NCE116 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| L11 | H5 | 117 | 0 | MSPI3_2 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO117 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT117 | Timer/counter 117 | Output |
| | | | 7 | NCE117 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| L12 | J7 | 118 | 0 | MSPI3_3 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO118 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT118 | Timer/counter 118 | Output |
| | | | 7 | NCE118 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| M11 | J6 | 119 | 0 | MSPI3_4 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO119 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT119 | Timer/counter 119 | Output |
| | | | 7 | NCE119 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| M12 | J5 | 120 | 0 | MSPI3_5 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO120 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT120 | Timer/counter 120 | Output |
| | | | 7 | NCE120 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| M13 | J4 | 121 | 0 | MSPI3_6 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO121 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT121 | Timer/counter 121 | Output |
| | | | 7 | NCE121 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| N11 | K5 | 122 | 0 | MSPI3_7 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO122 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT122 | Timer/counter 122 | Output |
| | | | 7 | NCE122 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| N12 | K4 | 123 | 0 | MSPI3_8 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO123 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT123 | Timer/counter 123 | Output |
| | | | 7 | NCE123 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| N13 | K3 | 124 | 0 | MSPI3_9 | MSPI Manager 3 Interface Signal | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO124 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT124 | Timer/counter 124 | Output |
| | | | 7 | NCE124 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| C12 | B5 | 125 | 0 | SDIF1_DAT0 | SD/SDIO/MMC Data0 pin | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO125 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT125 | Timer/counter 125 | Output |
| | | | 7 | NCE125 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| D12 | A5 | 126 | 0 | SDIF1_DAT1 | SD/SDIO/MMC Data1 pin | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO126 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT126 | Timer/counter 126 | Output |
| | | | 7 | NCE126 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| A11 | A6 | 127 | 0 | SDIF1_DAT2 | SD/SDIO/MMC Data2 pin | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO127 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT127 | Timer/counter 127 | Output |
| | | | 7 | NCE127 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| B11 | B6 | 128 | 0 | SDIF1_DAT3 | SD/SDIO/MMC Data3 pin | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO128 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT128 | Timer/counter 128 | Output |
| | | | 7 | NCE128 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| C11 | E7 | 129 | 0 | SDIF1_CLKOUT | SD/SDIO/MMC Clock to Card (CLK) | Output |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO129 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT129 | Timer/counter 129 | Output |
| | | | 7 | NCE129 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| D11 | D7 | 130 | 0 | SDIF1_DAT4 | SD/SDIO/MMC Data4 pin | I/O |
| | | | 1 | MNCE0_0 | MSPI Manager 0 nCE 0 Signal | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO130 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT130 | Timer/counter 130 | Output |
| | | | 7 | NCE130 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | I2S1_CLK | I2S1 Bit clock | Bidirectional |
| | | | 12 | - | - | - |
| A10 | C7 | 131 | 0 | SDIF1_DAT5 | SD/SDIO/MMC Data5 pin | I/O |
| | | | 1 | MNCE3_0 | MSPI Manager 3 nCE 0 Signal | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO131 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT131 | Timer/counter 131 | Output |
| | | | 7 | NCE131 | - | - |
| | | | 9 | I2S1_DATA | I2S1 Data | Bidirectional |
| | | | 10 | I2S1_SDOUT | I2S1 Data output | Output |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| B10 | B7 | 132 | 0 | SDIF1_DAT6 | SD/SDIO/MMC Data6 pin | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO132 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT132 | Timer/counter 132 | Output |
| | | | 7 | NCE132 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | I2S1_WS | I2S1 L/R clock | Bidirectional |
| | | | 12 | - | - | - |
| C10 | D8 | 133 | 0 | SDIF1_DAT7 | SD/SDIO/MMC Data7 pin | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO133 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT133 | Timer/counter 133 | Output |
| | | | 7 | NCE133 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | I2S1_SDIN | I2S1 Data input | Input |
| | | | 12 | - | - | - |
| D10 | C8 | 134 | 0 | SDIF1_CMD | SD1/SD4/MMC Command pin | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO134 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT134 | Timer/counter 134 | Output |
| | | | 7 | NCE134 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| C15 | NC | 136 | 0 | - | - | - |
| | | | 1 | DISP_D15 | Display Data 15 | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO136 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT136 | Timer/counter 136 | Output |
| | | | 7 | NCE136 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| N3 | K13 | 138 | 0 | MNCE1_1 | MSPI Manager 1 nCE 1 Signal | - |
| | | | 1 | 32KHzXT | 32kHZ from analog | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO138 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT138 | Timer/counter 138 | Output |
| | | | 7 | NCE138 | - | - |
| | | | 9 | I2S0_SDIN | I2S0 Data input | Input |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| K2 | NC | 139 | 0 | - | - | - |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO139 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT139 | Timer/counter 139 | Output |
| | | | 7 | NCE139 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | I2S0_SDIN | I2S0 Data input | Input |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| L4 | NC | 140 | 0 | - | - | - |
| | | | 1 | DISP_D16 | Display Data 16 | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO140 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT140 | Timer/counter 140 | Output |
| | | | 7 | NCE140 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| K3 | NC | 141 | 0 | - | - | - |
| | | | 1 | DISP_D17 | Display Data 17 | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO141 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT141 | Timer/counter 141 | Output |
| | | | 7 | NCE141 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| G5 | F13 | 142 | 0 | SWTRACECLK | Serial Wire Debug Trace Clock | Output |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO142 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT142 | Timer/counter 142 | Output |
| | | | 7 | NCE142 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| F6 | F14 | 143 | 0 | SWTRACE0 | Serial Wire Debug Trace Output 0 | Output |
|    |     |     | 1 | - | - | - |
|    |     |     | 2 | - | - | - |
|    |     |     | 3 | GPIO143 | General purpose I/O | I/O |
|    |     |     | 4 | - | - | - |
|    |     |     | 5 | - | - | - |
|    |     |     | 6 | CT143 | Timer/counter 143 | Output |
|    |     |     | 7 | NCE143 | - | - |
|    |     |     | 9 | - | - | - |
|    |     |     | 10 | - | - | - |
|    |     |     | 12 | - | - | - |
| E5 | E13 | 144 | 0 | SWTRACE1 | Serial Wire Debug Trace Output 1 | Output |
|    |     |     | 1 | - | - | - |
|    |     |     | 2 | - | - | - |
|    |     |     | 3 | GPIO144 | General purpose I/O | I/O |
|    |     |     | 4 | - | - | - |
|    |     |     | 5 | - | - | - |
|    |     |     | 6 | CT144 | Timer/counter 144 | Output |
|    |     |     | 7 | NCE144 | - | - |
|    |     |     | 9 | - | - | - |
|    |     |     | 10 | - | - | - |
|    |     |     | 12 | - | - | - |
| G2 | E14 | 145 | 0 | SWTRACE2 | Serial Wire Debug Trace Output 2 | Output |
|    |     |     | 1 | - | - | - |
|    |     |     | 2 | - | - | - |
|    |     |     | 3 | GPIO145 | General purpose I/O | I/O |
|    |     |     | 4 | - | - | - |
|    |     |     | 5 | - | - | - |
|    |     |     | 6 | CT145 | Timer/counter 145 | Output |
|    |     |     | 7 | NCE145 | - | - |
|    |     |     | 9 | - | - | - |
|    |     |     | 10 | - | - | - |
|    |     |     | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| F2 | D13 | 146 | 0 | SWTRACE3 | Serial Wire Debug Trace Output 3 | Output |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO146 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT146 | Timer/counter 146 | Output |
| | | | 7 | NCE146 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| E6 | C12 | 147 | 0 | SWTRACECTL | Serial Wire Debug Trace Control | Output |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO147 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT147 | Timer/counter 147 | Output |
| | | | 7 | NCE147 | - | - |
| | | | 9 | MNCE0_0 | MSPI Manager 0 nCE 0 Signal | - |
| | | | 10 | MNCE3_0 | MSPI Manager 3 nCE 0 Signal | - |
| | | | 12 | - | - | - |
| E7 | NC | 148 | 0 | - | - | - |
| | | | 1 | DISP_D5 | Display Data 5 | Output |
| | | | 2 | DBIB_D8 | Display DBIB Data 8 | Output |
| | | | 3 | GPIO148 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT148 | Timer/counter 148 | Output |
| | | | 7 | NCE148 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| F7 | NC | 149 | 0 | - | - | - |
| | | | 1 | DISP_D6 | Display Data 6 | Output |
| | | | 2 | DBIB_D9 | Display DBIB Data 9 | Output |
| | | | 3 | GPIO149 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT149 | Timer/counter 149 | Output |
| | | | 7 | NCE149 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| G4 | NC | 150 | 0 | - | - | - |
| | | | 1 | DISP_D7 | Display Data 7 | Output |
| | | | 2 | DBIB_D10 | Display DBIB Data 10 | Output |
| | | | 3 | GPIO150 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT150 | Timer/counter 150 | Output |
| | | | 7 | NCE150 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| G3 | NC | 151 | 0 | - | - | - |
| | | | 1 | DISP_D8 | Display Data 8 | Output |
| | | | 2 | DBIB_D11 | Display DBIB Data 11 | Output |
| | | | 3 | GPIO151 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT151 | Timer/counter 151 | Output |
| | | | 7 | NCE151 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| F3 | NC | 152 | 0 | - | - | - |
| | | | 1 | DISP_D9 | Display Data 9 | Output |
| | | | 2 | DBIB_D12 | Display DBIB Data 12 | Output |
| | | | 3 | GPIO152 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT152 | Timer/counter 152 | Output |
| | | | 7 | NCE152 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| F8 | NC | 153 | 0 | - | - | - |
| | | | 1 | DISP_D10 | Display Data 10 | Output |
| | | | 2 | DBIB_D13 | Display DBIB Data 13 | Output |
| | | | 3 | GPIO153 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT153 | Timer/counter 153 | Output |
| | | | 7 | NCE153 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| F5 | NC | 154 | 0 | - | - | - |
| | | | 1 | DISP_D11 | Display Data 11 | Output |
| | | | 2 | DBIB_D14 | Display DBIB Data 14 | Output |
| | | | 3 | GPIO154 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT154 | Timer/counter 154 | Output |
| | | | 7 | NCE154 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| F4 | NC | 155 | 0 | - | - | - |
| | | | 1 | DISP_D12 | Display Data 12 | Output |
| | | | 2 | DBIB_D15 | Display DBIB Data 15 | Output |
| | | | 3 | GPIO155 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT155 | Timer/counter 155 | Output |
| | | | 7 | NCE155 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| M8 | H10 | 156 | 0 | SDIF0_DAT4 | SD/SDIO/MMC Data4 pin | I/O |
| | | | 1 | MNCE2_1 | MSPI Manager 2 nCE 1 Signal | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO156 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT156 | Timer/counter 156 | Output |
| | | | 7 | NCE156 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| N9 | J9 | 157 | 0 | SDIF0_DAT5 | SD/SDIO/MMC Data5 pin | I/O |
| | | | 1 | MNCE2_0 | MSPI Manager 2 nCE 0 Signal | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO157 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT157 | Timer/counter 157 | Output |
| | | | 7 | NCE157 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| M9 | K9 | 158 | 0 | SDIF0_DAT6 | SD/SDIO/MMC Data6 pin | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO158 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT158 | Timer/counter 158 | Output |
| | | | 7 | NCE158 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| J9 | L9 | 159 | 0 | SDIF0_DAT7 | SD/SDIO/MMC Data7 pin | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO159 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT159 | Timer/counter 159 | Output |
| | | | 7 | NCE159 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| N10 | G9 | 160 | 0 | SDIF0_CMD | SD1/SD4/MMC Command pin | I/O |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO160 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT160 | Timer/counter 160 | Output |
| | | | 7 | NCE160 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| T2 | F1 | 165 | 0 | - | - | - |
| | | | 1 | - | - | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO165 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT165 | Timer/counter 165 | Output |
| | | | 7 | NCE165 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| H2 | NC | 186 | 0 | - | - | - |
| | | | 1 | DISP_SD | Display RGB Shutdown | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO186 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT186 | Timer/counter 186 | Output |
| | | | 7 | NCE186 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| H3 | NC | 187 | 0 | - | - | - |
| | | | 1 | DISP_DE | Display RGB Data Enable | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO187 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT187 | Timer/counter 187 | Output |
| | | | 7 | NCE187 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| H4 | NC | 188 | 0 | - | - | - |
| | | | 1 | DISP_CM | Display RGB Color Mode | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO188 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT188 | Timer/counter 188 | Output |
| | | | 7 | NCE188 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| L6 | NC | 189 | 0 | - | - | - |
| | | | 1 | DISP_PCLK | Display RGB Pixel Clock | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO189 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT189 | Timer/counter 189 | Output |
| | | | 7 | NCE189 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| L13 | NC | 195 | 0 | - | - | - |
| | | | 1 | MNCE3_1 | MSPI Manager 3 nCE 1 Signal | - |
| | | | 2 | MNCE3_0 | MSPI Manager 3 nCE 0 Signal | - |
| | | | 3 | GPIO195 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT195 | Timer/counter 195 | Output |
| | | | 7 | NCE195 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| E14 | C3 | 199 | 0 | - | - | - |
| | | | 1 | MNCE0_0 | MSPI Manager 0 nCE 0 Signal | - |
| | | | 2 | - | - | - |
| | | | 3 | GPIO199 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT199 | Timer/counter 199 | Output |
| | | | 7 | NCE199 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| F13 | E2 | 200 | 0 | - | - | - |
| | | | 1 | MNCE0_1 | MSPI Manager 0 nCE 1 Signal | - |
| | | | 2 | MNCE0_0 | MSPI Manager 0 nCE 0 Signal | - |
| | | | 3 | GPIO200 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT200 | Timer/counter 200 | Output |
| | | | 7 | NCE200 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| F12 | NC | 208 | 0 | - | - | - |
| | | | 1 | 32KHzXT | 32kHZ from analog | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO208 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT208 | Timer/counter 208 | Output |
| | | | 7 | NCE208 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| H5 | NC | 209 | 0 | DISP_ENB | Display enable signal for pixel memory (JDI) | Output |
| | | | 1 | DISP_D1 | Display Data 1 | Output |
| | | | 2 | DBIB_CSX | Display DBIB Chip Select | Output |
| | | | 3 | GPIO209 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT209 | Timer/counter 209 | Output |
| | | | 7 | NCE209 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| H6 | NC | 210 | 0 | DISP_XRST | Display Reset signal for Horizontal and Vertical drivers (JDI) | Output |
| | | | 1 | DISP_D4 | Display Data 4 | Output |
| | | | 2 | DBIB_DCX | Display DBIB Data/Command Selection | Output |
| | | | 3 | GPIO210 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT210 | Timer/counter 210 | Output |
| | | | 7 | NCE210 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| G6 | NC | 211 | 0 | DISP_R1 | Display RGB Red Signal for even Pixels | Output |
| | | | 1 | DISP_D23 | Display Data 23 | Output |
| | | | 2 | DBIB_WRX | Display DBIB Write Enable | Output |
| | | | 3 | GPIO211 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT211 | Timer/counter 211 | Output |
| | | | 7 | NCE211 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| G8 | NC | 212 | 0 | DISP_R2 | Display Even red data (JDI) | Output |
| | | | 1 | DISP_D22 | Display Data 22 | Output |
| | | | 2 | DBIB_RDX | Display DBIB Read Enable | Output |
| | | | 3 | GPIO212 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT212 | Timer/counter 212 | Output |
| | | | 7 | NCE212 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| H7 | NC | 213 | 0 | DISP_G1 | Display RGB Green Signal for even Pixels | Output |
| | | | 1 | DISP_D21 | Display Data 21 | Output |
| | | | 2 | DBIB_D0 | Display DBIB Data 0 | Output |
| | | | 3 | GPIO213 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT213 | Timer/counter 213 | Output |
| | | | 7 | NCE213 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| J8 | NC | 214 | 0 | DISP_G2 | Display Even green data (JDI) | Output |
| | | | 1 | DISP_D20 | Display Data 20 | Output |
| | | | 2 | DBIB_D1 | Display DBIB Data 1 | Output |
| | | | 3 | GPIO214 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT214 | Timer/counter 214 | Output |
| | | | 7 | NCE214 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| G7 | NC | 215 | 0 | DISP_B1 | Display RGB Blue Signal for even Pixels | Output |
| | | | 1 | DISP_D19 | Display Data 19 | Output |
| | | | 2 | DBIB_D2 | Display DBIB Data 2 | Output |
| | | | 3 | GPIO215 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT215 | Timer/counter 215 | Output |
| | | | 7 | NCE215 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| J5 | NC | 216 | 0 | DISP_B2 | Display Even blue data (JDI) | Output |
| | | | 1 | DISP_D18 | Display Data 18 | Output |
| | | | 2 | DBIB_D3 | Display DBIB Data 3 | Output |
| | | | 3 | GPIO216 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT216 | Timer/counter 216 | Output |
| | | | 7 | NCE216 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| K5 | NC | 217 | 0 | DISP_HST | Display Horizontal Start signal (JDI) | Output |
| | | | 1 | DISP_HS | Display RGB HSYNC | Output |
| | | | 2 | DBIB_D4 | Display DBIB Data 4 | Output |
| | | | 3 | GPIO217 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT217 | Timer/counter 217 | Output |
| | | | 7 | NCE217 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---------|---------|--------------|----------------|-------------------|-------------|----------|
| K6 | NC | 218 | 0 | DISP_VST | Display Veritical Start signal (JDI) | Output |
| | | | 1 | DISP_VS | Display RGB VSYNC | Output |
| | | | 2 | DBIB_D5 | Display DBIB Data 5 | Output |
| | | | 3 | GPIO218 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT218 | Timer/counter 218 | Output |
| | | | 7 | NCE218 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| K7 | NC | 219 | 0 | DISP_HCK | Display Horizontal Shift clock (JDI) | Output |
| | | | 1 | DISP_D3 | Display Data 3 | Output |
| | | | 2 | DBIB_D6 | Display DBIB Data 6 | Output |
| | | | 3 | GPIO219 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT219 | Timer/counter 219 | Output |
| | | | 7 | NCE219 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| J6 | NC | 220 | 0 | DISP_VCK | Display Vertical Shift clock (JDI) | Output |
| | | | 1 | DISP_D0 | Display Data 0 | Output |
| | | | 2 | DBIB_D7 | Display DBIB Data 7 | Output |
| | | | 3 | GPIO220 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT220 | Timer/counter 220 | Output |
| | | | 7 | NCE220 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Table 24: Apollo510 SoC Pin List and Function Table**

| BGA PIN | CSP Pin | GPIO Pad No. | Fcn Select No. | Pad Function Name | Description | Pin Type |
|---|---|---|---|---|---|---|
| H8 | NC | 221 | 0 | - | - | - |
| | | | 1 | DISP_D2 | Display Data 2 | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO221 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT221 | Timer/counter 221 | Output |
| | | | 7 | NCE221 | - | - |
| | | | 9 | - | - | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| J7 | NC | 222 | 0 | - | - | - |
| | | | 1 | DISP_D13 | Display Data 13 | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO222 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT222 | Timer/counter 222 | Output |
| | | | 7 | NCE222 | - | - |
| | | | 9 | MNCE0_0 | MSPI Manager 0 nCE 0 Signal | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |
| K8 | NC | 223 | 0 | - | - | - |
| | | | 1 | DISP_D14 | Display Data 14 | Output |
| | | | 2 | - | - | - |
| | | | 3 | GPIO223 | General purpose I/O | I/O |
| | | | 4 | - | - | - |
| | | | 5 | - | - | - |
| | | | 6 | CT223 | Timer/counter 223 | Output |
| | | | 7 | NCE223 | - | - |
| | | | 9 | MNCE3_0 | MSPI Manager 3 nCE 0 Signal | - |
| | | | 10 | - | - | - |
| | | | 12 | - | - | - |

