DG0808 Demo Guide PolarFire FPGA PCIe EndPoint and DDR4 Memory Controller Data Plane Using Splash Kit





а 🔨 Міскоснір company

Microsemi Headquarters

One Enterprise, Aliso Viejo, CA 92656 USA Within the USA: +1 (800) 713-4113 Outside the USA: +1 (949) 380-6100 Sales: +1 (949) 380-6136 Fax: +1 (949) 215-4996 Email: sales.support@microsemi.com www.microsemi.com

©2018 Microsemi, a wholly owned subsidiary of Microchip Technology Inc. All rights reserved. Microsemi and the Microsemi logo are registered trademarks of Microsemi Corporation. All other trademarks and service marks are the property of their respective owners. Microsemi makes no warranty, representation, or guarantee regarding the information contained herein or the suitability of its products and services for any particular purpose, nor does Microsemi assume any liability whatsoever arising out of the application or use of any product or circuit. The products sold hereunder and any other products sold by Microsemi have been subject to limited testing and should not be used in conjunction with mission-critical equipment or applications. Any performance specifications are believed to be reliable but are not verified, and Buyer must conduct and complete all performance and other testing of the products, alone and together with, or installed in, any end-products. Buyer shall not rely on any data and performance specifications or parameters provided by Microsemi. It is the Buyer's responsibility to independently determine suitability of any products and to test and verify the same. The information provided by Microsemi hereunder is provided "as is, where is" and with all faults, and the entire risk associated with such information is entirely with the Buyer. Microsemi does not grant, explicitly or implicitly, to any party any patent rights, licenses, or any other IP rights, whether with regard to such information itself or anything described by such information. Information provided in this document is proprietary to Microsemi, and Microsemi reserves the right to make any changes to the information in this

About Microsemi

Microsemi, a wholly owned subsidiary of Microchip Technology Inc. (Nasdaq: MCHP), offers a comprehensive portfolio of semiconductor and system solutions for aerospace & defense, communications, data center and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; enterprise storage and communication solutions, security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Learn more at www.microsemi.com.



Contents

1	Revisi 1.1 1.2	on History 1 Revision 2.0 1 Revision 1.0 1
2	PolarF 2.1 2.2 2.3 2.4 2.5 2.6 2.7	Fire FPGA PCIe EndPoint and DDR4 Memory Controller Data Plane2Design Requirements3Prerequisites4Demo Design42.3.1Design Data Flow62.3.2Design Implementation92.3.3IP Configuration11Clocking Structure23Reset Structure24Throughput Measurement24Simulating the Design252.7.1Simulation Flow25
3	Libero 3.1 3.2 3.3 3.4 3.5	Design Flow27Synthesize273.1.1Resource Utilization27Place and Route28Verify Timing29Generate Bitstream30Run PROGRAM Action30
4	Progra	amming the Device Using FlashPro
5	Runni 5.1 5.2 5.3	ng the Demo32Installing PCIe Demo Application32Running the Demo Through PCIe345.2.1Connecting the Board to the Host PC PCIe Slot345.2.2Driver Installation365.2.3Running the PCIe Demo Application38Running the Demo Through UART515.3.1UART—DMA Operations52
6	Apper	ndix: References



Figures

F :	DOLE Device Devices Target event Discher Discher aus	~
Figure 1	PCIe Demo Design Top-Level Block Diagram	. 5
Figure 2	DMA0 – Example of SG DMA Operation	. /
Figure 3	DMA1 – Example of SG DMA Operation	. 8
Figure 4	PCIe EndPoint Reference Design	. 9
Figure 5	PCIe_EP SmartDesign	. 9
Figure 6	PCIe_TL_CLK SmartDesign	10
Figure 7	AXI_to_APB SmartDesign	10
Figure 8	CoreDMA_IO_CTRL SmartDesign	10
Figure 9	UART SmartDesign	11
Figure 10	PCIe Configurator	12
Figure 11	PCIe Configurator—BAR 0 Master Settings	13
Figure 12	PCIe Configurator—BAR 2 Master Settings	13
Figure 13	Transceiver Reference Clock Configurator	14
Figure 14	Transmit PLL Configurator	14
Figure 15	DDR4 Configurator	15
Figure 16	DDR4 Configurator—Memory Initialization	16
Figure 17	DDR4 Configurator—Memory Timing	17
Figure 18	DDR4 Configurator—Controller	18
Figure 19	DDR4 Configurator—Misc	19
Figure 20	CoreAXI4DMAController IP Configurator	20
Figure 21	CoreAXI4Interconnect IP Core Configuration	21
Figure 22	CoreAXI4Interconnect IP Master Configuration	21
Figure 23	CoreAXI4Interconnect IP Slave Configuration	22
Figure 24	PolarFire SRAM IP Configurator	23
Figure 25	Clocking Structure	23
Figure 26	Reset Structure	24
Figure 27	Simulating the Design	25
Figure 28	Simulating the Design Window	26
Figure 20	Simulation Waveform Window	20
Figure 30	Simulation Waveform Window	20
Figure 31		20
Figure 31		21
Figure 32		20
Figure 33		29
Figure 34		29
Figure 35	Design Flow	29
Figure 30		30
Figure 37		30
Figure 38		31
Figure 39		32
Figure 40	Pule Demo Application Installation Steps	33
Figure 41		33
Figure 42		34
Figure 43		35
Figure 44		36
Figure 45	Browse for Driver Software	36
Figure 46	Browse for Driver Software Continued	37
Figure 47	Windows Security	37
⊢igure 48	Successful Driver Installation	37
Figure 49	Device Manager—PCIe Device Detection	38
Figure 50	PCIe EndPoint Demo Application	39
Figure 51	Device Info	39
Figure 52	Demo Controls	40
Figure 53	Demo Controls—Continued	40
Figure 54	Configuration Space	41



Figure 55 Figure 56 Figure 57 Figure 58	PCIe BAR2 Memory Access—LSRAM PCIe BAR2 Memory Access—DDR4 Continuous DMA—Operations Continuous DMA Operations with DMA Transfer Type Selection as Both PC and LSRAM	41 42 43 44
Figure 59	DMA Transfer Type Selection—Continuous Memory Test	45
Figure 60	Continuous DMA Memory Test—Memory Test Successful	46
Figure 61	SGDMA—Operations	47
Figure 62	SGDMA—Memory Test	48
Figure 63	SGDMA Memory Test—Memory Test Successful	49
Figure 64	Core DMA—Operations	50
Figure 65	Device Manager—UART Ports	51
Figure 66	PCIe EndPoint Demo Application	51
Figure 67	UART—DMA Operations	52
Figure 68	UART—Memory Test	53



Tables

Table 1	Design Requirements	. 3
Table 2	Resource Utilization	27
Table 3	Jumper Settings	30
Table 4	Jumper Settings	31
Table 5	PolarFire Throughput Summary—Continuous DMA Mode	54
Table 6	PolarFire Throughput Summary—SGDMA Mode	54
Table 7	PolarFire Throughput Summary—Core DMA Mode	54



1 Revision History

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the most current publication.

1.1 Revision 2.0

The document was updated for Libero SoC PolarFire v2.2 release.

1.2 Revision 1.0

The first publication of this document.



2 PolarFire FPGA PCIe EndPoint and DDR4 Memory Controller Data Plane

Microsemi PolarFire[®] FPGAs contain fully integrated PCIe EndPoint and Root Port subsystems with optimized embedded controller blocks that use the physical layer interface (PHY) of the transceiver. Each PolarFire device includes two embedded PCIe subsystem (PCIESS) blocks that can be configured either separately, or as a pair, using the PCIESS configurator in the Libero[®] SoC PolarFire software.

The PCIESS is compliant with the PCI Express Base Specification, Revision 2.1. It implements memorymapped advanced microcontroller bus architecture (AMBA) advanced extensible interface 4 (AXI4) access to the PCIe space, and the PCIe access to the memory-mapped AXI4 space. For more information, see *UG0685: PolarFire FPGA PCI Express User Guide*.

The DDR subsystem addresses memory solution requirements for a wide range of applications with varying power consumption and efficiency levels. The subsystem can be configured to support DDR4 and LPDDR3 memory devices. The subsystem is intended for accessing DDR memories for applications that require high-speed data transfers and code execution. For more information DDR memory controller, see *UG0676: PolarFire FPGA DDR Memory Controller User Guide*.

This document explains how to use the accompanying reference design to demonstrate the high-speed data transfer capability of the PolarFire FPGA using the hardened PCIe EndPoint, and DDR4 controller IP. The PCIe controller, built-in direct memory access (DMA) controller, and the CoreAXI4DMAController IP are used to achieve high-speed, bulk data transfers, as follows:

- The PCIe controller's built-in DMA controller perform bulk-data transfer between contiguous/scatter gather memory locations on a host PC and contiguous memory locations of DDR4/LSRAM.
- The CoreAXI4DMACcontroller performs data transfers between DDR4 memory and LSRAM using the CoreAXI4DMA controller.

The demo also shows how to use pre-synthesized design simulations using PCIe BFM script to initiate the PCIe EndPoint DMA to perform data transfers between LSRAM, DDR4, and PCIe.

The Windows kernel mode PCIe device driver, developed using the Windows Driver Kit (WDK) platform, interacts with the PolarFire PCIe EndPoint from the host PC. A GUI application that runs on the host PC is provided to set up and initiate the DMA transactions between the host PC memory, DDR4, and the LSRAM memories of the PolarFire splash kit through the PCIe interface.

A user application interface is provided for the GUI to interact with the PCIe driver. The GUI can also initiates the DMA transactions between DDR4 and LSRAM through UART IF. If the host PC PCIe slot is not available, the DMA between DDR4 and LSRAM is exercised through UART IF.

The PCIe EndPoint reference design can be programmed using any of the following options:

- Using the stp file: To program the device using the stp file provided along with the design files, see Programming the Device Using FlashPro, page 31.
- Using Libero SoC PolarFire: To program the device using Libero SoC PolarFire, see Libero Design Flow, page 27. Use this option when the reference design is modified.



2.1 Design Requirements

The following table lists the hardware, software, and IP requirements for this demo design.

Requirement	Version
Operating system	64-bit Windows 7 or 10
Hardware	
PolarFire Splash Kit (MPF300T-Splash-KIT) – PolarFire Splash Board – 12 V/5 A power adapter – USB 2.0 A-male to mini-B cable for UART and programming	Rev 2 or later
PCIe Edge card ribbon cable	
Host PC with PCIe compliant slot with x4 or higher width	
Software	
Libero SoC PolarFire	v2.2
Modelsim	10.5c Pro
Synplify Pro	L-2016.09M-SP1-5
IP	
PF_XCVR_REF_CLK	1.0.103
PF_TX_PLL	1.0.112
PF_PCIE	1.0.242
PF_CCC	1.0.113
PF_RESET	2.1.100
PF_OSC	1.0.102
NGMUX	1.0.101
PF_INIT_MONITOR	2.0.103
CoreAXI4Interconnect	2.5.100
COREAXI4DMACONTROLLER	2.0.100
DDR4	2.3.108
CoreAHBLite	5.3.101
CoreAPB	4.1.100
CoreAXItoAHBL	3.2.104
CoreAHBtoAPB3	3.1.100
CoreUART	5.6.102
CoreAPB3	4.1.100
PolarFire SRAM	1.1.125
PCIe_DRI	1.0.101



2.2 **Prerequisites**

Before you start:

- 1. Download the design files from the following link: http://soc.microsemi.com/download/rsc/?f=mpf_dg0808_liberosocpolarfirev2p2_df
- Download and install Libero SoC PolarFire on the host PC from the following location: https://www.microsemi.com/products/fpga-soc/design-resources/design-software/libero-soc-polarfire#downloads

The latest versions of ModelSim, Synplify Pro, and FTDI drivers are included in the Libero SoC PolarFire installation package.

