Zynq UltraScale+ Processing System v1.2

LogiCORE IP Product Guide

PG201 June 8, 2016

XILINX
ALL PROGRAMMABLE™

Table of Contents

Introduction

The Xilinx® Zynq® UltraScale+™ Processing System LogiCORE™ IP core is the software interface around the Zynq UltraScale+ Processing System. The Zynq UltraScale+ MPSoC family consists of a system-on-chip (SoC) style integrated processing system (PS) and a Programmable Logic (PL) unit, providing an extensible and flexible SoC solution on a single die.

Features

- Enable/Disable I/O Peripherals (IOP)
- Enable/Disable AXI I/O ports
- Multiplexed I/O (MIO) Configuration
- Extended Multiplexed I/Os (EMIO)
- PL Clocks and Interrupts, resets
- Interconnect logic for Vivado Design Suite IP – PS interface
- PS internal clocking
- Generation of System Level Configuration Registers (SLCRs)
- High Speed SerDes Configuration

LogiCORE IP Facts Table	
Core Specifics	
Supported Device Family(1)	Zynq UltraScale+ MPSoC
Supported User Interfaces	Not Applicable
Resources	Not Applicable
Provided with Core	
Design Files	Verilog
Example Design	See Chapter 5, Example Design.
Test Bench	Not Provided
Constraints File	Not Provided
Simulation Model	Not Provided
Supported S/W Driver	N/A
Tested Design Flows(2)	
Design Entry	Vivado Design Suite
Simulation	Not Applicable
Synthesis	Vivado Synthesis
Support	
Provided by Xilinx @ Xilinx Support web page	

Notes:

1. For a complete list of supported devices, see Vivado IP catalog.
2. For the supported versions of the tools, see the Xilinx Design Tools: Release Notes Guide.

Overview

The Zynq® UltraScale+™ MPSoC family is based on the Xilinx All Programmable system-on-chip (AP MPSoC) architecture. The Zynq UltraScale+ Processing System core acts as a logic connection between the PS and the Programmable Logic (PL) while assisting you to integrate customized and integrated IP cores with the processing system using the Vivado® IP integrator.

For a detailed overview of the core, see Chapter 2, Product Specification.

Feature Summary

See Features on the IP Facts page.

Unsupported Features and Known Limitations

The core provides a Vivado Integrated Design Environment (Vivado IDE) configuration of the PS instance and its I/O. Due to the flexibility of the PS, only the most common features, I/O configurations, and peripheral settings are configured by this core. Additional register settings might be necessary by your own register accesses.

Xilinx frequently updates the list of known issues each release, for the most up to date information always access the master Answer Record 66183, *Zynq UltraScale+ MPSoC Processing System IP - Release Notes and Known Issues*.

Licensing and Ordering Information

This Xilinx LogiCORE™ IP module is provided at no additional cost with the Xilinx Vivado Design Suite under the terms of the Xilinx End User License. Information about this and other Xilinx LogiCORE IP modules is available at the Xilinx Intellectual Property page. For information about pricing and availability of other Xilinx LogiCORE IP modules and tools, contact your local Xilinx sales representative.

For more information, visit the Zynq UltraScale+ MPSoC Processing System IP product page.

Product Specification

Functional Description

The Zynq® UltraScale+™ MPSoC Processing System wrapper instantiates the processing system section of the Zynq UltraScale+ MPSoC for the programmable logic and external board logic. The wrapper includes unaltered connectivity and, for some signals, some logic functions. For a description of the architecture of the processing system, see the *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085) [Ref 1].

The core connects the interface signals with the rest of the embedded system in the programmable logic. The interfaces between the processing system and programmable logic mainly consist of three main groups: the extended multiplexed I/O (EMIO), programmable logic I/O, and the AXI I/O groups. The device configuration wizard configures the Zynq UltraScale+ MPSoC Processing System core. The core performs the functions described in the following subsections.

Figure 2-1 shows a top-level block diagram.

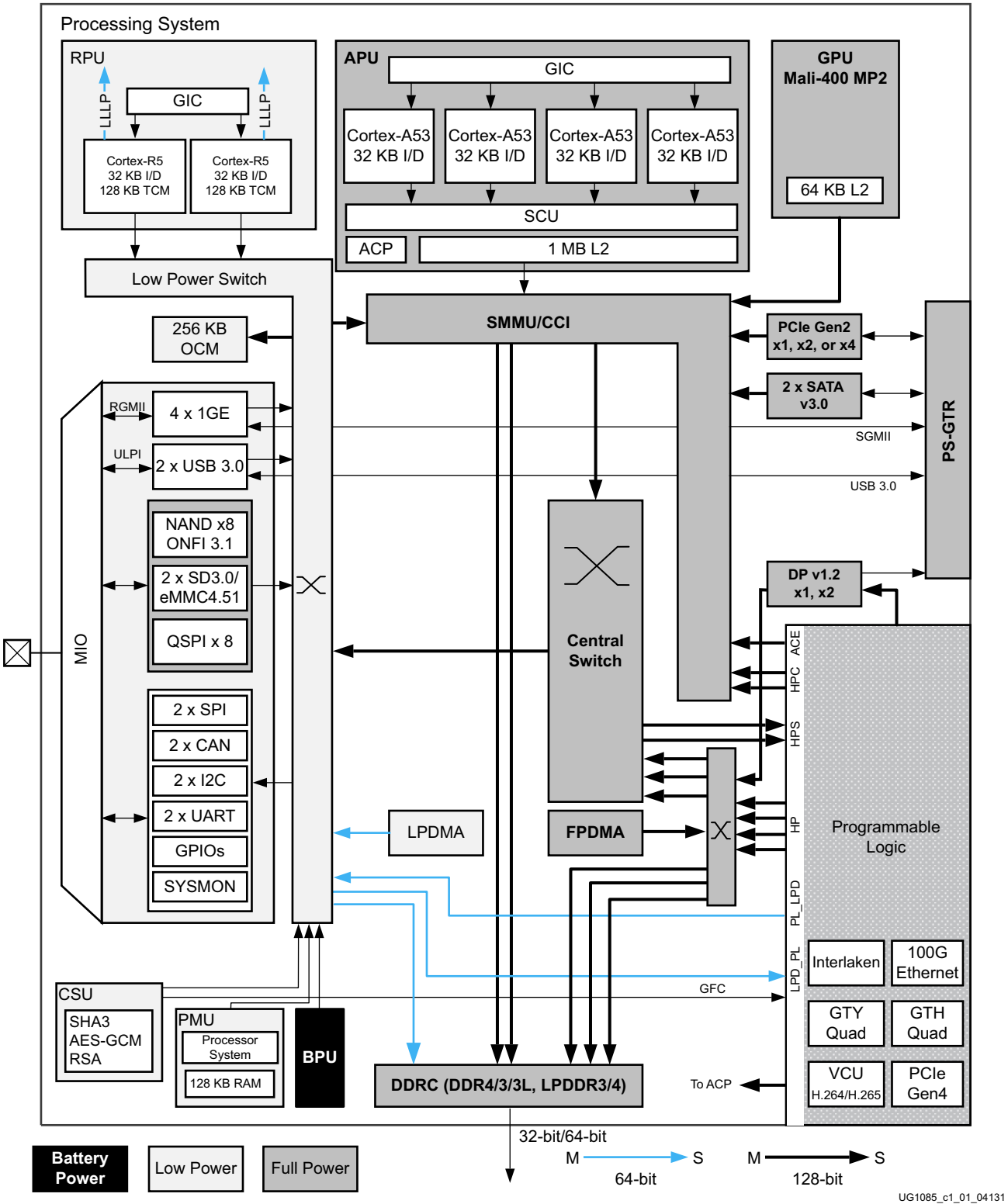


Figure 2-1: **Zynq UltraScale+ MPSoc Top Level Block Diagram**

Connectivity

`ddr`, `mio`, `por/clk/srst` ports are unaltered.

- `fclk` are also made of individual signals instead of the array `FCLKCLK (3:0)`.
- PS PL IRQ are made of individual signals ps_pl_irq_can0, ps_pl_irq_can1, ps_pl_irq_enet0, ps_pl_irq_enet1, ps_pl_irq_enet2, ps_pl_irq_enet3, ps_pl_irq_enet0_wake0, ps_pl_irq_enet0_wake1, ps_pl_irq_enet0_wake2, ps_pl_irq_enet0_wake3, ps_pl_irq_gpio, ps_pl_irq_i2c0, ps_pl_irq_i2c1, ps_pl_irq_uart0, ps_pl_irq_uart1, ps_pl_irq_sdio0, ps_pl_irq_sdio1, ps_pl_irq_sdio0_wake, ps_pl_irq_sdio1_wake, ps_pl_irq_spi0, ps_pl_irq_spi1, ps_pl_irq_qspi, ps_pl_irq_ttc0_0, ps_pl_irq_ttc0_1, ps_pl_irq_ttc0_2, ps_pl_irq_ttc1_0, ps_pl_irq_ttc1_1, ps_pl_irq_ttc1_2, ps_pl_irq_ttc2_0, ps_pl_irq_ttc2_1, ps_pl_irq_ttc2_2, ps_pl_irq_ttc3_0, ps_pl_irq_ttc3_1, ps_pl_irq_ttc3_2, ps_pl_irq_csu_pmu_wdt, ps_pl_irq_lp_wdt, ps_pl_irq_usb3_0_endpoint, ps_pl_irq_usb3_0_otg, ps_pl_irq_usb3_1_endpoint, ps_pl_irq_usb3_1_otg, ps_pl_irq_adma_chan, ps_pl_irq_usb3_0_pmu_wakeup, ps_pl_irq_gdma_chan, ps_pl_irq_csu, ps_pl_irq_csu_dma, ps_pl_irq_efuse, ps_pl_irq_xmpu_lpd, ps_pl_irq_ddr_ss, ps_pl_irq_nand, ps_pl_irq_fp_wdt, ps_pl_irq_pcie_msi, ps_pl_irq_pcie_legacy, ps_pl_irq_pcie_dma, ps_pl_irq_pcie_msc, ps_pl_irq_dport, ps_pl_irq_fpd_apb_int, ps_pl_irq_fpd_atb_error, ps_pl_irq_dpdma, ps_pl_irq_apm_fpd, ps_pl_irq_gpu, ps_pl_irq_sata, ps_pl_irq_xmpu_fpd, ps_pl_irq_apu_cpumnt, ps_pl_irq_apu_cti, ps_pl_irq_apu_pmu, ps_pl_irq_apu_comm, ps_pl_irq_apu_l2err, ps_pl_irq_apu_exterr, ps_pl_irq_apu_regs, ps_pl_irq_intf_ppd_cci, ps_pl_irq_intf_fpd_smmu, ps_pl_irq_atb_err_lpd, ps_pl_irq_aib_axi, ps_pl_irq_ams, ps_pl_irq_lpd_apm ps_pl_irq_rtc_alaram, ps_pl_irq_rtc_seconds, ps_pl_irq_clkmon, ps_pl_irq_pl_ipi, ps_pl_irq_rpu_ipi, ps_pl_irq_apu_ipi, ps_pl_irq_rpu_pm, ps_pl_irq_ocm_error, ps_pl_irq_lpd_apb_intr, ps_pl_irq_r5_core0_ecc_error, and ps_pl_irq_r5_core1_ecc_error.
- `spi or spi* sson` are made of individual signals `spi*_ss2_o`, `spi*_ss1_o`, and `spi*_ss_o`.

I/O Peripherals

I/O Peripherals (IOP) include the following.

- Quad serial peripheral interface (SPI) flash memory
- NAND flash
- UART
- I2C
- SPI flash memory
- secure digital Input Output (SDIO)
- general purpose I/O (GPIO)
- controller area network (CAN)

- USB
- Ethernet

The interfaces for these I/O peripherals (IOPs) can be routed to MIO ports and the extended multiplexed I/O (EMIO) interfaces as described in the *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085) [Ref 1].

- Low power domain (LPD) peripherals available in PS:
  - 4 X Gigabit Ethernet
  - 2 X USB3
  - 2 X SDIO
  - 2 X SPI
  - 2 X CAN
  - 2 X I2C
  - 2 X UART
  - NAND Controller
  - Quad SPI flash memory
  - Controller, GPIOs
  - System Monitor
- Full power domain (FPD) peripherals available in PS:
  - PCIe® Gen2
  - 2 X Serial Advanced Technology Attachment (SATA)
  - Display Port V1.2

MIO Ports

The Zynq UltraScale+ MPSoC design tools are used to configure the core MIO ports. There are up to 78 MIO ports available from the processing system. The wizard allows you to choose the peripheral ports to be connected to MIO ports.

Extended MIO Ports

Because there are only up to 78 MIO available ports, many peripheral I/O ports beyond these can still be routed to the programmable logic through the Extended MIO (EMIO) interface. Alternative routing for IOP interfaces through programmable logic enables you to take full advantage of the IOP available in the processing system.

The EMIO for I2C, SPI flash memory, Ethernet management data input/output (MDIO), ARM® JTAG (PJTAG), SDIO, GPIO 3-state enable signals are inverted in the Zynq UltraScale+ MPSoC Processing System core.

The Zynq UltraScale+ MPSoC Processing System core allows you to select GPIO up to 96 bits. The Zynq UltraScale+ MPSoC Processing System has control logic to adjust user-selected width to flow into processing system.

See MIO Voltage Standard in Chapter 4.

AXI4 I/O Compliant Interfaces

Following are the AMBA® AXI4 compliant interfaces:

- Three PS General Purpose Master interfaces user configurable as 32, 64, and 128 bits in width. The default width is 128.
- Seven PL General Purpose Master interfaces user configurable as 32, 64, and 128 bits in width. The default width is 128.
- A 128-bit PL Master AXI coherency extension (ACE) interface for coherent I/O to A53 L1 and L2 cache systems
- A 128-bit PL Master ACP interface to support L2 cache allocation from PL masters. Limited to 64-byte cache line transfers only

See PS-PL Configuration in Chapter 4.

Logic for Vivado Design Suite IP - Processing System Interface

The Zynq UltraScale+ MPSoC Processing System core allows you to add Vivado® IP cores in the programmable logic to interface with the processing system. Custom direct memory access (DMA) functions can be implemented in the PL to oversee data movement irrespective of the processor intervention.

Programmable Logic Clocks and Interrupts

The interrupts from the processing system I/O peripherals (IOP) are routed to the PL and assert asynchronously to the `fclk` clocks.

The PL can asynchronously assert up to 20 interrupts to the PS.

- 16 interrupt signals are mapped to the interrupt controller as a peripheral interrupt where each interrupt signal is set to a priority level and mapped to one or both of the CPUs. To use more that one interrupt signal, use a Concat block in the Vivado IP integrator to automatically size the width of the interrupt vector.
- The remaining four PL interrupt signals are inverted and routed to the `nFIQ` and `nIRQ` interrupt directly to the signals to the private peripheral interrupt (PPI) unit of the interrupt controller. There is an `nFIQ` and `nIRQ` interrupt for each of two CPUs.

The PS to PL, and PL to PS interrupts are listed in Table 2-1. For details on the interrupt signals, see the Interrupts chapter in the *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085) [Ref 1].

See PS-PL Configuration in Chapter 4 for Vivado Design Suite implementation.

Table 2-1: **Interrupt Map for PS Configuration Wizard (PCW)**

S.No	Interrupt ID	Interrupt Name	Description	Type
PL-PS Interrupts (Interrupts that go from PL to PS)				
1	121-128, 137-144	IRQ-F2P[15:0]	Shared Interrupts from PL logic to GICs of real-time processing unit (RPU) or application processing unit (APU)	Shared Interrupts
2	31	A53-Core_0 nIRQ	Cortex™ A53 Core0 Private Peripheral Legacy IRQ Interrupt	Private Peripheral Interrupt
3	31	A53-Core_1 nIRQ	Cortex A53 Core1 Private Peripheral Legacy IRQ Interrupt	Private Peripheral Interrupt
4	31	A53-Core_2 nIRQ	Cortex A53 Core2 Private Peripheral Legacy IRQ Interrupt	Private Peripheral Interrupt
5	31	A53-Core_3 nIRQ	Cortex A53 Core3 Private Peripheral Legacy IRQ Interrupt	Private Peripheral Interrupt
6	28	A53-Core_0 nFIQ	Cortex A53 Core0 Private Peripheral Legacy FIQ Interrupt	Private Peripheral Interrupt
7	28	A53-Core_1 nFIQ	Cortex A53 Core1 Private Peripheral Legacy FIQ Interrupt	Private Peripheral Interrupt

Table 2-1: **Interrupt Map for PS Configuration Wizard (PCW) *(Cont'd)***

S.No	Interrupt ID	Interrupt Name	Description	Type
8	28	A53-Core_2 nFIQ	Cortex A53 Core2 Private Peripheral Legacy FIQ Interrupt	Private Peripheral Interrupt
9	28	A53-Core_3 nFIQ	Cortex A53 Core3 Private Peripheral Legacy FIQ Interrupt	Private Peripheral Interrupt
PS -PL Interrupts (Interrupts coming from PS to PL)				
1	1	IRQ_P2F_RPU Performance Monitor 0	RPU Performance Monitor 0 Interrupt	Shared Interrupt
2	1	IRQ_P2F_RPU Performance Monitor 1	RPU Performance Monitor 1 Interrupt	Shared Interrupt
3	1	IRQ_P2F_OCM Error	On-chip RAM (OCM) Error Interrupt	Shared Interrupt
4	1	IRQ_P2F_LPD APB Interrupts	OR of all AMBA peripheral bus (APB) interrupts from LPD. Refer to the technical reference manual for APB Interrupt and Register Information.	Shared Interrupt
5	1	IRQ_P2F_R5 Core0_ECC _Error	RPU CPU0 error-correction code (ECC) errors interrupt. All ECC interrupts of CPU0 are combined into this interrupt.	Shared Interrupt
6	1	IRQ_P2F_R5 Core1_ECC_Error	RPU CPU1 ECC errors interrupt. All ECC interrupts of CPU1 are combined into this interrupt.	Shared Interrupt
7	1	IRQ_P2F_NAND	NAND/NOR/SRAM Static Memory Controller Interrupt	Shared Interrupt
8	1	IRQ_P2F_QSPI	SPI flash memory interrupt	Shared Interrupt
9	1	IRQ_P2F_GPIO	GPIO interrupt	Shared Interrupt
10	1	IRQ_P2F_I2C0	I2C0 interrupt	Shared Interrupt
11	1	IRQ_P2F_I2C1	I2C1 interrupt	Shared Interrupt
12	1	IRQ_P2F_SPI0	SPI0 interrupt	Shared Interrupt
13	1	IRQ_P2F_SPI1	SPI1 interrupt	Shared Interrupt
14	1	IRQ_P2F_UART0	UART0 interrupt	Shared Interrupt
15	1	IRQ_P2F_UART1	UART1 interrupt	Shared Interrupt
16	1	IRQ_P2F_CAN0	CAN0 interrupt	Shared Interrupt

Table 2-1: **Interrupt Map for PS Configuration Wizard (PCW) *(Cont'd)***

S.No	Interrupt ID	Interrupt Name	Description	Type
17	1	IRQ_P2F_CAN1	CAN1 interrupt	Shared Interrupt
18	1	IRQ_P2F_LPD_APM	Or of all LPD AXI performance monitors (APMs)	Shared Interrupt
19	1	IRQ_P2F_RTC_ALARM	RTC Alarm Interrupt	Shared Interrupt
20	1	IRQ_P2F_RTC_SECONDS	RTC Seconds Interrupt	Shared Interrupt
21	1	IRQ_P2F_CLKMON	Clock monitor coming from CRL	Shared Interrupt
22	1	IRQ_P2F_PL_IPI0	OR' of all of inter-processor interrupt (IPIs) targeted to RPU PL0	Shared Interrupt
23	1	IRQ_P2F_PL_IPI1	OR' of all of IPIs targeted to RPU PL1	Shared Interrupt
24	1	IRQ_P2F_PL_IPI2	OR' of all of IPIs targeted to RPU PL2	Shared Interrupt
25	1	IRQ_P2F_PL_IPI3	OR' of all of IPIs targeted to RPU PL3	Shared Interrupt
26	1	IRQ_P2F_RPU_IPI0	OR' of all of IPIs targeted to RPU CPU0	Shared Interrupt
27	1	IRQ_P2F_RPU_IPI1	OR' of all of IPIs targeted to RPU CPU1	Shared Interrupt
28	1	IRQ_P2F_APU_IPI0	OR' of all of IPIs targeted to APU CPU	Shared Interrupt
29	1	IRQ_P2F_TTC0_0	Triple Timer 0 Counter 0 Interrupt	Shared Interrupt
30	1	IRQ_P2F_TTC0_1	Triple Timer 0 Counter 1 Interrupt	Shared Interrupt
31	1	IRQ_P2F_TTC0_2	Triple Timer 0 Counter 2 Interrupt	Shared Interrupt
32	1	IRQ_P2F_TTC1_0	Triple Timer 1 Counter 0 Interrupt	Shared Interrupt
33	1	IRQ_P2F_TTC1_1	Triple Timer 1 Counter 1 Interrupt	Shared Interrupt
34	1	IRQ_P2F_TTC1_2	Triple Timer 1 Counter 2 Interrupt	Shared Interrupt
35	1	IRQ_P2F_TTC2_0	Triple Timer 2 Counter 0 Interrupt	Shared Interrupt
36	1	IRQ_P2F_TTC2_1	Triple Timer 2 Counter 1 Interrupt	Shared Interrupt
37	1	IRQ_P2F_TTC2_2	Triple Timer 2 Counter 2 Interrupt	Shared Interrupt

Table 2-1: **Interrupt Map for PS Configuration Wizard (PCW) *(Cont'd)***

S.No	Interrupt ID	Interrupt Name	Description	Type
38	1	IRQ_P2F_TTC3_0	Triple Timer 3 Counter 0 Interrupt	Shared Interrupt
39	1	IRQ_P2F_TTC3_1	Triple Timer 3 Counter 1 Interrupt	Shared Interrupt
40	1	IRQ_P2F_TTC3_2	Triple Timer 3 Counter 2 Interrupt	Shared Interrupt
41	1	IRQ_P2F_SDIO0	SDIO0 interrupt	Shared Interrupt
42	1	IRQ_P2F_SDIO1	SDIO1 interrupt	Shared Interrupt
43	1	IRQ_P2F_SDIO0_wake	SDIO0 wake interrupt	Shared Interrupt
44	1	IRQ_P2F_SDIO1_wake	SDIO1 wake interrupt	Shared Interrupt
45	1	IRQ_P2F_LP_WDT	Watchdog timer (WDT) in the LPD (IOU) (IOU is Input Output Unit)	Shared Interrupt
46	1	IRQ_P2F_CSUPMU_WDT	WDT in the Configuration Security Unit Performance monitoring unit (CSUPMU)	Shared Interrupt
47	1	IRQ_P2F_ATB Err LPD	AMBA trace bus (ATB) interrupt	Shared Interrupt
48	1	IRQ_P2F_AIB_AXI	AXI Isolation Block (AIB) AXI interrupt	Shared Interrupt
49	1	IRQ_P2F_AMS	Analog mixed-signal unit (AMS) interrupt	Shared Interrupt
50	1	IRQ_P2F_GigabitEth0	Ethernet0 interrupt	Shared Interrupt
51	1	IRQ_P2F_GigabitEth_ Wake0	Ethernet0 wake-up interrupt	Shared Interrupt
52	1	IRQ_P2F_GigabitEth1	Gigabit Ethernet1 interrupt	Shared Interrupt
53	1	IRQ_P2F_GigabitEth_ wakeup1	Gigabit Ethernet1 wake-up interrupt	Shared Interrupt
54	1	IRQ_P2F_GigabitEth2	Gigabit Ethernet2 interrupt	Shared Interrupt
55	1	IRQ_P2F_GigabitEth2_ wakeup	Gigabit Ethernet2 wake-up interrupt	Shared Interrupt
56	1	IRQ_P2F_GigabitEth3	Gigabit Ethernet3 interrupt	Shared Interrupt
57	1	IRQ_P2F_GigabitEth3_ wake up	Gigabit Ethernet3 wake-up interrupt	Shared Interrupt

Table 2-1: **Interrupt Map for PS Configuration Wizard (PCW) *(Cont'd)***

S.No	Interrupt ID	Interrupt Name	Description	Type
58	4	IRQ_P2F_USB3_0_Endpoint	USB3_0 Endpoint related interrupts. Four Interrupts Enabled. One interrupt each for Bulk, Isochronous, Interrupt and Control type.	Shared Interrupt
59	1	IRQ_P2F_USB3_0_OTG	USB3_0 OTG interrupt	Shared Interrupt
60	4	IRQ_P2F_USB3_1_Endpoint	USB3_1 Endpoint related interrupts. Four Interrupts Enabled. One interrupt each for Bulk, Isochronous, Interrupt and Control type.	Shared Interrupt
61	1	IRQ_P2F_USB3_1_OTG	USB3_1 OTG interrupt	Shared Interrupt
62	1	IRQ_P2F_USB3_0_1 PMU_WAKEUP	Bit 0 is wake up from USB3_0 to power monitoring unit (PMU) while bit 1 is wake up from USB3_1 to PMU	Shared Interrupt
63	1	IRQ_P2F_ADMA_Chan_0	ACP DMA (ADMA) channel 0 interrupt	Shared Interrupt
64	1	IRQ_P2F_ADMA_Chan_1	ADMA channel 1 interrupt	Shared Interrupt
65	1	IRQ_P2F_ADMA_Chan_2	ADMA channel 2 interrupt	Shared Interrupt
66	1	IRQ_P2F_ADMA_Chan_3	ADMA channel 3 interrupt	Shared Interrupt
67	1	IRQ_P2F_ADMA_Chan_4	ADMA channel 4 interrupt	Shared Interrupt
68	1	IRQ_P2F_ADMA_Chan_5	ADMA channel 5 interrupt	Shared Interrupt
69	1	IRQ_P2F_ADMA_Chan_6	ADMA channel 6 interrupt	Shared Interrupt
70	1	IRQ_P2F_ADMA_Chan_7	ADMA channel 7 interrupt	Shared Interrupt
71	1	IRQ_P2F_CSU	Device Configuration Module Interrupt	Shared Interrupt
72	1	IRQ_P2F_CSU_DMA	DMA for Configuration and Security Unit (CSU) interrupt	Shared Interrupt
73	1	IRQ_P2F_EFUSE	EFUSE interrupt	Shared Interrupt
74	1	IRQ_P2F_XMPU_LPD	Xilinx memory protection unit (XMPU) error Interrupt for LPD	Shared Interrupt
75	1	IRQ_P2F_DDR_SS	DDR controller subsystem interrupt	Shared Interrupt
76	1	IRQ_P2F_FP_WDT	Top Level Watch Dog Timer Interrupt.	Shared Interrupt

Table 2-1: **Interrupt Map for PS Configuration Wizard (PCW) *(Cont'd)***

S.No	Interrupt ID	Interrupt Name	Description	Type
77	1	IRQ_P2F_PCIE_MSI	PCIE_MSI[0]=PCIe interrupt for MSI vectors 31 to 0 PCIE_MSI[1]=PCIe interrupt for MSI vectors 63 to 32	Shared Interrupt
78	1	IRQ_P2F_PCIE_Legacy	PCIE legacy (INTA/BC/D) interrupts	Shared Interrupt
79	1	IRQ_P2F_PCIE_DMA	PCIE Bridge DMA interrupts	Shared Interrupt
80	1	IRQ_P2F_PCIE_MSC	PCIE misc (error etc) interrupts	Shared Interrupt
81	1	IRQ_P2F_DPORT	Display port general purpose interrupt	Shared Interrupt
82	1	IRQ_P2F_FPD_APB_INT	OR'd of all APB interrupts from LPD	Shared Interrupt
83	1	IRQ_P2F_FPD ATB Error	ATB interrupt for FPD	Shared Interrupt
84	1	IRQ_P2F_DPDMA interrupt	DPDMA interrupt	Shared Interrupt
85	1	IRQ_P2F_APM FPD	Or of all APMs for FPD	Shared Interrupt
86	1	IRQ_P2F_GDMA_Chan_0	Interrupt from general purpose DMA (GDMA) Channel 0	Shared Interrupt
87	1	IRQ_P2F_GDMA_Chan_1	Interrupt from GDMA Channel 1	Shared Interrupt
88	1	IRQ_P2F_GDMA_Chan_2	Interrupt from GDMA Channel 2	Shared Interrupt
89	1	IRQ_P2F_GDMA_Chan_3	Interrupt from GDMA Channel 3	Shared Interrupt
90	1	IRQ_P2F_GDMA_Chan_4	Interrupt from GDMA Channel 4	Shared Interrupt
91	1	IRQ_P2F_GDMA_Chan_5	Interrupt from GDMA Channel 5	Shared Interrupt
92	1	IRQ_P2F_GDMA_Chan_6	Interrupt from GDMA Channel 6	Shared Interrupt
93	1	IRQ_P2F_GDMA_Chan_7	Interrupt from GDMA Channel 7	Shared Interrupt
94	1	IRQ_P2F_GPU	All of GPU interrupts are OR-ed together	Shared Interrupt
95	1	IRQ_P2F_SATA	SATA controller interrupt	Shared Interrupt
96	1	IRQ_P2F_XMPU FPD	XMPU error interrupt for all of FPD	Shared Interrupt

Table 2-1: **Interrupt Map for PS Configuration Wizard (PCW) *(Cont'd)***

S.No	Interrupt ID	Interrupt Name	Description	Type
97	4	IRQ_P2F_APU_CPUMNT	VCPUMT	Shared Interrupt
98	4	IRQ_P2F_APU_CTI	Cross trigger interface (CTI)	Shared Interrupt
99	4	IRQ_P2F_APU_PMU	Performance Monitor Unit Interrupt	Shared Interrupt
100	4	IRQ_P2F_APU_COMM	APU Communication Error	Shared Interrupt
101	1	IRQ_P2F_APU_L2ERR	L2 Error	Shared Interrupt
102	1	IRQ_P2F_APU_EXTERR	EXTERR	Shared Interrupt
103	1	IRQ_P2F_APU_REGS	REGISTER Interrupt	Shared Interrupt
104	1	IRQ_P2F_INTF_PPD_CCI	Cache coherent interconnect (CCI) Interrupt from FPD	Shared Interrupt
105	1	IRQ_P2F_INTF_FPD_SMMU	System Memory Management Unit (SMMU) Interrupt from FPD	Shared Interrupt

The Zynq UltraScale+ MPSoC Processing System core employs logic to handle PL interrupts, the number which varies from 1 to 16 depending on your selection. The number of interrupts connected to IRQ_F2P are calculated and the logic ensures the correct order of an interrupt assignment.

The Zynq UltraScale+ MPSoC Processing System interrupts from IOPs are available to custom master interfaces in PL.

PL Clocks

The Zynq UltraScale+ MPSoC Processing System provides four clocks to the PL. Zynq UltraScale+ MPSoC Processing System enables configuration of these clocks to be used in the PL. Zynq UltraScale+ MPSoC Processing System inserts a BUFG for each of the PL clocks through parameters similar to C_FCLK_CLK0_BUF. Fabric clocks are configured for 100 MHz by default.

Standards

The Zynq UltraScale+ MPSoC Processing System core is compatible with the AXI4 Interface. AXI interfaces can be used by an AXI4-compliant master or slave connected to the ARM core.

See the "Interconnect" chapter in the *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085) [Ref 1].

Performance

For information, see the "PL and Memory System Performance Overview" section in the "Programmable Logic Design Guide" chapter of the *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085) [Ref 1].