**Blank Page**

# 29. Package Mechanical Information[1]

## 29.1 Apollo510 BGA Package
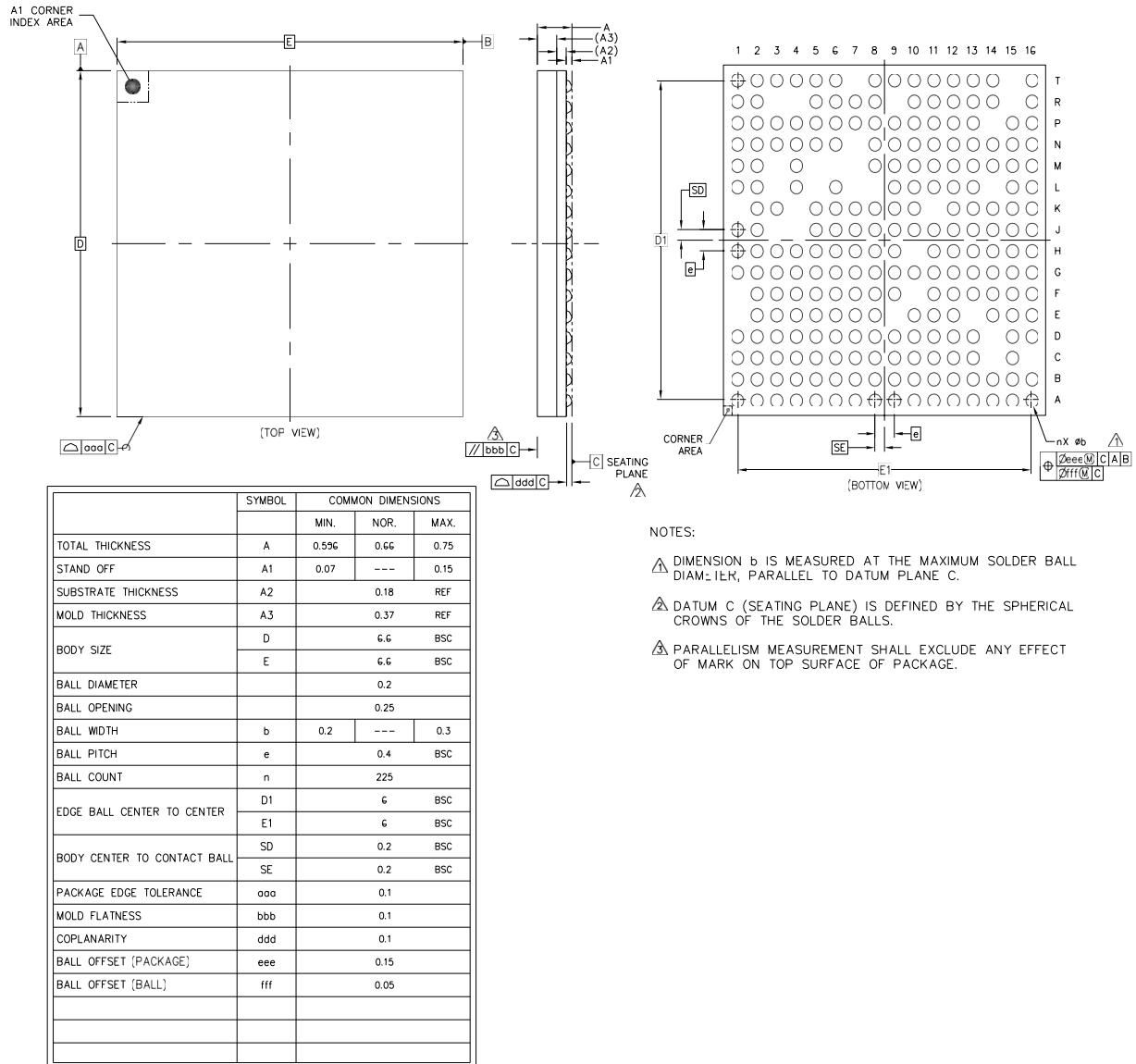


**Figure 51. Apollo510 BGA Package Drawing**

| SYMBOL | | COMMON DIMENSIONS | | |
|---|---|---|---|---|
| | | MIN. | NOR. | MAX. |
| TOTAL THICKNESS | A | 0.596 | 0.66 | 0.75 |
| STAND OFF | A1 | 0.07 | --- | 0.15 |
| SUBSTRATE THICKNESS | A2 | | 0.18 | REF |
| MOLD THICKNESS | A3 | | 0.37 | REF |
| BODY SIZE | D | | 6.6 | BSC |
| | E | | 6.6 | BSC |
| BALL DIAMETER | | | 0.2 | |
| BALL OPENING | | | 0.25 | |
| BALL WIDTH | b | 0.2 | --- | 0.3 |
| BALL PITCH | e | | 0.4 | BSC |
| BALL COUNT | n | | 225 | |
| EDGE BALL CENTER TO CENTER | D1 | | 6 | BSC |
| | E1 | | 6 | BSC |
| BODY CENTER TO CONTACT BALL | SD | | 0.2 | BSC |
| | SE | | 0.2 | BSC |
| PACKAGE EDGE TOLERANCE | aaa | | 0.1 | |
| MOLD FLATNESS | bbb | | 0.1 | |
| COPLANARITY | ddd | | 0.1 | |
| BALL OFFSET (PACKAGE) | eee | | 0.15 | |
| BALL OFFSET (BALL) | fff | | 0.05 | |

NOTES:

1. DIMENSION b IS MEASURED AT THE MAXIMUM SOLDER BALL DIAMETER, PARALLEL TO DATUM PLANE C.

2. DATUM C (SEATING PLANE) IS DEFINED BY THE SPHERICAL CROWNS OF THE SOLDER BALLS.

3. PARALLELISM MEASUREMENT SHALL EXCLUDE ANY EFFECT OF MARK ON TOP SURFACE OF PACKAGE.

---

1. All dimensions in mm unless otherwise noted.

---

### 29.1.1  Package Top Side Marking

Package type: 6.6 x 6.6 mm, 225-pin BGA

Part Number: AP510NFA-CBR

Memory Size: 4 MB
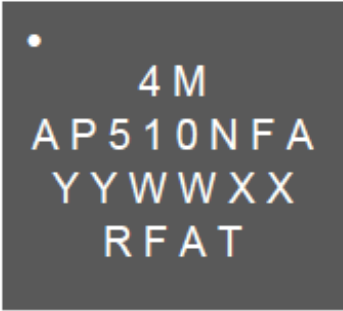


**Figure 52. Apollo510 BGA Package Top Side Marking**

**Table 25: Apollo510 BGA Marking Description**

| Line | Field | Description |
|------|-------|-------------|
| 1 | Part Description | 4M |
| 2 | Part Name | AP510NFA |
| 3 | YY = Year<br>WW = Work Week<br>XX = Sequential Lot Number | YY corresponds to the year.<br>WW is the release week of assembly lot.<br>XX is the sequential lot number assigned by the assembly house |
| 4 | R = Production Revision<br>F = Foundry Site<br>A = Assembly Site<br>T = Test Site | RFAT designates the product revision and the supply chain. |

## 29.1.2   Tape & Reel Data
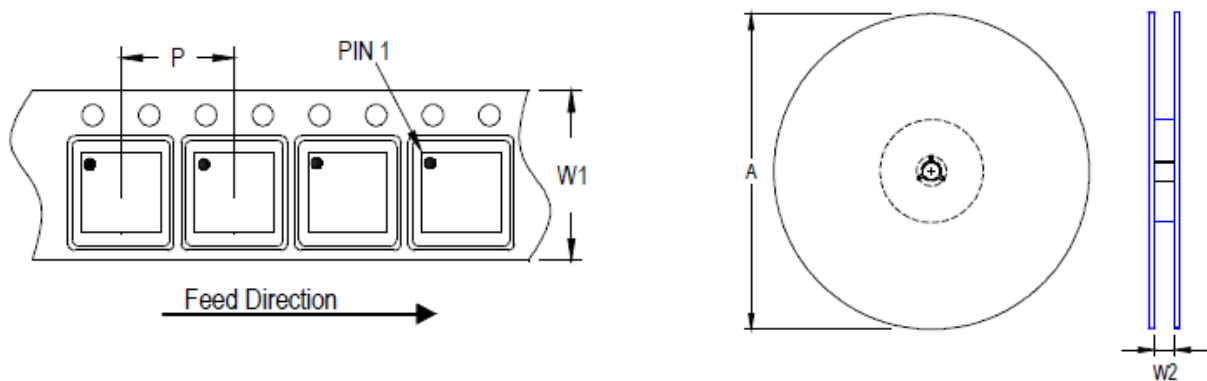


**Figure 53. Tape Feed Direction and Reel Dimension Drawing**

**Table 26: Tape & Reel Dimension Table**

| Tape Size (W1) (mm) | Pocket Pitch (P) (mm) | Reel Size (A) | | Reel Width (W2) Min/Max (mm) |
|---|---|---|---|---|
| | | (mm) | (in) | |
| 16 | 12 | 330 | 13 | 16.4/17.4 |

## 29.1.3   Tape & Reel Packaging

**Table 27: Tape & Reel Packaging Dimensions**

| Reel | | Inner Box | | | Outer Carton | | |
|---|---|---|---|---|---|---|---|
| Size (mm) | Units | Size (mm) | Reels | Units | Size (mm) | Boxes | Units |
| 330 | 4,000 | 358 x 338 x 50 | 1 | 4,000 | 375 x 284 x 362 | 5 | 20,000 |

### 29.1.4 Label Format

**Table 28: Label Location and Format**

| Step | Label Location | Label Format |
|---|---|---|
| **Reel** |  | One Lot:<br> |
| **MBB (HIC & Desiccant inside)** |  | Two lots in one reel: *For combined lot, allows at most two 4-digits of date codes to be packed in the one reel. |
| **Inner Box** |  |  |

**Table 28: Label Location and Format**

| Step | Label Location | Label Format |
|---|---|---|
| **Outer Carton** |  |  |

## 29.2 Apollo510 WLCSP Package[1]

[TOP VIEW]

[BOTTOM VIEW]

| | SYMBOL | MIN | NOM | MAX |
|---|---|---|---|---|
| TOTAL THICKNESS | A | 0.397 | 0.455 | 0.513 |
| STAND OFF | A1 | 0.125 | --- | 0.185 |
| WAFER THICKNESS | A2 | 0.25 | 0.275 | 0.3 |
| FILM THICKNESS | A3 | 0.022 | 0.025 | 0.028 |
| BODY SIZE X | D | | 4.913 | |
| BODY SIZE Y | E | | 4.676 | |
| BALL/BUMP PITCH X | SD | 0.175 | | BSC |
| BALL/BUMP PITCH Y | SE | --- | | BSC |
| EDGE BALL CENTER TO CENTER X | D1 | 4.55 | | BSC |
| EDGE BALL CENTER TO CENTER Y | E1 | 4.2 | | BSC |
| PITCH | e | 0.35 | | BSC |
| BALL DIAMETER (SIZE) | | | 0.2 | |
| BALL/BUMP WIDTH | b | 0.19 | --- | 0.25 |
| BALL/BUMP COUNT | n | | 182 | |
| PACKAGE EDGE TOLERANCE | aaa | | 0.03 | |
| WAFER FLATNESS | bbb | | 0.06 | |
| COPLANARITY | ccc | | 0.03 | |
| BALL/BUMP OFFSET (PACKAGE) | ddd | | 0.015 | |
| | | | | |
| | | | | |
| | | | | |

**Figure 54. Apollo510 WLCSP Package Drawing**

---

1. The WLCSP package of the Apollo510 SoC is sensitive to light incident on either the backside or edges of the die. Light exposure can result in increased current and erratic behavior which may include system crash or reset. To prevent exposure of the die to light, the recommendation is to apply an opaque epoxy coating (or similar) over the Apollo510 SoC WLCSP unless the MCU will be in an enclosure that blocks all light.

## 29.3 WLCSP Marking, Packaging and Labeling

### 29.3.1 Top Side Marking

Package type: 4.913 x 4.676 mm, 182-pin WLCSP (IR PASS BSC)

Part Number: AP510NFA-CCR

Memory Size: 4 MB
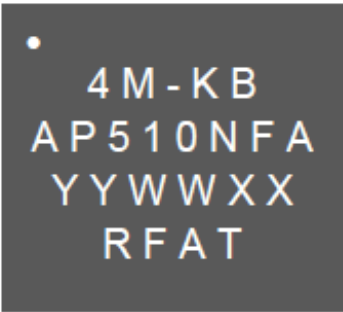


**Figure 55. Apollo510 WLCSP Top Side Marking**

**Table 29: Apollo510 WLCSP Marking Description**

| Line | Field | Description |
|------|-------|-------------|
| 1 | Part Description | 4M-KB |
| 2 | Part Name | AP510NFA |
| 3 | YY = Year<br>WW = Work Week<br>XX = Sequential Lot Number | YY corresponds to the year.<br>WW is the release week of assembly lot.<br>XX is the sequential lot number assigned by the assembly house |
| 4 | R = Production Revision<br>F = Foundry Site<br>A = Assembly Site<br>T = Test Site | RFAT designates the product revision and the supply chain. |

### 29.3.2 Tape & Reel Data
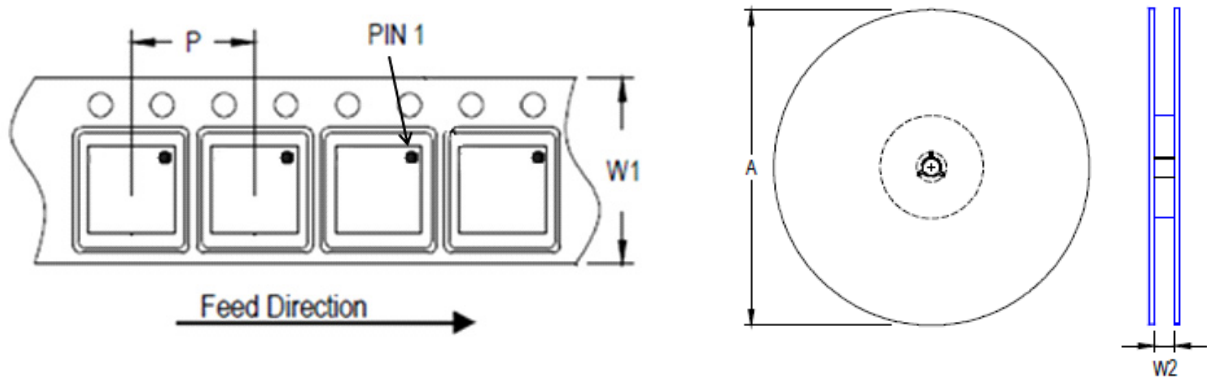


**Figure 56. Tape Feed Direction and Reel Dimension Drawing**

**Table 30: Tape & Reel Dimension Table**

| Tape Size (W1) (mm) | Pocket Pitch (P) (mm) | Reel Size (A) | | Reel Width (W2) Min/Max (mm) |
| --- | --- | --- | --- | --- |
| | | (mm) | (in) | |
| 12.0 ± 0.3 | 8.00 ± 0.10 | 330 ± 2 | 13 | 18.2 Max |

### 29.3.3 Tape & Reel Packaging

**Table 31: Tape & Reel Packaging Dimensions**

| Reel | | Inner Box | | | Outer Carton | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Size (mm) | Units | Size (mm) | Reels | Units | Size (mm) | Boxes | Units |
| 330 | 5,000 | 362 x 356 x 47 | 1 | 5,000 | 377 x 267 x 387 | 5 | 25,000 |

### 29.3.4  Label Format

**Table 32: Label Location and Format**

| Step | Label Location | Label Format |
|---|---|---|
| **Reel** |  | One Lot:  |
| **MBB (HIC & Desiccant inside)** |  | Two lots in one reel: *For combined lot, allows at most two 4-digits of date codes to be packed in the one reel. |
| **Inner Box** |  |  |

**Table 32: Label Location and Format**

| Step | Label Location | Label Format |
|---|---|---|
| **Outer Carton** |  |  |

## 29.4 Reflow Profile

Table 33 lists the reflow conditions for the lead-free package. Reference IR Reflow Profile for Moisture Sensitivity Test (J-STD-020).

Reflow times: 3 cycles

**Table 33: Reflow Condition (260 °C) for Pb-free Package**

| Profile Features | Pb-Free Assembly |
|---|---|
| Average ramp-up rate (include 217°C to Peak) | 3°C/second max. |
| Temperature maintained above 217°C | 60 to 150 seconds |
| Time within 5°C of actual peak temperature | 20 - 40 seconds |
| Peak temperature (minimum) | 260 +0/-5°C |
| Ramp-down rate | 6°C /second max. |
| Time 25°C to peak temperature | 8 minutes max. |

Figure 57 illustrates the temperature profile for reflow soldering requirements.
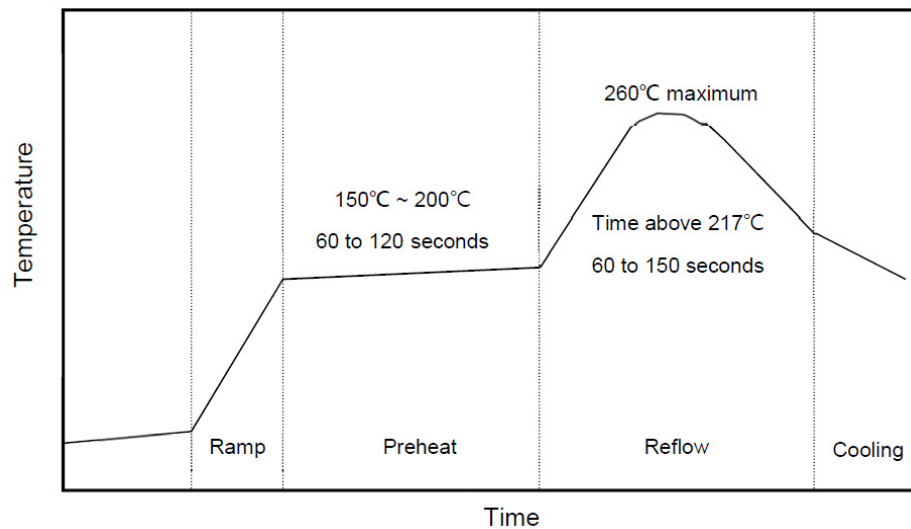


**Figure 57. Reflow Profile**

# 30. Electrical Characteristics

## IMPORTANT NOTICE

Specifications and other information in this Apollo510 SoC Datasheet are preliminary and subject to change.

Timing characteristics for the various communication/data interfaces have been determined by either measurements on chip or through simulations. It is noted in each timing set how its characteristics were derived. Characteristics derived through simulation are preliminary and subject to change, and cover the worst-case fastest and slowest corners.

Unless stated otherwise in a timing table, the following conditions are used for timing specifications throughout.

- **Output load: 30 pF**
- **GPIO drive strength DS[1:0]: 0x2**
- **GPIO pull: Disabled**
- **GPIO slew rate control: 0**
- **Nominal voltage: 1.8 V**
- **All output timing and rise/fall times are to 0.7 x VDDHn / 0.3 x VDDHn, respectively.**

**Contact Ambiq sales with questions about specifications.**

## 30.1 Absolute Maximum Ratings

The absolute maximum ratings are the limits to which the device can be subjected without permanently damaging the device and are stress ratings only. Device reliability may be adversely affected by exposure to absolute-maximum ratings for extended periods. Functional operation of the device at the absolute maximum ratings or any other conditions beyond the recommended operating conditions is not implied.

**Table 34: Absolute Maximum Ratings**

| Symbol | Parameter | Test Conditions | Min | Max | Unit |
|---|---|---|---|---|---|
| VDDP | SIMO/LDO Buck Supply voltage | | -0.2 | 3.63 | V |
| VDDA | Analog Supply voltage | | | | |
| VDDH | High voltage domain supply voltage | | | | |
| VDDH1 | High voltage domain 1 IO Supply voltage | | -0.2 | 3.63 | V |
| VDDH2 | High voltage domain 2 IO Supply voltage | | -0.2 | 3.63 | V |
| VDDH3 | High voltage domain 3 IO Supply voltage | | -0.2 | 3.63 | V |
| VDDH4 | High voltage domain 4 IO Supply voltage | | -0.2 | 3.63 | V |
| VDDH5[1] | High voltage domain 5 IO Supply voltage | | -0.2 | 3.63 | V |
| VDD18 | MIPI, DSI, DISPPLL Supply voltage | | -0.2 | 1.98 | V |
| VDDAUDA | Analog Audio Supply voltage | | -0.2 | 1.98 | V |
| VDDUSB33 | USB Analog 3.3 V Supply voltage | | -0.2 | 3.63 | V |
| VDDUSB0P9 | USB Analog 0.9 V Supply voltage | | -0.2 | 0.99 | V |
| $V_{IO}$ | Voltage on all input and output pins | | -0.30 | VDDHn[2] + 0.30 | V |
| VDDX$_{FSLEW}$ | Falling slew rate for VDDP, VDDA, VDDH and VDDHn power supplies | | - | 2 | kV/s |
| T$_{OPEC}$ | Extended commercial operating temperature range | Specific to extended commercial temperature range SKUs | -20 | 70 | ºC |
| T$_{OPI}$ | Industrial operating temperature range | Specific to industrial temperature range SKUs | -40 | 85 | ºC |
| T$_{REFLOW}$ | Reflow temperature | Reflow Profile per JEDEC J-STD-020D.1 | - | 260 | ºC |
| I$_{LU}$ | Latch-up current | EIA/JESD78, 25°C, ±100 mA trigger current and Over voltage at 1.5 Vmax | - | 100 | mA |
| V$_{ESDHBM}$ | ESD Human Body Model (HBM) | JS-001-2017 | - | 2000 | V |
| V$_{ESDCDM}$ | ESD Charged Device Model (CDM) | JS-002-2014 | - | 250 | V |

1. VDDH5 is not used/required on the CSP package of the Apollo510 SoC.
2. The designation "n" corresponds to the voltage source for the pin, e.g., VDDH1. This specification is for all VDDH supplies powering GPIO.

## 30.2 Recommended Operating Conditions

### 30.2.1 External Voltage Supplies

**Table 35: External Voltage Supplies**

| Supply | Description | Source | Min | Typ | Max | Units | Notes |
|---|---|---|---|---|---|---|---|
| VDDA | Analog Supply | Battery / External Regulator | 1.71 | 1.8 - 2.0 | 2.20 | V | VDDA, VDDP and VDDH must be connected to the same supply. |
| VDDP | SIMO/LDO Buck Supply | | | | | | |
| VDDH | High voltage domain supply voltage | | | | | | |
| VDDH1 | Secondary I/O Supply (Primary Interface: HWTRACE) | Battery / External Regulator | 1.14 | 1.8 - 2.0 | 2.20 | V | |
| VDDH2 | Secondary I/O Supply | Battery / External Regulator | 1.71 | 1.8 - 2.0 | 3.63 | V | |
| VDDH3 | Secondary I/O Supply (Primary Interface: MSPI0) | Battery / External Regulator | 1.14 | 1.8 - 2.0 | 2.20 | V | |
| VDDH4 | Secondary I/O Supply(Primary Interface: MSPI3) | Battery / External Regulator | 1.14 | 1.8 - 2.0 | 2.20 | V | |
| VDDH5[1] | Secondary I/O Supply (Primary Interface: MiP Display) | Battery / External Regulator | 1.14 | 1.8 - 2.0 | 3.63 | V | |
| VDDAUDA | Analog Audio and XTALHS supply | External LDO, Low quiescent current, low noise preferred | 1.62 | 1.75 - 1.85 | 1.98 | V | Quiescent current: < 1 µA PSRR: > 40 dB in the range 1 kHz to 10 kHz |
| VDD18 | MIPI DPHY LP LDO and transceivers | Battery / External Regulator | 1.62 | 1.8 | 1.98 | V | Noise/ripple: ± 2% (72mVpk-pk) Frequency range: 10 MHz - 3 GHz Leakage current: 2-3 µA (typ) |
| VDDUSB33 | USB Analog 3.3 V Supply | Battery / External Regulator | 3.0 | 3.3 | 3.63 | V | |
| VDDUSB0P9 | USB Analog 0.9 V Supply | External Regulator | 0.84 | 0.9 | 0.99 | V | Noise/ripple: < 3% (pk-pk). |

1. VDDH5 is not used/required on the CSP package of the Apollo510 SoC.

### 30.2.2   Recommended External Components for Voltage Supplies

### 30.2.2.1  Components for External Voltage Supplies

**Table 36: Recommended Bypass Capacitors for External Supplies**

| External Supply | Bypass Capacitor |
|---|---|
| VDDH, VDDH1, VDDH2, VDDH5[1] | 2.2 µF to ground |
| VDDH3 | 4.7 µF to ground |
| VDDH4 | 4.7 µF to ground |
| VDDA | 1 µF to ground |
| VDDP | 10 µF to ground (SIMO Buck converter output capacitance (4) (VDDC, VDDC_LV, VDDF, VDDS)) |
| VDDAUDA | 2.2 µF to ground (Typ); follow recommendations of LDO supplier. See notes below. |
| VDDUSB33 | 2.2 µF to ground |
| VDDUSB0P9 | 2.2 µF to ground |
| VDD18 | 2.2 µF to ground |

1. For the CSP package of the Apollo510 SoC, VDDH5 is not used and therefore the bypass capacitor is not needed.

Additional Notes:

1.  For VDDH3 and VDDH4, it might be possible to use a bypass capacitor smaller than 4.7 µF (possibly 2.2 µF) depending on the speed of the external LDO and how close the LDO is placed to the Apollo510 device and external PSRAM power pins, and the interface speed being used. It is also recommended that the same size bypass capacitor be used for the I/O supply of the external (PSRAM) device.