2.3 Demo Design

The top-level block diagram of the PCIe EndPoint demo design is shown in Figure 1, page 5. Any external PCIe root-port or bridge can establish PCIe link with the PolarFire FPGA PCIe EndPoint and access the control registers, DDR4, and fabric memory through BAR space using the memory write (MWr) and memory read (MRd) transaction layer packets (TLPs). The PCIe EndPoint converts these MWr and MRd TLPs into AXI4 master interface transactions and accesses the fabric memory through CoreAXI4Interconnect IP.

The PCIe Demo application on the host PC initiates the DMA transfers through the PCIe device drivers. The driver on the host PC allocates memory and initiates the DMA Engine in the PolarFire PCIe controller by accessing the PCIe DMA registers through BAR0. The PCIe controller has two independent DMA Engines:

- DMA Engine0: performs DMA from host PC memory to DDR4/LSRAM.
- DMA Engine1: performs DMA from DDR4/LSRAM to host PC memory.
- **Note:** For SGDMA type of DMA operations, the PCIe driver finds the available memory locations and creates the buffer descriptor chain for the different memory locations. It also configures the PCIe DMA for SGDMA and the base address of the first buffer descriptor.

The PCIe demo application initiates CoreAXI4DMA controller IP to perform the DMA between DDR4 memory and LSRAM. The following are the two channels of the CoreAXI4DMA controller IP:

- Channel0: performs DMA from—DDR4 to LSRAM
- Channel1: performs DMA from—LSRAM to DDR4

The host PC application initiates the CoreAXI4DMA controller IP depending on the DMA type through BAR2 when the PCIe edge connector is connected to the host PC PCIe slot. The host PC application also initiates the CoreAXI4DMA controller IP through UART IF. This option is provided to exercise the DDR throughputs when the PolarFire Splash kit is not connected to the host PC PCIe slot.







----- Hard DMA Path

----- Soft DMA Path

2.3.1 Design Data Flow

The demo design performs the following control plane operations:

- LED Blink: host PC driver performs BAR2 memory write operation (MWr) to EndPoint. The PCIe controller generates AXI write transaction on AXI_IO_CTRL logic's to blink LEDs.
- DIP Switch Read: host PC driver performs BAR2 memory read operation (MRd) to EndPoint. The PCIe controller generates AXI read transaction on AXI_IO_CTRL logic's to blink LEDs.
- MSI Interrupt Count: when on-board push button is pressed, the PCIe EndPoint generates interrupt to host PC and the host PC driver increments the corresponding interrupt counter.
- Memory Read/Write: host PC driver configures the ATR2 translation address to DDR4/LSRAM base address. It performs BAR2 memory read/write transactions to DDR4/LSRAM memories.

The demo design supports three types of DMA operations.

- Continuous DMA operations
- SDGMA Operations
- Core DMA Operations

2.3.1.1 Continuous DMA Operations

The PCIe DMA0/DMA1 controllers perform DMA between continuous memory locations when SGDMA mode is disabled. The following sections explain the data flow of DMA0 and DMA1.

2.3.1.1.1 DMA0 – Host PC Memory to DDR4/LSRAM

PCIe DMA Engine0 performs continuous DMA from host PC memory to DDR4/LSRAM memories as described in the following steps:

- 1. PolarFire_PCIe_GUI application sets up the DMA controller through the PCIe link. This includes DMA source and destination, address, and size.
- 2. DMA controller initiates a read transaction to the PCIe core.
- 3. The PCIe core sends the memory read (MRd) transaction layer packets (TLP) to the host PC.
- 4. The host PC returns a completion (CpID) TLP to the PCIe link.
- 5. This returned data is written to the DDR4/LSRAM memories using PCIe AXI master interface.
- 6. The DMA controller repeats this process (from step 2 to 5) until the DMA size of data transfer is completed.
- 7. The DMA controller sends the MSI0 interrupt to the host PC, the driver on the host PC detects the interrupt, reads the DMA status, and the number of clock cycles consumed to complete the DMA transaction to the PolarFire_PCIe_GUI application.

2.3.1.1.2 DMA1 – DDR4/LSRAM to Host PC Memory

PCIe DMA Engine1 performs continuous DMA from DDR4/LSRAM memories to host PC memory as described in the following steps.:

- 1. PolarFire_PCIe_GUI application sets up the DMA controller through the PCIe link. This includes DMA source and destination, address, and size.
- 2. DMA controller initiates an AXI burst read transaction to read the data from DDR4/LSRAM memories.
- 3. The DMA controller initiates write transaction to PCIe core with the read data. The PCIe core sends a memory write (MWr) TLP to the host PC.
- 4. The DMA controller repeats this process (steps 2 and 3) until the DMA size of data transfer is completed.
- 5. The DMA controller sends the MSI1 interrupt to the host PC. The driver on the host PC detects the interrupt, reads the DMA status, and the number of clock cycles consumed to complete the DMA transaction to the PolarFire_PCIe_GUI application.

2.3.1.2 SGDMA Operations

The PCIe DMA0/DMA1 performs DMA between scattered host PC memory locations and continuous memories of PolarFire when SGDMA mode is enabled.

2.3.1.2.1 Host PC Memory to DDR4

PCIe DMA Engine0 performs DMA from host PC memory to DDR4 memories as shown in the following figure.

The following steps describe the SGDMA operation of PCIe DMA0:

- 1. PolarFire_PCIe_GUI application requests the PCIe driver for SG DMA. The driver on the host PC allocates the available memory location and creates the buffer descriptors with the scattered memory location addresses and location size.
- 2. The destination DDR4 memory is treated as the continuous memory. The driver configures the PCIe DMA0 with the first buffer descriptor address and initiates the DMA.
- 3. DMA controller initiates read transaction to the PCIe core with the buffer descriptor address.
- 4. The PCIe core sends the memory read (MRd) transaction layer packets (TLP) to the host PC. The host PC returns a completion (CpID) TLP to the PCIe link.
- 5. The DMA controller extracts these buffer descriptors and initiates the read transaction to PCIe core with the host PC memory location address in the descriptor.
- 6. The PCIe core sends the memory read (MRd) transaction layer packets (TLP) to the host PC. The host PC returns a completion (CpID) TLP to the PCIe link.
- 7. This return data is written to the DDR4 memories using PCIe AXI master interface.
- 8. The DMA controller repeats this process (from step 3 to 7) until the DMA size of data transfer is completed.
- The DMA controller sends the MSI0 interrupt to the host PC. The driver on the host PC detects the interrupt, reads the DMA status, and the number of clock cycles consumed to complete the DMA transaction to the PolarFire_PCIe_GUI application.

Figure 2 • DMA0 – Example of SG DMA Operation

2.3.1.2.2 DDR4 to Host PC Memory:

PCIe DMA Engine1 performs DMA from DDR4 memories to host PC memory as shown in the following figure.

The following steps describe the SGDMA operation of PCIe DMA1:

- 1. PolarFire_PCIe_GUI application requests the PCIe driver for SG DMA. The driver on the host PC allocates the available memory locations and creates the buffer descriptors with the scattered memory location addresses and location size.
- The source DDR4 memory is treated as the continuous memory. Single buffer descriptor is created in LSRAM with the base address of DDR4 memory. The LSRAM base address is provided to DMA controller for source descriptor address.
- 3. The driver configures the PCIe DMA1 with the first host PC destination buffer descriptor address and initiates the DMA.
- 4. DMA controller initiates read transaction to the PCIe core with the buffer descriptor address.
- 5. The PCIe core sends the memory read (MRd) transaction layer packets (TLP) to the host PC. The host PC returns a completion (CpID) TLP to the PCIe link.
- 6. The DMA controller extracts these buffer descriptors and initiates an AXI burst read transaction to read the data from DDR4 memories.
- 7. With this read data, DMA controller initiates the write transaction to PCIe core with the host PC memory location address in the descriptor.
- 8. The PCIe core sends the memory write (MWr) transaction layer packets (TLP) to the host PC.
- 9. The DMA controller repeats this process (from step 4 to 8) until the DMA size of data transfer is completed.
- 10. The DMA controller sends the MSI1 interrupt to the host PC. The driver on the host PC detects the interrupt, reads the DMA status, and the number of clock cycles consumed to complete the DMA transaction to the PolarFire_PCle_GUI application.

Figure 3 • DMA1 – Example of SG DMA Operation

2.3.2 Design Implementation

The following figure shows the Libero SoC PolarFire software design implementation of the PCIe EndPoint reference design.

Figure 4 • PCIe EndPoint Reference Design

The top-level design includes the following SmartDesign components:

- PCle_EP
- AXI_to_APB
- CoreDMA_IO_CTRL

The PCIe_EP SmartDesign implements PCIe EndPoint and its clocking scheme as shown in the following figure. It also includes the sw_debounce module, which is used to suppress bounces from onboard push buttons and to generate a pulse to the PCIe controller interrupt line. The rst_controller logic is used to reset the PCIe EndPoint when host PC generates the EndPoint reset through PCIe PERSTn side band signal. The rst_controller.v fabric logic monitors the PERSTn signal of PCIe Edge card. It performs the assertion and de-assertion of PCIe and PCS soft resets on raising edge of PERSTn signal. It uses the dynamic reconfiguration interface to access the PCIe and PCS soft reset registers.

Figure 5 • PCIe_EP SmartDesign

The PCIe_TL_CLK SmartDesign implements PCIe TL CLK for PolarFire devices as shown in the following figure. PCIe TL CLK needs to be connected to CLK_125MHZ of Tx PLL. In PolarFire devices, TL CLK is available only after PCIe initialization. The 80 MHz clock is derived from the on-chip 160 MHz oscillator to drive the TL CLK during PCIe initialization. The NGMUX is used to switch this clock to the required CLK_125MHz after PCIe initialization.

Figure 6 • PCIe_TL_CLK SmartDesign

The AXI_to_APB SmartDesign implements AXI to APB using different IP cores as shown in the following figure. AXI to APB IF accesses the PCIe control registers through the PCIe APB IF from the BAR0 space.

Figure 7 • AXI_to_APB SmartDesign

The CoreDMA_IO_CTRL SmartDesign implements fabric registers, CoreDMA4DMA IP initialization, and UART_SD as shown in the following figure.

The UART_SD SmartDesign implements logic required for UART IF as shown in the following figure.

Figure 9 • UART SmartDesign

2.3.3 IP Configuration

The following IPs and macros need to be configured before simulating the demo design:

- PCle
- Transceiver reference clock
- Transmit PLL
- CoreAXI4Interconnect IP
- PolarFire SRAM IP
- DDR4
- CoreRESET_PF
- CoreAXI4DMAController IP

2.3.3.1 PCIe

The PCIESS is configured as an EndPoint with maximum link speed and maximum link width—Gen2 (5.0 Gbps) link speed and x4 link width. The **Simulation Level** in the configurator is set to BFM to simulate the design using PCIe BFM script, as shown in the following figure. The PCIe fabric interface is always the same regardless of the link width or lane rate. APB interface is enabled to access the PCIe DMA and Address translation registers.