Maximum Frequencies

For information, see the *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085) [Ref 1]

Latency

For information, see the *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085) [Ref 1]

Throughput

For information, see the *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085) [Ref 1]

Power

For information, see the *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085) [Ref 1]

Resource Utilization

Zynq UltraScale+ MPSoC is a hard IP core so this section does not apply to this core.

Port Descriptions

See Appendix B, Port Descriptions for all of the tables.

Register Space

Note: For register information, see the *Zynq UltraScale+ MPSoC Register Reference User Guide* (UG1087) [Ref 2].

The Zynq UltraScale+ MPSoC Processing System core provides access from PL masters to PS internal peripherals, and memory through AXI FIFO interface (AFI) interfaces. The Vivado IP integrator address editor provides various address segments with a fixed address for each slave interface. The availability of the address segments is controlled through the following addressing parameters.

- **Detailed IOP address space**: Provides individual address spaces for PS internal peripherals.
- **Allow access to PS/SLCR registers**: Allows address mapping to PS and system level control registers (SLCR) register space.
- **Detailed PS/SLCR address space**: Provides individual address spaces for PS/SLCR registers.

The PS address space accessible from the PL consists of DDR, OCM, static memory controller (SMC) memories, SLCR registers, PS I/O peripheral registers, and PS system registers. For more information, see the "System Addresses" chapter of the *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085) [Ref 1].

Designing with the Core

This chapter includes guidelines and additional information to facilitate designing with the core.

General Design Guidelines

There are three interfaces through which the Zynq® UltraScale+™ Processing System core can access the PL side peripherals and vice versa. For more details, see the individual sections of AXI_HP and AXI_ACP interfaces in the "Interconnect" chapter of the *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085) [Ref 1].

Interrupts

To connect multiple interrupts in IP integrator, use a concat block to merge the individual signals together before connecting to the core. The interrupt port will be automatically expanded to match the resulting output with the concat block.

Clocking

There are four clock groups.

- Main Clock Group (MCG). This group has five PLLs.
  - I/O PLL
  - RPU PLL
  - APU PLL
  - DDR PLL
  - Video PLL

- Secure Clock Group (SCG). This group has two PLLs.
  - eFuse
  - PMU
- RTC Clock Group (RCG). There is a clock provided explicitly to RTC in Battery Power unit (BPU.)
- Interface Clock Group (ICG). This group has clocks that are provided externally, like clocks from physical-side interface (PHY) and PL.

PL side peripherals can be operated through a fabric clock (FCLK_CLK0...3). They generate the frequency ranges from 0.1 to 250 MHz.

Resets

There are many applicable resets:

- External power on reset (POR) - Triggered by external pin assertion
- Internal POR - Triggered by software register write or safety errors.
- "System" reset - Triggered by external pin assertion, or register write or safety errors. This reset does not reset debug logic.
- PS "System" reset - Triggered by a hardware error or by a register write. This is a PS only reset and PL remains active.
- PS POR reset - Similar to External POR but only for PS
- Full power subsystem (FPS) reset - Triggered by error or register write and used to reset Full Power Domain
- RPU Reset - Triggered by errors or register write, explicitly to reset RPU

See Fabric Reset Enable in Chapter 4. Also for more details about the individual resets, see the *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085) [Ref 1].

Design Flow Steps

This chapter describes customizing and generating the core, constraining the core, and the simulation, synthesis and implementation steps that are specific to this IP core. More detailed information about the standard Vivado® design flows and the IP integrator can be found in the following Vivado Design Suite user guides:

- *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994) [Ref 3]
- *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 4]
- *Vivado Design Suite User Guide: Getting Started* (UG910) [Ref 5]
- *Vivado Design Suite User Guide: Logic Simulation* (UG900) [Ref 6]

Customizing and Generating the Core

This section includes information about using the Vivado Design Suite to customize and generate the core.

If you are customizing and generating the core in the Vivado IP integrator, see the *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994) [Ref 3] for detailed information. The IP integrator might auto-compute certain configuration values when validating or generating the design, as noted in this section. To check whether the values do change, see the description of the parameters in this chapter. To view the parameter value, run the `validate_bd_design` command in the Tcl Console.

You can customize the IP for use in your design by specifying values for the various parameters associated with the IP core in the IP integrator using the following steps:

1. Select the IP from the Vivado IP catalog.
2. Double-click the selected IP, or select the **Customize IP** command from the toolbar or right-click menu.

For details, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 4], and the *Vivado Design Suite User Guide: Getting Started* (UG910). [Ref 5].

The **Zynq UltraScale+ Block Design** page with a block diagram appears in the window (Figure 4-1). Review the contents of the block diagram. The green colored blocks in the diagram are configurable.

TIP: *To open the corresponding configuration page, you can click a green block, or select the page in the Page Navigator at the left side.*

Note: Figures in this chapter are illustrations of the Vivado Integrated Design Environment (IDE). This layout might vary from the current version.

PS Zynq UltraScale+ MPSoC Block Design

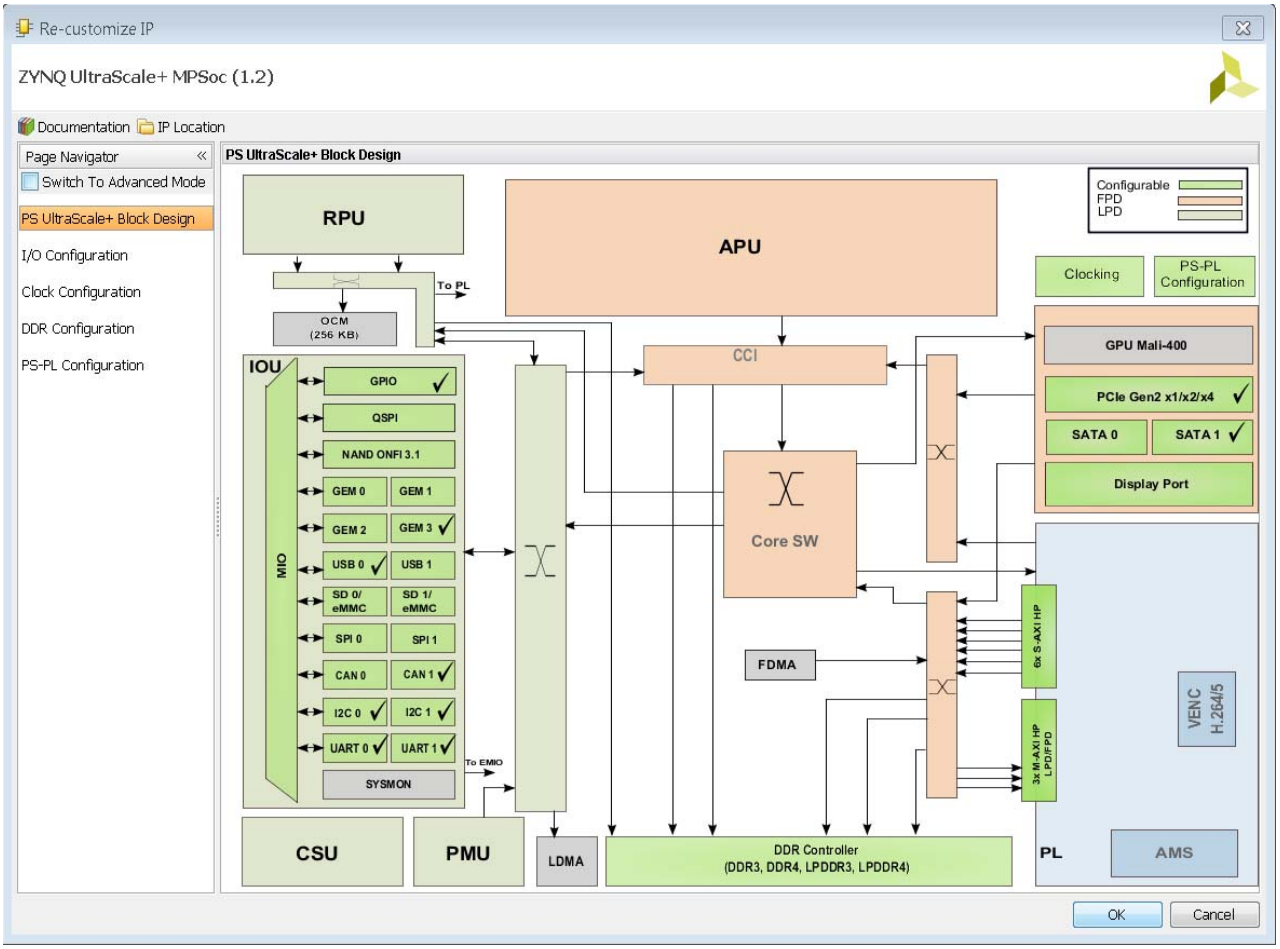


Figure 4-1: **PS UltraScale Block Design Page**

The colors in the PS UltraScale™ Block Design page have the following meanings:

- Light Green color shows Low Power Domain
- Light Orange color shows Full Power Domain
- Dark Green color shows the components that you can configure.

I/O Configuration

This page shows pin assignments for individual signals of an interface of PS components. You can assign attributes for the signals. See the following subsections.

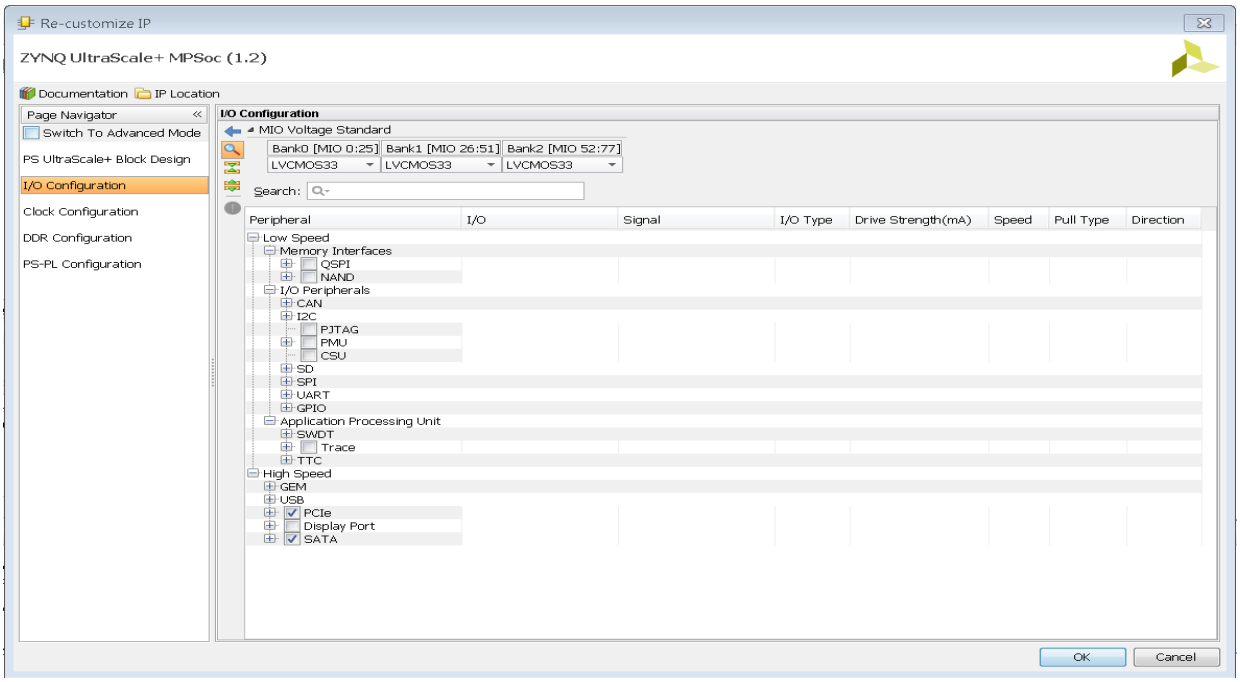


Figure 4-2: **I/O Configuration Page**

MIO Voltage Standard

Each of these I/O pins can be routed through MIOs, EMIOs, or GT Lanes as applicable. Each peripheral pin can be routed through a subset of 78 MIOs as required. Alternatively the same pins from each peripheral can be routed to EMIO signals which brings the signal to PL section of the device for further processing.

For more information on the MIO and EMIO, refer to the Multiplexed I/O, chapter 26 in the *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085) [Ref 1].

MIOs available for peripheral pinouts are divided into three Banks: Bank0 (MIO 0-25), Bank1 (MIO 26-51), and Bank2 (MIO 52-77). Each bank has a common I/O Voltage Standard for all its MIOs and the default value for this is LVCMOS33.

Peripheral

Low Speed

- Memory Interface. These are the static memory controllers present in the PS.
- I/O Peripherals. These are the I/O peripherals present in the PS.
- Application Processing Unit. These are APU specific resources such as watch dog timer, Trace and Triple Timer Counter.

High Speed

Pins from high-speed peripherals, like, PCIe, SATA, Gigabit Ethernet Module (GEM) (in SGMII mode), Display Port and USB 3.0 can be routed to SERDES by selecting the appropriate GT lanes in the I/O column.

I/O Configuration Columns

- **I/O** – Used to configure I/O pins of the respective peripherals.
- **Signal** – Displays information about the signal name driven by the respective I/O pins.
- **I/O Type** – CMOS/Schmitt. Select CMOS or Schmitt as the input I/O voltage type. The Schmitt Voltage type has a higher tolerance to noise than CMOS voltage type.
- **Drive Strength (mA)** – Used to select the drive strength. Possible values are 2, 4, 8, and 12.
- **Speed** – Fast/Slow. Specifies whether the device is fast or slow depending on the slew rate. If the slew rate is 0, the device is fast; else the device is slow.
- **Pull Type** – Used to enable/disable a device along with pull up or pull down. Possible values are **pullup**, **pulldown**, and **disable**.
- **Direction** – The direction can be fixed for certain signals.

Clock Configuration

This page enables you to configure the peripheral clocks, fabric clocks, DDR and CPU clocks. The PCW provides two options, Auto Mode and Manual Mode, to configure the various associated clocks.

Auto Mode

In this mode the tool automatically calculates the M (Multiplier) and D (Divisor) values to ensure that the tool meets the requested frequency to the nearest possible value. User has to enter his desired frequency and the tool does the calculations internally and provides the actual frequency.

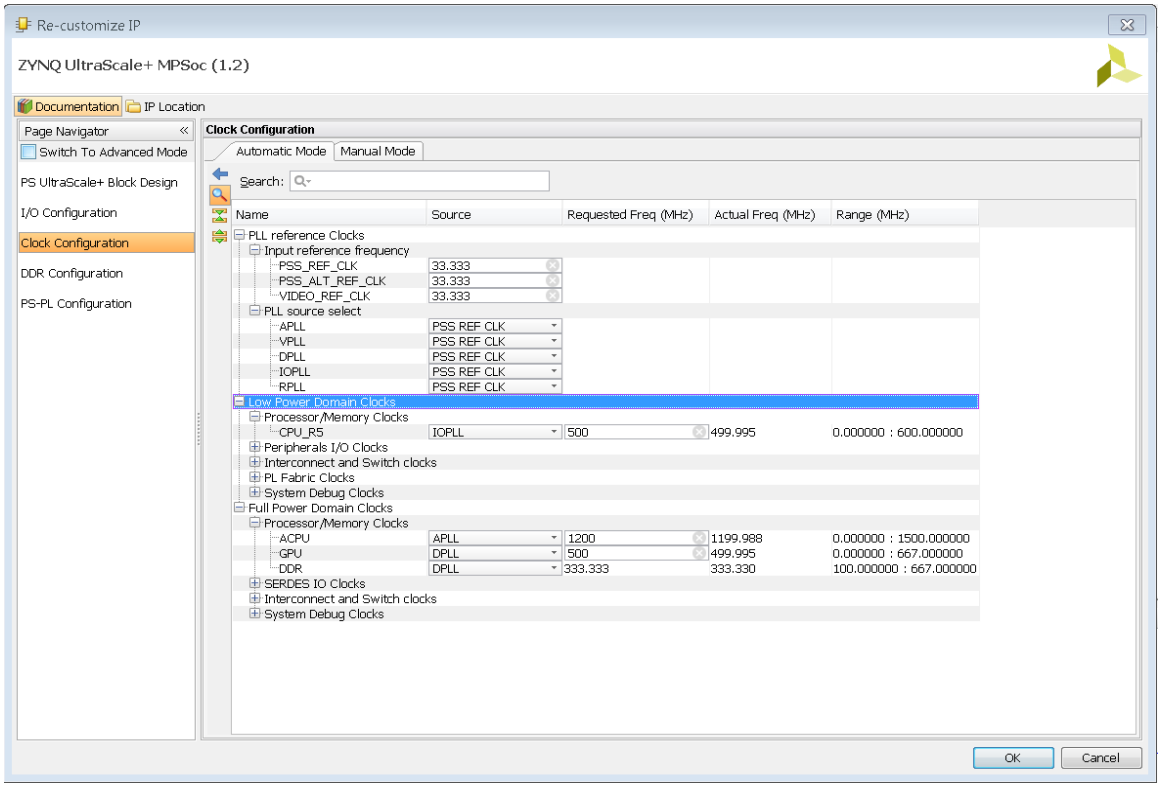


Figure 4-3: **Clock Configuration Page (Automatic Mode Tab)**

PLL Reference Clocks

- **Input reference frequency** – This is the frequency of the clock that is coming from the on-board clock source. There can be three reference clocks: PS_REF_CLK, PSS_ALT_REF_CLK, and VIDEO_REF_CLK.
- **PLL source select** –This is the source clock frequency that is an input to the five PLLs: APLL, DPLL, VPLL, IOPLL, and RPLL. The choice can be any one from PS_REF_CLK, PS_ALT_REF_CLK, and VIDEO_REF_CLK.
- **Interconnect and Switch clocks** – Clocks used by the interconnect and switches internal to the PS.

Low Power Domain Clocks

- **Processor/Memory Clocks** – Clock configuration for the CPU_R5 Processor
- **Interconnect and Switch clocks** – Clocks used by the interconnect and switches internal to the PS.
- **PL Fabric Clocks** – PS generated clock to PL Fabric: PL0, PL1, PL2, and PL3
- **System Debug** – Clocks Clock configuration for debug modules DBG_LPD

Full Power Domain Clocks

- **Processor/Memory Clocks** – Clock configuration for ARM® Cortex™-9 CPU (ACPU), GPU, and DDR
- **SERDES IO Clocks** – Clock configuration for video and high-speed I/Os: PCIE, SATA, DP_VIDEO, DP_AUDIO, and DP_STC
- **Interconnect and Switch clocks** – Clocks used by the interconnect and switches internal to the PS.
- **System Debug Clocks** – Clock configuration for debug modules: DBG_FPD,DBG_TRACE, and DBG_TSTMP
- **GT Lane clocking**
  There are four GT lanes namely GT Lane# (#-> 0 to 3) available to be used for High Speed I/O peripherals. There are four reference clocks for these four GT Lanes. Either the same reference clock can be used as a reference for all these GT lanes or each GT lane can have its own individual reference clock. The selection of the reference clock can be done from the Advanced Configuration tab in PCW while the frequency of these reference clocks can be set from within the **Auto Mode** of the clocking page.

  In the clock configuration page, go to manual mode and tick the **Enable Manual Mode** box. Then you can override the values.

Clock Configuration Columns for Auto Mode

- **Source** – This is the source PLL for the corresponding peripheral.
- **Requested Freq (MHz)** – This is the input frequency given to the corresponding peripheral
- **Actual Freq (MHz)** – This is the actual frequency calculated by the Processor Configuration Wizard. The clocking algorithm works with multiple factors, peripherals, PLLs and priorities; therefore, in certain cases, the actual frequency might be different than the Input Frequency.
- **Range (MHz)** – This is the Minimum/Maximum range of the frequency that the corresponding peripheral can work with

Manual Mode

In this mode, you must configure the M and D values to achieve the desired frequency. Once this mode is enabled; the values requested through auto mode will be overwritten

Note: In order to modify the clock frequencies/divisors in manual mode, the corresponding clock must be enabled in automatic mode.

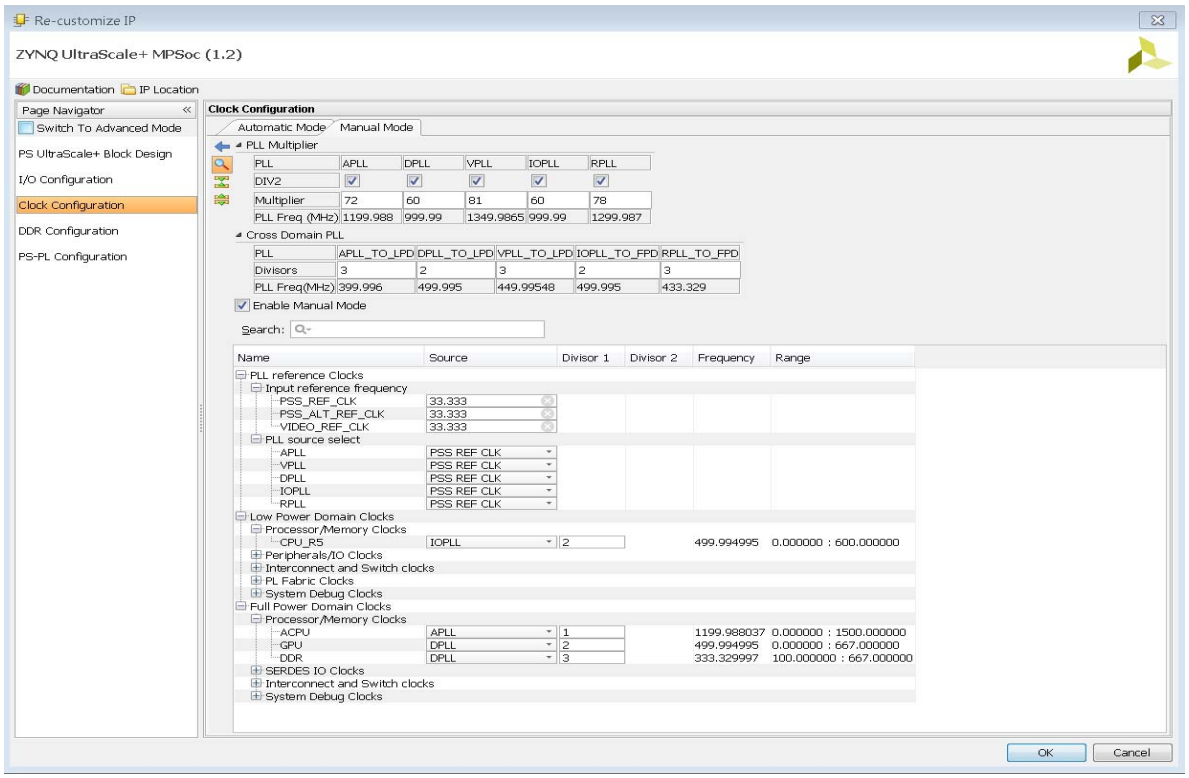


Figure 4-4: **Clock Configuration Page (Manual Mode)**

PLL Multiplier

- **PLL** – One of the five PLLs available in PS: APLL, VPLL, DPLL, IOPLL, and RPLL.
- **DIV2** – This turns on the divide by 2 feature that is inside of the PLL. This does not change the VCO frequency, just the output frequency. The user has got an option to disable the same while in manual mode. However in Auto mode; the calculations are done based on DIV2 as enabled.
- **Multiplier** – Multiplier value for the PLL. The PLL frequency is multiplied by this number to generate the PLL frequency.
- **PLL Freq (MHz)** – PLL frequency after multiplying the PLL frequency with Multiplier.

Cross-Domain PLL

There are five PLLs available in the MPSoC that are spread across the two domains; LPD and FPD. There are three PLLs namely APLL, DPLL and VPLL in the FPD domain while the RPLL and the IOPLL are in the LPD domain. PCW provides an option to make use of the cross domain PLLs to be used to source the cross-over peripheral. This gives additional options to select from a pool of all PLLs.

- **PLL** – One of the five PLLs available in PS: APLL, VPLL, DPLL, IOPLL, RPLL.
- **Divisors** – Divisor value for the cross domain PLL; the PLL frequency is divided by this number to generate the cross domain PLL frequency.
- **PLL Freq (MHz)** – Cross domain PLL frequency after multiplying the PLL frequency with Multiplier.

Enable Manual Mode

PLL Reference Clocks

- **Input Reference frequency** – There are five: PSS_REF_CLK, PSS_ALT_REF_CLK, VIDEO_REFCLK, AUX_REF_CLK, GT_REF_CLK
- **PLL source select** – There are five: APLL, VPLL, DPLL, IOPLL, RPLL

Low Power Domain Clocks

- **Processor/Memory Clocks** – Clock configuration for the CPU_R5 Processor
- **Peripherals/IO Clocks** – Clock configuration for low-speed peripheral devices.
- **Interconnect and Switch clocks** – Clocks used by the interconnect and switches internal to the PS.
- **PL Fabric Clocks** – PS generated clock to PL Fabric: PL0, PL1, PL2, and PL3
- **System Debug Clocks** – Clock configuration for debug modules DBG_LPD

Full Power Domain Clocks

- **Processor/Memory Clocks** – Clock configuration for ARM® Cortex™-9 CPU (ACPU), GPU, and DDR
- **SERDES IO Clock** – Clock configuration for video and high-speed I/Os: PCIE, SATA, DP_VIDEO, DP_AUDIO, and DP_STC
- **Interconnect and Switch clocks** – Clocks used by interconnect and switches internal to the PS.
- **System Debug Clocks** – Clock configuration for debug modules: DBG_FPD,DBG_TRACE, and DBG_TSTMP

Clock Configuration Columns for Auto Mode

- **Source** – This is the source PLL for the corresponding peripheral.
- **Requested Freq (MHz)** – This is the input frequency given to the corresponding peripheral
- **Actual Freq (MHz)** – This is the actual frequency calculated by the Processor Configuration Wizard. The clocking algorithm works with multiple factors, peripherals, PLLs and priorities; therefore, in certain cases, the actual frequency might be different than the Input Frequency.
- **Range (MHz)** – This is the Minimum/Maximum range of the frequency that the corresponding peripheral can work with.

DDR Configuration

The page allows you to set the DDR controller configurations.

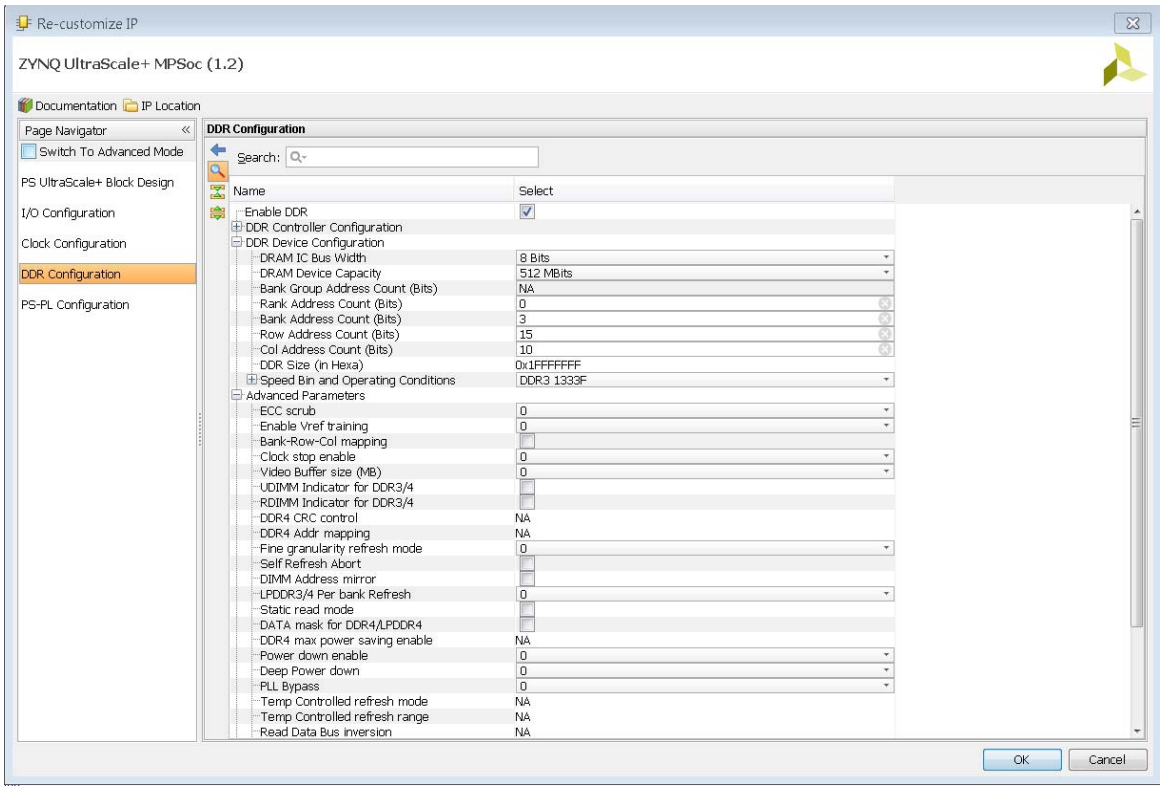


Figure 4-5: **DDR Configuration Page**

DDR Controller Configuration

- **Enable DDR** – Enable DDR controller for Zynq PS.
- **Memory Type** – Type of memory interface. For more details about the individual resets, see the *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085) [Ref 1].
- **Effective DRAM Bus Width** – Data width for DDR interface, not including ECC data width
- **ECC** – Enables Error correction code support. ECC is supported only for an effective data width of 16 bits.
- **Burst Length** – Minimum number of data beats the controller should use when communicating with the DDR component
- **Internal Vref** – Enables internal voltage reference source. Disable to use external Vref pins as Voltage reference.

DDR Device Configuration

- **DRAM IC Bus Width** – Width of individual DRAM components
- **DRAM Device Capacity** – Storage capacity of individual DRAM components
- **Bank Group Address Count (Bits)** – Number of bank address pins
- **Rank Address Count (Bits)** – Number of Rank address pins.
- **Row Address Count (Bits)** – Number of Row address pins
- **Col Address Count (Bits)** – Number of Column address pins
- **DDR Size (in Hex)** – Total DDR Size
- **Speed Bin and Operating Conditions**:
  - **Operating Frequency (MHz)** – Intended operating frequency.
  - **CAS Latency (cycles)** – Column Access strobe latency in memory clock cycles. It refers to the amount of time it takes for data to appear on the pins of the memory module.
  - **CAS Write Latency (cycles)** – Write latency setting in memory clock cycles.
  - **RAS To CAS (cycles)** – Row address to column address delay time. It is the time required between the memory controller asserting a row address strobe (RAS), and then asserting the column address strobe (CAS).
  - **Precharge Time (cycles)** – Precharge time is the number of clock cycles needed to terminate access to an open row of memory and open access to the next row
  - **tRC (ns)** – Row cycle time

- **tRASmin (ns)** – Minimum number of memory clock cycles required between an Active and Precharge command.
- **tFAW (ns)** – Determines the number of activates that can be performed within a certain window of time.

Advanced Parameters

Training/Board Details

- **DRAM Training** – Enable/Disable DRAM training. Default is disable.