2.  For VDDP 10 µF: Recommend 0402 package, 10 V. No more than 40% reduction in capacitance (6.0 µF) with a DC bias of 1.8 V. Suitable component: CL05A106MP8NUB8.

3.  Recommend use of 5 V or greater capacitors for 1.9 V rails

4.  Recommend use of 10 V capacitors for 3.3 V rails

5.  Do not float any supply input.  If a supply is not used, it should be tied to a supply within its operating range.

6.  Suitable standalone small form factor LDOs:
    - Microchip MCP1811A in 1.0 x 1.0 x 0.50 mm UDFN package
    - TI TPS7A02 in 1.0 x 1.0 x 0.40 mm X2SON package

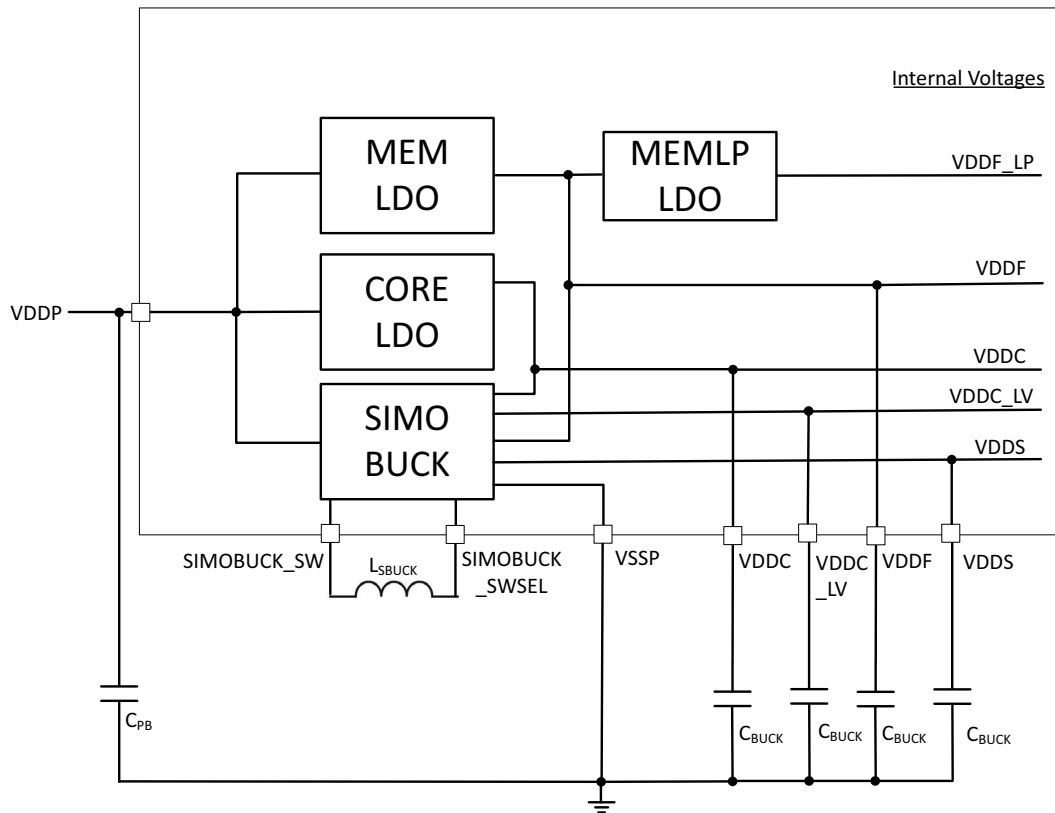### 30.2.2.2  Components for SIMO Buck Converter



**Figure 58. External Components for SIMO Buck**

**Table 37: SIMO Buck Converter External Components**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|--------|-----------|-----------------|-----|-----|-----|------|
| $L_{SBUCK}$ | SIMO Buck converter inductance ($V_{SIMO}$) | | - | 2.2 | - | µH |
| $C_{BUCK}$ | SIMO Buck converter output capacitance (4) (VDDC, VDDC_LV, VDDF, VDDS) | | - | 4.7/10 | - | µF |

NOTES:

1) For operation across the full voltage and temperature range, the SIMO Buck Inductor ($L_{SBUCK}$ connected between SIMOBUCK_SW and SIMOBUCK_SWSEL) with the following characteristics is required:

▪ 2.2 µH
▪ Saturation current >= 1 A
▪ Maximum DC resistance < 550 mΩ
▪ Operating frequency range > 20 MHz
▪ Recommended[1] parts:
  - Murata DFE201210U-2R2M=P2 (0805 size)
  - Taiyo Yuden LSCND1608HKT2R2MF (0603 size)
▪ Other options (all 2.2 µH):

**Table 38: SIMO Buck Converter Inductor Options**

| Part Number | Manufacturer | DCR (mOhm) | Current Rating (Amps) | Saturation Current (Amps) | Footprint | L x W x H (in) |
|---|---|---|---|---|---|---|
| DFE201210U-2R2M=P2 | Murata | 228 | 1.2 | 2 | 0805 | 0.079 x 0.047 x 0.039 |
| LSCND1608HKT2R2MF | Taiyo Yuden | 250 | 1.4 | 1.3 | 0603 | 0.063 x 0.031 x 0.031 |
| DFE252012F-2R2M=P2 | Murata | 82 | 2.3 | 3.3 | 1008 | 0.098 x 0.079 x 0.047 |
| 74479276222C | Wurth Elektronik | 135 | 1.6 | 2.5 | 0806 | 0.079 x 0.063 x 0.039 |
| CIGT201610EH2R2MNE | Samsung | 87 | 2.5 | 2.9 | 0806 | 0.079 x 0.063 x 0.039 |
| DFE21CCN2R2MELL | Murata | 138 | 1.8 | 2.1 | 0805 | 0.079 x 0.047 x 0.031 |
| AOTA-B201610S2R2MT | Abracon | 74 | 2.0 | 2.6 | 0806 | 0.079 x 0.063 x 0.039 |

2) For $C_{BUCK}$ the following specifications should be followed
▪ Smaller package option: 4.7 µF, ±20%,6.3 V, X5R, -55°C ~ 85°C, ceramic
▪ 0201 package to be placed as close to the MCU pins as possible
▪ No more than 20% reduction in capacitance with a DC bias of 0.9 V
▪ Recommended parts:
  - Murata GRM035R60J475ME15 or equivalent

Larger package option for up to 2% reduction in total power over smaller package:
▪ 10 µF, ±20%, 10 V, X5R, -55°C ~ 85°C, ceramic
▪ 0402 package to be placed as close to the MCU pins as possible
▪ No more than 20% reduction in capacitance with a DC bias of 0.9 V
▪ Recommended parts:
  - Murata GRT155R61A106ME13J or equivalent

### 30.2.2.3  Other External Components

▪ LPADC_VREF

---

1. Inductors with higher Q rating (Q > 20 between 2 MHz and 10 MHz) are recommended. Inductors with lower Q may be used, but will result in power consumption increase of a few percent.

- 100 nF capacitor from the LPADC_VREF pin to ground
▪ LPMICBIAS
  - LPMICBIAS can source up to 400 µA at 1.3 V, with a minimum VDDAUDA supply of 1.62 V.
  - 2.2 µF capacitor from the LPMICBIAS pin to ground. A smaller value is acceptable if current is less than 400 µA.

### 30.2.3   Power Sequencing

**Table 39: Power Supply Sequencing**

| External Supply[1] | Conditions/Notes |
|---|---|
| VDDP/VDDH/VDDA[2] | 1. Power up concurrently from same source to the same voltage (i.e., VDDP = VDDH = VDDA).<br>2. Should generally be supplied at nearly the same time as other rails.<br>3. Some skew is acceptable. |
| VDDH1-VDDH5[2, 3] | 1. May be powered up before, or at the same time as, VDDP/VDDH/VDDA are powered[4].<br>2. May be a separate supply from VDDP/VDDH/VDDA, but it may be tied to VDDP/VDDH/VDDA if not requiring a different voltage. |
| VDDAUDA[5] | 1. Must be powered before or at the same time as VDDP/VDDH/VDDA.<br>2. If not using audio, then should be tied to VDDP/VDDH/VDDA if they are within the specified VDDAUDA range.<br>3. Must be a very clean supply when used to power the Low Power Analog Audio Interface.<br>4. Primarily needed for internal clocking circuitry as well as the AUDADC. |
| VDD18 | 1. Should be tied to ground if not using MIPI DSI interface[6].<br>2. If using MIPI, then may be powered at the same time as VDDP/VDDH/VDDA but not before.<br>3. Preferably only enabled/powered up when the display is being used. |
| VDDUSB33/<br>VDDUSB0P9 | 1. May be powered before, at the same time as, or after VDDP/VDDH/VDDA.<br>2. For different USB power scenarios, see USB chapter.<br>3. Only required when using USB (i.e., when USB cable is plugged in).<br>4. Powering both supplies at the same time from VBUS source is recommended, with 0.9 V generated by LDO from the 3.3 V.[7] |

1. Recommended termination of unused interface:
   - USB data pads (USB0PP and USB0PN) left open
   - USB PHY power rails VDDUSB33 and VDDUSB0P9 connected to ground
   - DSI TX data and clock pads left open
2. Falling slew rate for these supplies cannot exceed 2 kV/s.
3. VDDH5 is not used/required on the CSP package of the Apollo510 SoC
4. Grounding any of these supplies while VDDP/VDDH/VDDA are powered will result in excess current draw and can have long term reliability implications. It is acceptable to have one VDDHn rail ramp up *after* VDDP/VDDH/VDDA as long as that rail ramps up while the CPU is still in reset (nRST pin is low).
5. To reduce power consumption and supply noise, VDDAUDA should be tied to a supply within the specified VDDAUDA operating range when not using the Low Power Analog Audio Interface which is powered by the VDDAUDA supply. In this specific scenario, the VDDAUDA supply does not need to meet the Audio Interface's required noise/PSRR conditions.
6. Note that VDD18 leakage current is 2-3 µA (typ).
7. Although it is recommended to power-up VDDUSB33 and VDDUSB0P9 together only when USB is to be used, it is possible to keep VDDUSB33 powered at all times. In which case it is required that a 2 MΩ pull-down resistor be included on each of the USB data lines to prevent leakage current. Note that in all cases, to minimize leakage current, the USB internal power rail must be enabled by software before external power is applied to VDDUSB0P9.

## 30.3  Current Consumption

**Table 40: Current Consumption in Active Mode and Sleep Modes**

| Symbol | Parameter | Test Conditions[1,2] | VDD (V) | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|---|
| $I_{RUNLPFB}$ | Coremark run current in low power mode | Executed from internal MRAM, cache enabled, buck enabled, core clock/HFRC=96 MHz | 1.8 | - | 35.3 | - | µW/MHz |
| $I_{RUNHPFB}$ | Coremark run current in high performance mode | Executed from internal MRAM, cache enabled, buck enabled, core clock/HFRC=250 MHz | 1.8 | - | 46.8 | - | µW/MHz |
| $I_{SS1}$ | System Sleep mode 1 current | WFI instruction with SLEEP=1, all core & peripheral clocks gated, HFRC on, XTAL off, SIMO buck enabled, NVM powered down, no SRAM bank enabled/retained, cache retained, 160 kB TCM enabled/retained | 1.8 | - | 750 | - | µW |
| $I_{SDS2-160RET}$ | System Deep Sleep mode 2 current - 160 kB SRAM retained | WFI instruction with SLEEPDEEP=1, LFRC on (HFRC and XTAL off), SIMO buck enabled, NVM powered down, cache off (no retention), 160 kB TCM retained | 1.8 | - | 23.8 | - | µW |
| $I_{SDS2-768RET}$ | System Deep Sleep mode 2 current - 768 kB SRAM retained | WFI instruction with SLEEPDEEP=1, LFRC on (HFRC and XTAL off), SIMO buck enabled, NVM powered down, cache off (no retention), 768 kB TCM retained | 1.8 | - | 36.0 | - | µW |
| $I_{SDS2-3840RET}$ | System Deep Sleep mode 2 current - 3840 kB SRAM retained | WFI instruction with SLEEPDEEP=1, LFRC on (HFRC and XTAL off), SIMO buck enabled, NVM powered down, cache off (no retention), 3840 kB TCM retained | 1.8 | - | 57.0 | - | µW |
| $I_{SDS3}$ | System Deep Sleep mode 3 current | WFI instruction with SLEEPDEEP=1, LFRC on (HFRC and XTAL off), SIMO buck enabled, NVM powered down, all SRAM/TCM/cache off (no retention). | 1.8 | - | 23.1 | - | µW |

[1] All values measured at 25°C.

[2] All I/O power domains and peripherals quiesced (powered off).

## 30.4 Power-On Reset (POR) and Brown-Out Detector (BOD)

**Table 41: Power-On Reset (POR) and Brown-Out Detector (BOD)**

| Symbol | Parameter | Min | Typ | Max | Unit |
|--------|-----------|-----|-----|-----|------|
| $V_{POR\_RISING}$ | POR rising threshold voltage | - | 1.635 - 1.675 | - | V |
| $V_{BODL\_FALLING}$ | Brownout detection low falling threshold voltage | - | 1.63 - 1.70 | - | V |

## 30.5   Resets

**Table 42: Resets**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|--------|-----------|-----------------|-----|-----|-----|------|
| $PU_{RST}$ | Internal nRST pull-up resistor | | - | 24 | - | kΩ |

## 30.6 General Purpose Input/Output (GPIO)

**Table 43: General Purpose Input/Output (GPIO)**

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| **Standard GPIOs[1]** | | | | | | |
| $V_{OH}$ | High-level output voltage | | $0.8 \times V_{DDHn}$[2] | - | - | V |
| $V_{OL}$ | Low-level output voltage | | - | - | $0.2 \times V_{DDHn}$ | V |
| $V_{IH}$ | Positive going input threshold voltage | | $0.7 \times V_{DDHn}$ | - | - | V |
| $V_{IL}$ | Negative going input threshold voltage | | - | - | $0.3 \times V_{DDHn}$ | V |
| $C_{GPI}$ | Input capacitance | | - | 5 | - | pF |
| $I_{IN}$ | Input pin leakage current | | - | 1 | 20 | nA |
| $I_{INOD}$ | Open drain output leakage current | | - | 1 | 20 | nA |
| $R_{PD50K}$ | 50 kΩ pull-down resistance, PULLCFG = 1 | PULLCFG = 1 | 42.38 | 51.01 | 59.43 | kΩ |
| $R_{PU15K}$ | 1.5 kΩ pull-up resistance, PULLCFG = 2 | PULLCFG = 2 | 1.32 | 1.61 | 1.89 | kΩ |
| $R_{PU6K}$ | 6 kΩ pull-up resistance, PULLCFG = 3 | PULLCFG = 3 | 5.17 | 6.22 | 7.24 | kΩ |
| $R_{PU12K}$ | 12 kΩ pull-up resistance, PULLCFG = 4 | PULLCFG = 4 | 10.37 | 12.5 | 14.55 | kΩ |
| $R_{PU24K}$ | 24 kΩ pull-up resistance, PULLCFG = 5 | PULLCFG = 5 | 20.6 | 24.85 | 28.94 | kΩ |
| $R_{PU50K}$ | 50 kΩ pull-up resistance, PULLCFG = 6 | PULLCFG = 6 | 42.23 | 50.64 | 58.82 | kΩ |
| $R_{PU100K}$ | 100 kΩ pull-up resistance, PULLCFG = 7 | PULLCFG = 7 | 84.03 | 100.5 | 116.6 | kΩ |
| $I_{SRC\_0p1DS}$ | Output source current, 0.1x drive strength | VDDHn = 1.8 V | 3.132 | 3.791 | 4.786 | mA |
| $I_{SNK\_0p1DS}$ | Output sink current, 0.1x drive strength | VDDHn = 1.8 V | 3.491 | 4.514 | 5.999 | mA |
| $I_{SRC\_0p5DS}$ | Output source current, 0.5x drive strength | VDDHn = 1.8 V | 15.64 | 18.93 | 23.9 | mA |
| $I_{SNK\_0p5DS}$ | Output sink current, 0.5x drive strength | VDDHn = 1.8 V | 13.93 | 18.01 | 23.93 | mA |
| $I_{SRC\_0p75DS}$ | Output source current, 0.75x drive strength | VDDHn = 1.8 V | 25.03 | 30.29 | 38.25 | mA |
| $I_{SNK\_0p75DS}$ | Output sink current, 0.75x drive strength | VDDHn = 1.8 V | 20.89 | 27.01 | 35.89 | mA |
| $I_{SRC\_1p0DS}$ | Output source current, 1.0x drive strength | VDDHn = 1.8 V | 31.28 | 37.86 | 47.8 | mA |
| $I_{SNK\_1p0DS}$ | Output sink current, 1.0x drive strength | VDDHn = 1.8 V | 27.86 | 36.02 | 47.86 | mA |

**Table 43: General Purpose Input/Output (GPIO)**

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| T$_{RISE\_STD\_0P1X}$ | Rise time, 30 pF load, 0.1x drive strength, GPIOx | VDDHn = 1.8 V | - | 11.85 | - | ns |
| T$_{FALL\_STD\_0P1X}$ | Fall time, 30 pF load, 0.1x drive strength, GPIOx | VDDHn = 1.8 V | - | 9.25 | - | ns |
| T$_{RISE\_STD\_0P5X}$ | Rise time, 30 pF load, 0.5x drive strength, GPIOx | VDDHn = 1.8 V | - | 2.4 | - | ns |
| T$_{FALL\_STD\_0P5X}$ | Fall time, 30 pF load, 0.5x drive strength, GPIOx | VDDHn = 1.8 V | - | 2.45 | - | ns |
| T$_{RISE\_STD\_0P75X}$ | Rise time, 30 pF load, 0.75x drive strength, GPIOx | VDDHn = 1.8 V | - | 1.55 | - | ns |
| T$_{FALL\_STD\_0P75X}$ | Fall time, 30 pF load, 0.75x drive strength, GPIOx | VDDHn = 1.8 V | - | 1.65 | - | ns |
| T$_{RISE\_STD\_1P0X}$ | Rise time, 30 pF load, 1.0x drive strength, GPIOx | VDDHn = 1.8 V | - | 1.25 | - | ns |
| T$_{FALL\_STD\_1P0X}$ | Fall time, 30 pF load, 1.0x drive strength, GPIOx | VDDHn = 1.8 V | - | 1.3 | - | ns |
| **High-speed GPIOs: VDDHn = 1.8V** | | | | | | |
| V$_{OH}$ | High-Level output Voltage | | 0.8 x VDDHn | - | | V |
| V$_{OL}$ | Low-Level output Voltage | | | - | 0.2 x VDDHn | V |
| V$_{IH}$ | Positive going input threshold voltage | | 0.7 x VDDHn | - | - | V |
| V$_{IL}$ | Negative going input threshold voltage | | - | - | 0.3 x VDDHn | V |
| V$_{HYS}$ | Input Hysteresis | | 0.12 | 0.14 | 0.16 | V |
| R$_{PD}$ | PULL DOWN resistance | | 42.69 | 51.46 | 60.10 | kΩ |
| R$_{PU}$ | PULL UP resistance | | 42.76 | 51.43 | 59.97 | kΩ |
| C$_{GPI}$ | Input capacitance | | - | 5 | | pF |
| I$_{IN}$ | Input pin leakage current | | - | 1 | 20 | nA |
| I$_{INOD}$ | Open drain output leakage current | | - | 1 | 20 | nA |
| T$_{RISE\_HS\_0P125X}$ | Rise time, 15pF load, 0.125x drive strength, GPIOx | VDDHn = 1.8V | - | 2.29 | - | ns |
| T$_{FALL\_HS\_0P125X}$ | Fall time, 15pF load, 0.125x drive strength, GPIOx | VDDHn = 1.8V | - | 2.05 | - | ns |
| T$_{RISE\_HS\_0P25X}$ | Rise time, 15pF load, 0.25x drive strength, GPIOx | VDDHn = 1.8V | - | 1.15 | - | ns |
| T$_{FALL\_HS\_0P25X}$ | Fall time, 15pF load, 0.25x drive strength, GPIOx | VDDHn = 1.8V | - | 1.03 | - | ns |
| T$_{RISE\_HS\_0P375X}$ | Rise time, 15pF load, 0.375x drive strength, GPIOx | VDDHn = 1.8V | - | 0.77 | - | ns |
| T$_{FALL\_HS\_0P375X}$ | Fall time, 15pF load, 0.375x drive strength, GPIOx | VDDHn = 1.8V | - | 0.70 | - | ns |
| T$_{RISE\_HS\_0P5X}$ | Rise time, 15pF load, 0.5x drive strength, GPIOx | VDDHn = 1.8V | - | 0.58 | - | ns |

