Au5329: 10-Outputs, Dual PLL Frequency Translation, Jitter Attenuator

General Description

The AU5329 is a programmable Dual Fractional Frequency translation based jitter attenuating clock synthesizer with flexible input to output frequency translation options. It supports up to 2 input clocks that are common for all the 2 fractional translations and provides 10 clock outputs. The clock outputs can be derived from the 2 PLLs in a highly flexible manner. It is fully programmable with the I2C / SPI interface or an on chip two time programmable non-volatile memory for factory pre-programmed devices. Using advanced design technology, it provides excellent integrated jitter performance as well as low frequency offset noise performance while working reliably for ambient temperatures from -40 °C to 85 °C. The chip has best in class transient performance features in terms of clock switching transients and repeatable input to output delays

Nomenclature:

AU5329: 2 input, 10 output, 64-QFN 9 mm X 9 mm
AU5329: VDDIN = VDD = 3.3 V or 2.5 V

Applications:

- Carrier Ethernet,
- OTN Equipment,
- Microwave Backhaul,
- Gigabit Ethernet,
- Wireless Infrastructure,
- Network Line Cards,
- Small Cells,
- Data Center/Storage,
- SONET/SDH,
- Test / Instrumentation,
- Broadcast Video

Features

- Flexible dual PLL frequency translation from a common input: fractional output domains from single input
- Fully Integrated Fractional N PLLs with integrated VCO and programmable loop filter (1 mHz to 4 kHz)
- Wide frequency support
  - Differential Output from 8 KHz to 2.1 GHz
  - Single Ended Output from 8 KHz to 250 MHz
  - Support for 1 Hz frequency on one output
  - Differential Input from 8 KHz to 2.1 GHz
  - Single Ended Input from 8 KHz to 250 MHz
  - Multiple Crystals / XO / TCXO / OCXO support
- LVPECL, CML, HCSL, LVDS and LVCMOS Outputs
- 150 fs typical rms integrated jitter performance
- Synchronized, holdover or free run operation modes
- Meets G.8262 EEC Option 1,2(Sync E)
- Hitless input clock switching: Auto or manual
  - Sub 50 ps phase build out mode transients
  - Phase Propagation with programmable slopes
  - Frequency ramp for plesiochronous clocks with programmable slopes
  - Robust and fast cycle slip and frequency step detection for input frequency steps (Clean frequency tracking for large frequency steps)
- Excellent Close-in Phase noise performance with no external discrete VCXOs
- Digitally Controlled Oscillator mode: to 0.005 ppb
- Programmable Output Delay Control
- Programmable Frequency Ramp Slopes for Switching Pleisochronous Clocks
- Indicators: Lock Loss, Clock Loss, Frequency Drift
- Repeatable Input to Output delays for each power up of chip
- Supports PCI Express Gen 1.0 / 2.0 / 3.0 / 4.0 and 5.0 with spread spectrum OFF [SSC = OFF]

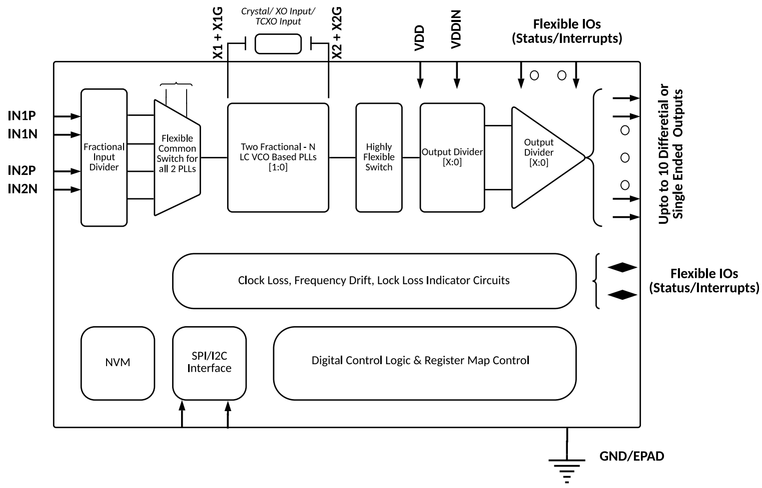


Figure 1 Functional Overview

Table of Contents

List of Tables

List of Figures

1 Detailed Pin Description

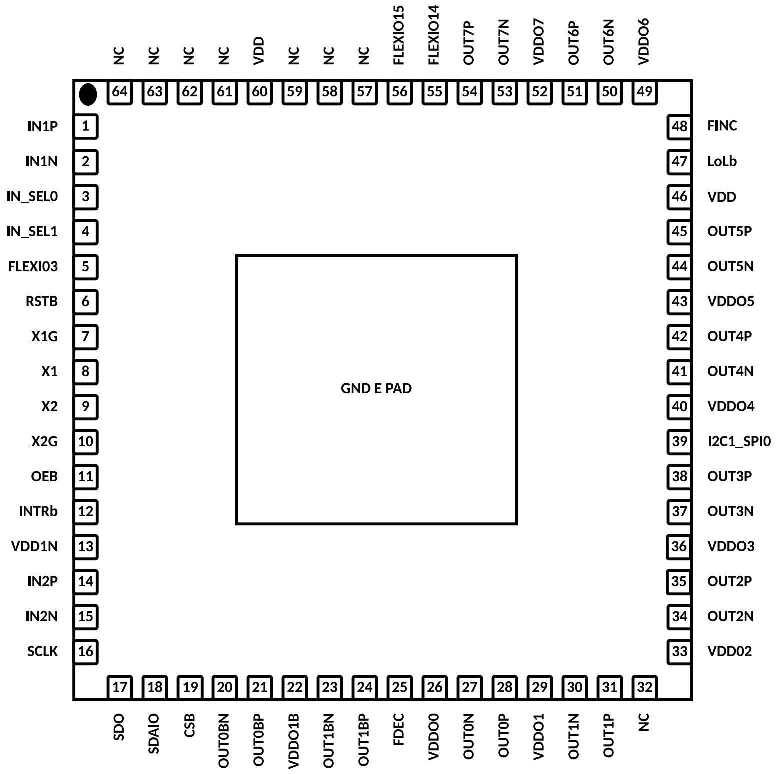


Figure 2 AU5329 Top View

Table 1 Detailed Pin Description

Pin Name	I/O Type	AU5329	Function	Comments
IN1P	Input	1	True input for IN1 differential pair. Input for LVCMOS IN1 input. Need series external capacitor for differential input.	IN1 / IN2 inputs can be used for output clock synchronization. An active clock and three spare clocks are chosen such that the same choice holds for all PLLs.
IN1N	Input	2	Complement input for IN1 differential pair. Ground with capacitor for LVCMOS IN1 input. Need series external capacitor for differential input.	
IN2P	Input	14	True input for IN2 differential pair. Input for LVCMOS IN2 input. Need series external capacitor for differential input.	
IN2N	Input	15	Complement input for IN2 differential pair. Ground with capacitor for LVCMOS IN2 input. Need series external capacitor for differential input.	
GND	Power	EPAD	Electrical and Package Ground	Exposed Ground on the bottom E-PAD
OUT0P	Output	28	Output 0 True Output or Output 0 LVCMOS.	LVPECL, LVDS, HCSL, CML and LVCMOS support.
OUT0N	Output	27	Output 0 Complement Output or Output 0 LVCMOS.	
OUT1P	Output	31	Output 1 True Output or Output 1 LVCMOS.	
OUT1N	Output	30	Output 1 Complement Output or Output 1 LVCMOS.	
OUT2P	Output	35	Output 2 True Output or Output 2 LVCMOS.	
OUT2N	Output	34	Output 2 Complement Output or Output 2 LVCMOS.	
OUT3P	Output	38	Output 3 True Output or Output 3 LVCMOS.	

Pin Name	I/O Type	AU5329	Function	Comments
OUT3N	Output	37	Output 3 Complement Output or Output 3 LVCMOS.	
OUT4P	Output	42	Output 4 True Output or Output 4 LVCMOS.	
OUT4N	Output	41	Output 4 Complement Output or Output 4 LVCMOS.	
OUT5P	Output	45	Output 5 True Output or Output 5 LVCMOS.	
OUT5N	Output	44	Output 5 Complement Output or Output 5 LVCMOS.	
OUT6P	Output	51	Output 6 True Output or Output 6 LVCMOS.	
OUT6N	Output	50	Output 6 Complement Output or Output 6 LVCMOS.	
OUT7P	Output	54	Output 7 True Output or Output 7 LVCMOS.	
OUT7N	Output	53	Output 7 Complement Output or Output 7 LVCMOS.	
OUT0BP	Output	21	Output 0B True Output or Output 0B LVCMOS.	
OUT0BN	Output	20	Output 0B Complement Output or Output 0B LVCMOS.	
OUT1BP	Output	24	Output 1B True Output or Output 1B LVCMOS.	
OUT1BN	Output	23	Output 1B Complement Output or Output 1B LVCMOS.	
VDDIN	Power	13	Power Supply Voltage pin	Decoupling capacitor close to supply pin required.
VDD	Power	46,60	Power Supply Voltage pin	Multiple Supply Pins, Decoupling capacitor close to each supply pin required.
IN_SEL0	Input	3	Input Clock Selection for Manual selection of active clock. Can be left floating or pulled down to GND if not used.	
IN_SEL1	Input	4		
FLEXIO3	Output	5	Flexible Status GPIO. Can be left floating or pulled down to GND if not used.	
RSTB	Input	6	Active low reset internally pulled up to VDDIO; Pull Up Resistor to VDDIO of fixed value (25 KΩ). Can be left floating or pulled up to VDD if not used.	Active low signal performs a complete reset of the part
OEb	Input	11	Used to disable (when 1) all the output clocks. Can be left floating or pulled down to GND if not used	
INTRb	Output	12	Active low indicator of programmable sticky notifies. Can be left floating if not used.	
SCLK	Input	16	I2C Serial Interface Clock or SPI Clock Input. Pull Up Resistor to VDDIO of fixed value (25 KΩ)	
SDO	Output	17	Serial Data Output (SPI Interface). In I2C mode this is the A1 address pin (see I2C section)	
SDAIO	Input/Output	18	I2C Serial Interface Data (SDA) / SPI Input data (SDI)	
CSB	Input	19	Chip Select for the SPI Interface. In I2C mode this is the A0 address pin (see I2C section)	
I2C1_SPI0	Input	39	Choose between SPI (0) and I2C(1) interface being used	

Pin Name	I/O Type	AU5329	Function	Comments
LOLb	Input/Output	47	Loss of Lock Indicator (NOR value of all PLLs' LOL active high indicators comes out on the LOLb pin). Can be left floating if not used.	
FDEC	Input/Output	25	DCO Decrement. Can be left floating or pulled down to GND if not used.	
FINC	Input/Output	48	DCO Increment. Can be left floating or pulled down to GND if not used. Can also be used as SYSREF Trigger to start the SYSREF clock. Please refer AN53006.	
FLEXIO14	Input/Output	55	Flexible Outputs can be used for programmable status monitoring (Refer AN53001 for more information). Can be left floating or pulled down to GND if not used.	
FLEXIO15	Input/Output	56		
{X1, X1G}	Input/Output	8,7	Crystal X1 Pin and accompanying ground pin	{X1G, X2G} land on a floating island on the PCB
{X2, X2G}	Input/Output	9,10	Crystal X2 Pin and accompanying ground pin	
VDDO0	Power	26	Output Power Supply for Bank 0 outputs	Decoupling capacitor close to each supply pin required.
VDDO1	Power	29	Output Power Supply for Bank 1 outputs	
VDDO2	Power	33	Output Power Supply for Bank 2 outputs	
VDDO3	Power	36	Output Power Supply for Bank 3 outputs	
VDDO4	Power	40	Output Power Supply for Bank 4 outputs	
VDDO5	Power	43	Output Power Supply for Bank 5 outputs	
VDDO6	Power	49	Output Power Supply for Bank 6 outputs	
VDDO7	Power	52	Output Power Supply for Bank 7 outputs	
VDDO1B	Power	22	Output Power Supply for Bank 0B and 1B outputs	
NC	No Connect	32, 57, 58, 59, 61, 62, 63, 64	No connect. This pin is not connected to the die.	

Notes:

1. VDDIO is the voltage used for all the status GPIOs and the serial interface. The default voltage for VDDIO can be chosen as either VDDIN or VDD through the programmable GUI.
2. All digital input/output GPIOs (FLEXIOs) have an on-chip 25 kΩ pull down resistor to ePAD ground (unless mentioned otherwise) and can be left unconnected if not used.
3. The I2C1_SPI0 pad has a an on-chip 25 kΩ pull up resistor to indicate default mode of communication as I2C unless this pin is pulled down on the board to indicate the SPI mode.
4. In I2C mode, the serial data and clock have an on-chip 25 kΩ pull up resistor to VDDIO.
5. The RSTB pin has an on-chip 25 kΩ pull up resistor to VDDIO. Writing 0xFE[0] to 1 with delay addition of 10ms has the same effect as the pulling RSTB pin to GND for chip reset.
6. SDO and CSB pins are used to set the I2C default address as 0x69 when floating since SDO and CSB has 25k pull down and pull up to GND and VDDIN respectively. Otherwise the I2C address can be changed as 11010{SDO},{CSB} by forcing the SDO and CSB externally to VDDIN or GND accordingly.
   a. The chip can be reset from the register map by writing address 0xFE as 0x01 using the current I2C address.
   b. To disable reset from register map by writing 0xFE register as 0x00, Address needs to be 0b11010{SDO}{CSB}, 5 MSB address bits are 11010, LSB 2 bits are the state of SDO and CSB pins. If these pins are floating, use 0x69 as the address. At all other times default slave address chosen for the part can be used.

2 Electrical Characteristics

Table 2 Absolute Maximum Ratings

Description	Conditions	Symbol	Min	Typ	Max	Units
Core supply voltage, Analog Input		V_{DDIN}	-0.5		+3.63	V
Core supply voltage, PLL		V_{DD}	-0.5		+3.63	
Output bank supply voltage		V_{DDO}	-0.5		+3.63	V
Input voltage, All Inputs	Relative to GND	V_{IN}	-0.5		+3.63	V
XO Inputs	Relative to GND	V_{XO}	-0.5		+1.4	V
I^2C Bus input voltage	SCLK, SDAT pins	V_{INI2C}	-0.5		+3.63	V
SPI Bus input voltage		V_{INSPI}	-0.5		+3.63	V
Storage temperature	Non-functional, Non-Condensing	T_S	-55		+150	°C
Programming Temperature		T_{PROG}	+25		+85	°C
Maximum Junction Temperature in Operation		T_{JCT}			+125	°C
Programming Voltage (for Programming the OTP Fuse Memory).		V_{PROG}	2.375	2.5	2.625	V
ESD (human body model)	JESD22A-114	ESD_{HBM}			2000	V
Latchup	JEDEC JESD78D	LU			100	mA
MSL	64-QFN	MSL	3			

Notes:

- Exceeding maximum ratings may shorten the useful life of the device.
- Stresses beyond those listed under Absolute Maximum Ratings may cause permanent damage to the device. These are stress ratings only and functional operation of the device at these or at any other conditions beyond those indicated under the DC Electrical Characteristics is not implied. Exposure to Absolute-Maximum-Rated conditions for extended periods may affect device reliability or cause permanent device damage.

Table 3 Operating Temperatures and Thermal Characteristics

Description	Conditions	Symbol	Min	Typ	Max	Units
Ambient temperature		TA	–40	–	+85	°C
Junction temperature		TJ			+125	°C
AU5329 64-QFN Package						
Thermal Resistance Junction to Ambient	Still Air	θJA	25.5			°C/W
	Air Flow 1m/s		20.8			°C/W
	Air Flow 2m/s		19.6			°C/W
Thermal Resistance Junction to Case		θJC	8.70			°C/W
Thermal Resistance Junction to Board		θJB	7.07			°C/W
Thermal Resistance Junction to Top Center		ψJT	0.2			°C/W

Table 4 DC Electrical Characteristics

Description	Conditions	Symbol	Min	Typ	Max	Units
AU5329						
	3.3 V range: ±10%	V_{DDIN}[1]	2.97	3.3	3.63	V

Description	Conditions	Symbol	Min	Typ	Max	Units
Supply voltage, Analog Input Pathways and XTAL Pathways	2.5 V range: ±5%		2.375	2.5	2.625	V
Supply voltage, PLL	3.3 V range: ±10%	V_{DD}[1]	2.97	3.3	3.63	V
	2.5 V range: ±5%		2.375	2.5	2.625	V
Supply Voltage, Output Drivers	1.8 V range: ±5%	V_{DDO}	1.71	1.80	1.89	V
	2.5 V range: ±5%		2.375	2.50	2.625	V
	3.3 V range: ±10%		2.97	3.3	3.63	V
AU5329 (VDDIN = VDD = 3.3 V; VDDO = 1.8 V)						
Total Power Dissipation (2.5V LVDS Outputs @ 156.25M)	2 PLLs, 10 Outputs (2 Independent Fractional Translations)	Pd		1175	1410	mW
	1 PLL, 2 Outputs			390	470	mW
Supply Current, VDDIN	Both Inputs assumed to be enabled	I_{DDIN}[1]		10	12	mA
Supply Current, VDD	Both PLLs and All 10 Outputs enabled (Maximum current mode)	I_{DD}		220	264	mA
Power supply current, VDDO	LVPECL, output pair terminated 50 Ω to V_{TT} (VDDO – 2 V).	I_{DDO}[2,3,4,5,6]		40	48	mA
	LVPECL2, output pair terminated 50 Ω to V_{TT} (VDDO – 2 V) or 0 V without common mode current.			28	36	mA
Power supply current, VDDO	CML, output pair terminated 50 Ω to VDDO	I_{DDO}[2,3,4,5,6,7		20	24	mA
Power supply current, VDDO	HCSL, output pair with HCSL termination	I_{DDO}[2,3,4,5,6,7		27	36	mA
Power supply current, VDDO	LVDS, output pair terminated with an AC or DC Coupled diff 100 Ω	I_{DDO}[2,3,4,5,6,7		16	19.2	mA
Power supply current, VDDO	LVDS Boost, output pair terminated with an AC or DC Coupled diff 100 Ω	I_{DDO}[2,3,4,5,6,7		20	24	mA
Power supply current, VDDO	LVCMOS, 250 MHz, 2.5 V output, 5 pF load	I_{DDO}[2,3,4,5,6,7		15	18	mA

Notes:

1. VDD and VDDIN are independent supplies that are expected to be at the same voltage level (either 3.3 V or 2.5 V) for AU5329. Additional current consumption of 3 mA for a third overtone crystal instead of a fundamental mode crystal.
2. LVPECL and LVDS Boost standards are supported for VDDO = {2.5 V, 3.3 V}. LVPECL2, HCSL, CML and LVDS standards are supported for VDDO = {1.8 V, 2.5 V, 3.3 V}.
3. LVPECL mode provides 6mA of common mode current on each output. LVPECL2 mode does not provide this common mode current.
4. A 50 Ω Termination resistor with a DC bias of VDDO – 2 V for LVPECL standards is supported for VDDO = {2.5 V, 3.3 V}.
5. IDDOx Output driver supply current specified for one output driver in the table. This includes current in each of the output module that includes output dividers, drivers and clock distributions.
6. The LVDS Boost Mode and the LVDS Mode can be used for AC Coupled output terminations. LVDS Boost provides an LVPECL like swing with an AC Coupled 100 Ω Differential termination.
7. Refer to Output Termination Information in the data sheet for the descipription of the various terminations that are supported.
8. For efuse programming in AU5329, VDD alongwith VDDIN can be set to 2.5 V and has no reliability concerns. Refer to Programming the Primary E-Fuse section for VDD/VDDIN voltage information for efuse programming.

Table 5 Input Clock Characteristics

Parameter	Test Condition	Symbol	Min	Typ	Max	Unit
Standard Input Buffer with Differential or Single-Ended — AC-coupled (IN1/IN1, IN2/IN2)						
Input Frequency Range	Differential	f_{IN}	0.008	—	2100	MHz
	All Single-ended signals (including LVCMOS)		0.008	—	250	MHz
Voltage Swing (Differential Amplitude Peak or Single Ended Peak to Peak for the differential signal) [1]	AC-coupled fIN < 400 MHz	V_{IN}	100	—		mV
	400 MHz< AC-coupled fIN < 750 MHz		225	—		mV
	750 MHz < AC-coupled fIN < 2100 MHz		350	—		mV
Single Ended AC Coupled Inputs (Single Ended Peak to Peak Input) [1][4]	AC-Coupled fIN < 250 MHz		500	–	3600	mV
Slew Rate [2,3]		SR	400	—	—	V/µs
Duty Cycle		DC	40	—	60	%
Input Capacitance		C_{IN}	—	0.3	—	pF
Input Resistance	AC Coupled SE	R_{IN}	—	15	—	kΩ
	Differential		—	10	—	kΩ
Pulsed CMOS Input Buffer — DC-coupled (IN1, IN2) [3]						
Input Frequency		$f_{IN\_PULSED\_CMOS}$	0.008	—	250	MHz
Input Voltage		V_{IL}	–0.2	—	0.4	V
		V_{IH}	0.8	—	—	V
Slew Rate [2,3]		SR	400	—	—	V/µs
Duty Cycle		DC	40	—	60	%
Minimum Pulse Width	Pulse Input	PW	1.6	—	—	ns
Input Resistance		R_{IN}	—	30	—	kΩ
Reference Clock (Applied to X1), Can be external XO, TCXO or OCXO						
Reference Clock Frequency	Range for best jitter	F_{IN_REF}	48	-	160	MHz
	Overall supported range		37.5	-	160	MHz
Input Voltage Swing	Single Ended peak to peak	V_{IN_SE}	365	-	2000	mVpp_se
	Differential peak to peak	V_{IN_DIFF}	365	-	2500	mVpp_diff
Slew rate		SR	400	-	-	V/us
Duty Cycle		DC	40	-	60	%

Notes:

1. AC Coupled input assumed with series capacitance for differential inputs or single ended AC Coupled inputs. Swing requirement at device pins.
2. Resistor termination for differential input followed by series capacitors for each of true and complement differential input connecting to the device pins.
3. LVCMOS single ended is direct coupled on the true input. Connect complement input to ground with a 100 nF capacitor.
4. Single Ended AC coupled Input Swing requirement (Single Ended Peak to Peak Input) [1][4] is for optimal noise performance.

Table 6 Serial and Control Input

Parameter	Test Condition	Symbol	Min	Typ	Max	Unit
Input Voltage		V_{IL}	—	—	0.3 x V_{DDIO} [1]	V
		V_{IH}	0.7 x V_{DDIO} [1]	—	—	V
Input Capacitance		C_{IN}	—	1	—	pF
Input Resistance		R_{IN}	—	25	—	kΩ
Minimum Pulse Width	FINC, FDEC	PW	100	—	—	ns
Update Rate	FINC, FDEC	F_{UR}	—	—	1	µs

Notes:

1. VDDIO is the voltage used for all the status GPIOs and the serial interface. The default voltage for VDDIO can be chosen as either VDDIN or VDD with a hard coded eFuse based selection.

Table 7 Output Serial and Status Pin

Parameter	Test Condition	Symbol	Min	Typ	Max	Unit
All VDDIO[1] Based GPIOs						
Output Voltage	I_{OH} = –2 mA	V_{OH}	V_{DDIO} x 0.75	—	—	V
	I_{OL} = 2 mA	V_{OL}	—	—	V_{DDIO} x 0.25	V

Notes:

1. VDDIO is the voltage used for all the status GPIOs and the serial interface. The default voltage for VDDIO can be chosen as either VDDIN or VDD with a hard coded eFuse based selection.

Table 8 Output Clock Characteristics

Description	Conditions	Symbol	Min	Typ	Max	Units
Differential output frequency	LVPECL, CML, LVDS outputs	$F_{OUT,DIFF}$ [1]	1		2100M	Hz
Differential output frequency	HCSL outputs	$F_{OUT,DIFFH}$ [1]	1		700 M	Hz
Single ended output frequency	LVCMOS outputs	$F_{OUN,SE}$ [1]	1		250 M	Hz
PLL loop bandwidth	Programmable	F_{BW}	0.001		4000	Hz
Jitter peaking	Meets SONET Jitter Peaking requirements in closed loop	J_{PEAK}			0.1	dB
Time delay before the Historical average for output Frequency is considered.	Programmable in register map	H_{DELAY} [2,3]	0.035	0.5	35	s
Length of time for which the Average of the frequency is considered	Programmable in register map	H_{AVG} [2,3]	0.07	1	70	s
Power Supply to I2C or SPI interface ready	No I2C transaction valid till 10ms after all power supplies are ramped to 90% of final value.	T_{START}			10	ms
With Speed-Up mode enabled	Speed-up mode is programmable. This is a Typical number. Actual wake up time depends on fast lock and normal BW settings	T_{LOCK} [4]		300		ms

Description	Conditions	Symbol	Min	Typ	Max	Units
DCO Mode Frequency Step Resolution	Frequency Increment or Decrement resolution. This is controlled through the register map.	$F_{RES,DCO}$ [5]		0.005		ppb
Resolution for output delay	Programmable per output clock with this resolution for a total delay of ±7.5 ns	T_{RES} [6]		35		ps
Maximum Phase Hit	Default Hitless Switching Mode (no phase propagation)	T_{MAX} [7]	-50		50	ps
Uncertainty in Input to Output Delay	Maximum variation in the static delay from input to output clock between repeated power ups of the chip	ΔT_{DELAY}	-175		175	ps
Pull Range		ω_P		500		ppm
POR to Serial Interface Ready		T_{RDY}			15	ms
One free run PLL clock on fuse locked parts	Using a special mode for fuse locked parts to generate one free run output from one PLL	$T_{START,Special}$			10	ms

Notes:

1. 1 Hz Output Available only on output OUT0B (OUT0BP, OUT0BN). Range supported is 8 kHz to 2100 MHz for all the other outputs.
2. Hitless Switching enables PLL to switch between input clocks when the current clock is lost,
   a. Clock Loss can be defined as 2 / 4 / 8 / 16 consecutive missing pulses.
   b. Priority list for the input clocks can be set in the register map independently for each PLL.
   c. Output is truly hitless (no phase transient and 0 ppb relative error in frequency) for exactly same frequency input clocks that are switched.
   d. Hitless switching support is both revertive and non-revertive
   e. Revertive / Non-revertive Support: Assume Clock Input 0 is lost and switch is made to Clock Input 1. Then, PLL reverts to Clock Input 0 when it becomes valid again in Revertive mode. It does not switch back to Clock Input 0 even when it becomes valid again in the non-Revertive mode.
3. PLL enters holdover mode when the active input clock and all spare clocks in the clock priority list for hitless switching are lost,
   a. Clock Loss can be defined as 2 / 4 / 8 / 16 consecutive missing pulses
   b. Entering hold over mode is supported with the frequency frozen at a historical average determined from the HDELAY and HAVG settings.
4. For low PLL Loop Bandwidths, wake up time can be very large unless the speed up feature is used. The speed up feature provides the user options to use a completely independent loop bandwidth for the wake up transitioning to the regular bandwidth after frequency and phase are locked.
   a. Fast Lock Bandwidth needs to be less than 100 times smaller than the input clock frequency (divided input at PLL phase detector) for stable and bounded (in time) lock trajectory of the PLL
5. The 0.005 ppb specification is for the smallest frequency step resolution available. Larger frequency step resolutions up to 100 ppm can be used also. The frequency resolution for the DCO mode frequency step is independently programmable for each DCO step.
6. All output clocks from one specific PLL are phase aligned. Relative delay adjustment is then possible on each clock individually as defined by the TRES parameter.
7. This test is for 2 inputs at 8 M that are switched to get a 622.08 M output.

Table 9 Fault Monitoring Indicators

Description	Conditions	Symbol	Min	Typ	Max	Units
Clock Loss Indicator Thresholds	Clock Loss Indicators can be set on any of the two inputs. Loss of 2 / 4 / 8 / 16 consecutive pulses can be used to indicate a clock loss. Programmable in the register map.	CL_X [1,4]	2	4	16	Pulses
Fine Frequency Drift Indicator Thresholds: Step Size	Frequency drift threshold is programmable in the range with the step size resolution specified. Frequency drift hysteresis is programmable in the range with the step size resolution specified.	FD_X [2,3,4]		±2		ppm
Fine Frequency Drift Indicator Thresholds: Hysteresis Range			±2		±500	ppm
Fine Frequency Drift Indicator Thresholds: Range			±2		±500	ppm
Coarse Frequency Drift Indicator Thresholds	Coarse Drift Indicators programmable from {Up to ±1600 ppm in steps of ±100 ppm}		±100		±1600	ppm
Lock Loss Indicator Threshold	Lock Loss Indicator threshold is programmable in the range specified from the following choices for setting and clearing LL: {±0.2, ±0.4} ppm, {±2, ±4} ppm, {±20, ±40} ppm, {±200, ±400} ppm, {±2000, ±4000} ppm	LL	±0.2		±4000	ppm

Notes:
1. Clock Loss Indicators are used for:
   a. Hitless Switching Triggers
   b. Update in Status Registers in the register map
2. Frequency Drift Indicators can use any one of the two inputs or the Crystal / Reference input as the golden reference with respect to which FDx for all other clocks can be recorded in the Status Registers. FDx thresholds for each clock input for each clock can be set independently.
3. Coarse and Fine Frequency Drift indicators can be concurrently enabled. This enables the user to detect fast drifting frequencies since detecting fine drifts will take longer measurements.
4. Clock loss and Lock loss indicators are available as alerts on flexible IO pins as described in the functional description section of the data sheet.
5. Clock Loss can be combined with either of the frequency drift monitors (coarse and fine) to trigger the hitless switching event in the PLLs. The trigger for a hitless switching event in the PLL can therefore be either the Clock Loss event or either of Clock Loss or Frequency Drift.

Table 10 Crystal Requirements

Description	Conditions	Symbol	Min	Typ	Max	Units
High Fundamental Frequency Crystal Reference (HFF)						
Crystal Frequency	Can be supported with a fundamental crystal of 100-160 MHz range.	$XTAL_{IN}$	100		160	MHz
C0 cap for crystal		$XTAL_{C0}$			2	pF
CL cap for crystal	Small range around CL only	$XTAL_{CL}$		5		pF

Description	Conditions	Symbol	Min	Typ	Max	Units
ESR for crystal	ESR defined at frequency of oscillation	$XTAL_{ESR}$			40	Ω
Rm1 for crystal		$XTAL_{Rm1}$			20	Ω
Power delivered to crystal	Drive Level to the crystal	$XTAL_{PWR}$		100		µW
Third Overtone Crystal Reference (OT3)						
Crystal Frequency	Can be supported with an OT3 crystal of 100-160 MHz range.	$XTAL_{IN}$	100		160	MHz
C0 cap for crystal		$XTAL_{C0}$			2	pF
CL cap for crystal	Small range around CL only	$XTAL_{CL}$		5		pF
ESR for crystal	ESR defined at frequency of oscillation	$XTAL_{ESR}$[1]			80	Ω
Rm3 for crystal		$XTAL_{Rm3}$			40	Ω
Power delivered to crystal	Drive Level to the crystal	$XTAL_{PWR}$		100		µW
Low Frequency Fundamental Crystal (LFF)						
Crystal Frequency	Can be supported with a fundamental crystal > 37.5 MHz range. For Best Performance use an LFF crystal > 48 MHz	$XTAL_{IN}$	48		54	MHz
C0 cap for crystal		$XTAL_{C0}$			2	pF
CL cap for crystal	Small range around CL only	$XTAL_{CL}$		8		pF
ESR for crystal	ESR defined at frequency of oscillation	$XTAL_{ESR}$ [1]			60	Ω
Rm1 for crystal		$XTAL_{Rm1}$			40	Ω
Power delivered to crystal	Drive Level to the crystal	$XTAL_{PWR}$		100		µW

Notes:
1. ESR relates to the motional resistance Rm with the relationship $ESR = Rm\ (1 + C0/CL)^2$

Table 11 Output RMS Jitter in Frequency Translation Modes

Description	Conditions	Symbol	Min	Typ	Max	Units
RMS Jitter for 12 kHz-20 MHz Integration Bandwidth FIN = 38.88 MHz, PLL BW = 100 Hz, Single PLL Profile	F_{OUT} = 622.08 MHz,	RMS_{JIT}[1,2]		140		fs rms
	F_{OUT} = 156.25 MHz,			150		fs rms

Notes:
1. For best noise performance in jitter attenuation mode, use lowest usable loop bandwidth for the PLL.
2. Does not include noise from the input clocks to the PLL

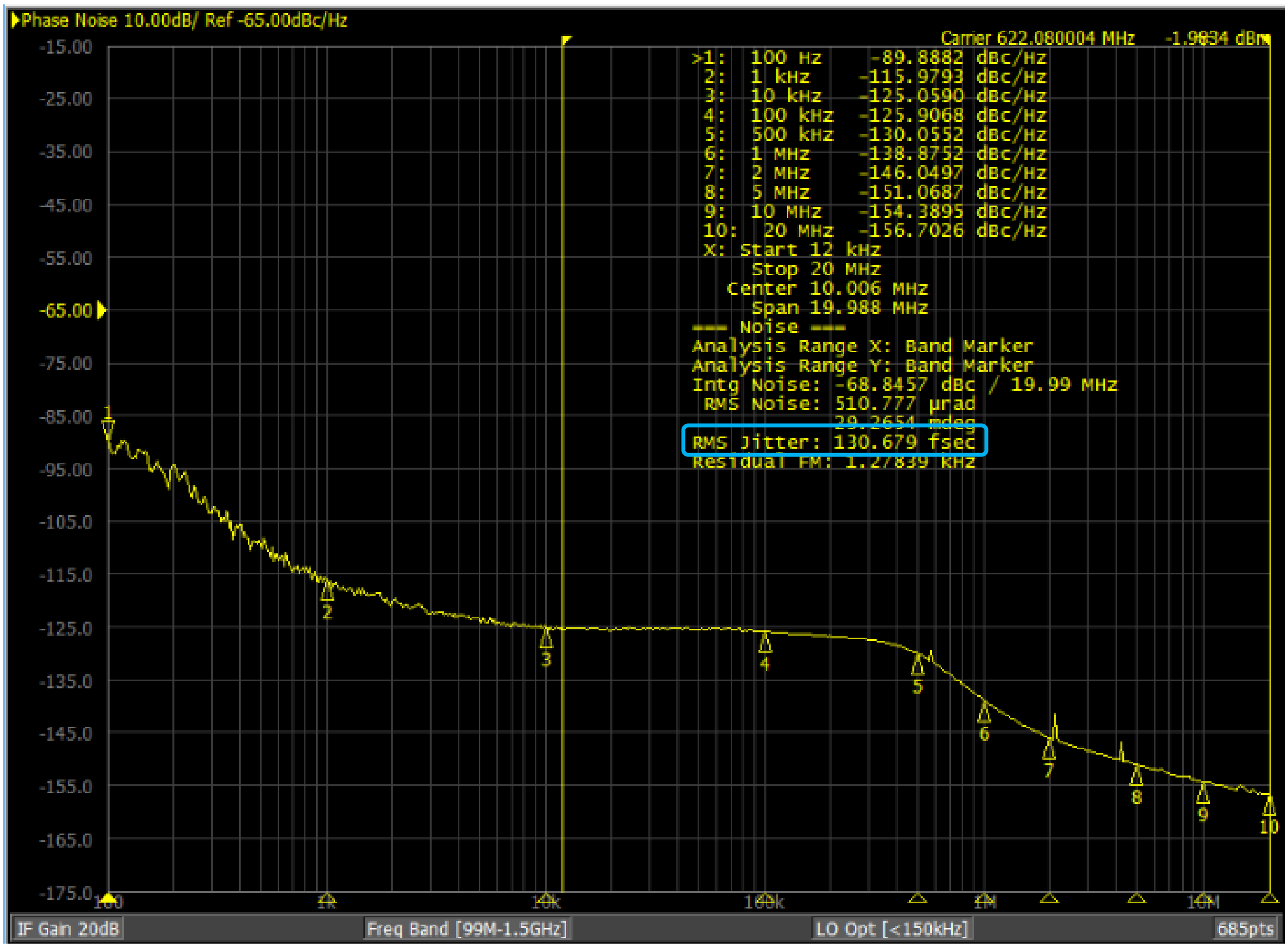


Figure 3 Representative Phase Noise Measurement

Note: FOUT = 622.08 MHz, FIN = 38.88 MHz, BW = 100 Hz, FREF = 54M XO

Table 12 PCI Express Jitter Performance (Spread Spectrum = OFF)

Parameter	Condition	Symbol	Min	Typ	Max	Limit	Units
PCIe Phase Jitter (Commom Clocked Architectures)	PCIe Gen1(2.5 GT/s) [1][2] SSC = OFF	$t_{jphPCIeG1-CC}$		7.1	10	86	ps (p-p)
	PCIe Gen2 (5.0 GT/s) [1][2] SSC = OFF	$t_{jphPCIeG2-CC}$		0.62	1.0	3.1	ps (RMS)
	PCIe Gen3 (8.0 GT/s) [1][2] SSC = OFF	$t_{jphPCIeG3-CC}$		0.2	0.3	1	ps (RMS)
	PCIe Gen4 (16.0 GT/s) [1][2][3][6] SSC = OFF	$t_{jphPCIeG4-CC}$		0.24	0.3	0.5	ps (RMS)
	PCIe Gen5 (32.0 GT/s) [1][2][3][7] SSC = OFF	$t_{jphPCIeG5-CC}$		0.08	0.1	0.15	ps (RMS)

Notes:

1. Refclk Jitter is measured after applying the filter functions found in PCI Express Base Specification 5.0, Rev 1.0.
2. RMS Jitter Measurements were made using DSA90804A for minimum waveform length of ≥ 100k cycles with a minimum sampling rate of ≥ 40GSa/s with the waveform covering 90% of the DSO screen. All the post processing the DSO is disabled to decrease the additonal jitter impact from oscilloscope. Broadband oscilloscope noise is also minimized in the measurement.