Figure 10 • PCIe Configurator

CIEXPIESS crosemi:SgCore:PF_PCIE:1.0.242 General Identification Power Management Ini	terrupts and Auxiliary Settings Master Settings	Slave Settings	
	PCIe 0	PCle 1	
PCIe Controller	Disabled	Enabled	
Port Type		End Point	
Number of Lanes		×4	
Lane Rate	-	Gen2 (5.0 Gbps)	
Reference Clock Frequency	-	100	PF_PCIE_0
CDR Reference Clock Source	-	Dedicated	
Number of CDR Reference Clocks	-	1	
General Settings			
AXI Clock Frequency is less than 100MH	2		
TX PLL base data rate	-	5000 Mbps	
TX PLL bit clock frequency		2500 MHz	
Ontional Interfaces			
Enable Dynamic Heconfiguration Interface	e [DHI] for XLVH lane access		
Simulation Level Settings Simulation Level BFM			PF_PCE
Simulation Level Settings Simulation Level BFM			PF_PCIE

Figure 11 • PCIe Configurator—BAR 0 Master Settings

aeneral Identification Power Management Interrupts and Auxiliary Settings Master Settings Slave Settings						
□ Bar û Table						
	PCIe 1					
BAR Туре	64-bit prefetchable memory					
BAR Size	64 KB					
AXI Address (32 bit)	0x03000000					
H Barlladie						
Bar 2 Table						
Bar 3 Table						
🗄 Bar 4 Table						
🗄 Bar 5 Table						

Figure 12 • PCIe Configurator—BAR 2 Master Settings

General Identification Power Management Interrupts and Auxiliary Set	tings Master Settings Slave Settings						
▪ Bar O Table							
Bar 1 Table							
Bar 2 Table							
	PCIe 1						
BAR Type	64-bit prefetchable memory						
BAR Size	1 MB						
AXI Address (32 bit)	0x1000000						
🗄 Bar 3 Table							
Bar 4 Table							
🗄 Bar 5 Table							

The following two BARs are configured in 64-bit:

- BAR0: accesses the PCIe DMA, address translation, and interrupt registers through the PCIe controller's APB interface. The address translation register associated with BAR0 is configured to translate the BAR0 address to the PCIe APB IF base address (0x0300_0000).
- BAR2: accesses the fabric control registers and AXI LSRAM, and DDR4 memories. By default, the address translation register associated with BAR2 is configured to access the fabric control registers (0x1000_0000). To access the LSRAM, and DDR4 memories, the driver on the host PC configures the BAR2 address translation register (TRSL_ADDR) to LSRAM (0x3000_0000)/DDR4 (0x4000_0000) memory base address using the PCIe APB IF through BAR0.

2.3.3.2 Transceiver Reference Clock

The transceiver reference clock can be configured either as a differential, or two single-ended REFCLKs. In this reference design, the **Reference Clock 0** is configured as a differential reference clock as shown in the following figure. This demo requires only one REFCLK (Reference Clock 0). The REFCLK is the clock source for transceivers and global clock network in this design.

Transceiver Reference Clock Configu Microsemi:SgCore:PF_XCVR_REF_CLK:1.0.103	urator
Microsemi:SgCore:PF_XCVR_REF_CLK:1.0.103	
Configuration	
Reference Clock 0	
Enable reference clock 0 🔽	PF_AUVK_KEF_ULK_U
Reference Clock 0 Mode Differential	
Enable fabric dock output	-REF_CLK_PAD_P
Reference Clock 1	-REF_CLK_PAD_N
Enable reference clock 1	
Reference Clock 1 Mode LVCMOS	PF XCVR RFF CIK
Enable fabric clock output	
	Symbol /
Help -	OK Cancel

2.3.3.3 Transmit PLL

The Transmit PLL **Reference Clock** and **Desired Output Clock** are set to 100 MHz and 5000 Mbps, respectively, as shown in the following figure.

The PolarFire FPGA transceiver is a half-rate architecture, that is, the internal high-speed path uses both edges of the clock to keep the clock rates down. Therefore, the clock can run at half of the data rate, thereby consuming less dynamic power. The transceiver in PCIe mode requires a 2500 MHz bit clock.

Figure 14 •	Transmit PLL	Configurator
-------------	--------------	--------------

Configurator	- 🗆 ×
Transmit PLL Microsemik5gCore:PF_TX_PLL:1.0.112	
Transmit PLL Clock Inputs/Outputs Reference Clock Source Dedicated Desired Output Bit Clock Source Dedicated Desired Output Bit Clock Source Dedicated Mthz Clock Options C Normal Mode C 3 Itter Cleaning Mode 1005 SyncE 32Bit C SSCG Modulation Features Fenable Dynamic Reconfiguration Interface (DRI) Log	PF_TX_PLL 0
🗏 Messages 😵 Errors 🗼 Warnings 👔 Info	
Help •	OK Cancel

2.3.3.4 DDR4

The DDR4 subsystem is configured to access the 16-bit DDR4 memory through an AXI4 64-bit interface. The DDR4 memory initialization and timing parameters are configured as per the DDR4 memory on PolarFire Splash kit. The following figure shows general configuration settings for the DDR4 memory.

Note: The PolarFire Splash kit supports 32-bit DDR4 memory. This demo design uses only 16-bit DDR4 memory to meet the 200 MHz fabric logic Place and Route timing.

Figure 15 • DDR4 Configurator

Configurator			- 🗆 X
DDR4			
Microsemi:SystemBuilder:PF_DDR4:2.3.108			
	General Memory Initialization	Memory Timing Cont	oller Misc.
PF_DDR4_UI_default_configuration	🗆 Тор		
	Protocol DDR4 -		
± 4 Gb x 16	Generate PHY only		
⊕ 8 Gb x 16 ⊕ 16 Gb x 8	Clock		
	Memory Clock Frequency (MHz)	800.0	
DDR4-1600K (11-11-11)	Henry Clock requercy (Hirz)	1800.0	
DDR4-1600L (12-12-12) - 16 Gb x 16 - C	CCC PLL Clock Multiplier	16 💌	
DDR4-1600L (12-12-12) - 16 Gb x 16 - C DDR4-1600L (12-12-12) - 16 Gb x 16 - C DDR4-1600L (12-12-12) - 16 Gb x 16 - C	CCC PLL Reference Clock Frequency	y (MHz) 50.000	
DDR4-1600L (12-12-12) - 16 Gb x 16 - C DDR4-1600L (12-12-12) - 16 Gb x 16 - C	User Logic Clock Rate	QUAD 💌	
	User Clock Frequency	200.0	
	Topology		
	Memory Format		
	DQ Width	16	
	SDRAM Number of Ranks	1	
	Enable address mirroring on odd ran	iks 🗆	
	DQ/DQS group size	8 👻	
	Row Address width	16	
	Column Address Width	10	
	Bank Address Width	2	
	Bank Group Address Width	1	
	Enable DM	DM	
	READ DBI enable	Disabled	
Apply New preset	Enable Parity/Alert		
	Enable ECC	Г	
	Number of dark subsubs		
	Number of dock outputs		
Help		_	OK Cancel

The following figure shows initialization configuration settings for the DDR4 memory.

Figure 16 • DDR4 Configurator—Memory Initialization

DDR4		
licrosemi:SystemBuilder:PF_DDR4:2.3.108		
Q	General Memory Initialization Memory Timing Controller	Misc.
PF_DDR4_UI_default_configuration	Mode Register 0	
in JEDEC ⊡ 2 Gb x 8		
⊕ 2 Gb x 16	Burst Length Fixed BL8	
4 Gb x 8		
	Read Burst Type Sequential	
. 16 Gb x 8	Memory CAS Latency 12	
⊡ 16 Gb x 16		
DDR4-1600J (10-10-10)	Mode Register 1	
DDR4-1600L (12-12-12)		
DDR4-1600L (12-12-12) - 16 Gb x 16 - C	ODT Rtt Nominal Value RZQ/6	
DDR4-1600L (12-12-12) - 16 Gb x 16 - C		
DDR4-1600L (12-12-12) - 16 Gb x 16 - C DDR4-1600L (12-12-12) - 16 Gb x 16 - C	Memory Additive CAS Latency Disabled	
DDR4-1600L (12-12-12) - 16 Gb x 16 - C	Output Drive Strength	
DDR4-1600L (12-12-12) - 16 Gb x 16 - C		
	Memory Write CAS Latency 11	
	Mode Perister 3	
	Fine Granularity Refresh Mode Normal mode (Fixed 1x)	
	Mode Register 4	
	Temperature Refresh Range Normal 💌	
	Temperature Refresh Mode Disabled 💌	
	Internal VRef Monitor Disabled 💌	
	Self Refresh Abort Mode Disabled 💌	
	READ Preamble 2CK	
	WRITE Preamble 1CK	
Apply New preset		
	Mode Register 5	
	CA Parity Latency Mode Disabled 💌	
	ODT Input Buffer for Power-down Disabled	
	🗀 Mode Register 6	
	Vist College Dense Dense (CON CO. 50)	
	vret Calibration Range Range 1(60% - 92.5%)	
	Vest California Value	
	vrer Calibration Value 60	

The following figure shows timing configuration settings for the DDR4 memory.

Figure 17 • DDR4 Configurator—Memory Timing

DDR4			
Microsemi:SystemBuilder:PF_DDR4:2.3.108			
Q	General Memory Initial	ization Memory Timing	Controller Misc.
PF_DDR4_UI_default_configuration	Timing parameters dependent	andent on speed bin	
∃ JEDEC III. 2 Gb x 8			
+ 2 Gb x 8	tRAS (ns) 34		
+ 4 Gb x 8	,		
⊕ 4 Gb x 16	tRCD (ns) 13.92		
	11002		
🕀 8 Gb x 16	tPP (ps) 12.02		
	13.52		
⊡ 16 Gb x 16	100 ()		
DDR4-16000 (10-10-10) DDR4-1600K (11-11-11)	tRC (ns) 47.92		
DDR4-1600L (12-12-12)			
DDR4-1600L (12-12-12) - 16 Gb x 16 - C	tWR (ns) 20.0		
DDR4-1600L (12-12-12) - 16 Gb x 16 - C			
DDR4-1600L (12-12-12) - 16 Gb x 16 - C	tCCD_L (cycles) 5		
DDR4-1600L (12-12-12) - 16 Gb x 16 - C			
DDR4-1600L (12-12-12) - 16 Gb x 16 - C	tCCD S (cycles) 4		
DDR4-1600L (12-12-12) - 16 Gb x 16 - C			
	Timing parameters dependent	endent on operating condition	
	tREFI (us) 7.8		
	,		
		endent on speed bin and page siz	ze
		indent on speed bin and page su	~
	1050 ()		
	tRFC (ns) 350.0		
	tFAW (ns) 20		
	Timing parameters dependent	endent on speed bin and clock fre	equency
	tWTR_L (cycles) 6		
	tWTR_S (cycles) 2		
	,		
	tRRD L (cycles) 5		
	0000 <u>1</u> 0(0)000) 0		
	tDDD_S_(system)		
	tkkD_5 (cycles) 4		
	tRTP (ns) 7.5		
	Other Timing parameter	5	
	t70init (ourloo)	1024	
Andre Damana I	(zqinit (cycles)	1024	
Apply New preset			
	2Q Calibration Type	Short	
	tZQCS (cycles)	128	
	tZQoper (cycles)	512	
	Enable User ZQ Calibration	Controls	
	-		
	Automatic ZO Calibration Pe	eriod (us) 200	
		()	

The following figure shows controller configuration settings for the DDR4 memory.