DDR Advanced Parameters

- **ECC scrub** – Performs a Read-Modify-Write during a single-bit ECC error.
- **Enable Vref training** – Enable Vref training (DDR4/LPDDR4) when the 'train' flag is set.
- **Bank-Row-Col mapping** – Indicates the mapping between the User Interface address bus and the physical memory.
- **Clock Stop enable** – Disables the output clock whenever it is not required by the SDRAM.
- **DDR4 CRC control** – (DDR4) Enable Write CRC generation
- **DDR4 Addr mapping** – DDR4 address mapping, places BG0 bit in the BL8 transaction location to take advantage of tCCD_L vs tCCD_S; set to 1 in DDR4.
- **Fine granularity refresh mode (DDR4)** – Sends more frequent refreshes.
- **DIMM Address mirror** – Compensates for swapped address bits on the odd rank.
- **Data mask for DDR4/LPDDR4** – Use DM signals.
- **DDR4 max power saving enable (DDR4)**– Puts the SDRAM into maximum power saving mode when the transaction store is empty.
- **Deep Power down** – Parameter only for LPDDR3. Puts the SDRAM into deep power-down mode when the transaction store is empty.
- **PLL Bypass** – Based on the bit selection; it bypasses the source PLL.
- **Temp controlled refresh mode** – Enables Temperature Controlled Refresh mode.
- **Temp controlled refresh range** – Maximum operating temperature range
- **Read Data Bus Inversion** – Performs Read DBI.
- **Write Data bus inversion** – Performs Write DBI.
- **PHy performs data bus inversion** – Performs DBI operations in the PHY rather than the controller.
- **Address Copy enable** – Duplicates address on unused address signals.

- **Parity** – Enables generation and detection of command/address parity errors.
- **CAL mode** – DDR4 CAL mode (CS_n to Command Address Latency)
- **Low power auto self-refresh** – Put the SDRAM into Self Refresh after a programmable number of idle cycles.

PS-PL Configuration

This page allows you to configure PS-PL interfaces including AXI, HP, and ACP bus interfaces.

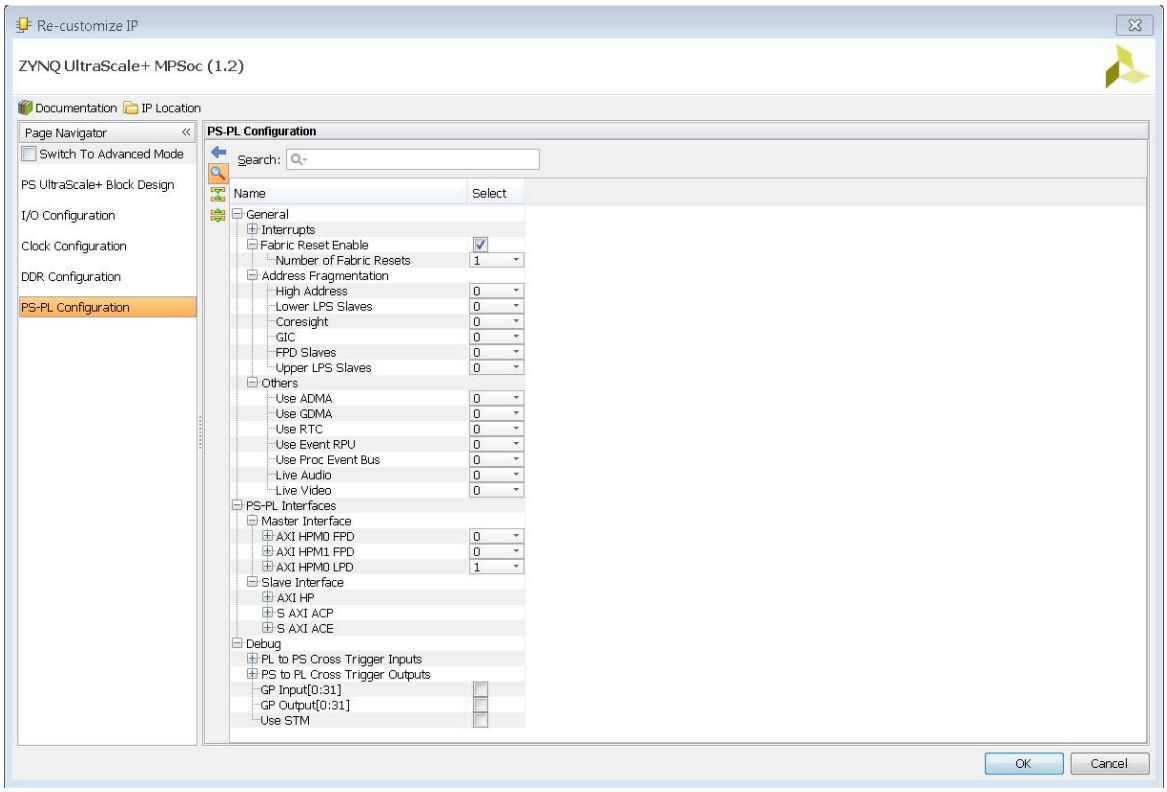


Figure 4-6: **PS-PL Configuration Page**

General

Interrupts

Fabric Reset Enable

Fabric Resets can be enabled from **PS - PL Configuration > General > Fabric Reset Enable**. Up to four PL Reset signals can be enabled with the default being one reset signal enabled.

There are a total of four PS-PL resets that are available. These four PL resets that are user selectable from within the PCW use the available last four out of the 96 EMIOs. Based on their selection from 0-4, the number of EMIO is reduced from 96 to 92 which should be taken into account. The selection for the fabric reset can be done from the **General** node available in the PS-PL configuration page.

The corresponding registers required to toggle the EMIOs for realizing the PL resets is taken care of by the PCW through output files that are generated as a part of output products.

Address Fragmentation

With the availability of several peripherals within PS, PCW provides an organized way to access these peripherals. The **Address Fragmentation** allows you to expand the peripherals based on the address space in which they are assigned within the Zynq Ultra Scale+ MPSoC. Lower LPD slaves, Upper LPD slaves, FPD slaves and others are few of the available choices. Based on the selection, only the selected segments will be shown up in the address editor in Vivado along with the addresses to which they will be mapped to the PL- master.

This way only the list of selected peripherals will appear in the address editor. This can be used where the requirement is to have more address space available for the PL components, rather than a single address block assigned to Zynq Ultra Scale+ MPSoC addressable components.

Notes:

1. High DDR segment is not enabled if the DDR size is less than or equal to 2GB.
2. When the DDR size is greater than 2GB, the High DDR segment can be used to have DDR addressed in a higher address space, this is limited to 4GB of DDR size.
3. You must have a 64-bit master in the PL in order to access higher address space above 4GB.

For more information, see the Zynq UltraScale All Programmable MPSoC Technical Reference Manual (UG1085) [Ref 1].

Others

- Use ADMA – DMA in Low power domain
- Use GDMA – DMA in Full power domain
- USE RTC – Real Time clock
- Use Event RPU and Use Proc Event Bus – The processor includes logic to detect various events that can occur, for example, a cache miss. These events provide useful information about the behavior of the processor for use when debugging or profiling code. The events are made visible on an output event bus and can be counted using registers in the performance monitoring unit.
- Live Audio and Live Video – The DisplayPort controller supports live audio and video channels from the PL. These audio and video streams interface to the DisplayPort controller and provide live audio and video overlays from the PL.

PS-PL Interfaces

Master Interface

- AXI HPM0 FPD – High performance master 0 in full power domain
- AXI HPM1 FPD – High performance master 1 in full power domain
- AXI HPM0 LPD – High performance master 0 in low power domain.

Each interface supports 32, 64, 1nd 128 data widths.

Slave Interface

- AXI HP and sub options – There are two (AXI HPC0 FPD, AXI HPC1 FPD) high performance AXI I/O coherent master interfaces in full-power domain; four (AXI HP0 FPD, AXI HP1 FPD, AXI HP2 FPD, AXI HP3 FPD) high performance slave AXI interfaces in full-power domain; one (AXI LPD) AXI interface in low-power domain.

  Each interface supports 32, 64, 1nd 128 data widths.
- S AXI ACP – There is one Accelerator Coherency Port that can be connected to a DMA engine or a non-cached coherent master.
- S AXI ACE – There is one AXI Coherency Extension slave.

Debug

The debug feature enables configuration of cross trigger signals. This provides debug capability for accessing the PS debug structure allowing integrated test and debug on both PS and PL simultaneously.

Fabric Trigger Macrocell (FTM) For Programmable Logic Test And Debug

It is based on the ARM® CoreSight® architecture. The FTM receives trace data from the PL and formats it into trace packets to be combined with the trace packets from other trace source components such as PTM and Instrumentation Trace Macrocell (ITM). With this capability, PL events can easily be traced simultaneously with PS events.

The FTM also supports cross-triggering between the PS and PL, except for the trace dumping feature. In addition, the FTM provides general-purpose debug signals between the PS and PL.

This block provides:

- General purpose I/Os, 32 bits to the PL and 32 bits from the PL. These are accessed through reads and writes to registers.

- Trigger signals, four pairs to the PL and four pairs from the PL. Each pair consists of a trigger signal and an acknowledge signal, and follows the ARM standard CTI handshake protocol

Options

- PL to PS Cross Trigger Inputs – Trigger signals, four pairs from the PL. Each pair consists of a trigger signal and an acknowledge signal, and follows the ARM® standard CTI handshake protocol.
- PL to PS Cross Trigger Outputs and sub options – Trigger signals, four pairs to the PL. Each pair consists of a trigger signal and an acknowledge signal, and follows the ARM standard CTI handshake protocol.
- GP Input[0:31] GP Output[0:31] – General purpose I/Os, 32 bits to the PL and 32 bits from the PL. These are accessed through reads and writes to registers.

Advanced Configuration

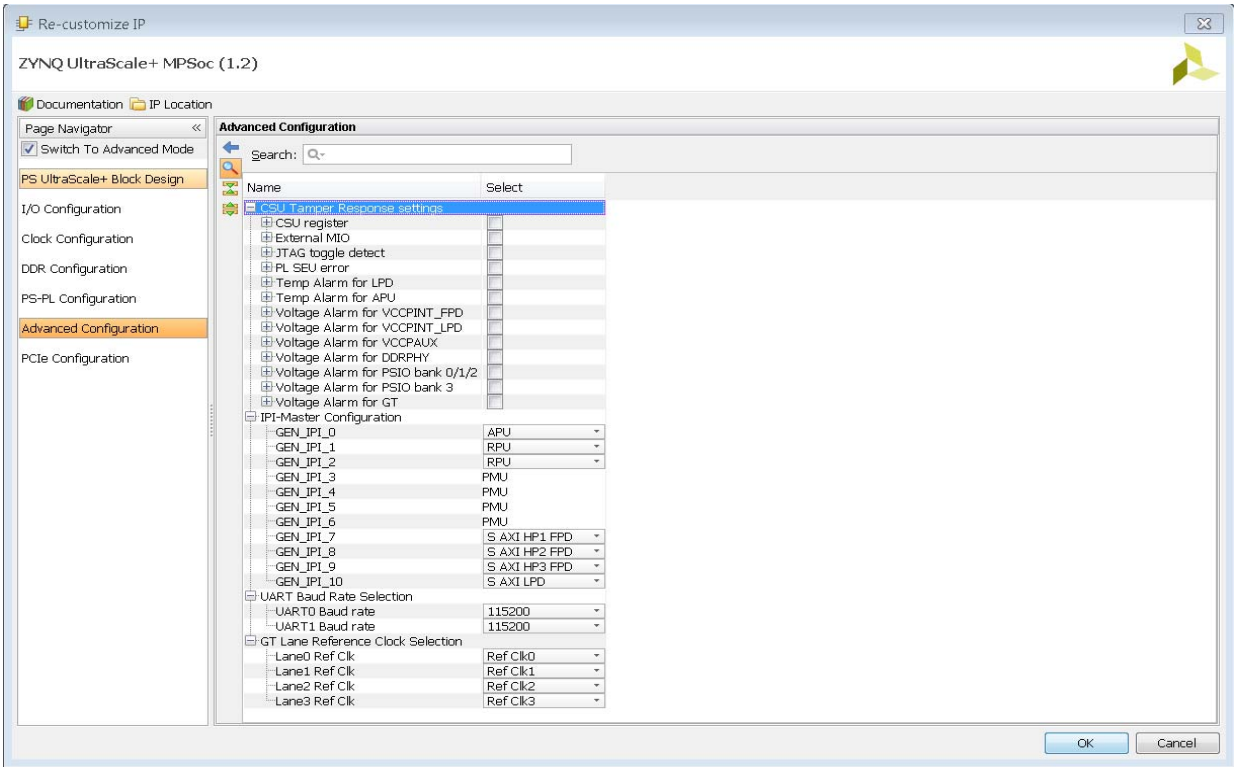


Figure 4-7: **Advanced Configuration**

CSU and Tamper Response Settings

CSU is responsible for loading the processing system (PS) first-stage boot loader (FSBL) code into the on-chip RAM (OCM) in both secure and non-secure boot modes. You can select, through the boot header, to execute the FSBL on the Cortex®-R5 or the Cortex-A53 processor. After FSBL execution starts, the CSU enters the post-configuration stage, which is responsible for system tamper response.

The CSU can be configured to have secure lock down, system reset, and system interrupt for some of the errors like PL single event upset (SEU) error, Temperature alarm, voltage alarm, etc. These options are available under **CSU Tamper Response settings** on the Advanced Configuration page.

Interrupts

One external tamper interrupt is mapped to CSU through MIO. There are three interrupts from CSU (PS) to PL as CSU WDT Interrupt, CSU DMA Interrupt, and CSU interrupt.

The CSU Interrupt is used to indicate that something in the CSU logic has caused an interrupt. The CSU interrupt status register holds the interrupt bits for all of the CSU logic except for the DMA. The following values can cause an interrupt in the CSU:

For more information, see the Zynq UltraScale All Programmable MPSoC Technical Reference Manual (UG1085) [Ref 1].

- AES done – Bit to notify Advanced Encryption done.
- PL INIT complete – Bit to indicate PL initialization is complete.
- AES error – Bit to indicate Advanced Encryption error.
- RSA done – Bit to Indicate RSA Encryption done.
- PL POR_B – Bit to indicate PL power on reset status.
- TMR fatal error – Bit to indicate Triple-Mode Redundant (TMR) fatal error
- SHA done – Bit to indicate Secure Hash Algorithm Encryption done.
- PL SEU error flag – Bit to indicate Single Even Upset error.
- APB SLVERR – An error bit to indicate the failure of a transfer.
- PL CFG done – Status bit to indicate PL configuration complete.
- PCAP FIFO overflow – Status bit to indicate Processor Configuration Access Port FIFO overflow.
- CSU RAM ECC error – Bit to indicate CSU RAM ECC error.

The CSU_DMA_IRQ will alert the system that the DMA has generated an interrupt. The CSU WDT Interrupt is from the CSU watch dog timer interrupt.

Options

For more information, see the Zynq UltraScale All Programmable MPSoC Technical Reference Manual (UG1085) [Ref 1].

- CSU Register – Setting bits in this register causes the CSU ROM to issue a system interrupt when the tamper event occurs.
- External MIO – Observation of MIOs that causes the CSU ROM to issue a system interrupt when the tamper event occurs.
- JTAG toggle detect – Bit to identify the change in the JTAG mode.
- PLU SEU error – Bit to indicate Single Even Upset error.
- Temp Alarm for LPD – Temperature alarm for Low Power/RPU domain.
- Temp Alarm for APU – Temperature alarm for APU/ Full power domain.
- Voltage Alarm for VCCPINT_FPD – Power rail removal alarm when VCCPINT_FPD is removed.
- Voltage Alarm for VCCPINT_LPD – Power rail removal alarm when VCCPINT_LPD is removed.
- Voltage Alarm for VCCPAUX – Power rail removal alarm when VCCPAUX is removed.
- Voltage Alarm for DDRPHY – Reference voltage observation signal for DDR PHY.
- Voltage Alarm for PSIO bank 0/1/2 – Reference voltage observation signal for PSIO bank 0/1/2.
- Voltage Alarm for PSIO bank 3 – Reference voltage observation signal for PSIO bank 3.
- Voltage Alarm for GT – Reference voltage observation signal for Gigabit transceivers.

IPI Master Slave Configuration

The Inter Processor Interrupt Block provides the ability for any processing unit to interrupt another processing unit by performing a register write.

There are 11 IPI channels (GEN_IPI_0 through GEN_IPI_10), out of which four channels (Channel 3, 4, 5, 6) are dedicated to PMU. The rest of the channels can be assigned to APU, RPU, and PL. With this Master assignment to each IPI channel protects corresponding channel using XPPU from unmapped masters.

Each IPI channel provides the registers to trigger the interrupts to any destination. The XPPU only allows the master that is associated with channel to access those registers. In addition to the registers, IPI channels are provided with the payload buffers.

XPPU only allows the master that is associated with buffers to access those buffers.

UART Baud Rate Selection

- UART0 Baud Rate – Specifies the UART baud rate for the UART0.
- UART1 Baud Rate – Specifies the UART baud rate for the UART1.

GT Lane Reference Clock Selection

- Lane0 Ref Clk – Reference clock for GT Lane0
- Lane1 Ref Clk – Reference clock for GT Lane1
- Lane2 Ref Clk – Reference clock for GT Lane2
- Lane3 Ref Clk – Reference clock for GT Lane3

User Parameters

The core can be parameterized for individual applications. Parameters related to enabling interfaces or functions reflect the state of the Zynq UltraScale+ MPSoC configuration. The device configuration custom Vivado Integrated Design Environment (IDE) is available in the Vivado IP integrator and should be used to update the parameters mentioned in Table C-1.

These parameter are updated in the IP integrator. Ports related to specific peripherals are either valid or invalid. Invalid ports are not visible. The IP integrator database uses these parameters to initialize associated PS registers in the `ps_init.tcl` or First Stage Boot Loader (FSBL). The FSBL enables you to configure the design as needed, including the PS and PL. By default, the JTAG interface is enabled to give you access to the PS and PL for test and debug purposes.

In batch mode, the core can be configured using the `set_property` Tcl Console command.

Table C-1 in Appendix C, User Parameters shows the relationship between the fields in the Vivado IDE and the User Parameters (which can be viewed in the Tcl Console).

Output Generation

For details about common core output files, see "Generating IP Output Products" in the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 4].

The Vivado design tool exports the Hardware Platform Specification for your design to the Software Development Kit (SDK). The following five files are exported to SDK:

- The `system.hdf` file opens by default when SDK launches. The address map of your system read from this file is shown by default in the SDK window.
- The `psu_init.tcl`, `psu_init.c` and `psu_init.h` files contain the initialization code for the Zynq UltraScale MPSoC processing system and initialization settings for DDR, clocks, plls, and MIOs. SDK uses these settings when initializing the processing system so that applications can be run on top of the processing system.
  - `psu_init.tcl`: This Zynq UltraScale+ MPSoC Processor System initialization with the Tcl file is used for the device initialization Xilinx Microprocessor Debugger (XMD) flow.
  - `psu_init.c`: Generated by the PS Configuration Wizard (PCW), this header file for the first stage boot loader (FSBL) contains proc of a `psu_init()` and the return values. The FSBL uses only this file, and it calls the `psu_init()` functions, and checks return values.
  - `psu_init.h`: Generated by the PCW, this file implements the `psu_init()`. This file also contains some testing code. This testing code enhances the testing performed by the PCW.

The PS Configuration Wizard tool generates output code that is a table of words, which is interpreted by a small engine, looping through the table and performing the actions.

All the `EMIT_*` are #defines, which adds 1 to 4 words to the `ps_init_data` array.

The supporting `.c` and `.h` files (described earlier) are also produced by the PCW.

The Zynq UltraScale MPSoC Processing System core overwrites all files when regenerated.

Constraining the Core

TO BE SUPPLIED

Required Constraints

This section is not applicable for this core.

Device, Package, and Speed Grade Selections

This section is not applicable for this core.

Clock Frequencies

This section is not applicable for this core.

Clock Management

This section is not applicable for this core.

Clock Placement

This section is not applicable for this core.

Banking

This section is not applicable for this core.

Transceiver Placement

This section is not applicable for this core.

I/O Standard and Placement

This section is not applicable for this core.

Simulation

AXI BFM for Zynq UltraScale+ MPSoC is not yet supported. This section is not applicable for this core.

Synthesis and Implementation

For details about synthesis and implementation, see "Synthesizing IP" and "Implementing IP" in the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 4].

Example Design

This chapter gives an example of how to set up a DDR Configuration.

The PS Configuration Wizard (PCW), provides you with the means to configure the DDR controller for your specific DDR Memory Part in an easy and intuitive manner. The following procedure demonstrates how to build a complete DDR configuration using the PCW and taking as an example Micron's MT41K1G8SN-125:A.

1. To access the DDR configuration, select the **DDR Configuration** from the PCW.

   Looking at the DDR Configuration page, notice that it is split into four sections, these are:

   - DDR Controller Configuration
   - DDR Device Configuration (which is further subdivided to Speed Bin and Operating Conditions
   - Advanced Parameters
   - Training/Board Details

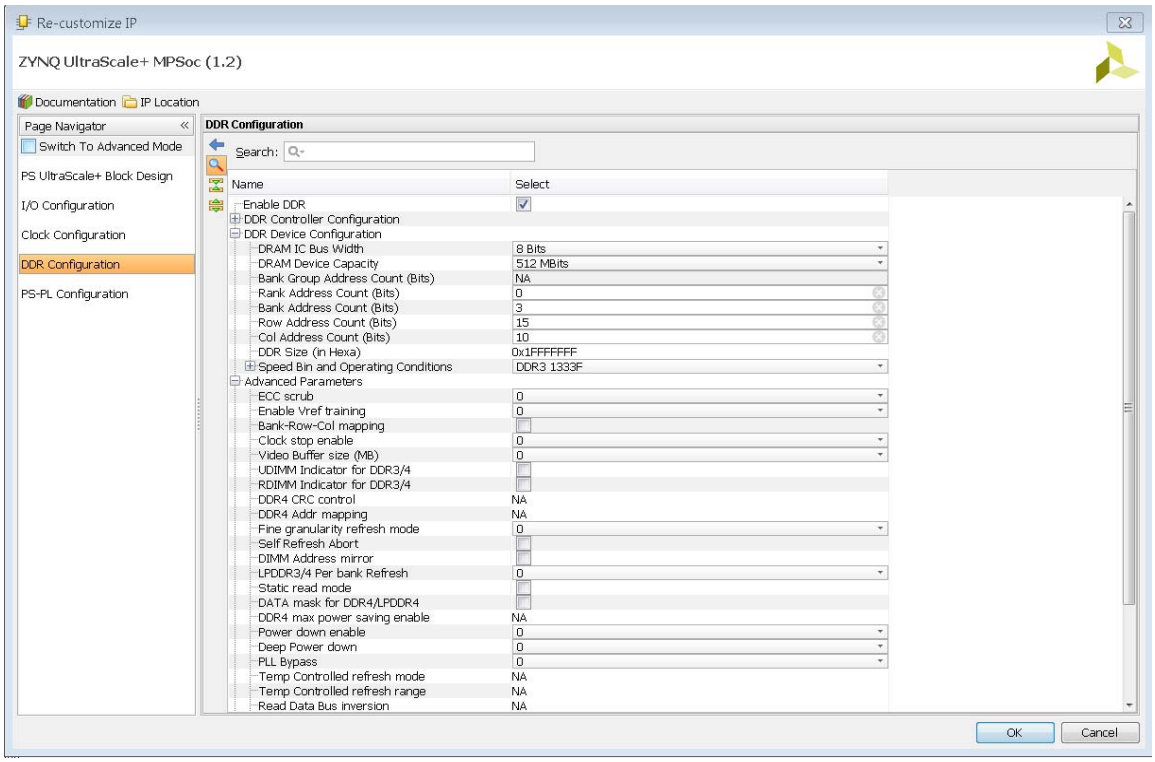
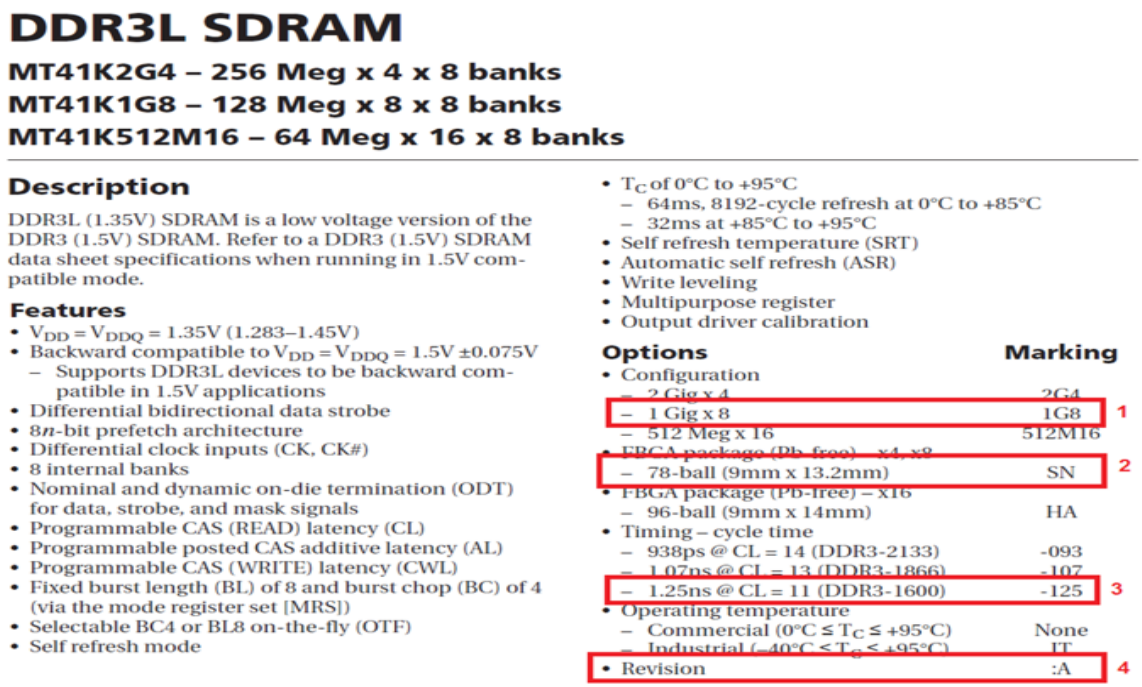


Figure 5-1: **DDR Configuration**

2. From the DDR Configuration page, create a DDR Configuration using as an example the Micron MT41K1G8SN-125:A, which denotes a DDR3 device. For this example the focus is on the **DDR Controller Configuration** and **DDR Device Configuration**.

   Note: The Micron data sheet MT41K1G8SN-125:A content in Figure 5-2 through Figure 5-6 is provided with permission of Micron Technology Inc. [Ref 9]



DDR3L SDRAM

MT41K2G4 – 256 Meg x 4 x 8 banks
MT41K1G8 – 128 Meg x 8 x 8 banks
MT41K512M16 – 64 Meg x 16 x 8 banks

Description

DDR3L (1.35V) SDRAM is a low voltage version of the DDR3 (1.5V) SDRAM. Refer to a DDR3 (1.5V) SDRAM data sheet specifications when running in 1.5V compatible mode.

Features

- $V_{DD} = V_{DDQ}$ = 1.35V (1.283–1.45V)
- Backward compatible to $V_{DD} = V_{DDQ}$ = 1.5V ±0.075V
  - Supports DDR3L devices to be backward compatible in 1.5V applications
- Differential bidirectional data strobe
- 8*n*-bit prefetch architecture
- Differential clock inputs (CK, CK#)
- 8 internal banks
- Nominal and dynamic on-die termination (ODT) for data, strobe, and mask signals
- Programmable CAS (READ) latency (CL)
- Programmable posted CAS additive latency (AL)
- Programmable CAS (WRITE) latency (CWL)
- Fixed burst length (BL) of 8 and burst chop (BC) of 4 (via the mode register set [MRS])
- Selectable BC4 or BL8 on-the-fly (OTF)
- Self refresh mode
- T_C of 0°C to +95°C
  - 64ms, 8192-cycle refresh at 0°C to +85°C
  - 32ms at +85°C to +95°C
- Self refresh temperature (SRT)
- Automatic self refresh (ASR)
- Write leveling
- Multipurpose register
- Output driver calibration

Options	Marking	
• Configuration		
– 2 Gig x 4	2G4	
– 1 Gig x 8	1G8	1
– 512 Meg x 16	512M16	
• FBGA package (Pb-free) – x4, x8		
– 78-ball (9mm x 13.2mm)	SN	2
• FBGA package (Pb-free) – x16		
– 96-ball (9mm x 14mm)	HA	
• Timing – cycle time		
– 938ps @ CL = 14 (DDR3-2133)	-093	
– 1.07ns @ CL = 13 (DDR3-1866)	-107	
– 1.25ns @ CL = 11 (DDR3-1600)	-125	3
• Operating temperature		
– Commercial (0°C ≤ T_C ≤ +95°C)	None	
– Industrial (–40°C ≤ T_C ≤ +95°C)	IT	
• Revision	:A	4

Figure 5-2: **Micron Data Sheet**

Note: Content of Figure 5-2 used with permission by Micron Technology, Inc.
© 2010/09/04 Micron Technology, Inc., All Rights Reserved

3. Examine the first page of the data sheet in Figure 5-2 and in particular the device name. You can identify the information that is required in order to fill in the **DDR Controller Configuration** and **DDR Device Configuration** sections of the DDR Configuration Page.
   - The Device Part name provides a lot of information, for instance, **1G8** is the capacity of the device. In this case it is a 1 Gigabit Device by 8, which makes this an 8 Gigabit Device as shown as the first red rectangle Figure 5-2. There is a more in-depth calculation in the next steps.
   - The Device Part name also gives information as to the speed grade of the device. In this case it is designated as -125 as in 1.25 ns which is the maximum clock period in nanoseconds in this case and a CAS latency of 11 cycles for a DDR3-1600 Speed Bin as shown as the third red rectangle in Figure 5-2. You will see a more in-depth calculation in the next steps.
   - Using as an example the MT41K1G8SN-125:A device translates to the following.
     - Capacity = 1 GBits x 8 = 8 GBits
     - Speed Grade = -125
       1.25 ns @CL =11(DDR3-1600)
       1.25 ns clock cycle == operating frequency of 800 MHz

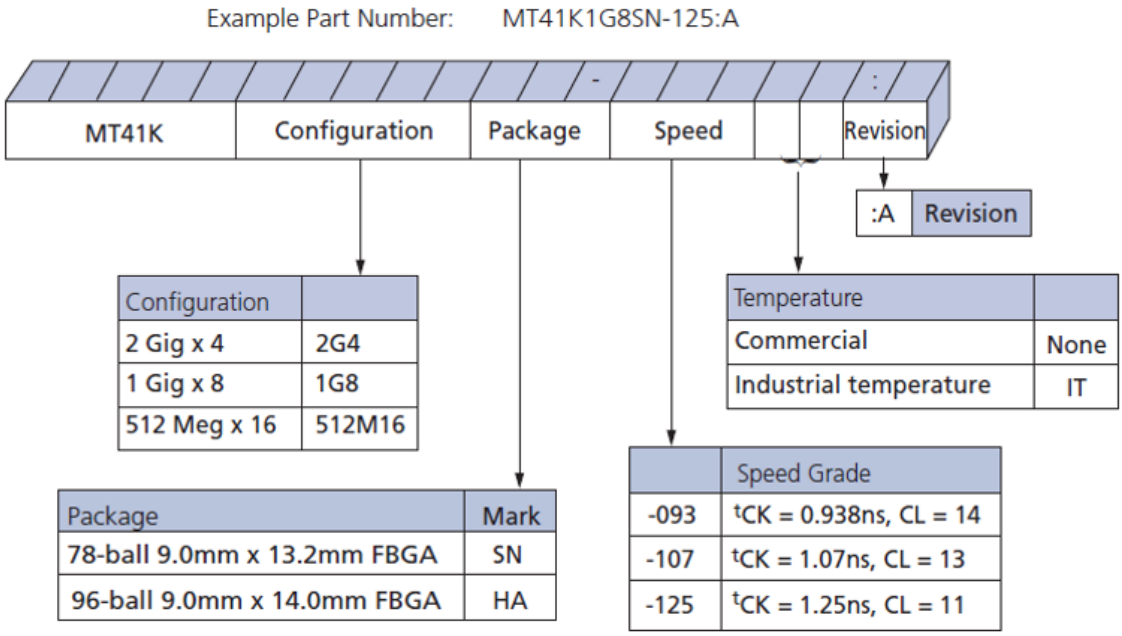


Figure 5-3: **DDR Example Part Number**

Note: Content of Figure 5-3 used with permission by Micron Technology, Inc.
© 2010/09/04 Micron Technology, Inc., All Rights Reserved

- The Micron data sheet in Figure 5-3 shows an example part number and how to identify specific information of interest. For MT41K1G8SN-125:A:
  - Configuration is Row 2 (1 Gig x 8, 1G8)
  - Speed Grade is Row 3 (-125, ^{t}CLK =1.25ns, CL = 11)
  - Temperature is Row 2 (Industrial temperature, IT)

4. Examine the following figure. It is important to understand the addressing scheme.

Table 2: Addressing

Parameter	2 Gig x 4	1 Gig x 8	512 Meg x 16
Configuration	256 Meg x 4 x 8 banks	128 Meg x 8 x 8 banks	64 Meg x 16 x 8 banks
Refresh count	8K	8K	8K
Row address	64K (A[15:0])	64K (A[15:0])	64K (A[15:0])
Bank address	8 (BA[2:0])	8 (BA[2:0])	8 (BA[2:0])
Column address	4K (A[13,11, 9:0])	2K (A[11,9:0])	1K (A[9:0])
Page size	2KB	2KB	2KB

Figure 5-4: **Addressing**

Note: Content of Figure 5-4 used with permission by Micron Technology, Inc.
© 2010/09/04 Micron Technology, Inc., All Rights Reserved

The device capacity is expressed in bits. In this case the capacity is based on the addressable range of the Row, Column and Banks.