3. For the 16.0 GT/s and 32.0 GT/s measurements SSC spurs from the fundamnetal and harmonics are removed up to a cut off frequency of 2MHz taking care to minimize the removal of any non-SSC content.
4. AU5329 is configured as 100MHz HCSL Output Driver [VDDO = 3.3V] and fed to the channels of DSA90804A using the exact measurement set up [Refer Note 8]
5. AU5329 does not support support Spread Spectrum(SS),all the measurements are with SS off.
6. Note that 0.7ps RMS is to be used in channel simulations to account for additional noise in a real system.
7. Note that 0.25ps RMS is to be used in channel simulations to account for additional noise in a real system.
8. AU5329 PCI Express RMS Jitter Measurment Set Up Configuration.

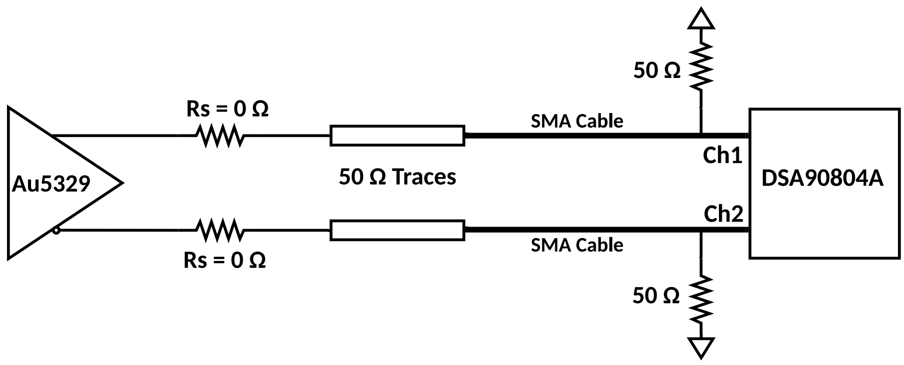


Table 13 Close In Offset Phase Noise

Description	Conditions	Symbol	Min	Typ	Max	Units
Phase Noise Skirt F_{OUT} = 122.88 MHz, PLL BW = 100 Hz	Offset Frequency = 100 Hz	PN[1]		-113		dBc/Hz
	Offset Frequency = 1 kHz			-130		
	Offset Frequency = 10 kHz			-138		

Notes:

1. This is the noise contribution of the chip only without including the input and reference self contributions

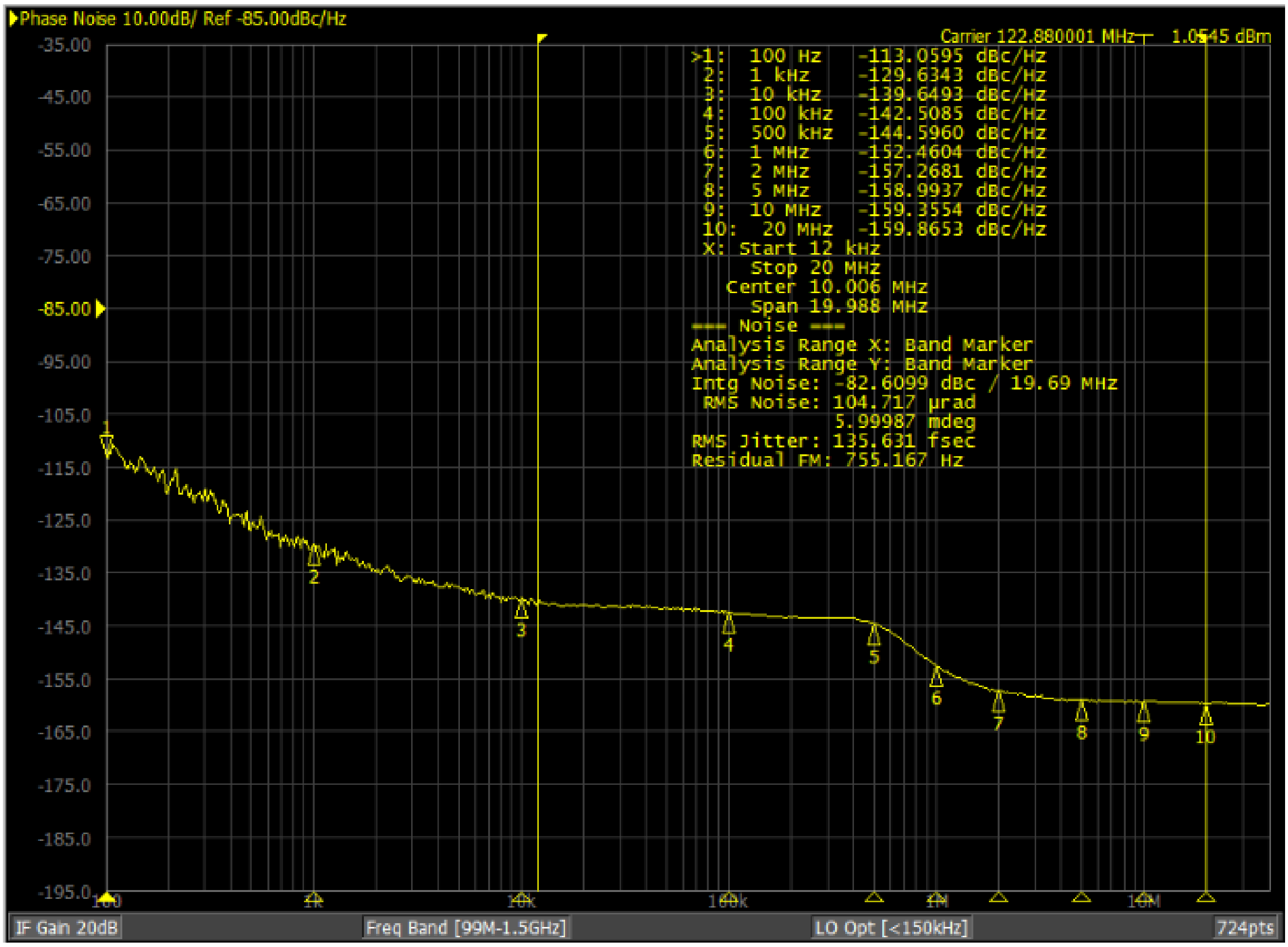


Figure 4 Representative Close In Phase Noise Measurement

Note: FOUT = 122.88M, BW = 100 Hz, FREF and FIN are provided from R&S SMA100 equipment to ensure a low close in phase noise for the reference and input to illustrate the chip contribution to close in phase noise.

Table 14 Power Supply Rejection

Description	Conditions	Symbol	Min	Typ	Max	Units
F_{OUT} = 156.25 MHz, F_{SPUR} = 100 kHz, BW = 100 Hz PSRR on VDD Supply	VDD = 3.3 V	$PSRR_{VDD}$		-85		dBc
	VDD = 2.5 V			-85		
F_{OUT} = 156.25 MHz, F_{SPUR} = 100 kHz, BW = 100 Hz PSRR on VDDIN Supply	VDDIN = 3.3 V	$PSRR_{VDDIN}$		-100		dBc
	VDDIN = 2.5 V			-100		
F_{OUT} = 156.25 MHz, F_{SPUR} = 100 kHz, BW = 100 Hz PSRR on VDDO Supply	VDDO = 3.3 V	$PSRR_{VDDO}$		-80		dBc

Notes:

1. The PSRR is measured with a 50 mVpp sinusoid in series with the supply and checking the spurious level relative to the carrier on the output in terms of phase disturbance impact.
2. Output PSRR measured with LVDS standard which (along with the LVDS boost) are the recommended standards for AC Coupled terminations

Table 15 Adjacent Output Cross Talk

Description	Conditions	Symbol	Min	Typ	Max	Units
156.25 M and 155.52 M on adjacent outputs		XTALK		-75		dBc

Notes:

- Measured across adjacent outputs- All adjacent outputs are covered and the typical value for the worst case output to output coupling is reported.
- The adjacent output pairs are chosen at 155.52 MHz and 156.25 MHz frequencies.
- This cross talk between outputs is mainly package dependent therefore terminated outputs are used for reporting these numbers ensuring that there is signal current in the bond wires.

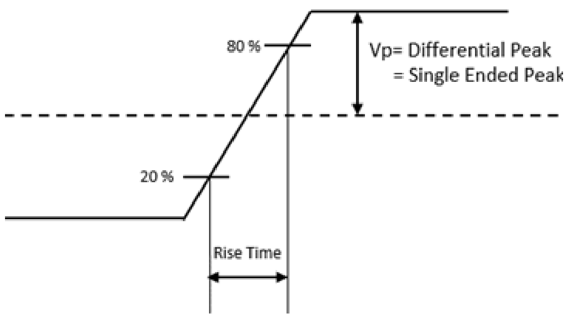
Table 16 Output Clock Specifications

Description	Conditions	Symbol	Min	Typ	Max	Units
DC Electrical Specifications - LVCMOS output (Complementary Out of Phase Outputs or One CMOS Output per Output Driver)						
Output High Voltage	4 mA load, VDDO = 3.3 V	V_{OH}	VDDO - 0.3		-	V
Output High Voltage	4 mA load, VDDO = 1.8 V and 2.5 V	V_{OH}	VDDO - 0.4		-	V
Output Low Voltage	4 mA load	V_{OL}			0.3	V
DC Electrical Specifications - LVCMOS output (In Phase Outputs)						
Output High Voltage	4 mA load, VDDO = 3.3 V	V_{OH}	VDDO - 0.35		-	V
Output High Voltage	4 mA load, VDDO = 2.5 V	V_{OH}	VDDO - 0.45		-	V
Output High Voltage	4 mA load, VDDO = 1.8 V	V_{OH}	VDDO - 0.5		-	V
DC Electrical Specifications – LVDS Outputs (VDDO = 1.8 V, 2.5 V or 3.3 V range)						
Output Common-Mode Voltage	VDDO = 2.5 V or 3.3 V range	V_{OCM}	1.125	1.2	1.375	V
Change in VOCM between complementary output states		ΔV_{OCM}			50	mV
Output Leakage Current	Output Off, VOUT = 0.75 V to 1.75 V	I_{OZ}	-20		20	µA
DC Electrical Specifications - LVPECL Outputs (VDDO = 2.5 V or 3.3 V range)						
Output High Voltage	Rterm = 50 Ω to VTT(VDDO – 2.0 V)	V_{OH}	VDDO-1.165		VDDO – 0.800	V
Output Low Voltage	Rterm = 50 Ω to VTT(VDDO – 2.0 V), w/o common mode current	V_{OL}	VDDO-2.0		VDDO –1.45	
AC Electrical Specifications - HCSL Outputs (VDDO = 1.8 V, 2.5 V or 3.3 V range)						
Output High Voltage Max	Measurement on single-ended signal	V_{MAX}			1150	mV
Output Low Voltage Min	Measurement on single-ended signal	V_{MIN}	-300			mV
Differential Voltage	Measurement taken from differential waveform	V_P	300			mV
Absolute Crossing point voltage	Measurement taken from single ended waveform	V_{CROSS}	250		600	mV
Variation of VCROSS over all rising clock edges	Measurement taken from single ended waveform	$V_{CROSS\ DELTA}$			140	mV
DC Electrical Specifications - CML Outputs (VDDO = 1.8 V, 2.5 V or 3.3 V range)						
Output High Voltage	Rterm = 50 Ω to VDDO	V_{OH}	VDDO-0.085	VDDO – 0.01	VDDO	V
Output Low Voltage	Rterm = 50 Ω to VDDO	V_{OL}	VDDO-0.6	VDDO –0.4	VDDO – 0.3	V
AC Electrical Specifications LVCMOS Output Load: 10 pF < 100 MHz, 7.5 pF < 150 MHz, 5 pF > 150 MHz > 200 MHz						
Output Frequency		f_{OUT}	8k		250M	Hz
Output Duty cycle	Measured at 1/2 VDDO, loaded, fOUT < 100 MHz	t_{DC}	45		55	%

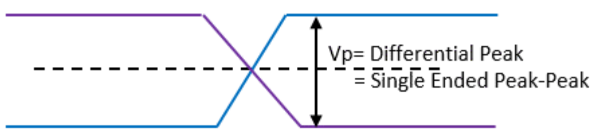
Description	Conditions	Symbol	Min	Typ	Max	Units
Output Duty cycle	Measured at 1/2 VDDO, loaded, fOUT > 100 MHz	t_{DC}	40		60	%
Rise/Fall time	VDDO = 1.8 V, 20-80%, Highest Drive setting	t_{RFCMOS}			2	ns
Rise/Fall time	VDDO = 2.5 V, 20-80%, Highest Drive setting	t_{RFCMOS}			1.5	ns
Rise/Fall time	VDDO = 3.3 V, 20-80%, Highest Drive setting	t_{RFCMOS}			1.2	ns
AC Electrical Specifications (LVPECL, LVDS, CML)						
Clock Output Frequency		f_{OUT}	8k		2100M	Hz
PECL Output Rise/Fall Time	20% to 80% of AC levels. Measured at 156.25 MHz for PECL outputs.	t_{RF}			350	ps
CML Output Rise/Fall Time	20% to 80% of AC levels. Measured at 156.25 MHz for CML outputs	t_{RF}			350	ps
LVDS Output Rise/Fall Time	20% to 80% of AC levels. Measured at 156.25 MHz for LVDS outputs.	t_{RF}			350	ps
Output Duty Cycle	Measured at differential 50% level, 156.25 MHz	t_{ODC}	45	50	55	%
LVDS Output differential peak	Measured at 156.25 M Output	V_P	247	350	454	mV
Boosted LVDS Output differential peak	Measured at 156.25 M Output	V_P	500	700		mV
LVPECL Output Differential peak	Measured at 156.25 M Output	V_P	450	750		mV
CML Output Differential Peak	Measured at 156.25 M Output	V_P	250		600	mV

Notes: Convention for Wave Forms

Differential Signal = OUTP - OUTN



Single Ended Signals {OUTP,OUTN}



3 Functional Description

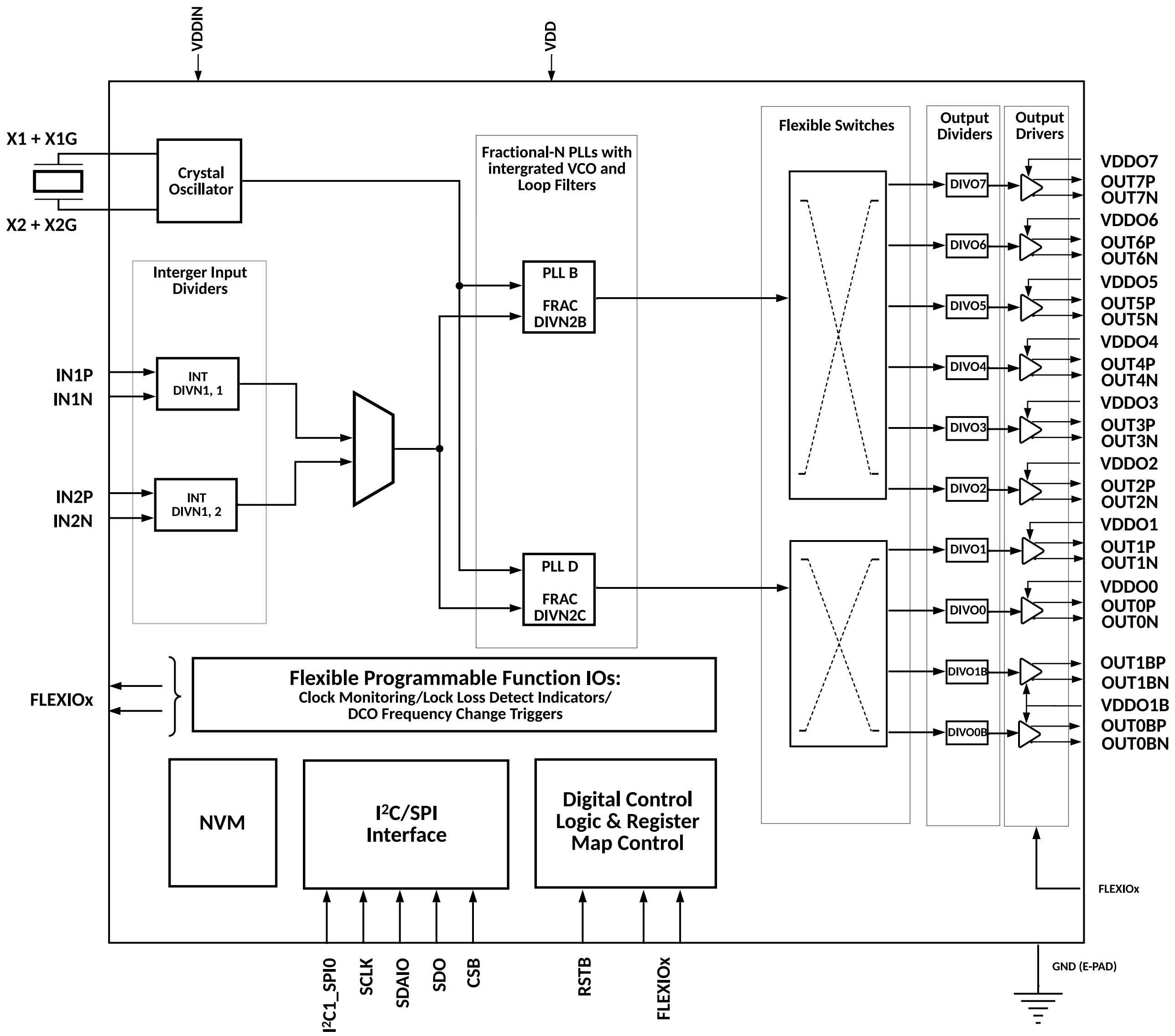


Figure 5 AU5329 Overall Architecture

AU5329 is a jitter attenuating frequency translation device family that offers two PLLs for 2 frequency translation pathways from the same input. The two clock inputs map to the two PLLs such that clock priority is the same across the 2 PLLs. This creates an arrangement that provides up to 2 fractional translations from one input at any given time. The output high frequency voltage controlled oscillators (VCOs) associated with each PLL are mapped to the 10 outputs in a very flexible fashion. This offers a very flexible frequency translation arrangement with independent control of each PLL in terms of jitter attenuation, bandwidth control and input clock selection with redundancy.

The hierarchy of the clocks, nomenclature of the various frequency dividers as well as the clock translation pathways available on the chip are shown in Figure 5

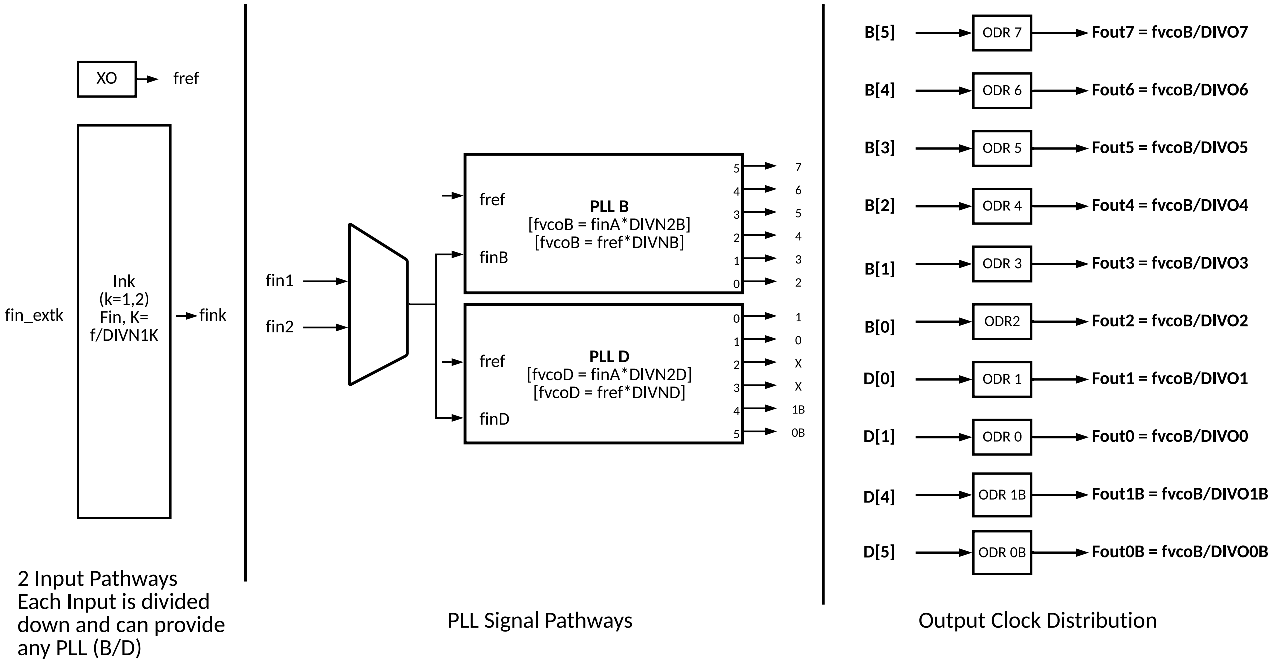


Figure 6 AU5329 Overall Hierarchy of Clocks

The two input clocks with frequencies fin_extk translate to PLL input clocks fink following division by the respective input dividers with fractional or integer frequency division ratios DIVN1k where the index k ϵ {1, 2}. See Figure 6. All of the PLLs choose one of the two divided input clocks fink as its active input clock and set the priority for up to three spare clocks from the remaining three input clocks if required for hitless switching to a redundant input.

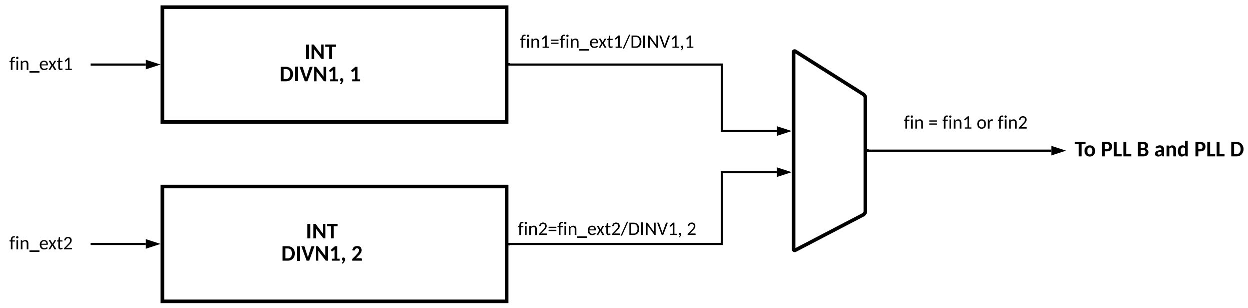


Figure 7 AU5329 Input Clock Distribution

The crystal oscillator reference input (called fref) is also routed to each PLL. A TCXO or OCXO based input reference clock can also be used directly in place of the crystal oscillator. Each PLLx (x ϵ {B, D}) has a high frequency VCO whose frequency is determined in the free run mode by fref with the relation fVCOx = DIVNx*fref. In the frequency translation synchronized mode, the VCO frequency is corrected from its free run frequency to satisfy the relation fVCOx = DIVN2x*finx where finx is chosen from one of fink input clocks per the desired input clock priority for PLLx. Nominally the fractional dividers DIVNx and DIVN2x are chosen such that the relation DIVNx*fref = DIVN2x*fin,x = fVCOx is satisfied. See Figure 8.

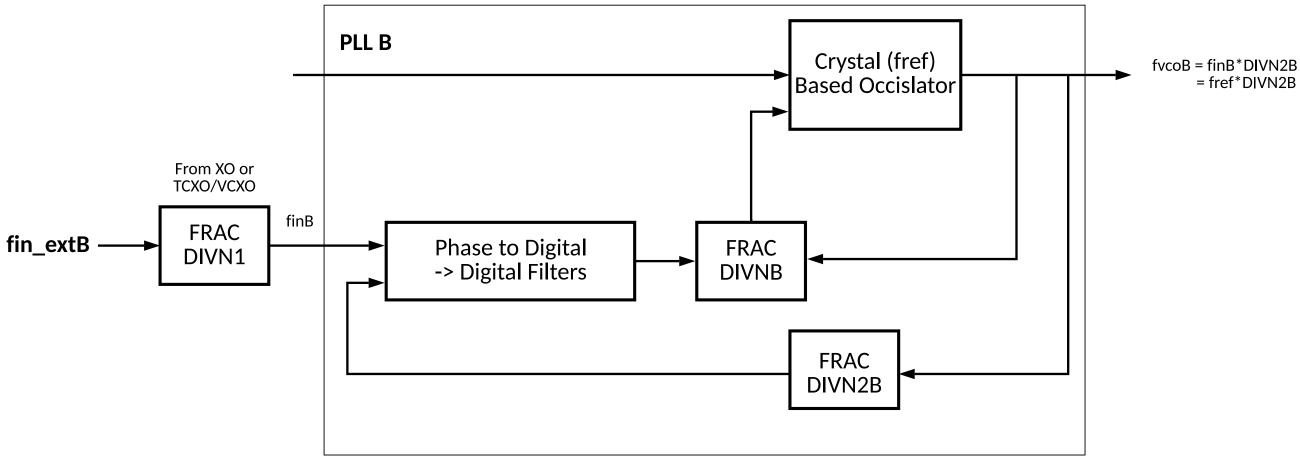


Figure 8 AU5329 PLL Divider

Each of the Output Drivers (ODRj, j ϵ {0, 1, 2, 3, 4, 5, 6, 7, 1B, 0B}) then chooses an appropriate VCO frequency and divides it using their respective integer divider DIVOj to get the output frequency foutj. See Figure 9.

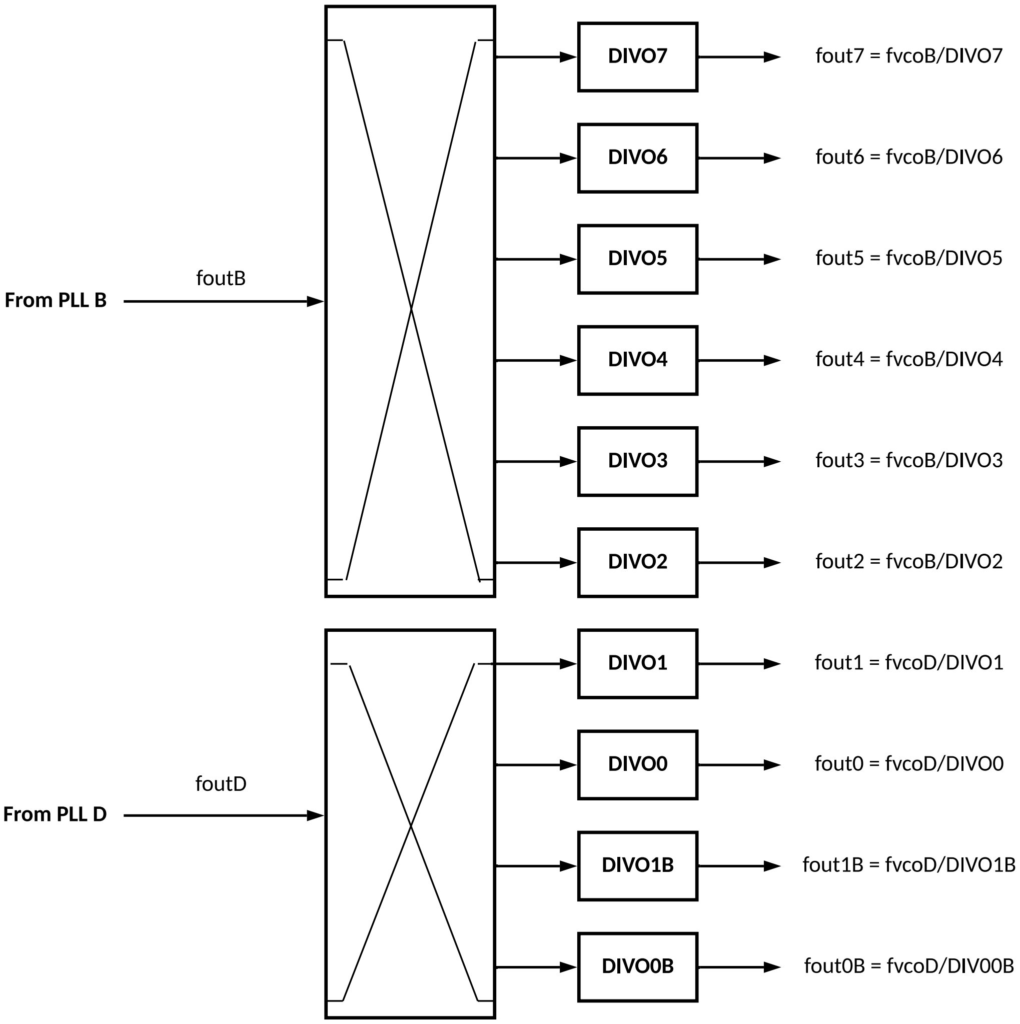


Figure 9 Output Clock Distribution

The choice of the fractional dividers {DIVN1k, DIVNx, DIVN2x, DIVOj} as well as the placement of foutj frequencies at various outputs is facilitated by associated software tools.

The digital architecture of the chip is partitioned into a master digital controller and five slave controllers. The master controller and each of the five controllers has an associated volatile programmable interface (PIF). The overall PIF structure is a register map that is divided into several pages according to function. Each controller (master and slaves) has an associated unique Page number. Each Page has an independent 8 bit addressable PIF memory. In all the pages, the last address, FF, holds the current page number and is reserved for changing the page. The current page to be communicated with can be set by writing the page number in hexadecimal form {0x00, 0x01, 0x02, 0x03, 0x0B, 0x0D} corresponding to pages {0, 1, 2, 3, B, D} in the address FF on any page. Table 17 shows a summary of the PIF contents residing on each page.

Table 17 PIF Description

Page	Contents	Summary of contents
0	Master	All Generic Information related to the chip Chip Configuration details Control for the master sequencer FSM Crystal Reference Related Information Fuse Pointer for each of the remaining pages
1	ClkMon Slave	Clock Loss related function Frequency Drift related function
2	Input Slave	Input 2 / 1 related information (Input type, DIVN1 divider configuration)
3	Output Slave	Flexible Outputs 7 / 6 / 5 / 4 / 3 / 2 / 1 / 0 (ODR Standards, DIVO, Programmable delay configurations for each)
		Fixed Outputs 0B (ODR Standards, DIVO, Programmable delay configurations for each)
B	PLL B Slave	All PLL related functionality
D	PLL D Slave	All PLL related functionality

4 Master and Slaves: Architecture Description and Programming Procedures

The Master controller is the first system to autonomously wake up on the application of power to the chip due to on-chip power on reset circuitry. All generic system information resides in the Master controller memory and it proceeds to wake up the Slaves as required based on this information. The relative wake up sequences of the Master and the various Slaves are described in more detail later in this section after a description of the memory structures. A complete power up of the chip is also emulated with the release of an active low hard reset (RSTB) from pin while selective Master and Slave sub-system resets are enacted from software using the serial interface (I2C/SPI).

The Master memory structure is shown in Figure 10. It contains a one-time programmable non-volatile memory (NVM) that stores the settings for the chip associated with the master controller. The master controller also contains a volatile PIF bank (NVMCopy) that has an exact copy of the NVM at every chip power up. This NVMCopy is the memory that is addressable using the serial interface (I2C/SPI) on Page 0 and can be overwritten from the I2C/SPI interface. The “Chip Settings” is the memory space that is not addressable from the I2C/SPI control and is the actual control for the chip. The NVM contains a two bit “Lock Pattern” that can be set to “10” or “01” to ‘lock’ the chip configuration once the final configuration is determined and wake up of the entire chip is desired in this configuration. Additionally, there is a bit in the NVM that is an active low indicator of a manual wake up. This bit set to “1” along with the ‘lock’ for the configuration leads to an autonomous wake up of the chip using the ‘locked’ configuration. Any number of different configurations can alternatively be tried at all times using only the volatile NVMCopy PIF section. This is useful for evaluations as well as allowing real time programming of the chip in various configurations with complete flexibility. The Master Controller finite state machine (FSM) described later in this section controls the device behavior in accordance with the configuration in this memory structure and as per the wake up mode.

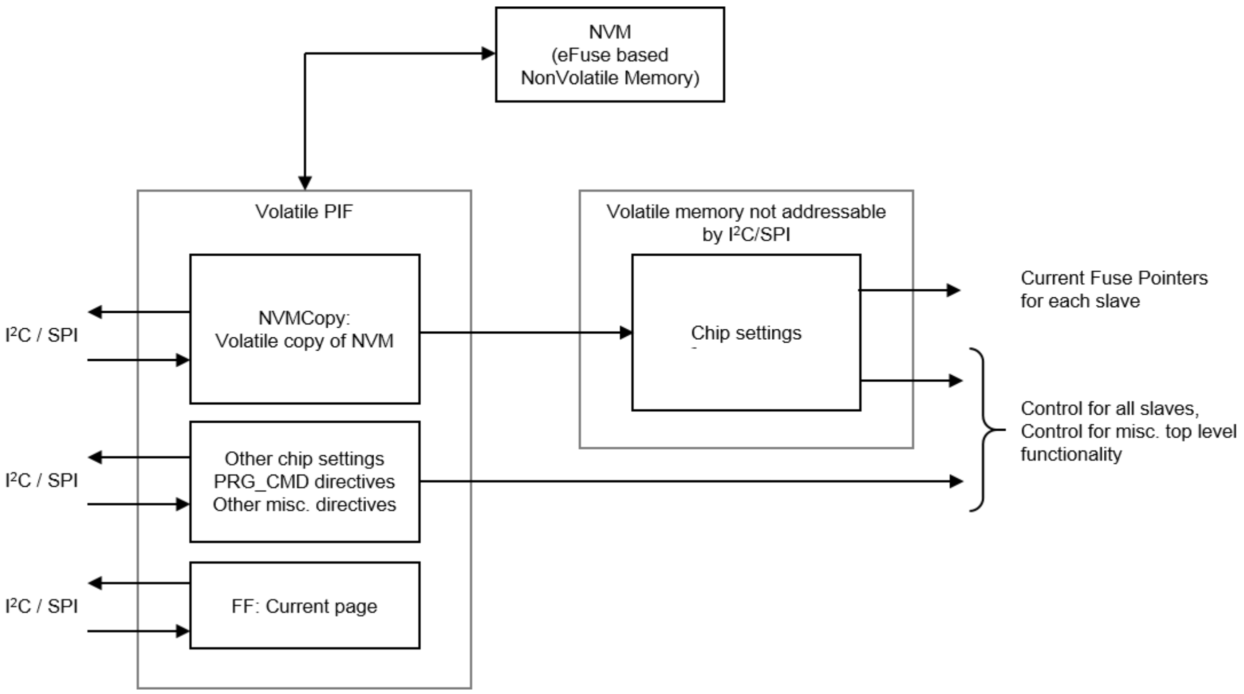


Figure 10 Master Memory Structure

The memory structure for each slave is shown in Figure 11 and is similar in construction to the master controller memory structure with some minor differences. The NVMCopy volatile PIF for the slave is addressable by the serial interface with the unique Page number associated with the slave. The “Slave Settings” is the memory space that is not addressable from the I2C/SPI control and is the actual control for the slave. Each Slave has a two time programmable NVM by virtue of two copies of the NVM memory. This makes the slave settings two time programmable with the fuse pointer from the master controller determining which of the two NVM banks is used. The presence of two NVM banks is transparent to the slave controller since the current pointer which determines which of the two NVM banks is used is set by the master controller independently.