Figure 18 • DDR4 Configurator—Controller

DDR4		
Microsemi:SystemBuilder:PF_DDR4:2.3.108		•
	General Memory Initialization Memory Timing Controller Misc.	
PF_DDR4_UI_default_configuration	Instance Select	
the 2 Gb x 8		
1 2 Gb x 16	Instance Number 0 🔻	
+ 4 Gb x 8		
+ 4 Gb x 16		
+ 8 Gb x 8		
. B Gb x 16		
. 16 Gb x 8	Fabric Interface AXI4	
⊟ 16 Gb x 16		
DDR4-1600J (10-10-10)	AXI Width 64	
DDR4-1600K (11-11-11)		
DDR4-1600L (12-12-12)		
DDR4-1600L (12-12-12) - 16 Gb x 16 - C	AXI ID Width 6	
DDR4-1600L (12-12-12) - 16 Gb x 16 - C		
DDR4-1600L (12-12-12) - 16 Gb x 16 - C	Low Power	
DDR4-1600L (12-12-12) - 16 Gb x 16 - C		
DDR4-1600L (12-12-12) - 16 Gb x 16 - C	Cashla Ulasa Dawa Dawa	
DDR4-1600L (12-12-12) - 16 Gb x 16 - C	Enable User Power Down	
	Efficiency	
	Enable Activate/Precharge look-abead	
	Address Ordering	
	ODT Activation Settings on write	
	Enable Rank0 - ODT0 V Enable Rank0 - ODT1 V	
	Enable Rank1 - ODT0 🔽 Enable Rank1 - ODT1 🔽	
test. I November 1	ODT Activation Settings on Read	
Apply New preset		
	Enable Rank0 - ODT0 🗍 Enable Rank0 - ODT1 🗍	
	Enable Pank1 - ODTO Enable Pank1 - ODT1 E	
	7	
	⊡ Misc	
	Enable RE-INIT Controls	
		-
		Ŀ

The following figure shows miscellaneous configuration settings for the DDR4 memory.

Figure 19 • DDR4 Configurator—Misc

	General Memory Initialization Memory Timing Controller Misc.
PF_DDR4_UI_default_configuration JEDEC ⊕ 2 Gb x 16 ⊕ 4 Gb x 16 ⊕ 8 Gb x 16 ⊕ 16 Gb x 8 ⊕ 16 Gb x 16 ⊕ 16 Gb x 16 ⊕ 16 Gb x 16 ⊕ DDR4-1600J (10-10-10) ⊕ DDR4-1600K (11-11-11) ⊕ DDR4-1600K (12-12-12)	Simulation Options Simulation Mode Fast (skip training and settling time)
Apply New preset	

2.3.3.5 CoreAXI4DMAController IP

The CoreAXI4DMAController IP is configured for 64-bit AXI4 data width, and to generate interrupts for descriptor0 and descriptor1. Descriptor0 is used for DDR4 to LSRAM DMA and descriptor1 is used for LSRAM to DDR4 DMA. The following figure shows the configuration settings for the CoreAXI4DMAController IP.

Figure 20 • CoreAXI4DMAController IP Configurator

Configurator	_		×
CoreAXI4DMAController Configurator			
Microsemi:DirectCore:COREAXI4DMACONTROLLER:2.0.100			
Configuration			-
DMA Configuration			
AXI4 data width 64			
AXI4 ID width 4			
AXI4-Stream support			
Arbiter Configuration			
Number of priority levels			
Priority 0 Maximum transfers 256			
Priority 1 Maximum transfers 128 👻			
Priority 2 Maximum transfers 64 📃			
Priority 3 Maximum transfers 32 💌			
Priority 4 Maximum transfers 16 📃			
Priority 5 Maximum transfers 8			
Priority 6 Maximum transfers 4			
Priority 7 Maximum transfers 1			
-Interrupt Configuration			
Number of interrupt outputs 2			
Interrupt 0 queue depth 1			
Interrupt 1 queue depth 1			
Interrupt 2 queue depth 1			
Interrupt 3 queue depth 1			
Descriptor Configuration			
Help •	OK	Cano	el

2.3.3.6 CoreAXI4Interconnect IP

The CoreAXI4Interconnect IP is configured for the following master and slave ports:

- Master0: PCIe
- Master1: CoreAXI4DMAController IP
- Slave0: AXItoAPB bridge (0x0000_0000 to 0x0FFF_FFFF)
- Slave1: AXI Slave Fabric Registers (0x1000 0000 to 0x1FFF FFFF)
- Slave2: AXI4 LSRAM (0x3000 0000 to 0x3FFF FFFF)
- Slave3: DDR4 Subsystem (0x4000_0000 to 0x4FFF_FFFF)

Slave0 is configured to convert AXI4 transactions to AXI3 transactions. The following figure show the CoreAXI4Interconnect IP configurations.

Figure 21 • CoreAXI4Interconnect IP Core Configuration

Configurator	-		×									
CoroAVI/Interconnect Configurator												
PICPOSEMEDIPECCORECCIREAX141NTERCONNECT2.5.100												
Configuration Master Configuration Slave Configuration Crossbar Configuration			<u> </u>									
Bus Configuration			\vdash									
Number of Masters: 3 Number of Slaves: 4												
ID Width: 4 Address Width: 32												
User Width:												
Other Configuration												
Number of Threads: 1 Max Outstanding Transactions: 8	•											
Slave FIFO Address Depth: 4 Slave FIFO Data Depth: 4	_											
DWC Address FIFO Depth Ceiling 10 Read Arbitration Enable:			•									
Help 👻	к	Cancel										

The following figure show the CoreAXI4Interconnect IP master configurations.

Figure 22 • CoreAXI4Interconnect IP Master Configuration

Configurator	-		×
CoreAXI4Interconnect Configurator			
Microsemi:DirectCore:COREAXI4INTERCONNECT:2.5.100			
Configuration Master Configuration Slave Configuration Crossbar Configuration			E
Master0 Configuration			-
M0 Type: AXI4 💌 M0 Data Width: 64 💌			
M0 DWC Data FIFO Depth: 16 💌 M0 Register Slice: 🔽			
M0 Clock Domain Crossing:			
Master1 Configuration			
M1 Type: AXI4 💌 M1 Data Width: 64 💌			
M1 DWC Data FIFO Depth: 16 💌 M1 Register Slice: 🔽			
M1 Clock Domain Crossing:			
Help •	ОК	Car	icel

The following figure show the CoreAXI4Interconnect IP slave configurations.

Configurator	-		×
CoreAXI4Interconnect Configurator			
Microsemi:DirectCore:COREAX14INTERCONNECT:2.5.100			
Configuration Master Configuration Slave Configuration Crossbar Configuration			-
Slave0 Configuration			
S0 Type: AXI3 💌 S0 Data Width: 64 💌			
S0 DWC Data FIFO Depth: 16 S0 Register Slice:			
S0 SLAVE Start Address: 0x0 S0 SLAVE End Address: 0xffffff			
S0 Clock Domain Crossing: 🔽			
□ Slave1 Configuration			
S1 Type: AXI4 💌 S1 Data Width: 64 💌			
S1 DWC Data FIFO Depth: 16 S1 Register Slice: 🔽			
S1 SLAVE Start Address: 0x10000000 S1 SLAVE End Address: 0x1fffffff			
S1 Clock Domain Crossing:			
Slave2 Configuration			
S2 Type: AXI4 💌 S2 Data Width: 64 💌			
S2 DWC Data FIFO Depth: 16 S2 Register Slice:			
S2 SLAVE Start Address: 0x30000000 S2 SLAVE End Address: 0x3ffffff			
S2 Clock Domain Crossing:			
Slave3 Configuration			
S3 Type: AXI4 💌 S3 Data Width: 64 💌			
S3 DWC Data FIFO Depth: 16 S3 Register Slice:			
S3 SLAVE Start Address: 0x40000000 S3 SLAVE End Address: 0x4ffffff			
S3 Clock Domain Crossing:			
Slave4 Configuration			
S4 Type: AXI4 💌 S4 Data Width: 64 💌			
S4 DWC Data FIFO Depth: 16 S4 Register Slice:			
Help Y	ОК	Canc	el

The CoreAXI4Interconnect IP is designed for high bandwidth data movement. It supports bus protocol and bus width converters for each master and slave interface.

2.3.3.7 PolarFire SRAM IP

The SRAM IP is configured to access the 4 KB fabric memory (LSRAM) using the AXI4 interface.

Figure 24 • PolarFire SRAM IP Configurator

Memory Indultation Settings Memory Settings	
Optimize for High Speed Low power SRAM type SRAM Memory Depth(in words) 512 In Use Native Interface	
Interface Settings Fabric Interface type AXI Data Width 64 AXI4 interface options Address Width 32 Width of ID 6 Image: Write Interface Image: Read Interface Image: Wrap Burst support	SRAM_AHB_AXI_UI_O

2.4 Clocking Structure

The following figure shows the clocking structure of PCIe EndPoint reference design.

- Clock Domain 1: generates PCIe TL_CLK. At power-up, it uses 80 MHz clock and switches to 125 MHz after completion of PCIe initialization using NGMUX.
- Clock Domain 2: generates CDR reference and XCVR clocks for PCIe.
- Clock Domain 3: generates 50 MHz clock for PCIe APB, DDR4 PLL reference. DDR4 subsystem generates a 200 MHz clock for fabric AXI interface logic.

2.5 Reset Structure

The CoreReset_PF synchronizes the external USER_RESETN (**SW2** on PolarFire Splash kit) to DDR4 system clock (200 MHz) and generates the FABRIC_RESET_N, which drives the fabric AXI interface logic. CoreReset_PF uses the DEVICE_INIT_DONE signal, which is asserted when the device initialization is complete. For more information about device initialization, see *UG0725: PolarFire FPGA Device Power-Up and Resets User Guide*.

For more information on CoreReset_PF IP core, see CoreReset_PF handbook from the Libero catalog.

The DDR4 subsystem does not require a synchronization reset as it has the reset synchronization logic. The following figure shows the reset structure in the reference design.

Figure 26 • Reset Structure

2.6 Throughput Measurement

The fabric logic uses 32-bit counters to count the number of clock cycles in each DMA transfer. The host PC application starts these counters while initiating the DMA transfers, and the fabric logic stops these counters at the end of the DMA transfer. The DMA Engine interrupts the host PC at the end of the DMA transfer and the host PC application reads the counters to calculate throughput as follows:

Throughput = Transfer Size (Byte) × Clock Frequency/Number of clock cycles taken for a transfer

The throughput includes all of the overhead of the AXI, PCIe, and DMA controller transactions.

2.7 Simulating the Design

Before you start:

- 1. Start Libero SoC PolarFire, and in the **Project** menu, click **Open Project**.
- 2. Browse the Libero Project > PCIe_EP_Demo Libero project folder and open the PCIe EP Demo.prjx file. The PolarFire PCIe EndPoint project opens.
- 3. Open the **Design Hierarchy** window and double-click the **PCIe_EP_Demo** component. The SmartDesign page opens on the right pane and displays the high-level design. You can view the design blocks and IP cores instantiated for the PCIe EndPoint interface design.
- 4. Download the PF_XCVR_REF_CLK, PF_TX_PLL, PF_CCC, PF_PCIE, CoreAXI4Interconnect, CoreAXI4DMAController, DDR4, CoreAHBLite, CoreAPB, CoreAXItoAHBL, CoreAHBLtoAPB, CoreUART, and PolarFire SRAM IP cores under Libero SoC PolarFire > Catalog.