Device Capacity = (Row Addressable Range x Column Addressable Range x Bank Addressable Range) x Arrangement

For MT41K1G8SN:A, Look at the second column of the Addressing table designated as 1 Gig x 8, this provides the following values:

- Row Addressable Range = A[15:0] = 2^{16}
- Column Addressable Range = A[11, 9:0] = 2^{11}
- Bank Addressable Range = BA[2:0] = 2^{3}.
- Arrangement = 8 (i.e. 1Gig x 8)

With these values and the Device Capacity equation, gives the following:

- Device Capacity = 2^{16} x 2^{11} x 2^{3} x 8 = 8589934592 = 0x200000000 = 8Gbits

5. Examine the Speed Bin and Operating Conditions in Figure 5-5.

Table 41: DDR3L-1600 Speed Bins

DDR3L-1600 Speed Bin			-125[1]			
CL-tRCD-tRP			11-11-11			
Parameter		Symbol	Min	Max	Unit	Notes
Internal READ command to first data		tAA	13.75	–	ns	
ACTIVATE to internal READ or WRITE delay time		tRCD	13.75	–	ns	
PRECHARGE command period		tRP	13.75	–	ns	
ACTIVATE-to-ACTIVATE or REFRESH command period		tRC	48.75	–	ns	
ACTIVATE-to-PRECHARGE command period		tRAS	35	9 x tREFI	ns	2
CL = 5	CWL = 5	tCK (AVG)	3.0	3.3	ns	3
	CWL = 6, 7, 8	tCK (AVG)	Reserved		ns	4
CL = 6	CWL = 5	tCK (AVG)	2.5	3.3	ns	3
	CWL = 6	tCK (AVG)	Reserved		ns	4
	CWL = 7, 8	tCK (AVG)	Reserved		ns	4
CL = 7	CWL = 5	tCK (AVG)	Reserved		ns	4
	CWL = 6	tCK (AVG)	1.875	<2.5	ns	3
	CWL = 7	tCK (AVG)	Reserved		ns	4
	CWL = 8	tCK (AVG)	Reserved		ns	4
CL = 8	CWL = 5	tCK (AVG)	Reserved		ns	4
	CWL = 6	tCK (AVG)	1.875	<2.5	ns	3
	CWL = 7	tCK (AVG)	Reserved		ns	4
	CWL = 8	tCK (AVG)	Reserved		ns	4
CL = 9	CWL = 5, 6	tCK (AVG)	Reserved		ns	4
	CWL = 7	tCK (AVG)	1.5	<1.875	ns	3
	CWL = 8	tCK (AVG)	Reserved		ns	4
CL = 10	CWL = 5, 6	tCK (AVG)	Reserved		ns	4
	CWL = 7	tCK (AVG)	1.5	<1.875	ns	3
	CWL = 8	tCK (AVG)	Reserved		ns	4
CL = 11	CWL = 5, 6, 7	tCK (AVG)	Reserved		ns	4
	CWL = 8	tCK (AVG)	1.25	<1.5	ns	3
Supported CL settings			5, 6, 7, 8, 9, 10, 11		CK	
Supported CWL settings			5, 6, 7, 8		CK	

Notes:
1. The -125 speed grade is backward compatible with 1333, CL = 9 (-15E) and 1066, CL = 7 (-187E).
2. tREFI depends on T_{OPER}.
3. The CL and CWL settings result in tCK requirements. When making a selection of tCK, both CL and CWL requirement settings need to be fulfilled.
4. Reserved settings are not allowed.

Figure 5-5: **DDR3L-1600 Speed Bins**

Note: Content of Figure 5-5 used with permission by Micron Technology, Inc.
© 2010/09/04 Micron Technology, Inc., All Rights Reserved

Table 1: Key Timing Parameters

Speed Grade	Data Rate (MT/s)	Target tRCD-tRP-CL	tRCD (ns)	tRP (ns)	CL (ns)
-093[1, 2]	2133	14-14-14	13.09	13.09	13.09
-107[1]	1866	13-13-13	13.91	13.91	13.91
-125	1600	11-11-11	13.75	13.75	13.75

Notes:
1. Backward compatible to 1600, CL = 11 (-125).
2. Backward compatible to 1866, CL = 13 (-107).

Figure 5-6: **DDR3L-1600 Speed Bins**

Note: Content of Figure 5-6 used with permission by Micron Technology, Inc.
© 2010/09/04 Micron Technology, Inc., All Rights Reserved

This following information can be derived by looking at both Table 41 in Figure 5-5 and Table 1 in Figure 5-6.

- The device supports 800 MHz (speed grade -125 [1/1.25ns]) operating frequency and because you are accessing a Double Data Rated (DDR) device the maximum transfer is 1600 Million Transfers per second. See Table 41. Row 1.
- Cas Latency (cycles) = Looking at Table 1 – 3rd Row, 3rd Column – Target tRCD – tRP - CL) CL = 11 cycles
- Cas Write Latency (CWL) == Using CL = 11 and looking at Table 41 we can determine that CLW is set at 8 cycles.
- RAS to DAS Delay (cycles) == tRCD/clock cycle = 13.75 ns/1.25 ns = 11 cycles
- tRC = 48.75ns
- tRASmin = 35ns

6. With that information you can now complete the DDR Configuration page. Expand the **DDR Controller Configuration** and **DDR Device Configuration** sections.

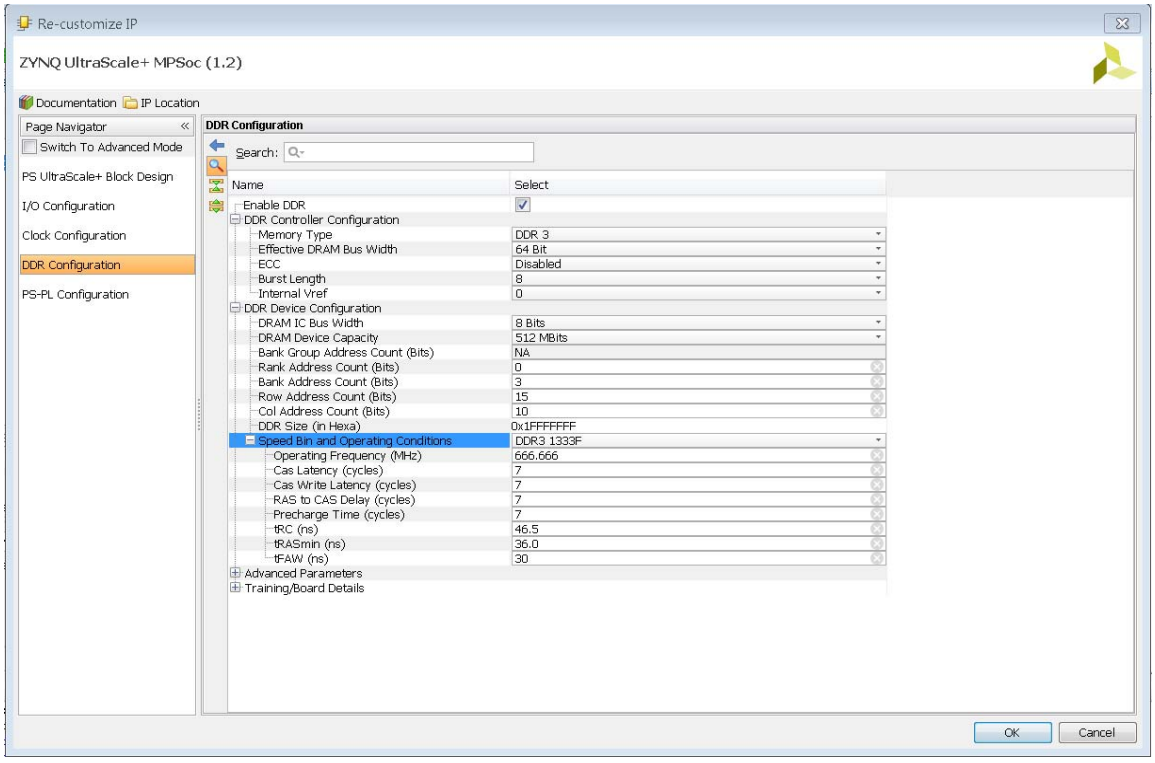


Figure 5-7: **DDR Configuration**

7. Change the **Speed Bin and Operating Conditions**. As stated previously this is a DDR3-1600 Device. Click and select **DDR3 1600K** from the drop down list.

   Notice that PCW has auto-populated a number of fields such as:

   - **Operating Frequency**: In the DDR Controller Configuration to keep the settings in sync.
   - **CAS Latency**: Changed to 11 cycles
   - **CAS Write Latency**: Changed to 8 cycles
   - **RAS to CAS Delay**: Changed to 11 cycles
   - **Precharge Time**: Changed to 11 cycles.
   - **tRC**: Set to 48.75 nanoseconds
   - **tRASmin**: Set to 35 nanoseconds
   - **tFAW**: Set to 30 nanoseconds

   Even though these settings have been auto calculated you are still able to further fine tune them for your own specific part. Looking back at the settings that were calculated when reviewing the DDR from the Micron spreadsheet, notice that the values match.

8. Continue by reviewing the rest of the settings from the previous calculations. Looking at the **DRAM IC Bus Width**, select **8** as a 1G8 memory which implies a "by 8" arrangement as shown.
9. For **DRAM Device Capacity**, based on the previous calculations, select **8192 MBits** which is equal to 8 Gigabits as shown.
10. For **Bank Address Count (bits)** the Bank Addressable Range was 2 to the power of 3, therefore, keep **3** as the bits of **Bank Address Count (bits)**.
11. For the **Row Address Count (bits)** the Row Addressable Range was 2 to the power of 16, therefore, keep **16** as the bits for **Row Address Count (bits)**.
12. For the **Column Address Count (bits)** it was stated that the Column Addressable Range was 2 to the power of 11, therefore, select **11 bits** for **Column Address Count (bits)**.
13. Having concluded the calculations, click **OK** and then **Save the Project**.

Migrating and Upgrading

For changes to the Zynq UltraScale+ MPSoC core from Version 1.1 to Version 1.2, refer to the change log.

Port Descriptions

The signals for the design are listed in the following tables.

Table B-1: **CAN0**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
can0_phy_tx	O	CAN bus transmit signal to first CAN physical-side interface (PHY)
can0_phy_rx	I	CAN bus receive signal from first CAN PHY

Table B-2: **CAN1**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
can1_phy_tx	O	CAN bus transmit signal to second CAN PHY
can1_phy_rx	I	CAN bus receive signal from second CAN PHY

Table B-3: **Event IO**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
pl_ps_eventi	I	Causes one or both CPUs to wake up from a wait for event (WFE) state.
ps_pl_evento	O	Asserted when one of the CPUs has executed the Send EVENT (SEV) instruction
ps_pl_standbywfe	O	CPU standby mode: asserted when a CPU is waiting for an event
ps_pl_standbywfi	O	CPU standby mode: asserted when a CPU is waiting for an interrupt.

Table B-4: **FIFO_ENET0**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet0_tx_r_data_rdy	I	When set to logic 1. Indicates enough data is present in the external FIFO for Ethernet frame transmission to commence on the current packet.
enet0_tx_r_rd	O	Single tx_clk clock cycle wide active-High output requesting a 32-bit word of information from the external FIFO interface. Synchronous to the tx_clk clock domain.

Table B-4: **FIFO_ENET0 *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet0_tx_r_valid	I	Single tx_clk clock cycle wide active-High input indicating requested FIFO data is now valid. Validates the following inputs: tx_r_data[31:0], tx_r_sop, tx_r_eop, tx_r_err and tx_r_mod[1:0]
enet0_tx_r_data	I	FIFO data for transmission; this output is only valid while tx_r_valid is High.
enet0_tx_r_sop	I	Start of packet. Indicates the word received from the external FIFO interface is the first in a packet. This input is only valid while tx_r_valid is High.
enet0_tx_r_eop	I	End of packet. Indicates the word received from the external FIFO interface is the last in a packet. This input is only valid while tx_r_valid is High.
enet0_tx_r_err	I	Error, active-High input indicating the current packet contains an error. This signal is only valid while tx_r_valid is High and can be set at any time during the packet transfer.
enet0_tx_r_underflow	I	FIFO underflow. Indicates the transmit FIFO was empty when a read was attempted. This signal is only valid when a read has been attempted and the tx_r_valid signal has not yet been received.
enet0_tx_r_flushed	I	FIFO flush in progress. Indicates the transmit FIFO is currently removing any residue data content.
enet0_tx_r_control	I	tx_no_crc, set active-High at start of packet (SOP) to indicate the current frame is to be transmitted without crc being appended. This input is only valid while both tx_r_valid and tx_r_sop are High.
enet0_dma_tx_end_tog	O	Toggled to indicate that a frame has been completed and status is now valid on the tx_r_status output. Note that this signal is not activated when a frame is being retired due to a collision.
enet0_dma_tx_status_tog	I	This signal must be toggled each time either tx_end_tog or collision_occured are activated. Indicates that the status has been acknowledged.
enet0_tx_r_status	O	[3]: fifo_underrun—status output indicating that the Ethernet media access control (MAC) transmitter has underrun due to one of the following conditions. Data under run indicated by tx_r_underflow input from the external FIFO interface during the last frame transfer. Reset once efifo_tx_status_tog changes logic state. [2]:collision_occured—status output Indicating that the frame in progress has suffered a collision and that re-transmission of the frame should take place. [1]: late_coll_occured—status output indicating that the frame in progress suffered a late collision and can be optionally retired. [0]:too_many_retires—status output indicating the frame in progress experienced excess collisions and was aborted.
enet0_rx_w_wr	O	Single rx_clk clock cycle wide active-High output indicating a write to the external FIFO interface.

Table B-4: **FIFO_ENET0 *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet0_rx_w_data	O	Received data for output to the external FIFO interface. This output is only when rx_w_wr is High.
enet0_rx_w_sop	O	Start of packet. Indicates the word output to the external FIFO interface is the first in a packet. This output is only valid when rx_w_wr is High.
enet0_rx_w_eop	O	End of packet. Indicates the word output to the external FIFO interface is the last in a packet. This output is only valid when rx_w_wr is High.
enet0_rx_w_status	O	Status signals, valid when rx_w_eop is High and rx_w_err is Low, otherwise driven to zero. [29]:Rx_w_type_match—indicates the received frame was matched on type ID register [28]:rx_w_add_match4—indicates the received frame was matched on specific address register4 [27]:rx_w_add_match3—indicates the received frame was matched on specific address register3. [26]:rx_w_add_match3—indicates the received frame was matched on specific address register2. [25]:rx_w_add_match3—indicates the received frame was matched on specific address register1. [24]:rx_w_ext_match—indicates the received frame was matched externally by the eam input pin. [23]:rx_w_uni_hash_match—indicates the received frame was matched as a unicast hash frame. [22]:rx_w_mult_hash_match—indicates the received frame was matched as a multicast hash frame. [21]:rx_w_broadcast_frame—indicates the received frame is a broadcast frame. [20]:rx_w_prty_tagged—indicates a VLAN priority tag detected with received packet. [19:16]:rx_w_tci [3:0]—indicates VLAN priority of received packet. [15]:rx_w_vlan_tagged—indicates VLAN tag detected with received packet. [14]:rx_w_bad_frame—indicates received packet is bad, or the FIFO has overflowed. [13:0]: rx_w_frame_length—indicates number of bytes in received packet.
enet0_rx_w_err	O	Error, active-High output indicating the current packet contains error. This signal is only valid when both rx_w_wr and rx_w_eop are active-High. Rx_w_err is also set if the frame has not been matched by one of the filters.

Table B-4: **FIFO_ENET0 *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet0_rx_w_overflow	I	FIFO overflow. Indicates to the Ethernet MAC that the external RX FIFO has overflowed. The Ethernet MAC uses this signal for status reporting at the end of frame (EOF).
enet0_rx_w_flush	O	FIFO flush, active-High output indicating that the external RX FIFO must be cleared of all data.

Table B-5: **FIFO_ENET1**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet1_tx_r_data_rdy	I	When set to logic 1. Indicates enough data is present in the external FIFO for Ethernet frame transmission to commence on the current packet.
enet1_tx_r_rd	O	Single tx_clk clock cycle wide active-High output requesting a 32-bit word of information from the external FIFO interface. Synchronous to the tx_clk clock domain.
enet1_tx_r_valid	I	Single tx_clk clock cycle wide active-High input indicating requested FIFO data is now valid. Validates the following inputs: tx_r_data[31:0], tx_r_sop, tx_r_eop, tx_r_err and tx_r_mod[1:0].
enet1_tx_r_data	I	FIFO data for transmission. This output is only valid while tx_r_valid is High.
enet1_tx_r_sop	I	Start of packet. Indicates the word received from the external FIFO interface is the first in a packet. This input is only valid while tx_r_valid is High.
enet1_tx_r_eop	I	End of packet. Indicates the word received from the external FIFO interface is the last in a packet. This input is only valid while tx_r_valid is High.
enet1_tx_r_err	I	Error, active-High input indicating the current packet contains an error. This signal is only valid while tx_r_valid is High and can be set at any time during the packet transfer.
enet1_tx_r_underflow	I	FIFO underflow. Indicates the transmit FIFO was empty when a read was attempted. This signal is only valid when a read has been attempted and the tx_r_valid signal has not yet been received.
enet1_tx_r_flushed	I	FIFO flush in progress. Indicates the transmit FIFO is currently removing any residue data content.
enet1_tx_r_control	I	tx_no_crc, set active-High at SOP to indicate current frame is to be transmitted without crc being appended. This input is only valid while both tx_r_valid and tx_r_sop are High.
enet1_dma_tx_end_tog	O	Toggled to indicate that a frame has been completed and status is now valid on the tx_r_status output. Note that this signal is not activated when a frame is being retired due to a collision.
enet1_dma_tx_status_tog	I	This signal must be toggled each time either tx_end_tog or collision_occured are activated. Indicates that the status has been acknowledged.

Table B-5: **FIFO_ENET1 *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet1_tx_r_status	O	[3]: fifo_underrun—status output indicating that the Ethernet MAC transmitter has underrun due to one of the following conditions. Data under run indicated by tx_r_underflow input from the external FIFO interface during the last frame transfer. Reset once efifo_tx_status_tog changes logic state. [2]:collision_occured—status output Indicating that the frame in progress has suffered a collision and that re-transmission of the frame should take place. [1]: late_coll_occured—status output indicating that the frame in progress suffered a late collision and can be optionally retired. [0]:too_many_retires—status output indicating the frame in progress experienced excess collisions and was aborted.
enet1_rx_w_wr	O	Single rx_clk clock cycle wide active-High output indicating a write to the external FIFO interface.
enet1_rx_w_data	O	Received data for output to the external FIFO interface. This output is only when rx_w_wr is High.
enet1_rx_w_sop	O	Start of packet. Indicates the word output to the external FIFO interface is the first in a packet. This output is only valid when rx_w_wr is High.
enet1_rx_w_eop	O	End of packet. Indicates the word output to the external FIFO interface is the last in a packet. This output is only valid when rx_w_wr is High.

Table B-5: **FIFO_ENET1 *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet1_rx_w_status	O	Status signals, valid when rx_w_eop is High and rx_w_err is Low, otherwise driven to zero. [29]:Rx_w_type_match—indicates the received frame was matched on type ID register. [28]:rx_w_add_match4—indicates the received frame was matched on specific address register4. [27]:rx_w_add_match3—indicates the received frame was matched on specific address register3. [26]:rx_w_add_match3—indicates the received frame was matched on specific address register2. [25]:rx_w_add_match3—indicates the received frame was matched on specific address register1. [24]:rx_w_ext_match—indicates the received frame was matched externally by the eam input pin. [23]:rx_w_uni_hash_match—indicates the received frame was matched as a unicast hash frame. [22]:rx_w_mult_hash_match—indicates the received frame was matched as a multicast hash frame. [21]:rx_w_broadcast_frame—indicates the received frame is a broadcast frame. [20]:rx_w_prty_tagged—indicates a VLAN priority tag detected with received packet. [19:16]:rx_w_tci [3:0]—indicates VLAN priority of received packet. [15]:rx_w_vlan_tagged—indicates VLAN tag detected with received packet. [14]:rx_w_bad_frame—indicates received packet is bad or the FIFO has overflowed. [13:0]: rx_w_frame_length—indicates number of bytes in received packet.
enet1_rx_w_err	O	Error, active-High output indicating the current packet contains an error. This signal is only valid when both rx_w_wr and rx_w_eop are active-High. rx_w_err is also set if the frame has not been matched by one of the filters.
enet1_rx_w_overflow	I	FIFO overflow. Indicates to the Ethernet MAC that the external RX FIFO has overflowed. The Ethernet MAC uses this signal for status reporting at the EOF.
enet1_rx_w_flush	O	FIFO flush, active-High output indicating that the external RX FIFO must be cleared of all data.

Table B-6: **FIFO_ENET2**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet2_tx_r_data_rdy	I	When set to logic 1. Indicates enough data is present in the external FIFO for Ethernet frame transmission to commence on the current packet.
enet2_tx_r_rd	O	Single tx_clk clock cycle wide active-High output requesting a 32-bit word of information from the external FIFO interface. Synchronous to the tx_clk clock domain.
enet2_tx_r_valid	I	Single tx_clk clock cycle wide active-High input indicating requested FIFO data is now valid. Validates the following inputs: tx_r_data[31:0], tx_r_sop, tx_r_eop, tx_r_err and tx_r_mod[1:0].
enet2_tx_r_data	I	FIFO data for transmission. This output is only valid while tx_r_valid is High.
enet2_tx_r_sop	I	Start of packet. Indicates the word received from the external FIFO interface is the first in a packet. This input is only valid while tx_r_valid is High.
enet2_tx_r_eop	I	End of packet. Indicates the word received from the external FIFO interface is the last in a packet. This input is only valid while tx_r_valid is High.
enet2_tx_r_err	I	Error. Active-High input indicating the current packet contains an error. This signal is only valid while tx_r_valid is High and can be set at any time during the packet transfer.
enet2_tx_r_underflow	I	FIFO underflow. Indicates the transmit FIFO was empty when a read was attempted. This signal is only valid when a read has been attempted and the tx_r_valid signal has not yet been received.
enet2_tx_r_flushed	I	FIFO flush in progress. Indicates the transmit FIFO is currently removing any residue data content.
enet2_tx_r_control	I	tx_no_crc. Set active-High at SOP to indicate current frame is to be transmitted without crc being appended. This input is only valid while both tx_r_valid and tx_r_sop are High.
enet2_dma_tx_end_tog	O	Toggled to indicate that a frame has been completed and status is now valid on the tx_r_status output. Note that this signal is not activated when a frame is being retired due to a collision.
enet2_dma_tx_status_tog	I	This signal must be toggled each time either tx_end_tog or collision_occured are activated. Indicates that the status has been acknowledged.

Table B-6: **FIFO_ENET2 *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet2_tx_r_status	O	[3]: fifo_underrun—status output indicating that the Ethernet MAC transmitter has under run due to one of the following conditions. Data under run indicated by tx_r_underflow input from the external FIFO interface during the last frame transfer. Reset once efifo_tx_status_tog changes logic state. [2]:collision_occured—status output Indicating that the frame in progress has suffered a collision and that re-transmission of the frame should take place. [1]: late_coll_occured—status output indicating that the frame in progress suffered a late collision and can be optionally retired. [0]:too_many_retires—status output indicating the frame in progress experienced excess collisions and was aborted.
enet2_rx_w_wr	O	Single rx_clk clock cycle wide active-High output indicating a write to the external FIFO interface.
enet2_rx_w_data	O	Received data for output to the external FIFO interface. This output is only when rx_w_wr is High.
enet2_rx_w_sop	O	Start of packet. Indicates the word output to the external FIFO interface is the first in a packet. This output is only valid when rx_w_wr is High.
enet2_rx_w_eop	O	End of packet. Indicates the word output to the external FIFO interface is the last in a packet. This output is only valid when rx_w_wr is High.

Table B-6: **FIFO_ENET2 *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet2_rx_w_status	O	Status signals. Valid when rx_w_eop is High and rx_w_err is Low, otherwise driven to zero. [29]:Rx_w_type_match—indicates the received frame was matched on type ID register [28]:rx_w_add_match4—indicates the received frame was matched on specific address register4 [27]:rx_w_add_match3—indicates the received frame was matched on specific address register3. [26]:rx_w_add_match3—indicates the received frame was matched on specific address register2. [25]:rx_w_add_match3—indicates the received frame was matched on specific address register1. [24]:rx_w_ext_match—indicates the received frame was matched externally by the eam input pin. [23]:rx_w_uni_hash_match—indicates the received frame was matched as a unicast hash frame. [22]:rx_w_mult_hash_match—indicates the received frame was matched as a multicast hash frame. [21]:rx_w_broadcast_frame—indicates the received frame is a broadcast frame. [20]:rx_w_prty_tagged—indicates a VLAN priority tag detected with received packet. [19:16]:rx_w_tci [3:0]—indicates VLAN priority of received packet. [15]:rx_w_vlan_tagged—indicates VLAN tag detected with received packet. [14]:rx_w_bad_frame—indicates received packet is bad, or the FIFO has overflowed. [13:0]: rx_w_frame_length—indicates number of bytes in received packet.
enet2_rx_w_err	O	Error, active-High output indicating the current packet contains error. This signal is only valid when both rx_w_wr and rx_w_eop are active-High. Rx_w_err is also set if the frame has not been matched by one of the filters.
enet2_rx_w_overflow	I	FIFO overflow. Indicates to the Ethernet MAC that the external RX FIFO has overflowed. The Ethernet MAC uses this signal for status reporting at the EOF.
enet2_rx_w_flush	O	FIFO flush, active-High output indicating that the external RX FIFO must be cleared of all data.

Table B-7: **FIFO_ENET3**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet3_tx_r_data_rdy	I	When set to logic 1, indicates enough data is present in the external FIFO for Ethernet frame transmission to commence on the current packet.
enet3_tx_r_rd	O	Single tx_clk clock cycle wide. Active-High output requesting a 32-bit word of information from the external FIFO interface. Synchronous to the tx_clk clock domain.
enet3_tx_r_valid	I	Single tx_clk clock cycle wide. Active-High input indicating requested FIFO data is now valid. Validates the following inputs: tx_r_data[31:0], tx_r_sop, tx_r_eop, tx_r_err and tx_r_mod[1:0]
enet3_tx_r_data	I	FIFO data for transmission. This output is only valid while tx_r_valid is High.
enet3_tx_r_sop	I	Start of packet. Indicates the word received from the external FIFO interface is the first in a packet. This input is only valid while tx_r_valid is High.
enet3_tx_r_eop	I	End of packet. Indicates the word received from the external FIFO interface is the last in a packet. This input is only valid while tx_r_valid is High.
enet3_tx_r_err	I	Error. Active-High input indicating the current packet contains an error. This signal is only valid while tx_r_valid is High and can be set at any time during the packet transfer.
enet3_tx_r_underflow	I	FIFO underflow. Indicates the transmit FIFO was empty when a read was attempted. This signal is only valid when a read has been attempted and the tx_r_valid signal has not yet been received.
enet3_tx_r_flushed	I	FIFO flush in progress. Indicates the transmit FIFO is currently removing any residue data content.
enet3_tx_r_control	I	tx_no_crc. Set active-High at SOP to indicate current frame is to be transmitted without crc being appended. This input is only valid while both tx_r_valid and tx_r_sop are High.
enet3_dma_tx_end_tog	O	Toggled to indicate that a frame has been completed and status is now valid on the tx_r_status output. Note that this signal is not activated when a frame is being retired due to a collision.
enet3_dma_tx_status_tog	I	This signal must be toggled each time either tx_end_tog or collision_occured are activated. Indicates that the status has been acknowledged.

Table B-7: **FIFO_ENET3 *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet3_tx_r_status	O	[3]: fifo_underrun—status output indicating that the Ethernet MAC transmitter has under-run due to one of the following conditions. Data under run indicated by tx_r_underflow input from the external FIFO interface during the last frame transfer. Reset once efifo_tx_status_tog changes logic state. [2]:collision_occured—status output Indicating that the frame in progress has suffered a collision and that re-transmission of the frame should take place. [1]: late_coll_occured—status output indicating that the frame in progress suffered a late collision and can be optionally retired. [0]:too_many_retires—status output indicating the frame in progress experienced excess collisions and was aborted.
enet3_rx_w_wr	O	Single rx_clk clock cycle wide active-High output indicating a write to the external FIFO interface.
enet3_rx_w_data	O	Received data for output to the external FIFO interface. This output is only when rx_w_wr is High.
enet3_rx_w_sop	O	Start of packet. Indicates the word output to the external FIFO interface is the first in a packet. This output is only valid when rx_w_wr is High.
enet3_rx_w_eop	O	End of packet. Indicates the word output to the external FIFO interface is the last in a packet. This output is only valid when rx_w_wr is High.