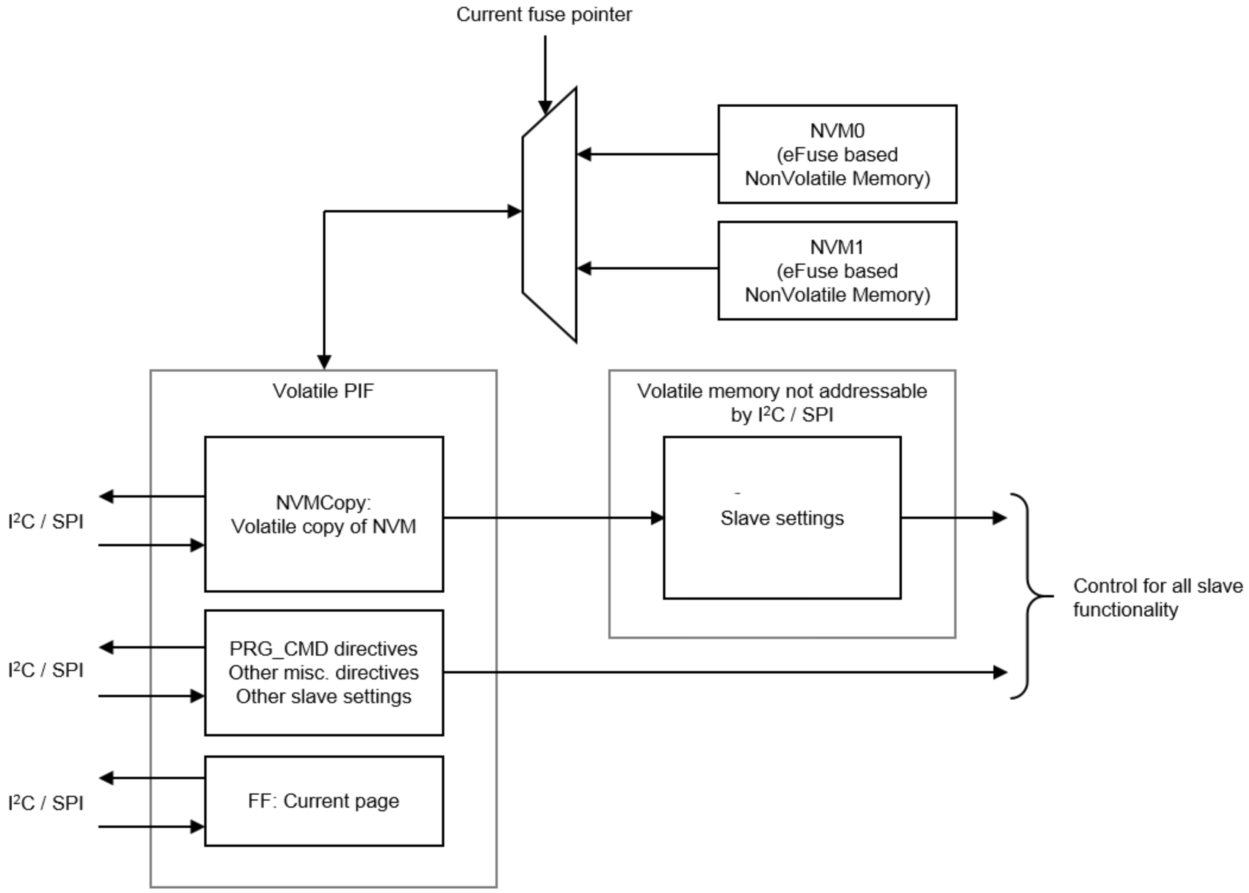


Figure 11 Slave Memory Structure

The Master Wake-Up Finite State Machine (FSM) is shown in more detail in Figure 12. At every power up of the device (or release from hard reset), the power-on-reset circuitry resets all systems and then autonomously releases only the master controller from reset. The NVM contents are copied to the NVMCopy volatile space on Page 0 which is in turn copied to the “Chip Settings”. The master controller now decides if the chip configuration is locked and it is an autonomous wake up of the entire chip or if a manual wake up is desired through the PIF based on the contents in the “Chip Settings”.

In case a “Lock” is detected and an autonomous wake up is desired, the Master controller proceeds to enable the Crystal oscillator and associated fref pathways followed by the Slave systems in a pre-determined sequence. This finally leads the chip to the “Active State” with all desired outputs available as a result of all slave systems released from reset by the master controller. This is according to the requested settings that are programmed in the Master and the Slave NVM banks.

For the case where the final chip settings are not frozen hence the “Lock” pattern is not exercised, the master controller FSM reaches the Program Command Wait State (PRG_CMD). The desired chip settings can be written in the NVMCopy on Page 0 using the serial interface and desired slave sub-systems can be enabled. Several PRG_CMD state directives are available that are exercisable only in this state. Using these directives, the desired settings written in NVMCopy can now be copied to “Chip Settings” followed by issuing the directive for the FSM to proceed to the “Active” state where each slave can now be manually written with the desired settings and in turn asked to proceed to its “Active” state.

A similar “Lock” pattern is available in the NVM bank of each slave. The currently used NVM bank for a slave (as determined by the current pointer from the master controller) can be locked for the autonomous wake up of each slave. The slave wake-up FSM is shown in Figure 13 and it similarly has a PRG_CMD state with associated directives. On Proceed to Active state directive on the slave, the slave controller wakes up the various blocks in its sub-system with the correct pre-determined sequence.

The NVM bank for the master and each slave can be programmed with a PRG_CMD directive in that state to lock a configuration / setting specific to the respective sub-system.

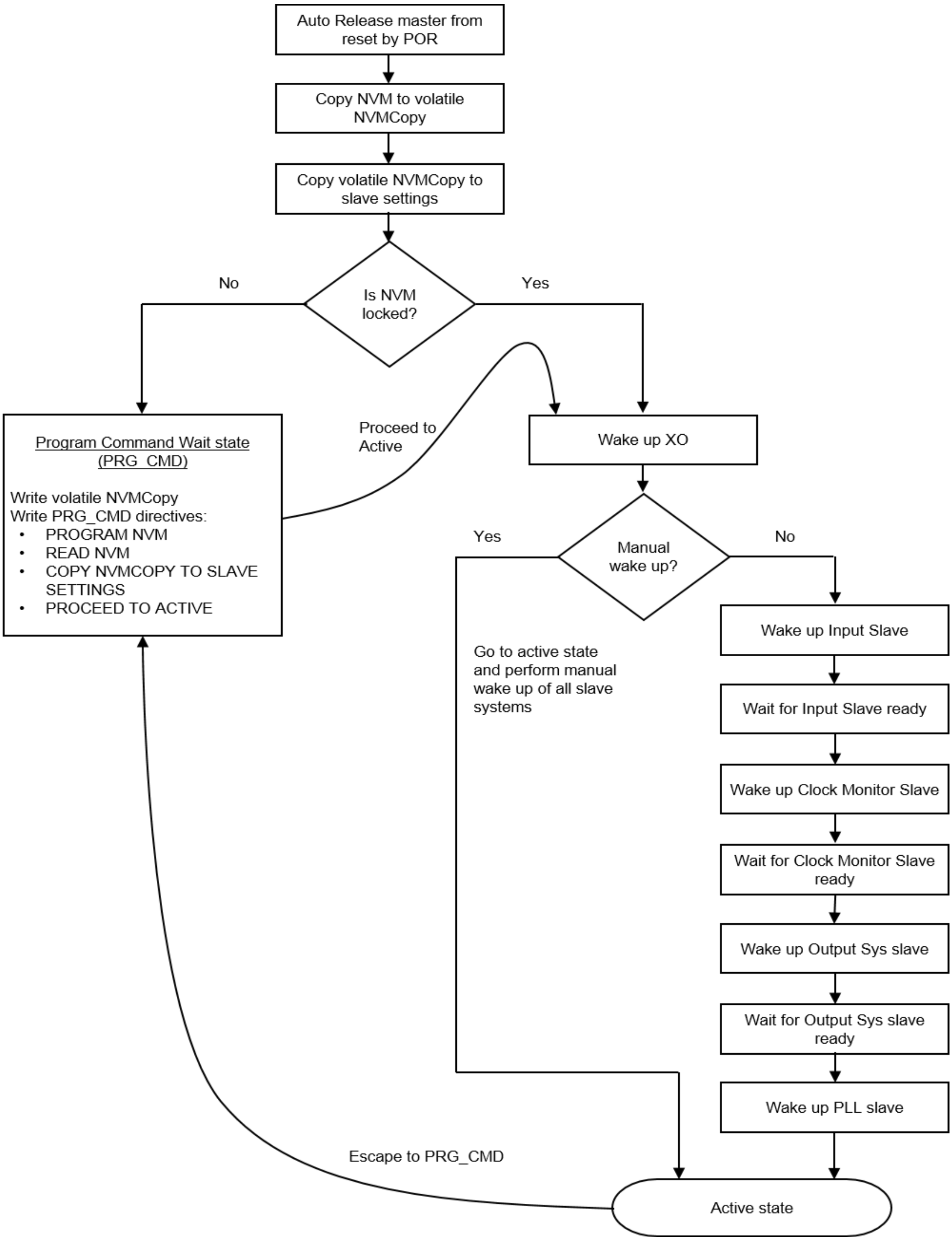


Figure 12 Master Wake-up Finite State Machine

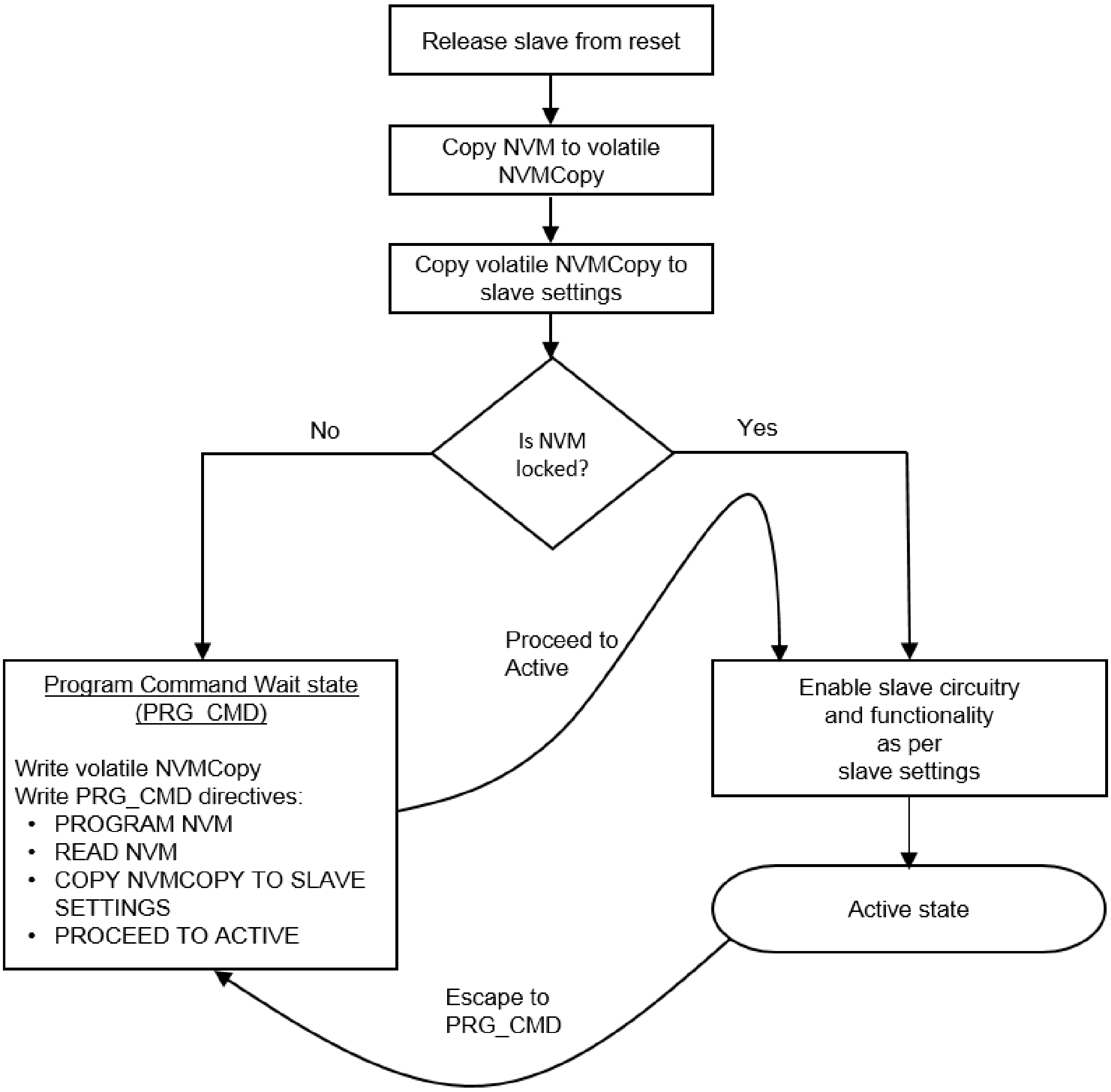


Figure 13 Slave Wake-up Finite State Machine

Each FSM (Master and Slaves) allows an escape sequence to go back to PRG_CMD state from its Active State. This can be used to selectively change the settings for that particular sub-system. Such an escape to the PRG_CMD state in the master FSM can be used for example to change current NVM pointers for any of the slaves.

Note that the NVM for the master controller and current NVM for all slaves should be locked after writing desired settings for a completely autonomous wake up of the entire chip. The NVM pointer can then be changed for any slave independently if alternate settings are desired for that slave. In that case, the new NVM is unlocked and can be written with new settings and locked. For evaluations of the chip as well as cases where flexible on-the-fly programmable settings are desired, the chip can be used without engaging the NVM banks at all by using the NVMCopy space for the master and each slave in conjunction with the PRG_CMD directives. It is also possible to lock some of the slaves (to not re-write their settings for each wake up) while use programmable settings for other slaves.

This provides complete flexibility in terms of programming and using the chip in all scenarios.

5 Input Slave Description

Two independent clock inputs are available on the chip that can be routed to any PLL with complete flexibility. Both single ended and AC coupled differential clock inputs are possible. The input clock receiver settings (to receive a single ended or differential clock) as well as the input clock divider settings are configurable on Page 2 that is assigned to the Input Slave. It is possible to bypass the input clock divider and use the input clock directly as an input to the PLL.

6 Clock Monitor Slave Description

Various fault monitoring indicators are available on the chip. The Clock Loss and the Frequency Drift indicators are configurable with the Clock Monitor Slave that is accessible on Page 1. The specifications of these fault monitors are indicated in the specifications section of the data sheet.

Defect monitoring on any of the clock monitors can be accessed using multiple techniques. The current status of the defect is available as an Active High defect that can be read from the PIF. The “status” is a current indicator of the defect that is high only during the defect (for example during the time that a Clock Loss event is on-going). Additionally, a sticky indicator of the defect called “Notify” can be enabled in the PIF. In this case, the concerned “notify” bit is high the first time the respective defect occurs and stays high till cleared.

There are multiple FLEXIOs (Flexible IOs) available in the system that can be programmed to monitor individual “notify” signals or a combination of them (as an OR logic). The choice of which fault defect is monitored as an output on the FLEXIO pin is flexible and can be programmed. Additionally there are selected GPIOs that are hard coded for the information for the clock defects.

6.1 Fault Monitoring

The AU5329 parts provide an elaborate arrangement of fault monitoring indicators. There are 4 categories of clock monitoring that are necessary for the chip namely: Clock Loss Monitor (CL), Frequency Drift Monitor (FD), Lock Loss Monitor (LL) and XO Clock Loss Monitor (CL_XO).

Clock Loss (CL) monitors loss of input clocks defined as a pre-determined number of consecutive edges missing.

Frequency Drift (FD) monitors frequency drift of a particular clock against a pre-determined Golden Reference.

Lock Loss (LL) monitors the loss of lock in any PLL by monitoring the difference in frequency between the feedback and input clocks.

XO Clock Loss (CL_XO) monitors the loss of the XO reference that is generated from either an external oscillator (XO / TCXO / OCXO) or using the on chip XO amplifier that can work with a crystal blank on the PCB.

Each of these categories monitors the health of a particular clock for a certain failure type as illustrated in the name of the clock monitoring category.

For each clock failure observed by the clock monitor block there are two types of indicators provided to the user using the register map:

1. Live Failure Bit: There is a bit to indicate the live status of a particular failure. [Status]
2. Sticky Failure Bit: For each live failure bit there is a corresponding sticky bit that is set the first time that corresponding failure is encountered and stays set even if the failure has gone away. Only when the user clears the bit does it clear. [Notify]

The status of these can be either read from the register map or from the pins as a dynamic alarm monitoring arrangement. Additionally, sticky notify registers are available which have sticky status read back from the register map for the various defects. These can be selectively chosen to create an INTRB de-assertion on the INTRB pin as well.

An important point to note is that all of the fault monitoring indicators mentioned above that work with respect to the input clock work on the divided input clock post the DIVN1,k dividers (refer to the data sheet for the respective AU5329 part). This implies that the fault monitoring indicators use the frequency fin*k* that is input to the PLL (*k*ϵ {1, 2}) post the DIVN1,*k* divider translation rather than the external frequencies fin_ext*k* (*k* ϵ {1, 2}).

Section 20 describes the read back of the alarms for the various fault monitoring arrangements using the chip register map.

6.1.1 Clock Loss Monitors

Each of the 2 inputs (IN1, IN2) are monitored for Clock Loss in terms of missing edges to indicate a loss of input signal. The number of edges used to indicate a clock loss (or recovery from a clock loss) is programmable in the AU5329 GUI interface allowing for flexibility in choosing these thresholds. In addition there is a programmable "Wait Time" all of which are to be interpreted as follows:

Assertion of Clock Loss-

We declare a CL if "Trigger Edge" number of consecutive edges are missing. The "Trigger Edge" parameter is programmable in the chip GUI.

De-Assertion of Clock Loss-

We declare a ~CL if the clock is back and has less than "Clear Edge" consecutive edges missing. The "Clear Edge" parameter is programmable in the chip GUI.

Wait Time: After the clock is established to have returned, it is ensured that no CL error as defined by the de-assertion threshold occurs for "Val Time" seconds. This valid wait time is programmable using the chip GUI using the "Val Time" parameter which is programmable from the following options: {2 m, 100 m, 200 m, 1} sec. The use of the this valid wait time ensures that sporadic edges in the input clock (such as ones caused by noise on floating nodes or intermittent unstable clock edges) does not de-assert clock loss and it is established over a user determined period of time that the input clock is available and stable.

6.1.2 Frequency Drift Monitors

Any one of the 2 input clocks or the XO clock can be used as the Golden Clock for calculating the frequency drifts of the other 2 clocks. The Golden Clock can be chosen in the GUI and is used as the "0 ppm" Reference Clock for all monitoring.

Fine Frequency Drift has a step size of ±2 ppm.

Fine Frequency Drift has a range of ±2 to ±510 ppm and an independent threshold is programmable for "Set" (for setting the FD monitor) and for "Clear" (for clearing the FD monitor).

Fine Frequency Drift has an implicit hysteresis with resolution of ±2 ppm since the same range is available for the FD assertion and de-assertion. Use of hysteresis prevents unwanted oscillation of the FD monitor output at the decision threshold and is recommended for robust operation. The value of the FD threshold hysteresis is implicit in the choice of the set and clear thresholds.

The Fine Frequency Drift monitors provide precise information for input clock frequency drift. However, since the resolution of the measurement determines time for the measurement- an alternate faster measurement mechanism for drift is needed. This is Coarse Frequency Drift which has coarser measurement but is fast. It is available for cases where the drift is very fast in the input frequency and is programmable from options as shown below.

Coarse Frequency Drift has a step size of ±100 ppm.

Coarse Frequency Drift has a range of ±100 to ±1600 ppm and an independent threshold is programmable for "Set" (for setting the FD monitor) and for "Clear" (for clearing the FD monitor).

Coarse Frequency Drift has an implicit hysteresis with resolution of ±100 ppm since the same range is available for the FD assertion and de-assertion. Use of hysteresis prevents unwanted oscillation of the FD monitor output at the decision threshold and is recommended for robust operation. The value of the FD threshold hysteresis is implicit in the choice of the set and clear thresholds.

Important Note regarding the above monitors with respect to clock switch in the PLL:

Normally the CL monitor is used for ascertaining a clock is lost for the PLL to switch to a secondary reference or proceed to Holdover. However, the Fine and/or Coarse FD monitors can also be used in addition to the CL monitor to cause a PLL switch. This implements an "OR" logic for the FD Monitors to be used in addition to the CL monitors for triggering a PLL input clock switch or entry to Holdover. This is programmable as an option in the GUI.

6.1.3 Lock Loss Monitors

Lock loss is programmable for each PLL with lock loss triggered if the frequency of the input reference to the PLL phase detection arrangement and the feedback clock to same PLL are different as per the programmed assertion and de-assertion thresholds.

The Set threshold for asserting the LL monitor is programmable from {±0.2, ±0.4, ±2, ±4, ±20, ±40, ±200, ±400, ±2000, ±4000} ppm while the Clear threshold for de-asserting the LL monitor is programmable from {±0.2, ±0.4, ±2, ±200} ppm. A pre-determined level of hysteresis is implicit by choosing appropriately the set and clear thresholds for the LL monitor.

Additionally from the point of view of LL de-assertion, there is a delay from the point in time that lower than the specified ppm value is achieved to the point where the actual LL is de-asserted to the user such that LL never asserts during this delay period. The choice of this delay is with a timer that ensures that the delay is in line with the BW of the PLL loop. It is fully programmable from the GUI and is useful to ensure complete settling of the PLL without un-necessary toggling before LL de-assertion.

6.1.4 XO Clock Loss Monitors

The XO Clock Loss Monitor asserts the XO Clock Loss Alarm when the external reference input to the X1 pin (XO or TCXO or OCXO) or the internal XO clock generated with the crystal blank is not available.

7 Output Slave Description

The Output Slave accessible on Page 3 is used to configure the output divider (DIVO) and output standard for each output individually. The output load and terminations for each differential output standard are shown in the Output Terminations section of the data sheet. The LVDS and LVDS Boosted modes are recommended for AC coupled termination loads with the termination at the far end. Additionally, an internal termination mode for differential outputs is available where the resistive terminations are internally provided and a differential output is available that can be AC coupled to a clock receiver. The differential clock output pins are shared for LVCMOS outputs as well. LVCMOS outputs can be either enabled on both outputs individually or on any one of the two differential outputs {OUTjP, OUTjN}. The LVCMOS outputs can be used in-phase or out-of-phase on {OUTjP, OUTjN} in case both outputs are chosen. Out of phase LVCMOS toggling on the complementary outputs is recommended for best spur performance.

8 PLL Slave Description

All settings with respect to each PLLx slave ($x \in$ {B, D}) are accessible on the respective Page {B, D}. The PLL architecture is shown in Figure 14. There are three distinct modes of operation of the PLL: free run mode, synchronized mode and holdover mode. The frequency of the high frequency VCO in the PLL is determined by the specific mode of operation. The VCO frequency is then divided down to get the output frequency on the ODR as described with relation to the overall hierarchy of clocks described earlier.

The PLL in the free run mode can be described as a crystal based oscillator where the output frequency is determined by the relation fVCOx = DIVNx*fref. This is the mode of operation before the loop is locked to the selected input clock or the mode of operation for the case none of the input clocks is available. After locking to the chosen input clock, the PLL enters the synchronized mode of operation where the output is now locked to the input frequency with the relation fVCOx = DIVN2x*finx. The PLL Loop that synchronizes (locks) the output to the input clock has a programmable loop bandwidth between 1 mHz to 4 KHz and is not affected by static or dynamic drifts in the crystal oscillator based fref frequency. In case the input clock is lost, the PLL locks to the highest priority spare clock available. If all specified input clocks are lost, the PLL remembers the correction based on historical average of the input clock as specified to enter the Holdover mode of operation.

In synchronized mode, the PLL is also able to lock to a Gapped Input clock with some edges missing producing a smooth output clock without any gaps with the requested frequency translation from input to output. Frequency translation ratios in this case should be specified with respect to the average input frequency of the gapped clock rather than the faster instantaneous frequency.

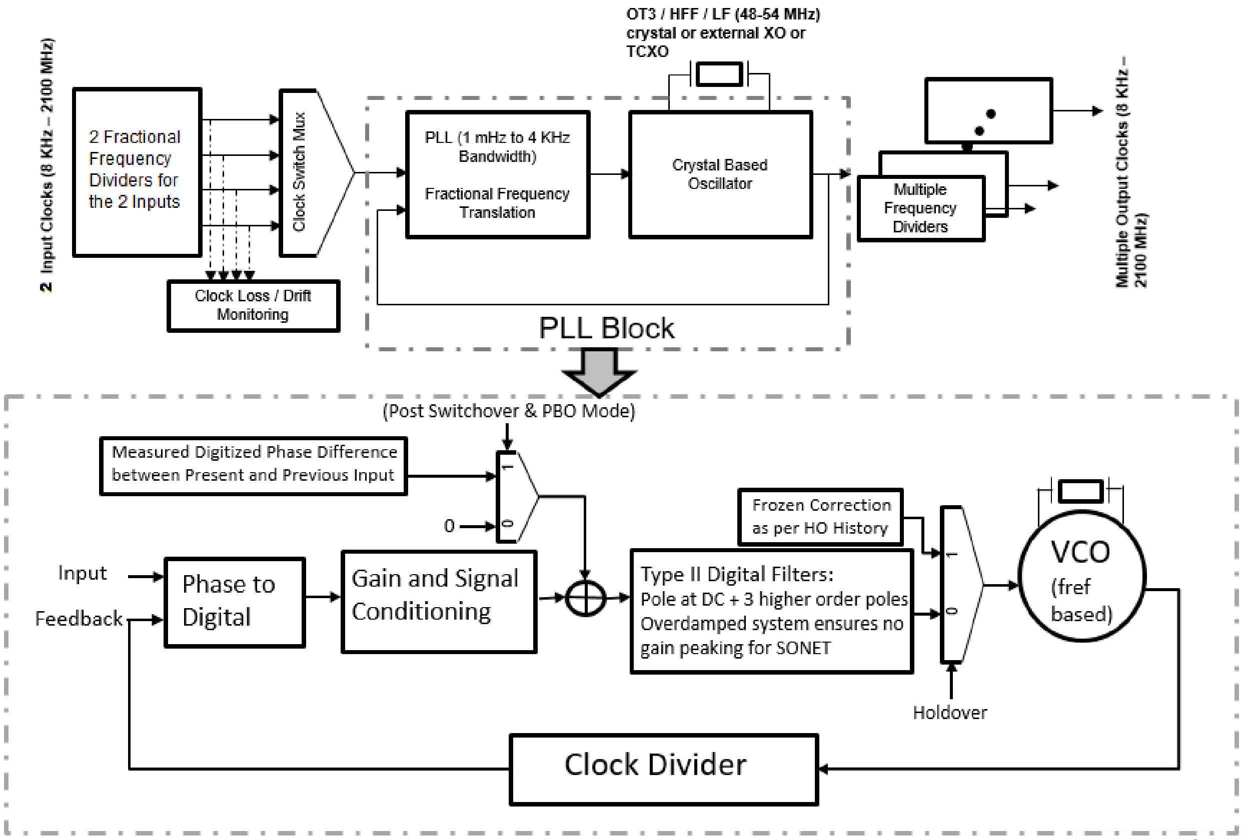


Figure 14 PLL Architecture

9 PLL Input Selection: Manual and Hitless Switching

All PLL Slaves share the same clock priority in terms of the two input clocks. This is programmed in to the Clock Monitor slave memory. The PLL Slave then looks at Clock Loss status from the Clock Monitor slave to lock to the highest priority available clock to lock. One spare clocks with an order of priority can be specified in case the highest priority active clock is not available. Additionally, a forced manual selection of the active clock with no spares is possible.

Phase Build Out Mode of hitless switching ensures that phase transients are not propagated to the output (the phase difference between redundant input clocks is absorbed by the PLL) and desired MTIE characteristics are seen in the output clock. This is the default mode of hitless switching for the PLL. The transition of input clock for a PLL from one clock to another is hitless in nature (with maximum phase hit limited to be less than 50 ps) for the case of the switched input clocks being same in frequency. Hitless switch is also supported for the switched clocks being fractionally related such that the same frequency can be obtained for both clocks at the input of the PLL using the input clock dividers (DIVN1k).

For redundant input clocks to the PLL that are not exactly the same frequency (plesichronous clocks), the frequency ramp feature can be enabled that ramps the output frequency of the PLL at a slope that is programmable to one of the following 4 settings: {0.2, 2, 20, 40000} ppm/s. For redundant input clocks to the PLL that are exactly the same frequency, the frequency ramp feature should not be enabled.

An alternate mode of hitless switching is the Phase Propagation mode where the phase difference between redundant input clocks is not absorbed by the PLL but is rather propagated to the output. The phase difference that is propagated to the output can either be allowed to propagate as per the PLL bandwidth or can be limited to a phase propagation slope that is programmable to one of the following 3 settings: {10, 40, 160} us/s.

10 PLL Bandwidth Control

Each PLL Slave independently chooses the Bandwidth for jitter attenuation from 1 mHz to 4 KHz. This is the bandwidth that is normally used for steady state operation. However, an independent choice for a fast bandwidth is also available that can be used for speeding up the initial lock. After the PLL lock is achieved and the system is in the synchronized mode, the bandwidth is automatically transitioned to the steady state jitter attenuation bandwidth. This feature avoids the abnormally large wake up times that may be needed for very low PLL bandwidths. For stability considerations of the PLL, the fast lock bandwidth or regular bandwidth for the PLL should be no larger than $1/100^{th}$ of the input frequency at the input of the PLL (post the DIVN1k dividers).

11 PLL Crystal Clock Reference

An external crystal can be connected between the {X1, X2} pins on the die to work with the internal crystal oscillator circuitry to produce the fref clock for the system. Alternatively, a TCXO based external clock source can be directly connected on the X1 pin. The requirements for the crystal and the external clock source are presented in Section 2. It is recommended to place the crystal on a floating metal island on the PCB that is provided the ground connection by the chip with the X1G and X2G pins. This metal island should not be connected to the PCB ground to prevent extraneous fref related currents to distribute on the board.

12 PLL Lock Loss Defect Monitoring

PLL Lock Loss is another fault monitor whose specifications are available in the Electrical Specifications section of this data sheet. Various programmable thresholds are available that can be used to detect lock loss in the PLL. Lock loss is indicated by the programmable drift between the frequency of the input clock for the PLL and the divided VCO clock. Similar to the faults monitored by the Clock Monitor Slave, this defect can be tracked with status, notify and on the FLEXIOs. This defect monitoring is described in detail in the section on “Clock Monitor Slave Description”.

13 PLL DCO Mode operation

The Digitally Controlled Oscillator (DCO) mode of operation is used for changing the output frequency of a PLL using software control on the serial interface or pin control. A pre-defined change in frequency is programmed in the PIF of the respective PLL. After that an increase (FINC) or decrease (FDEC) command can be given on the PIF of the same PLL to make the change in output frequency effective. Alternatively, appropriate GPIOs are chosen for the trigger of the DCO function. A low to high transition (as an edge detect) is used for the trigger of the DCO increment or decrement. Any relative change in frequency from as fine as 5 ppt to as coarse as 100 ppm is available with the DCO mode. DCO mode is available in both free run and synchronized modes of operation.

14 Programmable Interface Top Level View

Table 18 PIF Overview (Top Level Summary of the Programmable Interface)

All Address are in Hexadecimal

Page Number	Function	Comments	Fuse repeated twice
0h	Generic	***2Fh[7:6]: Lock Pattern for the Fuse*** ***FFh[7:0]: Current Page Number*** ***22h[7:0]: Current Fuse Pointer*** 00h - 01h: Customer- Chip Information 02h - 04h: First set of Defect / Notify / Interrupt 06h - 08h: Second set of Defect / Notify / Interrupt 05h: DCO increment/decrement control 0Fh: Program Command Directives and Active Trigger Directives 10h: PLL enable control 11h - 18h: Fuse GPIO (FlexIO) Multiplexed Control 19h: VDD Padring Control and External CLKIN Switch Control 1Ah - 21h: Die ID + Wafer Co-ordinates 22h: Fuse Pointer Generic 24h: Clock Input / Output Enable Control 25h: Clock Output Enable Control 26h: OEb, Clock Output Enable Control Settings 27h - 28h: Masking of sticky bits status for Interrupt generation (INTR_b) 29h: Chip GPIOs (FlexIO) Configuration 2Ah: Fuse Based I2C Addr 2Bh: Calibrations and Misc Settings 2Ch - 2Fh: XO/XTAL/Reference Pathway Settings	NO
1h	Clock Monitor	***2Fh[7:6] : Customer - Lock Pattern for the Fuse*** ***FFh[7:0]: Current Page Number*** 00h - 01h: Chip Information 02h - 04h: First set Defect / Notify / Interrupt for Clock Monitor Sub-system. 06h - 08h: Second set of Defect / Notify / Interrupt for Clock Monitor Sub-system 0Fh: Program Command Directives and Active Trigger Directives 10h - 29h, 46h - 48h: Clock Loss Monitor Configuration 2Ah - 45h, 4Ch - 4Fh: Frequency Drift Coarse/Fine Configuration 49h - 4Bh: PLLs Input Clock Priority Information	YES
2h	Input	***2Fh[7:6] : Customer - Lock Pattern for the Fuse*** ***FFh[7:0]: Current Page Number*** 00h - 01h: Chip Information 02h - 04h: Defect / Notify / Interrupt for Input Sub-system 0Fh: Program Command Directives and Active Trigger Directives 20h - 2Fh: CLKIN1 Fuse Configuration (IDR, DIVN1, Clock MUX) 30h - 3Fh: CLKIN2 Fuse Configuration (IDR, DIVN1, Clock MUX)	YES

Page Number	Function	Comments	Fuse repeated twice
3h	8 Flexi-Outputs / 4 Fixed-Output Blocks	***2Fh[7:6] : Customer - Lock Pattern for the Fuse*** ***FFh[7:0]: Current Page Number*** 00h - 01h: Chip Information 02h - 04h: Defect / Notify / Interrupt for Output Sub-system 0Fh: Program Command Directives and Active Trigger Directives 10h - 17h: Output Block 0 Fuse Configuration (ODR, DIVO, DIVO-Delay) 18h - 1Fh: Output Block 1 Fuse Configuration (ODR, DIVO, DIVO-Delay) 20h - 27h: Output Block 2 Fuse Configuration (ODR, DIVO, DIVO-Delay) 28h - 2Fh: Output Block 3 Fuse Configuration (ODR, DIVO, DIVO-Delay) 30h - 37h: Output Block 4 Fuse Configuration (ODR, DIVO, DIVO-Delay) 38h - 3Fh: Output Block 5 Fuse Configuration (ODR, DIVO, DIVO-Delay) 40h - 47h: Output Block 6 Fuse Configuration (ODR, DIVO, DIVO-Delay) 48h - 4Fh: Output Block 7 Fuse Configuration (ODR, DIVO, DIVO-Delay) 50h - 57h: Output Block 0T Fuse Configuration (ODR, DIVO, DIVO-Delay) 58h - 5Fh: Output Block 1T Fuse Configuration (ODR, DIVO, DIVO-Delay) 60h - 67h: Output Block 0B Fuse Configuration (ODR, DIVO, DIVO-Delay) 68h - 6Fh: Output Block 1B Fuse Configuration (ODR, DIVO, DIVO-Delay)	YES
Bh	PLL B	***2Fh[7:6] : Customer - Lock Pattern for the Fuse*** ***FFh[7:0]: Current Page Number*** 00h - 01h: Customer- Chip Information 02h - 04h: First set of Defect / Notify / Interrupt for PLLB 06h - 08h: Second set of Defect / Notify / Interrupt for PLLB 05h: Customer- PLL Generic Directives 0Fh: Program Command Directives and Active Trigger Directives 10h - 2Fh: PLL Fuse Configuration (All PLL specific settings for this PLL) 30h - 37h: Customer- DCO Functionality	YES

15 Serial Programming Interface Description

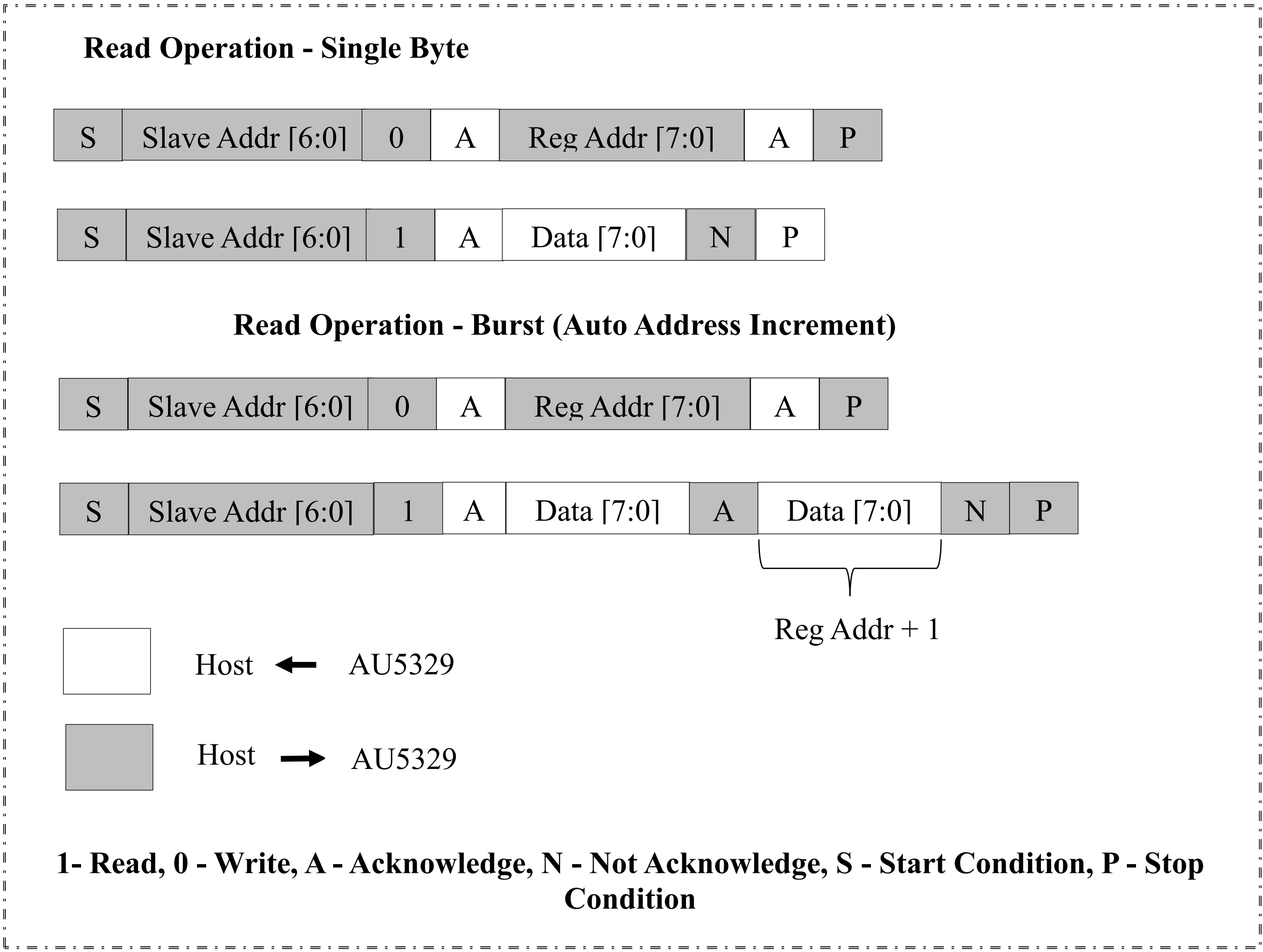
The device has two serial programming interface options, I2C and SPI, for reconfiguring the device settings. The protocol option can be selected through the I2C1_SPI0 pin. A 1/HIGH on the pin sets the device in I2C mode and a 0/LOW in SPI mode.