The PCIe BFM performs 1 KB DMA operations between PCIe and DDR4/LSRAM memories by initiating AXI burst transactions. The PCIe BFM simulation model replaces the entire PCIe EndPoint interface with a simple BFM that can send write transactions and read transactions over the AXI interface. These transactions are driven by a script file (.bfm) and allow easy simulation of the FPGA design connected to a PCIe interface. For more information about BFM commands, see *UG0685: PolarFire FPGA PCI Express User Guide*. The micron DDR4 memory models are instantiated in the testbench for simulating DDR4 memory controllers.

Note: In the Design Flow tab, system verilog is selected, as the memory models from Micron are in the system verilog.

In the **Project settings > Design Flow** tab, double-click **Simulate** under **Verify Pre-Synthesized Design** to simulate the design, as shown in the following figure. The ModelSim tool takes about 10 to 15 minutes to complete the simulation.

Figure 27 • Simulating the Design

2.7.1 Simulation Flow

The following steps describe the PCIe BFM simulation flow:

- 1. At the start, the NSYSREST signal, reset all the components.
- 2. DDR4 memory controllers initializes the DDR4 memories and release the CTRLR_READY.
- 3. The PCIe BFM starts executing the BFM script PCIex4 PCIex4 0 PF PCIE PCIE 1 user.bfm.
- 4. The PCIe EndPoint AXI4 master interface initiates write and read burst transactions to SRAM AXI 0, DDR4 through CoreAXI4Interconnect as per the .bfm script.
- 5. After 13 μs, the simulation completes. **PCIE1 BFM Simulation Complete 272 Instructions NO ERRORS** message is highlighted, as shown in Figure 28, page 26.

The ModelSim transcript window displays the BFM commands execution messages, as shown in the following figure. For more information about BFM commands, see the *SmartFusion2 FPGA Microcontroller Subsystem BFM Simulation User Guide*.

Figure 28 • Simulation Transcript Window

- #	SFM:	Data	Read	30000080	0000002100000022	at	12526.250000ns	
ŧ	SFM:	Data	Read	30000088	0000002300000024	at	12531.250000ns	
ŧ	SFM:	Data	Read	30000090	0000002500000026	at	12536.250000ns	
ŧ	SFM:	Data	Read	30000098	0000002700000028	at	12541.250000ns	
ŧ	SFM:	Data	Read	300000a0	000000290000002a	at	12546.250000ns	
ŧ	SFM:	Data	Read	300000a8	0000002b0000002c	at	12551.250000ns	
ŧ	SFM:	Data	Read	300000b0	0000002d0000002e	at	12556.250000ns	
ŧ	SFM:	Data	Read	300000b8	0000002f00000030	at	12561.250000ns	
ŧ	SFM:	Data	Read	300000c0	0000003100000032	at	12566.250000ns	
ŧ	SFM:	Data	Read	300000c8	0000003300000034	at	12571.250000ns	
ŧ	SFM:	Data	Read	300000d0	0000003500000036	at	12576.250000ns	
ŧ	SFM:	Data	Read	300000d8	0000003700000038	at	12581.250000ns	
- #	SFM:	Data	Read	300000e0	000000390000003a	at	12586.250000ns	
ŧ								
ŧ	1	DMA TI	RANSFI	ER DONE (FF	ROM FABRIC ADDRES	s s	PACE TO PCIe ADDRE:	SS SPACE)
+++++++++++++++++++++++++++++++++++++++]	MA T	RANSFI	ER DONE (FF	ROM FABRIC ADDRES	s s	PACE TO PCIe ADDRE	35 SPACE)
* * * *	SFM:	DMA T	RANSFI Read	ER DONE (FF	ROM FABRIC ADDRES	S S at	PACE TO PCIe ADDRE: 12591.250000ns	55 SPACE)
* * * * *	SFM: BFM:	DMA T Data 203:w	RANSFI Read ait 1	ER DONE(FF	COM FABRIC ADDRES	S S at	PACE TO PCIe ADDRE:	55 SPACE)
* * * * * * *	SFM: BFM: SFM: BFM:	DATA Data 203:w Data	RANSFI Read ait 1 Read	ER DONE (FF 300000e8 starting 300000f0	COM FABRIC ADDRES 0000003b0000003c g at 12596 ns 0000003d0000003e	at	PACE TO PCIe ADDRES 12591.250000ns 12596.250000ns	55 SPACE)
* * * * * * * *	SFM: BFM: SFM: BFM: BFM:	DATA Data 203:w Data ****	RANSFI Read ait 1 Read	ER DONE (FF 300000e8 starting 300000f0	COM FABRIC ADDRES 0000003b0000003c g at 12596 ns 0000003d0000003e *******************************	s s at at PC	PACE TO PCIe ADDRE 12591.250000ns 12596.250000ns Ie BFM Simulation*	SS SPACE)
* * * * * * * * *	SFM: BFM: SFM: BFM: BFM: SFM:	DATA Data 203:w Data ****	RANSFI Read ait 1 Read *****	2R DONE (FF 300000e8 starting 300000f0	COM FABRIC ADDRES	at at PC	PACE TO PCIe ADDRE: 12591.250000ns 12596.250000ns Ie BFM Simulation*1	SS SPACE)
* * * * * * * * * *	SFM: BFM: SFM: BFM: BFM: SFM:	DATA Data 203:w Data **** 206:r Data	RANSF Read ait 1 Read ***** eturn Read	ER DONE (FF 300000e8 starting 300000f0	COM FABRIC ADDRES 0000003b0000003c j at 12596 ns 0000003d0000003e **********End of 0000003f00000040	at at PC	PACE TO PCIe ADDRE 12591.250000ns 12596.250000ns Ie BFM Simulation* 12601.250000ns	SS SPACE)
* * * * * * * * * * *	SFM: BFM: SFM: BFM: BFM: SFM: SFM:	DATA Data 203:w Data **** 206:r Data	RANSFI Read ait 1 Read ***** eturn Read #####	CR DONE (FF 300000e8 starting 300000f0 ********* 300000f8	COM FABRIC ADDRES 0000003b0000003c 1 at 12596 ns 0000003d0000003e **********End of 0000003f00000040	at at PC at	PACE TO PCIe ADDRE: 12591.250000ns 12596.250000ns Ie BFM Simulation*: 12601.250000ns	SS SPACE)
*********	SFM: BFM: SFM: BFM: SFM: SFM: PCIE	DMA TI Data 203:w Data **** 206:r Data	RANSFI Read ait 1 Read ***** Read ##### Simu	CR DONE (FF 300000e8 starting 300000f0 ********* 300000f8	<pre>KOM FABRIC ADDRES 0000003b0000003c y at 12596 ns 0000003d0000003e *******************************</pre>	at at PC at	PACE TO PCIe ADDRE: 12591.250000ns 12596.250000ns Ie BFM Simulation*: 12601.250000ns	SS SPACE)
**********	SFM: BFM: SFM: BFM: SFM: SFM: PCIE	DMA TI Data 203:w Data **** 206:r Data #####	RANSFI Read ait 1 Read ***** eturn Read ##### Simu	ER DONE (FF 300000e8 starting 300000f0 ********* 300000f8	COM FABRIC ADDRES 0000003b0000003c y at 12596 ns 0000003d0000003e **********End of 0000003f00000040	at at PC at ###	PACE TO PCIe ADDRE: 12591.250000ns 12596.250000ns Te BFM Simulation* 12601.250000ns tions - NO ERRORS	55 SPACE)
**********	SFM: BFM: SFM: BFM: SFM: SFM: PCIE	DMA T) Data 203:w Data **** 206:r Data #####	RANSFI Read ait 1 Read ***** Read ##### Simu:	ER DONE (FF 300000e8 starting 300000f0 ********* 300000f8	<pre>ROM FABRIC ADDRES 0000003b0000003c j at 12596 ns 0000003d0000003e *********End of 0000003f00000040 plete - 272 Inst</pre>	 S S: at PC at ###	PACE TO PCIe ADDRE: 12591.250000ns 12596.250000ns Ie BFM Simulation*: 12601.250000ns tions - NO ERRORS	SS SPACE)

The following figure shows the actual waveform window showing the sequence of data being written and read using the BFM.

Figure 29 • Simulation Waveform Window

A SYSCI K	0	anananinanan	anninanananinanananan	no no mino a maine a mana a	niamanianianianianianianiania	ninonononinomonino	nanananinanananinananan
DDR4 CTRLR READY	1						
	-						
- PCIE AXI Interface							
AXI_CLK_STABLE	1						
AXI_CLK	0						
AXI Write Address Channel							
PCIESS_AXI_1_M_AWID	0	0					
PCIESS_AXI_1_M_AWADDR	10000040	00000000	[] 1](0		(00000000) eo) ococococ	X3 X00000000
PCIESS_AXI_1_M_AWBURST	1	0	I1 X0		Io	χ, χο	l1 l0
PCIESS_AXI_1_M_AWLEN	00	00		1 100		X f X00	(1f (00
🕞 🔷 PCIESS_AXI_1_M_AWSIZE	2	0	[2]0	12	χo.	χε χο	X3 X0
PCIESS_AXI_1_M_AWVALID	0		1111				
PCIESS_AXI_1_M_AWREADY	1						
AXI Write Data Channel							
💿 🔷 PCIESS_AXI_1_M_WDATA	00000000000	000000000000000000000000000000000000000	I X I X X0			χ χ χ σοσοσοσοσοσο	00 1 0000000000000000000000000000000000
PCIESS_AXI_1_M_WLAST	0		1111				
PCIESS_AXI_1_M_WSTRB	Of	00	lof	f (of		χf	
PCIESS_AXI_1_M_WVALID	0		1111				
PCIESS_AXI_1_M_WREADY	1					n	1
AXI Write Response Channel							
PCIESS_AXI_1_M_BID	0	0					
PCIESS_AXI_1_M_BRESP	0	0					
PCIESS_AXI_1_M_BVALID	0						
PCIESS_AXI_1_M_BREADY	0						
AXI Read Address Channel							
PCIESS_AXI_1_M_ARID	0	0					
PCIESS_AXI_1_M_ARLEN	00	00)(01	χο χ1f	100 X 1f
PCIESS_AXI_1_M_ARSIZE	0	0) 2		(3	χ) χз	<u>(0 (3</u>
PCIESS_AXI_1_M_ARBURST	0	1	[0)[1		X1	X) X1	10 X1
PCIESS_AXI_1_M_ARVALID	0						
PCIESS_AXI_1_M_ARADDR	00000000	00000000		0000000		(X 0 X 40000000	X0 X 30000000
PCIESS_AXI_1_M_ARREADY	1						
AXI Read Data Channel			Read and Write	PCIe Read and Write Transactions to LSRAM		PCle	DMA Transactions
PCIESS_AXI_1_M_RID	0	0	Transactions to				
PCIESS_AXI_1_M_RVALID	0		Demo Registers				
PCIESS_AXI_1_M_RREADY	0						
PCIESS_AXI_1_M_RLAST	1					11	
🕞 🧼 PCIESS_AXI_1_M_RDATA	00000000000	000000000000000000000000000000000000000	XXX	0000000000000		6789acc789abcee	000 0000003f00000040
PCIESS AXI 1 M RRESP	0	0				-	

Figure 30 • Simulation Waveform Window

_	DDR4 SDRAM Interface													
	DDR4_ACT_N	1												
	DDR4_CAS_N	1									J		יששענות היות היות היות	
	DDR4_CKE	1												
	DDR4_CK	0												
	DDR4_CK_N	1												
	DDR4_CS_N	1									- m			
	DDR4_ODT	0											.m	
	DDR4_RAS_N	1												
	DDR4_RESET_N	1												
3	DDR4_WE_N	1									U		,IU	
+	DDR4_ADDR	0000	0000								X 0000		0_000000000000000	0000
	DDR4_BA	0	0											
	DDR4_BG	0												
+	DDR4_DM_N	z										 	10	
•	DDR4_DQS	z			 	 						 		
•	DDR4_DQS_N	z		 		 						 	<u> 2000000</u>	
	DDR4_DQ	2222				DDR	4 Read and	Write Trans	actions on D	DR4 SDRA	M Interface			
_	Throughput counters													

3 Libero Design Flow

The Libero design flow involves the following steps:

- Synthesize
- Place and route
- Verify timing
- Generate Bitstream
- Run PROGRAM Action

3.1 Synthesize

Go to the **Design Flow** window and double-click **Synthesize**.