**Table 43: General Purpose Input/Output (GPIO)**

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $T_{FALL\_HS\_0P5X}$ | Fall time, 15pF load, 0.5x drive strength, GPIOx | VDDHn = 1.8V | - | 0.53 | - | ns |
| $T_{RISE\_HS\_0P625X}$ | Rise time, 15pF load, 0.625x drive strength, GPIOx | VDDHn = 1.8V | - | 0.47 | - | ns |
| $T_{FALL\_HS\_0P625X}$ | Fall time, 15pF load, 0.625x drive strength, GPIOx | VDDHn = 1.8V | - | 0.43 | - | ns |
| $T_{RISE\_HS\_0P75X}$ | Rise time, 15pF load, 0.75x drive strength, GPIOx | VDDHn = 1.8V | - | 0.39 | - | ns |
| $T_{FALL\_HS\_0P75X}$ | Fall time, 15pF load, 0.75x drive strength, GPIOx | VDDHn = 1.8V | - | 0.37 | - | ns |
| $T_{RISE\_HS\_0P875X}$ | Rise time, 15pF load, 0.875x drive strength, GPIOx | VDDHn = 1.8V | - | 0.34 | - | ns |
| $T_{FALL\_HS\_0P875X}$ | Fall time, 15pF load, 0.875x drive strength, GPIOx | VDDHn = 1.8V | - | 0.32 | - | ns |
| $T_{RISE\_HS\_1P0X}$ | Rise time, 15pF load, 1.0x drive strength, GPIOx | VDDHn = 1.8V | - | 0.30 | - | ns |
| $T_{FALL\_HS\_1P0X}$ | Fall time, 15pF load, 1.0x drive strength, GPIOx | VDDHn = 1.8V | - | 0.28 | - | ns |
| $I_{SRC\_HS\_0p125DS}$ | Output source current, 0.125x drive strength | VDDHn = 1.8V, 0.8*VDDH | 8.25 | 9.71 | 11.75 | mA |
| $I_{SNK\_HS\_0p125DS}$ | Output sink current, 0.125x drive strength | VDDHn = 1.8V, 0.2*VDDH | 8.07 | 9.86 | 12.51 | mA |
| $I_{SRC\_HS\_0p25DS}$ | Output source current, 0.25x drive strength | VDDHn = 1.8V, 0.8*VDDH | 16.51 | 19.43 | 23.51 | mA |
| $I_{SNK\_HS\_0p25DS}$ | Output sink current, 0.25x drive strength | VDDHn = 1.8V, 0.2*VDDH | 16.13 | 19.72 | 25.01 | mA |
| $I_{SRC\_HS\_0p375DS}$ | Output source current, 0.375x drive strength | VDDHn = 1.8V, 0.8*VDDH | 24.77 | 29.14 | 35.27 | mA |
| $I_{SNK\_HS\_0p375DS}$ | Output sink current, 0.375x drive strength | VDDHn = 1.8V, 0.2*VDDH | 24.21 | 29.59 | 37.52 | mA |
| $I_{SRC\_HS\_0p5DS}$ | Output source current, 0.5x drive strength | VDDHn = 1.8V, 0.8*VDDH | 33.03 | 38.86 | 47.03 | mA |
| $I_{SNK\_HS\_0p5DS}$ | Output sink current, 0.5x drive strength | VDDHn = 1.8V, 0.2*VDDH | 32.28 | 39.45 | 50.02 | mA |
| $I_{SRC\_HS\_0p625DS}$ | Output source current, 0.625x drive strength | VDDHn = 1.8V, 0.8*VDDH | 41.29 | 48.58 | 58.8 | mA |
| $I_{SNK\_HS\_0p62DS}$ | Output sink current, 0.625x drive strength | VDDHn = 1.8V, 0.2*VDDH | 40.33 | 49.29 | 62.53 | mA |
| $I_{SRC\_HS\_0p75DS}$ | Output source current, 0.75x drive strength | VDDHn = 1.8V, 0.8*VDDH | 49.56 | 58.30 | 70.56 | mA |
| $I_{SNK\_HS\_0p75DS}$ | Output sink current, 0.75x drive strength | VDDHn = 1.8V, 0.2*VDDH | 48.40 | 59.15 | 75.03 | mA |
| $I_{SRC\_HS\_0p875DS}$ | Output source current, 0.875x drive strength | VDDHn = 1.8V, 0.8*VDDH | 57.81 | 68.01 | 82.32 | mA |
| $I_{SNK\_HS\_0p875DS}$ | Output sink current, 0.875x drive strength | VDDHn = 1.8V, 0.2*VDDH | 56.48 | 69.02 | 87.54 | mA |

**Table 43: General Purpose Input/Output (GPIO)**

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| I<sub>SRC_HS_1p0DS</sub> | Output source current, 1.0x drive strength | VDDHn = 1.8V, 0.8*VDDH | 66.07 | 77.73 | 94.08 | mA |
| I<sub>SNK_HS_1p0DS</sub> | Output sink current, 1.0x drive strength | VDDHn = 1.8V, 0.2*VDDH | 64.55 | 78.88 | 100 | mA |
| **High-speed GPIOs: VDDHn = 1.2V** | | | | | | |
| V<sub>OH</sub> | High-Level output Voltage | | 0.8 x VDDHn | - | - | V |
| V<sub>OL</sub> | Low-Level output Voltage | | - | - | 0.2 x VDDHn | V |
| V<sub>IH</sub> | Positive going input threshold voltage | | 0.7 x VDDHn | - | - | V |
| V<sub>IL</sub> | Negative going input threshold voltage | | - | - | 0.3 x VDDHn | V |
| V<sub>HYS</sub> | Input Hysteresis | | 0.075 | 0.09 | 0.11 | V |
| R<sub>PD</sub> | PULL DOWN resistance | | 43.17 | 52.08 | 60.89 | kΩ |
| R<sub>PU</sub> | PULL UP resistance | | 43.83 | 52.75 | 61.43 | kΩ |
| C<sub>GPI</sub> | Input capacitance | | - | 5 | - | pF |
| I<sub>IN</sub> | Input pin leakage current | | - | 1 | 20 | nA |
| I<sub>INOD</sub> | Open drain output leakage current | | - | 1 | 20 | nA |
| T<sub>RISE_HS_0P125X</sub> | Rise time, 15pF load, 0.125x drive strength, GPIOx | VDDHn = 1.2V | - | 4.09 | - | ns |
| T<sub>FALL_HS_0P125X</sub> | Fall time, 15pF load, 0.125x drive strength, GPIOx | VDDHn = 1.2V | - | 3.69 | - | ns |
| T<sub>RISE_HS_0P25X</sub> | Rise time, 15pF load, 0.25x drive strength, GPIOx | VDDHn = 1.2V | - | 2.03 | - | ns |
| T<sub>FALL_HS_0P25X</sub> | Fall time, 15pF load, 0.25x drive strength, GPIOx | VDDHn = 1.2V | - | 1.27 | - | ns |
| T<sub>RISE_HS_0P375X</sub> | Rise time, 15pF load, 0.375x drive strength, GPIOx | VDDHn = 1.2V | - | 1.36 | - | ns |
| T<sub>FALL_HS_0P375X</sub> | Fall time, 15pF load, 0.375x drive strength, GPIOx | VDDHn = 1.2V | - | 1.25 | - | ns |
| T<sub>RISE_HS_0P5X</sub> | Rise time, 15pF load, 0.5x drive strength, GPIOx | VDDHn = 1.2V | - | 1.03 | - | ns |
| T<sub>FALL_HS_0P5X</sub> | Fall time, 15pF load, 0.5x drive strength, GPIOx | VDDHn = 1.2V | - | 0.95 | - | ns |
| T<sub>RISE_HS_0P625X</sub> | Rise time, 15pF load, 0.625x drive strength, GPIOx | VDDHn = 1.2V | - | 0.83 | - | ns |
| T<sub>FALL_HS_0P625X</sub> | Fall time, 15pF load, 0.625x drive strength, GPIOx | VDDHn = 1.2V | - | 0.77 | - | ns |
| T<sub>RISE_HS_0P75X</sub> | Rise time, 15pF load, 0.75x drive strength, GPIOx | VDDHn = 1.2V | - | 0.69 | - | ns |
| T<sub>FALL_HS_0P75X</sub> | Fall time, 15pF load, 0.75x drive strength, GPIOx | VDDHn = 1.2V | - | 0.66 | - | ns |
| T<sub>RISE_HS_0P875X</sub> | Rise time, 15pF load, 0.875x drive strength, GPIOx | VDDHn = 1.2V | - | 0.60 | - | ns |

**Table 43: General Purpose Input/Output (GPIO)**

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $T_{FALL\_HS\_0P875X}$ | Fall time, 15pF load, 0.875x drive strength, GPIOx | VDDHn = 1.2V | - | 0.57 | - | ns |
| $T_{RISE\_HS\_1P0X}$ | Rise time, 15pF load, 1.0x drive strength, GPIOx | VDDHn = 1.2V | - | 0.53 | - | ns |
| $T_{FALL\_HS\_1P0X}$ | Fall time, 15pF load, 1.0x drive strength, GPIOx | VDDHn = 1.2V | - | 0.51 | - | ns |
| $I_{SRC\_HS\_0p125DS}$ | Output source current, 0.125x drive strength | VDDHn = 1.2V, 0.8*VDDH | 2.97 | 3.60 | 4.51 | mA |
| $I_{SNK\_HS\_0p125DS}$ | Output sink current, 0.125x drive strength | VDDHn = 1.2V, 0.2*VDDH | 2.79 | 3.74 | 5.09 | mA |
| $I_{SRC\_HS\_0p25DS}$ | Output source current, 0.25x drive strength | VDDHn = 1.2V, 0.8*VDDH | 5.94 | 7.20 | 9.027 | mA |
| $I_{SNK\_HS\_0p25DS}$ | Output sink current, 0.25x drive strength | VDDHn = 1.2V, 0.2*VDDH | 5.57 | 7.47 | 10.18 | mA |
| $I_{SRC\_HS\_0p375DS}$ | Output source current, 0.375x drive strength | VDDHn = 1.2V, 0.8*VDDH | 8.92 | 10.81 | 13.55 | mA |
| $I_{SNK\_HS\_0p375DS}$ | Output sink current, 0.375x drive strength | VDDHn = 1.2V, 0.2*VDDH | 8.37 | 11.22 | 15.29 | mA |
| $I_{SRC\_HS\_0p5DS}$ | Output source current, 0.5x drive strength | VDDHn = 1.2V, 0.8*VDDH | 11.90 | 14.42 | 18.07 | mA |
| $I_{SNK\_HS\_0p5DS}$ | Output sink current, 0.5x drive strength | VDDHn = 1.2V, 0.2*VDDH | 11.16 | 14.96 | 20.38 | mA |
| $I_{SRC\_HS\_0p625DS}$ | Output source current, 0.625x drive strength | VDDHn = 1.2V, 0.8*VDDH | 14.88 | 18.03 | 22.6 | mA |
| $I_{SNK\_HS\_0p62DS}$ | Output sink current, 0.625x drive strength | VDDHn = 1.2V, 0.2*VDDH | 13.93 | 18.68 | 25.45 | mA |
| $I_{SRC\_HS\_0p75DS}$ | Output source current, 0.75x drive strength | VDDHn = 1.2V, 0.8*VDDH | 17.85 | 21.63 | 27.12 | mA |
| $I_{SNK\_HS\_0p75DS}$ | Output sink current, 0.75x drive strength | VDDHn = 1.2V, 0.2*VDDH | 16.72 | 22.41 | 30.54 | mA |
| $I_{SRC\_HS\_0p875DS}$ | Output source current, 0.875x drive strength | VDDHn = 1.2V, 0.8*VDDH | 20.83 | 25.25 | 31.65 | mA |
| $I_{SNK\_HS\_0p875DS}$ | Output sink current, 0.875x drive strength | VDDHn = 1.2V, 0.2*VDDH | 19.52 | 26.17 | 35.65 | mA |
| $I_{SRC\_HS\_1p0DS}$ | Output source current, 1.0x drive strength | VDDHn = 1.2V, 0.8*VDDH | 23.81 | 28.85 | 36.17 | mA |
| $I_{SNK\_HS\_1p0DS}$ | Output sink current, 1.0x drive strength | VDDHn = 1.2V, 0.2*VDDH | 22.31 | 29.9 | 40.74 | mA |

1. All GPIOs have Schmitt trigger inputs.
2. The designation "n" corresponds to the voltage source for the pin, e.g., VDDH1.

## 30.7 Clocks/Oscillators

**Table 44: Primary Internal Clocks**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{HFRC\_LP}$ | HFRC frequency - Low Power | | - | 96 | - | MHz |
| $F_{HFRC\_HP}$ | HFRC frequency - High Performance Burst Mode | | - | 192[1] | - | MHz |
| $DC_{HFRC}$ | HFRC duty cycle | | - | 50 | - | % |
| $F_{HFRC2\_LP}$ | HFRC2 frequency - Low Power | | - | 125 | - | MHz |
| $F_{HFRC2\_HP}$ | HFRC2 frequency - High Performance Burst Mode | | - | 250[2] | - | MHz |
| $F_{LFRC}$ | LFRC frequency | | - | 887 | - | Hz |
| $DC_{LFRC}$ | LFRC duty cycle | CLKGEN_CLKOUT_CKSEL = LFRC_DIV2 | - | 50 | - | % |
| $F_{PLL\_OUT}$ | System PLL output frequency | | 1.22 | - | 48 | MHz |
| $F_{PLL\_IN\_INT}$ | System PLL input frequency - Integer Mode | | 1 | - | 48 | MHz |
| $F_{PLL\_IN\_FRAC}$ | System PLL input frequency - Fractional Mode | | 10 | - | 48 | MHz |
| $ACC_{PLL\_OUT\_FRAC}$ | System PLL output accuracy - Fractional Mode | | - | 24 | - | bit |
| $Jitter_{PLL\_OUT\_INT}$ | Maximum integrated jitter (10 kHz to Nyquist) - Integer Mode[3] | | - | - | 15 | ps |
| $Jitter_{PLL\_OUT\_FRAC}$ | Maximum integrated jitter (10 kHz to Nyquist) - Fractional Mode[1] | | - | - | 30 | ps |

1. The HFRC oscillator frequency is used to supply the 192 MHz HFRC_HP clock and is divided by 2 to provide the 96 MHz HFRC_LP clock.
2. The HFRC2 oscillator frequency is used to supply the 250 MHz HFRC2_HP clock and is further divided by 2 to provide the 125 MHz HFRC2_LP clock.
3. This is PLL jitter only and actual jitter seen on clock sinks, either internal or external, will be higher as it includes the jitter added by clocks routings.

**Table 45: Low-Frequency Crystal Oscillator**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{XT}$ | XT frequency | | - | 32.768 | - | kHz |
| $DC_{XT}$ | XT duty cycle | | 50 | 56 | 60 | % |
| $\Delta FXTAL$ | Frequency tolerance[1] | Untrimmed; include initial tolerance/aging/ temperature drift | -50 | - | 50 | PPM |
| ESR | Equivalent serial resistance | | - | - | 90 | KΩ |
| $C_{EXT\_XT\_LOAD}$ | Allowed external XI/XO load capacitance per pin[2] | | - | - | 10 | pF |

1. Recommended to achieve target internal clock frequencies which use the XTAL as reference clock
2. If oscillator allowance (≥ 4x) is insufficient after adding the load caps, the XTALGENCTRL_XTALBIASTRIM field value in MCUCTRL can be increased to help increase the allowance. Default value for this field is 0x38 and can be increased as high as 0x3F (about a 12% increase in bias current).

**Table 46: High-speed Crystal Oscillator**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|--------|-----------|-----------------|-----|-----|-----|------|
| $F_{XTAL}$ | Crystal frequency | | - | 48 | - | MHz |
| ΔFXTAL | Frequency tolerance | Untrimmed; include initial tolerance/aging/temperature drift | -40 | - | 40 | PPM |
| ESR | Equivalent serial resistance | | - | - | 100 | Ω |
| $T_{XTAL}$ | Startup time | | - | 1 | - | ms |
| $C_{EXT\_XTHS\_LOAD}$ | Allowed external XI32M/XO32M load capacitance per pin | | - | 6 | - | pF |

## 30.8  Counter/Timer (TIMER)

**Table 47: Timer (TIMER)**

| Symbol | Parameter | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| F$_{TIMER}$ | Input frequency | - | - | 48 | MHz |

## 30.9   Display Controller (DC)

All DC: Timing characteristics have been determined through simulations.

### 30.9.1   DC QSPI/SPI Interface

**Table 48: DC QSPI/SPI Timing[1]**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{SCLK(OUT)}$ | SCLK OUT frequency | | - | - | 48[2] | MHz |
| $F_{SCLK(IN)}$ | SCLK IN frequency | | - | - | 12 | MHz |
| $T_{LOW(SCLK)}$ | Clock low time | | $0.4/F_{SCLK}$ | - | $0.6/F_{SCLK}$ | s |
| $T_{HIGH(SCLK)}$ | Clock high time | | $0.4/F_{SCLK}$ | - | $0.6/F_{SCLK}$ | s |
| $T_{RISE(SCLK)}$ | Clock rise time | | - | - | 2 | ns |
| $T_{FALL(SCLK)}$ | Clock fall time | | - | - | 1 | ns |
| $T_{SETUP(IN)}$ | Data input data setup time | | 40 | - | - | ns |
| $T_{HOLD(in)}$ | Data input data hold time | | 1 | - | - | ns |
| $T_{HOLD(OUT)}$ | Data output data hold time | | -1 | - | - | ns |
| $T_{VALID(OUT)}$ | Data output data valid time | | - | - | 5 | ns |

1. The CE pin is assumed to be controlled by SW.
2. Note that the specified $T_{VALID(OUT)}$ time of 5ns exceeds a half cycle of 96 MHz clock. If the external SPI peripheral supports full-cycle data capture, then a 96 MHz clock is supported provided it can meet the hold time requirement with the $T_{HOLD(OUT)}$ of -1ns. It is recommended that customers perform compliance testing in their systems to ensure reliable operation.
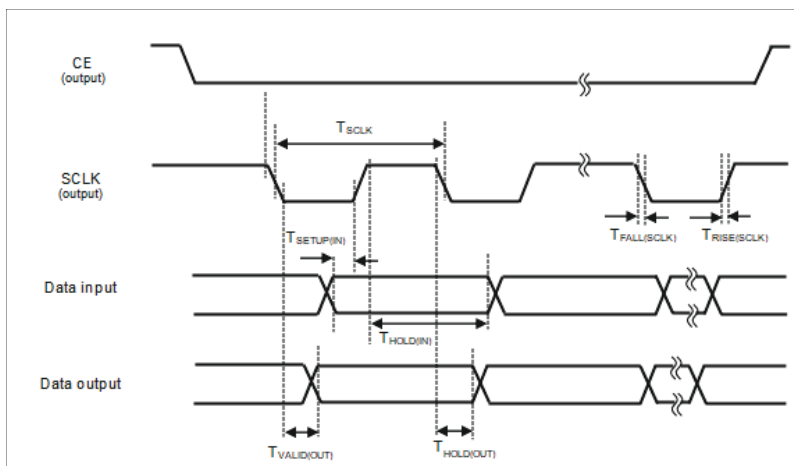


**Figure 59. DC QSPI/SPI Timing Diagram**

### 30.9.2 DC MIPI DPI-2 Interface

**Table 49: DC MIPI DPI-2 Timing**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{PCLK}$ $(1/T_{PCLK})$ | PCLK frequency range | | - | - | 48 | MHz |
| TLOW(PCLK) | Clock low time | | $0.4/F_{PCLK}$ | - | $0.6/F_{PCLK}$ | s |
| THIGH(PCLK) | Clock high time | | $0.4/F_{PCLK}$ | - | $0.6/F_{PCLK}$ | s |
| TRISE(PCLK) | Clock rise time | | - | - | 2 | ns |
| TFALL(PCLK) | Clock fall time | | - | - | 1 | ns |
| THOLD(OUT) | Data output data hold time | | -2 | - | - | ns |
| TVALID(OUT) | Data output data valid time | | - | - | 6.8 | ns |
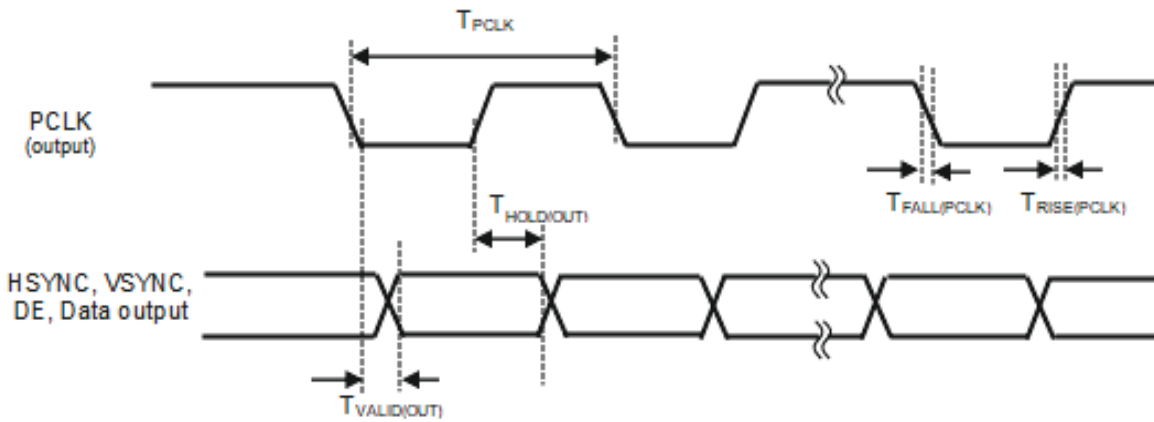


**Figure 60. DC MIPI DPI-2 Timing Diagram**

### 30.9.3 DC MIPI DBI-B Interface

**Table 50: DC MIPI DBI-B Timing[1]**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{WRX}$ | WRX frequency range | | - | - | 48 | MHz |
| $F_{RDX}$ | RDX frequency range | | - | - | $6^2$ | MHz |
| $T_{RDXWRX\_LO}$ | RDX, WRX low time | | $0.4/F_{RDXWRX}$ | - | $0.6/F_{RDXWRX}$ | s |
| $T_{RDXWRX\_HI}$ | RDX, WRX high time | | $0.4/F_{RDXWRX}$ | - | $0.6/F_{RDXWRX}$ | s |
| $T_{RISE(RDX)}$ | RDX rise time | | - | - | 4 | ns |
| $T_{FALL(RDX)}$ | RDX fall time | | - | - | 1 | ns |
| $T_{RISE(WRX)}$ | WRX rise time | | - | - | 2 | ns |
| $T_{FALL(WRX)}$ | WRX fall time | | - | - | 1 | ns |
| $T_{DCXtoRDXWRX}$ | DCX to falling RDX, WRX edge | | -6.5 | - | 4.5 | ns |
| $T_{RDXWRXtoDCX}$ | Rising RDX, WRX edge to DCX edge | | -6.5 | - | 4.5 | ns |
| $T_{SETUP(IN)}$ | Data input setup time | | 30 | - | - | ns |
| $T_{HOLD(IN)}$ | Data input hold time | | 1 | - | - | ns |
| $T_{HOLD(OUT)}$ | Data output hold time | | -6.5 | - | - | ns |
| $T_{VALID(OUT)}$ | Data output valid time | | - | - | 4.5 | ns |