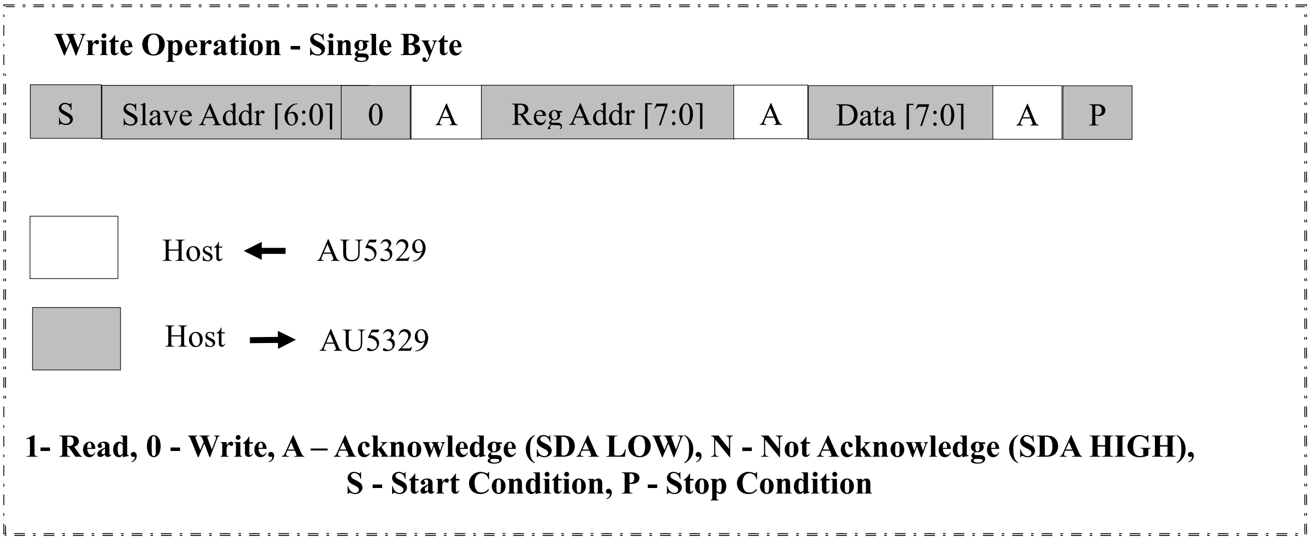
15.1 I2C protocol

The device uses the SDAIO and SCLK pins for a 2-wire serial interface that operates up to 400 Kb/s in Read and Write modes. It complies with the I2C bus standard. The I2C access protocol in device is byte access (random access) only for Write mode and both random and sequential access for Read mode.

The I2C serial interface can operate at either Standard rate (100 Kbps) or Fast rate (400 Kbps). For Write operation, the device supports only single write operation. For Read, the device supports both single and multiple read operation.

The default Slave address is 11010{SDO},{CSB} where SDO and CSB values are controlled by pins on the device in the I2C mode. Default addess is 0x69. The device also supports variable Slave addresses which can be provided via the efuse. Therein too, the LSbs of A1 and A0 are controlled via the pins on the device. This allows four choices of Slave addresses for any system where in the first 5 bits of the slave address can be the same.





- Single Byte Write
  - The master initiates the transaction by issuing a start condition, writes 7 bit slave address and then the read/write bit is written as 0 (write)
  - The slave acknowledges by driving zero on the bus
  - The master then writes the 8 bit register map address
  - The slave acknowledges by driving zero on the bus
  - The master then writes the 8 bit data to be written to the register map address specified
  - The slave acknowledges by driving zero on the bus
  - The master ends the transaction by issuing a stop condition
- Single Byte Read
  - The master initiates the transaction by issuing a start condition, writes 7 bit slave address and then the read/write bit is written as 0 (write)
  - The slave acknowledges by driving zero on the bus
  - The master then writes the 8 bit register map address
  - The slave acknowledges by driving zero on the bus
  - The master ends the transaction by issuing a stop condition
  - The master re-initiates the transaction by issuing a start condition, writes 7 bit slave address and then the read/write bit is written as 1 (read)
  - The slave then writes the 8 bit data to be written to the register map address specified
  - The master does not acknowledge this transaction as the slave may assume a multi-byte read operation and there is a risk of slave holding the bus low
  - The master ends the transaction by issuing a stop condition
- Multi Byte Read

The multi-byte read mode is used to read a continuous segment of the register map. The multi-byte read is faster than performing multiple single byte reads as the device address and register map address need not be specified for every byte read from the register map

  - The master initiates the transaction by issuing a start condition, writes 7 bit slave address and then the read/write bit is written as 0 (write)
  - The slave acknowledges by driving zero on the bus
  - The master then writes the 8 bit register map address
  - The slave acknowledges by driving zero on the bus
  - The master ends the transaction by issuing a stop condition
  - The master re-initiates the transaction by issuing a start condition, writes 7 bit slave address and then the read/write bit is written as 1 (read)
  - The slave then writes the 8 bit data to be written to the register map address specified

- The master acknowledges by driving zero on the bus
- The slave automatically increments the register map address and writes the data in at that address to the bus and the master acknowledges
- When all bytes of data are read, master ends the operation by not acknowledging the last read
- The master then ends the transaction by issuing a stop condition

15.2 12C Timing

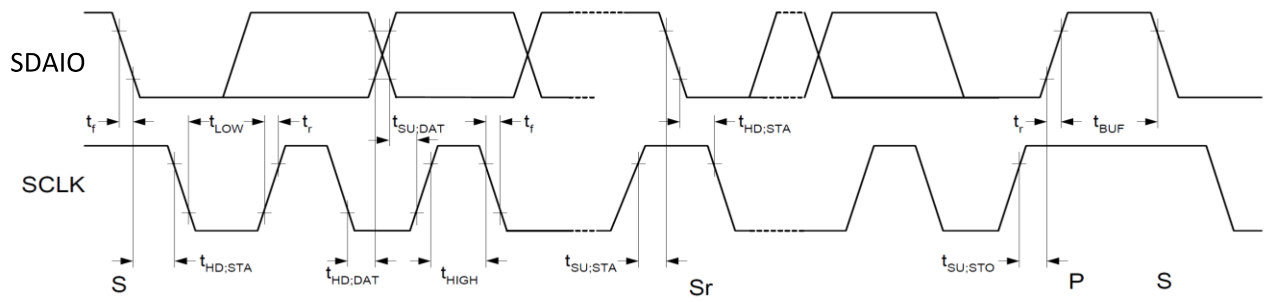


Figure 15 I2C Timing Waveform

Table 19 I2C Bus Timing Specifications

Description	Symbol	Standard Mode		Fast Mode		Units
		Min	Max	Min	Max	
SCLK clock frequency	f_{SCL}	–	100	–	400	kHz
Hold time START condition	$t_{HD:STA}$	4.0	–	0.6	–	µs
Low period of the SCK clock	t_{LOW}	4.7	–	1.3	–	µs
High period of the SCK clock	t_{HIGH}	4.0	–	0.6	–	µs
Setup time for a repeated START condition	$t_{SU:STA}$	4.7	–	0.6	–	µs
Data hold time	$t_{HD:DAT}$	10	–	10	–	ns
Data setup time	$t_{SU:DAT}$	100	–	100	–	ns
Rise time	t_R	–	1000	–	300	ns
Fall time	t_F	–	300	–	300	ns
Setup time for STOP condition	$t_{SU:STO}$	4.0	–	0.6	–	µs
Bus-free time between STOP and START conditions	t_{BUF}	4.7	–	1.3	–	µs
Data valid time	$t_{VD;DAT}$	–	3.45	–	0.9	µs
Data valid acknowledge time	$t_{VD;ACK}$	–	0.9	–	0.9	µs

15.3 SPI Protocol

The SPI is a four-pin interface with Chip Select (CSB), Serial Input (SDAIO), Serial Output (SDO), and Serial Clock (SCLK) pins. The SPI bus on the device can run at speed up to 20 MHz. The SPI is a synchronous serial interface, which uses clock and data pins for serial access. When I2C1_SPI0 pin is Low, a Low on the CSB pin activates the SPI access.

1. The SPI can operate up to 20 MHz for regular write/read operations.
2. The SPI receives serial data from the external master and provides Wr/rdn (set to 0x01h), address and data to the register map during the write operation.
3. The SPI receives serial data from the external master and provides Wr/rdn (set to 0x00h), address to the register map and uses the read data obtained from the register map, serializes the same and transmit to the master.

4. In AU5329, the total packet size for each SPI transaction is 24 bits where the 8 bits are Wr/rdn (0x01 for write and 0x00 for read), the next 8 bits are address and the last 8 bits are data
5. In AU5329 for write operation, the master assembles the Wr/rdn byte, address and data for write operation on the falling edge of the spi clock and the slave in the AU5329 captures the same on the rising edge of the SPI clock. There is no loopback provided here.
6. In AU5329 for read operation, the master assembles the Wr/rdn byte, address for read operation on the falling edge of the spi clock and the slave in the AU5329 captures the same on the rising edge of the SPI clock and there is no loopback. The falling edge after the 16^{th} rising SPI clock (i.e. the last address bit), is used by the slave to assemble the first read data which is captured by the master on the 17^{th} edge of the SPI clock. Subsequent 7 more clocks are used for the 7 remaining data bits.
7. In AU5329 the transmitter always sends data on the falling edge of the SPI clock to be captured in the receiver by the rising edge of the SPI clock. The transmitter can be the master for the whole operation of the write and for the control and address portions of the read. The slave is the transmitter during the data portion of the read cycle.
8. The register can be written to or read from one address at a time. The SPI implemented in AU5329 does not support burst address write or read operations.

15.4 SPI Timing Details

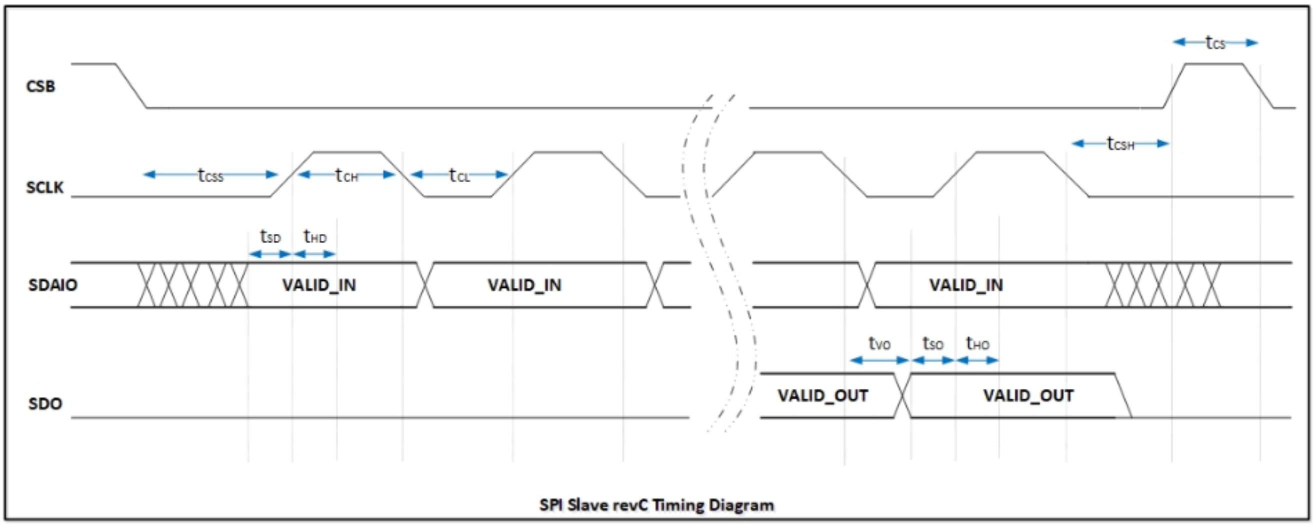


Figure 16 SPI Timing Diagram

Table 20 SPI Timing Specifications

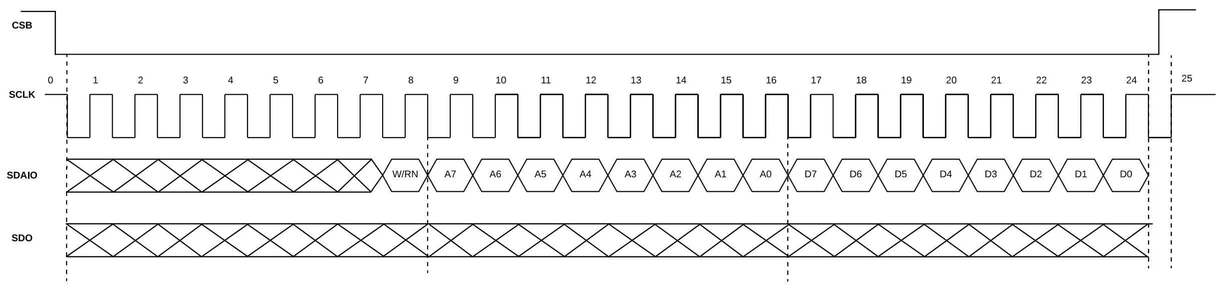
Description	Symbol	Min	Typ	Max	Units
SCLK clock frequency	f_{SCLK}	-	–	20	MHz
Clock pulse width HIGH	t_{CH}	20			ns
Clock pulse width LOW	t_{CL}	20			ns
CSB HIGH time	t_{CS}	50			ns
CSB setup time	t_{CSS}	25			ns
CSB hold time	t_{CSH}	25			ns
Data in setup time	t_{SD}	10			ns
Data in hold time	t_{HD}	10			ns
Output valid	tV_O			10	ns
Output valid	tS_O			10	ns
Output valid	tH_O			10	ns

15.4.1 SPI Single byte write

- The master initiates the transaction by issuing a start condition by pulling csb_i to active low
- The master assembles the serial data on the falling edge of the SPI clock so the SPI slave can capture the same on the rising edge of the SPI clock
- The first 7 bits are don't care with the 8th bit being set to 1 to indicate a write operation
- The next 8 bits (second byte) are used for the register map address
- The next 8 bits (third byte) are used for the register map data
- The 24th rising edge of the SPI clock is used to capture the last data bit. The SPI slave then assembles the address, data, enable and wr_rdn to the PIF slave block. The inverted version of the next falling edge of the SPI clock is used by the SPI slave to capture the address, data, enable and wr_rdn to write to the respective registers.
- The CSB is then de-activated (by going high) by the master
- For the next write operation, CSB is held high for at least a duration of two spi clocks following which the entire operation can start again.

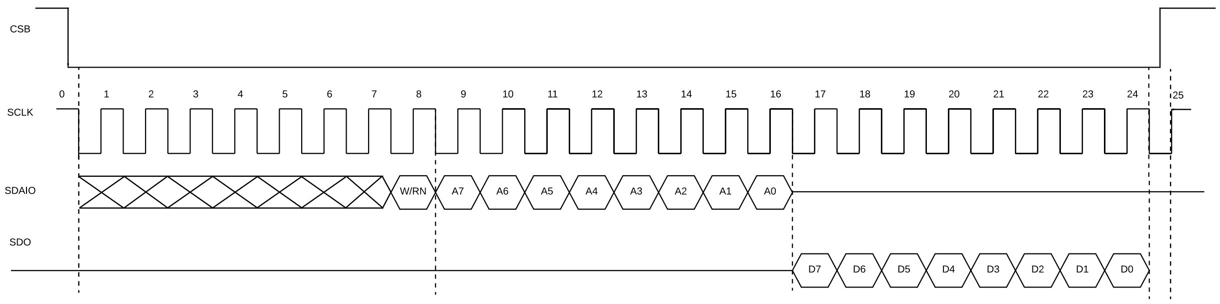
15.4.2 SPI Single byte read

- The master initiates the transaction by issuing a start condition of pulling csb_i to active low
- The master assembles the serial data on the falling edge of the SPI clock so the SPI slave can capture the same on the rising edge of the SPI clock
- The first 7 bits are don't care with the 8th bit being set to 0 to indicate a read operation.
- The next 8 bits (second byte) are used for the register map address
- The next 8 bits (third byte) are used for the register map read data that is supplied by the pif slave block
- The 16th rising edge of the SPI clock is used to capture the last address bit. The SPI slave then assembles the address, enable and wr_rdn to the PIF slave block. The slave block then uses the address when enable is high to provide the read back data via a multiplexer. This operation has to be completed within half a SPI clock since the SPI slave has to assemble the first read back data bit on the falling edge of the SPI clock so the SPI slave can capture the same on the next rising edge. After 7 additional clocks, all the 8 serial read back data bits are sent out from the SPI slave.
- The CSB is then de-activated (by going high) by the master.
- For the next read operation, CSB is held high for at least a duration of two spi clocks following which the entire operation can start again.



Note:
W/RN - 1:SPI WRITE 0:SPI READ
Ax - Address
Dx - Data In

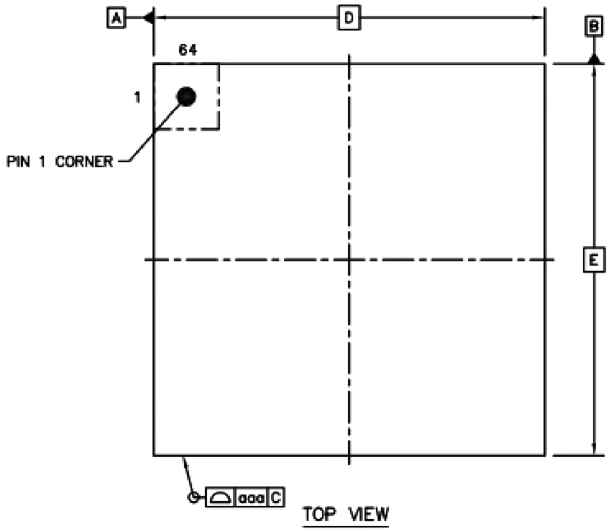
Figure 17 SPI Write – AU5329



Note:
W/RN - 1:SPI WRITE 0:SPI READ
Ax - Address
Dx - Data Out

Figure 18 SPI Read – AU5329

16 Package Information



		SYMBOL	MIN	NOM	MAX
TOTAL THICKNESS		A	0.8	0.85	0.9
STAND OFF		A1	0	0.035	0.05
MOLD THICKNESS		A2	---	0.65	---
L/F THICKNESS		A3		0.203 REF	
LEAD WIDTH		b	0.2	0.25	0.3
BODY SIZE	X	D		9 BSC	
	Y	E		9 BSC	
LEAD PITCH		e		0.5 BSC	
EP SIZE	X	J	4.7	4.8	4.9
	Y	K	4.7	4.8	4.9
LEAD LENGTH		L	0.35	0.4	0.45
		L1	0.3	0.4	0.45
PACKAGE EDGE TOLERANCE		aaa		0.1	
MOLD FLATNESS		bbb		0.1	
COPLANARITY		ccc		0.08	
LEAD OFFSET		ddd		0.1	
EXPOSED PAD OFFSET		eee		0.1	

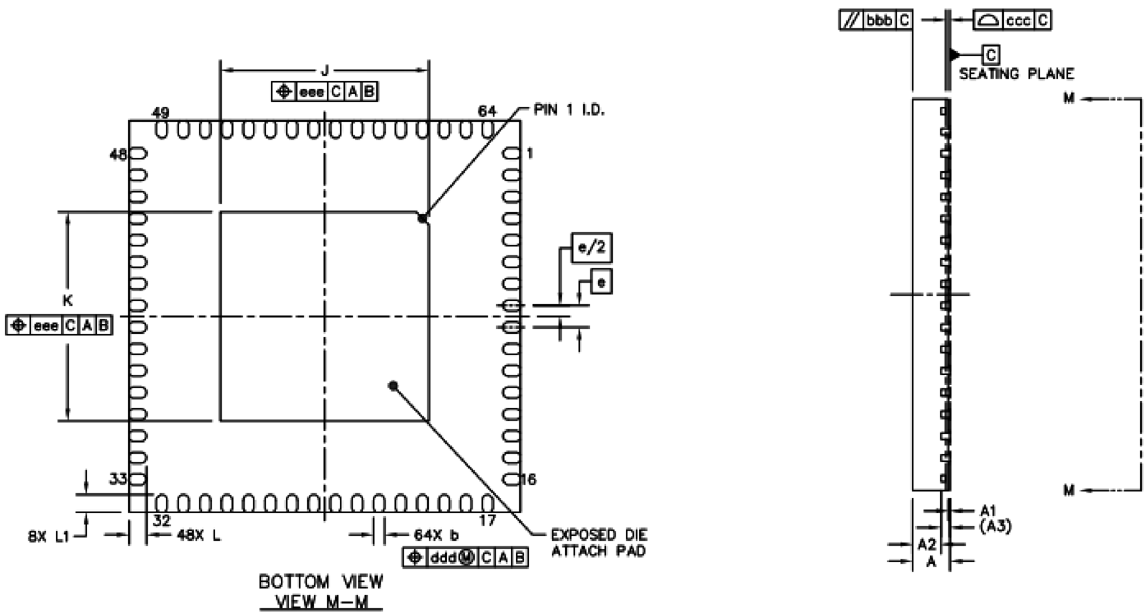


Figure 19 AU5329 – 64 QFN Package Description

3. Notes:
   1. Coplanarity applies to LEADS, CORNER LEADS and DIE ATTACH PAD
   2. Total Thickness does not include SAW BURR

17 Output Termination Information

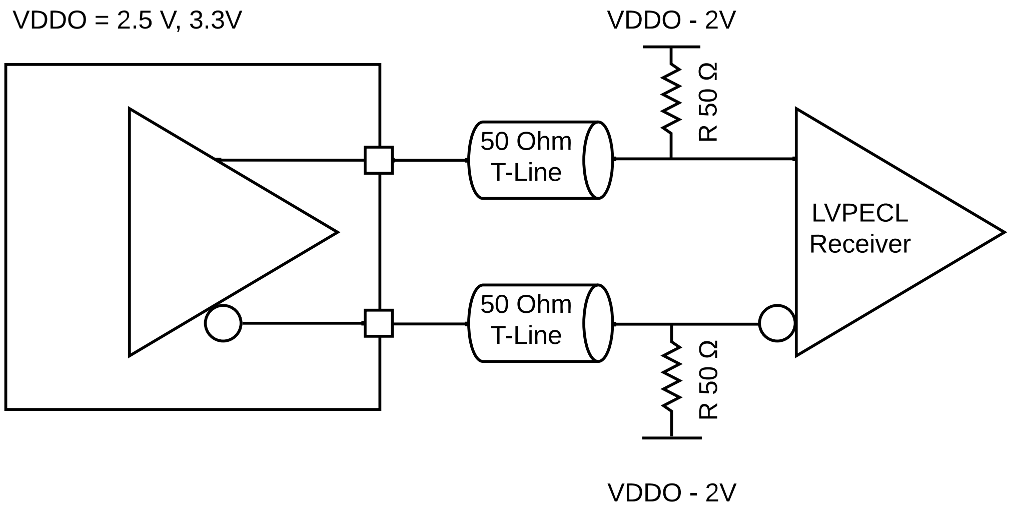
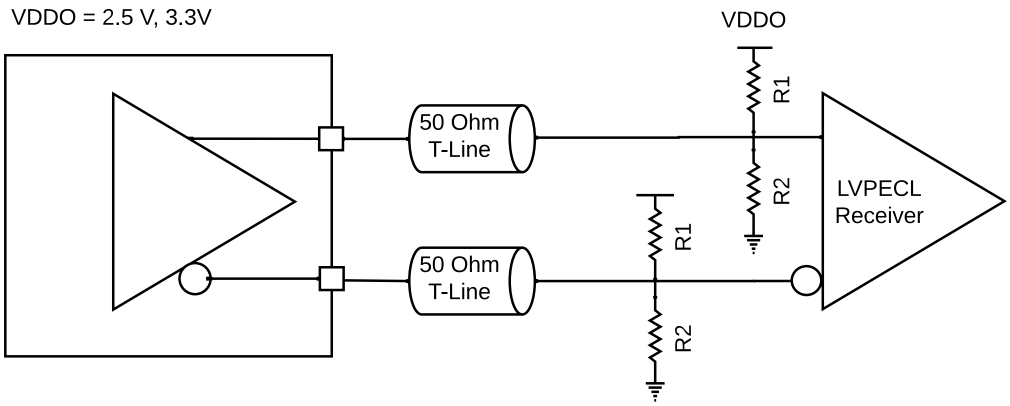


Figure 20 LVPECL DC Termination to VDDO – 2V



VDDO	R1	R2
2.5 V	250 Ω	62.5 Ω
3.3 V	127 Ω	82.5 Ω

Figure 21 LVPECL Alternate DC Termination: Thevenin Equivalent

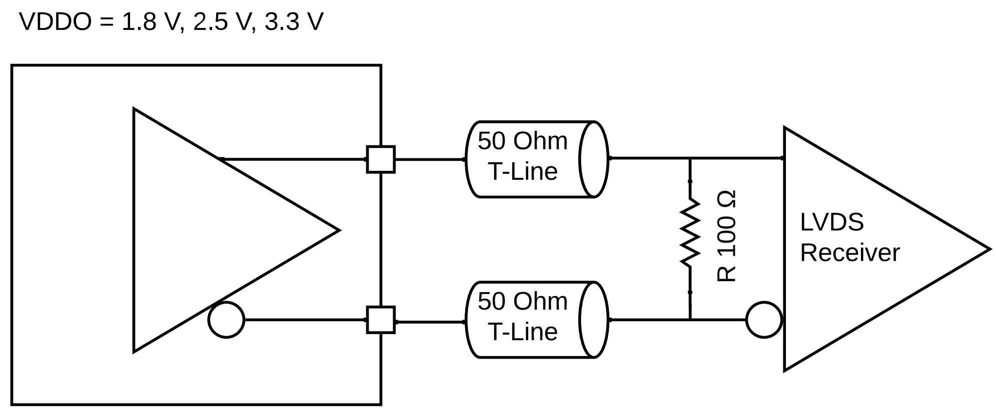


Figure 22 DC Coupled LVDS Termination

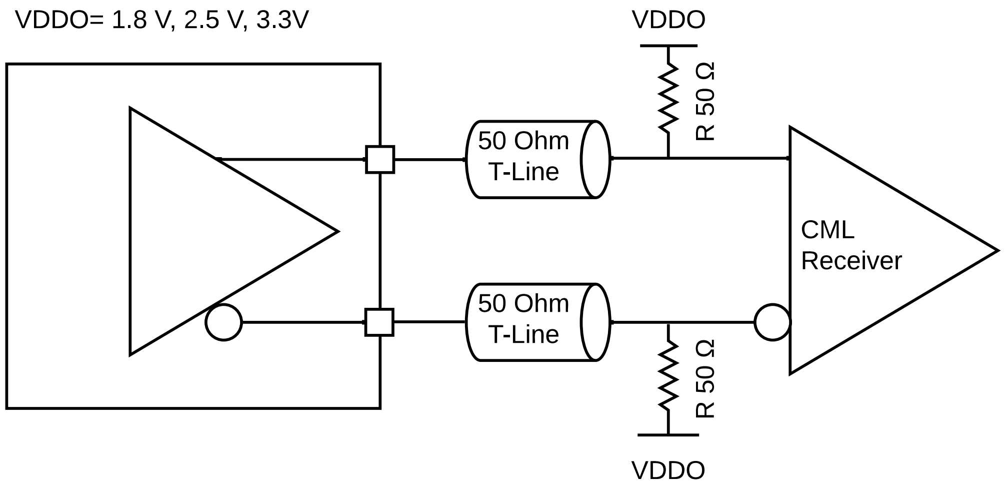


Figure 23 DC Coupled CML

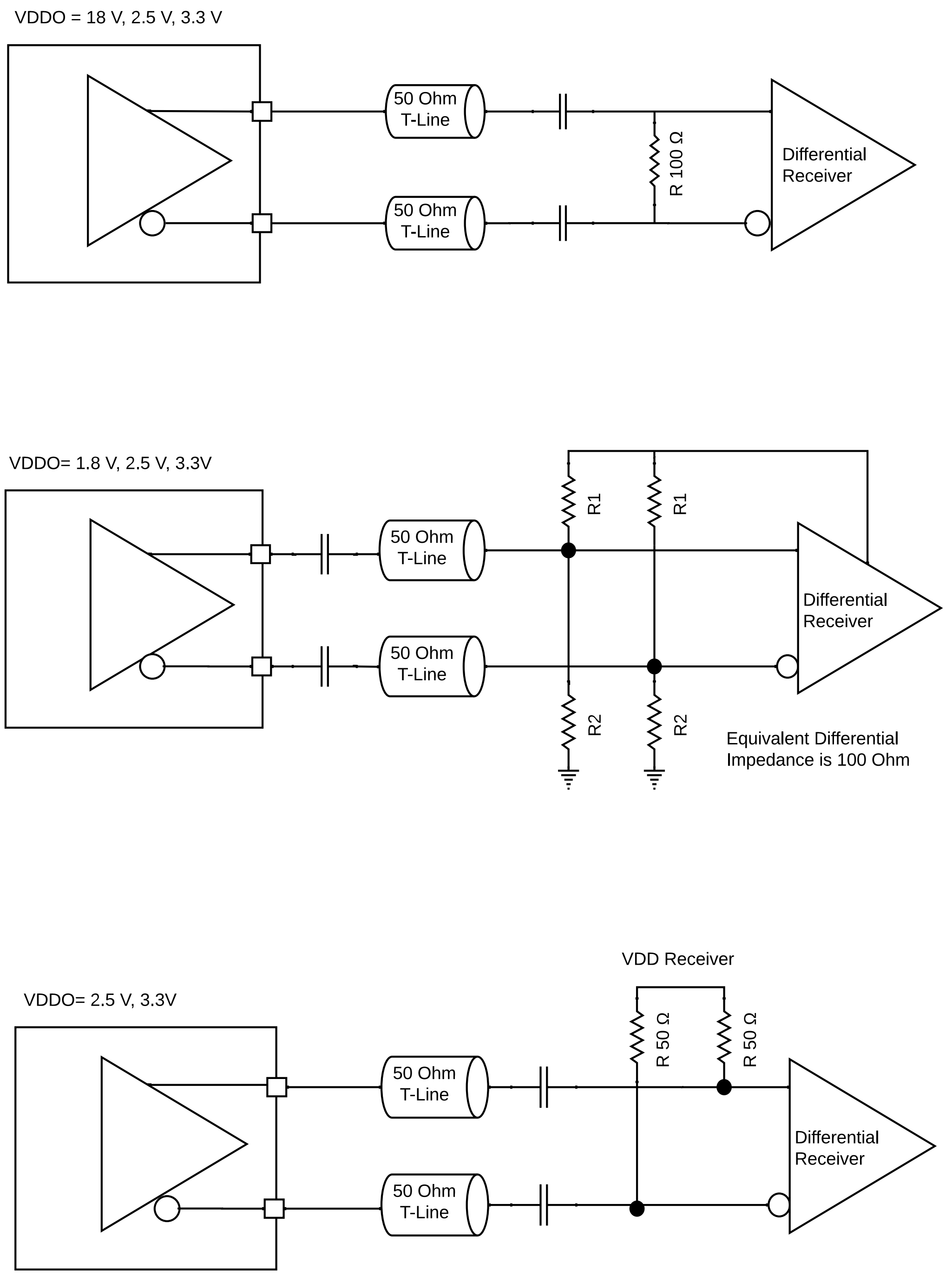


Figure 24 AC Coupled Receiver side resistive Termination options

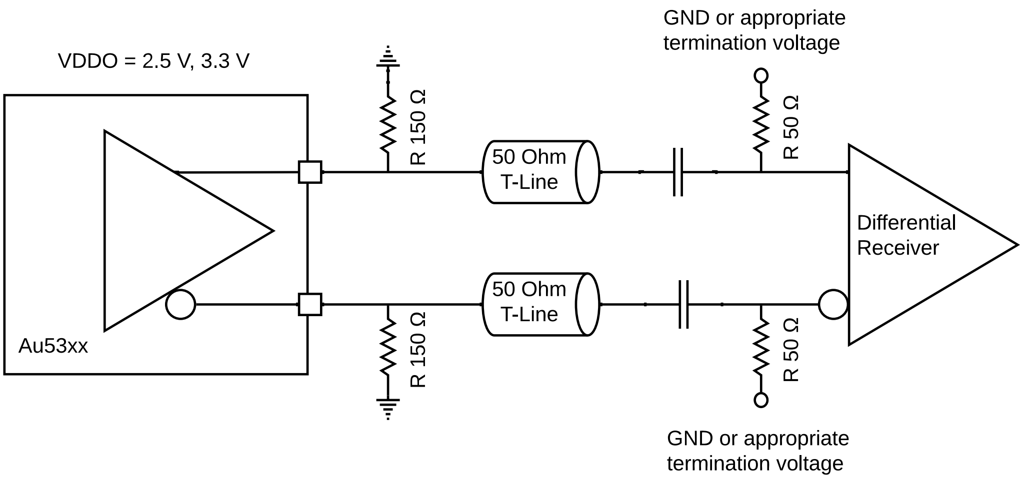
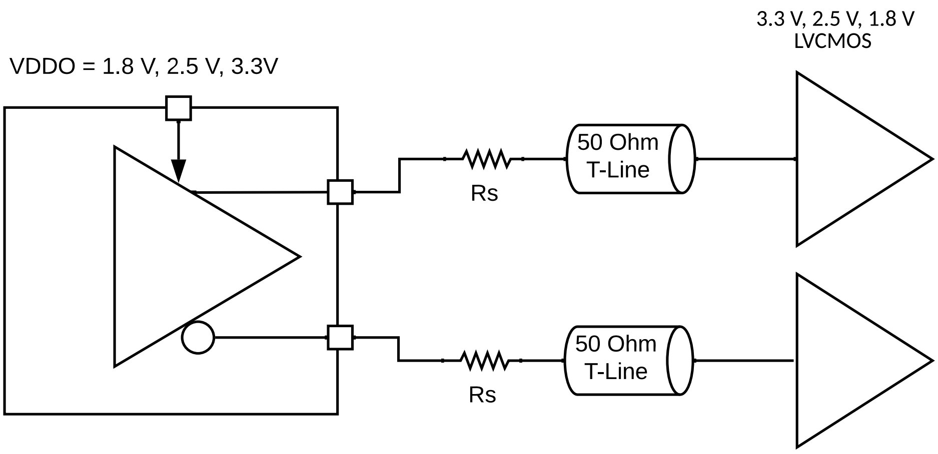