Table B-7: **FIFO_ENET3 *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet3_rx_w_status	O	Status signals. Valid when rx_w_eop is High and rx_w_err is Low, otherwise driven to zero. [29]:Rx_w_type_match—indicates the received frame was matched on type ID register. [28]:rx_w_add_match4—indicates the received frame was matched on specific address register4. [27]:rx_w_add_match3—indicates the received frame was matched on specific address register3. [26]:rx_w_add_match3—indicates the received frame was matched on specific address register2. [25]:rx_w_add_match3—indicates the received frame was matched on specific address register1. [24]:rx_w_ext_match—indicates the received frame was matched externally by the eam input pin. [23]:rx_w_uni_hash_match—indicates the received frame was matched as a unicast hash frame. [22]:rx_w_mult_hash_match—indicates the received frame was matched as a multicast hash frame. [21]:rx_w_broadcast_frame—indicates the received frame is a broadcast frame. [20]:rx_w_prty_tagged—indicates a VLAN priority tag detected with received packet. [19:16]:rx_w_tci [3:0]—indicates VLAN priority of received packet. [15]:rx_w_vlan_tagged—indicates VLAN tag detected with received packet. [14]:rx_w_bad_frame—indicates received packet is bad, or the FIFO has overflowed. [13:0]: rx_w_frame_length—indicates number of bytes in received packet.
enet3_rx_w_err	O	Error. Active-High output indicating the current packet contains error. This signal is only valid when both rx_w_wr and rx_w_eop are active-High. Rx_w_err is also set if the frame has not been matched by one of the filters.
enet3_rx_w_overflow	I	FIFO overflow. Indicates to the Ethernet MAC that the external RX FIFO has overflowed. The Ethernet MAC uses this signal for status reporting at the EOF.
enet3_rx_w_flush	O	FIFO flush, active-High output indicating that the external RX FIFO must be cleared of all data.

Table B-8: **FTM**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
pl_ps_trigack	I	Trigger acknowledgement from PL
pl_ps_trigger	O	Trigger output to PL
ps_pl_trigack	O	Trigger acknowledgement to PL
ps_pl_trigger	I	Trigger input from PL
gpo	O	General purpose output
gpi	I	General purpose input

Table B-9: **GMII_ENET0**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet0_gmii_rx_clk	I	GEM 0 Receive clock to the system clock generator
enet0_speed_mode	O	Indicates speed and external interface that the GEM is currently configured to use to the system clock generator
enet0_gmii_crs	I	Carrier sense from the PHY
enet0_gmii_col	I	Collision detect from the PHY
enet0_gmii_rxd	I	Receive data from the PHY
enet0_gmii_rx_er	I	Receive error signal from the PHY
enet0_gmii_rx_dv	I	Receive data valid signal from the PHY
enet0_gmii_tx_clk	I	GEM 0 Transmit clock from the system clock generator
enet0_gmii_txd	O	Transmit data to the PHY
enet0_gmii_tx_en	O	Transmit enable to the PHY
enet0_gmii_tx_er	O	Transmit error signal to the PHY. Asserted if the DMA block fails to fetch data from memory during frame transmission.

Table B-10: **GMII_ENET1**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet1_gmii_rx_clk	I	GEM 1 Receive clock to the system clock generator
enet1_speed_mode	O	Indicates speed and external interface that the GEM is currently configured to use to the system clock generator
enet1_gmii_crs	I	Carrier sense from the PHY
enet1_gmii_col	I	Collision detect from the PHY
enet1_gmii_rxd	I	Receive data from the PHY
enet1_gmii_rx_er	I	Receive error signal from the PHY
enet1_gmii_rx_dv	I	Receive data valid signal from the PHY
enet1_gmii_tx_clk	I	GEM 1 Transmit clock from the system clock generator
enet1_gmii_txd	O	Transmit data to the PHY
enet1_gmii_tx_en	O	Transmit enable to the PHY
enet1_gmii_tx_er	O	Transmit error signal to the PHY. Asserted if the DMA block fails to fetch data from memory during frame transmission

Table B-11: **GMII_ENET2**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet2_gmii_rx_clk	I	GEM 2 Receive clock to the system clock generator
enet2_speed_mode	O	Indicates speed and external interface that the GEM is currently configured to use to the system clock generator
enet2_gmii_crs	I	Carrier sense from the PHY
enet2_gmii_col	I	Collision detect from the PHY
enet2_gmii_rxd	I	Receive data from the PHY
enet2_gmii_rx_er	I	Receive error signal from the PHY
enet2_gmii_rx_dv	I	Receive data valid signal from the PHY
enet2_gmii_tx_clk	I	GEM 3 Transmit clock from the system clock generator
enet2_gmii_txd	O	Transmit data to the PHY
enet2_gmii_tx_en	O	Transmit enable to the PHY
enet2_gmii_tx_er	O	Transmit error signal to the PHY. Asserted if the DMA block fails to fetch data from memory during frame transmission.

Table B-12: **GMII_ENET3**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet3_gmii_rx_clk	I	GEM 3 Receive clock to the system clock generator
enet3_speed_mode	O	Indicates speed and external interface that the GEM is currently configured to use to the system clock generator.
enet3_gmii_crs	I	Carrier sense from the PHY
enet3_gmii_col	I	Collision detect from the PHY
enet3_gmii_rxd	I	Receive data from the PHY
enet3_gmii_rx_er	I	Receive error signal from the PHY
enet3_gmii_rx_dv	I	Receive data valid signal from the PHY
enet3_gmii_tx_clk	I	GEM 3 Transmit clock from the system clock generator
enet3_gmii_txd	O	Transmit data to the PHY
enet3_gmii_tx_en	O	Transmit enable to the PHY
enet3_gmii_tx_er	O	Transmit error signal to the PHY. Asserted if the DMA block fails to fetch data from memory during frame transmission.

Table B-13: **GPIO_0**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
gpio_i	I	GPIO port input
gpio_o	O	GPIO port output
gpio_t	O	3-state enable signal for GPIO port

Table B-14: **IIC0**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
i2c0_scl_i	I	Actual state of the external serial clock (SCL) clock signal
i2c0_scl_o	O	Clock level to be placed on SCL pin
i2c0_scl_t	O	3-state enable for the SCL output buffer. This signal has a direct connection to i2c0_scl_oe.
i2c0_sda_i	I	Actual state of the external serial data (SDA) signal
i2c0_sda_o	O	Data bit to be placed on external SDA signal
i2c0_sda_t	O	3-state enable for the SDA output buffer This signal has a direct connection to i2c0_sda_oe.

Table B-15: **IIC1**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
i2c1_scl_i	I	Actual state of the external SCL clock signal
i2c1_scl_o	O	Clock level to be placed on SCL pin
i2c1_scl_t	O	3-state enable for the SCL output buffer. This signal has a direct connection to i2c1_scl_oe.
i2c1_sda_i	I	Actual state of the external SDA signal
i2c1_sda_o	O	Data bit to be placed on external SDA signal
i2c1_sda_t	O	3-state enable for the SDA output buffer. This signal has a direct connection to i2c1_sda_oe.

Table B-16: **MDIO_ENET0**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet0_mdio_mdc	O	Management data clock to pin
enet0_mdio_i	I	Management data input from MDIO pin
enet0_mdio_o	O	Management data output to MDIO pin
enet0_mdio_t	O	3-state enable to MDIO pin, active-Low. At the top-level the three MDIO pins are all used to drive a single 3-state pin.

Table B-17: **MDIO_ENET1**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet1_mdio_mdc	O	Management data clock to pin
enet1_mdio_i	I	Management data input from MDIO pin
enet1_mdio_o	O	Management data output to MDIO pin
enet1_mdio_t	O	3-state enable to MDIO pin, active-Low. At the top-level the three MDIO pins are all used to drive a single 3-state pin.

Table B-18: **MDIO_ENET2**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet2_mdio_mdc	O	Management data clock to pin
enet2_mdio_i	I	Management data input from MDIO pin

Table B-18: **MDIO_ENET2 *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet2_mdio_o	O	Management data output to MDIO pin
enet2_mdio_t	O	3-state enable to MDIO pin, active-Low. At the top-level the three MDIO pins are all used to drive a single 3-state pin.

Table B-19: **MDIO_ENET3**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
enet3_mdio_mdc	O	Management data clock to pin
enet3_mdio_i	I	Management data input from MDIO pin
enet3_mdio_o	O	Management data output to MDIO pin
enet3_mdio_t	O	3-state enable to MDIO pin, active-Low. At the top-level the three MDIO pins are all used to drive a single 3-state pin.

Table B-20: **PL_CLK0**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
pl_clk0	O	PL Clock 0

Table B-21: **PL_CLK1**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
pl_clk1	O	PL Clock 1

Table B-22: **PL_CLK2**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
pl_clk2	O	PL Clock 2

Table B-23: **PL_CLK3**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
pl_clk3	O	PL Clock 3

Table B-24: **PL_PS_IRQ0**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
pl_ps_irq0	I	pl to ps interrupt 0

Table B-25: **PL_PS_IRQ1**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
pl_ps_irq1	I	pl to ps interrupt 1

Table B-26: **SDIO0**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
sdio0_clkout	O	Clock output to SD/SDIO0 slave device
sdio0_fb_clk_in	I	Clock feedback from sd0_clk_out from pad
sdio0_cmdout	O	Command indicator output
sdio0_cmdin	I	Command indicator input
sdio0_cmdena	O	Command indicator enable
sdio0_datain	I	7-bit input data bus. Can also be used in SPI flash memory, serial or 2-bit modes.
sdio0_dataout	O	7-bit output data bus. Can also be used in SPI flash memory, serial or 2-bit modes.
sdio0_dataena	O	Enable control for data bus
sdio0_cd_n	I	Card detection for single slot
sdio0_wp	I	Secure digital non-volatile memory card (SD card) write protect, active-Low
sdio0_ledcontrol	O	LED ON. Cautions you not to remove the card while the SD card is being accessed.
sdio0_buspower	O	Control SD card power supply
sdio0_bus_volt	O	SD bus volt select

Table B-27: **SDIO1**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
sdio1_clkout	O	Clock output to SD/SDIO1 slave device
sdio1_fb_clk_in	I	Clock feedback from sd1_clk_out from pad

Table B-27: **SDIO1 *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
sdio1_cmdout	O	Command indicator output
sdio1_cmdin	I	Command indicator input
sdio1_cmdena	O	Command indicator enable
sdio1_datain	I	7-bit input data bus. Can also be used in SPI flash memory, serial or 2-bit modes.
sdio1_dataout	O	7-bit output data bus. Can also be used in SPI flash memory, serial or 2-bit modes.
sdio1_dataena	O	Enable control for data bus
sdio1_cd_n	I	Card detection for single slot
sdio1_wp	I	SD card write protect, active-Low
sdio1_ledcontrol	O	LED ON: Cautions you not to remove the card while the SD card is being accessed.
sdio1_bus_power	O	Control SD card power supply
sdio1_bus_volt	O	SD bus volt select

Table B-28: **SPI0**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
spi0_sclk_i	I	SPI flash memory slave clock
spi0_sclk_o	O	SPI flash memory master clock output
spi0_sclk_t	O	SPI flash memory clock 3-state enable, active-Low. This signal is a version of spi0_n_sclk_en.
spi0_m_i	I	SPI flash memory master in slave out (MISO) signal, master input
spi0_m_o	O	SPI flash memory master out slave in (MOSI) signal, master output
spi0_mo_t	O	SPI flash memory MOSI signal, 3-state enable, active-Low. This signal is a version of spi0_n_mo_en.
spi0_s_i	I	SPI flash memory MOSI signal, slave input
spi0_s_o	O	SPI flash memory MISO signal, slave output
spi0_n_ss_o_n	O	SPI flash memory slave select outputs
spi0_ss_n_t	O	SPI flash memory slave select 3-state enable, active-Low. This signal is a version of spi0_n_ss_en.

Table B-29: **SPI1**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
spi1_sclk_i	I	SPI flash memory slave clock. Can be passed directly from pin if low speed (< 50 MHz).
spi1_sclk_o	O	SPI flash memory master clock output. Can be passed directly to pin if low speed (< 50 MHz).
spi1_sclk_t	O	SPI flash memory clock 3-state enable, active-Low. This signal is a version of spi1_n_sclk_en
spi1_m_i	I	SPI flash memory MISO signal, master input
spi1_m_o	O	SPI flash memory MOSI signal, master output
spi1_mo_t	O	SPI flash memory MOSI signal, 3-state enable, active-Low. This signal is a version of spi1_n_mo_en.
spi1_s_i	I	SPI flash memory MOSI signal, slave input
spi1_s_o	O	SPI flash memory MISO signal, slave output
spi1_n_ss_o_n	O	SPI flash memory peripheral select outputs
spi1_ss_n_t	O	SPI flash memory slave select 3-state enable, active-Low. This signal is a version of spi1_n_ss_en.

Table B-30: **Trace0**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
tracectl	O	Trace control
tracedata	O	Trace data

Table B-31: **UART0**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
uart0_ctsn	I	Clear-to-send flow control
uart0_rtsn	O	Request-to-send flow control
uart0_dsrn	I	Modem data set ready
uart0_dcdn	I	Modem data carrier detect
uart0_rin	I	Modem ring indicator
uart0_dtrn	O	Modem data terminal ready

Table B-32: **UART1**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
uart1_ctsn	I	Clear-to-send flow control
uart1_rtsn	O	Request-to-send flow control
uart1_dsrn	I	Modem data set ready
uart1_dcdn	I	Modem data carrier detect
uart1_rin	I	Modem ring indicator
uart1_dtrn	O	Modem data terminal ready

Table B-33: **TTC0**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
ttc0_wave_o	O	Triple timer counter (TTC) clock (Waveform generated)
ttc0_clk_i	I	TTC0 clock input

Table B-34: **TTC1**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
ttc1_wave_o	O	TTC clock (Waveform generated)
ttc1_clk_i	I	TTC1 clock input

Table B-35: **TTC2**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
ttc3_wave_o	O	TTC clock (Waveform generated)
ttc2_clk_i	I	TTC2 clock input

Table B-36: **TTC3**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
ttc3_wave_o	O	TTC clock (Waveform generated)
ttc3_clk_i	I	TTC3 clock input

Table B-37: **WDT0**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
wdt0_clk_i	I	WDT0 clock input
wdt0_rst_o	O	WDT0 reset

Table B-38: **WDT1**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
wdt1_clk_i	I	WDT1 clock input
wdt1_rst_o	O	WDT1 reset

Table B-39: **Interrupt Signals**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
ps_pl_irq_can0	O	CAN0 interrupt
ps_pl_irq_can1	O	CAN1 interrupt
ps_pl_irq_enet0	O	Ethernet0 interrupt
ps_pl_irq_enet1	O	Gigabit ethernet1 interrupt
ps_pl_irq_enet2	O	Gigabit ethernet2 interrupt
ps_pl_irq_enet3	O	Gigabit ethernet3 interrupt
ps_pl_irq_enet0_wake0	O	Ethernet0 wake-up interrupt
ps_pl_irq_enet0_wake1	O	Gigabit ethernet1 wake-up interrupt
ps_pl_irq_enet0_wake2	O	Gigabit ethernet2 wake-up interrupt
ps_pl_irq_enet0_wake3	O	Gigabit ethernet3 wake-up interrupt
ps_pl_irq_gpio	O	GPIO interrupt
ps_pl_irq_i2c0	O	I2C0 interrupt

Table B-39: **Interrupt Signals *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
ps_pl_irq_i2c1	O	I2C1 interrupt
ps_pl_irq_uart0	O	UART0 interrupt
ps_pl_irq_uart1	O	UART1 interrupt
ps_pl_irq_sdio0	O	SDIO0 interrupt
ps_pl_irq_sdio1	O	SDIO1 interrupt
ps_pl_irq_sdio0_wake	O	SDIO0 wake interrupt
ps_pl_irq_sdio1_wake	O	SDIO1 wake interrupt
ps_pl_irq_spi0	O	SPI0 interrupt
ps_pl_irq_spi1	O	SPI1 interrupt
ps_pl_irq_qspi	O	SPI flash memory interrupt
ps_pl_irq_ttc0_0	O	Triple Timer 0 Counter 0 Interrupt
ps_pl_irq_ttc0_1	O	Triple Timer 0 Counter 1 Interrupt
ps_pl_irq_ttc0_2	O	Triple Timer 0 Counter 2 Interrupt
ps_pl_irq_ttc1_0	O	Triple Timer 1 Counter 0 Interrupt
ps_pl_irq_ttc1_1	O	Triple Timer 1 Counter 1 Interrupt
ps_pl_irq_ttc1_2	O	Triple Timer 1 Counter 2 Interrupt
ps_pl_irq_ttc2_0	O	Triple Timer 2 Counter 0 Interrupt

Table B-40: **M_AXI_HPM0_FPD**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp0_awid	O	Write address ID. This signal is the identification tag for the write address group of signals.
maxigp0_awaddr	O	Write address. The write address bus gives the address of the first transfer in a write burst transaction. The associated control signals are used to determine the addresses of the remaining transfers in the burst.
maxigp0_awlen	O	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
maxigp0_awsize	O	Burst size. This signal indicates the size of each transfer in the burst. Byte lane strobes indicate exactly which byte lanes to update.
maxigp0_awburst	O	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
maxigp0_awlock	O	Lock type. This signal provides additional information about the atomic characteristics of the transfer.

Table B-40: **M_AXI_HPM0_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp0_awcache	O	Cache type. This signal indicates the bufferable, cacheable, write-through, write-back, and allocate attributes of the transaction.
maxigp0_awprot	O	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.
maxigp0_awvalid	O	Write address valid. This signal indicates that valid write address and control information are available. 1 = address and control information available 0 = address and control information not available The address and control information remain stable until the address acknowledge signal, AWREADY, goes High.
maxigp0_awuser	O	User-defined address write (AW) channel signals
maxigp0_awready	I	Write address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready
maxigp0_wdata	O	Write data. The write data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
maxigp0_wstrb	O	Write strobes. This signal indicates which byte lanes to update in memory. There is one write strobe for each eight bits of the write data bus.
maxigp0_wlast	O	Write last. This signal indicates the last transfer in a write burst.
maxigp0_wvalid	O	Write valid. This signal indicates that valid write data and strobes are available. 1 = write data and strobes available 0 = write data and strobes not available
maxigp0_wready	I	Write ready. This signal indicates that the slave can accept the write data. 1 = slave ready 0 = slave not ready
maxigp0_bid	I	Response ID. The identification tag of the write response
maxigp0_bresp	I	Write response. This signal indicates the status of the write transaction. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
maxigp0_bvalid	I	Write response valid. This signal indicates that a valid write response is available. 1 = write response available 0 = write response not available

Table B-40: **M_AXI_HPM0_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp0_bready	O	Response ready. This signal indicates that the master can accept the response information. 1 = master ready 0 = master not ready
maxigp0_arid	O	Read address ID. This signal is the identification tag for the read address group of signals.
maxigp0_araddr	O	Read address. The read address bus gives the initial address of a read burst transaction.
maxigp0_arlen	O	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
maxigp0_arsize	O	Burst size. This signal indicates the size of each transfer in the burst
maxigp0_arburst	O	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
maxigp0_arlock	O	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
maxigp0_arcache	O	Cache type. This signal provides additional information about the cacheable characteristics of the transfer
maxigp0_arprot	O	Protection type. This signal provides protection unit information for the transaction.
maxigp0_arvalid	O	Read address valid. This signal indicates, when High, that the read address and control information is valid and remains stable until the address acknowledge signal, ARREADY, is High.
maxigp0_aruser	O	User-defined address read (AR) channel signals
maxigp0_arready	I	Read address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready
maxigp0_rid	I	Read ID tag. This signal is the ID tag of the read data group of signals.
maxigp0_rdata	I	Read data. The read data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
maxigp0_rresp	I	Read response. This signal indicates the status of the read transfer. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
maxigp0_rlast	I	Read last. This signal indicates the last transfer in a read burst.
maxigp0_rvalid	I	Read valid. This signal indicates that the required read data is available and the read transfer can complete.
maxigp0_rready	O	Read ready. This signal indicates that the master can accept the read data and response information. 1= master ready 0 = master not ready

Table B-40: **M_AXI_HPM0_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp0_awqos	O	Wr addr channel quality of service (QOS) input
maxigp0_arqos	O	Rd addr channel QOS input

Table B-41: **M_AXI_HPM0_FPD_ACLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp0_awid	O	Write address ID. This signal is the identification tag for the write address group of signals.
maxihpm0_fpd_aclk	I	Input clock signal

Table B-42: **M_AXI_HPM0_LPD**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp2_awid	O	Write address ID. This signal is the identification tag for the write address group of signals.
maxigp2_awaddr	O	Write address. The write address bus gives the address of the first transfer in a write burst transaction. The associated control signals are used to determine the addresses of the remaining transfers in the burst.
maxigp2_awlen	O	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
maxigp2_awsize	O	Burst size. This signal indicates the size of each transfer in the burst. Byte lane strobes indicate exactly which byte lanes to update.
maxigp2_awburst	O	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
maxigp2_awlock	O	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
maxigp2_awcache	O	Cache type. This signal indicates the buffer able, cacheable, write-through, write-back, and allocate attributes of the transaction.
maxigp2_awprot	O	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.

Table B-42: **M_AXI_HPM0_LPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp2_awvalid	O	Write address valid. This signal indicates that valid write address and control information are available. 1 = address and control information available 0 = address and control information not available The address and control information remain stable until the address acknowledge signal, AWREADY, goes High.
maxigp2_awuser	O	User-defined address write (AW) channel signals
maxigp2_awready	I	Write address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready
maxigp2_wdata	O	Write data. The write data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
maxigp2_wstrb	O	Write strobes. This signal indicates which byte lanes to update in memory. There is one write strobe for each 8 bits of the write data bus.
maxigp2_wlast	O	Write last. This signal indicates the last transfer in a write burst.
maxigp2_wvalid	O	Write valid. This signal indicates that valid write data and strobes are available. 1 = write data and strobes available 0 = write data and strobes not available
maxigp2_wready	I	Write ready. This signal indicates that the slave can accept the write data. 1 = slave ready 0 = slave not ready
maxigp2_bid	I	Response ID. The identification tag of the write response
maxigp2_bresp	I	Write response. This signal indicates the status of the write transaction. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
maxigp2_bvalid	I	Write response valid. This signal indicates that a valid write response is available. 1 = write response available 0 = write response not available
maxigp2_bready	O	Response ready. This signal indicates that the master can accept the response information. 1 = master ready 0 = master not ready
maxigp2_arid	O	Read address ID. This signal is the identification tag for the read address group of signals.

Table B-42: **M_AXI_HPM0_LPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp2_araddr	O	Read address. The read address bus gives the initial address of a read burst transaction.
maxigp2_arlen	O	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
maxigp2_arsize	O	Burst size. This signal indicates the size of each transfer in the burst.
maxigp2_arburst	O	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
maxigp2_arlock	O	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
maxigp2_arcache	O	Cache type. This signal provides additional information about the cacheable characteristics of the transfer.
maxigp2_arprot	O	Protection type. This signal provides protection unit information for the transaction.
maxigp2_arvalid	O	Read address valid. This signal indicates, when High, that the read address and control information is valid and remains stable until the address acknowledge signal, ARREADY, is High.
maxigp2_aruser	O	User-defined AR channel signals
maxigp2_arready	I	Read address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready
maxigp2_rid	I	Read ID tag. This signal is the ID tag of the read data group of signals.
maxigp2_rdata	I	Read data. The read data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
maxigp2_rresp	I	Read response. This signal indicates the status of the read transfer. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR
maxigp2_rlast	I	Read last. This signal indicates the last transfer in a read burst.
maxigp2_rvalid	I	Read valid. This signal indicates that the required read data is available and the read transfer can complete.
maxigp2_rready	O	Read ready. This signal indicates that the master can accept the read data and response information. 1= master ready 0= master not ready

Table B-42: **M_AXI_HPM0_LPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp2_awqos	O	Wr addr channel QOS input
maxigp2_arqos	O	Rd addr channel QOS input

Table B-43: **M_AXI_HPM0_LPD_ACLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp2_awid	O	Write address ID. This signal is the identification tag for the write address group of signals.
maxihpm0_lpd_aclk	I	Input clock signal

Table B-44: **M_AXI_HPM1_FPD**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp1_awid	O	Write address ID. This signal is the identification tag for the write address group of signals.
maxigp1_awaddr	O	Write address. The write address bus gives the address of the first transfer in a write burst transaction. The associated control signals are used to determine the addresses of the remaining transfers in the burst.
maxigp1_awlen	O	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
maxigp1_awsize	O	Burst size. This signal indicates the size of each transfer in the burst. Byte lane strobes indicate exactly which byte lanes to update.
maxigp1_awburst	O	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
maxigp1_awlock	O	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
maxigp1_awcache	O	Cache type. This signal indicates the buffer able, cacheable, write-through, write-back, and allocate attributes of the transaction.
maxigp1_awprot	O	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.

Table B-44: **M_AXI_HPM1_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp1_awvalid	O	Write address valid. This signal indicates that valid write address and control information are available. 1 = address and control information available 0 = address and control information not available The address and control information remain stable until the address acknowledge signal, AWREADY, goes High.
maxigp1_awuser	O	User-defined AW channel signals
maxigp1_awready	I	Write address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready
maxigp1_wdata	O	Write data. The write data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
maxigp1_wstrb	O	Write strobes. This signal indicates which byte lanes to update in memory. There is one write strobe for each eight bits of the write data bus.
maxigp1_wlast	O	Write last. This signal indicates the last transfer in a write burst.
maxigp1_wvalid	O	Write valid. This signal indicates that valid write data and strobes are available. 1 = write data and strobes available 0 = write data and strobes not available
maxigp1_wready	I	Write ready. This signal indicates that the slave can accept the write data. 1 = slave ready 0 = slave not ready
maxigp1_bid	I	Response ID. The identification tag of the write response
maxigp1_bresp	I	Write response. This signal indicates the status of the write transaction. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
maxigp1_bvalid	I	Write response valid. This signal indicates that a valid write response is available. 1 = write response available 0 = write response not available
maxigp1_bready	O	Response ready. This signal indicates that the master can accept the response information. 1 = master ready 0 = master not ready
maxigp1_arid	O	Read address ID. This signal is the identification tag for the read address group of signals.

Table B-44: **M_AXI_HPM1_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp1_araddr	O	Read address. The read address bus gives the initial address of a read burst transaction.
maxigp1_arlen	O	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
maxigp1_arsize	O	Burst size. This signal indicates the size of each transfer in the burst.
maxigp1_arburst	O	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
maxigp1_arlock	O	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
maxigp1_arcache	O	Cache type. This signal provides additional information about the cacheable characteristics of the transfer.
maxigp1_arprot	O	Protection type. This signal provides protection unit information for the transaction.
maxigp1_arvalid	O	Read address valid. This signal indicates, when High, that the read address and control information is valid and remains stable until the address acknowledge signal, ARREADY, is High.
maxigp1_aruser	O	User-defined AR channel signals
maxigp1_arready	I	Read address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready
maxigp1_rid	I	Read ID tag. This signal is the ID tag of the read data group of signals.
maxigp1_rdata	I	Read data. The read data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
maxigp1_rresp	I	Read response. This signal indicates the status of the read transfer. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
maxigp1_rlast	I	Read last. This signal indicates the last transfer in a read burst.
maxigp1_rvalid	I	Read valid. This signal indicates that the required read data is available and the read transfer can complete.
maxigp1_rready	O	Read ready. This signal indicates that the master can accept the read data and response information. 1= master ready 0 = master not ready

Table B-44: **M_AXI_HPM1_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp1_awqos	O	Wr addr channel QOS input
maxigp1_arqos	O	Rd addr channel QOS input

Table B-45: **M_AXI_HPM1_FPD_ACLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
maxigp1_awid	O	Write address ID. This signal is the identification tag for the write address group of signals.
maxihpm1_fpd_aclk	I	Input clock signal

Table B-46: **S_AXI_ACE_FPD**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
sacefpd_wuser	I	User signal. Optional user-defined signal in the write data channel.
sacefpd_buser	O	User signal. Optional user-defined signal in the write response channel.
sacefpd_ruser	O	User signal. Optional user-defined signal in the read data channel.
sacefpd_awuser	I	User signal. Optional user-defined signal in the write address channel.
sacefpd_awsnoop	I	This signal indicates the transaction type for shareable write transactions.
sacefpd_awsize	I	Burst size. This signal indicates the size of each transfer in the burst. Byte lane strobes indicate exactly which byte lanes to update.
sacefpd_awregion	I	Region identifier. Permits a single physical interface on a slave to be used for multiple logical interfaces.
sacefpd_awqos	I	Quality of service. Identifier sent for each write transaction.
sacefpd_awprot	I	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.
sacefpd_awlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
sacefpd_awid	I	Write address ID. This signal is the identification tag for the write address group of signals.
sacefpd_awdomain	I	The signal indicates the shareability domain of a write transaction.
sacefpd_awcache	I	Cache type. This signal indicates the buffer able, cacheable, write-through, write-back, and allocate attributes of the transaction.
sacefpd_awburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
sacefpd_awbar	I	This signal indicates a write barrier transaction.
sacefpd_awaddr	I	Write address. The write address bus gives the address of the first transfer in a write burst transaction. The associated control signals are used to determine the addresses of the remaining transfers in the burst.
sacefpd_awlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.

Table B-46: **S_AXI_ACE_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
sacefpd_awvalid	I	Write address valid. This signal indicates that valid write address and control information are available. 1 = address and control information available 0 = address and control information not available The address and control information remain stable until the address acknowledge signal, AWREADY, goes High.
sacefpd_awready	O	Write address channel ready signal
sacefpd_wstrb	I	Write strobes. This signal indicates which byte lanes to update in memory. There is one write strobe for each eight bits of the write data bus.
sacefpd_wdata	I	Write data. The write data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
sacefpd_wlast	I	Write last. This signal indicates the last transfer in a write burst.
sacefpd_wvalid	I	Write valid. This signal indicates that valid write data and strobes are available. 1 = write data and strobes available 0 = write data and strobes not available
sacefpd_wready	O	Write ready. This signal indicates that the slave can accept the write data. 1 = slave ready 0 = slave not ready
sacefpd_bresp	O	Write response. This signal indicates the status of the write transaction. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
sacefpd_bid	O	Response ID. The identification tag of the write response
sacefpd_bvalid	O	Write response valid. This signal indicates that a valid write response is available. 1 = write response available 0 = write response not available
sacefpd_bready	I	Response ready. This signal indicates that the master can accept the response information. 1 = master ready 0 = master not ready
sacefpd_aruser	I	User signal. Optional User-defined signal in the read address channel.
sacefpd_arsnoop	I	This signal indicates the transaction type for shareable read transactions.
sacefpd_arsize	I	Burst size. This signal indicates the size of each transfer in the burst.

Table B-46: **S_AXI_ACE_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
sacefpd_arregion	I	Region Identifier. Permits a single physical interface on a slave to be used for multiple logical interfaces.
sacefpd_arqos	I	Quality of service, identifier sent for each read transaction.
sacefpd_arprot	I	Protection type. This signal provides protection unit information for the transaction.
sacefpd_arlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
sacefpd_arid	I	Read address ID. This signal is the identification tag for the read address group of signals.
sacefpd_ardomain	I	This signal indicates the shareability domain of a read transaction.
sacefpd_arcache	I	Cache type. This signal provides additional information about the cacheable characteristics of the transfer.
sacefpd_arburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
sacefpd_arbar	I	This signal indicates a read barrier transaction.
sacefpd_araddr	I	Read address. The read address bus gives the initial address of a read burst transaction.
sacefpd_arlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
sacefpd_arvalid	I	Read address valid. This signal indicates, when High, that the read address and control information is valid and remains stable until the address acknowledge signal, ARREADY, is High.
sacefpd_arready	O	Read address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready
sacefpd_rresp	O	Read response. This signal indicates the status of the read transfer. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
sacefpd_rid	O	Read ID tag. This signal is the ID tag of the read data group of signals.
sacefpd_rdata	O	Read data. The read data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
sacefpd_rlast	O	Read last. This signal indicates the last transfer in a read burst.
sacefpd_rvalid	O	Read valid. This signal indicates that the required read data is available and the read transfer can complete.