1. The CSX pin is assumed to be controlled by SW.
2. The maximum RDX (read) frequency is limited to 6 Mhz due to the required data input setup time.
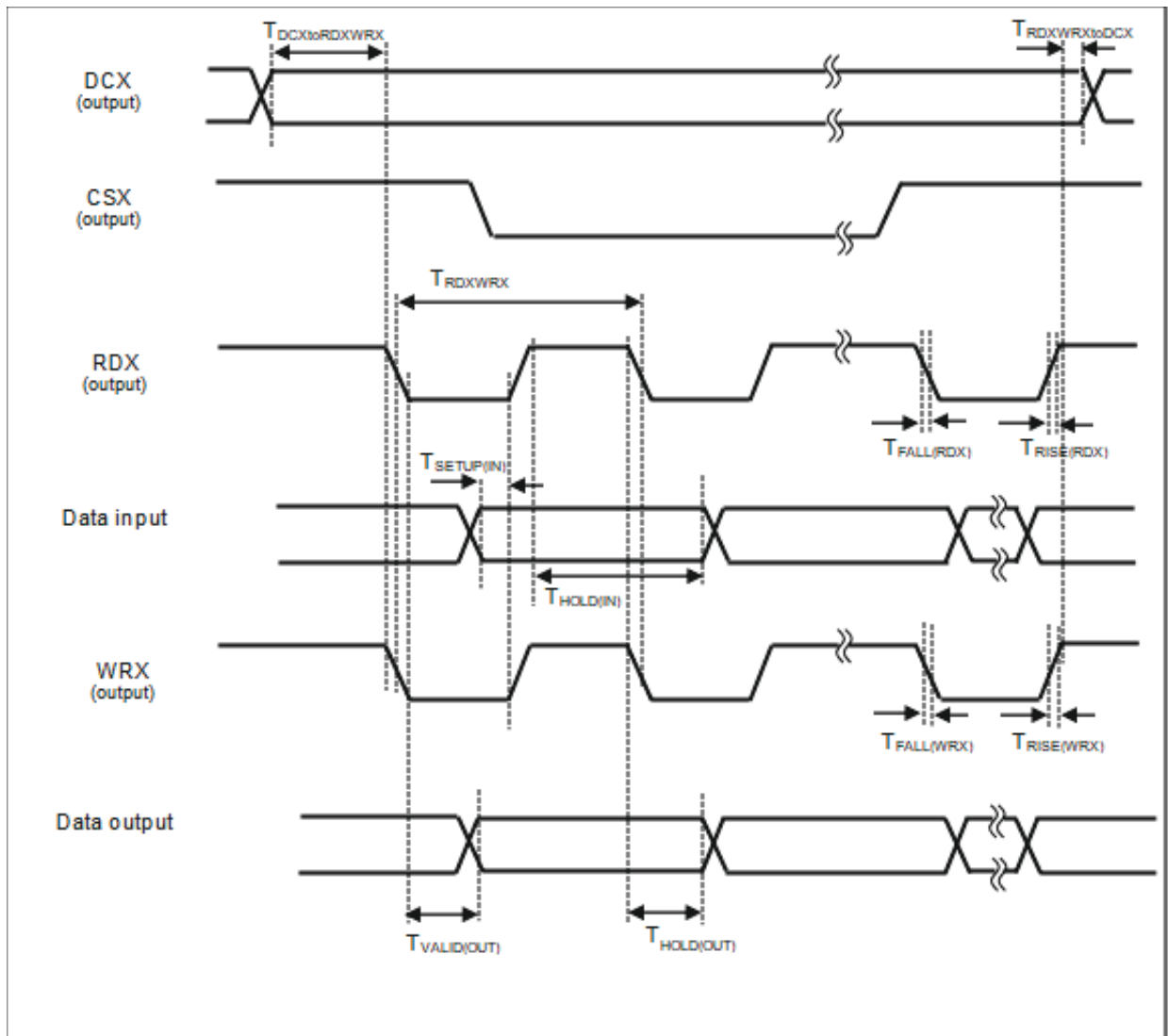
**Figure 61. DC MIPI DBI-B Timing Diagram**

### 30.9.4   DC JDI MIP Interface

**Table 51: DC JDI MIP Timing**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{PCLK}$ (1/$T_{PCLK}$) | SCLK frequency range | | - | - | 48 | MHz |
| TLOW(PCLK) | Clock low time | | $0.4/F_{PCLK}$ | - | $0.6/F_{PCLK}$ | s |
| THIGH(PCLK) | Clock high time | | $0.4/F_{PCLK}$ | - | $0.6/F_{PCLK}$ | s |
| TRISE(PCLK) | Clock rise time | | - | - | 2 | ns |
| TFALL(PCLK) | Clock fall time | | - | - | 1 | ns |
| THOLD(OUT) | Data output data hold time | | -6.2 | - | - | ns |
| TVALID(OUT) | Data output data valid time | | - | - | 6.8 | ns |
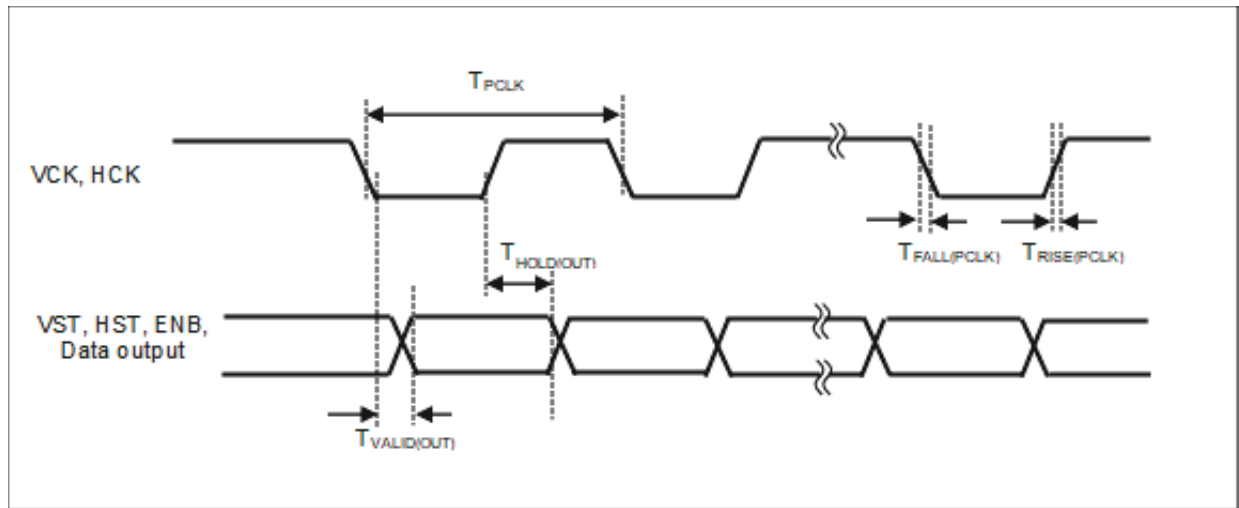


**Figure 62. DC JDI MIP Timing Diagram**

## 30.10 Multi-bit Serial Peripheral Interface (MSPI)

The following sections provide preliminary timing specifications for MSPI, where the sections are broken out to cover MSPI0/MSPI3 and MSPI1/MSPI2.

For the MSPI0 and MSPI3 sections, the specifications cover nominal 1.2 V and 1.8 V operation. For the MSPI1/MSPI2 sections, the specifications cover nominal 1.8 V and 3.3 V operation for MSPI1 and 1.8 V for MSPI2.

**Table 52: MSPI General Specifications**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|--------|-----------|-----------------|-----|-----|-----|------|
| $T_{DLY\_STEP\_SIZE}$ | DDR Delay Step Size | | 80 | - | 200 | ps |

### 30.10.1 MSPI0/MSPI3 Standard 4-Wire Serial SPI Manager Mode

The standard 4-wire serial mode specifications assume that the MSPI is configured to use the same SCLK edge to launch MOSI and to capture MISO. This allows maximum speed to be achieved in serial SPI mode. If the MSPI is configured to follow the standard SPI timing mode, using a different SCLK edge to launch MOSI and to capture MISO, the setup and hold times of MISO are the same, but now with respect to the configured SCLK capture edge.

Timing characteristics have been determined through simulations.

**Table 53: MSPI0/MSPI3 Standard 4-Wire Serial Mode Timing - SPI Manager**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|--------|-----------|-----------------|-----|-----|-----|------|
| $F_{SCLK(MOSI)}$ | SCLK frequency for MOSI output | | - | - | 125 | MHz |
| $F_{SCLK(MISO)}$ | SCLK frequency for MISO input | | - | - | 48 | MHz |
| $T_{LOW(SCLK)}$ | Clock low time | | $0.46/F_{SCLK}$ | - | $0.54/F_{SCLK}$ | s |
| $T_{HIGH(SCLK)}$ | Clock high time | | $0.46/F_{SCLK}$ | - | $0.54/F_{SCLK}$ | s |
| $T_{RISE(SCLK)}$ | Clock rise time | 15 pF Load, GPIO Drive Strength=0x3 | - | - | 2 | ns |
| $T_{FALL(SCLK)}$ | Clock fall time | 15 pF Load, GPIO Drive Strength=0x3 | - | - | 1 | ns |
| $T_{CEtoSCLK}$ | CE to first SCLK edge | | $0.5/F_{SCLK}$ | - | $2.0/F_{SCLK}$ | s |
| $T_{SCLKtoCE}$ | Last SCLK to CE | | $0.5/F_{SCLK}$ | - | $2.0/F_{SCLK}$ | s |
| $T_{SETUP(MISO)}$ | MISO input data setup time | | 7.0 | - | - | ns |
| $T_{HOLD(MISO)}$ | MISO input data hold time | | -5.0 | - | - | ns |
| $T_{HOLD(MOSI)}$ | MOSI output data hold time | 15 pF Load, GPIO Drive Strength=0x3 | -1.3 | - | - | ns |
| $T_{VALID(MOSI)}$ | MOSI output data valid time | 15 pF Load, GPIO Drive Strength=0x3 | - | - | 0.6 | ns |
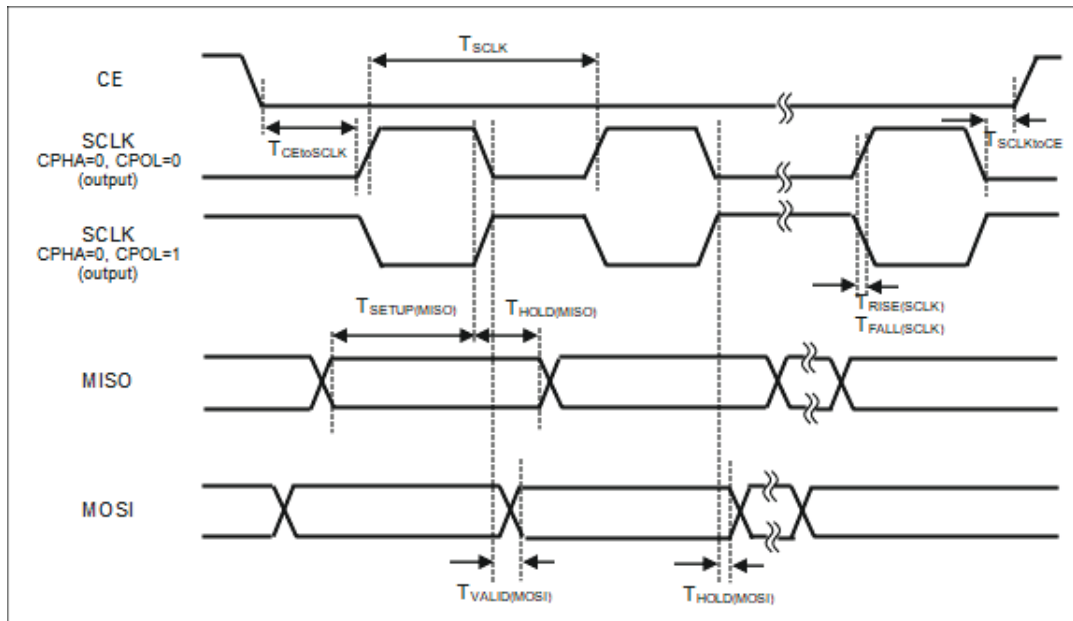
**Figure 63. MSPI0/MSPI3 Standard 4-Wire Mode Timing Diagram - SPI Manager, CPHA=0**
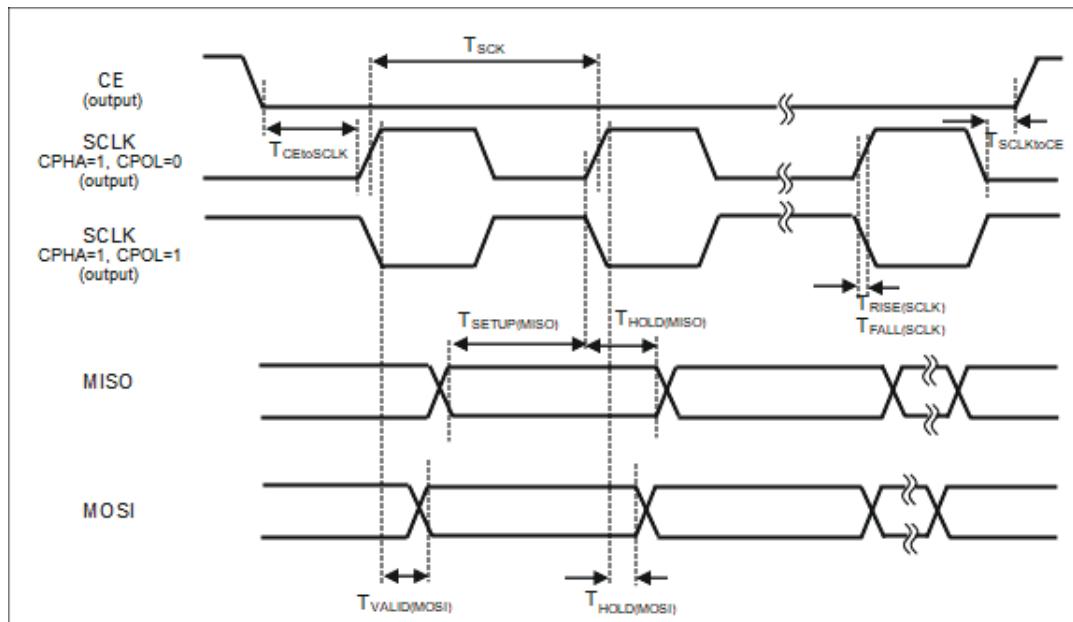


**Figure 64. MSPI0/MSPI3 Standard 4-Wire Mode Timing Diagram - SPI Manager, CPHA=1**

### 30.10.2  MSPI0/MSPI3 SDR Non-DQS Mode (Quad Data Width)

The following specifications for MSPI0 and MSPI3 cover QuadSPI (x4) data width.

Timing characteristics have been determined through simulations.

Timing is based on 0-tap delay on the RX and TX delay lines.

To use on-chip RX delay lines, the external data skew needs to be less than or equal to 4.0 ns.

Timing parameters are applicable to both 1.2 V and 1.8 V nominal IO voltage.

**Table 54: MSPI0/MSPI3 Timing - SDR Non-DQS Mode**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{SCLK}$ | SCLK frequency range | | - | - | 125 | MHz |
| $T_{SCLK\_LO}$ | Clock low time | | $0.46/F_{SCLK}$ | - | $0.54/F_{SCLK}$ | s |
| $T_{SCLK\_HI}$ | Clock high time | | $0.46/F_{SCLK}$ | - | $0.54/F_{SCLK}$ | s |
| $T_{RISE(SCLK)}$ | Clock rise time | 15 pF Load, GPIO Drive Strength=0x3 | - | - | 2 | ns |
| $T_{FALL(SCLK)}$ | Clock fall time | 15 pF Load, GPIO Drive Strength=0x3 | - | - | 1 | ns |
| $T_{CEtoSCLK}$ | CE to first SCLK edge | | $0.5/F_{SCLK}$ | - | $2.0/F_{SCLK}$ | s |
| $T_{SCLKtoCE}$ | Last SCLK to CE | | $0.5/F_{SCLK}$ | - | $3.0/F_{SCLK}$ | s |
| $T_{SETUP(IN)}$ | Data input setup time | | 7.0 | - | - | ns |
| $T_{HOLD(IN)}$ | Data input hold time | | -5.0 | - | - | ns |
| $T_{HOLD(OUT)}$ | Data output hold time | 15 pF Load, GPIO Drive Strength=0x3 | -1.3 | - | - | ns |
| $T_{VALID(OUT)}$ | Data output valid time | 15 pF Load, GPIO Drive Strength=0x3 | - | - | 0.6 | ns |
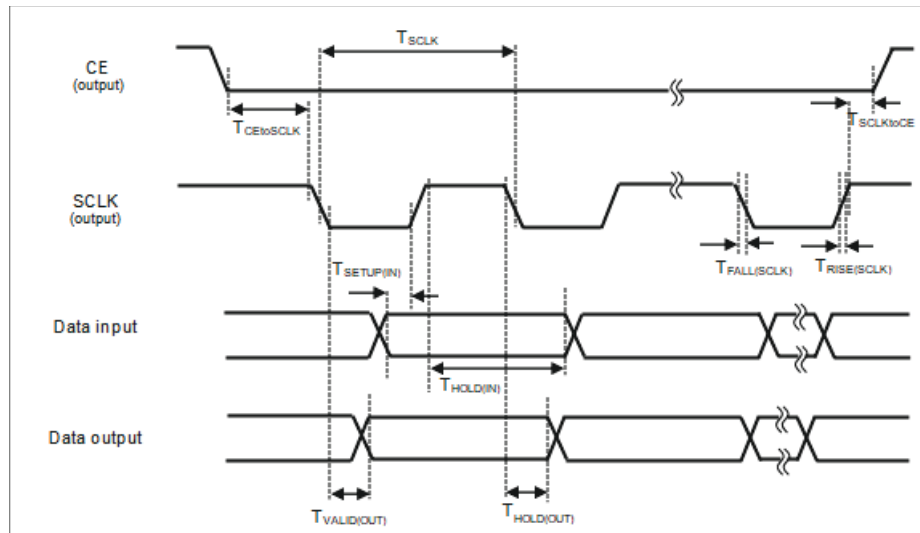
**Figure 65. Timing Diagram for MSPI0/MSPI3 SDR Non-DQS Mode**

Page 254 of 279

### 30.10.3  MSPI0/MSPI3 DDR with DQS Mode (All Supported Data Widths)

Timing characteristics have been determined by measurements on chip.

The following specifications for MSPI0 and MSPI3 cover all supported data widths from QuadSPI (x4) up to HexSPI (x16).

To use on-chip RX delay lines, the external data skew needs to be less than or equal to 1.6 ns.

Timing parameters are applicable to both 1.2V and 1.8V nominal IO voltage.

Timing characteristics have been determined by measurements on chip.

**Table 55: MSPI0/MSPI3 Timing - DDR with DQS Mode**

| Symbol | Parameter | Test Conditions[1] | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{SCLK}$ | SCLK frequency range | | - | - | 125 | MHz |
| $T_{SCLK\_LO}$ | Clock low time | 15 pF Load, GPIO Drive Strength=0.5x (Data) / 1.0x (DQS/CE/CLK) | $0.46/F_{SCLK}$ | - | $0.54/F_{SCLK}$ | s |
| $T_{SCLK\_HI}$ | Clock high time | 15 pF Load, GPIO Drive Strength=0.5x (Data) / 1.0x (DQS/CE/CLK) | $0.46/F_{SCLK}$ | - | $0.54/F_{SCLK}$ | s |
| $T_{RISE(SCLK)}$ | Clock rise time | 15 pF Load, GPIO Drive Strength=0.5x (Data) / 1.0x (DQS/CE/CLK) | - | 1 | - | ns |
| $T_{FALL(SCLK)}$ | Clock fall time | 15 pF Load, GPIO Drive Strength=0.5x (Data) / 1.0x (DQS/CE/CLK) | - | 1.6 | - | ns |
| $T_{CEtoSCLK}$ | CE to first SCLK edge | 15 pF Load, GPIO Drive Strength=0.5x (Data) / 1.0x (DQS/CE/CLK) | $0.5/F_{SCLK}$ | - | $2.0/F_{SCLK}$ | s |
| $T_{SCLKtoCE}$ | Last SCLK to CE | 15 pF Load, GPIO Drive Strength=0.5x (Data) / 1.0x (DQS/CE/CLK) | $0.5/F_{SCLK}$ | - | $2.0/F_{SCLK}$ | s |
| $T_{SETUP(IN)}$ | Data input setup time | 15 pF Load, GPIO Drive Strength=0.5x (Data)/1.0x (DQS/CE/CLK), TURN-AROUND0=10, TXDQSDLY=0, RXDQS-DLY=7 | 0.6 | - | - | ns |
| $T_{HOLD(IN)}$ | Data input hold time | 15 pF Load, GPIO Drive Strength=0.5x (Data)/1.0x (DQS/CE/CLK), TURN-AROUND0=10, TXDQSDLY=0, RXDQS-DLY=7 | 2 | - | - | ns |
| $T_{HOLD(OUT)}$ | Data output hold time | 15 pF Load, GPIO Drive Strength=0.5x (Data) / 1.0x (DQS/CE/CLK) | $(0.225/F_{SCLK})$ $- 0.9(10)^{-9}$ | - | - | s |
| $T_{VALID(OUT)}$ | Data output valid time | 15 pF Load, GPIO Drive Strength=0.5x (Data) / 1.0x (DQS/CE/CLK) | - | - | $(0.275/F_{SCLK})$ $+ 0.9(10)^{-9}$ | s |

1. Turnaround set per memory target specification. RXDQS/TXDQS delay values determined by timing scan. See AmbiqSuite SDK for more details.
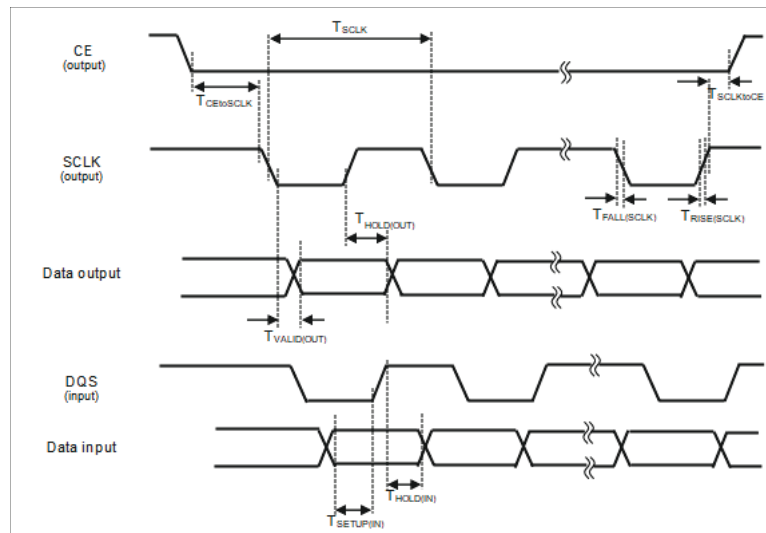
**Figure 66. MSPI0/MSPI3 DDR with DQS Mode Timing Diagram**

### 30.10.4  MSPI1/MSPI2 Standard 4-Wire Serial SPI Manager Mode

The following timing specifications assume that the MSPI is configured to use the same SCLK edge to launch MOSI and to capture MISO. This allows maximum speed to be achieved in serial SPI mode. If the MSPI is configured to follow the standard SPI timing mode, using a different SCLK edge to launch MOSI and to capture MISO, the setup and hold times of MISO is the same, but now with respect to the configured SCLK capture edge.