Figure 25 Alternate AC Coupled LVPECL with DC coupled resistors on Chip side



VDDO	R1
3.3 V	18 Ω
2.5 V	16 Ω
1.8 V	10 Ω

Figure 26 DC Coupled LVCMOS

VDDO = 1.8 V, 2.5 V, 3.3 V

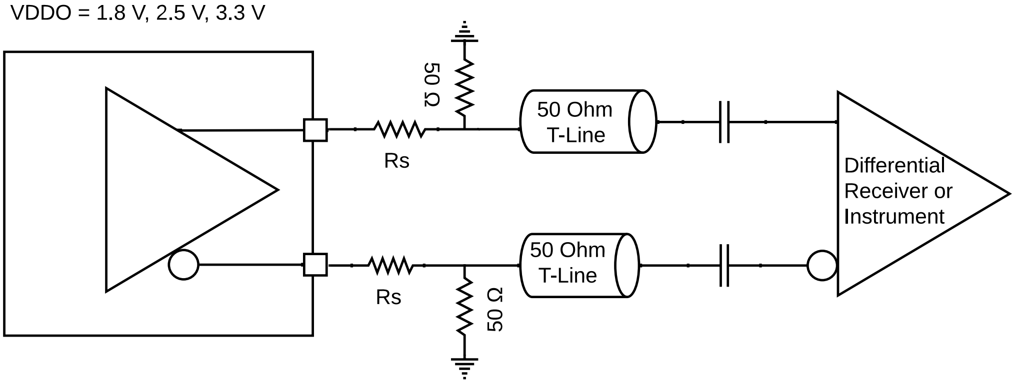


Figure 27 HCSL AC Coupled Termination. Source Terminated 50 Ohm

Note: Rs is sometimes used for limiting overshoot - Can be 0 Ohm

VDDO = 1.8 V, 2.5 V, 3.3 V

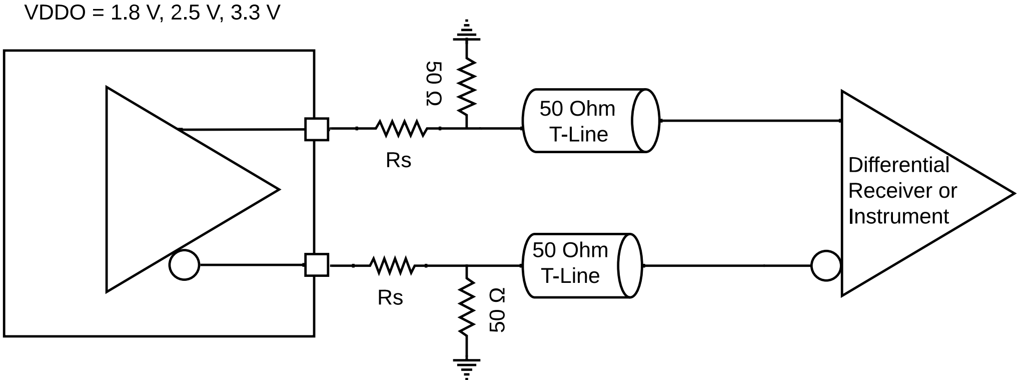


Figure 28 HCSL DC Coupled Termination. Source Terminated 50 Ohm

Note: Rs is sometimes used for limiting overshoot - Can be 0 Ohm

18 Input Termination Information

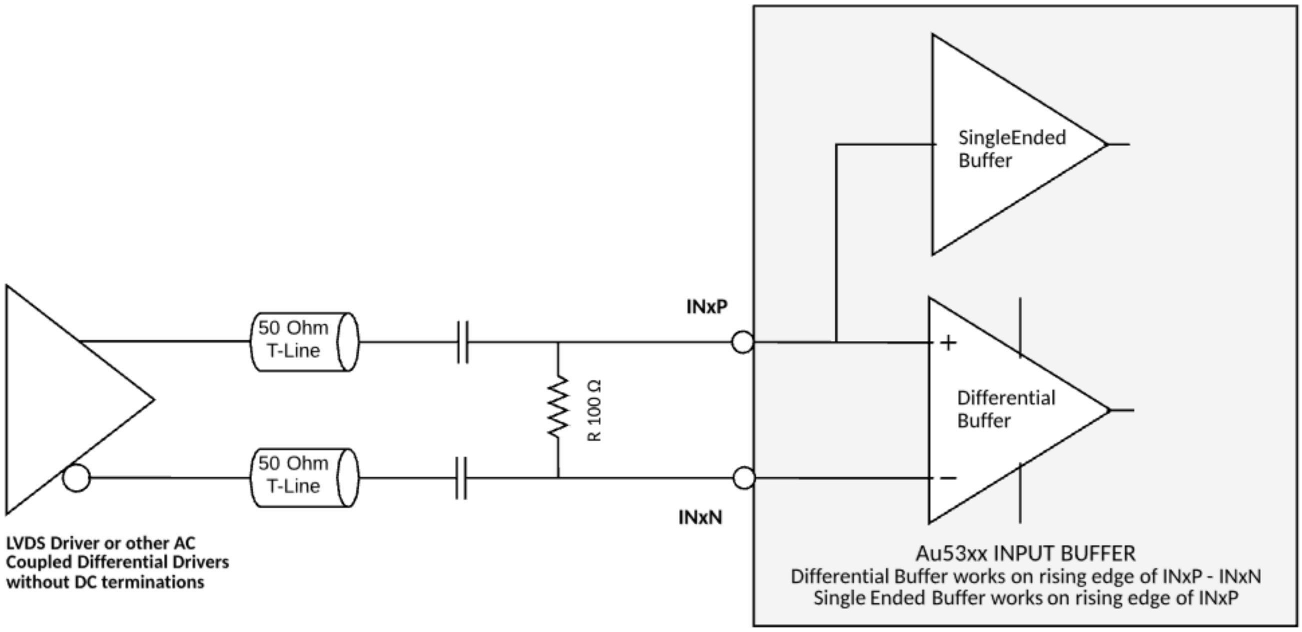


Figure 29 AC Coupled Differential LVDS Input / Other AC Coupled Driver without DC Terminations

Note: Uses Differential Buffer Pathway

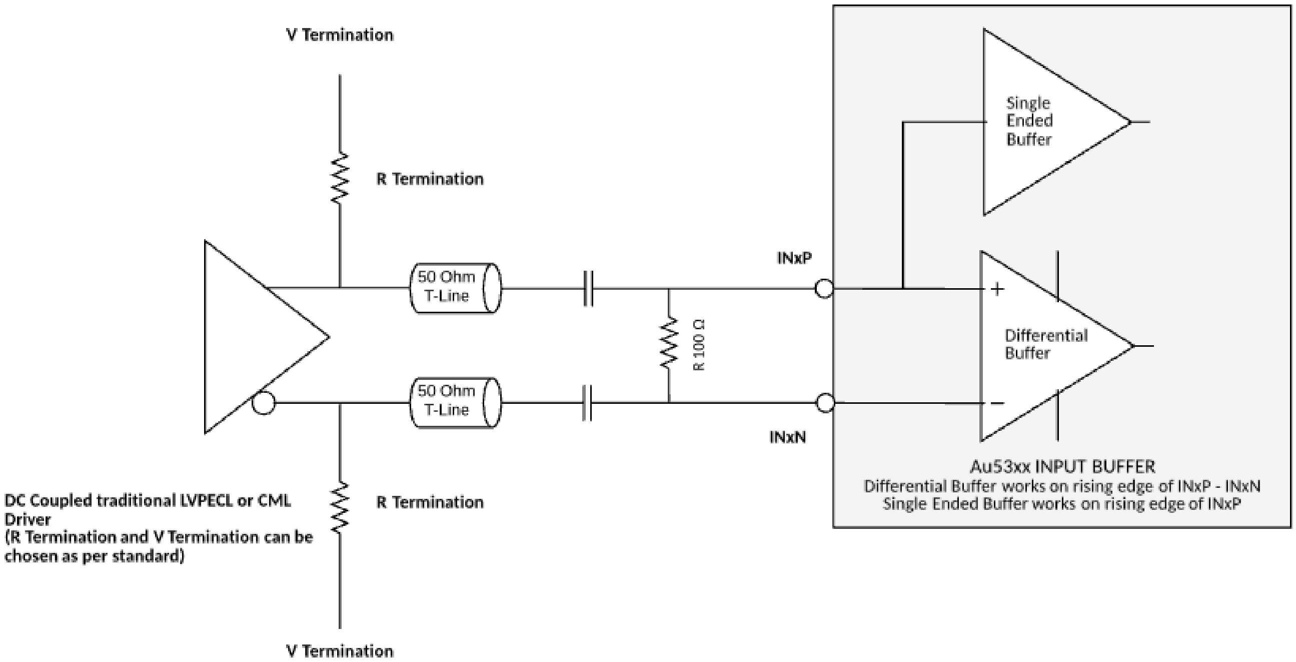
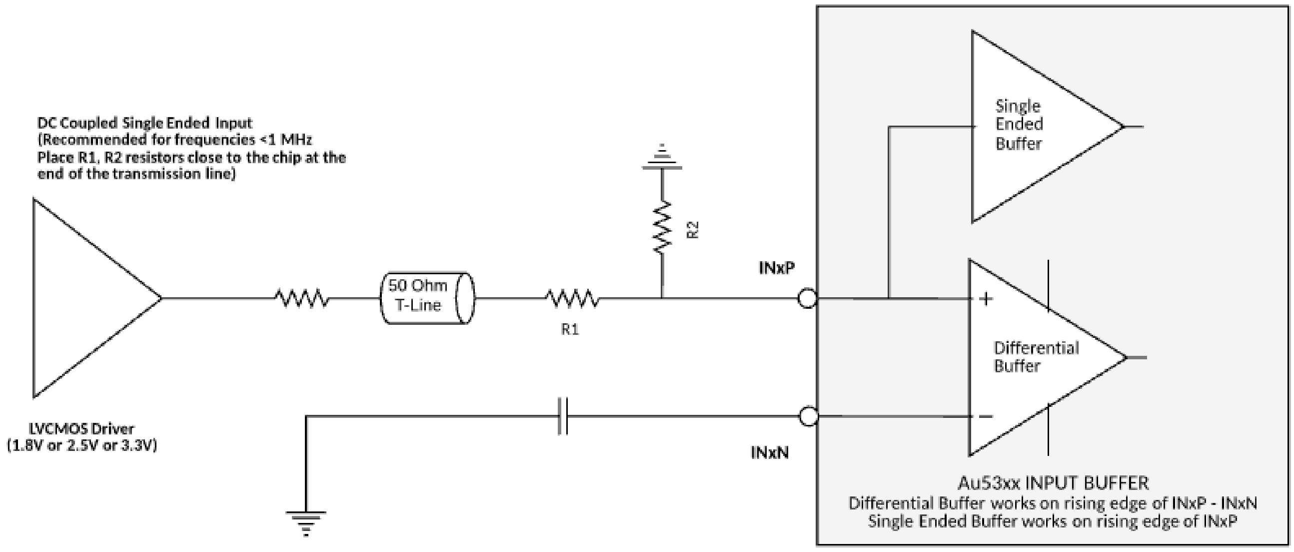


Figure 30 AC Coupled Differential LVPECL or CML

Note: Resistor and Voltage termination is as per the standard. Uses Differential Buffer Pathway. Please refer to the termination requirements of the driver.



Drive Supply	R1 (Ohms)	R2 (Ohms)
1.8 V	140	665
2.5 V	325	475
3.3 V	445	365

Figure 31 DC Coupled Single Ended Driver

Note: Uses Single Ended Buffer Pathway in DC Coupled Mode. Recommended for non-standard duty cycle applications. Please refer above table for the recommended resistor values for frequencies < 1 MHz.

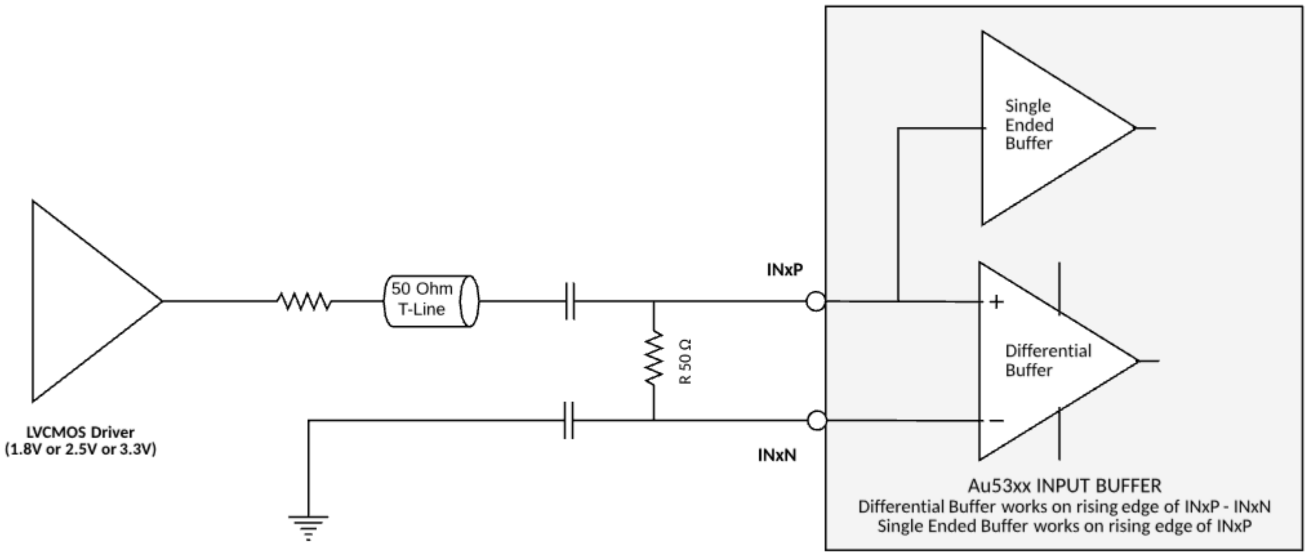


Figure 32 AC Coupled Single Ended Driver with 50 Ohm Termination on receiver (chip) side

Note: Uses Single Ended Buffer pathway in AC coupled mode.

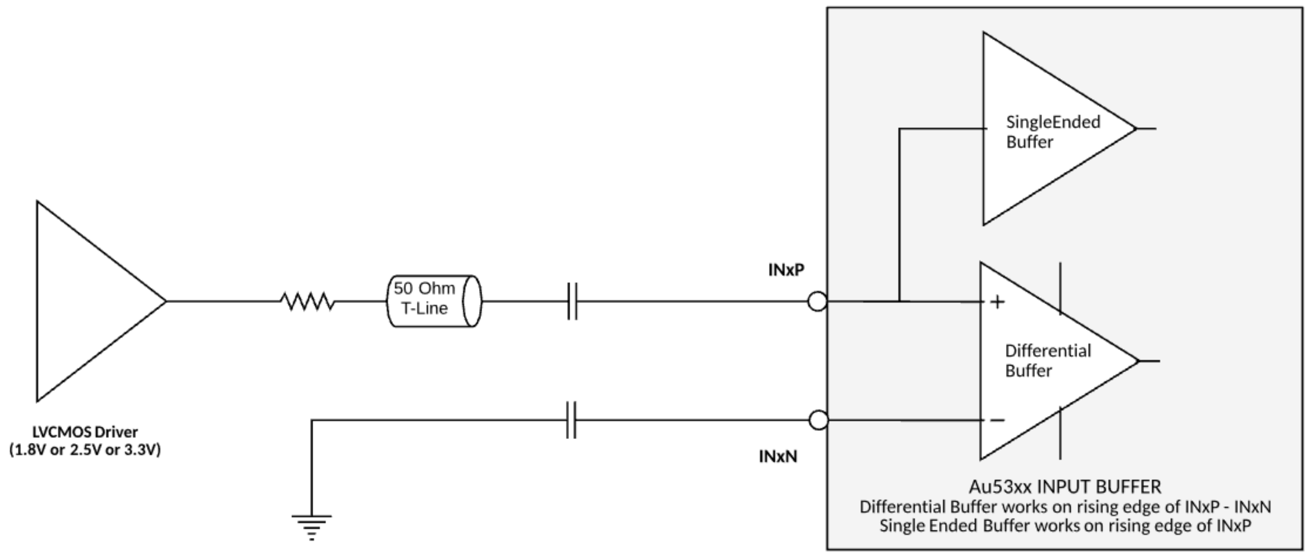


Figure 33 AC Coupled Single Ended LVCMOS input without 50 Ohm Termination

Note: Uses Single Ended Buffer pathway in AC Coupled Mode. The LVCMOS driver in this case needs to ensure source termination to match to the transmission line.

19 Crystal Pathway Connectivity Options

The CMOS XO/TCXO output and the termination components should be placed as close as possible to the X1/X2 pins

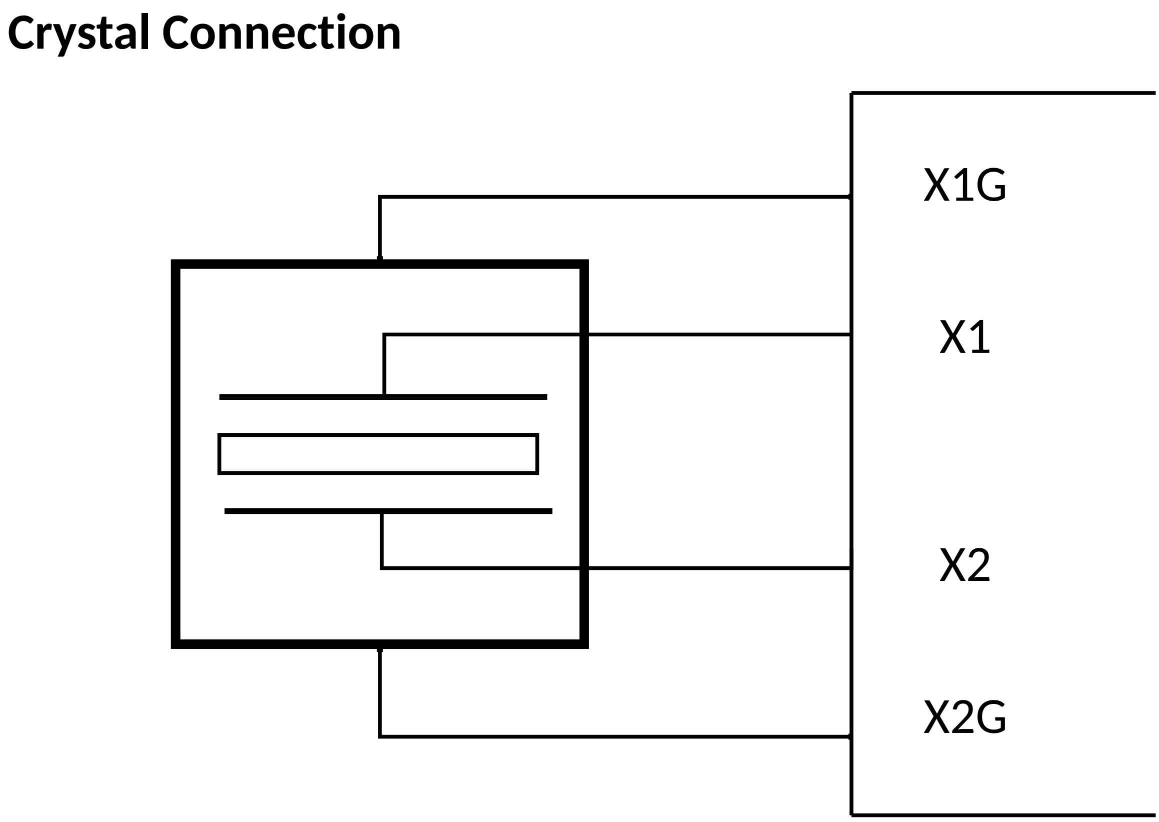
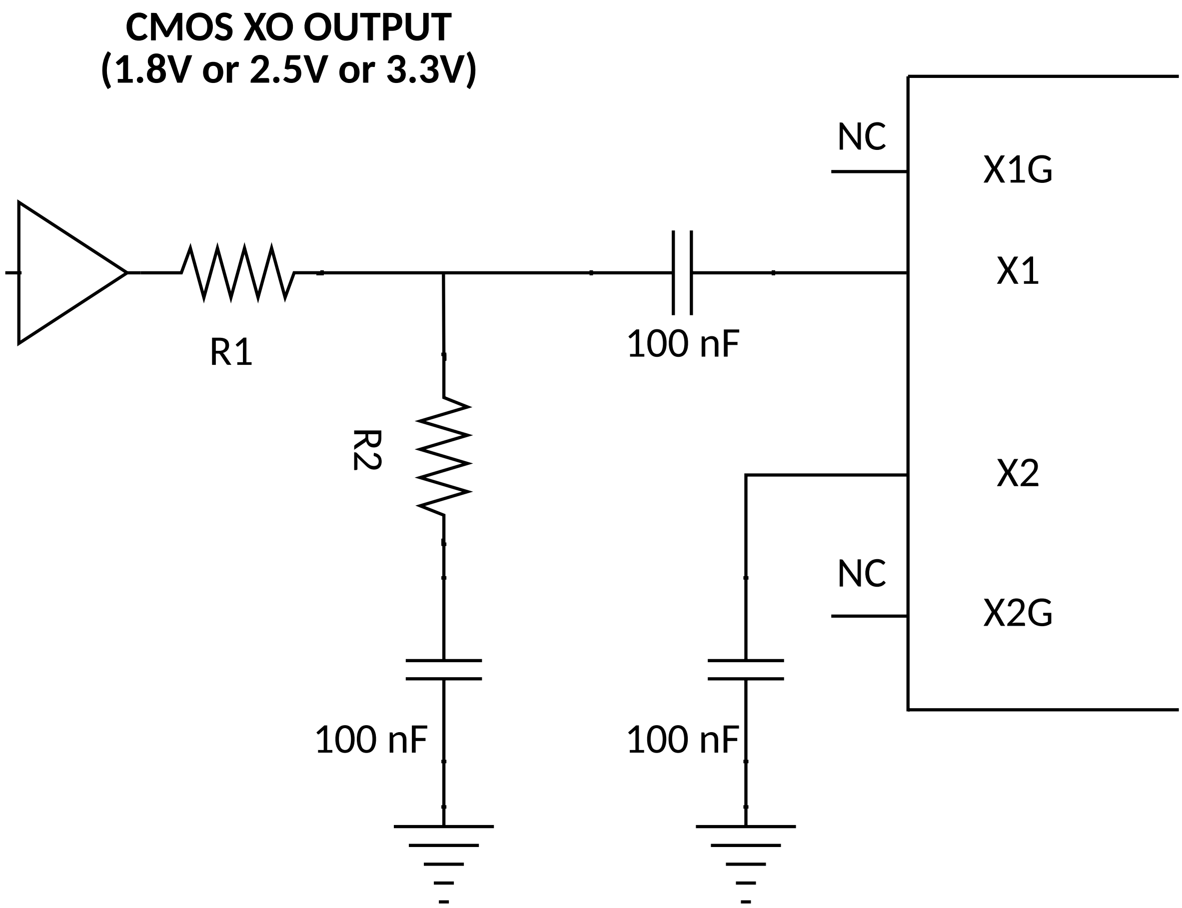


Figure 34 Crystal Connection



CMOS XO Driver Supply	R1 (Ohms)	R2 (Ohms)
1.8 V	0	DNP
2.5 V	274	732
3.3 V	453	549

Figure 35 CMOS XO Connection

Note:

R1 and R2 can be reduced by 2x to improve the slew rate. Additionally, the 5pF can be placed across R1 to improve the slew rate further if PCB routing length is significant from the attenuator network to the X1 pin.

Differential XO/Clock

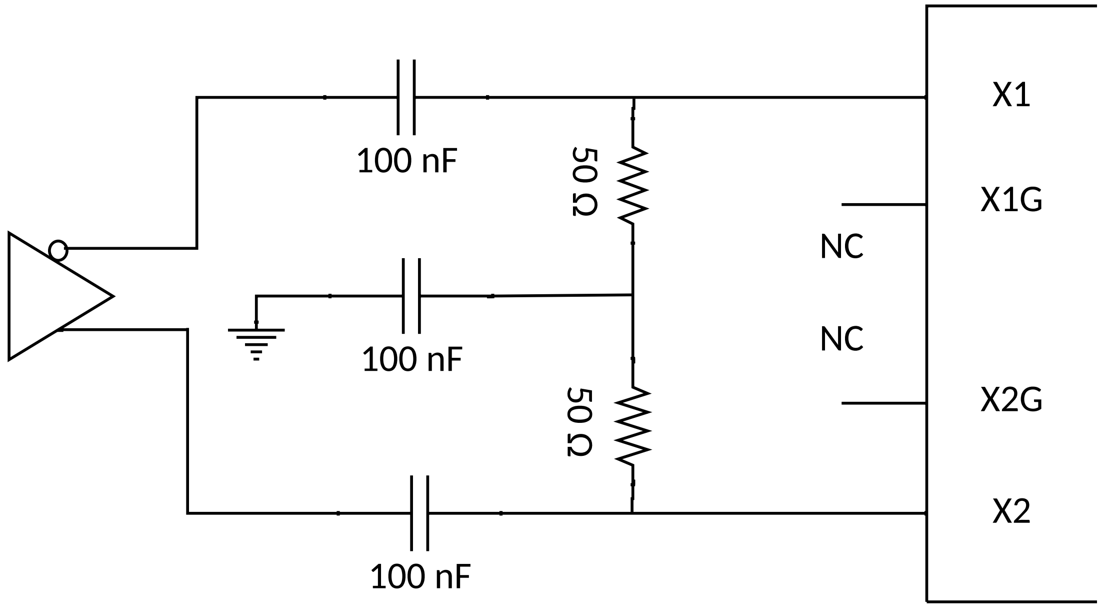


Figure 36 Differential XO Connection

20 Monitoring through the register map read back: Status and Notify

AU5329 provides various Status and Notify bits that can be accessed from the register map. Below are the details of the procedure to be followed to access the same.

The alarm registers are a set of three types of registers distributed between the various pages as described in the Table 21 and illustrated with examples in the section 20.2 and 20.3.

- The Status registers are the current dynamic status of a defect. The live status defects are active high with a '1' indicating the defect is present.
- The Notify registers are the sticky bits for a defect. Sometimes, we may get a very short pulse for the status and it may not be possible for the external user to capture the same. Hence a Notify register is provided. The Notify register is set to 1 whenever there is a rising edge of the corresponding status register. This is a sticky bit and stays at 1 till the user writes a 1 to that specific bit to clear it.
- Each notify has a masking bit to enable or disable its operation. The notify sticky bit operates only if the corresponding masking bit is set to 1. If the masking register bit is set to 0, notify will not be asserted even when status toggles. The default value for the mask register is 0xff so all the notify signals are enabled. Once the user writes a 1 to clear the notify, the notify bit can again go high on the next rising edge of the status.

These registers operate on the internal 4 MHz RC clock. When there is a defect (i.e. status) of any bit in register it gets asserted and de-asserted in a live mode. User can read the corresponding register location to see the current status at any time.

On the other hand, the default value of the Notify and Masking register is 0xff – user has to write 0xff to both these registers to clear them at the beginning and use all notifies.

The INTRB pin is used as a NOR operation of the selected notifies. The choice of the Notify listing that is used for the INTRB pin is selectable in the GUI when creating the profile. The sticky notifies that are selected and used for INTRB need to be cleared for restoring the INTRB to 1 for further sticky defect monitoring using this pin.

20.1 Tabular Listing

Table 21 details the name of the alarm, the page it is located in, the address in the page and the bit number in the address. In order to access a page of the register map, the particular page number has to be written to address 0xff. For instance, to either write to or read from page 1, first the user needs to write to 0xff a value of 0x01. Following this, any number of write or read operations can be done with page 1.

Table 21 Tabular Listing of Alarm Registers

S.No	Name of Signal	Description	Page Number	Register Address	Bit Number
1	xo_clkloss_dynamic_status	Xo clock dynamic status	00	0x02	2
2	xo_clkloss_dynamic_ntfy	Xo clock dynamic Notify (write 0x04 with 0x02 to enable)	00	0x03	2
3	pllb_lol_dyn_status	pll lol dynamic status	00	0x06	1
4	plld_lol_dyn_status	pll lol dynamic status	00	0x06	3
5	pllb_ho_frz_status	pll ho freeze dynamic status	00	0x06	5
6	plld_ho_frz_status	pll ho freeze dynamic status	00	0x06	7
7	pllb_lol_dyn_ntfy	pll lol dynamic Notify (write 0x08 with 0x02 to enable selectively, 0xff to enable all notifies in 0x07)	00	0x07	1
8	plld_lol_dyn_ntfy	pll lol dynamic Notify (write 0x08 with 0x08 to enable selectively, 0xff to enable all notifies in 0x07)	00	0x07	3

S.No	Name of Signal	Description	Page Number	Register Address	Bit Number
9	pllb_ho_frz_ntfy	pll ho freeze dynamic Notify (write 0x08 with 0x20 to enable selectively, 0xff to enable all notifies in 0x07)	00	0x07	5
10	plld_ho_frz_ntfy	pll ho freeze dynamic Notify (write 0x08 with 0x80 to enable selectively, 0xff to enable all notifies in 0x07)	00	0x07	7
11	In1_clock_loss_dyn_status	Clock Loss Dynamic Status	01	0x02	1
12	In2_clock_loss_dyn_status	Clock Loss Dynamic Status	01	0x02	2
13	In1_clock_loss_fd_dyn_status	Clock Loss + FD Dynamic Status	01	0x02	5
14	In2_clock_loss_fd_dyn_status	Clock Loss + FD Dynamic Status	01	0x02	6
15	In1_clock_loss_ntfy	Clock Loss Dynamic Notify (write 0x04 with 0x02 to enable selectively, 0xff to enable all notifies in 0x03)	01	0x03	0
16	In2_clock_loss_ntfy	Clock Loss Dynamic Notify (write 0x04 with 0x04 to enable selectively, 0xff to enable all notifies in 0x03)	01	0x03	2
17	In1_clock_loss_ntfy	Clock Loss with Fd_dynamic Notify (write 0x04 with 0x20 to enable selectively, 0xff to enable all notifies in 0x03)	01	0x03	5
18	In2_clock_loss_ntfy	Clock Loss with Fd_dynamic Notify (write 0x04 with 0x40 to enable selectively, 0xff to enable all notifies in 0x03)	01	0x03	6
19	In1_fd_fine_dyn_status	Frequency drift dynamic status	01	0x06	1
20	In2_fd_fine_dyn_status	Frequency drift dynamic status	01	0x06	2
21	In1_fd_coarse_dyn_status	Frequency drift dynamic status	01	0x06	5
22	In2_fd_coarse_dyn_status	Frequency drift dynamic status	01	0x06	6
23	In1_fd_fine_ntfy	Frequency drift dynamic notify (write 0x08 with 0x02 to enable selectively, 0xff to enable all notifies in 0x07)	01	0x07	1
24	In2_fd_fine_ntfy	Frequency drift dynamic notify (write 0x08 with 0x04 to enable selectively, 0xff to enable all notifies in 0x07)	01	0x07	2
25	In1_fd_coarse_ntfy	Frequency drift dynamic notify (write 0x08 with 0x20 to enable selectively, 0xff to enable all notifies in 0x07)	01	0x07	5
26	In2_fd_coarse_ntfy	Frequency drift dynamic notify (write 0x08 with 0x40 to enable selectively, 0xff to enable all notifies in 0x07)	01	0x07	6
27	fast_lock_dynamic_status_pllb	PLL Fast lock dynamic status	0b	0x06	1
28	fast_lock_dynamic_status_plld	PLL Fast lock dynamic status	0d	0x06	1
29	fast_lock_dynamic_ntfy_pllb	PLL Fast lock dynamic Notify (write 0x08 with 0x02 to enable selectively, 0xff to enable all notifies in 0x07)	0b	0x07	1

S.No	Name of Signal	Description	Page Number	Register Address	Bit Number
30	fast_lock_dynamic_ntfy_plld	PLL Fast lock dynamic Notify (write 0x08 with 0x02 to enable selectively, 0xff to enable all notifies in 0x07)	0d	0x07	1
31	ho_valid_dynamic_status_pllb	PLL Ho Valid dynamic status	0b	0x06	2
32	ho_valid_dynamic_status_plld	PLL Ho Valid dynamic status	0d	0x06	2
33	ho_valid_dynamic_ntfy_pllb	PLL Ho Valid dynamic Notify (write 0x08 with 0x04 to enable selectively, 0xff to enable all notifies in 0x07)	0b	0x07	2
34	ho_valid_dynamic_ntfy_plld	PLL Ho Valid dynamic Notify (write 0x08 with 0x04 to enable selectively, 0xff to enable all notifies in 0x07)	0d	0x07	2

20.2 Examples for Live Status Read Back

Some examples are presented based on the table above for reading the live status of the defects.

In the pseudo code presented below:

wr_cmd(address, data): refers to a “Write Command” where the corresponding data is written in to the specified register address

x= rd_cmd(address): refers to a “Read Command” where the corresponding data is read from the specified register address and stored in the variable ‘x’

y >> x: denotes a bit wise right shift on the number y by x bit locations

y << x:denotes a bit wise left shift on the number y by x bit locations

& is the logical AND operation (bit wise)

Dynamic registers to read the various alarm registers in the RealTime page.

1. Input Clocks CL and FD Related Real Time live status read back:

wr_cmd(0xff, 0x01)   Program the CLKMON_SYS page number

Clock Loss dynamic status

clock_loss_dyn_status = rd_cmd(0x02) & 0xff

(clock_loss_dyn_status >> 1) & 0x01   IN1 Status for CL, Read bit position [1]

(clock_loss_dyn_status >> 2) & 0x01   IN2 Status for CL, Read bit position [2]

Frequency Drift dynamic status

fd_fine_dyn_status = rd_cmd(0x06) & 0x0f

fine = (fd_fine_dyn_status >> 1) & 0x01   IN1 Status for Fine FD, Read bit position [1]

fine = (fd_fine_dyn_status >> 2) & 0x01   IN2 Status for Fine FD, Read bit position [2]

fd_coarse_dyn_status = rd_cmd(0x06) >> 4

coarse = (fd_coarse_dyn_status >> 1) & 0x01   IN1 Status for Coarse FD, Read bit position [5]

coarse = (fd_coarse_dyn_status >> 2) & 0x01   IN2 Status for Coarse FD, Read bit position [6]

2. PLL Related Real Time live status read back:

wr_cmd(0xff, 0x00) Program the GENERIC_SYS page number, Page 0

pll_lol_ho_freeze_dyn_status = rd_cmd(0x06) & 0xff

PLL Lock Loss dynamic status

(pll_lol_ho_freeze_dyn_status >> 1) & 0x01 PLLB Status for LL, Read bit position [1]

(pll_lol_ho_freeze_dyn_status >> 3) & 0x01 PLLD Status for LL, Read bit position [3]

Holdover Status

(pll_lol_ho_freeze_dyn_status >> (1 + 4)) & 0x01 PLLB Status for HO, Read bit position [5]

(pll_lol_ho_freeze_dyn_status >> (3 + 4)) & 0x01 PLLD Status for HO, Read bit position [7]

3. XO clock loss Related Real Time live status read back:

CLOS_X1X2, XO Clock Loss

wr_cmd(0xff, 0x00) Program the GENERIC_SYS page number, Page 0

clos_x1x2 = rd_cmd(0x02) & 0x04 XO CL Status, Read bit position [2]

20.3 Examples of Sticky Bit Clearing

As described earlier, the sticky notify bits are cleared by writing a ‘1’ to the corresponding notify bit itself. The notify bit by itself is enabled by writing a ‘1’ to the corresponding mask bit.

In the pseudo code presented below, rmw_cmd(addr,bit_loc,no_of_bits,data): denotes the read/modify/write operation where no_of_bits number of bits at bit_loc location (denoted as 7:0) is replaced with the data at address location addr.

```
def clr_intb_XO_CL():
    This function is used to clear the sticky notify for XO Clock Loss
    Write the page number
    wr_cmd(0xff, 0)
    Information to clr  Page 0: reg03[2]=1
    addr       = 0x3
    bit_loc    = 2
    no_of_bits = 1
    data       = 1
    rmw_cmd(addr,bit_loc,no_of_bits,data)

def clr_intb_LOL_HO_Freeze():
    This function is used to clear the sticky notify for loss of lock and holdover notify for all PLLs
    Write the page number
    wr_cmd(0xff, 0)
     Information to clr  Page 0: reg07[7:0] = 0xff
    addr       = 0x7
    bit_loc    = 7
    no_of_bits = 8
    data       = 0xff
    rmw_cmd(addr,bit_loc,no_of_bits,data)

def clr_intb_CL():
    This function is used to clear the sticky notify for clear clock loss notify
    Write the page number
    wr_cmd(0xff, 1)
     Information to clr   Page1: reg03[3:0]=0x0f
    addr       = 0x3
    bit_loc    = 3
    no_of_bits = 4
    data       = 0x0f
    rmw_cmd(addr,bit_loc,no_of_bits,data)

def clr_intb_drift():
    This function is used to clear the sticky notify for clear drift notify
    Write the page number
    wr_cmd(0xff, 1)
    Information to clr  Page1: reg07[7:0]=0xff
    addr       = 0x7
    bit_loc    = 7
    no_of_bits = 8
    data       = 0xff
    rmw_cmd(addr,bit_loc,no_of_bits,data)

def clr_intrb():

    This is the main clear function
    which calls the 4 clear functions

    clr_intb_XO_CL()
    clr_intb_LOL_HO_Freeze()
    clr_intb_CL()
    clr_intb_drift()
```

21 Device Initialization for non-programmed device

This section describes a device initialization flow chart for an unlocked device. An unlocked device is a device on which the NVM is not programmed and where an autonomous wake up does not happen. It is assumed that the user is using a device that is not programmed for the description in this section.

The AU5329 device register initialization flowchart for Master Control Page and the Slave pages is as below. As explained in the chip functional description in the data sheet, the following is the sequence of the wake up of the various sub systems in the chip.

- First, the master control (Page 0) is initialized and programmed. The slaves are powered up at this stage.
- Next, the Input System (Page 2) is initialized and programmed for cases where at least one input is enabled.
- Next, the Clock Monitor System (Page 1) is initialized and programmed for cases where at least one input is enabled.
- Next, the Output System (Page 3) is initialized and programmed.
- Finally, the PLLs that are expected to be used for the particular profile (PLLs B, D correspond to Pages B, D) are initialized and programmed.

The register 0xFF is written with the Page Number the user would like to access. The chip changes the current page once the user has written the register 0xFF. The page numbers corresponding to each slave and master are described in the respective table in the data sheet. At any point in time the register 0xFF can be read to find out the current page.

Please note that the flow chart in the following pages is to be used together with the sequence of register writes that are obtained from the GUI with the “Save NVM” button after loading a profile. This file obtained from the GUI describes the set of registers to be written in the exact order and with appropriate delays. This file obtained from the GUI is the master sequence to be followed for programming a part. The figures on the following pages describe the flow of the same register write sequence file using flow charts.