Table B-46: **S_AXI_ACE_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
sacefpd_rready	I	Read ready. This signal indicates that the master can accept the read data and response information. 1= master ready 0 = master not ready
sacefpd_acsnoop	O	Snoop transaction type. This signal indicates the transaction type of the snoop transaction.
sacefpd_acprot	O	Snoop protection type. This signal indicates the security level of the snoop transaction.
sacefpd_acaddr	O	Snoop Address. This signal indicates the address of a snoop transaction. The snoop address width must match the width of the read and write address buses.
sacefpd_acvalid	O	Snoop address valid. This signal indicates that the snoop address and control information is valid.
sacefpd_acready	I	Snoop address ready. This signal indicates that the snoop address and control information can be accepted in the current cycle.
sacefpd_cddata	I	Snoop data. Transfer data from a snooped master.
sacefpd_cdlast	I	This signal indicates the last data transfer of a snoop transaction.
sacefpd_cdvalid	I	Snoop data valid. This signal indicates that the snoop is valid.
sacefpd_cdready	O	Snoop data ready. This signal indicates that the snoop data can be accepted in the current cycle.
sacefpd_crresp	I	Snoop response. This signal indicates the response to a snoop transaction and how it completes.
sacefpd_crvalid	I	Snoop response valid. This signal indicates that the snoop response is valid.
sacefpd_crready	O	Snoop response ready. This signal indicates the snoop response can be accepted in the current cycle.
sacefpd_wack	I	Write acknowledge. This signal indicates that a master has completed a write transaction.
sacefpd_rack	I	Read acknowledge. This signal indicates that a master has completed a read transaction.

Table B-47: **S_AXI_ACP_FPD**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxiacp_awuser	I	User signal. Optional user-defined signal in the write address channel.
saxiacp_buser	O	User signal. Optional user-defined signal in the write response channel.
saxiacp_wuser	I	User signal. Optional user-defined signal in the write data channel.
saxiacp_awid	I	Write address ID. This signal is the identification tag for the write address group of signals.
saxiacp_awaddr	I	Write address. The write address bus gives the address of the first transfer in a write burst transaction. The associated control signals are used to determine the addresses of the remaining transfers in the burst.
saxiacp_awlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
saxiacp_awsize	I	Burst size. This signal indicates the size of each transfer in the burst. Byte lane strobes indicate exactly which byte lanes to update.
saxiacp_awburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
saxiacp_awlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
saxiacp_awcache	I	Cache type. This signal indicates the buffer able, cacheable, write-through, write-back, and allocate attributes of the transaction.
saxiacp_awprot	I	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.
saxiacp_awvalid	I	Write address valid. This signal indicates that valid write address and control information are available. 1 = address and control information available 0 = address and control information not available The address and control information remain stable until the address acknowledge signal, AWREADY, goes High.
saxiacp_awready	I	Write address channel ready signal
saxiacp_wdata	I	Write data. The write data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
saxiacp_wstrb	I	Write strobes. This signal indicates which byte lanes to update in memory. There is one write strobe for each eight bits of the write data bus.

Table B-47: **S_AXI_ACP_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxiacp_wlast	I	Write last. This signal indicates the last transfer in a write burst.
saxiacp_wvalid	O	Write valid. This signal indicates that valid write data and strobes are available. 1 = write data and strobes available 0 = write data and strobes not available
saxiacp_wready	O	Write ready. This signal indicates that the slave can accept the write data. 1 = slave ready 0 = slave not ready
saxiacp_bid	O	Response ID. The identification tag of the write response
saxiacp_bresp	O	Write response. This signal indicates the status of the write transaction. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
saxiacp_bvalid	I	Write response valid. This signal indicates that a valid write response is available. 1 = write response available 0 = write response not available
saxiacp_bready	I	Response ready. This signal indicates that the master can accept the response information. 1 = master ready 0 = master not ready
saxiacp_arid	I	Read address ID. This signal is the identification tag for the read address group of signals.
saxiacp_araddr	I	Read address. The read address bus gives the initial address of a read burst transaction.
saxiacp_arlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
saxiacp_arsize	I	Burst size. This signal indicates the size of each transfer in the burst.
saxiacp_arburst	I	Burst type. The burst type and the size information determine how the address for each transfer within the burst is calculated.
saxiacp_arlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
saxiacp_arcache	I	Cache type. This signal provides additional information about the cacheable characteristics of the transfer.
saxiacp_arprot	I	Protection type. This signal indicates the privilege and security level of the transaction, and whether the transaction is a data access or an instruction access.

Table B-47: **S_AXI_ACP_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxiacp_arvalid	O	Read address valid. This signal indicates, when High, that the read address and control information is valid and remains stable until the address acknowledge signal, ARREADY, is High
saxiacp_arready	O	Read address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready
saxiacp_rid	O	Read ID tag. This signal is the ID tag of the read data group of signals.
saxiacp_rdata	O	Read data. The read data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
saxiacp_rresp	O	Read response. This signal indicates the status of the read transfer. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
saxiacp_rlast	I	Read last. This signal indicates the last transfer in a read burst.
saxiacp_rvalid	O	Read valid. This signal indicates that the required read data is available and the read transfer can complete.
saxiacp_rready	O	Read ready. This signal indicates that the master can accept the read data and response information. 1= master ready 0 = master not ready
saxiacp_awqos	O	Wr addr channel QOS input.
saxiacp_arqos	O	Rd addr channel QOS input. Quality of service, sent for each read transaction.

Table B-48: **S_AXI_ACP_FPD_ACLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxiacp_awuser	I	User signal. Optional user-defined signal in the write address channel
saxiacp_fpd_aclk	I	Input clock signal

Table B-49: **S_AXI_HP0_FPD**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
saxigp3_awuser	I	User-defined AW channel signals
saxigp3_awid	I	Write address ID. This signal is the identification tag for the write address group of signals.
saxigp3_awaddr	I	Write address. The write address bus gives the address of the first transfer in a write burst transaction. The associated control signals are used to determine the addresses of the remaining transfers in the burst.
saxigp3_awlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
saxigp3_awsize	I	Burst size. This signal indicates the size of each transfer in the burst. Byte lane strobes indicate exactly which byte lanes to update.
saxigp3_awburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
saxigp3_awlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
saxigp3_awcache	I	Cache type. This signal indicates the buffer able, cacheable, write-through, write-back, and allocate attributes of the transaction.
saxigp3_awprot	I	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.
saxigp3_awvalid	I	Write address valid. This signal indicates that valid write address and control information are available. 1 = address and control information available 0 = address and control information not available. The address and control information remain stable until the address acknowledge signal, AWREADY, goes High.
saxigp3_awready	O	Write address channel ready signal
saxigp3_wdata	I	Write data. The write data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide
saxigp3_wstrb	I	Write strobes. This signal indicates which byte lanes to update in memory. There is one write strobe for each eight bits of the write data bus.
saxigp3_wlast	I	Write last. This signal indicates the last transfer in a write burst.

Table B-49: **S_AXI_HP0_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_wvalid	I	Write valid. This signal indicates that valid write data and strobes are available. 1 = write data and strobes available 0 = write data and strobes not available
saxigp3_wready	O	Write ready. This signal indicates that the slave can accept the write data. 1 = slave ready 0 = slave not ready
saxigp3_bid	O	Response ID. The identification tag of the write response
saxigp3_bresp	O	Write response. This signal indicates the status of the write transaction. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
saxigp3_bvalid	O	Write response valid. This signal indicates that a valid write response is available. 1 = write response available 0 = write response not available
saxigp3_bready	I	Response ready. This signal indicates that the master can accept the response information. 1 = master ready 0 = master not ready
saxigp3_arid	I	Read address ID. This signal is the identification tag for the read address group of signals.
saxigp3_araddr	I	Read address. The read address bus gives the initial address of a read burst transaction.
saxigp3_arlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
saxigp3_arsize	I	Burst size. This signal indicates the size of each transfer in the burst.
saxigp3_arburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
saxigp3_arlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
saxigp3_arcache	I	Cache type. This signal provides additional information about the cacheable characteristics of the transfer.
saxigp3_arprot	I	Protection type. This signal provides protection unit information for the transaction.

Table B-49: **S_AXI_HP0_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_arvalid	I	Read address valid. This signal indicates, when High, that the read address and control information is valid and remains stable until the address acknowledge signal, ARREADY, is High.
saxigp3_arready	O	Read address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready
saxigp3_rid	O	Read ID tag. This signal is the ID tag of the read data group of signals.
saxigp3_rdata	O	Read data. The read data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
saxigp3_rresp	O	Read response. This signal indicates the status of the read transfer. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
saxigp3_rlast	O	Read last. This signal indicates the last transfer in a read burst.
saxigp3_rvalid	O	Read valid. This signal indicates that the required read data is available and the read transfer can complete.
saxigp3_rready	I	Read ready. This signal indicates that the master can accept the read data and response information. 1= master ready 0 = master not ready
saxigp3_awqos	O	Wr addr channel QOS input
saxigp3_arqos	O	Rd addr channel QOS input
saxigp3_rcount	O	Rd data channel fill level
saxigp3_wcount	O	Wr data channel fill level
saxigp3_racount	O	Rd addr channel fill level
saxigp3_wacount	O	Wr addr channel fill level

Table B-50: **S_AXI_HP0_FPD_ACLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
saxihp0_fpd_aclk	I	Input clock signal

Table B-51: **S_AXI_HP0_FPD_RCLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
saxihp0_fpd_rclk	I	Read clock signal

Table B-52: **S_AXI_HP0_FPD_WCLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals0
saxihp0_fpd_wclk	I	Write clock signal

Table B-53: **S_AXI_HP1_FPD**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
saxigp3_awuser	I	User-defined AW channel signals
saxigp3_awid	I	Write address ID. This signal is the identification tag for the write address group of signals.
saxigp3_awaddr	I	Write address. The write address bus gives the address of the first transfer in a write burst transaction. The associated control signals are used to determine the addresses of the remaining transfers in the burst.
saxigp3_awlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
saxigp3_awsize	I	Burst size. This signal indicates the size of each transfer in the burst. Byte lane strobes indicate exactly which byte lanes to update.
saxigp3_awburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
saxigp3_awlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
saxigp3_awcache	I	Cache type. This signal indicates the buffer able, cacheable, write-through, write-back, and allocate attributes of the transaction.
saxigp3_awprot	I	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.

Table B-53: **S_AXI_HP1_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_awvalid	I	Write address valid. This signal indicates that valid write address and control information are available. 1 = address and control information available 0 = address and control information not available The address and control information remain stable until the address acknowledge signal, AWREADY, goes High.
saxigp3_awready	O	Write address channel ready signal
saxigp3_wdata	I	Write data. The write data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
saxigp3_wstrb	I	Write strobes. This signal indicates which byte lanes to update in memory. There is one write strobe for each eight bits of the write data bus.
saxigp3_wlast	I	Write last. This signal indicates the last transfer in a write burst.
saxigp3_wvalid	I	Write valid. This signal indicates that valid write data and strobes are available. 1 = write data and strobes available 0 = write data and strobes not available
saxigp3_wready	O	Write ready. This signal indicates that the slave can accept the write data. 1 = slave ready 0 = slave not ready
saxigp3_bid	O	Response ID. The identification tag of the write response
saxigp3_bresp	O	Write response. This signal indicates the status of the write transaction. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
saxigp3_bvalid	O	Write response valid. This signal indicates that a valid write response is available. 1 = write response available 0 = write response not available
saxigp3_bready	I	Response ready. This signal indicates that the master can accept the response information. 1 = master ready 0 = master not ready
saxigp3_arid	I	Read address ID. This signal is the identification tag for the read address group of signals.
saxigp3_araddr	I	Read address. The read address bus gives the initial address of a read burst transaction.
saxigp3_arlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.

Table B-53: **S_AXI_HP1_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_arsize	I	Burst size. This signal indicates the size of each transfer in the burst
saxigp3_arburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
saxigp3_arlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
saxigp3_arcache	I	Cache type. This signal provides additional information about the cacheable characteristics of the transfer.
saxigp3_arprot	I	Protection type. This signal provides protection unit information for the transaction.
saxigp3_arvalid	I	Read address valid. This signal indicates, when High, that the read address and control information is valid and remains stable until the address acknowledge signal, ARREADY, is High.
saxigp3_arready	O	Read address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready
saxigp3_rid	O	Read ID tag. This signal is the ID tag of the read data group of signals.
saxigp3_rdata	O	Read data. The read data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
saxigp3_rresp	O	Read response. This signal indicates the status of the read transfer. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
saxigp3_rlast	O	Read last. This signal indicates the last transfer in a read burst.
saxigp3_rvalid	O	Read valid. This signal indicates that the required read data is available and the read transfer can complete.
saxigp3_rready	I	Read ready. This signal indicates that the master can accept the read data and response information. 1= master ready 0 = master not ready
saxigp3_awqos	O	Wr addr channel QOS input
saxigp3_arqos	O	Rd addr channel QOS input
saxigp3_rcount	O	Rd data channel fill level
saxigp3_wcount	O	Wr data channel fill level
saxigp3_racount	O	Rd addr channel fill level
saxigp3_wacount	O	Wr addr channel fill level

Table B-54: **S_AXI_HP1_FPD_ACLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
Saxihp1_fpd_aclk	I	Input clock signal

Table B-55: **S_AXI_HP1_FPD_RCLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
Saxihp1_fpd_rclk	I	Read clock signal

Table B-56: **S_AXI_HP1_FPD_WCLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
saxihp1_fpd_wclk	I	Write clock signal

Table B-57: **S_AXI_HP2_FPD**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
saxigp3_awuser	I	User-defined AW channel signals
saxigp3_awid	I	Write address ID. This signal is the identification tag for the write address group of signals.
saxigp3_awaddr	I	Write address. The write address bus gives the address of the first transfer in a write burst transaction. The associated control signals are used to determine the addresses of the remaining transfers in the burst.
saxigp3_awlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
saxigp3_awsize	I	Burst size. This signal indicates the size of each transfer in the burst. Byte lane strobes indicate exactly which byte lanes to update.

Table B-57: **S_AXI_HP2_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_awburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
saxigp3_awlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
saxigp3_awcache	I	Cache type. This signal indicates the buffer able, cacheable, write-through, write-back, and allocate attributes of the transaction.
saxigp3_awprot	I	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.
saxigp3_awvalid	I	Write address valid. This signal indicates that valid write address and control information are available. 1 = address and control information available 0 = address and control information not available The address and control information remain stable until the address acknowledge signal, AWREADY, goes High.
saxigp3_awready	O	Write address channel ready signal
saxigp3_wdata	I	Write data. The write data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
saxigp3_wstrb	I	Write strobes. This signal indicates which byte lanes to update in memory. There is one write strobe for each eight bits of the write data bus.
saxigp3_wlast	I	Write last. This signal indicates the last transfer in a write burst.
saxigp3_wvalid	I	Write valid. This signal indicates that valid write data and strobes are available. 1 = write data and strobes available 0 = write data and strobes not available
saxigp3_wready	O	Write ready. This signal indicates that the slave can accept the write data. 1 = slave ready 0 = slave not ready
saxigp3_bid	O	Response ID. The identification tag of the write response
saxigp3_bresp	O	Write response. This signal indicates the status of the write transaction. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
saxigp3_bvalid	O	Write response valid. This signal indicates that a valid write response is available. 1 = write response available 0 = write response not available

Table B-57: **S_AXI_HP2_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_bready	I	Response ready. This signal indicates that the master can accept the response information. 1 = master ready 0 = master not ready
saxigp3_arid	I	Read address ID. This signal is the identification tag for the read address group of signals.
saxigp3_araddr	I	Read address. The read address bus gives the initial address of a read burst transaction.
saxigp3_arlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
saxigp3_arsize	I	Burst size. This signal indicates the size of each transfer in the burst.
saxigp3_arburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
saxigp3_arlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
saxigp3_arcache	I	Cache type. This signal provides additional information about the cacheable characteristics of the transfer.
saxigp3_arprot	I	Protection type. This signal provides protection unit information for the transaction.
saxigp3_arvalid	I	Read address valid. This signal indicates, when High, that the read address and control information is valid and remains stable until the address acknowledge signal, ARREADY, is High.
saxigp3_arready	O	Read address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready
saxigp3_rid	O	Read ID tag. This signal is the ID tag of the read data group of signals.
saxigp3_rdata	O	Read data. The read data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
saxigp3_rresp	O	Read response. This signal indicates the status of the read transfer. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
saxigp3_rlast	O	Read last. This signal indicates the last transfer in a read burst.
saxigp3_rvalid	O	Read valid. This signal indicates that the required read data is available and the read transfer can complete.

Table B-57: **S_AXI_HP2_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_rready	I	Read ready. This signal indicates that the master can accept the read data and response information. 1= master ready 0 = master not ready
saxigp3_awqos	O	Wr addr channel QOS input
saxigp3_arqos	O	Rd addr channel QOS input
saxigp3_rcount	O	Rd data channel fill level
saxigp3_wcount	O	Wr data channel fill level
saxigp3_racount	O	Rd addr channel fill level
saxigp3_wacount	O	Wr addr channel fill level

Table B-58: **S_AXI_HP2_FPD_ACLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
Saxihp2_fpd_aclk	I	Input clock signal

Table B-59: **S_AXI_HP2_FPD_RCLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
Saxihp2_fpd_rclk	I	Read clock signal

Table B-60: **S_AXI_HP2_FPD_WCLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
Saxihp2_fpd_wclk	I	Write clock signal

Table B-61: **S_AXI_HP3_FPD**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
saxigp3_awuser	I	User-defined AW channel signals
saxigp3_awid	I	Write address ID. This signal is the identification tag for the write address group of signals.
saxigp3_awaddr	I	Write address. The write address bus gives the address of the first transfer in a write burst transaction. The associated control signals are used to determine the addresses of the remaining transfers in the burst.
saxigp3_awlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
saxigp3_awsize	I	Burst size. This signal indicates the size of each transfer in the burst. Byte lane strobes indicate exactly which byte lanes to update.
saxigp3_awburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
saxigp3_awlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
saxigp3_awcache	I	Cache type. This signal indicates the buffer able, cacheable, write-through, write-back, and allocate attributes of the transaction.
saxigp3_awprot	I	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.
saxigp3_awvalid	I	Write address valid. This signal indicates that valid write address and control information are available. 1 = address and control information available 0 = address and control information not available The address and control information remain stable until the address acknowledge signal, AWREADY, goes High.
saxigp3_awready	O	Write address channel ready signal
saxigp3_wdata	I	Write data. The write data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
saxigp3_wstrb	I	Write strobes. This signal indicates which byte lanes to update in memory. There is one write strobe for each eight bits of the write data bus.
saxigp3_wlast	I	Write last. This signal indicates the last transfer in a write burst.

Table B-61: **S_AXI_HP3_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_wvalid	I	Write valid. This signal indicates that valid write data and strobes are available. 1 = write data and strobes available 0 = write data and strobes not available
saxigp3_wready	O	Write ready. This signal indicates that the slave can accept the write data. 1 = slave ready 0 = slave not ready
saxigp3_bid	O	Response ID. The identification tag of the write response
saxigp3_bresp	O	Write response. This signal indicates the status of the write transaction. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
saxigp3_bvalid	O	Write response valid. This signal indicates that a valid write response is available. 1 = write response available 0 = write response not available
saxigp3_bready	I	Response ready. This signal indicates that the master can accept the response information. 1 = master ready 0 = master not ready
saxigp3_arid	I	Read address ID. This signal is the identification tag for the read address group of signals.
saxigp3_araddr	I	Read address. The read address bus gives the initial address of a read burst transaction.
saxigp3_arlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
saxigp3_arsize	I	Burst size. This signal indicates the size of each transfer in the burst.
saxigp3_arburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
saxigp3_arlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
saxigp3_arcache	I	Cache type. This signal provides additional information about the cacheable characteristics of the transfer.
saxigp3_arprot	I	Protection type. This signal provides protection unit information for the transaction.

Table B-61: **S_AXI_HP3_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_arvalid	I	Read address valid. This signal indicates, when High, that the read address and control information is valid and remains stable until the address acknowledge signal, ARREADY, is High.
saxigp3_arready	O	Read address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready
saxigp3_rid	O	Read ID tag. This signal is the ID tag of the read data group of signals.
saxigp3_rdata	O	Read data. The read data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
saxigp3_rresp	O	Read response. This signal indicates the status of the read transfer. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
saxigp3_rlast	O	Read last. This signal indicates the last transfer in a read burst.
saxigp3_rvalid	O	Read valid. This signal indicates that the required read data is available and the read transfer can complete.
saxigp3_rready	I	Read ready. This signal indicates that the master can accept the read data and response information. 1= master ready 0 = master not ready
saxigp3_awqos	O	Wr addr channel QOS input
saxigp3_arqos	O	Rd addr channel QOS input
saxigp3_rcount	O	Rd data channel fill level
saxigp3_wcount	O	Wr data channel fill level
saxigp3_racount	O	Rd addr channel fill level
saxigp3_wacount	O	Wr addr channel fill level

Table B-62: **S_AXI_HP3_FPD_ACLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
Saxihp3_fpd_aclk	I	Input clock signal

Table B-63: **S_AXI_HP3_FPD_RCLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
Saxihp3_fpd_rclk	I	Read clock signal

Table B-64: **S_AXI_HP1_FPD_WCLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp3_aruser	I	User-defined AR channel signals
Saxihp3_fpd_wclk	I	Write clock signal

Table B-65: **S_AXI_HPC0_FPD**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp0_aruser	I	User-defined AR channel signals
saxigp0_awuser	I	User-defined AW channel signals
saxigp0_awid	I	Write address ID. This signal is the identification tag for the write address group of signals.
saxigp0_awaddr	I	Write address. The write address bus gives the address of the first transfer in a write burst transaction. The associated control signals are used to determine the addresses of the remaining transfers in the burst.
saxigp0_awlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
saxigp0_awsize	I	Burst size. This signal indicates the size of each transfer in the burst. Byte lane strobes indicate exactly which byte lanes to update.
saxigp0_awburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
saxigp0_awlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
saxigp0_awcache	I	Cache type. This signal indicates the buffer able, cacheable, write-through, write-back, and allocate attributes of the transaction.

Table B-65: **S_AXI_HPC0_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp0_awprot	I	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.
saxigp0_awvalid	I	Write address valid. This signal indicates that valid write address and control information are available. 1 = address and control information available 0 = address and control information not available The address and control information remain stable until the address acknowledge signal, AWREADY, goes High.
saxigp0_awready	O	Write address channel ready signal
saxigp0_wdata	I	Write data. The write data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
saxigp0_wstrb	I	Write strobes. This signal indicates which byte lanes to update in memory. There is one write strobe for each eight bits of the write data bus.
saxigp0_wlast	I	Write last. This signal indicates the last transfer in a write burst.
saxigp0_wvalid	I	Write valid. This signal indicates that valid write data and strobes are available. 1 = write data and strobes available 0 = write data and strobes not available
saxigp0_wready	O	Write ready. This signal indicates that the slave can accept the write data. 1 = slave ready 0 = slave not ready
saxigp0_bid	O	Response ID. The identification tag of the write response
saxigp0_bresp	O	Write response. This signal indicates the status of the write transaction. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
saxigp0_bvalid	O	Write response valid. This signal indicates that a valid write response is available. 1 = write response available 0 = write response not available
saxigp0_bready	I	Response ready. This signal indicates that the master can accept the response information. 1 = master ready 0 = master not ready
saxigp0_arid	I	Read address ID. This signal is the identification tag for the read address group of signals.

Table B-65: **S_AXI_HPC0_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp0_araddr	I	Read address. The read address bus gives the initial address of a read burst transaction.
saxigp0_arlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
saxigp0_arsize	I	Burst size. This signal indicates the size of each transfer in the burst.
saxigp0_arburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
saxigp0_arlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
saxigp0_arcache	I	Cache type. This signal provides additional information about the cacheable characteristics of the transfer.
saxigp0_arprot	I	Protection type. This signal provides protection unit information for the transaction.
saxigp0_arvalid	I	Read address valid. This signal indicates, when High, that the read address and control information is valid and remains stable until the address acknowledge signal, ARREADY, is High.
saxigp0_arready	O	Read address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready.
saxigp0_rid	O	Read ID tag. This signal is the ID tag of the read data group of signals.
saxigp0_rdata	O	Read data. The read data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
saxigp0_rresp	O	Read response. This signal indicates the status of the read transfer. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
saxigp0_rlast	O	Read last. This signal indicates the last transfer in a read burst.
saxigp0_rvalid	O	Read valid. This signal indicates that the required read data is available and the read transfer can complete.
saxigp0_rready	I	Read ready. This signal indicates that the master can accept the read data and response information. 1= master ready 0 = master not ready
saxigp0_awqos	O	Wr addr channel QOS input
saxigp0_arqos	O	Rd addr channel QOS input

Table B-65: **S_AXI_HPC0_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp0_rcount	O	Rd data channel fill level
saxigp0_wcount	O	Wr data channel fill level
saxigp0_racount	O	Rd addr channel fill level
saxigp0_wacount	O	Wr addr channel fill level

Table B-66: **S_AXI_HPC0_FPD_ACLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp0_aruser	I	User-defined AR channel signals
saxihpc0_fpd_aclk	I	Input clock signal

Table B-67: **S_AXI_HPC0_FPD_RCLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp0_aruser	I	User-defined AR channel signals
saxihpc0_fpd_rclk	I	Read clock signal

Table B-68: **S_AXI_HPC0_FPD_WCLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp0_aruser	I	User-defined AR channel signals
saxihpc0_fpd_wclk	I	Write clock signal

Table B-69: **S_AXI_HPC1_FPD**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
Saxigp1_aruser	I	User-defined AR channel signals
Saxigp1_awuser	I	User-defined AW channel signals
Saxigp1_awid	I	Write address ID. This signal is the identification tag for the write address group of signals.

Table B-69: **S_AXI_HPC1_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
Saxigp1_awaddr	I	Write address. The write address bus gives the address of the first transfer in a write burst transaction. The associated control signals are used to determine the addresses of the remaining transfers in the burst.
Saxigp1_awlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
Saxigp1_awsize	I	Burst size. This signal indicates the size of each transfer in the burst. Byte lane strobes indicate exactly which byte lanes to update.
Saxigp1_awburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
Saxigp1_awlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
Saxigp1_awcache	I	Cache type. This signal indicates the buffer able, cacheable, write-through, write-back, and allocate attributes of the transaction.
Saxigp1_awprot	I	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.
Saxigp1_awvalid	I	Write address valid. This signal indicates that valid write address and control information are available. 1 = address and control information available 0 = address and control information not available The address and control information remain stable until the address acknowledge signal, AWREADY, goes High.
Saxigp1_awready	O	Write address channel ready signal
Saxigp1_wdata	I	Write data. The write data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
Saxigp1_wstrb	I	Write strobes. This signal indicates which byte lanes to update in memory. There is one write strobe for each eight bits of the write data bus.
Saxigp1_wlast	I	Write last. This signal indicates the last transfer in a write burst.
Saxigp1_wvalid	I	Write valid. This signal indicates that valid write data and strobes are available. 1 = write data and strobes available 0 = write data and strobes not available

Table B-69: **S_AXI_HPC1_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
Saxigp1_wready	O	Write ready. This signal indicates that the slave can accept the write data. 1 = slave ready 0 = slave not ready
Saxigp1_bid	O	Response ID. The identification tag of the write response
Saxigp1_bresp	O	Write response. This signal indicates the status of the write transaction. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
Saxigp1_bvalid	O	Write response valid. This signal indicates that a valid write response is available. 1 = write response available 0 = write response not available
Saxigp1_bready	I	Response ready. This signal indicates that the master can accept the response information. 1 = master ready 0 = master not ready
Saxigp1_arid	I	Read address ID. This signal is the identification tag for the read address group of signals.
Saxigp1_araddr	I	Read address. The read address bus gives the initial address of a read burst transaction.
Saxigp1_arlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
Saxigp1_arsize	I	Burst size. This signal indicates the size of each transfer in the burst.
Saxigp1_arburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
Saxigp1_arlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
Saxigp1_arcache	I	Cache type. This signal provides additional information about the cacheable characteristics of the transfer.
Saxigp1_arprot	I	Protection type. This signal provides protection unit information for the transaction.
Saxigp1_arvalid	I	Read address valid. This signal indicates, when High, that the read address and control information is valid and remains stable until the address acknowledge signal, ARREADY, is High.

Table B-69: **S_AXI_HPC1_FPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
Saxigp1_arready	O	Read address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready
Saxigp1_rid	O	Read ID tag. This signal is the ID tag of the read data group of signals.
Saxigp1_rdata	O	Read data. The read data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
Saxigp1_rresp	O	Read response. This signal indicates the status of the read transfer. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
Saxigp1_rlast	O	Read last. This signal indicates the last transfer in a read burst.
Saxigp1_rvalid	O	Read valid. This signal indicates that the required read data is available and the read transfer can complete.
Saxigp1_rready	I	Read ready. This signal indicates that the master can accept the read data and response information. 1= master ready 0 = master not ready
Saxigp1_awqos	O	Wr addr channel QOS input
Saxigp1_arqos	O	Rd addr channel QOS input
Saxigp1_rcount	O	Rd data channel fill level
Saxigp1_wcount	O	Wr data channel fill level
Saxigp1_racount	O	Rd addr channel fill level
Saxigp1_wacount	O	Wr addr channel fill level

Table B-70: **S_AXI_HPC1_FPD_ACLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
Saxigp1_aruser	I	User-defined AR channel signals
Saxihpc1_fpd_aclk	I	Input clock signal

Table B-71: **S_AXI_HPC1_FPD_RCLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
Saxigp1_aruser	I	User-defined AR channel signals
Saxihpc1_fpd_rclk	I	Read clock signal

Table B-72: **S_AXI_HPC1_FPD_WCLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
Saxigp1_aruser	I	User-defined AR channel signals
Saxihpc1_fpd_wclk	I	Write clock signal

Table B-73: **S_AXI_PL_LPD**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp2_aruser	I	User-defined AR channel signals
saxigp2_awuser	I	User-defined AW channel signals
saxigp2_awid	I	Write address ID. This signal is the identification tag for the write address group of signals.
saxigp2_awaddr	I	Write address. The write address bus gives the address of the first transfer in a write burst transaction. The associated control signals are used to determine the addresses of the remaining transfers in the burst.
saxigp2_awlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
saxigp2_awsize	I	Burst size. This signal indicates the size of each transfer in the burst. Byte lane strobes indicate exactly which byte lanes to update.
saxigp2_awburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
saxigp2_awlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
saxigp2_awcache	I	Cache type. This signal indicates the buffer able, cacheable, write-through, write-back, and allocate attributes of the transaction.