Timing characteristics have been determined through simulations.

**Table 56: MSPI1/MSPI2 Standard 4-Wire Serial Mode Timing - SPI Manager**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{SCLK(MOSI)}$ | SCLK frequency for MOSI output | | - | - | 96 | MHz |
| $F_{SCLK(MISO)}$ | SCLK frequency for MISO input | | - | - | 48 | MHz |
| $T_{LOW(SCLK)}$ | Clock low time | | $0.45/F_{SCLK}$ | - | $0.55/F_{SCLK}$ | s |
| $T_{HIGH(SCLK)}$ | Clock high time | | $0.45/F_{SCLK}$ | - | $0.55/F_{SCLK}$ | s |
| $T_{RISE(SCLK)}$ | Clock rise time | 15 pF Load, GPIO Drive Strength=0x3 | - | - | 2 | ns |
| $T_{FALL(SCLK)}$ | Clock fall time | 15 pF Load, GPIO Drive Strength=0x3 | - | - | 1 | ns |
| $T_{CEtoSCLK}$ | CE to first SCLK edge | | $0.5/F_{SCLK}$ | - | $2.0/F_{SCLK}$ | s |
| $T_{SCLKtoCE}$ | Last SCLK to CE | | $0.5/F_{SCLK}$ | - | $2.0/F_{SCLK}$ | s |
| $T_{SETUP(MISO)}$ | MISO input data setup time | | 10.0 | - | - | ns |
| $T_{HOLD(MISO)}$ | MISO input data hold time | | -6.5 | - | - | ns |
| $T_{HOLD(MOSI)}$ | MOSI output data hold time | 15 pF Load, GPIO Drive Strength=0x3 | -2.3 | - | - | ns |

**Table 56: MSPI1/MSPI2 Standard 4-Wire Serial Mode Timing - SPI Manager**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $T_{VALID(MOSI)}$ | MOSI output data valid time | 15 pF Load, GPIO Drive Strength=0x3 | - | - | 1.65 | ns |
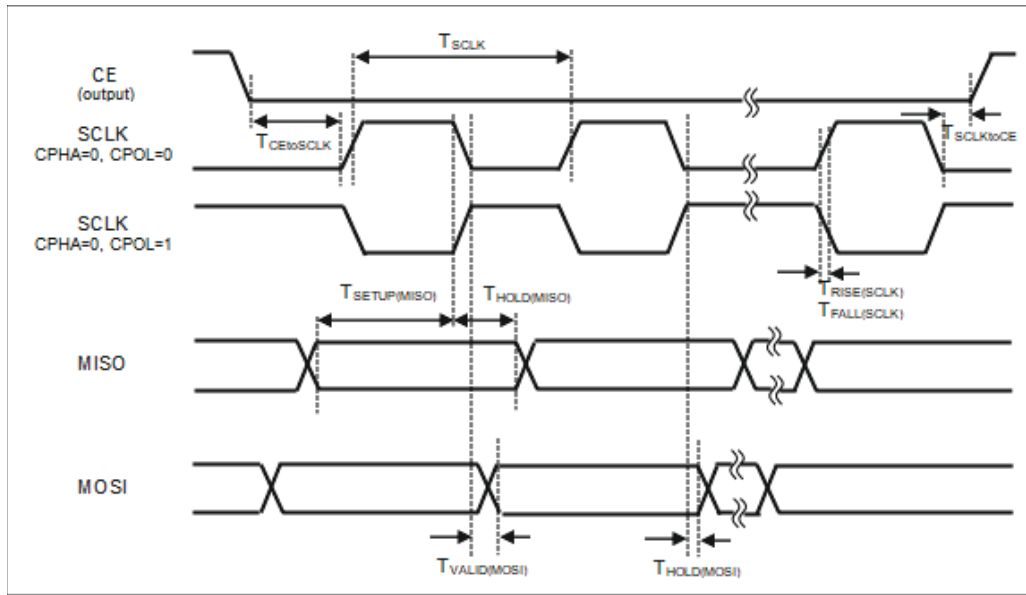


**Figure 67. MSPI1/MSPI2 Standard 4-Wire Mode Timing Diagram - SPI Manager, CPHA=0**
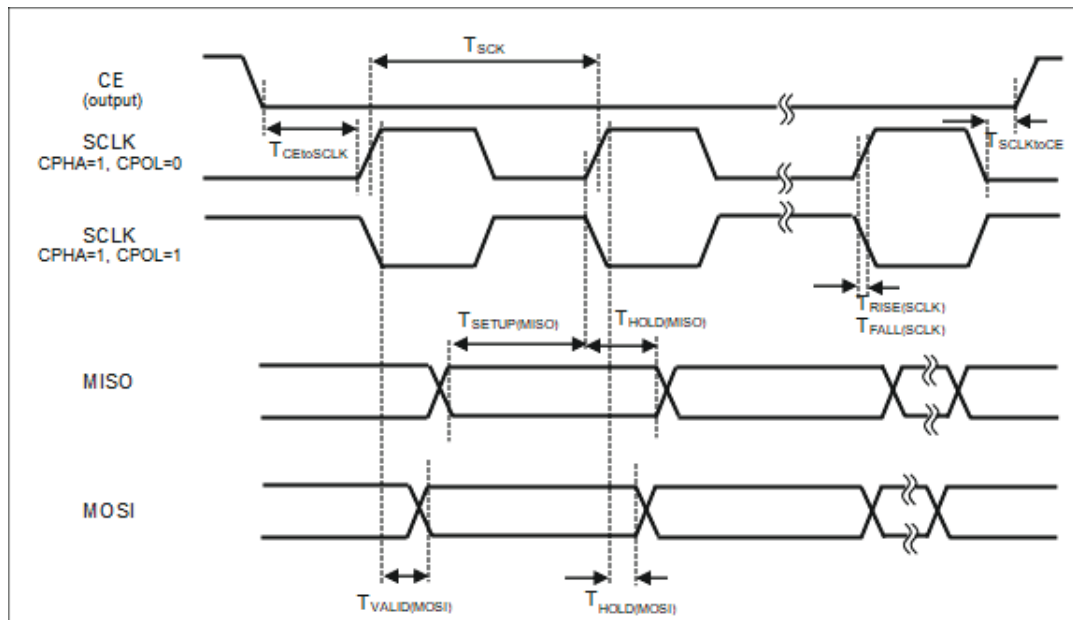


**Figure 68. MSPI1/MSPI2 Standard 4-Wire Mode Timing Diagram - SPI Manager, CPHA=1**

### 30.10.5 MSPI1/MSPI2 SDR Non-DQS Mode *(Quad Data Width)*

The following specifications for MSPI1 and MSPI2 cover QuadSPI (x4) data width.

Timing characteristics have been determined by measurements on chip.

To use on-chip RX delay lines, the external data skew needs to be less than or equal to 4.0 ns.

Timing parameters are applicable to both 1.2 V and 1.8 V nominal IO voltage.

**Table 57: MSPI1/MSPI2 Timing - SDR Non-DQS Mode**

| Symbol | Parameter | Test Conditions[1] | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{SCLK}$ | SCLK frequency range | | - | - | 96 | MHz |
| $T_{SCLK\_LO}$ | Clock low time | 15 pF Load, GPIO Drive Strength=0.1x | $0.45/F_{SCLK}$ | - | $0.55/F_{SCLK}$ | s |
| $T_{SCLK\_HI}$ | Clock high time | 15 pF Load, GPIO Drive Strength=0.1x | $0.45/F_{SCLK}$ | - | $0.55/F_{SCLK}$ | s |
| $T_{RISE(SCLK)}$ | Clock rise time | 15 pF Load, GPIO Drive Strength=0.1x | - | 1.0 | - | ns |
| $T_{FALL(SCLK)}$ | Clock fall time | 15 pF Load, GPIO Drive Strength=0.1x | - | 1.4 | - | ns |
| $T_{CEtoSCLK}$ | CE to first SCLK edge | 15 pF Load, GPIO Drive Strength=0.1x | $0.5/F_{SCLK}$ | - | $2.0/F_{SCLK}$ | s |
| $T_{SCLKtoCE}$ | Last SCLK to CE | 15 pF Load, GPIO Drive Strength=0.1x | $0.5/F_{SCLK}$ | - | $3.0/F_{SCLK}$ | s |
| $T_{SETUP(IN)}$ | Data input setup time | 15 pF Load, GPIO Drive Strength=0.1x, TURNAROUND0=6, TXDQSDELAY0=3, RXDQSDELAY=6 | 3.4 | - | - | ns |
| $T_{HOLD(IN)}$ | Data input hold time | 15 pF Load, GPIO Drive Strength=0.1x, TURNAROUND0=6, TXDQSDELAY0=3, RXDQSDELAY=6 | 0 | - | - | ns |
| $T_{HOLD(OUT)}$ | Data output hold time | 15 pF Load, GPIO Drive Strength=0.1x | -2.3 | - | - | ns |
| $T_{VALID(OUT)}$ | Data output valid time | 15 pF Load, GPIO Drive Strength=0.1x | - | - | 2.0 | ns |

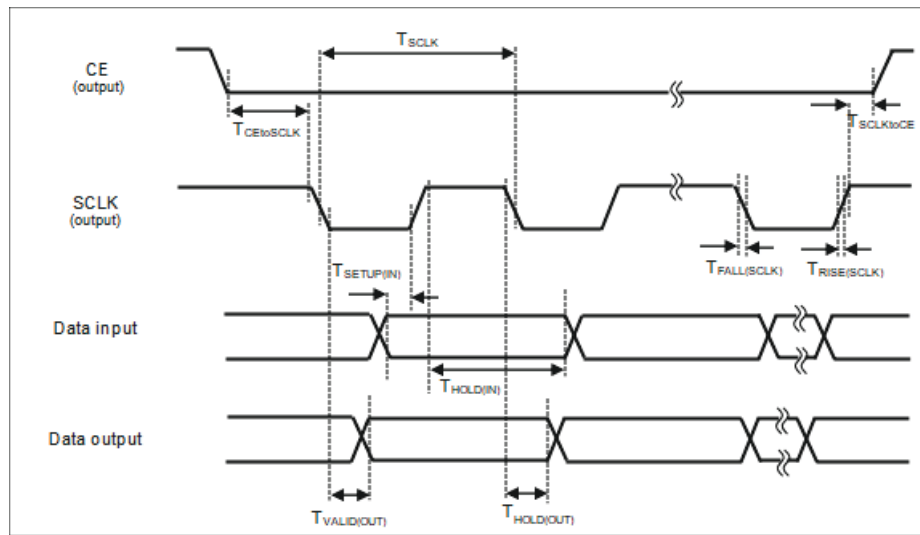1. Turnaround set per memory target specification.

**Figure 69. MSPI1/MSPI2 SDR Non-DQS Mode Timing Diagram**

### 30.10.6  MSPI1/MSPI2 DDR with DQS Mode *(All Supported Data Widths)*

The following specifications for MSPI1 and MSPI2 cover data widths QuadSPI (x4) and OctalSPI (x8).

Timing characteristics have been determined by measurements on chip.

**Table 58: MSPI1/MSPI2 Timing - DDR with DQS Mode**

| Symbol | Parameter | Test Conditions[1] | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{SCLK}$ | SCLK frequency range | | - | - | 48 | MHz |
| $T_{SCLK\_LO}$ | Clock low time | 15 pF Load, GPIO Drive Strength=0.5x | $0.45/F_{SCLK}$ | - | $0.55/F_{SCLK}$ | s |
| $T_{SCLK\_HI}$ | Clock high time | 15 pF Load, GPIO Drive Strength=0.5x | $0.45/F_{SCLK}$ | - | $0.55/F_{SCLK}$ | s |
| $T_{RISE(SCLK)}$ | Clock rise time | 15 pF Load, GPIO Drive Strength=0.5x | - | 1.5 | - | ns |
| $T_{FALL(SCLK)}$ | Clock fall time | 15 pF Load, GPIO Drive Strength=0.5x | - | 1.7 | - | ns |
| $T_{CEtoSCLK}$ | CE to first SCLK edge | 15 pF Load, GPIO Drive Strength=0.5x | $0.5/F_{SCLK}$ | - | $2.0/F_{SCLK}$ | s |
| $T_{SCLKtoCE}$ | Last SCLK to CE | 15 pF Load, GPIO Drive Strength=0.5x | $0.5/F_{SCLK}$ | - | $2.0/F_{SCLK}$ | s |
| $T_{SETUP(IN)}$ | Data input setup time | 15 pF Load, GPIO Drive Strength=0.5x, TURN-AROUND0=6, TXDQSDELAY0=0, RXDQSDELAY=6 | 1.5 | - | - | ns |
| $T_{HOLD(IN)}$ | Data input hold time | 15 pF Load, GPIO Drive Strength=0.5x, TURN-AROUND0=6, TXDQSDELAY0=0, RXDQSDELAY=6 | 2.5 | - | - | ns |
| $T_{HOLD(OUT)}$ | Data output hold time | 15 pF Load, GPIO Drive Strength=0.5x | $(0.225/F_{SCLK}) - 0.3(10)^{-9}$ | - | - | s |
| $T_{VALID(OUT)}$ | Data output valid time | 15 pF Load, GPIO Drive Strength=0.5x | - | - | $(0.275/F_{SCLK}) + 2.9(10)^{-9}$ | s |

1. Turnaround set per memory target specification.
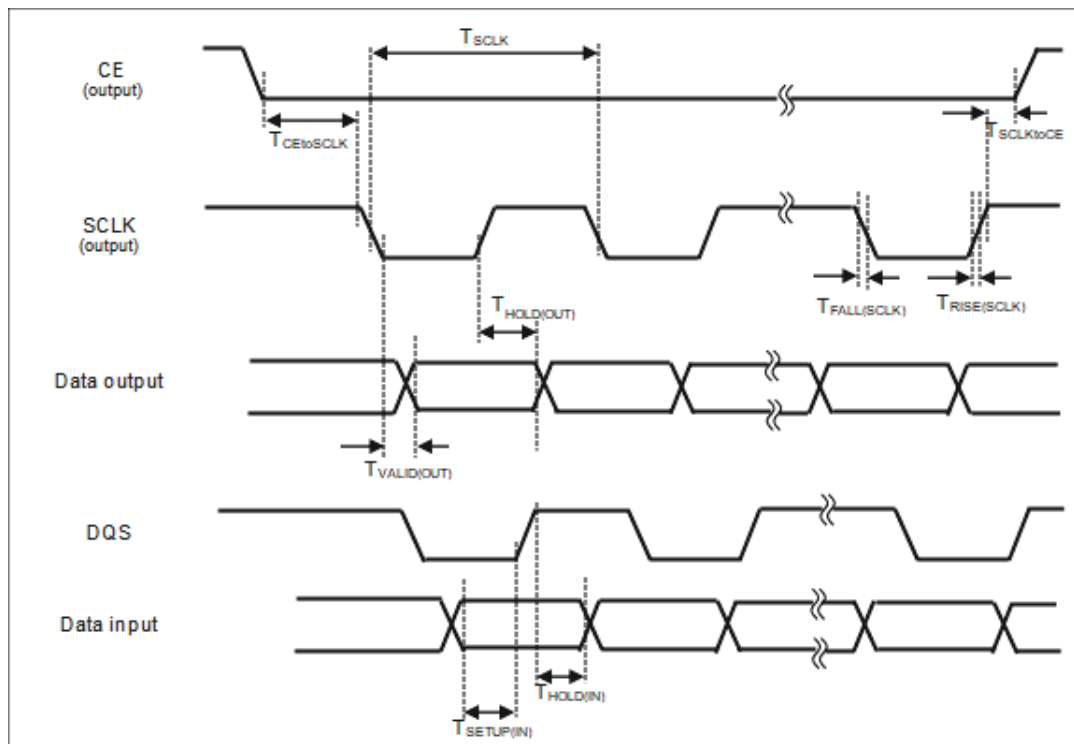
**Figure 70. MSPI1/MSPI2 DDR with DQS Mode Timing Diagram**

## 30.11  I$^2$C/SPI Manager (IOM)

### 30.11.1  Serial Peripheral Interface (SPI) Manager Interface

Timing characteristics have been determined by measurements on chip.

Settings: The following specifications are with a drivestrength of 0.5x for MISO/CE and 0.1x for SCK/MOSI.

**Table 59: SPI Manager Interface Timing**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| F$_{SCLK}$ (1/T$_{SCLK}$) | SCLK frequency range | | - | - | 48 | MHz |
| F$_{SCLK-3WIRE}$ | SCLK frequency range - 3-wire mode | | - | - | 12 | MHz |
| TLOW(SCLK) | Clock low time | DS=see settings, 30pF load, | 0.45/F$_{SCLK}$ | - | 0.55/F$_{SCLK}$ | s |
| THIGH(SCLK) | Clock high time | DS=see settings, 30pF load | 0.45/F$_{SCLK}$ | - | 0.55/F$_{SCLK}$ | s |
| TRISE(SCLK) | Clock rise time | DS=see settings, 30pF load | - | 1.5 | - | ns |
| TFALL(SCLK) | Clock fall time | DS=see settings, 30pF load | - | 2.8 | - | ns |
| TCEtoSCLK | CE to first SCLK edge | DS=see settings, 30pF load | 0.5/F$_{SCLK}$ | - | 2.0/F$_{SCLK}$ | s |
| TSCLKtoCE | Last SCLK to CE | DS=see settings, 30pF load | 0.5/F$_{SCLK}$ | - | 2.0/F$_{SCLK}$ | s |
| TSETUP (MISO) | MISO input data setup time | DS=see settings, 30pF load | 5.5 | - | - | ns |
| THOLD (MISO) | MISO input data hold time | DS=see settings, 30pF load | 2 | - | - | ns |
| TSETUP_3WIRE (MOSI_IN) | MOSI input data setup time for 3-wire mode[1] | DS=see settings, 30pF load | 34 | - | - | ns |
| THOLD_3WIRE (MOSI_IN) | MOSI input data hold time for 3-wire mode | DS=see settings, 30pF load | 2 | - | - | ns |
| TVALID_HOLD (MOSI) | MOSI output data hold time | DS=see settings, 30pF load | 0 | - | - | ns |
| TVALID (MOSI) | MOSI output data valid time | DS=see settings, 30pF load | | - | 6 | ns |
| TVALID_HOLD_3WIRE (MOSI) | MOSI output data hold time for 3-wire mode | DS=see settings, 30pF load | -4 | - | - | ns |
| TVALID_3WIRE (MOSI) | MOSI output data valid time for 3-wire mode | DS=see settings, 30pF load | - | - | 32 | ns |

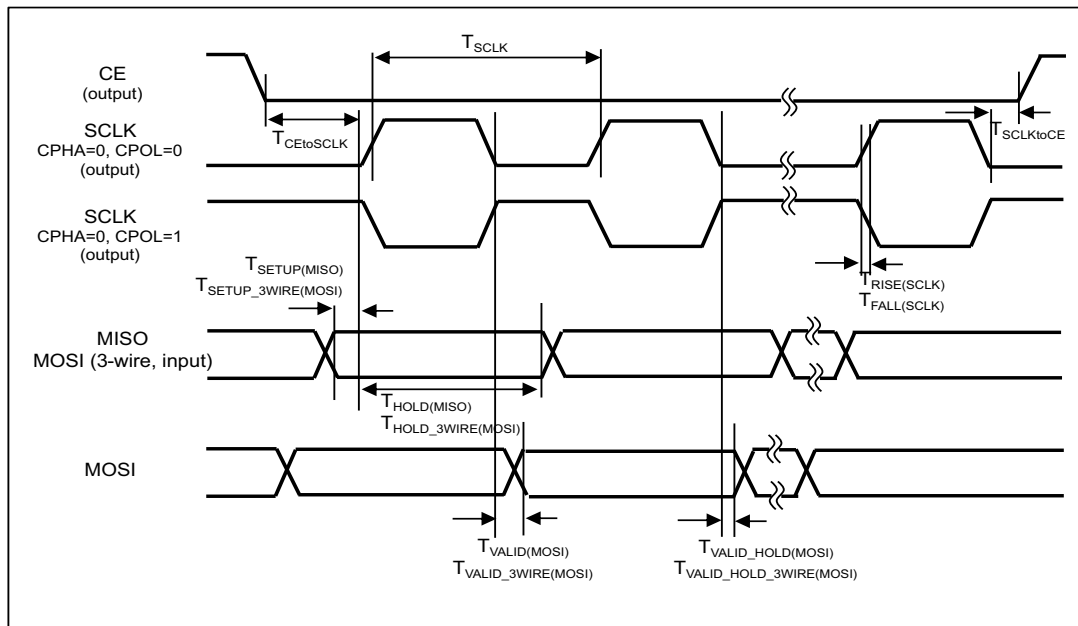1. For 3-wire mode, MOSI is a bidirectional pin. All 3-wire mode times are from simulation data.