When the synthesis is successful, a green tick mark appears as shown in the following figure.

Figure 31 • Synthesize

3.1.1 Resource Utilization

The following table lists the resource utilization of the PCIe Endpoint design. These values may vary slightly for different Libero runs, settings, and seed values.

Туре	Used	Total	Percentage
4LUT	29761	299544	9.94
DFF	21840	299544	7.29
I/O Register	0	732	0.00
User I/O	75	244	30.74
– Single-ended I/O	69	244	28.28
– Differential I/O Pairs	3	122	2.46
μSRAM	168	2772	6.06
LSRAM	27	952	2.84

Table 2 • Resource Utilization

Туре	Used	Total	Percentage
Math	0	924	0.00
H-Clip Global	8	48	16.67
PLL	1	8	12.50
DLL	1	8	12.50
CRN_INT	1	24	4.17
INIT	1	1	100.00
DRI	1	1	100.00
OSC_RC160MHZ	1	1	100.00
Transceiver Lanes	4	8	50.00
Transceiver PCIe	1	2	50.00
TX_PLL	1	11	9.09
LINK	1	10	10.00
XCVR_REF_CLK	1	6	16.67
PCIE_COMMON	1	1	100.00
ICB_CLKDIV	1	24	4.17
ICB_CLKINT	2	72	2.78
NGMUX	1	12	8.33
ICB_INT	1	12	8.33

Table 2 • Resource Utilization (continued)

3.2 Place and Route

To place and route the design, the TX_PLL, XCVR_REF_CLK, and DDR4 need to be constrained using the **I/O Editor** as shown in the following figures.

Figure 33 • I/O Editor—DDR4 Memory View

Design View	Me	mory View [active]	Port View	Pin View 🗗
Logical I 1	Men	nory Type: DDR4		
Primitives AXI4_Interconnect_0		Port Function	🕇 🛛 Port Name 💌	Pin Number 💌
	1	NORTH_NE	Assigned	
	98	NORTH_NW	PF_DDR4_SS_0	

Go to the **Design Flow** window and double-click **Place and Route**. When place and route is successful, a green tick mark appears as shown in the following figure.

Figure 34 • Place and Route

3.3 Verify Timing

Go to the **Design Flow** window and double-click **Verify Timing**. When the design successfully meets the timing requirements, a green tick mark appears as shown in the following figure.

Figure 35 • Design Flow

3.4 Generate Bitstream

To generate the bitstream:

- 1. Double-click **Generate Bitstream** from the **Design Flow** tab. When the bitstream is successfully generated, a green tick mark appears as shown in Figure 37, page 30.
- 2. Right-click **Generate Bitstream** and select **View Report** to view the corresponding log file in the **Reports** tab.

3.5 Run PROGRAM Action

After generating the bitstream, the PolarFire device must be programmed. Follow these steps to program the PolarFire device:

1. Ensure that the jumper settings on the board are the same as those listed in the following table.

Jumper	Description
J5, J6, J7, J8, J9	Short pin 2 and 3 for programming the PolarFire FPGA through FTDI
J11	Short pin 1 and 2 for programming through the FTDI chip
J10	Short pin 1 and 2 for programming through the FTDI SPI
J4	Short pin 1 and 2 for manual power switching using SW1
J3	Open pin 1 and 2 for 1.0 V

Table 3 • Jumper Settings

- 2. Connect the power supply cable to the **J2** connector on the board.
- 3. Connect the USB cable from the Host PC to **J1** (FTDI port) on the board.
- 4. Power on the board using the **SW1** slide switch.

Figure 36 • Board Setup

5. Double-click Run PROGRAM Action from the Libero > Design Flow tab.

When the device is programmed successfully, a green tick mark appears as shown in the following figure. See Running the Demo, page 32 to run the PCIe EndPoint demo.

Figure 37 • Programming the Device

4 **Programming the Device Using FlashPro**

This section describes how to program the PolarFire device with the .stp programming file using FlashPro. The .stp file is available at the following design files folder location:

mpf_dg0808_liberosocpolarfirev2p2_df\ProgrammingFile

To program the PolarFire device using FlashPro, complete the following steps:

1. Ensure that the jumper settings on the board are the same as those listed in the following table. **Note:** The power supply switch must be switched off while making the jumper connections.

Jumper	Description
J5, J6, J7, J8, J9	Short pin 2 and 3 for programming the PolarFire FPGA through FTDI
J11	Short pin 1 and 2 for programming through the FTDI chip
J10	Short pin 1 and 2 for programming through the FTDI SPI
J4	Short pin 1 and 2 for manual power switching using SW1
J3	Open pin 1 and 2 for 1.0 V

Table 4 • Jumper Settings

2. Connect the power supply cable to the **J2** connector on the board.

- 3. Connect the USB cable from the Host PC to the J1 (FTDI port) on the board.
- 4. Power on the board using the SW1 slide switch.

The following figure shows the board setup of the PolarFire Splash Kit.

Figure 38 • Board Setup

- 5. On the host PC, launch the FlashPro software.
- 6. Click **New Project** to create a new project. In the New Project window, enter a project name.
- 7. Click Browse and navigate to the location where you want to save the project.
- 8. Select Single device as the programming mode and click OK to save the project.
- 9. Click Configure Device.
- 10. Click **Browse**, and navigate to the location where the PCIe_EP_Demo.stp file is located and select the file. The default location is:

<download_folder>\mpf_dg0808_liberosocpolarfirev2p2_df\ProgrammingFile

- 11. Click **Open**. The required programming file is selected and ready to be programmed in the device.
- 12. Click **PROGRAM** to program the device.

When the device is programmed successfully, a **Run PASSED** status is displayed. See Running the Demo, page 32 to run the PCIe EndPoint demo.

5 Running the Demo

This section describes how to install and use the PCIe Demo application. The PolarFire PCIe demo application is a simple graphic user interface (GUI) that runs on the host PC to communicate with the PolarFire PCIe EndPoint device. It provides PCIe link status, driver information, and demo controls. The PolarFire PCIe demo application invokes the PCIe driver installed on the host PC and provides commands to the driver according to the selection made.

This section also describes how to connect the kit to the Host PC PCIe Slot. If the host PC PCIe slot is not available, the DMA between DDR4 and LSRAM can be exercised through UART IF.

5.1 Installing PCIe Demo Application

To install the PolarFire PCIe Demo application:

- 1. Install the GUI_Installer (setup.exe) from the following design files folder: mpf_dg0808_liberosocpolarfirev2p2_df\GUI_Installer.
- 2. Double-click the setup.exe in the provided GUI installation (GUI_Installer\setup.exe).
- 3. Apply default options as shown in the following figure.

Figure 39 • Installing PCIe Demo Application

🐙 PolarFire_PCIe_GUI		x
Destination Directory Select the primary installation directory.		
All software will be installed in the following locations. To install software into a different location, click the Browse button and select another directory.		
Directory for PolarFire_PCIe_GUI C.\Program Files (x86)\PolarFire_PCIe_Demo\ Br	owse]
Directory for National Instruments products C:\Program Files (x86)\National Instruments\ Br	owse]
<pre></pre>	Cano	cel

4. Click **Next** to start the installation.

Figure 40 • PCIe Demo Application Installation Steps

🕼 PolarFire_PCIe_GUI	
Start Installation Review the following summary before continuing,	
Adding or Changing • NI-VISA 14.0.1 Run Time Support	
Click the Next button to begin installation. Click the Back button to change the installation settings.	
Save File) << Back Next >>	Cancel

5. Click **Finish** to complete the installation.

Figure 41 • Successful Installation of PCIe Demo Application

PolarFire_PCIe_GUI	
Installation Complete	
The installer has finished updating your system.	
< Back 1	Vext >> Finish

5.2 Running the Demo Through PCle

This section shows how to connect the board to host PC PCIe slot, installing the PCIe drivers and running the demo application.

5.2.1 Connecting the Board to the Host PC PCIe Slot

- 1. After successful programming, power OFF the PolarFire Splash board and shut down the host PC.
- Connect the CON3 PCIe Edge connector of the PolarFire Splash board to the host PC's PCIe slot through the PCI Edge card ribbon cable.
 This demo is designed to work with any PCIe Gen 2 compliant slot. If the host PC does not support Gen 2 compliant slot, the demo switches to Gen 1 mode.
- **Note:** Power OFF the host PC while inserting the PCIe Edge connector. If it is not powered OFF, the PCIe device detection and the selection of Gen1 or Gen2 mode may fail. The device detection and selection depend on the host PC PCIe configuration.
- **Note:** After connecting the board to the host PC, the host PC may power on without manually switching on the PC.
- **Note:** PCIe hot reset is not supported in this version of the demo.

The following figure shows the board setup for the host PC in which PolarFire Splash Kit is connected to the host PC PCIe slot.

Figure 42 • PolarFire Splash Kit Setup for Host PC

3. Power on the power supply switch SW1.

4. Power on the host PC and check the **Device Manager** of the Host PC for the PCIe Device. The following figure shows the example **Device Manager** window.

Figure 43 • Device Manager

Note: If the device is still not detected, check if the BIOS version in the host PC is the latest, and if PCI is enabled in the host PC BIOS.

5.2.2 Driver Installation

Perform the following steps to install the PCIe drivers on the host PC:

1. Right-click **PCI Device** in the **Device Manager** and select **Update Driver Software...** as shown in the following figure. To install the drivers, administrative rights are required.

Figure 44 • Update Driver Software

Bevice Manager	
File Action View Help	
 W764d-test123 Computer Disk drives Diplay adapters DVD/CD-ROM drives Human Interface Devices Keyboards Keyboards Mice and other pointing devices Mice and other pointing devices NoMachine USB Host Adapter NoMachine USB Host Adapter Other devices Port S(CO Update Driver Software Processor Disable Uninstall Scan for hardware changes Viniversal 	
Launches the Update Driver Software Wizard for the selected devi	ce.

Note: Uninstall the existing Microsemi PolarFire drivers on the host PC before proceeding to next step.

2. In the Update Driver Software - PCle Device window, select Browse my computer for driver software as shown in the following figure.