Table B-73: **S_AXI_PL_LPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp2_awprot	I	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.
saxigp2_awvalid	I	Write address valid. This signal indicates that valid write address and control information are available. 1 = address and control information available 0 = address and control information not available The address and control information remain stable until the address acknowledge signal, AWREADY, goes High.
saxigp2_awready	O	Write address channel ready signal
saxigp2_wdata	I	Write data. The write data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
saxigp2_wstrb	I	Write strobes. This signal indicates which byte lanes to update in memory. There is one write strobe for each eight bits of the write data bus.
saxigp2_wlast	I	Write last. This signal indicates the last transfer in a write burst.
saxigp2_wvalid	I	Write valid. This signal indicates that valid write data and strobes are available. 1 = write data and strobes available 0 = write data and strobes not available
saxigp2_wready	O	Write ready. This signal indicates that the slave can accept the write data. 1 = slave ready 0 = slave not ready
saxigp2_bid	O	Response ID. The identification tag of the write response
saxigp2_bresp	O	Write response. This signal indicates the status of the write transaction. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
saxigp2_bvalid	O	Write response valid. This signal indicates that a valid write response is available. 1 = write response available 0 = write response not available
saxigp2_bready	I	Response ready. This signal indicates that the master can accept the response information. 1 = master ready 0 = master not ready
saxigp2_arid	I	Read address ID. This signal is the identification tag for the read address group of signals.
saxigp2_araddr	I	Read address. The read address bus gives the initial address of a read burst transaction.

Table B-73: **S_AXI_PL_LPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp2_arlen	I	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
saxigp2_arsize	I	Burst size. This signal indicates the size of each transfer in the burst.
saxigp2_arburst	I	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
saxigp2_arlock	I	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
saxigp2_arcache	I	Cache type. This signal provides additional information about the cacheable characteristics of the transfer.
saxigp2_arprot	I	Protection type. This signal provides protection unit information for the transaction.
saxigp2_arvalid	I	Read address valid. This signal indicates, when High, that the read address and control information is valid and remains stable until the address acknowledge signal, ARREADY, is High.
saxigp2_arready	O	Read address ready. This signal indicates that the slave is ready to accept an address and associated control signals. 1 = slave ready 0 = slave not ready
saxigp2_rid	O	Read ID tag. This signal is the ID tag of the read data group of signals.
saxigp2_rdata	O	Read data. The read data bus can be 8, 16, 32, 64, 128, 256, 512, or 1,024 bits wide.
saxigp2_rresp	O	Read response. This signal indicates the status of the read transfer. The allowable responses are OKAY, EXOKAY, SLVERR, and DECERR.
saxigp2_rlast	O	Read last. This signal indicates the last transfer in a read burst.
saxigp2_rvalid	O	Read valid. This signal indicates that the required read data is available and the read transfer can complete.
saxigp2_rready	I	Read ready. This signal indicates that the master can accept the read data and response information. 1= master ready 0 = master not ready
saxigp2_awqos	O	Wr addr channel QOS input
saxigp2_arqos	O	Rd addr channel QOS input
saxigp2_rcount	O	Rd data channel fill level
saxigp2_wcount	O	Wr data channel fill level

Table B-73: **S_AXI_PL_LPD *(Cont'd)***

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp2_racount	O	Rd addr channel fill level
saxigp2_wacount	O	Wr addr channel fill level

Table B-74: **S_AXI_PL_LPD_ACLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp2_aruser	I	User-defined AR channel signals
saxipl_lpd_aclk	I	Input clock signal

Table B-75: **S_AXI_PL_LPD_RCLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp2_aruser	I	User-defined AR channel signals
saxipl_lpd_rclk	I	Read clock signal

Table B-76: **S_AXI_PL_LPD_WCLK**

Zynq Ultrascale + MPSoc PS I/O Name	I/O	Description
saxigp2_aruser	I	User-defined AR channel signals
saxipl_lpd_wclk	I	Write clock signal

Appendix C

User Parameters

Table C-1: **Vivado IDE Parameter to User Parameter Relationship**

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
Enable ADMA	PSU_USE_ADMA	0,1	0
Enable master AXI GPIO 0	PSU_USE_M_AXI_GP0	0,1	1
Data width of AXI GPIO 0	PSU_MAXIGP0_DATA_WIDTH	32,64,128	128
Enable master AXI GPIO 1	PSU_USE_M_AXI_GP1	0,1	0
Data width of AXI GPIO 1	PSU_MAXIGP1_DATA_WIDTH	32,64,128	128
Enable Master AXI GPIO 2	PSU_USE_M_AXI_GP2	0,1	0
Data width of AXI GPIO 2	PSU_MAXIGP2_DATA_WIDTH	32,64,128	128
Enable Slave AXI ACP	PSU_USE_S_AXI_ACP	0,1	0
Enable Slave AXI GPIO 0	PSU_USE_S_AXI_GP0	0,1	0
Data width of slave AXI GPIO 0	PSU_SAXIGP0_DATA_WIDTH	32,64,128	128
Enable Slave AXI GPIO 1	PSU_USE_S_AXI_GP1	0,1	0
Data width of slave AXI GPIO 1	PSU_SAXIGP1_DATA_WIDTH	32,64,128	128
Enable Slave AXI GPIO 2	PSU_USE_S_AXI_GP2	0,1	0
Data width of slave AXI GPIO 2	PSU_SAXIGP2_DATA_WIDTH	32,64,128	128
Enable Slave AXI GPIO 3	PSU_USE_S_AXI_GP3	0,1	0
Data width of slave AXI GPIO 3	PSU_SAXIGP3_DATA_WIDTH	32,64,128	128
Enable Slave AXI GPIO 4	PSU_USE_S_AXI_GP4	0,1	0
Data width of slave AXI GPIO 4	PSU_SAXIGP4_DATA_WIDTH	32,64,128	128
Enable Slave AXI GPIO 5	PSU_USE_S_AXI_GP5	0,1	0
Data width of slave AXI GPIO 5	PSU_SAXIGP5_DATA_WIDTH	32,64,128	128
Enable Slave AXI GPIO 6	PSU_USE_S_AXI_GP6	0,1	0
Data width of slave AXI GPIO 6	PSU_SAXIGP6_DATA_WIDTH	32,64,128	128
Enable slave AXI ACE	PSU_USE_S_AXI_ACE	0,1	0
Enable REMUS ports	PSU_USE_REMUS_PORTS	0,1	0
Enable Debug ports	PSU_USE_DEBUG_PORTS	0,1	0
Enable Audio ports	PSU_USE_AUDIO	0,1	0
Enable Video ports	PSU_USE_VIDEO	0,1	0

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
Enable Fabric Trace Module	PSU_USE_FTM	0,1	0
Enable Low Power DMA	PSU_USE_GDMA	0,1	0
Enable PL Interrupts	PSU_USE_IRQ	0,1	0
Enable PL clock 0	PSU_USE_CLK0	0,1	0
Enable PL clock 1	PSU_USE_CLK1	0,1	0
Enable PL clock 2	PSU_USE_CLK2	0,1	0
Enable PL clock 3	PSU_USE_CLK3	0,1	0
Enable PL reset 0	PSU_USE_RST0	0,1	0
Enable PL reset 1	PSU_USE_RST1	0,1	0
Enable PL reset 2	PSU_USE_RST2	0,1	0
Enable PL reset 3	PSU_USE_RST3	0,1	0
Enable	PSU_USE_RTC	0,1	0
Enable RPU IPI Interrupt	PSU_USE_EVENT_RPU	0,1	0
Enable CAN0 peripheral	PSU_CAN0_PERIPHERAL_ENABLE	0,1	0
Enable CAN0 peripheral IO	PSU_CAN0_PERIPHERAL_IO	MIO 2 .. 3 MIO 6 .. 7 MIO 10 .. 11 MIO 14 .. 15 MIO 18 .. 19 MIO 22 .. 23 MIO 26 .. 27 MIO 30 .. 31 MIO 34 .. 35 MIO 38 .. 39 MIO 42 .. 43 MIO 46 .. 47 MIO 50 .. 51 MIO 54 .. 55 MIO 58 .. 59 MIO 62 .. 63 MIO 66 .. 67 MIO 70 ..71 MIO 74 .. 75 EMIO	MIO 50.. 51
Enable clock for CAN 0	PSU_CAN0_GRP_CLK_ENABLE	0,1	0

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
MIO Pin for clock for CAN 0	PSU_CAN0_GRP_CLK_IO	MIO 0 to MIO 77	MIO 0
Enable CAN1 peripheral	PSU_CAN1_PERIPHERAL_ENABLE	0,1	0
Enable CAN1 peripheral IO	PSU_CAN1_PERIPHERAL_IO	MIO 0 .. 1 MIO 4 .. 5 MIO 8 .. 9 MIO12.. 13 MIO16.. 17 MIO20.. 21 MIO24.. 25 MIO28.. 29 MIO32.. 33 MIO36.. 37 MIO40.. 41 MIO44.. 45 MIO48.. 49 MIO52.. 53 MIO56.. 57 MIO60.. 61 MIO64.. 65 MIO68.. 69 MIO72.. 73 MIO76..77 EMIO	MIO28.. 29
Enable clock for CAN 1	PSU_CAN1_GRP_CLK_ENABLE	0,1	0
MIO Pin for clock for CAN 1	PSU_CAN1_GRP_CLK_IO	MIO 0 to MIO 77	MIO 0
Enable DPAUX peripheral IO	PSU_DPAUX_PERIPHERAL_IO	EMIO MIO 27..30 MIO 34..37	EMIO
Enable Display Port	PSU__DISPLAYPORT__PERIPHERAL__ENABLE	0,1	0
DP lane selection	PSU__DP__LANE_SEL	Dual Higher Dual Lower Single Higher Single Lower	Dual Higher
Enable Ethernet peripheral 0	PSU_ENET0_PERIPHERAL_ENABLE	0,1	0

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
Enable Ethernet peripheral IO	PSU_ENET0_PERIPHERAL_IO	MIO 26 ..37 EMIO	EMIO
Enable MDIO peripheral	PSU_ENET0_GRP_MDIO_ENABLE	0,1	0
Enable MDIO peripheral IO	PSU_ENET0_GRP_MDIO_IO	MIO 76 ..77	MIO 76 ..77
Enable Ethernet peripheral 1	PSU_ENET1_PERIPHERAL_ENABLE	0,1	0
Enable Ethernet peripheral IO 1	PSU_ENET1_PERIPHERAL_IO	MIO 38 .. 49 EMIO	EMIO
Enable MDIO peripheral	PSU_ENET1_GRP_MDIO_ENABLE	0,1	0
Enable MDIO peripheral IO	PSU_ENET1_GRP_MDIO_IO	MIO 76 .. 77	MIO 76 .. 77
Enable Ethernet peripheral 2	PSU_ENET2_PERIPHERAL_ENABLE	0,1	0
Enable Ethernet peripheral IO 2	PSU_ENET2_PERIPHERAL_IO	MIO 52 ..63 EMIO	MIO 52..63
Enable MDIO peripheral	PSU_ENET2_GRP_MDIO_ENABLE	0,1	0
Enable MDIO peripheral IO	PSU_ENET2_GRP_MDIO_IO	MIO 76 ..77	MIO 76 ..77
Enable Ethernet peripheral 3	PSU_ENET3_PERIPHERAL_ENABLE	0,1	0
Enable Ethernet peripheral IO 3	PSU_ENET3_PERIPHERAL_IO	MIO 64 ..75 EMIO	MIO 64 ..75
Enable MDIO peripheral	PSU_ENET3_GRP_MDIO_ENABLE	0,1	0
Enable MDIO peripheral IO	PSU_ENET3_GRP_MDIO_IO	MIO 76 ..77	MIO 76 ..77
Enable TSU peripheral	PSU_GEM_TSU_ENABLE	0,1	0
Enable TSU peripheral IO	PSU_GEM_TSU_IO	MIO 50 ..51	MIO 50 ..51
Enable GPIO mio 0 peripheral	PSU_GPIO0_MIO_PERIPHERAL_ENABLE	0,1	0
Enable GPIO mio IO peripheral	PSU_GPIO0_MIO_IO	MIO 0 ..31	MIO 0 ..31
Enable GPIO emio 0 peripheral	PSU_GPIO0_EMIO_PERIPHERAL_ENABLE	0,1	0
Enable GPIO mio 1 peripheral	PSU_GPIO1_MIO_PERIPHERAL_ENABLE	0,1	0
Enable GPIO mio IO peripheral	PSU_GPIO1_MIO_IO	MIO 32 ..63	MIO 32 ..63
Enable GPIO emio 1 peripheral	PSU_GPIO1_EMIO_PERIPHERAL_ENABLE	0,1	0
Enable GPIO mio 2 peripheral	PSU_GPIO2_MIO_PERIPHERAL_ENABLE	0,1	0

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
Enable GPIO mio IO peripheral	PSU_GPIO2_MIO_IO	MIO 64 ..77	MIO 64 ..77
Enable GPIO emio 2 peripheral	PSU_GPIO2_EMIO_PERIPHERAL_ENABLE	0,1	0
Enable I2C0 peripheral	PSU_I2C0_PERIPHERAL_ENABLE	0,1	0
Enable I2C0 peripheral I/O	PSU_I2C0_PERIPHERAL_IO	MIO 2 .. 3 MIO 6 .. 7 MIO 10 .. 11 MIO 14 .. 15 MIO 18 .. 19 MIO 22 .. 23 MIO 26 .. 27 MIO 30 .. 31 MIO 34 .. 35 MIO 38 .. 39 MIO 42 .. 43 MIO 46 .. 47 MIO 50 .. 51 MIO 54 .. 55 MIO 58 .. 59 MIO 62 .. 63 MIO 66 .. 67 MIO 70 ..71 MIO 74 .. 75 EMIO	MIO 2 .. 3
Enable I2C0 interrupt	PSU_I2C0_GRP_INT_ENABLE	0,1	0
Enable I2C0 interrupt I/O	PSU_I2C0_GRP_INT_IO	From MIO 1 to MIO 77	MIO 1
Enable I2C1 peripheral	PSU_I2C1_PERIPHERAL_ENABLE	0,1	0

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
Enable I2C1 peripheral I/O	PSU_I2C1_PERIPHEARL_IO	MIO 0 .. 1 MIO 4 .. 5 MIO 8 .. 9 MIO12.. 13 MIO16.. 17 MIO20.. 21 MIO24.. 25 MIO28.. 29 MIO32.. 33 MIO36.. 37 MIO40.. 41 MIO44.. 45 MIO48.. 49 MIO52.. 53 MIO56.. 57 MIO60.. 61 MIO64.. 65 MIO68.. 69 MIO72.. 73 MIO76..77 EMIO	MIO 0 .. 1
Enable I2C0 interrupt	PSU_I2C1_GRP_INT_ENABLE	0,1	0
Enable I2C0 interrupt I/O	PSU_I2C1_GRP_INT_IO	From MIO 1 to MIO 77	MIO 1
Enable NAND peripheral	PSU_NAND_PERIPHERAL_ENABLE	0,1	0
Enable NAND peripheral I/O	PSU_NAND_NAND_IO	MIO 9 12 .. 26 32	MIO 9 12 .. 26 32
Enable Readybusy	PSU_NAND_READY_BUSY_ENABLE	0,1	0
Enable Readybusy I/O	PSU_NAND_READY_BUSY_IO	MIO 10..11 MIO 27 ..28	MIO 10..11
Enable PJTAG peripheral	PSU_PJTAG_PERIPHERAL_ENABLE	0,1	0
Enable PJTAG peripheral I/O	PSU_PJTAG_PERIPHERAL_IO	MIO 0 .. 3 MIO 12 .. 15 MIO 26 .. 29 MIO 38 .. 41 MIO 52 .. 55 MIO 58 ..61	MIO 0 .. 3

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
Enable PMU peripheral	PSU_PMU_PERIPHERAL_ENABLE	0,1	0
Enable PMU peripheral I/O	PSU_PMU_PERIPHERAL_IO	MIO 64 ..75	MIO 64 ..75
Enable QSPI peripheral	PSU_QSPI_PERIPHERAL_ENABLE	0,1	0
Enable QSPI feedback clk	PSU_QSPI_GRP_FBCLK_ENABLE	0,1	0
QSPI feedback clk I/O	PSU_QSPI_GRP_FBCLK_IO	MIO 6	MIO 6
Enable QSPI single	PSU_QSPI_SINGLE_ENABLE	0,1	0
QSPI single I/O	PSU_QSPI_SINGLE_IO	MIO 0 ..5	MIO 0 ..5
QSPI single mode	PSU_QSPI_SINGLE_MODE	X1,X2,X4	X1
Enable QSPI dual stacked	PSU_QSPI_DUAL_STACKED_ENABLE	0,1	0
QSPI dual stacked I/O	PSU_QSPI_DUAL_STACKED_IO	MIO 0 ..7	MIO 0 ..7
QSPI dual stacked mode	PSU_QSPI_DUAL_STACKED_MODE	X1,X2,X4	X1
Enable QSPI dual parallel	PSU_QSPI_DUAL_PARALLEL_ENABLE	0,1	0
QSPI dual parallel I/O	PSU_QSPI_DUAL_PARALLEL_IO	MIO 0 ..12	MIO 0..12
QSPI dual parallel mode	PSU_QSPI_DUAL_PARALLEL_MODE	X1,X2,X4	X1
Enable SD0 peripheral	PSU_SD0_PERIPHERAL_ENABLE	0,1	0
SD0 peripheral I/O	PSU_SD0_PERIPHERAL_IO	MIO 13 .. 22 MIO 38 .. 48 MIO 38 .. 44 MIO 64 .. 70 MIO 64 .. 74 EMIO	MIO 13 .. 22
SD0 speed mode	PSU_SD0_SPEED_MODE	High speed, default speed	High Speed
Enable GRP CD	PSU_SD0_GRP_CD_ENABLE	0,1	0
GRP CD I/O	PSU_SD0_GRP_CD_IO	MIO 24, MIO 39,MIO 65	MIO 39
Enable GRP power	PSU_SD0_GRP_POW_ENABLE	0,1	0
GRP power I/O	PSU_SD0_GRP_POW_IO	MIO 25, MIO 50,MIO 76	MIO 50
GRP wp enable	PSU_SD0_GRP_WP_ENABLE	0,1	0
GRP wp I/O	PSU_SD0_GRP_WP_IO	MIO 23, MIO 49,MIO 75	MIO 23
Enable SD1 peripheral	PSU_SD1_PERIPHERAL_ENABLE	0,1	0

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
SD1 peripheral I/O	PSU_SD1_PERIPHERAL_IO	MIO 39..51 MIO 46..51 MIO 71..76	MIO 39..51
SD1 speed mode	PSU_SD1_SPEED_MODE	High speed, default speed	High Speed
Enable GRP CD	PSU_SD1_GRP_CD_ENABLE	0,1	0
GRP CD I/O	PSU_SD1_GRP_CD_IO	MIO 45, MIO 77	MIO 45
Enable GRP power	PSU_SD1_GRP_POW_ENABLE	0,1	0
GRP power I/O	PSU_SD1_GRP_POW_IO	MIO 43	MIO 43
GRP wp enable	PSU_SD1_GRP_WP_ENABLE	0,1	0
GRP wp I/O	PSU_SD1_GRP_WP_IO	MIO 44	MIO 44
Enable SPI0 peripheral	PSU_SPI0_PERIPHERAL_ENABLE	0,1	0
SPI0 peripheral I/O	PSU_SPI0_PERIPHERAL_IO	MIO 0 .. 5 MIO 12 .. 17 MIO 26 .. 31 MIO 38 .. 43 MIO 52 .. 57 MIO 64 .. 69	MIO 12 .. 17
Enable SPI0 GRP	PSU_SPI0_GRP_SS0_ENABLE	0,1	0
SPI0 GRP I/O	PSU_SPI0_GRP_SS0_IO	MIO 3 MIO 15 MIO 29 MIO 41 MIO 55 MIO 67	MIO 3
Enable SPI0 GRP	PSU_SPI0_GRP_SS1_ENABLE	0,1	0
SPI0 GRP SS1 I/O	PSU_SPI0_GRP_SS1_IO	MIO 2 MIO 14 MIO 28 MIO 40 MIO 54 MIO 66	MIO 14
Enable SPI0 GRP	PSU_SPI0_GRP_SS2_ENABLE	0,1	0

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
SPI0 GRP SS2 I/O	PSU_SPI0_GRP_SS2_IO	MIO 1 MIO 13 MIO 27 MIO 39 MIO 53 MIO 65	MIO 13
Enable SPI1 peripheral	PSU_SPI1_PERIPHERAL_ENABLE	0,1	0
SPI1 peripheral I/O	PSU_SPI1_PERIPHERAL_IO	MIO 6 .. 11 MIO 18 .. 23 MIO 32 .. 37 MIO 44 .. 49 MIO 58 .. 63 MIO 70 .. 75	MIO 6 .. 11
Enable SPI1 GRP	PSU_SPI1_GRP_SS0_ENABLE	0,1	0
SPI1 GRP SS0 I/O	PSU_SPI1_GRP_SS0_IO	MIO 9 MIO 21 MIO 35 MIO 47 MIO 61 MIO 73	MIO 9
Enable SPI1 GRP	PSU_SPI1_GRP_SS1_ENABLE	0,1	0
SPI1 GRP SS1 I/O	PSU_SPI1_GRP_SS1_IO	MIO 8 MIO 20 MIO 34 MIO 46 MIO 60 MIO 72	MIO 8
Enable SPI1 GRP	PSU_SPI1_GRP_SS2_ENABLE	0,1	0
SPI1 GRP SS2 I/O	PSU_SPI1_GRP_SS2_IO	MIO 1 MIO 13 MIO 33 MIO 40 MIO 59 MIO 71	MIO 33
Enable SWDT0 peripheral	PSU_SWDT0_PERIPHERAL_ENABLE	0,1	0

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
SWDT0 I/O	PSU_SWDT0_PERIPHERAL_IO	MIO 6 .. 7 MIO 10 .. 11 MIO 18 .. 19 MIO 22 .. 23 MIO 30 .. 31 MIO 34 .. 35 MIO 42 .. 43 MIO 46 .. 47 MIO 50 .. 51 MIO 62 .. 63 MIO 66 .. 67 MIO 70 .. 71 MIO 74 .. 75 EMIO	EMIO
Enable SWDT1 peripheral	PSU_SWDT1_PERIPHERAL_ENABLE	0,1	0
SWDT1 I/O	PSU_SWDT1_PERIPHERL_IO	MIO 4 .. 5 MIO 8 .. 9 MIO 16 .. 17 MIO 20 .. 21 MIO 24 .. 25 MIO 32 .. 33 MIO 36 .. 37 MIO 44 .. 45 MIO 48 .. 49 MIO 56 .. 57 MIO 64 .. 65 MIO 68 .. 69 MIO 72 .. 73 EMIO	MIO 4 .. 5
Enable peripheral test scan	PSU_TESTSCAN_PERIPHERAL_ENABLE	0,1	0
IO selection for peripheral test scan	PSU_TESTSCAN_PERIPHERAL_IO	MIO 0 .. 37	MIO 0 ..37
Enable Trace peripheral	PSU_TRACE_PERIPHERAL_ENABLE	0,1	0
TRACE I/O	PSU_TRACE_TRACE_IO	MIO 0 .. 17 MIO 26 .. 43 MIO 52 .. 69	MIO 52 .. 69
Enable TTC0 peripheral	PSU_TTC0_PERIPHERAL_ENABLE	0,1	0

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
TTC0 I/O	PSU_TTC0_PERIPHERAL_IO	MIO 6 .. 7 MIO 14 .. 15 MIO 22 .. 23 MIO 30 .. 31 MIO 38 .. 39 MIO 46 .. 47 MIO 54 .. 55 MIO 62 .. 63 MIO 70 .. 71 EMIO	MIO 6 .. 7
Enable TTC1 peripheral	PSU_TTC1_PERIPHERAL_ENABLE	0,1	0
TTC1 I/O	PSU_TTC1_PERIPHERAL_IO	MIO 4 .. 5 MIO 12 .. 13 MIO 20 .. 21 MIO 28 .. 29 MIO 36 .. 37 MIO 44 .. 45 MIO 52 .. 53 MIO 60 .. 61 MIO 68 .. 69 EMIO	MIO 12 .. 13
Enable TTC2 peripheral	PSU_TTC2_PERIPHERAL_ENABLE	0,1	0
TTC2 I/O	PSU_TTC2_PERIPHERAL_IO	MIO 2 .. 3 MIO 10 .. 11 MIO 18 .. 19 MIO 26 .. 27 MIO 34 .. 35 MIO 42 .. 43 MIO 50 .. 51 MIO 58 .. 59 MIO 66 .. 67 EMIO	MIO 18 .. 19
Enable TTC3 peripheral	PSU_TTC3_PERIPHERAL_ENABLE	0,1	0

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
TTC3 I/O	PSU_TTC3_PERIPHERAL_IO	MIO 0 .. 1 MIO 8 .. 9 MIO 16 .. 17 MIO 24 .. 25 MIO 32 .. 33 MIO 40 .. 41 MIO 48 .. 49 MIO 56 .. 57 MIO 64 .. 65 EMIO	MIO 0 .. 1
Enable UART0 peripheral	PSU_UART0_PERIPHERAL_ENABLE	0,1	0
UART0 I/O	PSU_UART0_PERIPHERAL_IO	MIO 2 .. 3 MIO 6 .. 7 MIO 10 .. 11 MIO 14 .. 15 MIO 18 .. 19 MIO 22 .. 23 MIO 26 .. 27 MIO 30 .. 31 MIO 34 .. 35 MIO 38 .. 39 MIO 42 .. 43 MIO 46 .. 47 MIO 50 .. 51 MIO 54 .. 55 MIO 58 .. 59 MIO 62 .. 63 MIO 66 .. 67 MIO 70 .. 71 MIO 74 .. 75 EMIO	MIO 6 .. 7
Enable UART1 peripheral	PSU_UART1_PERIPHERAL_ENABLE	0,1	0

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
UART1 I/O	PSU_UART1_PERIPHERAL_IO	MIO 0 .. 1 MIO 4 .. 5 MIO 8 .. 9 MIO 12 .. 13 MIO 16 .. 17 MIO 20 .. 21 MIO 24 .. 25 MIO 28 .. 29 MIO 32 .. 33 MIO 36 .. 37 MIO 40 .. 41 MIO 44 .. 45 MIO 48 .. 49 MIO 52 .. 53 MIO 56 .. 57 MIO 60 .. 61 MIO 64 .. 65 MIO 68 .. 69 MIO 72 .. 73 MIO 76 .. 77 EMIO	MIO 0 .. 1
Enable USB0 peripheral	PSU_USB0_PERIPHERAL_ENABLE	0,1	0
USB0 I/O	PSU_USB0_PERIPHERAL_IO	MIO 52 ..63	MIO 52 ..63
USB1 I/O	PSU_USB1_PERIPHERAL_IO	MIO 64 ..75	MIO 64 ..75
Enable USB0 peripheral	PSU_USB1_PERIPHERAL_ENABLE	0,1	0
Enable PCIE peripheral	PSU_PCIE_PERIPHERAL_ENABLE	0,1	0
PCIE I/O	PSU_PCIE_PERIPHERAL_IO	MIO 24 MIO 25 MIO 50 MIO 51	MIO 24
Enable CSU peripheral	PSU_CSU_PERIPHERAL_ENABLE	0,1	0
CSU IO	PSU_CSU_PERIPHERAL_IO	MIO 18 .. 25 MIO 44 .. 51	MIO 18 .. 25
Enable OCM main	PSU_CRL_APB_OCM_MAIN_ENABLE	0,1	0
Enable R5	PSU_CPU_R5_PERIPHERAL_ENABLE	0,1	0

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
Enable IOU Switch	PSU_CRL_APB_IOU_SWITCH_ENABLE	0,1	0
Enable LPD Switch	PSU_CRL_APB_LPD_SWITCH_ENABLE	0,1	0
Enable LPD LSBUS	PSU_CRL_APB_LPD_LSBUS_ENABLE	0,1	0
Enable TIMESTAMP	PSU_CRL_APB_TIMESTAMP_ENABLE	0,1	0
Enable AFI6	PSU_CRL_APB_AFI6_ENABLE	0,1	0
Enable USB3	PSU_CRL_APB_USB3_ENABLE	0,1	0
Enable PCAP	PSU_CRL_APB_PCAP_ENABLE	0,1	0
Enable LPD Debug	PSU_CRL_APB_DBG_LPD_ENABLE	0,1	0
Enable ADMA	PSU_CRL_APB_ADMA_ENABLE	0,1	0
Enable PL0 Clock	PSU_FPGA_PL0_ENABLE	0,1	0
Enable PL1 Clock	PSU_FPGA_PL1_ENABLE	0,1	0
Enable PL2 Clock	PSU_FPGA_PL2_ENABLE	0,1	0
Enable PL3 Clock	PSU_FPGA_PL3_ENABLE	0,1	0
Enable AMS	PSU_CRL_APB_AMS_ENABLE	0,1	0
Enable ACPU	PSU_CRF_APB_ACPU_ENABLE	0,1	0
Enable Debug Trace	PSU_CRF_APB_DBG_TRACE_ENABLE	0,1	0
Enable FPB Debug	PSU_CRF_APB_DBG_FPD_ENABLE	0,1	0
Enable AFI1	PSU_CRF_APB_AFI1_REF_ENABLE	0,1	0
Enable AFI2	PSU_CRF_APB_AFI2_REF_ENABLE	0,1	0
Enable AFI3	PSU_CRF_APB_AFI3_REF_ENABLE	0,1	0
Enable AFI4	PSU_CRF_APB_AFI4_REF_ENABLE	0,1	0
Enable AFI5	PSU_CRF_APB_AFI5_REF_ENABLE	0,1	0
Enable SATA	PSU_SATA_PERIPHERAL_ENABLE	0,1	0
Enable DDR Controller	PSU_DDRC_ENABLE	0,1	0
Enable GPU	PSU_CRF_APB_GPU_ENABLE	0,1	0
Enable GDMA	PSU_CRF_APB_GDMA_ENABLE	0,1	0
Enable DPDMA	PSU_CRF_APB_DPDMA_ENABLE	0,1	0
Enable main top switch	PSU_CRF_APB_TOPSW_MAIN_ENABLE	0,1	0
Enable top switch LS BUS	PSU_CRF_APB_TOPSW_LSBUS_ENABLE	0,1	0
Enable GT Reference Clock	PSU_CRF_APB_GTGREF0_ENABLE	0,1	0
	PSU_CRF_APB_DBG_TSTMP_ENABLE	0,1	0
0 – Basic/Auto clocking calculation 1 – Advanced Clocking Options	PSU_OVERRIDE_BASIC_CLOCK	0,1	0