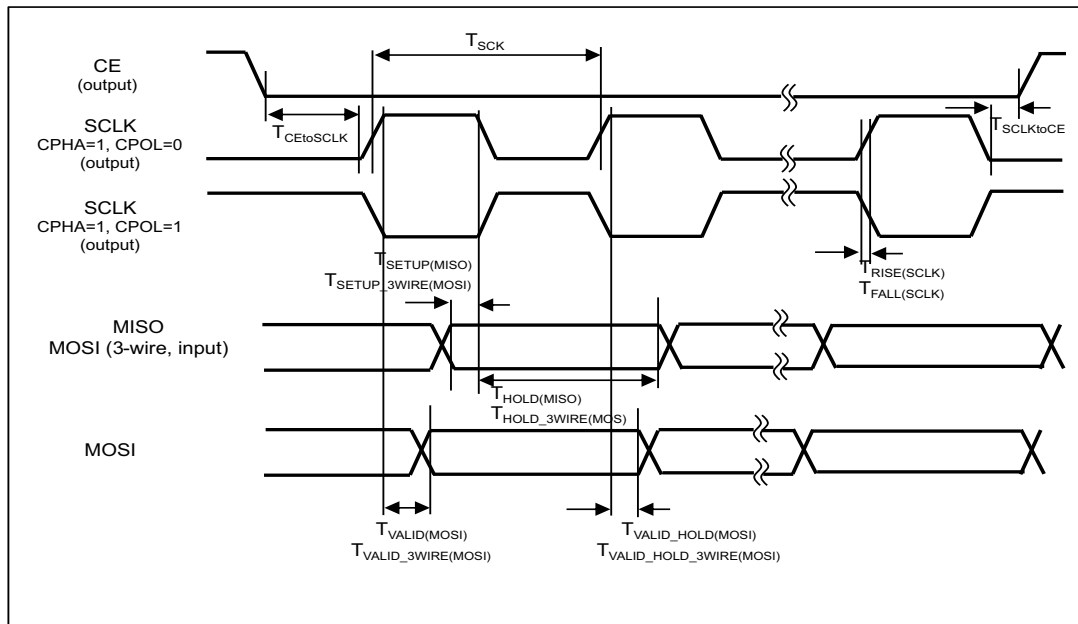**Figure 71. SPI Manager Interface Timing Diagram, CPHA = 0**



**Figure 72. SPI Manager Interface Timing Diagram, CPHA = 1**

## 30.12  I$^2$C/SPI Subordinate (IOS)

### 30.12.1  Serial Peripheral Interface (SPI) Subordinate Interface

Timing characteristics have been determined by measurements on chip.

Settings: The following specifications are with a drivestrength of 0.5x for MISO/CE and 0.1x for SCK/MOSI.

**Table 60: SPI Subordinate Interface Timing**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{SCLK}$ (1/$T_{SCLK}$) | SCLK frequency range | | - | 8 | 24 | MHz |
| $F_{SCLK-3WIRE}$ | SCLK frequency range - 3-wire mode | | - | 8 | 12 | MHz |
| $T_{LOW(SCLK)}$ | Clock low time | | 0.45/$F_{SCLK}$ | - | 0.55/$F_{SCLK}$ | s |
| $T_{HIGH(SCLK)}$ | Clock high time | | 0.45/$F_{SCLK}$ | - | 0.55/$F_{SCLK}$ | s |
| $T_{RISE(SCLK)}$ | Clock rise time | | - | 2 | - | ns |
| $T_{FALL(SCLK)}$ | Clock fall time | | - | 2 | - | ns |
| $T_{CEtoSCLK}$ | CE to first SCLK edge | | 18 | - | - | ns |
| $T_{SCLKtoCE}$ | Last SCLK to CE | | 18 | - | - | ns |
| $T_{SETUP(MOSI)}$ | MOSI input data setup time | | 3.9 | - | - | ns |
| $T_{HOLD(MOSI)}$ | MOSI input data hold time | | 3 | - | - | ns |
| $T_{SETUP\_3WIRE(MOSI\_IN)}$[1] | MOSI input data setup time for 3-wire mode[2] | | 35 | - | - | ns |
| $T_{HOLD\_3WIRE(MOSI\_IN)}$ | MOSI input data hold time for 3-wire mode | | 3 | - | - | ns |
| $T_{VALID\_HOLD(MISO)}$ | MISO output data hold time | DS=See settings, 15 pF Load | 4 | - | - | ns |
| $T_{VALID(MISO)}$ | MISO output data valid time | DS=See settings, 15 pF Load | - | - | 15.5 | ns |
| $T_{VALID\_HOLD\_3WIRE(MOSI)}$ | MOSI output data hold time for 3-wire mode | DS=See settings, 15 pF Load | 4 | - | - | ns |
| $T_{VALID\_3WIRE(MOSI)}$ | MOSI output data valid time for 3-wire mode | DS=See settings, 15 pF Load | - | - | 32 | ns |

1. TSETUP_3WIRE(MOSI_IN), THOLD_3WIRE(MOSI_IN), TVALID_HOLD_3WIRE(MOSI), and TVALID_3WIRE(MOSI) specifications are from simulation data.
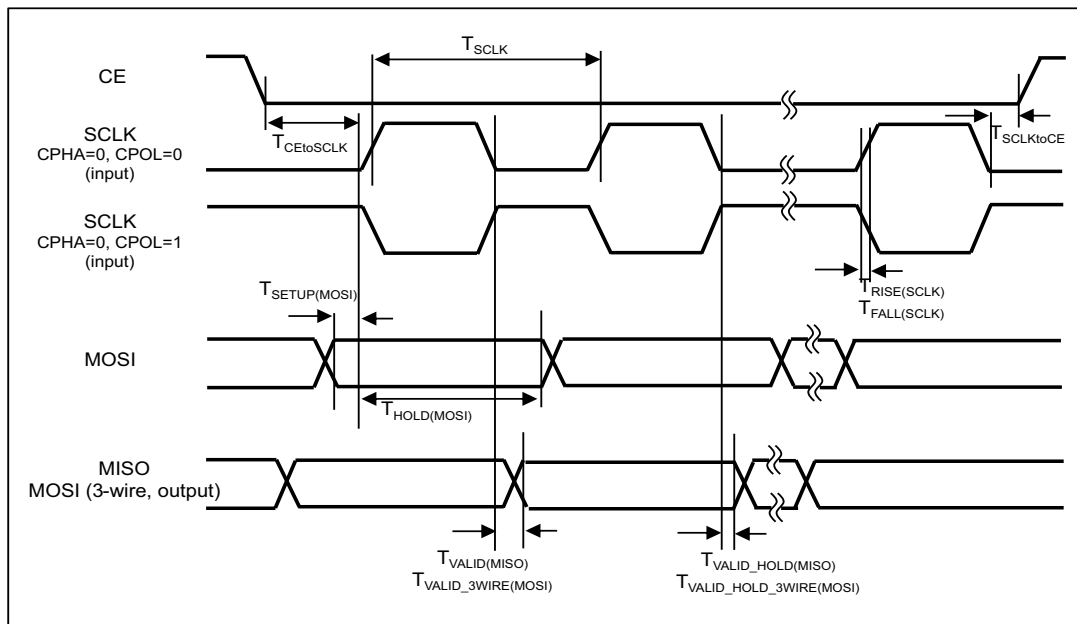2. For 3-wire mode, MOSI is a bidirectional pin.

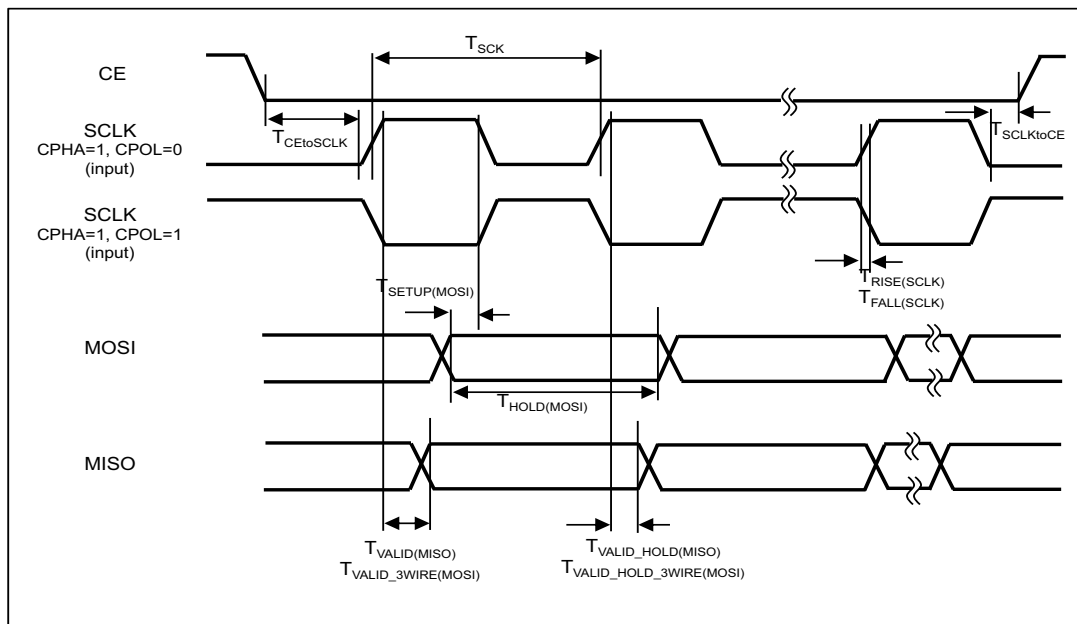**Figure 73. SPI Subordinate Interface Timing Diagram, CPHA = 0**



**Figure 74. SPI Subordinate Interface Timing Diagram, CPHA = 1**

## 30.13 Secure Digital Input Output (SDIO)

**Table 61: SDIO General Specifications**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $T_{TX\_DLY\_STEP\_SIZE}$ | TX Delay Step Size | | 0.12 | - | 0.40 | ns |
| $T_{RX\_DLY\_STEP\_SIZE}$ | RX Delay Step Size | | 0.09 | - | 0.35 | ns |

### 30.13.1 Default SD Interface

Timing characteristics have been determined through simulations.

1. Timing parameters are based on a nominal voltage of 3.3 V.
2. Data, CMD output hold time is based on FF/H/H/H/-40C corner.
3. The timing parameters are based on 0-tap TX and RX delay lines, with SDIO_HOSTCTRL1_HIS-PEEDEN bit = 0.
4. Default Speed Card specifications:

   Input setup/hold = 5 ns/5 ns, with respect to clock rise

   Output min/max = 0 ns/14 ns, with respect to clock fall

5. High Speed Card specifications with respect to clock rise:

   Input setup/hold = 6 ns/2 ns

   Output min/max = 2.5 ns/14 ns

**Table 62: Default and High-Speed SD Interface Timing**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{CLK}$ ($1/T_{CLK}$) | SCLK frequency range | | - | - | 48 | MHz |
| $T_{CLK\_LO}$ | Clock low time | | $0.45/F_{CLK}$ | - | $0.55/F_{CLK}$ | s |
| $T_{CLK\_HI}$ | Clock high time | | $0.45/F_{CLK}$ | - | $0.55/F_{CLK}$ | s |
| $T_{RISE(CLK)}$ | Clock rise time | 10 pF load | - | - | 2 | ns |
| $T_{FALL(CLK)}$ | Clock fall time | 10 pF load | - | - | 1 | ns |
| $T_{SETUP(IN)}$ | Data/CMD input setup time | | 2.0 | - | - | ns |
| $T_{HOLD(IN)}$ | Data/CMD input hold time | | 1.5 | - | - | ns |
| $T_{HOLD(OUT)}$ | Data/CMD output hold time | 10 pF load | -1 | - | - | ns |
| $T_{VALID(OUT)}$ | Data/CMD output valid time | 10 pF load | - | - | 3.2 | ns |
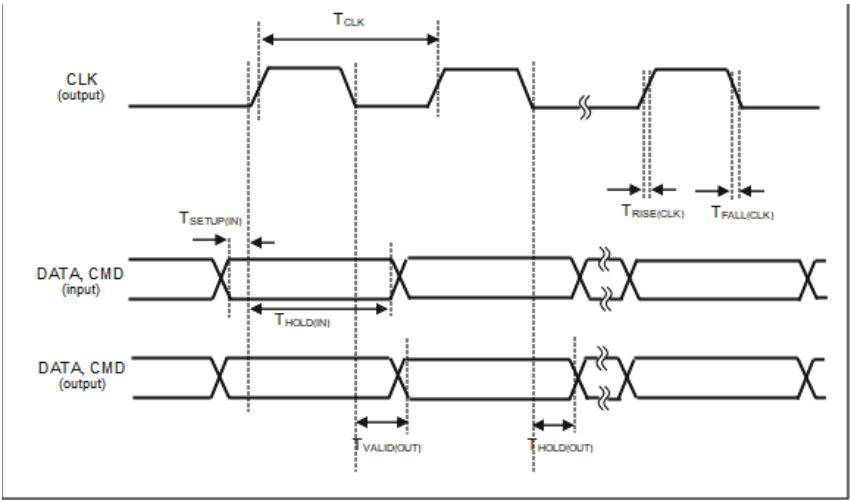
**Figure 75. Default and High-Speed SD Interface Timing Diagram**

### 30.13.2  SD SDR50 Mode Interface

Timing characteristics have been determined by measurements on chip.

The following specifications are with the following settings:

▪ MCUCTRL_SDIOnCTRL_SDIOnOTAPDLYSEL = 4 and MCUCTRL_SDIOnCTRL_SDIOnITAPDLYSEL = 6
▪ A drivestrength of 0.5x for D0/CMD and 1.0x for D1-D7/CLK.

**Table 63: SD SDR50 Mode Interface Timing**

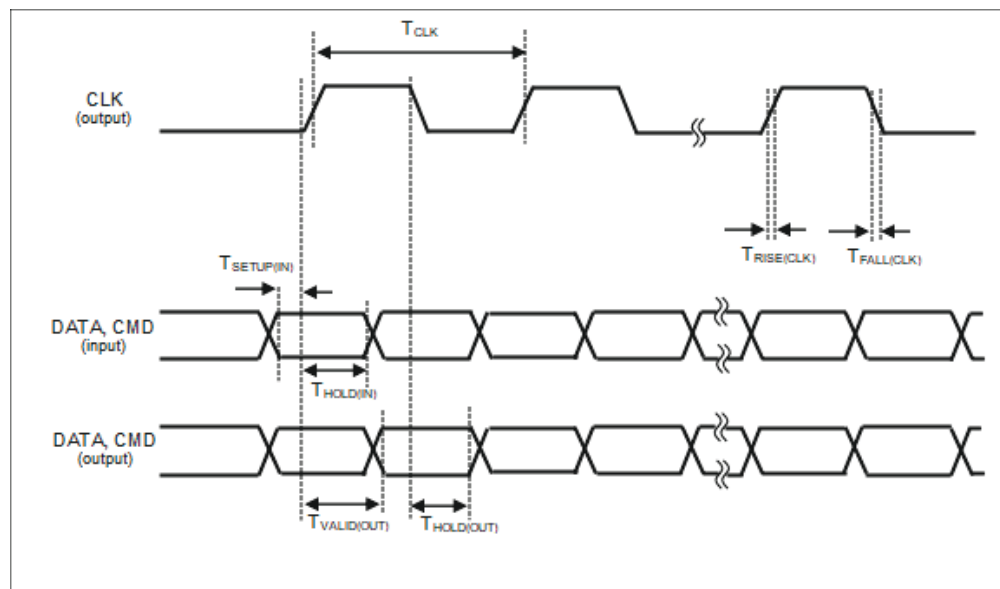| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|--------|-----------|-----------------|-----|-----|-----|------|
| $F_{CLK}$ ($1/T_{CLK}$) | SCLK frequency range | | - | - | 96 | MHz |
| $T_{CLK\_LO}$ | Clock low time | 15 pF Load, DS = See settings | $0.3/F_{CLK}$ | - | $0.6/F_{CLK}$ | s |
| $T_{CLK\_HI}$ | Clock high time | 15 pF Load, DS = See settings | $0.4/F_{CLK}$ | - | $0.7/F_{CLK}$ | s |
| $T_{RISE(CLK)}$ | Clock rise time | 15 pF Load, DS = See settings | - | 1.1 | - | ns |
| $T_{FALL(CLK)}$ | Clock fall time | 15 pF Load, DS = See settings | - | 1.8 | - | ns |
| $T_{SETUP(IN)}$ | Data/CMD input setup time | 15 pF Load, DS = See settings | 2 | - | - | ns |
| $T_{HOLD(IN)}$ | Data/CMD input hold time | 15 pF Load, DS = See settings | 2 | - | - | ns |
| $T_{HOLD(OUT)}$ | Data/CMD output hold time | 15 pF Load, DS = See settings | 2 | - | - | ns |
| $T_{VALID(OUT)}$ | Data/CMD output valid time | 15 pF Load, DS = See settings, $F_{CLK}$ = 96 MHz | - | - | 7 | ns |

Note: Card specifications with respect to clock rise:

- Input setup/hold = 3 ns/0.8 ns
- Output min/max = 1.5 ns/7.5 ns
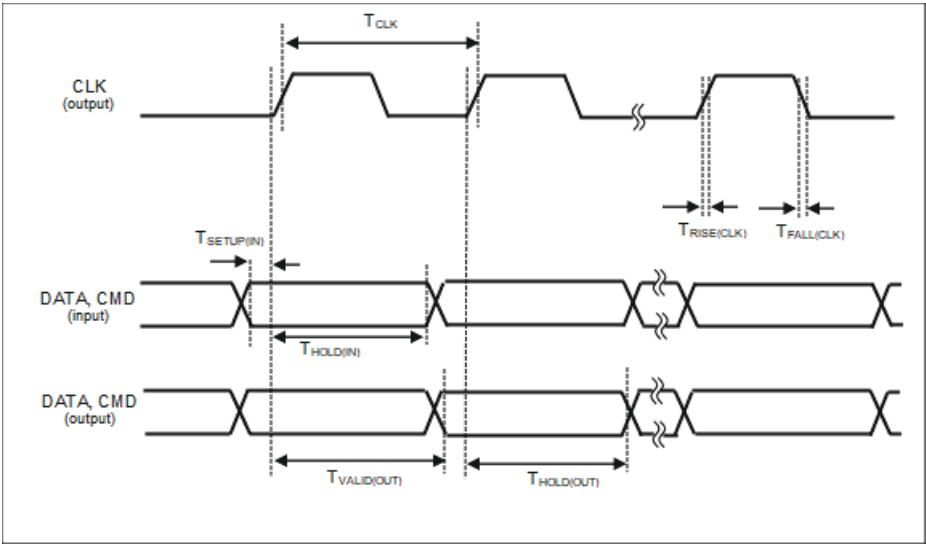
**Figure 76. SD SDR50 Mode Interface Timing Diagram**

### 30.13.3 SD DDR50 Mode Interface

Timing characteristics have been determined by measurements on chip.

The following specifications are with the following settings:

- MCUCTRL_SDIOnCTRL_SDIOnOTAPDLYSEL = 4 and MCUCTRL_SDIOnCTRL_SDIOnITAPDLYSEL = 6
- A drivestrength of 0.5x for D0/CMD and 1.0x for D1-D7/CLK.

**Table 64: SD DDR50 Mode Interface Timing**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|--------|-----------|-----------------|-----|-----|-----|------|
| $F_{CLK}$ (1/$T_{CLK}$) | SCLK frequency range | | - | - | 48 | MHz |
| $T_{CLK\_LO}$ | Clock low time | 15 pF Load, DS = See settings | $0.4/F_{CLK}$ | - | $0.6/F_{CLK}$ | s |
| $T_{CLK\_HI}$ | Clock high time | 15 pF Load, DS = See settings | $0.4/F_{CLK}$ | - | $0.6/F_{CLK}$ | s |
| $T_{RISE(CLK)}$ | Clock rise time | 15 pF Load, DS = See settings | - | 1.1 | - | ns |
| $T_{FALL(CLK)}$ | Clock fall time | 15 pF Load, DS = See settings | - | 1.7 | - | ns |
| $T_{SETUP(IN)}$ | Data/CMD input setup time | 15 pF Load, DS = See settings | 2 | - | - | ns |
| $T_{HOLD(IN)}$ | Data/CMD input hold time | 15 pF Load, DS = See settings | 2 | - | - | ns |
| $T_{HOLD(OUT)}$ | Data/CMD output hold time | 15 pF Load, DS = See settings | 1 | - | - | ns |
| $T_{VALID(OUT)}$ | Data/CMD output valid time | 15 pF Load, DS = See settings, $F_{CLK}$ = 96 MHz | - | - | 7 | ns |



**Figure 77. SD DDR50 Mode Interface Timing Diagram**

## 30.14  Audio

### 30.14.1  I$^2$S Interface

Timing characteristics have been determined through simulations.