Figure 45 • Browse for Driver Software

😡 🛯 Update Driver Software - PCI Device	×
How do you want to search for driver software?	
Search automatically for updated driver software Windows will search your computer and the Internet for the latest driver software for your device, unless you've disabled this feature in your device installation settings.	
Browse my computer for driver software Locate and install driver software manually.	
	Cancel

3. Browse the drivers folder:

mpf_dg0808_liberosocpolarfirev2p2_df\PCIe_Drivers\Win_64bit_PCIe_Driver and click **Next** as shown in the following figure.

Figure 46 • Browse for Driver Software Continued

- 4. The **Windows Security** dialog box is displayed. Click **Install** as shown in the following figure. After successful driver installation, a message appears. See Figure 48, page 37.
- Figure 47 Windows Security

Figure 48 • Successful Driver Installation

5.2.3 Running the PCIe Demo Application

The following steps describe how to run the demo design:

- 1. Click to expand the **PolarFire PCIe** device in the host PC **Device Manager** as shown in the following figure.
- Figure 49 Device Manager—PCle Device Detection

Note: If a warning message is displayed for PolarFire PCIe driver while accessing, uninstall and re-install the driver.

2. Go to All Programs > PolarFire_PCle_GUI > PolarFire_PCle_GUI. The PolarFire PCle Demo window is displayed as shown in the following figure.

Figure 50 • PCIe EndPoint Demo Application

vice Info	Demo Controls	Config Space	PCIe Read/Write	DMA Operati	ions UART	
[Device Vendor ID			Number of	Bars 0	
	Device Type		BAR	Address x 0	BAR0 Size	(Bytes) x 0
	Driver Version		BARI	Address x 0	BAR1 Size	(Bytes) x 0
Dr	iver Time Stamp		BAR	Address 0	BAR2 Size	(Byter)0
	Demo Type		DANA	XU	DAILY SIZE	(0ytes) XU
5	Supported Width		BAR	Address x 0	BAR3 Size	(Bytes) x 0
N	legotiated Width		BAR4	Address ×0	BAR4 Size	(Bytes) ×0
S	Supported Speed		BAR	Address ×0	BAR5 Size	(Bytes) × 0
N	legotiated Speed					-

3. Click **Connect**. The application detects and displays the information related to the connected kit such as Device Vendor ID, Device Type, Driver Version, Driver Time Stamp, Demo Type, Supported Link Width, Negotiated Link Width, Supported Speed, Negotiated Speed, Number of Bars, and BAR Address as shown in the following figure.

Figure 51 • Device Info

ice Info	Demo Contro	ols Config Space	PCIe Read/Wr	ite DMA	Operations	
I	Device Vendor ID	0x11AA		N	umber of Bars 2	1
	Device Type	PolarFire Splash kit	В	AR0 Address	× E010000C	BAR0 Size(Bytes) ×10000
	Driver Version	6.1.7600.16385	в	AR1 Address	× 0	BAR1 Size(Bytes) × 0
Di	river Time Stamp	03:13:01 14/11/2017				
	Demo Type	PolarFire PCIe Demo	В	AR2 Address	× E000000C	BAR2 Size(Bytes) ×10000
5	Supported Width	x4 (4 lanes)	В	AR3 Address	× 0	BAR3 Size(Bytes) × 0
N	Vegotiated Width	x4 (4 lanes)		ARA Address	×0	BAR4 Size(Bytes) ×0
1	Supported Speed	5 Gbps (Gen 2)		AIN Address		DAIG SIZE(Dytes)
Ν	Vegotiated Speed	5 Gbps (Gen 2)	В	AR5 Address	×O	BAR5 Size(Bytes) ×0

4. Click the **Demo Controls** tab to display the **LED Controls**, **DIP Switch Status**, and **Interrupt Counters** as shown in the following figure.

Figure 52 • Demo Controls

🔊 M	icrosemi	PolarFir	e PCIe Dem	o GUI O PCIe Connected
Device Info	Demo Controls	Config Space	PCIe Read/Write	DMA Operations
LED - LED - LED - LED - LED - LED - LED - LED -	LED Controls 1 2 3 4 5 5 6 7 8 8	ON ON OFF OFF	ON ON OFF OFF	Interrupt Counters No of MSI Requested 4 No of MSI Allocated 4 Interrupt Counter1 Interrupt Counter2 Interrupt Counter3 Interrupt Counter4
Start LE	ED ON/OFF Walk	Enable D	IP SW Session	Enable Interrupt Session
			Exit	

5. Click Start LED ON/OFF Walk, Enable DIP SW Session, and Enable Interrupt Session to view the controlling LEDs (observe LED1 to LED8 on the PolarFire Splash Kit), getting the DIP switch (ON/OFF the DIP1 to DIP4 on the PolarFire Splash Kit) status, and monitoring the interrupts (press SW3 to SW6 on the PolarFire Splash Kit to generate interrupt) simultaneously as shown in the following figure.

Figure 53 • Demo Controls—Continued

C Microsemi	PolarFire PCIe Dem	• GUI • PCIe Connected
Device Info Demo Controls	Config Space PCIe Read/Write	DMA Operations
LED Controls	DIP Switch Status	Interrupt Counters
LED 2	ON ON ON ON	No of MSI Requested 4
LED 3		Interrupt Counter1
LED 5		Interrupt Counter2 1
LED 6	OFF OFF OFF OFF	Interrupt Counter3
LED 7		Interrupt Counter4
Start LED ON/OFF Walk	Enable DIP SW Session	Enable Interrupt Session
Stop LED ON/OFF Walk	Disable DIP SW Session	Clear Interrupt Count
	Exit	

6. Click the **Config Space** tab to view the details about the PCIe configuration space as shown in the following figure.

Figure 54 • Configuration Space

Basic Advanced Extended Capability Type 0 Configur Descriptor Name Offs Vendor ID 0:	ation Settings set (0x) Value(0x)	Configuration Description
Type U Configur Descriptor Name Offs Vendor ID 0;	ation Settings set (0x) Value((x0)	
Descriptor Name Offs Vendor ID 0:	et (0x) Value((0x)	
Vendor ID 0:			
	x000 0x114	AA	
Device ID 0:	x002 0x15	56	
Command 0	x004 0x50	6	
Status 0:	x006 0x10	0	
Revision ID 02	x008 800x)	
Class Code 0:	x009 0x0)	
Cache Line Size 0x	d0C 0x10	D	
Latency Timer 0x	d0D 0x0		
Header Type 02	x00E 0x0) =	
BIST 0	x00F 0x0	P	
Base Address 0 02	x010 0xE0100	000C	
Base Address 1 02	x014 0x0)	
Base Address 2 0	x018 0xE0000	000C	
Base Address 3 0x	d1C 0x0	E	
	x020 0x0)	
Base Address 4 0:			
Base Address 4 0: Base Address 6 0:	x024 0x0		
Base Address 4 03 Base Address 6 03 Expansion ROM Base Address 03	x024 0x0 x028 0x0		
Base Address 4 0: Base Address 6 0: Expansion ROM Base Address 0: Subsystem Vendor ID 0:	x024 0x0 x028 0x0 x02C 0x0		
Base Address 4 00 Base Address 6 00 Expansion ROM Base Address 00 Subsystem Vendor ID 00 Subsystem ID 00	x024 0x0 x028 0x0 x02C 0x0 x02E 0x0		
BIST 00 Base Address 0 00 Base Address 1 00 Base Address 2 00 Base Address 3 00	x00F 0,x0 x010 0,xE0100 x014 0,x0 x018 0,xE0000 x01C 0,x0 x020 0,x0)00C))00C	

- 7. Click the **PCIe Read/Write** tab to perform read and write operations to DDR/LSRAM using BAR2 space.
- 8. Click **Read** to read the 4 KB memory mapped to BAR2 space for DDR and LSRAM as shown in the following figure.

Figure 55 • PCIe BAR2 Memory Access—LSRAM

🔍 М	licrosemi	PolarFire PCI	e Demo GUI	PCIe UART		
Device Info	Demo Controls C	onfig Space PCIe R	ead/Write DMA Operations			
PCIe-BAR2-LSRAM PCIe BAR2-DDR3 PCIe-BAR2-DDR4 PCIe DDR Offset Address ×0						
	0x0	0x4	0x8	0xC 🔺		
0x000	400	3FF	3FE	3FD E		
0x010	3FC	3FB	3FA	3F9		
0x020	3F8	3F7	3F6	3F5		
0x030	3F4	3F3	3F2	3F1		
0x040	3F0	3EF	3EE	3ED		
0x050	3EC	3EB	3EA	3E9		
0x060	3E8	3E7	3E6	3E5		
0x070	3E4	3E3	3E2	3E1		
0x080	3E0	3DF	3DE	3DD		
0x090	3DC	3DB	3DA	3D9		
0x0A0	3D8	3D7	3D6	3D5		
0x0B0	3D4	3D3	3D2	3D1		
0x0C0	3D0	3CF	3CE	3CD		
0x0D0	3CC	3CB	3CA	3C9		
0x0E0	3C8	3C7	3C6	3C5		
0x0F0	3C4	3C3	3C2	3C1		
0x100	3C0	3BF	3BE	3BD		
0x110	3BC	3BB	3BA	389		
0x120	3B8	3B7	3B6	3B5 *		
Read Pr	ogress			Read		
		E	ât			

	Microsemi	PolarFir	e PCIe Dem	no GUI	PCIe Connected		
Device In	fo Demo Controls	Config Space	PCIe Read/Write	DMA Operations			
PCIe-BAR2-LSRAM PCIe BAR2-DDR3 PCIe-BAR2-DDR4 PCIe DDR Offset Address x0							
	0x0		0x4	0x8	0xC 🔺		
0x000	400		3FF	3FE	3FD E		
0x010	3FC		3FB	3FA	3F9		
0x020	3F8		3F7	3F6	3F5		
0x030	3F4		3F3	3F2	3F1		
0x040	3F0		3EF	3EE	3ED		
0x050	3EC		3EB	3EA	3E9		
0x060	3E8		3E7	3E6	3E5		
0x070	3E4		3E3	3E2	3E1		
0x080	3E0		3DF	3DE	3DD		
0x090	3DC		3DB	3DA	3D9		
0x0A0	3D8		3D7	3D6	3D5		
0x0B0	3D4		3D3	3D2	3D1		
0x0C0	3D0		3CF	3CE	3CD		
0x0D0	3CC		3CB	3CA	3C9		
0x0E0	3C8		3C7	3C6	3C5		
0x0F0	3C4		3C3	3C2	3C1		
0x100	3C0		3BF	3BE	3BD		
0x110	3BC		3BB	3BA	3B9		
0x120	3B8		3B7	3B6	3B5 👻		
Read	Read Progress Read						
			Exit				

Figure 56 • PCIe BAR2 Memory Access—DDR4

9. Click the **DMA Operations** tab for different DMA operations such as DDR and LSRAM.

5.2.3.1 Continuous DMA—Operations

The following instructions describe running DMA operations between PC and DDR4, PC and LSRAM:

- 1. Select one of the following options from the DMA Transfer Type Selection drop-down list:
 - PC->DDR4—to transfer the data from host PC to PolarFire DDR4 memory
 - DDR4->PC—to transfer the data from PolarFire DDR4 memory to host PC
 - Both PC<->DDR4—to transfer the data from host PC to and from PolarFire DDR4 memory
 - **PC->LSRAM**—to transfer the data from host PC to PolarFire LSRAM memory
 - LSRAM->PC—to transfer the data from PolarFire LSRAM memory to host PC
 - Both PC<->LSRAM—to transfer the data from host PC to and from PolarFire LSRAM memory
- 2. Select **Transfer Size** (4KB to 64KB) from the drop-down list. Maximum contiguous DMA size is 64KB because the host PC may not have contiguous memory of more than 64 KB. For DMA operations that require more than 64 KB, use SGDMA.
- 3. Enter the **Loop Count** in the box.
- 4. Click **Start Transfer**. After a successful DMA operation, the GUI displays the throughput and average throughput in MBps. The following figure shows Continuous DMA—Operations.