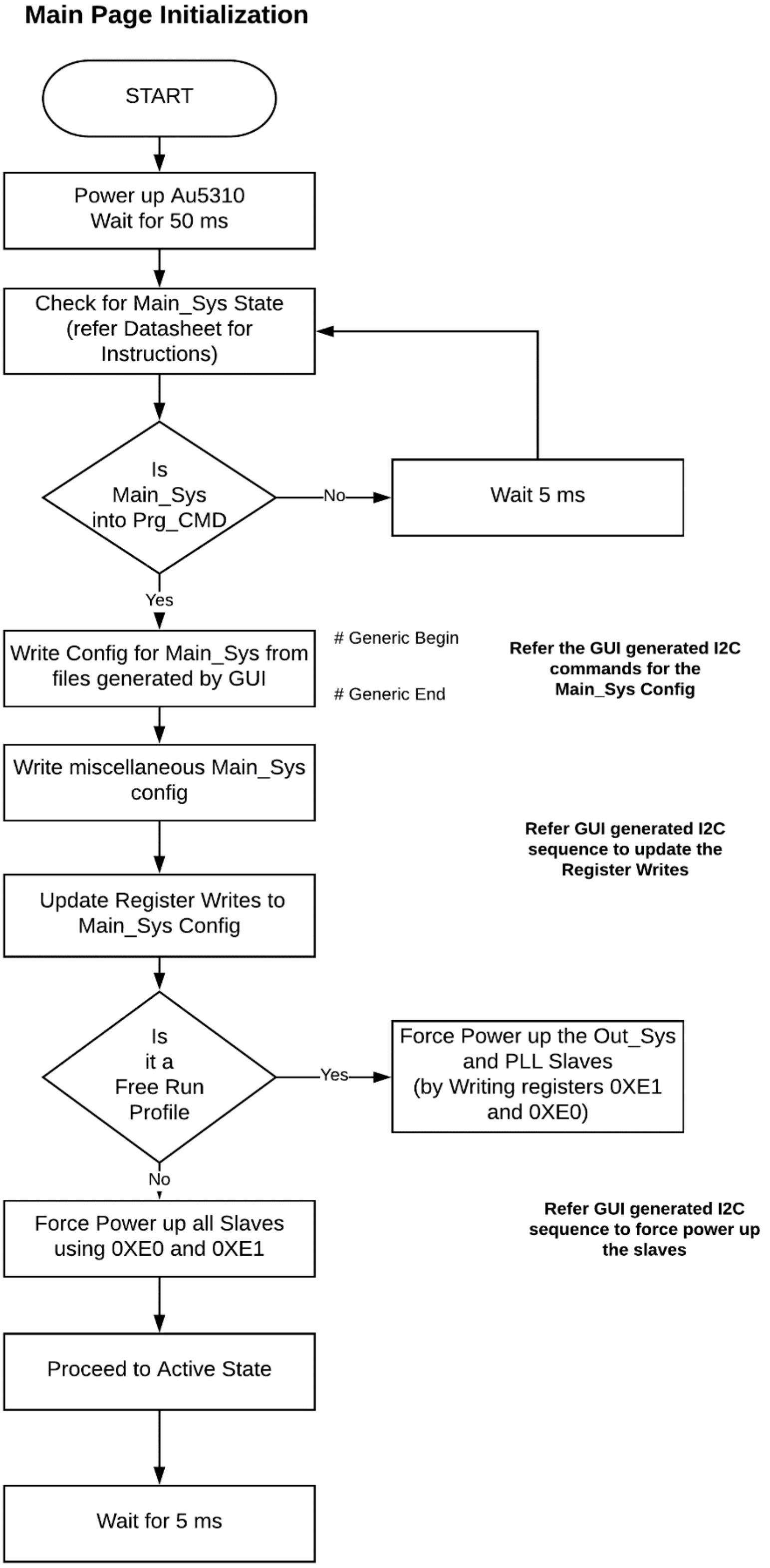


Figure 37 STEP 1: Initialize the Main Page- Page 0.

We can move to Page 0 by writing 0x00 to the address 0xFF. Read the current page at any time by reading the contents of the register 0xFF

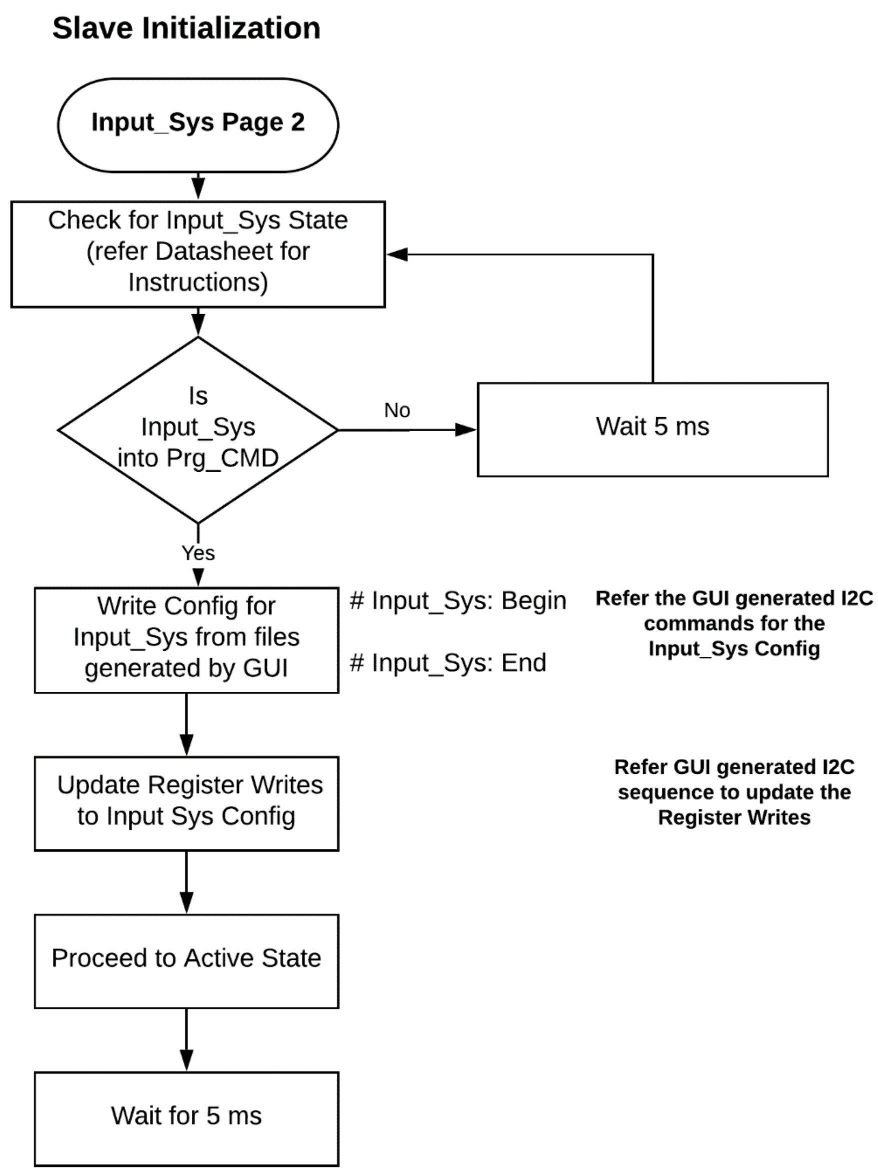


Figure 38 STEP 2: Initialize the Input System Page- Page 2.

We can move to Page 2 by writing 0x02 to the address 0xFF. Read the current page at any time by reading the contents of the register 0xFF. This page is not initialized for a purely free run profile where no inputs are engaged.

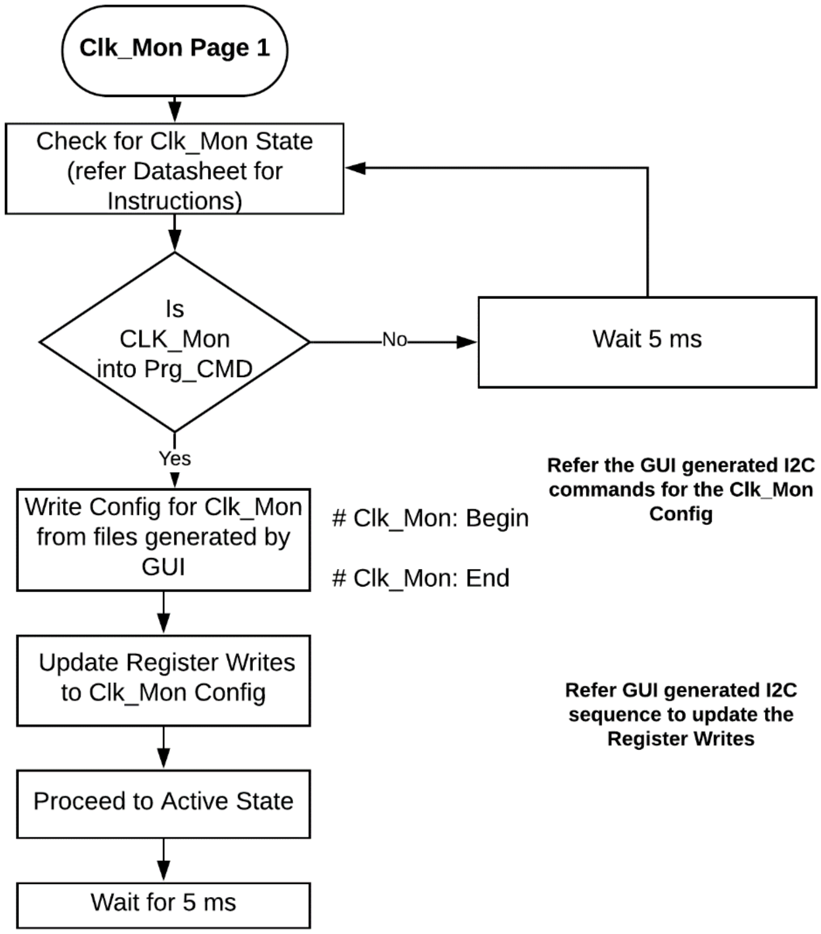


Figure 39 STEP 3: Initialise the Clock Monitor System Page- Page 1.

We can move to Page 1 by writing 0x01 to the address 0xFF. Read the current page at any time by reading the contents of the register 0xFF. This page is not initialized for a purely free run profile where no inputs are engaged.

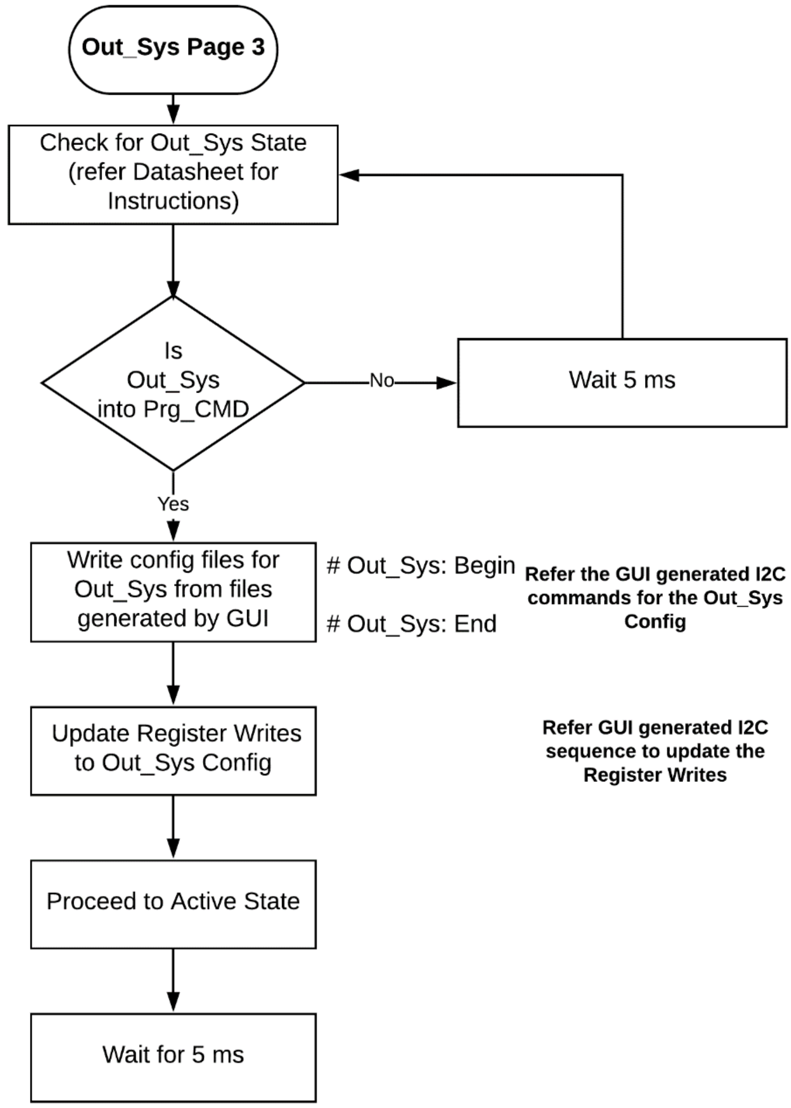


Figure 40 STEP 4: Initialise the Output System Page- Page 3.

We can move to Page 3 by writing 0x03 to the address 0xFF. Read the current page at any time by reading the contents of the register 0xFF.

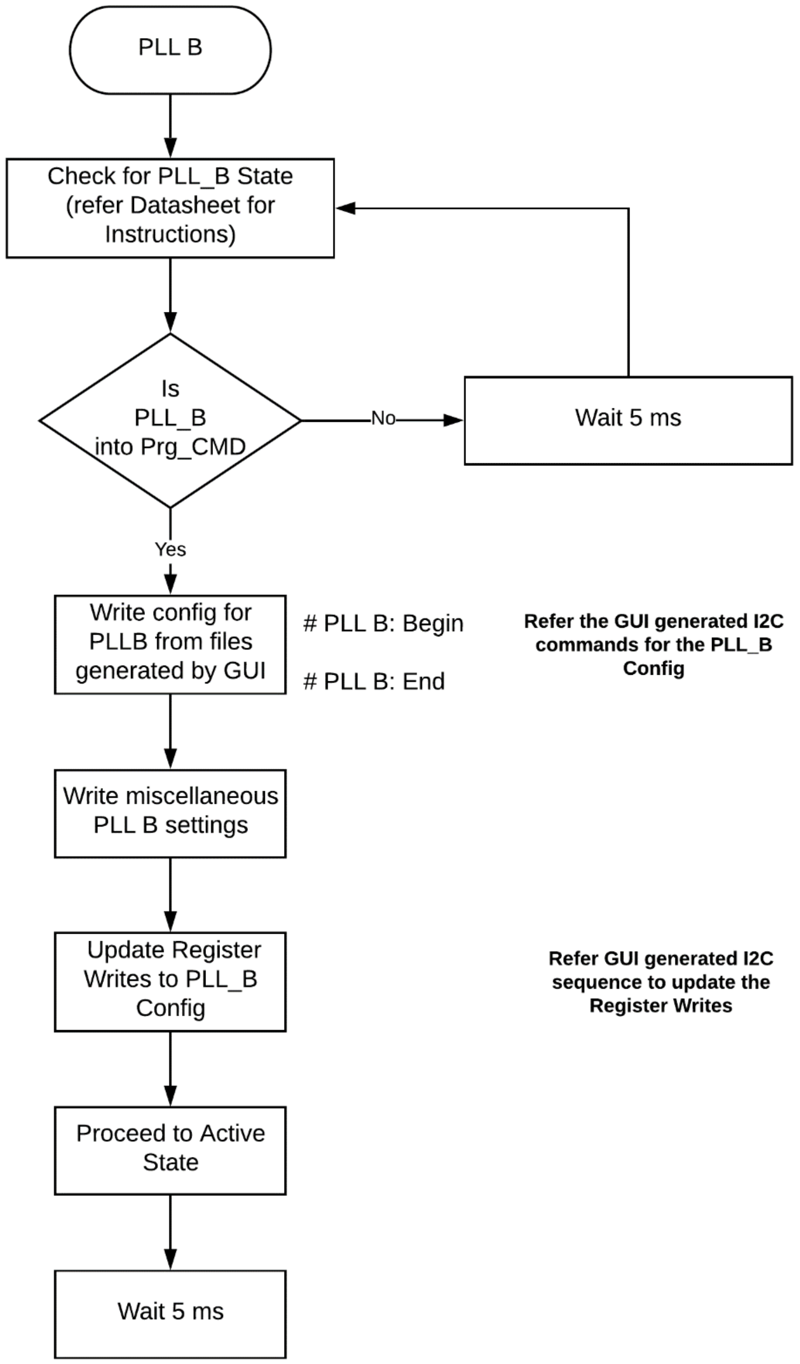


Figure 41 STEP 6: Initialise the PLL B System Page- Page B.

We can move to Page B by writing 0x0B to the address 0xFF. Read the current page at any time by reading the contents of the register 0xFF. Initialize PLL B only if it is used in the profile being used.

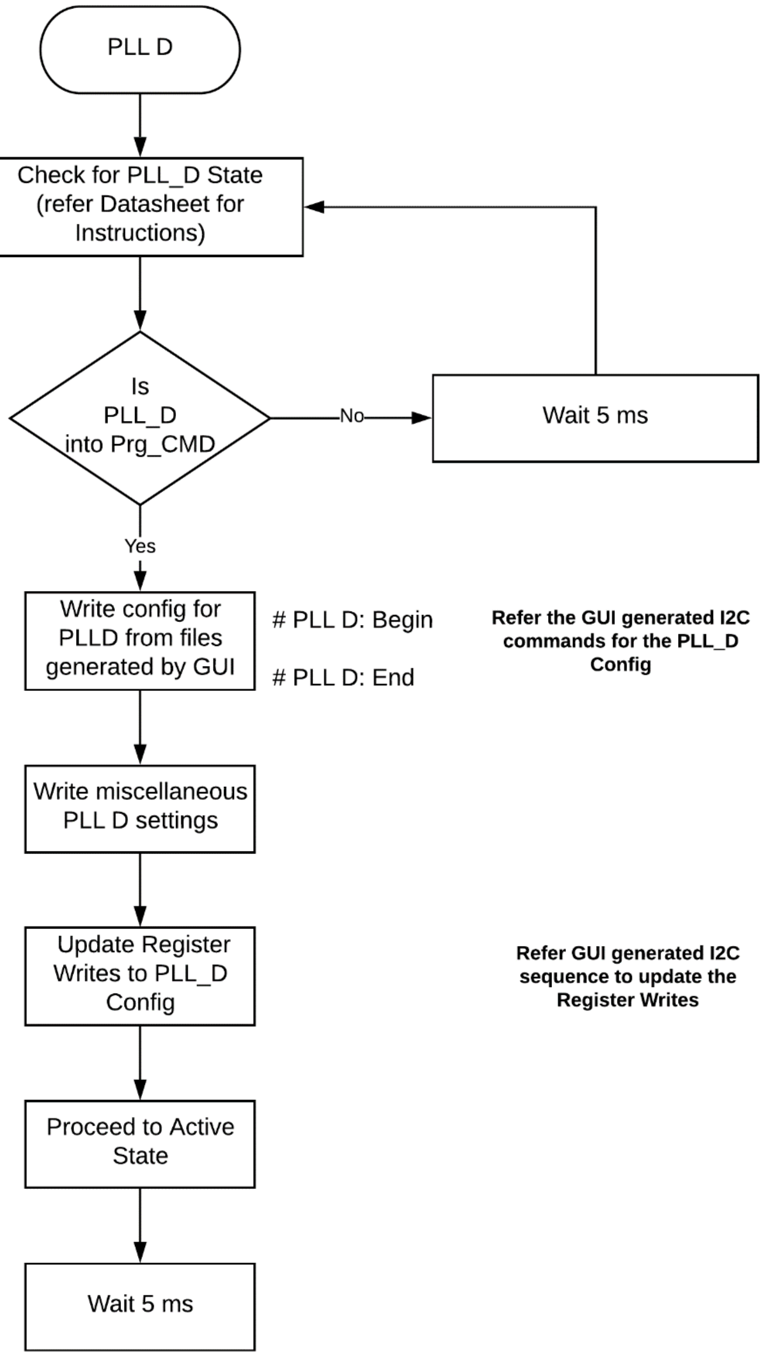


Figure 42 STEP 8: Initialise the PLL D System Page- Page D.

We can move to Page D by writing 0x0D to the address 0xFF. Read the current page at any time by reading the contents of the register 0xFF. Initialize PLL D only if it is used in the profile being used.

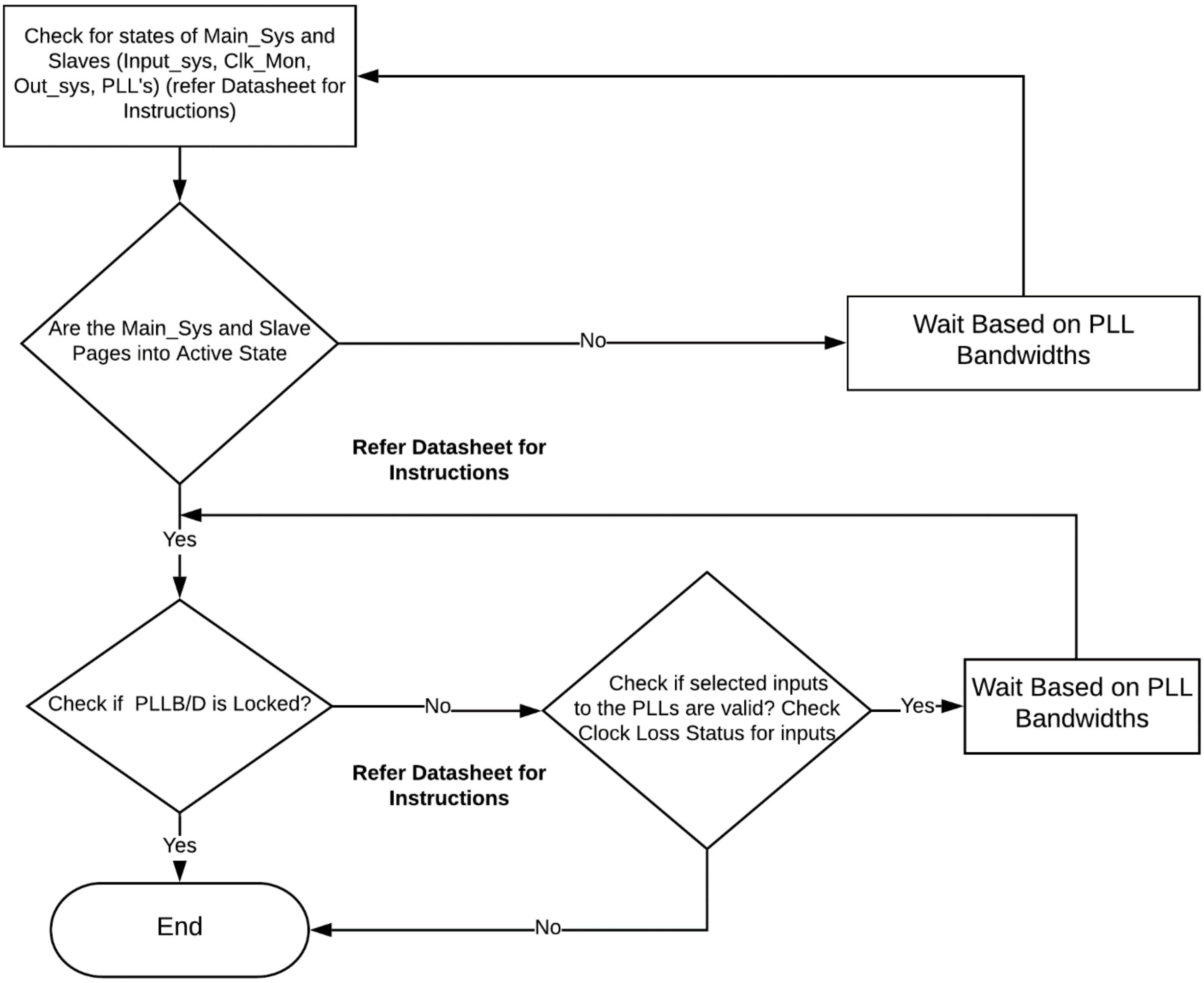


Figure 43 Active State Confirmation

Confirm from each page that was initialized that their respective state machines are in Active State. Check Lock Status of PLLs that are enabled.

22 Monitoring the Status for Master and Slave Pages

Once the AU5329 is powered up, the NVM register contents are read, it is preferable to provide sufficient time delay for the read operation.

As explained in the data sheet above, we need to read that the Master Control Page 0 is in the Program Command Wait State (PRG_CMD State) before starting to write to the part. Please use the register write sequence from the GUI generated from the "Save NVM" button to write in to the device.

Once the entire sequence to program the profile has been written, each Page (Pages 0,1, 2, 3 PLL Pages B, D based on which PLLs are enabled) can be checked to ensure they are in the ACTIVE state.

The sequence that needs to be followed to check the status of the Master and Slave Pages is as below:

MAIN PAGE (0)

1) Write 0xFF register to 0x00. Go to Page 0
2) Write 0XD1 register on Page 0 to 0x40.
3) Read the register 0xD0.

If read data = 9 (Main Sys is in PRG_CMD state)
read data = 0 (Main Sys is in IDLE state)
read data = 36 (Main Sys is in Active state)

CLK MON PAGE (1)

1) Write 0xFF register to 0x01. Go to Page 1
2) Write 0XD1 register on Page 1 to 0x40.
3) Read 0xD0[3:0]

If read data = 5 (CLK MON is in PRG_CMD state)
read data = 0 (CLK MON is in IDLE state)
read data = 12 (CLK MON is in Active state)

INPUT_SYS PAGE (2)

1) Write 0xFF register to 0x02. Go to Page 2
2) Write 0XD1 register on Page 2 to 0x40.
4) Read 0xD0[4:0]

3) If read data = 7(INPUT_SYS is in PRG_CMD state)
read data = 0 (INPUT_SYS is in IDLE state)
read data = 23 (INPUT_SYS is in Active state)

OUT_SYS PAGE (3)

1) Write 0xFF register to 0x03. Go to Page 3
2) Write 0XD1 register on Page 3 to 0x60.
3) Read 0xD0[4:0]

If read data = 12(OUT_SYS is in PRG_CMD state)
read data = 0 (OUT_SYS is in IDLE state)
read data = 20 (OUT_SYS is in Active state)

PLL PAGE (A, B, C, D)

1) Write 0xFF register to 0x0A(PLLA). (0x0A - PLLA, 0x0B – PLLB, 0x0C-PLLC, 0X0D-PLLD)
2) Write 0XD1 register on Page A to 0x20.
3) Read 0xD0.

If read data = 8(PLL PAGE is in PRG_CMD state)
read data = 0 (PLL PAGE is in IDLE state)
read data = 48 (PLL PAGE is in Active state)

Monitoring the Loss of Lock Status for PLL

The sequence that needs to be followed to monitor the dynamic LOL status of the PLL's is as below:

1) Write 0xFF register to 0x0A(PLLA). (0x0A - PLLA, 0x0B – PLLB, 0x0C-PLLC, 0X0D-PLLD)
2) Write 0x04 register on PLL Page to 0x01. Remove the mask for lock loss notify status
3) Read 0x02[0] LOL dynamic status_PLLA_B_C_D

If read data = 1 (Loss of Lock is asserted: PLL is not locked)
read data = 0 (PLL is locked)

Monitoring the Hold Over Status for PLLs

The sequence that needs to be followed to monitor the Hold Over status of the PLL's is as below:

1) Write 0xFF register to 0x0A(PLLA). (0x0A - PLLA, 0x0B – PLLB, 0x0C-PLLC, 0X0D-PLLD)
2) Read 0x17[7] To check if the PLL Outer Loop is Enabled/Disabled

If read data = 1 It indicates a Free Run Profile --- The PLL's will always be in Hold Over

read data = 0 It is not a Free Run Profile ------ PLL may or may not be in Hold Over

3) Write 0xB3 register to 0x0D
4) Read 0xb9[3] Read the 3rd bit in B9 register for Hold Over Status Information

If read data = 1PLL is in Holdover State

read data = 0PLL is not in Holdover State

Note:

0XD1 - Status Readback Address Register for Master and Slave Pages.

0XD0 - Status Readback Data Register for Master and Slave Pages. Need to be read atleast 3 times for reliable readback.

23 Programming the Primary E-Fuse

This section describes primary E-Fuse program configuration for all the pages [GENERIC_SYS, INPUT_SYS, CLKMON_SYS, OUTPUT_SYS, PLLB_SYS, PLLD_SYS].

VDD = 2.5 V and VDDIN = 2.5 V should be used for programming the E-Fuse

23.1 Configuration Bits to Force Power-up of Digital Slave Subsystems

The respective subsystems needs to be powered up before programming the E-Fuse

This section describes GENERIC_SYS page configuration required in register 0xe0 and 0xe1 to enable respective slave subsystems [INPUT_SYS, CLKMON_SYS, OUTPUT_SYS, PLLB_SYS, PLLD_SYS] as described in Table 22.

Table 22 Configuration Bits to Force Power-up of Digital Slave Subsystems

S.No	Page Number	Register Address	Bit Number	Value and Its Description
1.	Page 0	0xE0	0	1'h1 (write to bit number 0 in register address 0xE0 with value 1'h1 to enable force overwrite INPUT_SYS)
			1	1'h1 (write to bit number 1 in register address 0xE0 with value 1'h1 to enable force overwrite CLKMON_SYS)
			2	1'h1 (write to bit number 2 in register address 0xE0 with value 1'h1 to enable force overwrite OUTPUT_SYS)
			4	1'h1 (write to bit number 4 in register address 0xE0 with value 1'h1 to enable force overwrite PLLB_SYS)
			6	1'h1 (write to bit number 6 in register address 0xE0 with value 1'h1 to enable force overwrite PLLD_SYS)
2.	Page 0	0xE1	0	1'h1 (write to bit number 0 in register address 0xE1 with value 1'h1 INPUT_SYS to be Enabled)
			1	1'h1 (write to bit number 1 in register address 0xE1 with value 1'h1 CLKMON_SYS to be Enabled)
			2	1'h1 (write to bit number 2 in register address 0xE1 with value 1'h1 OUTPUT_SYS to be Enabled)
			4	1'h1 (write to bit number 4 in register address 0xE1 with value 1'h1 PLLB_SYS to be Enabled)
			6	1'h1 (write to bit number 5 in register address 0xE1 with value1'h1PLLD_SYS to be Enabled)

23.2 E-Fuse Lock Configuration Bits

This section describes the location of the two bits to lock the respective slave subsystem E-Fuse by writing into the register 0x2F in all pages as described in Table 23.

Table 23 E-Fuse Lock Configuration Bits for all pages

S.No	Page Number	Register Address	Bit Number	Value and Its Description
1.	Page 0	0x2f	7:6	2'h1 : E-Fuse of GENERIC_SYS is locked by writing into bit number [7:6] in register address 0x2f with value 2'h1
2.	Page 1	0x2f	7:6	2'h1 : E-Fuse of CLKMON_SYS is locked by writing into bit number [7:6] in register address 0x2f with value 2'h1
3.	Page 2	0x2f	7:6	2'h1 : E-Fuse of INPUT_SYS is locked by writing into bit number [7:6] in register address 0x2f with value 2'h1
4.	Page 3	0x2f	7:6	2'h1 : E-Fuse of OUTPUT_SYS is locked by writing into bit number [7:6] in register address 0x2f with value 2'h1
5.	Page B	0x2f	7:6	2'h1 : E-Fuse of PLLB_SYS is locked by writing into bit number [7:6] in register address 0x2f with value 2'h1
6.	Page D	0x2f	7:6	2'h1 : E-Fuse of PLLD_SYS is locked by writing into bit number [7:6] in register address 0x2f with value 2'h1

23.3 E-Fuse Write Configuration Bits

This section will describe how to program the E-Fuse by writing into the register 0x0F in all the pages as mentioned in Table 24.