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
Requested frequency	PSU_CRL_APB_TIMESTAMP_REF_CTRL_FREQMHZ	0 to 100	100
Requested frequency	PSU_CRL_APB_USB3_REF_CTRL_FREQMHZ	0 to 250	250
Requested frequency	PSU_CRL_APB_AFI6_REF_CTRL_FREQMHZ	0 to 600	500
Requested frequency	PSU_CRL_APB_PICDEBUG_TEMP_CTRL_FREQMHZ	0 to 1300	1300
Requested frequency	PSU_CRL_APB_IOPLL_TO_FPD_CTRL_FREQMHZ	0 to 534	500
Requested frequency	PSU_CRL_APB_RPLL_TO_FPD_CTRL_FREQMHZ	0 to 534	500
Requested frequency	PSU_CRF_APB_APLL_TO_LPD_CTRL_FREQMHZ	0 to 534	500
Requested frequency	PSU_CRF_APB_VPLL_TO_LPD_CTRL_FREQMHZ	0 to 534	500
Requested frequency	PSU_CRF_APB_DPLL_TO_LPD_CTRL_FREQMHZ	0 to 534	500
Requested frequency	PSU_CRF_APB_ACPU_CTRL_FREQMHZ	0 to 1300	1200
Requested frequency	PSU_CRF_APB_DBG_TRACE_CTRL_FREQMHZ	0 to 267	250
Requested frequency	PSU_CRL_APB_AMS_REF_CTRL_FREQMHZ	0 to 50	50
Requested frequency	PSU_CRF_APB_DBG_FPD_CTRL_FREQMHZ	0 to 267	250
Requested frequency	PSU_CRF_APB_DBG_TSTMP_CTRL_FREQMHZ	0 to 334	250
Requested frequency	PSU_CRF_APB_GTGREF0_REF_CTRL_FREQMHZ	0 to 125	125
Requested frequency	PSU_CRF_APB_DP_VIDEO_REF_CTRL_FREQMHZ	0 to 320	320
Requested frequency	PSU_CRF_APB_DP_AUDIO_REF_CTRL_FREQMHZ	0 to 25	25
Requested frequency	PSU_CRF_APB_DP_STC_REF_CTRL_FREQMHZ	0 to 27	27
Requested frequency	PSU_CRF_APB_DDR_CTRL_FREQMHZ	0 to 667	667
Requested frequency	PSU_CRF_APB_GPU_REF_CTRL_FREQMHZ	0 to 534	500
Requested frequency	PSU_CRF_APB_AFI0_REF_CTRL_FREQMHZ	0 to 667	667
Requested frequency	PSU_CRF_APB_AFI1_REF_CTRL_FREQMHZ	0 to 667	667
Requested frequency	PSU_CRF_APB_AFI2_REF_CTRL_FREQMHZ	0 to 667	667
Requested frequency	PSU_CRF_APB_AFI3_REF_CTRL_FREQMHZ	0 to 667	667
Requested frequency	PSU_CRF_APB_AFI4_REF_CTRL_FREQMHZ	0 to 667	667
Requested frequency	PSU_CRF_APB_AFI5_REF_CTRL_FREQMHZ	0 to 667	667
Requested frequency	PSU_CRF_APB_SATA_REF_CTRL_FREQMHZ	0 to 250	250
Requested frequency	PSU_CRF_APB_PCIE_REF_CTRL_FREQMHZ	0 to 250	250
Requested frequency	PSU_CRL_APB_PL0_REF_CTRL_FREQMHZ	0 to 400	400
Requested frequency	PSU_CRL_APB_PL1_REF_CTRL_FREQMHZ	0 to 400	400
Requested frequency	PSU_CRL_APB_PL2_REF_CTRL_FREQMHZ	0 to 400	400
Requested frequency	PSU_CRL_APB_PL3_REF_CTRL_FREQMHZ	0 to 400	400
Requested frequency	PSU_CRF_APB_GDMA_REF_CTRL_FREQMHZ	0 to 667	667
Requested frequency	PSU_CRF_APB_DPDMA_REF_CTRL_FREQMHZ	0 to 667	667

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
Requested frequency	PSU_CRF_APB_TOPSW_MAIN_CTRL_FREQMHZ	0 to 667	667
Requested frequency	PSU_CRF_APB_TOPSW_LSBUS_CTRL_FREQMHZ	0 to 100	100
Requested frequency	PSU_CRF_APB_DFT300_REF_CTRL_FREQMHZ	0 to 300	300
Requested frequency	PSU_CRF_APB_DFT270_REF_CTRL_FREQMHZ	0 to 270	270
Requested frequency	PSU_CRF_APB_DFT250_REF_CTRL_FREQMHZ	0 to 250	250
Requested frequency	PSU_CRF_APB_DFT125_REF_CTRL_FREQMHZ	0 to 125	125
Requested frequency	PSU_CRL_APB_GEM0_REF_CTRL_FREQMHZ	0 to 125	125
Requested frequency	PSU_CRL_APB_GEM1_REF_CTRL_FREQMHZ	0 to 125	125
Requested frequency	PSU_CRL_APB_GEM2_REF_CTRL_FREQMHZ	0 to 125	125
Requested frequency	PSU_CRL_APB_GEM3_REF_CTRL_FREQMHZ	0 to 125	125
Requested frequency	PSU_CRL_APB_GEM_TSU_REF_CTRL_FREQMHZ	0 to 400	400
Requested frequency	PSU_CRL_APB_USB0_BUS_REF_CTRL_FREQMHZ	0 to 250	250
Requested frequency	PSU_CRL_APB_USB1_BUS_REF_CTRL_FREQMHZ	0 to 250	250
Requested frequency	PSU_CRL_APB_QSPI_REF_CTRL_FREQMHZ	0 to 300	300
Requested frequency	PSU_CRL_APB_SDIO0_REF_CTRL_FREQMHZ	0 to 215	200
Requested frequency	PSU_CRL_APB_SDIO1_REF_CTRL_FREQMHZ	0 to 215	200
Requested frequency	PSU_CRL_APB_UART0_REF_CTRL_FREQMHZ	0 to 100	100
Requested frequency	PSU_CRL_APB_UART1_REF_CTRL_FREQMHZ	0 to 100	100
Requested frequency	PSU_CRL_APB_I2C0_REF_CTRL_FREQMHZ	0 to 100	100
Requested frequency	PSU_CRL_APB_I2C1_REF_CTRL_FREQMHZ	0 to 100	100
Requested frequency	PSU_CRL_APB_SPI0_REF_CTRL_FREQMHZ	0 to 215	214
Requested frequency	PSU_CRL_APB_SPI1_REF_CTRL_FREQMHZ	0 to 215	214
Requested frequency	PSU_CRL_APB_CAN0_REF_CTRL_FREQMHZ	0 to 100	100
Requested frequency	PSU_CRL_APB_CAN1_REF_CTRL_FREQMHZ	0 to 100	100
Requested frequency	PSU_CRL_APB_DEBUG_R5_ATCLK_CTRL_FREQMHZ	0 to 1000	1000
Requested frequency	PSU_CRL_APB_CPU_R5_CTRL_FREQMHZ	0 to 600	500
Requested frequency	PSU_CRL_APB_OCM_MAIN_CTRL_FREQMHZ	0 to 600	500
Requested frequency	PSU_CRL_APB_IOU_SWITCH_CTRL_FREQMHZ	0 to 267	267
Requested frequency	PSU_CRL_APB_CSU_PLL_CTRL_FREQMHZ	0 to 600	500
Requested frequency	PSU_CRL_APB_PCAP_CTRL_FREQMHZ	0 to 250	250
Requested frequency	PSU_CRL_APB_LPD_LSBUS_CTRL_FREQMHZ	0 to 100	100
Requested frequency	PSU_CRL_APB_LPD_SWITCH_CTRL_FREQMHZ	0 to 600	500
Requested frequency	PSU_CRL_APB_DBG_LPD_CTRL_FREQMHZ	0 to 267	250
Requested frequency	PSU_CRL_APB_NAND_REF_CTRL_FREQMHZ	0 to 100	100

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
Requested frequency	PSU_CRL_APB_ADMA_REF_CTRL_FREQMHZ	0 to 500	500
Requested frequency	PSU_CRL_APB_DLL_REF_CTRL_FREQMHZ	0 to 1500	1500
Requested frequency	PSU_CRL_APB_PICDEBUG_REF_CTRL_FREQMHZ	0 to 1300	1300
Requested frequency	PSU_CRL_APB_PICDEBUG_CTRL_FREQMHZ	0 to 1300	1300
Requested frequency	PSU_CRL_APB_PICDEBUG_TEMP_CTRL_FREQMHZ	0 to 1300	1300
Enabled Fractional PLL mode for floating frequency	PSU_CRF_APB_APLL_FRAC_CFG_ENABLED	0,1	0
Same as above	PSU_CRF_APB_VPLL_FRAC_CFG_ENABLED	0,1	0
Same as above	PSU_CRF_APB_DPLL_FRAC_CFG_ENABLED	0,1	0
Same as above	PSU_CRL_APB_IOPLL_FRAC_CFG_ENABLED	0,1	0
Same as above	PSU_CRL_APB_RPLL_FRAC_CFG_ENABLED	0,1	0
The integer portion of the feedback divider to the PLL	PSU_CRF_APB_APLL_CTRL_FBDIV	0 to 127	36
Same as above	PSU_CRF_APB_VPLL_CTRL_FBDIV		48
Same as above	PSU_CRF_APB_DPLL_CTRL_FBDIV		40
Same as above	PSU_CRL_APB_IOPLL_CTRL_FBDIV		45
Same as above	PSU_CRL_APB_RPLL_CTRL_FBDIV		48

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
	PSU_CRF_APB_APLL_TO_LPD_CTRL_DIVISOR0 PSU_CRF_APB_DPLL_TO_LPD_CTRL_DIVISOR0 PSU_CRF_APB_VPLL_TO_LPD_CTRL_DIVISOR0 PSU_CRF_APB_ACPU_CTRL_DIVISOR0 PSU_CRF_APB_DBG_TRACE_CTRL_DIVISOR0 PSU_CRF_APB_DBG_FPD_CTRL_DIVISOR0 PSU_CRF_APB_APM_CTRL_DIVISOR0, PSU_CRF_APB_DP_VIDEO_REF_CTRL_DIVISOR0 PSU_CRF_APB_DP_VIDEO_REF_CTRL_DIVISOR1, PSU_CRF_APB_DP_AUDIO_REF_CTRL_DIVISOR0 PSU_CRF_APB_DP_AUDIO_REF_CTRL_DIVISOR1, PSU_CRF_APB_DP_STC_REF_CTRL_DIVISOR0 PSU_CRF_APB_DP_STC_REF_CTRL_DIVISOR1 PSU_CRF_APB_DDR_CTRL_DIVISOR0 PSU_CRF_APB_GPU_REF_CTRL_DIVISOR0 PSU_CRF_APB_GPU_ENABLE PSU_CRF_APB_AFI0_REF_CTRL_DIVISOR0 PSU_CRF_APB_AFI0_REF_ENABLE PSU_CRF_APB_AFI1_REF_CTRL_DIVISOR0 PSU_CRF_APB_AFI1_REF_ENABLE PSU_CRF_APB_AFI2_REF_CTRL_DIVISOR0 PSU_CRF_APB_AFI2_REF_ENABLE		
	PSU_CRL_APB_TIMESTAMP_REF_CTRL_DIVISOR0 PSU_CRL_APB_TIMESTAMP_ENABLE PSU_CRL_APB_AFI6_REF_CTRL_DIVISOR0 PSU_CRL_APB_AFI6_ENABLE PSU_CRL_APB_USB3_DUAL_REF_CTRL_DIVISOR0 PSU_CRL_APB_USB3_DUAL_REF_CTRL_DIVISOR1 PSU_CRL_APB_USB3_ENABLE PSU_CRF_APB_GDMA_REF_CTRL_DIVISOR0 PSU_CRF_APB_GDMA_ENABLE PSU_CRF_APB_DPDMA_REF_CTRL_DIVISOR0 PSU_CRF_APB_DPDMA_ENABLE PSU_CRF_APB_TOPSW_MAIN_CTRL_DIVISOR0 PSU_CRF_APB_TOPSW_MAIN_ENABLE PSU_CRF_APB_TOPSW_LSBUS_CTRL_DIVISOR0		

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
6 bit Divider for individual components to be user programmed in Advance Clocking mode with PSU_OVERRIDE_BASIC_CLOCK = 1	PSU_CRF_APB_TOPSW_LSBUS_ENABLE PSU_CRF_APB_GTGREF0_REF_CTRL_DIVISOR0 PSU_CRF_APB_GTGREF0_ENABLE PSU_CRF_APB_DBG_TSTMP_CTRL_DIVISOR0 PSU_CRF_APB_DBG_TSTMP_ENABLE PSU_CRL_APB_IOPLL_TO_FPD_CTRL_DIVISOR0 PSU_CRL_APB_RPLL_TO_FPD_CTRL_DIVISOR0 PSU_CRL_APB_GEM0_REF_CTRL_DIVISOR0 PSU_CRL_APB_GEM1_REF_CTRL_DIVISOR0 PSU_CRL_APB_GEM2_REF_CTRL_DIVISOR0 PSU_CRL_APB_GEM3_REF_CTRL_DIVISOR0 PSU_CRL_APB_GEM0_REF_CTRL_DIVISOR1 PSU_CRL_APB_GEM1_REF_CTRL_DIVISOR1 PSU_CRL_APB_GEM2_REF_CTRL_DIVISOR1 PSU_CRL_APB_GEM3_REF_CTRL_DIVISOR1 PSU_CRL_APB_GEM_TSU_REF_CTRL_DIVISOR0 PSU_CRL_APB_GEM_TSU_REF_CTRL_DIVISOR1 PSU_CRL_APB_USB0_BUS_REF_CTRL_DIVISOR0 PSU_CRL_APB_USB0_BUS_REF_CTRL_DIVISOR1 PSU_CRL_APB_USB1_BUS_REF_CTRL_DIVISOR0	0-63	

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
	PSU_CRL_APB_USB1_BUS_REF_CTRL_DIVISOR1		
	PSU_CRL_APB_QSPI_REF_CTRL_DIVISOR0		
	PSU_CRL_APB_QSPI_REF_CTRL_DIVISOR1		
	PSU_CRL_APB_SDIO0_REF_CTRL_DIVISOR0		
	PSU_CRL_APB_SDIO0_REF_CTRL_DIVISOR1		
	PSU_CRL_APB_SDIO1_REF_CTRL_DIVISOR0		
	PSU_CRL_APB_SDIO1_REF_CTRL_DIVISOR1		
	PSU_CRL_APB_UART0_REF_CTRL_DIVISOR0		
	PSU_CRL_APB_UART0_REF_CTRL_DIVISOR1		
	PSU_CRL_APB_UART1_REF_CTRL_DIVISOR0		
	PSU_CRL_APB_UART1_REF_CTRL_DIVISOR1		
	PSU_CRL_APB_I2C0_REF_CTRL_DIVISOR0		
	PSU_CRL_APB_I2C0_REF_CTRL_DIVISOR1		
	PSU_CRL_APB_I2C1_REF_CTRL_DIVISOR0		
	PSU_CRL_APB_I2C1_REF_CTRL_DIVISOR1		
	PSU_CRL_APB_SPI0_REF_CTRL_DIVISOR0		
	PSU_CRL_APB_SPI0_REF_CTRL_DIVISOR1		
	PSU_CRL_APB_SPI1_REF_CTRL_DIVISOR0		
	PSU_CRL_APB_SPI1_REF_CTRL_DIVISOR1		
	PSU_CRL_APB_CAN0_REF_CTRL_DIVISOR0		
	PSU_CRL_APB_CAN0_REF_CTRL_DIVISOR1		
	PSU_CRL_APB_CAN1_REF_CTRL_DIVISOR0		
	PSU_CRL_APB_CAN1_REF_CTRL_DIVISOR1		
	PSU_CRL_APB_DEBUG_R5_ATCLK_CTRL_DIVISOR0		

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
	PSU_CRL_APB_CPU_R5_CTRL_DIVISOR0 PSU_CPU_R5_PERIPHERAL_ENABLE PSU_CRL_APB_OCM_MAIN_CTRL_DIVISOR0 PSU_CRL_APB_OCM_MAIN_ENABLE PSU_CRL_APB_IOU_SWITCH_CTRL_DIVISOR0 PSU_CRL_APB_IOU_SWITCH_ENABLE PSU_CRL_APB_CSU_PLL_CTRL_DIVISOR0 PSU_CRL_APB_PCAP_CTRL_DIVISOR0 PSU_CRL_APB_PCAP_ENABLE PSU_CRL_APB_LPD_LSBUS_ENABLE PSU_CRL_APB_LPD_SWITCH_CTRL_DIVISOR0 PSU_CRL_APB_DBG_LPD_CTRL_DIVISOR0 PSU_CRL_APB_NAND_REF_CTRL_DIVISOR0 PSU_CRL_APB_NAND_REF_CTRL_DIVISOR1 PSU_CRL_APB_ADMA_REF_CTRL_DIVISOR0 PSU_CRF_APB_DFT300_REF_CTRL_DIVISOR0 PSU_CRF_APB_DFT270_REF_CTRL_DIVISOR0 PSU_CRF_APB_DFT250_REF_CTRL_DIVISOR0 PSU_CRF_APB_DFT125_REF_CTRL_DIVISOR0 PSU_CRL_APB_PICDEBUG_REF_CTRL_DIVISOR, PSU_CRL_APB_PICDEBUG_CTRL_DIVISOR0 PSU_CRL_APB_PICDEBUG_TEMP_CTRL_DIVISOR0		
PLL source selection	PSU_CRF_APB_APLL_CTRL_SRCSEL	PSU_REF_CLK	PSU_REF_CLK
PLL source selection	PSU_CRF_APB_DPLL_CTRL_SRCSEL	PSU_REF_CLK	PSU_REF_CLK
PLL source selection	PSU_CRF_APB_VPLL_CTRL_SRCSEL	PSU_REF_CLK	PSU_REF_CLK
PLL source selection	PSU_CRF_APB_ACPU_CTRL_SRCSEL	APLL,VPLL, DPLL	APLL
PLL source selection	PSU_CRF_APB_DBG_TRACE_CTRL_SRCSEL	APLL,IOPLL,DPLL,FMIO_traceclk	DPLL
PLL source selection	PSU_CRF_APB_DBG_FPD_CTRL_SRCSEL	APLL,IOPLL, DPLL	DPLL
PLL source selection	PSU_CRF_APB_APM_CTRL_SRCSEL	APLL,VPLL, DPLL	DPLL
PLL source selection	PSU_CRF_APB_DP_VIDEO_REF_CTRL_SRCSEL	APLL,VPLL, DPLL	VPLL

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
PLL source selection	PSU_CRF_APB_DP_AUDIO_REF_CTRL_SRCSEL	APLL,VPLL,DPLL	VPLL
PLL source selection	PSU_CRF_APB_DP_STC_REF_CTRL_SRCSEL	APLL,VPLL, DPLL	VPLL
PLL source selection	PSU_CRF_APB_DDR_CTRL_SRCSEL	VPLL,DPLL	DPLL
PLL source selection	PSU_CRF_APB_GPU_REF_CTRL_SRCSEL	APLL,IOPLL, DPLL	VPLL
PLL source selection	PSU_CRF_APB_AFI0_REF_CTRL_SRCSEL	APLL,VPLL, DPLL	DPLL
PLL source selection	PSU_CRF_APB_AFI1_REF_CTRL_SRCSEL	APLL,VPLL, DPLL	DPLL
PLL source selection	PSU_CRF_APB_AFI2_REF_CTRL_SRCSEL	APLL,VPLL, DPLL	DPLL
PLL source selection	PSU_CRF_APB_AFI3_REF_CTRL_SRCSEL	APLL,VPLL, DPLL	DPLL
PLL source selection	PSU_CRF_APB_AFI4_REF_CTRL_SRCSEL	APLL,VPLL, DPLL	DPLL
PLL source selection	PSU_CRF_APB_AFI5_REF_CTRL_SRCSEL	APLL,VPLL, DPLL	DPLL
PLL source selection	PSU_CRF_APB_SATA_REF_CTRL_SRCSEL	APLL,VPLL, DPLL	APLL
PLL source selection	PSU_CRF_APB_PCIE_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_PL0_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_PL1_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_PL2_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_PL3_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRF_APB_GDMA_REF_CTRL_SRCSEL	APLL,VPLL,DPLL	DPLL
PLL source selection	PSU_CRF_APB_DPDMA_REF_CTRL_SRCSEL	APLL,VPLL, DPLL	DPLL
PLL source selection	PSU_CRF_APB_TOPSW_MAIN_CTRL_SRCSEL	APLL,VPLL, DPLL	DPLL
PLL source selection	PSU_CRF_APB_TOPSW_LSBUS_CTRL_SRCSEL	APLL,IOPLL, DPLL	DPLL

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
PLL source selection	PSU_CRF_APB_GTGREF0_REF_CTRL_SRCSEL	APLL,VPLL, DPLL	APLL
PLL source selection	PSU_CRF_APB_DBG_TSTMP_CTRL_SRCSEL	APLL,VPLL, DPLL	DPLL
PLL source selection	PSU_CRL_APB_IOPLL_CTRL_SRCSEL	PSU_REF_CLK	PSU_REF_CLK
PLL source selection	PSU_CRL_APB_RPLL_CTRL_SRCSEL	PSU_REF_CLK	PSU_REF_CLK
PLL source selection	PSU_CRL_APB_GEM0_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_GEM1_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_GEM2_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_GEM3_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_GEM_TSU_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_USB0_BUS_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_USB1_BUS_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_QSPI_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_SDIO0_REF_CTRL_SRCSEL	IOPLL,RPLL, VPLL	IOPLL
PLL source selection	PSU_CRL_APB_SDIO1_REF_CTRL_SRCSEL	IOPLL,RPLL, VPLL	IOPLL
PLL source selection	PSU_CRL_APB_UART0_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_UART1_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_I2C0_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_I2C1_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
PLL source selection	PSU_CRL_APB_SPI0_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_SPI1_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_CAN0_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_CAN1_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_DEBUG_R5_ATCLK_CTRL_SRCSEL	IOPLL,RPLL, DPLL	RPLL
PLL source selection	PSU_CRL_APB_CPU_R5_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_OCM_MAIN_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_IOU_SWITCH_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_CSU_PLL_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_PCAP_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_LPD_LSBUS_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_LPD_SWITCH_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_DBG_LPD_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_NAND_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_ADMA_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_PICDEBUG_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_PICDEBUG_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_DLL_REF_CTRL_SRCSEL	IOPLL,RPLLL	IOPLL

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
PLL source selection	PSU_CRL_APB_AMS_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_TIMESTAMP_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_AFI6_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
PLL source selection	PSU_CRL_APB_USB3_DUAL_REF_CTRL_SRCSEL	IOPLL,RPLL, DPLL	IOPLL
CSU register	PSU_CSU_CSU_TAMPER_0_ENABLE	0,1	0
External MIO	PSU_CSU_CSU_TAMPER_1_ENABLE	0,1	0
JTAG toggle detect	PSU_CSU_CSU_TAMPER_2_ENABLE	0,1	0
PL SEU error	PSU_CSU_CSU_TAMPER_3_ENABLE	0,1	0
AMS over temperature alarm for LPD	PSU_CSU_CSU_TAMPER_4_ENABLE	0,1	0
AMS over temperature alarm for APU	PSU_CSU_CSU_TAMPER_5_ENABLE	0,1	0
AMS voltage alarm for VCCPINT_FPD	PSU_CSU_CSU_TAMPER_6_ENABLE	0,1	0
AMS voltage alarm for VCCPINT_LPD	PSU_CSU_CSU_TAMPER_7_ENABLE	0,1	0
AMS voltage alarm for VCCPAUX	PSU_CSU_CSU_TAMPER_8_ENABLE	0,1	0
AMS voltage alarm for DDRPHY	PSU_CSU_CSU_TAMPER_9_ENABLE	0,1	0
AMS voltage alarm for PSIO bank 0/1/2	PSU_CSU_CSU_TAMPER_10_ENABLE	0,1	0
AMS voltage alarm for PSIO bank 3 (dedicated pins)	PSU_CSU_CSU_TAMPER_11_ENABLE	0,1	0
AMS voltage alarm for GT	PSU_CSU_CSU_TAMPER_12_ENABLE	0,1	0
Zeroize Non-volatile BBRAM key in addtion to the tamper resposne specified	PSU_CSU_CSU_TAMPER_0_ERASE_BBRAM	0,1	0
Same as above	PSU_CSU_CSU_TAMPER_1_ERASE_BBRAM	0,1	0
Same as above	PSU_CSU_CSU_TAMPER_2_ERASE_BBRAM	0,1	0
Same as above	PSU_CSU_CSU_TAMPER_3_ERASE_BBRAM	0,1	0
Same as above	PSU_CSU_CSU_TAMPER_4_ERASE_BBRAM	0,1	0
Same as above	PSU_CSU_CSU_TAMPER_5_ERASE_BBRAM	0,1	0
Same as above	PSU_CSU_CSU_TAMPER_6_ERASE_BBRAM	0,1	0

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
Same as above	PSU_CSU_CSU_TAMPER_7_ERASE_BBRAM	0,1	0
Same as above	PSU_CSU_CSU_TAMPER_8_ERASE_BBRAM	0,1	0
Same as above	PSU_CSU_CSU_TAMPER_9_ERASE_BBRAM	0,1	0
Same as above	PSU_CSU_CSU_TAMPER_10_ERASE_BBRAM	0,1	0
Same as above	PSU_CSU_CSU_TAMPER_11_ERASE_BBRAM	0,1	0
Same as above	PSU_CSU_CSU_TAMPER_12_ERASE_BBRAM	0,1	0
sec_lockdown_1: Setting this bit causes the CSU ROM to issue a secure lockdown and all GPIOB to be tri-stated when the tamper event occurs. Only the action of the most significant bit is taken **sec_lockdown_0**: Setting this bit causes the CSU ROM to issue a secure lockdown when the tamper event occurs. Only the action of the most significant bit is taken **sys_reset**: Setting this bit causes the CSU ROM to issue a system reset when the tamper event occurs. Only the action of the most significant bit is taken **sys_interrupt**: Setting this bit cuases the CSU ROM to issue a system interrupt when the tamper event occurs. Only the action of the most significant bit is taken	PSU_CSU_CSU_TAMPER_0_RESPONSE	SEC_ LOCKDOWN_0, SEC_ LOCKDOWN_1, SYS_RESET, SYS_INTERRUPT	SEC_LO CKDOW N_0
Same as above	PSU_CSU_CSU_TAMPER_1_RESPONSE	Same as above	Same as above
Same as above	PSU_CSU_CSU_TAMPER_2_RESPONSE	Same as above	Same as above
Same as above	PSU_CSU_CSU_TAMPER_3_RESPONSE	Same as above	Same as above
Same as above	PSU_CSU_CSU_TAMPER_4_RESPONSE	Same as above	Same as above
Same as above	PSU_CSU_CSU_TAMPER_5_RESPONSE	Same as above	Same as above
Same as above	PSU_CSU_CSU_TAMPER_6_RESPONSE	Same as above	Same as above
Same as above	PSU_CSU_CSU_TAMPER_7_RESPONSE	Same as above	Same as above

Table C-1: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value	Parameter Name	Possible Values	Default Value
Same as above	PSU_CSU_CSU_TAMPER_8_RESPONSE	Same as above	Same as above
Same as above	PSU_CSU_CSU_TAMPER_9_RESPONSE	Same as above	Same as above
Same as above	PSU_CSU_CSU_TAMPER_10_RESPONSE	Same as above	Same as above
Same as above	PSU_CSU_CSU_TAMPER_11_RESPONSE	Same as above	Same as above
Same as above	PSU_CSU_CSU_TAMPER_12_RESPONSE	Same as above	Same as above

Debugging

This appendix includes details about resources available on the Xilinx Support website and debugging tools.

Finding Help on Xilinx.com

To help in the design and debug process when using the Zynq® UltraScale+™ MPSoC Processing System, the Xilinx Support web page contains key resources such as product documentation, release notes, answer records, information about known issues, and links for obtaining further product support.

Documentation

This product guide is the main document associated with the core. This guide, along with documentation related to all products that aid in the design process, can be found on the Xilinx Support web page or by using the Xilinx Documentation Navigator.

Download the Xilinx Documentation Navigator from the Downloads page. For more information about this tool and the features available, open the online help after installation.

Solution Centers

See the Xilinx Solution Centers for support on devices, software tools, and intellectual property at all stages of the design cycle. Topics include design assistance, advisories, and troubleshooting tips.

Answer Records

Answer Records include information about commonly encountered problems, helpful information on how to resolve these problems, and any known issues with a Xilinx product. Answer Records are created and maintained daily ensuring that users have access to the most accurate information available.

Answer Records for this core can also be located by using the Search Support box on the main Xilinx support web page. To maximize your search results, use proper keywords such as

- Product name
- Tool message(s)
- Summary of the issue encountered

A filter search is available after results are returned to further target the results.

Master Answer Record for the Zynq UltraScale MPSoC Processing System

AR: 66183

Technical Support

Xilinx provides technical support at the Xilinx Support web page for this LogiCORE™ IP product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support if you do any of the following:

- Implement the solution in devices that are not defined in the documentation.
- Customize the solution beyond that allowed in the product documentation.
- Change any section of the design labeled DO NOT MODIFY.

To contact Xilinx Technical Support, navigate to the Xilinx Support web page.

Additional Resources and Legal Notices

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see Xilinx Support.

References

These documents provide supplemental material useful with this product guide:

1. *Zynq UltraScale All Programmable MPSoC Technical Reference Manual* (UG1085)
2. *Zynq UltraScale+ MPSoC Register Reference* (UG1087)
3. *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994)
4. *Vivado Design Suite User Guide: Designing with IP* (UG896)
5. *Vivado Design Suite User Guide: Getting Started* (UG910)
6. *Vivado Design Suite User Guide: Logic Simulation* (UG900)
7. *Vivado Design Suite User Guide: Programming and Debugging* (UG908)
8. *AMBA AXI4-Stream Protocol Specification*
9. *DDR3L SDRAM Data Sheet* (PDF location -- Micron Technology Inc.)

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
06/08/2016	1.2	• Updated Figures 4-2 through 4-7. • Changed all _t_n signals to _t and removed the word "INVERTED" from the descriptions. • Modified PSU_CAN0_PERIPHERAL_ENABLE and PSU_CAN1_PERIPHERAL_ENABLE parameter default to be 0. • Removed PSU_DPAUX_PERIPHERAL_ENABLE parameter. • Updated the possible values for the PSU_DPAUX_PERIPHERAL_IO, PSU_SD1_SPEED_MODE. and PSU_CRF_APB_TOPSW_MAIN_CTRL_FREQMHZ parameters. • Added the PSU__DISPLAYPORT__PERIPHERAL__ENABLE and PSU__DP__LANE_SEL parameters. • Modified "Gpio" to be "GPIO" • Updated many of the rows that were missing information in Table C-1.
04/06/2016	1.1	• Added High Speed SerDes configuration feature. • Renamed Unsupported Features" section as "Unsupported Features and Known Limitations." Removed all of the bulleted items. Added cross reference to the master answer record. • Removed ACP Transaction Checker section. • Removed NOR flash. • Updated AXI4 I/O Compliant Interfaces section. • Added data to Table 2-2, Device Utilization – Zynq UltraScale+ MPSoC. • Removed MicroBlaze information from the General Design Guidelines section. • Added or updated all screen displays in Chapter 4. • Replaced Drive 0 and Drive 1 fields with Drive Strength field. • Replaced Pull Enable and Pullup fields with Pull Type field • Added information about MIO and EMIO, Number of MIOs and their organization in the banks • Added brief details about SerDes configuration supported in PCW. • Added information about MIO Voltage standard; specified that the default voltage for the banks will be LVCMOM33 • Replaced **Input Frequency** field with **Requested Freq (MHz)**. • Replaced **Actual Frequency** field with **Actual Frequency (MHz).** • Replaced **Range** with **Range (MHz)**. • Added details about Cross Domain PLL, GT lane clocking, and Auto Vs Manual features.
11/18/2015	1.0	Initial version for public access.

Notice of Disclaimer

The information disclosed to you hereunder (the "Materials") is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of the Limited Warranties which can be viewed at http://www.xilinx.com/warranty.htm; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in Critical Applications: http://www.xilinx.com/warranty.htm#critapps.

© Copyright 2015–2016 Xilinx, Inc. Xilinx, the Xilinx logo, Artix, ISE, Kintex, Spartan, Virtex, Vivado, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. PCI, PCIe and PCI Express are trademarks of PCI-SIG and used under license. AMBA and ARM are registered trademarks of ARM in the EU and other countries. Cortex is a trademark of ARM in the EU and other countries. All other trademarks are the property of their respective owners.