**Table 65: I$^2$S Subordinate Interface Timing**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| F$_{SCLK}$ (1/TSCLK) | SCLK frequency range | | - | - | 12.288 | MHz |
| T$_{LOW(SCLK)}$ | Clock low time | | 0.4/F$_{SCLK}$ | - | 0.6/F$_{SCLK}$ | s |
| T$_{HIGH(SCLK)}$ | Clock high time | | 0.4/F$_{SCLK}$ | - | 0.6/F$_{SCLK}$ | s |
| T$_{RISE(SCLK)}$ | Clock rise time | | - | - | 2 | ns |
| T$_{FALL(SCLK)}$ | Clock fall time | | - | - | 2 | ns |
| T$_{SETUP(WSIN)}$ | WS input data setup time | | 10 | - | - | ns |
| T$_{HOLD(WSIN)}$ | WS input data hold time | | 6 | - | - | ns |
| T$_{SETUP(SDIN)}$ | SDIN input data setup time | | 8 | - | - | ns |
| T$_{HOLD(SDIN)}$ | SDIN input data hold time | | 6 | - | - | ns |
| T$_{HOLD(SDOUT)}$ | SDOUT output data hold time | | 3 | - | - | ns |
| T$_{VALID(SDOUT)}$ | SDOUT output data valid time | | - | - | 30 | ns |
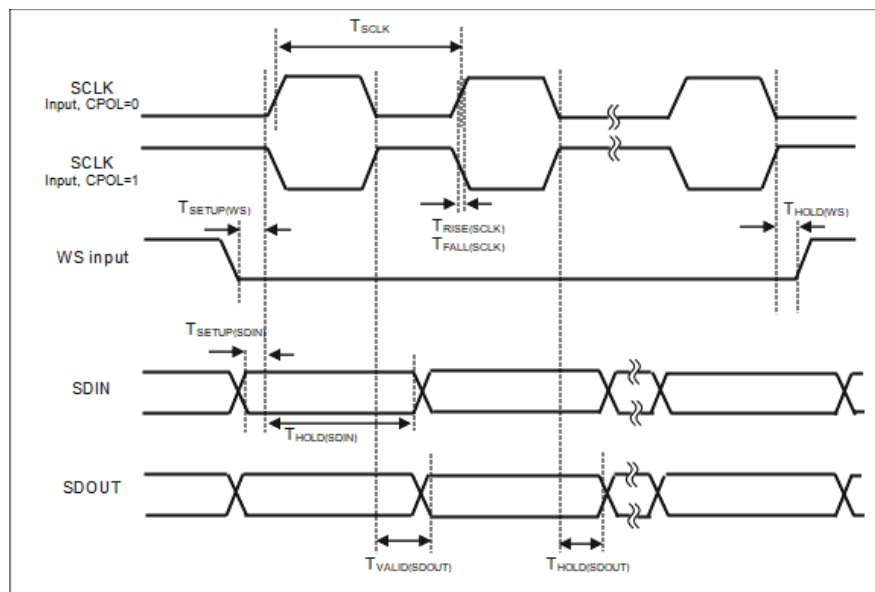


**Figure 78. I$^2$S Subordinate Interface Timing Diagram**

Timing characteristics have been determined through simulations.

**Table 66: I$^2$S Manager Interface Timing**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| F$_{SCLK}$ (1/TSCLK) | SCLK frequency range | | - | - | 12.288 | MHz |
| T$_{LOW(SCLK)}$ | Clock low time | | 0.4/F$_{SCLK}$ | - | 0.6/F$_{SCLK}$ | s |
| T$_{HIGH(SCLK)}$ | Clock high time | | 0.4/F$_{SCLK}$ | - | 0.6/F$_{SCLK}$ | s |
| T$_{RISE(SCLK)}$ | Clock rise time | | - | - | 4 | ns |
| T$_{FALL(SCLK)}$ | Clock fall time | | - | - | 3 | ns |
| T$_{HOLD(WSOUT)}$ | WS output data hold time | | 3 | - | - | ns |
| T$_{VALID(WSOUT)}$ | WS output data valid time | | - | - | 30 | ns |
| T$_{SETUP(SDIN)}$ | SDIN input data setup time | | 8 | - | - | ns |
| T$_{HOLD(SDIN)}$ | SDIN input data hold time | | 6 | - | - | ns |
| T$_{HOLD(SDOUT)}$ | SDOUT output data hold time | | 3 | - | - | ns |
| T$_{VALID(SDOUT)}$ | SDOUT output data valid time | | - | - | 30 | ns |
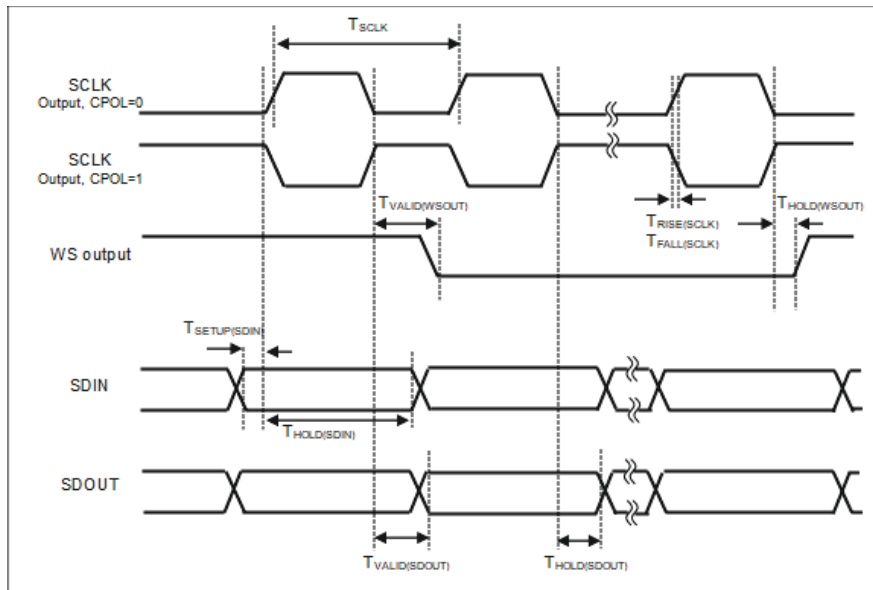


**Figure 79. I$^2$S Manager Interface Timing Diagram**

### 30.14.2 PDM Interface

Timing characteristics have been determined through simulations.

**Table 67: PDM Interface Timing**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{PDM\_CLK}$ ($1/T_{PDM\_CLK}$) | PDM_CLK frequency range | | - | - | 12.288 | MHz |
| $T_{LOW(PDM\_CLK)}$ | Clock low time | | $0.4/F_{SCLK}$ | - | $0.6/F_{SCLK}$ | s |
| $T_{HIGH(PDM\_CLK)}$ | Clock high time | | $0.4/F_{SCLK}$ | - | $0.6/F_{SCLK}$ | s |
| $T_{RISE(PDM\_CLK)}$ | Clock rise time | | - | - | 4 | ns |
| $T_{FALL(PDM\_CLK)}$ | Clock fall time | | - | - | 3 | ns |
| $T_{SETUP(PDM\_DAT)}$ | PDM input data setup time | | 25 | - | - | ns |
| $T_{HOLD(PDM\_DAT)}$ | PDM input data hold time | | 0 | - | - | ns |
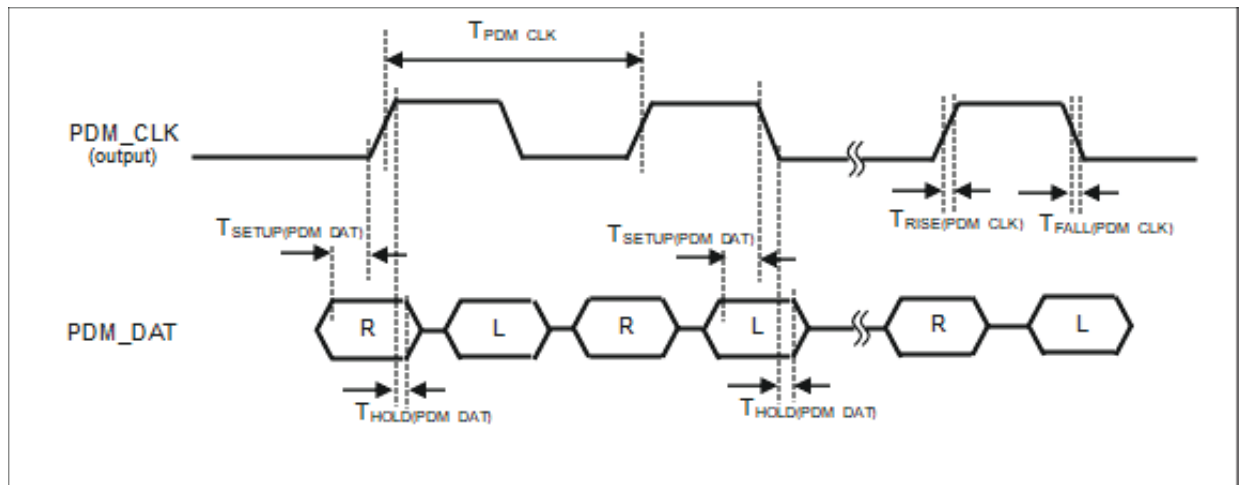


**Figure 80. PDM Interface Timing Diagram**

## 30.15  Serial Wire Debug (SWD)

Timing characteristics have been determined through simulations.

**Table 68: SWD Interface Timing**

| Symbol | Parameter | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{SWCLK}$ ($1/T_{SWCLK}$) | SWCLK frequency range | | - | - | 100 | MHz |
| $T_{LOW(SWCLK)}$ | SWCLK low time | | $0.45/F_{SWCLK}$ | - | $0.55/F_{SWCLK}$ | s |
| $T_{HIGH(CLK)}$ | SWCLK high time | | $0.45/F_{SWCLK}$ | - | $0.55/F_{SWCLK}$ | s |
| $T_{RISE(SWCLK)}$ | SWCLK rise time | | - | - | 2 | ns |
| $T_{FALL(SWCLK)}$ | SWCLK fall time | | - | - | 2 | ns |
| $T_{SETUP(IN)}$ | SWDIO input setup time | | 3 | - | - | ns |
| $T_{HOLD(IN)}$ | SWDIO input hold time | | 3 | - | - | ns |
| $T_{HOLD(OUT)}$ | SWDIO output hold time | | 1 | - | - | ns |
| $T_{VALID(OUT)}$ | SWDIO output valid time[1] | | - | - | 35 | ns |

1. The output valid time is larger than one SWCLK cycle period at max frequency. This means that during SWD read, the SWCLK frequency needs to be adjusted to meet the timing.
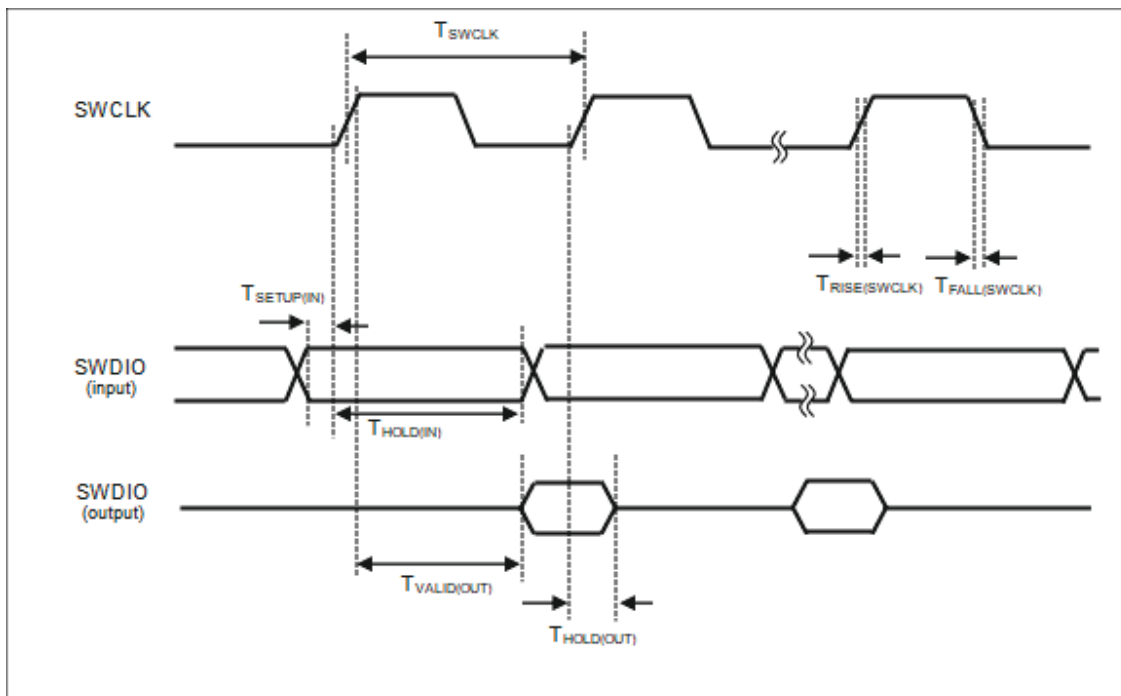


**Figure 81. SWD Interface Timing Diagram**

# 31. Ordering Information

**Table 69: Ordering Information for Apollo510 SoC**

| Device Name | Commercial Temp Range (-20°C to 70°C) | Industrial Temp Range (-40°C to 85°C) | Package Type | GPIOs | NVM (MRAM) | SRAM | Connectivity | Package[1] Size (mm) |
|---|---|---|---|---|---|---|---|---|
| Apollo510 SoC | AP510NFA-CBR | AP510NFA-IBR | BGA | 183 | 4 MB | 3.75 MB | No Connectivity | 6.6 x 6.6 x 0.75(max) 225-pin BGA |
| Apollo510 SoC | AP510NFA-CCR | AP510NFA-ICR | WLCSP | 144 | 4 MB | 3.75 MB | No Connectivity | 4.9 x 4.7 x 0.513(max) 182-pin WLCSP |

1. Packing: Tape and Reel

# 32. Document Revision History

**Table 70: Document Revision List - Apollo510 SoC**

| Revision | Date | Description |
|---|---|---|
| 0.7.3 | June 2024 | Initial release - all sections |
| 0.8.0 | March 2025 | Throughout:<br>- Erratum Notices added throughout the document to alert user of known limited/unexpected device operation.<br>- Names and references to master/slave entities changed to manager/subordinate.<br>- CSP designation changed to WLCSP (wafer-level chip scale package).<br>- Removed all references to FPIO (deprecated functionality).<br>Front Page Feature List and SoC Intro:<br>- Updated Apollo510 SoC High-Level Block Diagram.<br>- Updated display resolution with footnote.<br>- Added section in SoC Intro: "Functional Differences between BGA and WLCSP Packages of the Apollo510 SoC".<br>- Updated Apollo510 Detailed Block Diagram.<br>Memory SS:<br>- Updated Peripheral Memory Map.<br>RSTGEN:<br>- Removed mention of BODH as functionality has been deprecated.<br>MSPI:<br>- Section added - Board/Package Considerations for MSPI Pin Timing.<br>IOM:<br>- Updated IOM Block Diagram.<br>UART:<br>- Note added about requirement to receive data in DMA mode reliably at 3 Mbps.<br>SDIO:<br>- Updated feature bullet to specify that SDIO1 transfers data in 4-bit eMMC mode.<br>DC:<br>- Updated display resolution and added footnote.<br>I2S:<br>- Updated I2S Block Diagram.<br>VREG:<br>- Note added about maximum falling slew rate of a supply.<br>Pincfg:<br>- Updated several missing BGA pin numbers in the Pinlist and Functions Table<br>Package Mechanical Information:<br>- Added note about the need for light sealing for WLCSP package.<br>- Updated WLCSP package information - thickness (mechanical z height) has changed.<br>- Added tape & reel packaging and labeling specifications for BGA and CSP.<br>Electricals:<br>- AMR table updated.<br>- Updated Components for SIMO Buck Converter section.<br>- Updated VDDAUDA conditions of the Power Supply Sequencing table.<br>- Footnote added about max falling slew rate for VDDP/VDDH/VDDA in Power Supply Sequencing table. |
| 0.9.0 | April 2025 | Front Page Feature List and SoC Intro:<br>- Updated lists.<br>- Supply current section added.<br>Memory SS:<br>- Updated OTP section.<br>Electricals:<br>- Current Consumption section/table added. |

**Table 70: Document Revision List - Apollo510 SoC**

| Revision | Date | Description |
|---|---|---|
| 0.9.1 | May 2025 | Display Subsystem:<br>  - Removed support for video mode.<br>Package:<br>  - Corrected WLCSP Tape & Reel drawing. |
| 0.9.2 | August 2025 | Memory SS:<br>  - Added note that MSPI clock on MSPIn_4 configuration is not supported by the Boot ROM.<br>CLKGEN:<br>  - Note added that external load capacitors are required to tune the XTAL oscillator circuit<br>MSPI:<br>  - Added note that MSPI does not support full duplex operation.<br>IOM:<br>  - Added note that IOM does not support full duplex operation.<br>IOSFD:<br>  - Updated sections describing DMA transfers.<br>Electricals:<br>  - Updated VDD18 leakage current spec.<br>  - Corrected voltage rating for SIMO Buck bypass capacitors in "smaller package" option.<br>  - Units for rise/fall times in GPIO table corrected.<br>  - $C_{EXT\_XT\_LOAD}$ updated in Low-Frequency Crystal Oscillator table.<br>  - Timing characteristics have been updated for MSPI, IOM, IOS and SDIO sections. |

# Legal Information and Disclaimers

AMBIQ MICRO INTENDS FOR THE CONTENT CONTAINED IN THE DOCUMENT TO BE ACCURATE AND RELI-
ABLE. THIS CONTENT MAY, HOWEVER, CONTAIN TECHNICAL INACCURACIES, TYPOGRAPHICAL ERRORS
OR OTHER MISTAKES. AMBIQ MICRO MAY MAKE CORRECTIONS OR OTHER CHANGES TO THIS CONTENT
AT ANY TIME. AMBIQ MICRO AND ITS SUPPLIERS RESERVE THE RIGHT TO MAKE CORRECTIONS, MODIFI-
CATIONS, ENHANCEMENTS, IMPROVEMENTS AND OTHER CHANGES TO ITS PRODUCTS, PROGRAMS AND
SERVICES AT ANY TIME OR TO DISCONTINUE ANY PRODUCTS, PROGRAMS, OR SERVICES WITHOUT
NOTICE.

THE CONTENT IN THIS DOCUMENT IS PROVIDED "AS IS". AMBIQ MICRO AND ITS RESPECTIVE SUPPLIERS
MAKE NO REPRESENTATIONS ABOUT THE SUITABILITY OF THIS CONTENT FOR ANY PURPOSE AND DIS-
CLAIM ALL WARRANTIES AND CONDITIONS WITH REGARD TO THIS CONTENT, INCLUDING BUT NOT LIM-
ITED TO, ALL IMPLIED WARRANTIES AND CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR
PURPOSE, TITLE AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHT.

AMBIQ MICRO DOES NOT WARRANT OR REPRESENT THAT ANY LICENSE, EITHER EXPRESS OR IMPLIED, IS
GRANTED UNDER ANY PATENT RIGHT, COPYRIGHT, MASK WORK RIGHT, OR OTHER INTELLECTUAL PROP-
ERTY RIGHT OF AMBIQ MICRO COVERING OR RELATING TO THIS CONTENT OR ANY COMBINATION,
MACHINE, OR PROCESS TO WHICH THIS CONTENT RELATE OR WITH WHICH THIS CONTENT MAY BE USED.

USE OF THE INFORMATION IN THIS DOCUMENT MAY REQUIRE A LICENSE FROM A THIRD PARTY UNDER
THE PATENTS OR OTHER INTELLECTUAL PROPERTY OF THAT THIRD PARTY, OR A LICENSE FROM AMBIQ
MICRO UNDER THE PATENTS OR OTHER INTELLECTUAL PROPERTY OF AMBIQ MICRO.

INFORMATION IN THIS DOCUMENT IS PROVIDED SOLELY TO ENABLE SYSTEM AND SOFTWARE IMPLE-
MENTERS TO USE AMBIQ MICRO PRODUCTS. THERE ARE NO EXPRESS OR IMPLIED COPYRIGHT
LICENSES GRANTED HEREUNDER TO DESIGN OR FABRICATE ANY INTEGRATED CIRCUITS OR INTE-
GRATED CIRCUITS BASED ON THE INFORMATION IN THIS DOCUMENT. AMBIQ MICRO RESERVES THE
RIGHT TO MAKE CHANGES WITHOUT FURTHER NOTICE TO ANY PRODUCTS HEREIN. AMBIQ MICRO MAKES
NO WARRANTY, REPRESENTATION OR GUARANTEE REGARDING THE SUITABILITY OF ITS PRODUCTS FOR
ANY PARTICULAR PURPOSE, NOR DOES AMBIQ MICRO ASSUME ANY LIABILITY ARISING OUT OF THE
APPLICATION OR USE OF ANY PRODUCT OR CIRCUIT, AND SPECIFICALLY DISCLAIMS ANY AND ALL LIABIL-
ITY, INCLUDING WITHOUT LIMITATION CONSEQUENTIAL OR INCIDENTAL DAMAGES. "TYPICAL" PARAME-
TERS WHICH MAY BE PROVIDED IN AMBIQ MICRO DATA SHEETS AND/OR SPECIFICATIONS CAN AND DO
VARY IN DIFFERENT APPLICATIONS AND ACTUAL PERFORMANCE MAY VARY OVER TIME. ALL OPERATING
PARAMETERS, INCLUDING "TYPICALS" MUST BE VALIDATED FOR EACH CUSTOMER APPLICATION BY CUS-
TOMER'S TECHNICAL EXPERTS. AMBIQ MICRO DOES NOT CONVEY ANY LICENSE UNDER NEITHER ITS PAT-
ENT RIGHTS NOR THE RIGHTS OF OTHERS. AMBIQ MICRO PRODUCTS ARE NOT DESIGNED, INTENDED, OR
AUTHORIZED FOR USE AS COMPONENTS IN SYSTEMS INTENDED FOR SURGICAL IMPLANT INTO THE
BODY, OR OTHER APPLICATIONS INTENDED TO SUPPORT OR SUSTAIN LIFE, OR FOR ANY OTHER APPLICA-
TION IN WHICH THE FAILURE OF THE AMBIQ MICRO PRODUCT COULD CREATE A SITUATION WHERE PER-
SONAL INJURY OR DEATH MAY OCCUR. SHOULD BUYER PURCHASE OR USE AMBIQ MICRO PRODUCTS
FOR ANY SUCH UNINTENDED OR UNAUTHORIZED APPLICATION, BUYER SHALL INDEMNIFY AND HOLD
AMBIQ MICRO AND ITS OFFICERS, EMPLOYEES, SUBSIDIARIES, AFFILIATES, AND DISTRIBUTORS HARM-
LESS AGAINST ALL CLAIMS, COSTS, DAMAGES, AND EXPENSES, AND REASONABLE ATTORNEY FEES ARIS-
ING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PERSONAL INJURY OR DEATH ASSOCIATED WITH
SUCH UNINTENDED OR UNAUTHORIZED USE, EVEN IF SUCH CLAIM ALLEGES THAT AMBIQ MICRO WAS
NEGLIGENT REGARDING THE DESIGN OR MANUFACTURE OF THE PART.

**ambiq**