Table 24 E-Fuse Write Configuration Bits for All Pages

S.No	Page Number	Register Address	Bit Number	Value and Its Description
1.	Page 0	0x0f	7:3	5'h0C: writing to bit number [7:3] in register address 0x0f with value 5'h0C to do E-Fuse write in GENERIC_SYS
2.	Page 1	0x0f	7:3	5'h0C: writing to bit number [7:3] in register address 0x0f with value 5'h0C to do E-Fuse write in CLKMON_SYS
3.	Page 2	0x0f	7:3	5'h0C: writing to bit number [7:3] in register address 0x0f with value 5'h0C to do E-Fuse write in INPUT_SYS
4.	Page 3	0x0f	7:3	5'h0C: writing to bit number [7:3] in register address 0x0f with value 5'h0C to do E-Fuse write in OUTPUT_SYS
5	Page B	0x0f	7:3	5'h0C: writing to bit number [7:3] in register address 0x0f with value 5'h0C to do E-Fuse write in PLLB_SYS
6.	Page D	0x0f	7:3	5'h0C: writing to bit number [7:3] in register address 0x0f with value 5'h0C to do E-Fuse write in PLLD_SYS

23.4 Configuration Bit to Remove Manual Wake Up for Primary E-Fuse

Manual wake-up of slave subsystems is enabled while programming the E-Fuse. As a final step the manual wake-up mode needs to be disabled in the GENERIC_SYS page so that the chip will wake-up autonomously

This section will describe how to remove manual wakeup by writing into a register 0x2f in GENERIC_SYS as mentioned in Table 25

Table 25 Configuration Bit to Remove Manual Wakeup for Primary E-Fuse

S.No	Page Number	Register Address	Bit Number	Value & It's Description
1.	Page 0	0x2f	4	1'h1 : (writing to bit number 4 in register address 0x2f with value 'h1 to remove manual wake up)

23.5 Pseudo Code: Programming the Primary E-Fuse

VDD and VDDIN supply should be set to 2.5 V while programming the E-Fuse

23.5.1 GENERIC_SYS

STEP 1: Write the GENERIC_SYS page number configuration

```
i2c.i2cw(device_address,0xff,0x00)
```

STEP 2: Write the GENERIC_SYS NVM Registers configuration

STEP 3: Refer E-Fuse Write Configuration Bits section described earlier

Write to bit number [7:3] in register address 0x0f with value 5'hC to program E-Fuse registers of GENERIC_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0xc0)
i2c.i2cw(device_address,0x0f,0x00)
```

STEP 4: Refer Configuration Bits to Force Power-up of Digital Slave Subsystems described earlier

Force Enable all slaves [INPUT_SYS, CLKMON_SYS, OUTPUT_SYS, PLLB_SYS, PLLD_SYS] by programming 0xE0 & 0xE1 both with value 7'h7F

```
i2c.i2cw(device_address,0xe0,0x7f)
```

```
i2c.i2cw(device_address,0xe1,0x7f)
```

23.5.2 INPUT_SYS

STEP 5: Write the INPUT_SYS page number configuration

```
i2c.i2cw(device_address,0xff,0x02)
```

STEP 6: Refer E-Fuse Lock Configuration Bits

Write the INPUT_SYS NVM Registers configuration and write 0x2F register for bit number [7:6] with 2'h1 to lock E-Fuse of INPUT_SYS

STEP 7: Refer E-Fuse Write Configuration Bits

Write to bit number [7:3] in register address 0x0f with value 5'hC to program E-Fuse Registers of INPUT_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0xc0)
i2c.i2cw(device_address,0x0f,0x00)
```

23.5.3 CLKMON_SYS

STEP 8: Write the CLKMON_SYS page number configuration

```
i2c.i2cw(device_address,0xff,0x01)
```

STEP 9: Refer E-Fuse Lock Configuration Bits

Write the CLKMON_SYS NVM Registers configuration and write 0x2F register for bit number [7:6] with 2'h1 to lock E-Fuse of CLKMON_SYS

STEP 10: Refer E-Fuse Write Configuration Bits

Write to bit number [7:3] in register address 0x0F with value 5'hC to program E-Fuse Registers of CLKMON_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0xc0)
i2c.i2cw(device_address,0x0f,0x00)
```

23.5.4 OUTPUT_SYS

STEP 11: Write the OUTPUT_SYS page number configuration

```
i2c.i2cw(device_address,0xff,0x03)
```

STEP 12: Refer E-Fuse Lock Configuration Bits

Write the OUTPUT_SYS NVM Registers configuration and write 0x2F register for bit number [7:6] with 2'h1 to lock E-Fuse of OUTPUT_SYS

STEP 13: Refer E-Fuse Write Configuration Bits

Write to bit number [7:3] in register address 0x0F with value 5'hC to program E-Fuse Registers of OUTPUT_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0xc0)
i2c.i2cw(device_address,0x0f,0x00)
```

23.5.5 PLLB_SYS

STEP 17: Write the PLLB_SYS page number configuration

```
i2c.i2cw(device_address,0xff,0x0b)
```

STEP 18: Refer E-Fuse Lock Configuration Bits

Write the PLLB_SYS NVM Registers configuration and write 0x2F register for bit number [7:6] with 2'h1 to lock E-Fuse of PLLB_SYS

STEP 19: Refer E-Fuse Write Configuration Bits

Write to bit number [7:3] in register address 0x0F with value 5'hC to program E-Fuse Registers of PLLB_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0xc0)
i2c.i2cw(device_address,0x0f,0x00)
```

23.5.6 PLLD_SYS

STEP 23: Write the PLLD_SYS page number configuration

```
i2c.i2cw(device_address,0xff,0x0d)
```

STEP 24: Refer E-Fuse Lock Configuration Bits

Write the PLLD_SYS NVM Registers configuration and write 0x2F register for bit number [7:6] with 2'h1 to lock E-Fuse of PLLD_SYS

STEP 25: Refer E-Fuse Write Configuration Bits

Write to bit number [7:3] in register address 0x0F with value 5'hC to program E-Fuse Registers of PLLD_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0xc0)
i2c.i2cw(device_address,0x0f,0x00)
```

23.5.7 Removing Manual Wake-up and Locking GENERIC_SYS E-Fuse

STEP 26: Write the GENERIC_SYS page number configuration

```
i2c.i2cw(device_address,0xff,0x00)
```

STEP 27: Refer Configuration Bit to Remove Manual Wake Up for Primary E-Fuse

Write to bit number 4 in register address 0x2F with value 1'h1 in GENERIC_SYS to Configuration Bit to Remove Manual Wake Up

```
i2c.i2crmw(device_address,0x2f,0x4,0x1,0x01)
```

i2crmw function register 0x2f bit number 4 writing with the value 1'h1

Note: i2crmw (dev_address, register_address, bit_postion,total_bits,value)

STEP 28: Refer E-Fuse Lock Configuration Bits

Write to bit number [7:6] in register address 0x2F with value 2'h1 in in GENERIC_SYS to lock the E-Fuse

```
i2c.i2crmw(device_address,0x2f,0x7,0x2,0x01)
```

i2crmw function register 0x2f bit number 7:6 writing with the value 2'h1

STEP 29: Refer E-Fuse Write Configuration Bits

Write to bit number [7:3] in register address 0x0F with value 5'hC to program E-Fuse Registers of GENERIC_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0xc0)
i2c.i2cw(device_address,0x0f,0x00)
```

STEP 30: Reset the chip or recycle Power for the chip to wake-up autonomously based on NVM configuration programmed in the primary E-Fuse

24 Programming the Secondary E-Fuse

This section describes Secondary E-Fuse program configuration for all the pages [GENERIC_SYS, INPUT_SYS, CLKMON_SYS, OUTPUT_SYS, PLLB_SYS, PLLD_SYS].

24.1 Configuration Bit To Escape To PROGRAM_CMD State in GENERIC_SYS

With the primary E-Fuse programmed and locked the chip will autonomously wake-up and reach ACTIVE_STATE

For the E-Fuse writes the chip needs to be in PROGRAM_CMD state

This section will describe how to do escape from ACTIVE_STATE to PROGRAM_CMD state in GENERIC_SYS by writing into the register 0x0F in GENERIC_SYS as mentioned in Table 26

Table 26 Configuration Bit To Escape to PROGRAM_CMD State in GENERIC_SYS

S.No	Page Number	Register Address	Bit Number	Value & It's Description
1.	Page 0	0x0f	1	1'h1: writing to bit number 1 in register address 0x0f with value 1'h1 to do Escape to PROGRAM_CMD state in GENERIC_SYS

24.2 Configuration Bit to Change the E-Fuse pointer

This section will describe how to point slaves to secondary E-Fuse by writing into a register 0x22 in GENERIC_SYS as mentioned in Table 27

Table 27 Configuration Bit to change the E-Fuse pointer

S.No	Page Number	Register Address	Bit Number	Value & It's Description
1.	Page 0	0x22	7:0	8'hFF: (writing to bit number [7:0] in register address 0x22 with value 8'hFF to point slaves [INPUT_SYS, CLKMON_SYS, OUTPUT_SYS, PLLB_SYS, PLLD_SYS] to secondary E-Fuse)

24.3 Configuration Bit to Enable Manual Wake Up for Secondary E-Fuse

This section will describe how to enable manual wakeup for secondary E-Fuse by writing into a register 0x10 in GENERIC_SYS as mentioned in Table 28

Table 28 Configuration Bit to Enable Manual Wakeup for Secondary E-Fuse

S.No	Page Number	Register Address	Bit Number	Value & It's Description
1.	Page 0	0x10	7	1'h1: (writing to bit number 7 in register address 0x10 with value 1'h1 to select manual wake up)
2.	Page 0	0x10	6	1'h0: (writing to bit number 6 in register address 0x10 with value 1'h0 to select manual wake up)

24.4 Psuedo Code: Programming the Secondary E-Fuse

24.4.1 GENERIC_SYS

STEP 1: Write the GENERIC_SYS page number configuration

```
i2c.i2cw(device_address,0xff,0x00)
```

STEP 2: Refer Configuration Bit To Escape To PROGRAM_CMD State in GENERIC_SYS

Write to bit number 1 in register address 0x0F with value 1'h1 to escape to PROGRAM_CMD state in GENERIC_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0x02)
i2c.i2cw(device_address,0x0f,0x00)
```

STEP 3: Refer Configuration Bit to Change the E-Fuse pointer

Write to bit number [7:0] in register address 0x22 with value 8'hFF to point slaves [INPUT_SYS, CLKMON_SYS, OUTPUT_SYS, PLLB_SYS, PLLD_SYS] to secondary E-Fuse in GENERIC_SYS

```
i2c.i2cw(device_address,0x22,0xff)
```

STEP 4: Refer Configuration Bit to Enable Manual Wake Up for Secondary E-Fuse

Write to bit number 7 in register address 0x10 with value 1'h1 to select for manual wakeup in GENERIC_SYS

```
i2C.I2Crmw(device_address,0x10,0x7,0x1,0x01)
```

i2crmw function register 0x10 bit number 7 writing with the value 1'h1

STEP 5: Refer E-Fuse Write Configuration Bits

Write to bit number [7:3] in register address 0x0F with value 5'hC to program E-Fuse Registers of GENERIC_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0xc0)
i2c.i2cw(device_address,0x0f,0x00)
```

STEP 6: Reset the chip or recycle power for the E-Fuse pointers to get updated

STEP 7: Refer Configuration Bits to Force Power-up of Digital Slave Subsystems

Force Enable all slaves [INPUT_SYS, CLKMON_SYS, OUTPUT_SYS, PLLB_SYS, PLLD_SYS] by programming 0xE0 & 0xE1 both with value 7'h7F

```
i2c.i2cw(device_address,0xe0,0x7f)
i2c.i2cw(device_address,0xe1,0x7f)
```

24.4.2 INPUT_SYS

STEP 8: Write the INPUT_SYS page number configuration

```
i2c.i2cw(device_address,0xff,0x02)
```

STEP 9: Refer E-Fuse Lock Configuration Bits

Write the INPUT_SYS NVM Registers configuration and write 0x2F register for bit number [7:6] with 2'h1 to lock E-Fuse of INPUT_SYS

STEP 10: Refer E-Fuse Write Configuration Bits

Write to bit number [7:3] in register address 0x0F with value 5'hC to program E-Fuse Registers of INPUT_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0xc0)
i2c.i2cw(device_address,0x0f,0x00)
```

24.4.3 CLKMON_SYS

STEP 11: Write the CLKMON_SYS page number configuration

```
i2c.i2cw(device_address,0xff,0x01)
```

STEP 12: Refer E-Fuse Lock Configuration Bits

Write the CLKMON_SYS NVM Registers configuration and write 0x2F register for bit number [7:6] with 2'h1 to lock E-Fuse of CLKMON_SYS

STEP 13: Refer E-Fuse Write Configuration Bits

Write to bit number [7:3] in register address 0x0F with value 5'hC to program E-Fuse Registers of CLKMON_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0xc0)
i2c.i2cw(device_address,0x0f,0x00)
```

24.4.4 OUTPUT_SYS

STEP 14: Write the OUTPUT_SYS page number configuration

```
i2c.i2cw(device_address,0xff,0x03)
```

STEP 15: Refer E-Fuse Lock Configuration Bits

Write the OUTPUT_SYS NVM Registers configuration and write 0x2F register for bit number [7:6] with 2'h1 to lock E-Fuse of OUTPUT_SYS

STEP 16: Refer E-Fuse Write Configuration Bits

Write to bit number [7:3] in register address 0x0F with value 5'hC to program E-Fuse Registers of OUTPUT_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0xc0)
i2c.i2cw(device_address,0x0f,0x00)
```

24.4.5 PLLB_SYS

STEP 20: Write the PLLB_SYS page number configuration

```
i2c.i2cw(device_address,0xff,0x0b)
```

STEP 21: Refer E-Fuse Lock Configuration Bits

Write the PLLB_SYS NVM Registers configuration and write 0x2F register for bit number [7:6] with 2'h1 to lock E-Fuse of PLLB_SYS

STEP 22: Refer E-Fuse Write Configuration Bits

Write to bit number [7:3] in register address 0x0F with value 5'hC to program E-Fuse Registers of PLLB_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0xc0)
i2c.i2cw(device_address,0x0f,0x00)
```

24.4.6 PLLD_SYS

STEP 26: Write the PLLD_SYS page number configuration

```
i2c.i2cw(device_address,0xff,0x0d)
```

STEP 27: Refer E-Fuse Lock Configuration Bits

Write the PLLD_SYS NVM Registers configuration and write 0x2F register for bit number [7:6] with 2'h1 to lock E-Fuse of PLLD_SYS

STEP 28: Refer E-Fuse Write Configuration Bits

Write to bit number [7:3] in register address 0x0F with value 5'hC to program E-Fuse Registers of PLLD_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0xc0)
i2c.i2cw(device_address,0x0f,0x00)
```

24.4.7 Programming GENERIC_SYS E-Fuse to Remove Manual Wake Up in Secondary E-Fuse

STEP 29: Write the GENERIC_SYS page number configuration

```
i2c.i2cw(device_address,0xff,0x00)
```

STEP 30: Refer Configuration Bit To Escape To PROGRAM_CMD State in GENERIC_SYS

Write to bit number 1 in register address 0x0F with value 1'h1 to escape to PROGRAM_CMD state in GENERIC_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0x02)
i2c.i2cw(device_address,0x0f,0x00)
```

STEP 31: Write to bit number 6 in register address 0x10 with value 1'h1 in GENERIC_SYS to Configuration Bit to Remove Manual Wake Up in secondary E-Fuse

```
i2c.i2crmw(device_address,0x10,6,0x1,0x01)
```

i2crmw function register 0x10 bit number 6 writing with the value 1'h1

i2crmw(dev_address, register_address,bit_postion,total_bits,value)

STEP 32: Refer E-Fuse Write Configuration Bits

Write to bit number [7:3] in register address 0x0F with value 5'hC to program E-Fuse Registers of GENERIC_SYS

```
i2c.i2cw(device_address,0x0f,0x00)
i2c.i2cw(device_address,0x0f,0xc0)
i2c.i2cw(device_address,0x0f,0x00)
```

STEP 33: Reset Chip or recycle power for the chip to wake-up autonomously from the second E-Fuse

25 Register Map Details

Table 29 Page 0: Generic Masters System Related Registers

Registers from 10h to 4Fh are equivalent NVMCopy Registers for this Page.

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
VERSION_ID	01h	7:0	R	C1h	Overall Aura Platform Revision: AU5329A
				C2h	Overall Aura Platform Revision: AU5329B
STATUS_1_GENERIC	02h	7	R	00h	SPARE
		6	R	00h	SPARE
		5	R	00h	SPARE
		4	R	00h	SPARE
		3	R	00h	SPARE
		2	R	00h	Dynamic status for xoclk_loss
		1	R	00h	SPARE
		0	R	00h	Dynamic status for rccal_done
NOTIFY_1_GENERIC	03h	7	R/W	1h	SPARE
		6	R/W	1h	SPARE
		5	R/W	1h	SPARE
		4	R/W	1h	SPARE
		3	R/W	1h	SPARE
		2	R/W	1h	Sticky/Notify status for _xoclk_loss,
		1	R/W	1h	SPARE
		0	R/W	1h	SPARE
MASKb_1_GENERIC	04h	7	R/W	1h	SPARE
		6	R/W	1h	SPARE
		5	R/W	1h	SPARE
		4	R/W	1h	SPARE
		3	R/W	1h	SPARE
		2	R/W	1h	Mask bit for NOTIFY_1_GENERIC (03h) If programmed as '0': Mask sticky/Notify bit generation for 03h[2].
		1	R/W	1h	SPARE
		0	R/W	1h	SPARE
Directives_GENERIC	05h	3:2	R/W	00h	1 : DCO_PLLB 3 : DCO_PLLD
		1	R/W	00h	dco increment commonly used for all plls if 0xe7[7] is set as '1'
		0	R/W	00h	dco decrement commonly used for all plls if 0xe7[7] is set as '1'
STATUS_2_GENERIC	06h	7	R	00h	Dynamic status for plld_ho_freeze
		5	R	00h	Dynamic status for pllb_ho_freeze
STATUS_2_GENERIC	06h	3	R	00h	Dynamic status for plld_loss_of_lock
		1	R	00h	Dynamic status for pllb_loss_of_lock
NOTIFY_2_GENERIC	07h	7	R/W	1h	Sticky/Notify status for plld_ho_freeze
		5	R/W	1h	Sticky/Notify status for pllb_ho_freeze
		3	R/W	1h	Sticky/Notify status for plld_loss_of_lock
		1	R/W	1h	Sticky/Notify status for pllb_loss_of_lock
MASKb_2_GENERIC	08h	7	R/W	1h	Mask bit for NOTIFY_2_GENERIC (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[7].
		6	R/W	1h	Mask bit for NOTIFY_2_GENERIC (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[6].

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
MASKb_2_GENERIC	08h	5	R/W	1h	Mask bit for NOTIFY_2_GENERIC (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[5].
		4	R/W	1h	Mask bit for NOTIFY_2_GENERIC (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[4].
		3	R/W	1h	Mask bit for NOTIFY_2_GENERIC (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[3].
		2	R/W	1h	Mask bit for NOTIFY_2_GENERIC (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[2].
		1	R/W	1h	Mask bit for NOTIFY_2_GENERIC (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[1].
		0	R/W	1h	Mask bit for NOTIFY_2_GENERIC (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[0].
PRG_Directives_GENERIC	0Fh	7:3	R/W	0h	PRG_CMD Directives: 5b1_1000: PROGRAM_EFUSE 5b0_1100:READ_EFUSE 5b0_0110: Copy NVM Copy to Settings 5b1_1011: Proceed to Active
		2	R/W	0h	SPARE
		1	R/W	0h	Escape to the PRG_CMD state from ACTIVE state
NVMPLLEN_GENERIC	10h	7	R/W	0h	Selects between maual_wake_upb and manual_wake_up2b
		6	R/W	0h	select '0' to enable manual wake up sequence
		5:4	R/W	0h	VDD_DEF : VDD {1.8(00), 2.5(01), 3.3(10)}
		3:0	R/W	0h	Bits {3,2,1,0} correspond to enable of PLL {D,C,B,A}: Active Low.
NVMFLEXIO8_GENERIC	18h	7	R/W	0h	Select the direction of FLEXIO14. '0'- Input, '1' - Output
NVMFLEXIO8_GENERIC	18h	6:4	R/W	0h	Mux select to bring out internal signals on FLEXIO14, if programmed as output 3'b000 : 0 3'b001 : Ored_all_notify (clock loss for clkin1/2/ + Freq drift coarse for clkin 1/2 + Freq drift coarse for clkin 1/2 + Loss of lock for PLL B/D+HO freeze for PLL B/D) 3'b010 : Ored_all_pll_ntfy (Loss of lock for PLL B/D+HO freeze for PLL B/D) 3'b011 : Ored_all_clkmon_ntfy ((clock loss for clkin1/2 + Freq drift coarse for clkin 1/2 + Freq drift coarse for clkin 1/2) 3'b100 : Dynamic_Clock_Monitoring_Status 3'b101 : Internal debug 3'b110 : Internal debug 3'b111 : Internal debug
		3	R/W	0h	Select the direction of FLEXIO15. '0'- Input, '1' - Output

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
NVMFLEXIO8_GENERIC	18h	2:0	R/W	0h	Mux select to bring out internal signals on FLEXIO15, if programmed as output 3'b000 : 0 3'b001 : Ored_all_notify (clock loss for clkin1/2 + Freq drift coarse for clkin1/2 + Freq drift coarse for clkin1/2 + Loss of lock for PLL B/D+HO freeze for PLL B/D) 3'b010 : Ored_all_pll_ntfy (Loss of lock for PLL B/D+HO freeze for PLL B/D) 3'b011 : Ored_all_clkmon_ntfy ((clock loss for clkin1/2 + Freq drift coarse for clkin 1/2 + Freq drift coarse for clkin 1/2) 3'b100 : Dynamic_Clock_Monitoring_Status 3'b101 : Internal debug 3'b110 : Internal debug 3'b111 : Internal debug
NVMSPARE1_GENERIC	19h	7	R/W	0h	if set '1', Enables vdd padring functionality(0x23[7]) for padring selection. Keeping at 0 enables pad rail switch, select 1 to ensure no switch and keep rail to left
		6	R/W	0h	To map the external clock in select and allow the each pll independetly to force into holdover mode. Each pll ctrl - refer 0x2B[3:0]
FUSE_PTR_GENERIC	22h	7:0	R/W	0h	One Hot Decode for Fuse Pointer: Bits {7,6,5,4,2,0} correspond to fuse pointers for Pages 1, 2, 3, 4, B, D respectively Page 4 is reserved and not used.
BTOUT_IN_EN_GENERIC	24h	7:4	R/W	0h	One Hot Input Enable for the Clock Inputs 3:0 which are the 2 Clock inputs defined on Page 2: Active Low
		3:2	R/W	0h	One Hot Output Enable for Bottom Outputs 1:0 which are the 2 Bottom Fixed-Outputs defined on Page 3: Active Low
		1:0	R/W	0h	One Hot Output Enable for Top Outputs 1:0 which are the 2 Top Fixed-Outputs defined on Page 3: Active Low
FLEXOUTPUT_EN_GENERIC	25h	7:0	R/W	0h	One Hot Output Enable for Outputs 7:0 which are the 8 Flex-Outputs defined on Page 3: Active Low
OEB_CTRL	26h	7:6	R/W	0h	Selects which PLL to run in free mode at wake up , this feature is enables by 0x23[5]: 1: PLLB 3 : PLLD
		5:4	R/W	0h	Reserved
		3:2	R/W	0h	To program delay between enabling PLLB and other PLLs in fuse locked modes. 2'b00 - 4ms 2'b01 - 40 ms 2'b10 - 400 ms 2'b11 - 4s
		1	R/W	0h	if set as '1' then set PLLX_OEB as '1' if loss of lock status is asserted for respective PLL
		0	R/W	0h	if set as '1' then set PLLX_OEB as '1' if there is XO clock loss
INTR_MASK_1_CONFIG	27h	6	R/W	0h	If set '1', Interrupt will not be generated for DRIFT in freq for clk_in2
		5	R/W	0h	If set '1', Interrupt will not be generated for DRIFT in freq for clk_in1
		2	R/W	0h	If set '1', Interrupt will not be generated for CLOCK LOSS for clk_in2

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
INTR_MASK_1_CONFIG	27h	1	R/W	0h	If set '1', Interrupt will not be generated for CLOCK LOSS for clk_in1
INTR_MASK_2_CONFIG	28h	6	R/W	0h	If set '1', Interrupt will not be generated for LOSS OF LOCK for PLLB
INTR_MASK_2_CONFIG	28h	4	R/W	0h	If set '1', Interrupt will not be generated for LOSS OF LOCK for PLLD
		2	R/W	0h	If set '1', Interrupt will not be generated for HOLDOVER FREEZE for PLLB
		0	R/W	0h	If set '1', Interrupt will not be generated for HOLDOVER FREEZE for PLLD
CHIP_FLEXIO_CONFIG	29h	7	R/W	0h	Reserved
		6	R/W	0h	
		5	R/W	0h	
		4	R/W	0h	
		3	R/W	0h	
CHIP_FLEXIO_CONFIG	29h	2:0	R/W	0h	000: Reserved 001: AU5315/25/29 010: AU5317/27 011: AU5316/26 100: AU5314/24 101: AU5312/22
I2C_GENERIC	2Ah	7	R/W	0h	Enable the new I2C Address (if changed from the default 0x69)
		6:0	R/W	0h	I2C Address (when changed from the default 0x69)
XO2_GENERIC	2Dh	7:6	R/W	0h	OT3 GM programmability: Frequency selectivity configuration settings to support different frequency range of 3rd OT crystals
		5:0	R/W	0h	Crystal frequency trim settings
XO3_GENERIC	2Eh	7	R/W	0h	Crystal Pathway Settings: Use defaults from GUI based on crystal type and pathway used.
		6	R/W	0h	
		5:3	R/W	0h	
		2:0	R/W	0h	
OE_PATTERN_GENERIC	2Fh	7:6	R/W	0h	PATTERN :{'01'/'10'} =] Efuse Locked, {'00'/'11'} =] Efuse NOT Locked
		5	R/W	0h	High Speed Enable for the I2C pads, HS_EN=0 ensures true I2C function (no pull up) for I2C. Don't care for SPI.
		4	R/W	0h	select '0' to enable manual wake up sequence
		3:0	R/W	0h	Crystal Pathway Settings: Use defaults from GUI based on crystal type and pathway used.
RESET_REGISTER	FEh	7:0	R/W	0h	Register FE is a READ / WRITE register used to reset the chip. Writing 0x01 to this register will apply the reset to the digital and this is propagated asynchronously to the digital blocks. On writing a 0 to bit 0 of register FE, the reset is de-asserted and internal to the digital, we use the various clocks to perform the de-assertion reset synchronization for the appropriate clock domains.
PAGE_NUMBER	FFh	7:0	R/W	0h	On all pages register FF is a READ/WRITE register used to change the page number

Table 30 PAGE 1: Clock Monitor System Related Registers

Registers from 11h to 4Fh are equivalent NVMCopy Registers for this Page.

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
STATUS_CLKMON	02h	6	R	00h	Dynamic status for clk_in2_loss_OR_with_FD
		5	R	00h	Dynamic status for clk_in1_loss_OR_with_FD
		2	R	00h	Dynamic status for clk_in2_loss
		1	R	00h	Dynamic status for clk_in1_loss
NOTIFY_CLKMON	03h	6	R/W	1h	Sticky/Notify status for clk_in2_loss_OR_with_FD
		5	R/W	1h	Sticky/Notify status for clk_in1_loss_OR_with_FD
		2	R/W	1h	Sticky/Notify status for clk_in2_loss
		1	R/W	1h	Sticky/Notify status for clk_in1_loss
MASKb_CLKMON	04h	7	R/W	1h	Mask bit for NOTIFY_CLKMON (03h) If programmed as '0': Mask sticky/Notify bit generation for 03h[7]
		6	R/W	1h	Mask bit for NOTIFY_CLKMON (03h) If programmed as '0': Mask sticky/Notify bit generation for 03h[6]
		5	R/W	1h	Mask bit for NOTIFY_CLKMON (03h) If programmed as '0': Mask sticky/Notify bit generation for 03h[5]
		4	R/W	1h	Mask bit for NOTIFY_CLKMON (03h) If programmed as '0': Mask sticky/Notify bit generation for 03h[4]
MASKb_CLKMON	04h	3	R/W	1h	Mask bit for NOTIFY_CLKMON (03h) If programmed as '0': Mask sticky/Notify bit generation for 03h[3]
		2	R/W	1h	Mask bit for NOTIFY_CLKMON (03h) If programmed as '0': Mask sticky/Notify bit generation for 03h[2]
		1	R/W	1h	Mask bit for NOTIFY_CLKMON (03h) If programmed as '0': Mask sticky/Notify bit generation for 03h[1]
		0	R/W	1h	Mask bit for NOTIFY_CLKMON (03h) If programmed as '0': Mask sticky/Notify bit generation for 03h[0]
STATUS_FDCOARSE_CLKMON	06h	6	R	00h	Dynamic status for clk2_freq_coarse_drifted
		5	R	00h	Dynamic status for clk1_freq_coarse_drifted
STATUS_FDCOARSE_CLKMON	06h	2	R	00h	Dynamic status for clk2_freq_fine_drifted
		1	R	00h	Dynamic status for clk1_freq_fine_drifted
NOTIFY_FDCOARSE_CLKMON	07h	6	R/W	1h	Sticky/Notify status for clk2_freq_coarse_drifted
		5	R/W	1h	Sticky/Notify status for clk1_freq_coarse_drifted
		2	R/W	1h	Sticky/Notify status for clk2_freq_fine_drifted
		1	R/W	1h	Sticky/Notify status for clk1_freq_fine_drifted
MASKb_FDCOARSE_CLKMON	08h	7	R/W	1h	Mask bit for NOTIFY_FDCOARSE_CLKMON (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[7]

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
MASKb_FDCOARSE_CLKMON	08h	6	R/W	1h	Mask bit for NOTIFY_FDCOARSE_CLKMON (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[6]
		5	R/W	1h	Mask bit for NOTIFY_FDCOARSE_CLKMON (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[5]
		4	R/W	1h	Mask bit for NOTIFY_FDCOARSE_CLKMON (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[4]
		3	R/W	1h	Mask bit for NOTIFY_FDCOARSE_CLKMON (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[3]
		2	R/W	1h	Mask bit for NOTIFY_FDCOARSE_CLKMON (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[2]
MASKb_FDCOARSE_CLKMON	08h	1	R/W	1h	Mask bit for NOTIFY_FDCOARSE_CLKMON (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[1]
		0	R/W	1h	Mask bit for NOTIFY_FDCOARSE_CLKMON (07h) If programmed as '0': Mask sticky/Notify bit generation for 07h[0]
FD32_STATUS_COARSE_CLKMON	0Dh	7:0	R/W	0h	Reserved
FD10_STATUS_COARSE_CLKMON	0Eh	7:0	R/W	0h	Reserved
PRG_Directives_CLKMON	0Fh	7:3	R/W	0h	PRG_CMD Directives: 5b1_1000: PROGRAM_EFUSE 5b0_1100: READ_EFUSE 5b0_0110: Copy NVM Copy to Settings 5b1_1011: Proceed to Active
		2	R/W	0h	Spare
		1	R/W	0h	Escape to the PRG_CMD state from ACTIVE state
CL_REG2_CLKMON	11h	5:4	R/W	0h	IN2_VAL_TIME : timer setting for deassertion of clock loss for FD2 Values {0,1,2,3} correspond to {2ms, 100ms, 200ms, 1sec}
		3:2	R/W	0h	IN1_VAL_TIME : timer setting for deassertion of clock loss for FD1 Values {0,1,2,3} correspond to {2ms, 100ms, 200ms, 1sec}
CL2_SET_THR2_CLKMON	15h	7:3	R/W	0h	SPARE
		2:0	R/W	0h	IN2 19-bit Threshold Value GUI calculates the correct CL register threshold trigger value for IN2
CL2_SET_THR1_CLKMON	16h	7:0	R/W	0h	
CL2_SET_THR0_CLKMON	17h	7:0	R/W	0h	
CL1_SET_THR2_CLKMON	18h	7:3	R/W	0h	SPARE
		2:0	R/W	0h	IN1 19-bit Threshold Value GUI calculates the correct CL register threshold trigger value for IN1
CL1_SET_THR1_CLKMON	19h	7:0	R/W	0h	
CL1_SET_THR0_CLKMON	1Ah	7:0	R/W	0h	

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
CL2_CLR_THR2_CLKMON	21h	7:3	R/W	0h	SPARE
		2:0	R/W	0h	IN2 19-bit Threshold Value GUI calculates the correct CL register threshold clear value for IN2
CL2_CLR_THR1_CLKMON	22h	7:0	R/W	0h	
CL2_CLR_THR0_CLKMON	23h	7:0	R/W	0h	
CL1_CLR_THR2_CLKMON	24h	7:3	R/W	0h	SPARE
		2:0	R/W	0h	IN1 19-bit Threshold Value GUI calculates the correct CL register threshold clear value for IN1
CL1_CLR_THR1_CLKMON	25h	7:0	R/W	0h	
CL1_CLR_THR0_CLKMON	26h	7:0	R/W	0h	
FD_REG1_CLKMON	2Ah	7:4	R/W	0h	FDCOARSE Enable Bit Value of position {3,2,1,0} correspond to Inputs {3,2,1,0}
		3:0	R/W	0h	FDFINE Enable Bit Value of position {3,2,1,0} correspond to Inputs {3,2,1,0}
FD_REG2_CLKMON	2Bh	7	R/W	0h	SPARE
		6:3	R/W	0h	XO divider configuration for FD monitors. GUI configures these dividers
		2:0	R/W	0h	Golden reference clock selection for frequency drift monitorsValue of 4 corresponds to the XO Reference Value of {3,2,1,0} correspond to Inputs {3,2,1,0}
FD_REG3_CLKMON	2Ch	7:4	R/W	0h	FD3 divider configuration for FD monitors. GUI configures these dividers
		3:0	R/W	0h	FD2 divider configuration for FD monitors. GUI configures these dividers
FD_REG4_CLKMON	2Dh	7:4	R/W	0h	FD1 divider configuration for FD monitors. GUI configures these dividers
		3:0	R/W	0h	FD0 divider configuration for FD monitors. GUI configures these dividers
FXOBYFIN2_LOG2_BAND	2Fh	7:6	R/W	0h	PATTERN :{'01'/'10'} => Efuse Locked, {'00'/'11'} => Efuse NOT Locked
		5:4	R/W	0h	SPARE
FXOBYFIN2_LOG2_BAND	2Fh	3:0	R/W	0h	IN2 FD monitor configuration bits calculated by GUI
COMMON_PLL_ACT_SPARE_SEL	31h	7:6	R/W	0h	PLLs ACTIVE Clock Selection 1 : CLKIN1, 2 : CLKIN2
		5:4	R/W	0h	PLLs SPARE0 Clock Selection 1 : CLKIN1, 2 : CLKIN2
		3:2	R/W	0h	PLLs SPARE1 Clock Selection 1 : CLKIN1, 2 : CLKIN2
		1:0	R/W	0h	PLLs SPARE2 Clock Selection 1 : CLKIN1, 2 : CLKIN2
FD2_CLR_THRFINE_CLKMON	33h	7:0	R/W	0h	IN2 FD clear threshold is calculated as 2X of FD2_CLR_THRFINE_CLKMON
FD1_CLR_THRFINE_CLKMON	34h	7:0	R/W	0h	IN1 FD clear threshold is calculated as 2X of FD1_CLR_THRFINE_CLKMON
FD32_CLR_THRCOARSE_CLKMON	36h	3:0	R/W	0h	IN2 FD clear threshold is calculated as 100*(FD2_CLR_THRCOARSE_CLKMON +1)
FD10_CLR_THRCOARSE_CLKMON	37h	7:4	R/W	0h	IN1 FD clear threshold is calculated as 100*(FD1_CLR_THRCOARSE_CLKMON +1)
FD32_SET_THRCOARSE_CLKMON	38h	3:0	R/W	0h	IN2 FD set threshold is calculated as 100*(FD2_SET_THRCOARSE_CLKMON +1)
FD10_SET_THRCOARSE_CLKMON	39h	7:4	R/W	0h	IN1 FD set threshold is calculated as 100*(SET_TH FD1_SET_THRCOARSE_CLKMON +1)

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
FD2_FGBYFM2_CLKMON	3Dh	7:0	R/W	0h	IN2 FD monitor configuration bits calculated by GUI
FD2_FGBYFM1_CLKMON	3Eh	7:0	R/W	0h	IN2 FD monitor configuration bits calculated by GUI
FD2_FGBYFM0_CLKMON	3Fh	7:0	R/W	0h	IN2 FD monitor configuration bits calculated by GUI
FD1_FGBYFM2_CLKMON	40h	7:0	R/W	0h	IN1 FD monitor configuration bits calculated by GUI
FD1_FGBYFM1_CLKMON	41h	7:0	R/W	0h	IN1 FD monitor configuration bits calculated by GUI
FD1_FGBYFM0_CLKMON	42h	7:0	R/W	0h	IN1 FD monitor configuration bits calculated by GUI
CLKX_HITLESS_SW_SOURCE	46h	5:4	R/W	0h	0 : CL2 1 : CL2 + FD2 COARSE 2: CL2 + FD2 FINE 3 : CL2 + FD2 COARSE + FD2 FINE
		3:2	R/W	0h	0 : CL1 1 : CL1 + FD1 COARSE 2 : CL1 + FD1 FINE 3 : CL1 + FD1 COARSE + FD1 FINE
FD2_SET_THRFINE_CLKMON	4Dh	7:0	R/W	0h	IN2 FD set threshold is calculated as 2X of FD2_SET_THRFINE_CLKMON
FD1_SET_THRFINE_CLKMON	4Eh	7:0	R/W	0h	IN1 FD set threshold is calculated as 2X of FD1_SET_THRFINE_CLKMON
PAGE_NUMBER	FFh	7:0	R/W	0h	On all pages register FF is a READ / WRITE register used to change the page number

Table 31 Input System Related Registers

Registers from 10h to 4Fh are equivalent NVMCopy Registers for this Page.

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
PRG_Directives_INPAGE	0Fh	7:3	R/W	0h	PRG_CMD Directives: 5b1_1000: PROGRAM_EFUSE 5b0_1100: READ_EFUSE 5b0_0110: Copy NVM Copy to Settings 5b1_1011: Proceed to Active
		2	R/W	0h	Spare
		1	R/W	0h	Escape to the PRG_CMD state from ACTIVE state
CLKIN1_DIVN1_INT1_INPAGE	20h	7:0	R/W	0h	IN1 DIVN1 Divider Integer Value
CLKIN1_DIVN1_INT2_INPAGE	21h	7	R/W	0h	IN1 DIVN1 Integer Mode of Division
		6	R/W	0h	SPARE
		5:0	R/W	0h	IN1 DIVN1 Divider Integer Value
CLKIN1_DIVN1_FRACN1_INPAGE	22h	7:0	R/W	0h	IN1 DIVN1 Divider Fractional Value Numerator
CLKIN1_DIVN1_FRACN2_INPAGE	23h	7:0	R/W	0h	IN1 DIVN1 Divider Fractional Value Numerator
CLKIN1_DIVN1_FRACN3_INPAGE	24h	7:0	R/W	0h	IN1 DIVN1 Divider Fractional Value Numerator
CLKIN1_DIVN1_FRACN4_INPAGE	25h	7:0	R/W	0h	IN1 DIVN1 Divider Fractional Value Numerator
CLKIN1_DIVN1_FRACD1_INPAGE	26h	7:0	R/W	0h	IN1 DIVN1 Divider Fractional Value Denominator
CLKIN1_DIVN1_FRACD2_INPAGE	27h	7:0	R/W	0h	IN1 DIVN1 Divider Fractional Value Denominator
CLKIN1_DIVN1_FRACD3_INPAGE	28h	7:0	R/W	0h	IN1 DIVN1 Divider Fractional Value Denominator
CLKIN1_DIVN1_FRACD4_INPAGE	29h	7:0	R/W	0h	IN1 DIVN1 Divider Fractional Value Denominator
CLKIN1_CFG1_INPAGE	2Ah	7:6	R/W	0h	IN1 DIVN1 Divider Integer Value
		5	R/W	0h	IN1 Direct Bypass for DIVN1
		4	R/W	0h	IN1 Single Ended Enable, Default is Differential
CLKIN1_CFG2_INPAGE	2Bh	7:0	R/W	0h	IN1 FD Ramp monitor configuration bits calculated by GUI
CLKIN1_CFG3_INPAGE	2Ch	7:0	R/W	0h	
CLKIN1_CFG4_INPAGE	2Dh	7:0	R/W	0h	
		3:0	R/W	0h	IN1 divider configuration for FD Ramp monitors. GUI configures these dividers
CLKIN1_RAMPFD_MEAS_COUNT	2Fh	7:6	R/W	0h	PATTERN :{'01'/'10'} =] Efuse Locked , {'00'/'11'} =] Efuse NOT Locked
		4:0	R/W	0h	Exponent that determines the ideal measurement count for Ramp: Calculated by the GUI.
CLKIN2_DIVN1_INT1_INPAGE	30h	7:0	R/W	0h	IN2 DIVN1 Divider Integer Value
CLKIN2_DIVN1_INT2_INPAGE	31h	7	R/W	0h	IN2 DIVN1 Integer Mode of Division
		5:0	R/W	0h	IN2 DIVN1 Divider Integer Value

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
CLKIN2_DIVN1_FRACN1_INPAGE	32h	7:0	R/W	0h	IN2 DIVN1 Divider Fractional Value Numerator
CLKIN2_DIVN1_FRACN2_INPAGE	33h	7:0	R/W	0h	IN2 DIVN1 Divider Fractional Value Numerator
CLKIN2_DIVN1_FRACN3_INPAGE	34h	7:0	R/W	0h	IN2 DIVN1 Divider Fractional Value Numerator
CLKIN2_DIVN1_FRACN4_INPAGE	35h	7:0	R/W	0h	IN2 DIVN1 Divider Fractional Value Numerator
CLKIN2_DIVN1_FRACD1_INPAGE	36h	7:0	R/W	0h	IN2 DIVN1 Divider Fractional Value Denominator
CLKIN2_DIVN1_FRACD2_INPAGE	37h	7:0	R/W	0h	IN2 DIVN1 Divider Fractional Value Denominator
CLKIN2_DIVN1_FRACD3_INPAGE	38h	7:0	R/W	0h	IN2 DIVN1 Divider Fractional Value Denominator
CLKIN2_DIVN1_FRACD4_INPAGE	39h	7:0	R/W	0h	IN2 DIVN1 Divider Fractional Value Denominator
CLKIN2_CFG1_INPAGE	3Ah	7:6	R/W	0h	IN2 DIVN1 Divider Integer Value
		5	R/W	0h	IN2 Direct Bypass for DIVN1
		4	R/W	0h	IN2 Single Ended Enable, Default is Differential
CLKIN2_CFG2_INPAGE	3Bh	7:0	R/W	0h	IN2 FD Ramp monitor configuration bits calculated by GUI
CLKIN2_CFG3_INPAGE	3Ch	7:0	R/W	0h	
CLKIN2_CFG4_INPAGE	3Dh	7:0	R/W	0h	
CLKIN2_OUTSEL_INPAGE	3Eh	7:4	R/W	0h	Allows multiple trigger addressing within a single window of FD Ramp drift monitor for selected PLL [3:0] - {PLLB, PLLD}
		3:0	R/W	0h	IN2 divider configuration for FD Ramp monitors. GUI configures these dividers
		4:0	R/W	0h	Exponent that determines the ideal measurement count for Ramp: Calculated by the GUI.
PAGE_NUMBER	FFh	7:0	R/W	0h	On all pages register FF is a READ / WRITE register used to change the page number

Table 32 Output System and Output Dividers Related Registers

Registers from 10h to 67h are equivalent NVMCopy Registers for this Page.

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
PRG_Directives_OUTPAGE	0Fh	7:3	R/W	0h	PRG_CMD Directives: 5b1_1000: PROGRAM_EFUSE 5b0_1100: READ_EFUSE 5b0_0110: Copy NVM Copy to Settings 5b1_1011: Proceed to Active
		2	R/W	0h	Spare
		1	R/W	0h	Escape to the PRG_CMD state from ACTIVE state
		2	R/W	0h	
		1	R/W	0h	
		0	R/W	0h	
DIVO1_DIV2_OUTPAGE	18h	7:4	R/W	0h	Spare
		3:0	R/W	0h	DIVO Divider for this output: Bits [19:16]
DIVO1_DIV1_OUTPAGE	19h	7:0	R/W	0h	DIVO Divider for this output: Bits [15:8]
DIVO1_DIV0_OUTPAGE	1Ah	7:0	R/W	0h	DIVO Divider for this output: Bits [7:0]
DIVO1_PROG1_OUTPAGE	1Bh	7:6	R/W	0h	CMOS Driver Phase Selection phase_sel<1> phase_sel<0> ODR_P ODR_N 0 0 CLKP CLKN 0 1 CLKP CLKP 1 0 CLKN CLKN 1 1 CLKN CLKP
		2:0	R/W	0h	Single Ended Driver Programming of strength: Use 0b111= 0d7
DIVO1_PROG0_OUTPAGE	1Ch	7:0	R/W	0h	Programmable Output Delay PRG_DELAY[5:0] is the coarse delay on the clock output. It determines relative delay on this clock programmable from 0 to 63 VCO clock delays based on this number PRG_DELAY[7:6] is the fine delay on the clock output. It determines relative delay on this clock programmable in 3, 2, 1, 0 times 30 ps based on this 2 bit code
DIVO1_MISC2_OUTPAGE	1Dh	4:3	R/W	0h	VDD_DEF : VDD {1.8(00), 2.5(01), 3.3(10)}. VDD Definition for this particular output.
		2:0	R/W	0h	DRV_TYPE : Output Driver Standard: 0b010: DC Coupled CML 0b011: DC Coupled HCSL 0b000: LVDS (Can be AC Coupled or DC Coupled) 0b100: Boosted LVDS (Use for LVPECL like swings with AC Coupled loads) 0b001: DC Coupled LVPECL (with Common mode current) 0b101: DC Coupled LVPECL2 (without Common mode current)
DIVO2_DIV2_OUTPAGE	20h	7:4	R/W	0h	Spare
		3:0	R/W	0h	DIVO Divider for this output: Bits [19:16]

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
DIVO2_DIV1_OUTPAGE	21h	7:0	R/W	0h	DIVO Divider for this output: Bits [15:8]
DIVO2_DIV0_OUTPAGE	22h	7:0	R/W	0h	DIVO Divider for this output: Bits [7:0]
DIVO2_PROG1_OUTPAGE	23h	7:6	R/W	0h	CMOS Driver Phase Selection phase_sel<1> phase_sel<0> ODR_P ODR_N 0 0 CLKP CLKN 0 1 CLKP CLKP 1 0 CLKN CLKN 1 1 CLKN CLKP
		2:0	R/W	0h	Single Ended Driver Programming of strength: Use 0b111= 0d7
DIVO2_PROG0_OUTPAGE	24h	7:0	R/W	0h	Programmable Output Delay PRG_DELAY[5:0] is the coarse delay on the clock output. It determines relative delay on this clock programmable from 0 to 63 VCO clock delays based on this number PRG_DELAY[7:6] is the fine delay on the clock output. It determines relative delay on this clock programmable in 3, 2, 1, 0 times 30 ps based on this 2 bit code
DIVO2_MISC2_OUTPAGE	25h	4:3	R/W	0h	VDD_DEF : VDD {1.8(00), 2.5(01), 3.3(10)}. VDD Definition for this particular output.
		2:0	R/W	0h	DRV_TYPE : Output Driver Standard: 0b010: DC Coupled CML 0b011: DC Coupled HCSL 0b000: LVDS (Can be AC Coupled or DC Coupled) 0b100: Boosted LVDS (Use for LVPECL like swings with AC Coupled loads) 0b001: DC Coupled LVPECL (with Common mode current) 0b101: DC Coupled LVPECL2 (without Common mode current)
DIVO3_DIV2_OUTPAGE	28h	7:4	R/W	0h	Spare
		3:0	R/W	0h	DIVO Divider for this output: Bits [19:16]
DIVO3_DIV1_OUTPAGE	29h	7:0	R/W	0h	DIVO Divider for this output: Bits [15:8]
DIVO3_DIV0_OUTPAGE	2Ah	7:0	R/W	0h	DIVO Divider for this output: Bits [7:0]
DIVO3_PROG1_OUTPAGE	2Bh	7:6	R/W	0h	CMOS Driver Phase Selection phase_sel<1> phase_sel<0> ODR_P ODR_N 0 0 CLKP CLKN 0 1 CLKP CLKP 1 0 CLKN CLKN 1 1 CLKN CLKP
		2:0	R/W	0h	Single Ended Driver Programming of strength: Use 0b111= 0d7
DIVO3_PROG0_OUTPAGE	2Ch	7:0	R/W	0h	Programmable Output Delay PRG_DELAY[5:0] is the coarse delay on the clock output. It determines relative delay on this clock programmable from 0 to 63 VCO clock delays based on this number PRG_DELAY[7:6] is the fine delay on the clock output. It determines relative delay on this clock programmable in 3, 2, 1, 0 times 30 ps based on this 2 bit code

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
DIVO3_MISC2_OUTPAGE	2Dh	4:3	R/W	0h	VDD_DEF : VDD {1.8(00), 2.5(01), 3.3(10)}. VDD Definition for this particular output.
DIVO3_MISC2_OUTPAGE	2Dh	2:0	R/W	0h	DRV_TYPE : Output Driver Standard: 0b010: DC Coupled CML 0b011: DC Coupled HCSL 0b000: LVDS (Can be AC Coupled or DC Coupled) 0b100: Boosted LVDS (Use for LVPECL like swings with AC Coupled loads) 0b001: DC Coupled LVPECL (with Common mode current) 0b101: DC Coupled LVPECL2 (without Common mode current)
DIVO3_MISC0_OUTPAGE	2Fh	7:6	R/W	0h	PATTERN :{'01'/'10'} =] Efuse Written, {'00'/'11'} =] Efuse NOT Written
		4	R/W	0h	PRG_DELAY[8:6] is the fine delay on the clock output. It determines relative delay on this clock programmable in 4,3, 2, 1, 0 times 30 ps based on this 3 bit code
		3	R/W	0h	{0, 0, 0, 0} External Termination, Differential Output {0, 0, 1, 0} Internal Pull Up, Differential Output {0, 0, 0, 1} Internal Pull Dn, Differential Output {0, 1, 0, 0} CMOS On OutP, Nothing on OutN {1, 0, 0, 0} Nothing on OutP, CMOS on OutN {1, 1, 0, 0} CMOS on OutP, CMOS on OutN
		2	R/W	0h	
		1	R/W	0h	
		0	R/W	0h	
		2	R/W	0h	
		1	R/W	0h	
		0	R/W	0h	
DIVO5_DIV2_OUTPAGE	38h	7:4	R/W	0h	Spare
		3:0	R/W	0h	DIVO Divider for this output: Bits [19:16]
DIVO5_DIV1_OUTPAGE	39h	7:0	R/W	0h	DIVO Divider for this output: Bits [15:8]
DIVO5_DIV0_OUTPAGE	3Ah	7:0	R/W	0h	DIVO Divider for this output: Bits [7:0]
DIVO5_PROG1_OUTPAGE	3Bh	7:6	R/W	0h	CMOS Driver Phase Selection phase_sel<1> phase_sel<0> ODR_P ODR_N 0 0 CLKP CLKN 0 1 CLKP CLKP 1 0 CLKN CLKN 1 1 CLKN CLKP
		2:0	R/W	0h	Single Ended Driver Programming of strength: Use 0b111= 0d7
DIVO5_PROG0_OUTPAGE	3Ch	7:0	R/W	0h	Programmable Output Delay PRG_DELAY[5:0] is the coarse delay on the clock output. It determines relative delay on this clock programmable from 0 to 63 VCO clock delays based on this number PRG_DELAY[7:6] is the fine delay on the clock output. It determines relative delay on this clock programmable in 3, 2, 1, 0 times 30 ps based on this 2 bit code
DIVO5_MISC3_OUTPAGE	3Dh	4:3	R/W	0h	VDD_DEF : VDD {1.8(00), 2.5(01), 3.3(10)}. VDD Definition for this particular output.

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
DIVO5_MISC3_OUTPAGE	3Dh	2:0	R/W	0h	DRV_TYPE : Output Driver Standard: 0b010: DC Coupled CML 0b011: DC Coupled HCSL 0b000: LVDS (Can be AC Coupled or DC Coupled) 0b100: Boosted LVDS (Use for LVPECL like swings with AC Coupled loads) 0b001: DC Coupled LVPECL (with Common mode current) 0b101: DC Coupled LVPECL2 (without Common mode current)
DIVO6_DIV2_OUTPAGE	40h	7:4	R/W	0h	spare
		3:0	R/W	0h	DIVO Divider for this output: Bits [19:16]
DIVO6_DIV1_OUTPAGE	41h	7:0	R/W	0h	DIVO Divider for this output: Bits [15:8]
DIVO6_DIV0_OUTPAGE	42h	7:0	R/W	0h	DIVO Divider for this output: Bits [7:0]
DIVO6_PROG1_OUTPAGE	43h	7:6	R/W	0h	CMOS Driver Phase Selection phase_sel<1> phase_sel<0> ODR_P ODR_N 0 0 CLKP CLKN 0 1 CLKP CLKP 1 0 CLKN CLKN 1 1 CLKN CLKP
		2:0	R/W	0h	Single Ended Driver Programming of strength: Use 0b111= 0d7
DIVO6_PROG0_OUTPAGE	44h	7:0	R/W	0h	Programmable Output Delay PRG_DELAY[5:0] is the coarse delay on the clock output. It determines relative delay on this clock programmable from 0 to 63 VCO clock delays based on this number PRG_DELAY[7:6] is the fine delay on the clock output. It determines relative delay on this clock programmable in 3, 2, 1, 0 times 30 ps based on this 2 bit code
DIVO6_MISC2_OUTPAGE	45h	4:3	R/W	0h	VDD_DEF : VDD {1.8(00), 2.5(01), 3.3(10)}. VDD Definition for this particular output.
		2:0	R/W	0h	DRV_TYPE : Output Driver Standard: 0b010: DC Coupled CML 0b011: DC Coupled HCSL 0b000: LVDS (Can be AC Coupled or DC Coupled) 0b100: Boosted LVDS (Use for LVPECL like swings with AC Coupled loads) 0b001: DC Coupled LVPECL (with Common mode current) 0b101: DC Coupled LVPECL2 (without Common mode current)
DIVO6_MISC0_OUTPAGE	47h	7:6	R/W	0h	DIVO Divider for 0B output: Bits [33:32]
		4	R/W	0h	PRG_DELAY[8:6] is the fine delay on the clock output. It determines relative delay on this clock programmable in 4,3, 2, 1, 0 times 30 ps based on this 3 bit code
		3	R/W	0h	{0, 0, 0, 0} External Termination, Differential Output {0, 0, 1, 0} Internal Pull Up, Differential Output
		2	R/W	0h	
		1	R/W	0h	

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
DIVO6_MISC0_OUTPAGE	47h	0	R/W	0h	{0, 0, 0, 1} Internal Pull Dn, Differential Output {0, 1, 0, 0} CMOS On OutP, Nothing on OutN {1, 0, 0, 0} Nothing on OutP, CMOS on OutN {1, 1, 0, 0} CMOS on OutP, CMOS on OutN
DIVO7_DIV2_OUTPAGE	48h	7:4	R/W	0h	spare
		3:0	R/W	0h	DIVO Divider for this output: Bits [19:16]
DIVO7_DIV1_OUTPAGE	49h	7:0	R/W	0h	DIVO Divider for this output: Bits [15:8]
DIVO7_DIV0_OUTPAGE	4Ah	7:0	R/W	0h	DIVO Divider for this output: Bits [7:0]
DIVO7_PROG1_OUTPAGE	4Bh	7:6	R/W	0h	CMOS Driver Phase Selection phase_sel<1> phase_sel<0> ODR_P ODR_N 0 0 CLKP CLKN 0 1 CLKP CLKP 1 0 CLKN CLKN 1 1 CLKN CLKP
		2:0	R/W	0h	Single Ended Driver Programming of strength: Use 0b111= 0d7
DIVO7_PROG0_OUTPAGE	4Ch	7:0	R/W	0h	Programmable Output Delay PRG_DELAY[5:0] is the coarse delay on the clock output. It determines relative delay on this clock programmable from 0 to 63 VCO clock delays based on this number PRG_DELAY[7:6] is the fine delay on the clock output. It determines relative delay on this clock programmable in 3, 2, 1, 0 times 30 ps based on this 2 bit code
DIVO7_MISC2_OUTPAGE	4Dh	4:3	R/W	0h	VDD_DEF : VDD {1.8(00), 2.5(01), 3.3(10)}. VDD Definition for this particular output.
		2:0	R/W	0h	DRV_TYPE : Output Driver Standard: 0b010: DC Coupled CML 0b011: DC Coupled HCSL 0b000: LVDS (Can be AC Coupled or DC Coupled) 0b100: Boosted LVDS (Use for LVPECL like swings with AC Coupled loads) 0b001: DC Coupled LVPECL (with Common mode current) 0b101: DC Coupled LVPECL2 (without Common mode current)
DIVO7_MISC0_OUTPAGE	4Fh	7:6	R/W	0h	DIVO Divider for 0B output: Bits [31:30]
		4	R/W	0h	PRG_DELAY[8:6] is the fine delay on the clock output. It determines relative delay on this clock programmable in 4,3, 2, 1, 0 times 30 ps based on this 3 bit code
		3	R/W	0h	{0, 0, 0, 0} External Termination, Differential Output {0, 0, 1, 0} Internal Pull Up, Differential Output
		2	R/W	0h	
		1	R/W	0h	

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
DIVO7_MISC0_OUTPAGE	4Fh	0	R/W	0h	{0, 0, 0, 1} Internal Pull Dn, Differential Output {0, 1, 0, 0} CMOS On OutP, Nothing on OutN {1, 0, 0, 0} Nothing on OutP, CMOS on OutN {1, 1, 0, 0} CMOS on OutP, CMOS on OutN
DIVO0B_DIV2_OUTPAGE	60h	7:0	R/W	0h	DIVO Divider for 0B output: Bits [23:16]
DIVO0B_DIV1_OUTPAGE	61h	7:0	R/W	0h	DIVO Divider for 0B output: Bits [15:8]
DIVO0B_DIV0_OUTPAGE	62h	7:0	R/W	0h	DIVO Divider for 0B output: Bits [7:0]
DIVO0B_PROG1_OUTPAGE	63h	7:6	R/W	0h	CMOS Driver Phase Selection phase_sel<1> phase_sel<0> ODR_P ODR_N 0 0 CLKP CLKN 0 1 CLKP CLKP 1 0 CLKN CLKN 1 1 CLKN CLKP
		2:0	R/W	0h	Single Ended Driver Programming of strength: Use 0b111= 0d7
DIVO0B_PROG0_OUTPAGE	64h	7:0	R/W	0h	Programmable Output Delay PRG_DELAY[5:0] is the coarse delay on the clock output. It determines relative delay on this clock programmable from 0 to 63 VCO clock delays based on this number PRG_DELAY[7:6] is the fine delay on the clock output. It determines relative delay on this clock programmable in 3, 2, 1, 0 times 30 ps based on this 2 bit code
DIVO0B_MISC2_OUTPAGE	65h	4:3	R/W	0h	VDD_DEF : VDD {1.8(00), 2.5(01), 3.3(10)}. VDD Definition for this particular output.
		2:0	R/W	0h	DRV_TYPE : Output Driver Standard: 0b010: DC Coupled CML 0b011: DC Coupled HCSL 0b000: LVDS (Can be AC Coupled or DC Coupled) 0b100: Boosted LVDS (Use for LVPECL like swings with AC Coupled loads) 0b001: DC Coupled LVPECL (with Common mode current) 0b101: DC Coupled LVPECL2 (without Common mode current)
DIVO0B_MISC0_OUTPAGE	67h	7:6	R/W	0h	DIVO Divider for 0B output: Bits [27:26]
		4	R/W	0h	PRG_DELAY[8:6] is the fine delay on the clock output. It determines relative delay on this clock programmable in 4,3, 2, 1, 0 times 30 ps based on this 3 bit code
		3	R/W	0h	{0, 0, 0, 0} External Termination, Differential Output {0, 0, 1, 0} Internal Pull Up, Differential Output {0, 0, 0, 1} Internal Pull Dn, Differential Output {0, 1, 0, 0} CMOS On OutP, Nothing on OutN {1, 0, 0, 0} Nothing on OutP, CMOS on OutN {1, 1, 0, 0} CMOS on OutP, CMOS on OutN
		2	R/W	0h	
		1	R/W	0h	
DIVO0B_MISC0_OUTPAGE	67h	0	R/W	0h	

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
DIVO1B_DIV2_OUTPAGE	68h	7:4	R/W	0h	spare
		3:0	R/W	0h	DIVO Divider for this output: Bits [19:16]
DIVO1B_DIV1_OUTPAGE	69h	7:0	R/W	0h	DIVO Divider for this output: Bits [15:8]
DIVO1B_DIV0_OUTPAGE	6Ah	7:0	R/W	0h	DIVO Divider for this output: Bits [7:0]
DIVO1B_PROG1_OUTPAGE	6Bh	7:6	R/W	0h	CMOS Driver Phase Selection phase_sel<1> phase_sel<0> ODR_P ODR_N 0 0 CLKP CLKN 0 1 CLKP CLKP 1 0 CLKN CLKN 1 1 CLKN CLKP
		2:0	R/W	0h	Single Ended Driver Programming of strength: Use 0b111= 0d7
DIVO1B_PROG0_OUTPAGE	6Ch	7:0	R/W	0h	Programmable Output Delay PRG_DELAY[5:0] is the coarse delay on the clock output. It determines relative delay on this clock programmable from 0 to 63 VCO clock delays based on this number PRG_DELAY[7:6] is the fine delay on the clock output. It determines relative delay on this clock programmable in 3, 2, 1, 0 times 30 ps based on this 2 bit code
DIVO1B_MISC2_OUTPAGE	6Dh	4:3	R/W	0h	VDD_DEF : VDD {1.8(00), 2.5(01), 3.3(10)}. VDD Definition for this particular output.
		2:0	R/W	0h	DRV_TYPE : Output Driver Standard: 0b010: DC Coupled CML 0b011: DC Coupled HCSL 0b000: LVDS (Can be AC Coupled or DC Coupled) 0b100: Boosted LVDS (Use for LVPECL like swings with AC Coupled loads) 0b001: DC Coupled LVPECL (with Common mode current) 0b101: DC Coupled LVPECL2 (without Common mode current)
DIVO1B_MISC0_OUTPAGE	6Fh	7:6	R/W	0h	DIVO Divider for 0B output: Bits [31:30]
		4	R/W	0h	PRG_DELAY[8:6] is the fine delay on the clock output. It determines relative delay on this clock programmable in 4,3, 2, 1, 0 times 30 ps based on this 3 bit code
		3	R/W	0h	{0, 0, 0, 0} External Termination, Differential Output {0, 0, 1, 0} Internal Pull Up, Differential Output {0, 0, 0, 1} Internal Pull Dn, Differential Output {0, 1, 0, 0} CMOS On OutP, Nothing on OutN {1, 0, 0, 0} Nothing on OutP, CMOS on OutN {1, 1, 0, 0} CMOS on OutP, CMOS on OutN
		2	R/W	0h	
		1	R/W	0h	
DIVO1B_MISC0_OUTPAGE	6Fh	0	R/W	0h	
PAGE_NUMBER	FFh	7:0	R/W	0h	On all pages register FF is a READ / WRITE register used to change the page number

Table 33 Page B: PLL B Related Registers (similar for Pages D)

The Registers from 10h to 2Fh are equivalent NVMCopy Registers for this Page.

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
STATUS_PLLB	02h	7:5	R	0h	SPARE
		4	R	0h	Dynamic status for pseq_prg_restart_ntfy_1
		3	R	0h	Dynamic status for pseq_prg_restart_ntfy_0
		2	R	0h	Dynamic status for porb_line_not_ok_defect
		1	R	0h	Dynamic status for fcal_done
		0	R	0h	Dynamic status of Loss of lock for PLLB
NOTIFY_PLLB	03h	7:5	R/W	1h	SPARE
		4	R/W	1h	Sticky/Notify status for pseq_prg_restart_ntfy_1
		3	R/W	1h	Sticky/Notify status for pseq_prg_restart_ntfy_0
		2	R/W	1h	Sticky/Notify status for porb_line_not_ok_defect
		1	R/W	1h	Sticky/Notify status for fcal_done
		0	R/W	1h	Sticky/Notify status Loss of lock for PLLB
MASKb_PLLB	04h	7:5	R/W	1h	SPARE
		4	R/W	1h	Mask bit for NOTIFY_PLLB (03h) If programmed as ‘0’: Mask sticky/Notify bit generation for 03h[4]
		3	R/W	1h	Mask bit for NOTIFY_PLLB (03h) If programmed as ‘0’: Mask sticky/Notify bit generation for 03h[3]
		2	R/W	1h	Mask bit for NOTIFY_PLLB (03h) If programmed as ‘0’: Mask sticky/Notify bit generation for 03h[2]
		1	R/W	1h	Mask bit for NOTIFY_PLLB (03h) If programmed as ‘0’: Mask sticky/Notify bit generation for 03h[1]
		0	R/W	1h	Mask bit for NOTIFY_PLLB (03h) If programmed as ‘0’: Mask sticky/Notify bit generation for 03h[0]
Directives_GENERIC_PLLB	05h	7	R/W	0h	Reserved
		6	R/W	0h	Spare
		5	R/W	1h	DLPF co-efficient selection provided by the GUI
		4	R/W	0h	Force external clock in switch
		3	R/W	0h	Force the PLL in holdover mode
		2	R/W	0h	Large change for resetting entire DIVO system: Edge triggered
		0	R/W	0h	Spare
STATUS_1_PLLB	06h	7:4	R	0h	SPARE
		3	R	0h	Dynamic status for cycle slip detection
		2	R	0h	Dynamic status for indicating holdover window is valid
		1	R	0h	Dynamic status for fast lock mode
		0	R	0h	Dynamic status for holdover
NOTIFY_1_PLLB	07h	7:4	R/W	1h	SPARE
		3	R/W	1h	Sticky/Notify status for cycle slip detection

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
NOTIFY_1_PLLB	07h	2	R/W	1h	Sticky bit indicating holdover window is valid
		1	R/W	1h	Sticky/Notify status for fast lock mode
		0	R/W	1h	Sticky/Notify status for holdover
MASKb_1_PLLB	08h	7:4	R/W	1h	SPARE
		3	R/W	1h	Mask bit for NOTIFY_1_PLLB (07h)If programmed as '0': Mask sticky/Notify bit generation for 07h[3]
		2	R/W	1h	Mask bit for NOTIFY_1_PLLB (07h)If programmed as '0': Mask sticky/Notify bit generation for 07h[2]
		1	R/W	1h	Mask bit for NOTIFY_1_PLLB FASTLOCK (07h)If programmed as '0': Mask sticky/Notify bit generation for 07h[1]
		0	R/W	1h	Mask bit for NOTIFY_1_PLLB (07h)If programmed as '0': Mask sticky/Notify bit generation for 07h[0]
PRG_Directives_PLLB	0Fh	7:3	R/W	0h	PRG_CMD Directives: 5b1_1000: PROGRAM_EFUSE 5b0_1100: READ_EFUSE 5b0_0110: Copy NVM Copy to Settings 5b1_1011: Proceed to Active
		2	R/W	0h	Spare
		1	R/W	0h	Escape to the PRG_CMD state from ACTIVE state
PPATH_PLLB	10h	7:5	R/W	0h	DLPF Settings from the GUI
		4:0	R/W	0h	
IPATH1_PLLB	11h	7:3	R/W	0h	
		2:0	R/W	0h	
IPATH2_PLLB	12h	7:5	R/W	0h	
		4:0	R/W	0h	
FASTLOCK_PPATH_PLLB	13h	7:5	R/W	0h	
		4:0	R/W	0h	
FASTLOCK_IPATH1_PLLB	14h	7:3	R/W	0h	
		2:0	R/W	0h	
FASTLOCK_IPATH2_PLLB	15h	7:5	R/W	0h	
		4:0	R/W	0h	
CYCLESLIP_MISC_CTRL_PLLB	16h	7	R/W	0h	CP gain configuration settings provided by GUI
		6	R/W	0h	
		5	R/W	0h	PLL loop filer configuration provided by GUI
		4	R/W	0h	PLL loop filer configuration provided by GUI
		3	R/W	0h	ZDB related setting provided by GUI
		2:1		0h	Cycle Slip detector threshold settings computed by GUI
		0	R/W	0h	Enable for cycle slip detector for DLPF
DIVNINT_PLLB	17h	7	R/W	0h	Master Disable for PLLB in sync mode and all associated functions
		6:0	R/W	0h	DIVN Integer part computed by GUI
MISCXO_PLLB	18h	7	R/W	0h	Enable ADC Dither
		6	R/W	0h	Offset Enable in the PLL Charge Pump
		5	R/W	0h	Select ACTIVE clock manually from manual input select pins when in manual active select mode
MISCXO_PLLB	18h	4	R/W	0h	DIVN Integer part computed by GUI

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
MISCXO_PLLB	18h	3:1	R/W	0h	Charge Pump Binning for VCO KV
		0	R/W	0h	0: Fractional Mode; 1: Integer Mode
DLPF_PHI_CORRECTION_PLLB	19h	7:6	R/W	0h	Phase averager configuration settings
		5:2	R/W	0h	Phase measurement filter BW setting (3 is spare)
		1	R/W	0h	Phase Correction Setting: Use default from GUI
DIVNFRAC2_PLLB	1Ah	7:0	R/W	0h	DIVN Fractional part computed by GUI
DIVNFRAC3_PLLB	1Bh	7:0	R/W	0h	DIVN Fractional part computed by GUI
DIVNFRAC4_PLLB	1Ch	7:0	R/W	0h	DIVN Fractional part computed by GUI
DIVN2_INT1_PLLB	1Dh	7:0	R/W	0h	DIVN2 Integer part computed by GUI
DIVN2_INT2_PLLB	1Eh	7:0	R/W	0h	DIVN2 Integer part computed by GUI
DIVN2_INT3_PLLB	1Fh	7	R/W	0h	Enable Revertive switching for input clock switching
		6	R/W	0h	Force manual selection of ACTIVE clock
		5	R/W	0h	Force Integer mode for the DIVN2 DSM
		4:0	R/W	0h	DIVN2 Integer part computed by GUI
DIVN2_FRN1_PLLB	20h	7:0	R/W	0h	DIVN2 Fractional part computed by GUI
DIVN2_FRN2_PLLB	21h	7:0	R/W	0h	DIVN2 Fractional part computed by GUI
DIVN2_FRN3_PLLB	22h	7:0	R/W	0h	DIVN2 Fractional part computed by GUI
DIVN2_FRN4_PLLB	23h	7:0	R/W	0h	DIVN2 Fractional part computed by GUI
DIVN2_FRD1_PLLB	24h	7:0	R/W	0h	DIVN2 Fractional part computed by GUI
DIVN2_FRD2_PLLB	25h	7:0	R/W	0h	DIVN2 Fractional part computed by GUI
DIVN2_FRD3_PLLB	26h	7:0	R/W	0h	DIVN2 Fractional part computed by GUI
DIVN2_FRD4_PLLB	27h	7:0	R/W	0h	DIVN2 Fractional part computed by GUI
OUTPUT_EN_PLLB	28h	7:6	R/W	0h	PSEQ_SYNC_DELAY 0: 500us 1: 8.17ms 2: 131.07ms 3: 2.09sec
		5:0	R/W	0h	One Hot Output Enable for Outputs 5:0
LL_REG1_PLLB	29h	7:6	R/W	0h	Programmable Phase Propagation Mode slope for the PLL 00: Use the Bandwidth 01: 10 usec/sec 10: 40 usec/sec 11: 160 usec/sec
		5	R/W	0h	Wait for Input Clock in power up in the PLL wake-up sequence
		4	R/W	0h	Use Fast Lock BW for exit from holdover to latch on fast to the new clock
		3:1	R/W	0h	Loss of Lock Delay = (2^(26-(2*LLDELAYTIMER))/4M). Wait for this delay time before announcing LL de-assertion
LL_REG1_PLLB	29h	0	R/W	0h	LL Clear Threshold {LL_CLR_VALUE_PLLB[1], 0x2A[0]} 2'b00 : 0.2 PPM 2'b01 : 0.4 PPM 2'b10 : 2 PPM 2'b11 : 200 PPM
LL_REG2_PLLB	2Ah	7:5	R/W	0h	LL_SET_VALUE_PLLB[7:4]

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
LL_REG2_PLLB	2Ah	4	R/W	0h	0 : 0.2 PPM 1 : 0.4 PPM 2 : 2 PPM 3 : 4 PPM 4 : 20 PPM 5 : 40 PPM 6 : 200 PPM 7 : 400 PPM 8 : 2000 PPM 9 : 4000 PPM 10: 0.2 PPM 11: 0.2 PPM 12: 0.2 PPM 13: 0.2 PPM 14: 0.2 PPM 15: 0.2 PPM
		3:1	R/W	0h	PLL_PSEQ_DELAY 0: 500us 1: 2ms 2: 8.19ms 3: 32.76ms 4: 131.07ms 5: 524.28ms 6: 2.09sec 7: 8.38sec
		0	R/W	0h	LL Clear Threshold {0x29[0], LL_CLR_VALUE_PLLB[0]} 2'b00 : 0.2 PPM 2'b01 : 0.4 PPM 2'b10 : 2 PPM 2'b11 : 200 PPM
HOLDOVER1_PLLB	2Bh	7:5	R/W	0h	Holdover Tdelay settings for the PLL computed by the GUI
		3:1	R/W	0h	Holdover Average settings for the PLL computed by the GUI
HOLDOVER_PLLB	2C	7	R/W	0h	Cycle Slip Detector related default
		6	R/W	0h	1: Enable Phase Propagation during input clock switch 0 : Enable Phase Build Out Mode where the phase difference between input clocks is absorbed by the PLL
		5	R/W	1h	Dither configuration for DIVN2 DSM
		4	R/W	0h	Enable revert to spare input clock during clock switching
		3:1	R/W	0h	DLPF related constant from the GUI
		0	R/W	0h	Internal voltage programming: Use default from GUI Profile
DECIMATION_RATIO_PLLB	2Dh	7	R/W	0h	Enable fast lock mode based on loss of lock status
DECIMATION_RATIO_PLLB	2Dh	6:3	R/W	0h	Internal Rate Change factors in DLPF computed by GUI
		2:0	R/W	0h	Internal Rate Change factors in DLPF computed by GUI
ONEBYR2_PLLB	2Eh	7:3	R/W	0h	Internal Rate Change factors in DLPF computed by GUI
		2:0	R/W	0h	Internal Rate Change factors in DLPF computed by GUI
LOCKPATTERN_PLLB	2Fh	7:6	R/W	0h	PATTERN: {'01'/'10'} =] Efuse Locked, {'00'/'11'} =] Efuse NOT Locked

Reg Name	Register Number	Bit Range	Access Type	Default Value	Description
LOCKPATTERN_PLLB	2Fh	5:4	R/W	0h	Program the frequency ramp slope from the following- 00: 0.2 ppm/s, 01: 2 ppm/s, 10: 20 ppm/s, 11: 200 ppm/s
		3	R/W	0h	Enable the frequency ramp feature
		2:0	R/W	0h	Internal Frequency Divider: Computed by the GUI
DCO_FRAC1_PLLB	31h	7:0	R/W	0h	DCO fractional control code
DCO_FRAC2_PLLB	32h	7:0	R/W	0h	DCO fractional control code
DCO_FRAC3_PLLB	33h	7:0	R/W	0h	DCO fractional control code
DCO_FRAC4_PLLB	34h	7:0	R/W	0h	DCO fractional control code
DCO_FUNCTION_PLLB	35h	7:4	R/W	0h	Reserved
		3	R/W	0h	DCO Increment in Frequency from registers
		2	R/W	0h	DCO Decrement in Frequency from registers
		1	R/W	0h	Enable DCO free run mode
		0	R/W	0h	DCO Mask for this PLL
DCO_BUMP2_PLLB	36h	7:0	R/W	0h	DCO integer control code
DCO_BUMP3_PLLB	37h	7:6	R/W	0h	DCO integer control code
		5	R/W	0h	Enable DCO sync mode
		4	R/W	0h	SPARE
		3:0	R/W	0h	SPARE
PAGE_NUMBER	FFh	7:0	R/W	0h	On all pages register FF is a READ / WRITE register used to change the page number

26 Ordering Information

Table 34 Ordering Information for AU5329

Ordering Part Number (OPN)	Marking	No. of Input/ Output Clocks	Output Clock Frequency Range (MHz)	Supported Frequency Synthesis modes	VDDIN (VDDA), VDD	Package	Temp Range
AU5329							
AU5329B00-QMR[1][2]	AU5329B	2/10	8KHz – 2.1GHz	Integer and Fractional	3.3,2.5	64-QFN 9x9 mm	-40 to 85°C
AU5329B00-QMT[1][2]	AU5329B	2/10	8KHz – 2.1GHz	Integer and Fractional	3.3,2.5	64-QFN 9x9 mm	-40 to 85°C
AU5329Cx-EVB		—	—	—	—	Evaluation Board	

Notes:

1. Add an R at the end of the OPN to denote tape and reel ordering option. Add a T at the end of the OPN to denote tray ordering option.
2. Custom and factory preprogrammed devices are available. Ordering part numbers are assigned by Aurasemi, please contact local sales to request the unique part number. Custom part number format is “AU5329BZZ-QM” where “ZZ” is a unique numerical sequence representing the preprogrammed configuration.

27 Revision History

Table 35 Revision History

Version	Date	Description	Author
0.1	21st April, 2020	AU5329 Data Sheet Pre-Production Release	Aurasemi
1.0	11th Jan 2021	1. Aurasemi Logo changed and updated 2. Table 34 Ordering Information for AU5329 Updated 3. Figure 8 AU5329 PLL Divider updated 4. In Table 14 Power Supply Rejection, PSRR_VDD Specs Updated 5. Table 1 Pin 48 description updated. 6. Note #1 after Table 4 updated for AU5329 only. 7. Final Production (MP) Datasheet Released.	Aurasemi
1.1	16th April 2021	1. AU5329A and AU5329B register information added in Table 28 for register 01h. 2. Table 34 updated with the Ordering Information Details. 3. Added # 3 below Table 34 for AU5329A and AU5329B usage recommendation.	Aurasemi
1.2	26th May 2021	Added 'Meets G.8262 EEC Option 1,2 (Sync E)' to Features	Aurasemi
1.3	5th July 2021	1. Table 34 changed to update the Ordering Part Number and Marking information. 2. Table 19 I2C Bus Timing updated with standard and fast mode specifications	Aurasemi
1.4	10th Dec 2021	1. Added Table 12 PCI Express Jitter Performance (Spread Spectrum = OFF)	Aurasemi
1.5	24th May 2021	1. SPI Timing Read Figure 18 updated. 2. MSL Information included in Table 2 3. Additional recommendation Note for improving XO slew rate added below Figure 35 4.thd:DAT min spec changed to 10ns in Table 19 5.Table 34 Ordering Information table updated with AU5329B only information. 6. Table 12 updated for the PCIe Express Jitter Mean and Max values	Aurasemi

IMPORTANT NOTICE AND DISCLAIMER

Contact Information

For more information visit www.aurasemi.com

For sales related information please send an email to sales@aurasemi.com

Trademarks