Figure 57 • Continuous DMA—Operations

C Microsem	<i>i</i> PolarFire PCIe	Demo GUI	PCIe Connected
Device Info Demo Controls	Config Space PCIe Read	I/Write DMA Operatio	ns (
PCIe Continuous DMA PCIe SC	GDMA Fabric Core DMA		
Operations Memory Test			
DMA Transfer Type Selection	PC->LSRAM ▼	600 - 550 -	
PC to LSRAM	LSRAM->PC Both PC<->LSRAM	500 - 450 -	
Transfer Size(Bytes) 4K	PC->DDR4 DDR4->PC Both PC<->DDR4	400 - 350 -	
Throughput(MBps) 0	PC->DDR3 DDR3 ->PC Both PC<->DDR3	300 - 250 - 200 -	
Avg Thruput(MBps) 0	Avg Thruput(MBps) 0	150 -	
Loop Count 1	Start Transfer	50 - 0 -	
	Start Transfer	N DC to LEBANA	o of DMA Transfers
		PC to LSRAM PC to LSRAM Avg	LSRAM to PC Avg
	Exit		

Note: The AXI LSRAM in the design is configured for 4KB. This 4KB is over written if more than 4KB of DMA operation is performed on LSRAM. This option is provided to exercise the throughputs with larger DMA size.

The following figure shows the throughput and average throughput in MBps.

Figure 58 • Continuous DMA Operations with DMA Transfer Type Selection as Both PC and LSRAM

5.2.3.2 Continuous DMA—Memory Test

The following instructions describe running Memory Test between PC and DDR4/LSRAM:

- 1. Select one of the following options from the Test Selection drop-down list:
 - PC<->DDR4—to transfer the data from host PC to and from PolarFire DDR4 memory
 - PC<->LSRAM—to transfer the data from host PC to and from PolarFire LSRAM memory
- 2. Select Transfer Size (4KB to 64KB) from the drop-down list.
- 3. Select **Pattern Selection** from the drop-down list—Increment, Decrement, Random, Fill with Zeros, Fill with Ones, Fill with all A's, and Fill with all 5's.

The following figure shows Continuous DMA—Memory Test tab.

Figure 59 • DMA Transfer Type Selection—Continuous Memory Test

🔍 Mic	crosemi	PolarFire	e PCIe De	mo GU	● PCIe ● UART	Connected
Device Info	Demo Controls	Config Space	PCIe Read/Wr	ite DMA O	perations	
PCIe Continuous	DMA PCIe SGD	MA Fabric Core	DMA			
Operations N	femory Test					
Memo	ry Test			View Me	mory	
Test Selectio	'n	Address Offs	et ×0		View Memory	
PC<->DD V PC<->DD PC<->LSI 4K	DR3 DR4 RAM	0x0 0x10 0x20 0x30	0x0	0x4	0x8	0xC
Pattern Select	ction	0x50 0x60 0x70				
Star	rt	0x80 0x90 0xA0 0x80 0xC0 0xD0				
		0xE0				•
			Exit			

- 4. Click Start. GUI performs the following task:
 - The host PC creates a buffer and initializes the memory
 - Initiates the PC to DDR DMA
 - Erases the PC buffer
 - Initializes the DDR to PC DMA
 - · Compares the memory against expected memory

Memory Test Successful window appears, as shown in the following figure.

Figure 60 • Continuous DMA Memory Test—Memory Test Successful

Microsemi. PolarFire PCIe Demo GUI Ourt Connected								
Device Info Demo Controls	Config Spac	e PCIe Read/W	/rite DMA Op	perations				
PCIe Continuous DMA PCIe SGDMA Fabric Core DMA								
Operations Memory Test								
Memory Test			View Men	nory				
Test Selection	Address Offset ×0			View Memory				
PC<->DDR4	C			0x8	0xC			
	0x0			3	4	E		
Transfer Size (Bytes)	0x10 0x20 Memory Test Successfull		coorfull	7	8			
GAV			ccessiun	В	С			
04K	0x30	0x30 0x40		F	10			
	0x40			13	14			
Pattern Selection	0x50	15	10	17	18			
	0x60	19	1A	1B	1C			
Increment 👻	0x70	1D	1E	1F	20			
	0x80	21	22	23	24			
	0x90	25	26	27	28			
	0xA0	29	2A	2B	2C			
Start	0xB0	2D	2E	2F	30			
	0xC0	31	32	33	34			
	0xD0	35	36	37	38			
	0xE0	39	3A	3B	3C	-		
		Exit						

Note: If memory test fails, the GUI displays the first failed memory location.

Note: Change the Offset Address and click View Memory to read the RAM memory content.

5.2.3.3 SGDMA—Operations

The following instructions describe running SGDMA operations between PC and DDR4:

- 1. Select one of the following options from the DMA Transfer Type Selection drop-down list:
 - PC -> DDR4—to transfer the data from host PC to PolarFire DDR4 memory
 - **DDR4-> PC**—to transfer the data from PolarFire DDR4 memory to host PC
 - Both PC <->DDR4—to transfer the data from host PC to and from PolarFire DDR4 memory
- 2. Select **Transfer Size** (4KB to 64KB) from the drop-down list.
- 3. Enter the **Loop Count** in the box. The **Buffer Descriptors** show the number of descriptors created by the host driver for each SGDMA operation.
- 4. Click **Start Transfer**. After a successful DMA operation, the GUI displays the throughput and average throughput in MBps. The following figure shows SGDMA—Operations.

Figure 61 • SGDMA—Operations

C Microsem	i PolarFire PCIe De	emo GUI	PCIe UART Connected
Device Info Demo Controls	Config Space PCIe Read/W	rite DMA Operations	
PCIe Continuous DMA PCIe SC	DMA Fabric Core DMA		
Operations Memory Test			
DMA Transfer Type Selection PC to DDR4 Transfer Size(Bytes) 4K Throughput(MBps) 0 Avg Thruput(MBps) 0	PC->DDR4 ✓ PC->DDR4 DDR4->PC Both PC<->DDR4 PC->DDR3 DDR3->PC Both PC<->DDR3 Througnput(triup)37 0 Avg Thruput(MBps) 0	550 - 500 - 450 - 400 - 350 - 300 - 250 - 200 - 150 -	十 2 10
Buffer Descriptors 0	Buffer Descriptors 0	100 -	
Loop Count 1	Start Transfer	50 - 1 0 - 1 0 No of D	1 2 MA Transfers
		PC to DDR4 📌	DDR4 to PC
		PC to DDR4 Avg 📉	DDR4 to PC Avg 🔼
	Exit		

5.2.3.4 SGDMA—Memory Test

The following instructions describe running Memory Test between PC and DDR4/LSRAM:

- 1. Select one of the following options from the Test Selection drop-down list:
 - PC<->DDR4—to transfer the data from host PC to and from PolarFire DDR4 memory
 Soloct Transfer Size (4KB to 1MB) from the drop down list
- 2. Select **Transfer Size** (4KB to 1MB) from the drop-down list.
- 3. Select **Pattern Selection** from the drop-down list—Increment, Decrement, Random, Fill with Zeros, Fill with Ones, Fill with all A's, and Fill with all 5's.

The following figure shows SGDMA—Memory Test tab.

Figure 62 • SGDMA—Memory Test

C Microsem	PolarFire PCIe D	emo GUI	PCIeUART	Connected					
Device Info Demo Controls	Config Space PCIe Read/	Write DMA Oper	rations						
PCIe Continuous DMA PCIe SGDMA Fabric Core DMA									
Operations Memory Test	Operations Memory Test								
Memory Test		View Mem	ory						
Test Selection	Address Offset ×0	Vie	w Memory						
PC<->DDR4 PC<->DDR3 PC<->DDR4 Increment Transfer Size (Bytes) 4K Start	0x0 0x10 0x20 0x30 0x40 0x50 0x60 0x70 0x80 0x40 0x50 0x60 0x70 0x80 0x40 0x50 0x60 0x70 0x80 0xA0 0xE0 0xE0	0x4	0x8	0xC					
	Exit								

- 4. Click Start. GUI performs the following task:
 - The host PC creates a buffer and initializes the memory
 - Initiates the PC to DDR DMA
 - Erases the PC buffer
 - Initializes the DDR to PC DMA
 - Compares the memory against expected memory

Memory Test Successful window appears, as shown in the following figure.

Figure 63 • SGDMA Memory Test—Memory Test Successful

🌑 М	icrosem	<i>İ</i> Polar	Fire PCIe I	Demo GUI	PCIeUART	Connec	ted
Device Info	Demo Controls	Config Sp	ace PCIe Read,	/Write DMA Ope	rations		
PCIe Continuous DMA PCIe SGDMA Fabric Core DMA							
Operations Memory Test							
Memo	ory Test			View Men	nory		
Test Select	ion	Add	Iress Offset ×0	Vi	ew Memory		
Test Select		(0x8	0xC	*
PC<->DDR	3	0x0	<u></u>		3	4	E
		0x10	0x10 Memory Test Successfull 0x20 0x30 OK 0x40 OK OK		7	8	
Pattern Sel	lection	0x20			В	С	
Increment	•	0x30			F	10	
		0x40			13	14	
Transfer Siz	ze (Bytes)	0x50	15	10	17	18	
1MB	-	0x60	19	1A	1B	1C	
		0x70	1D	1E	1F	20	
		0x80	21	22	23	24	
		0x90	25	26	27	28	
	Start	0xA0	29	2A	2B	2C	
		0xB0	2D	2E	2F	30	
		0xC0	31	32	33	34	
		0xD0	35	36	37	38	
		0×E0	39	3A	3B	3C	-
		-					
1			Exit				

Note: Change the Offset Address and click View Memory to read the RAM memory content.

5. Click OK.

5.2.3.5 Core DMA—Operations

The following instructions describe running DMA operations between LSRAM and DDR4:

- 1. Select one of the following options from the DMA Transfer Type Selection drop-down list:
 - LSRAM -> DDR4—to transfer the data from LSRAM to PolarFire DDR4 memory
 - DDR4-> LSRAM—to transfer the data from PolarFire DDR4 memory to LSRAM
 - Both LSRAM <->DDR4—to transfer the data from LSRAM to and from PolarFire DDR4 memory
- 2. Select Transfer Size (4KB to 1MB) from the drop-down list.
- 3. Enter the Loop Count in the box.
- 4. Click **Start Transfer**. After a successful DMA operation, the GUI displays the throughput and average throughput in MBps. The following figure shows Core DMA—Operations.

Figure 64 • Core DMA—Operations

C Microsem	PolarFire PCIe Demo GUI	Connected
Device Info Demo Controls	Config Space PCIe Read/Write DMA Operations	
PCIe Continuous DMA PCIe SG	DMA Fabric Core DMA	
Operations		
DMA Transfer Type Selection DDR4 to LSRAM Transfer Size(Bytes) 4K Throughput(MBps) 0 Avg Thruput(MBps) 0	DDR4->LSRAM V DDR4->LSRAM LSRAM->DDR4 Both LSRAM<->DDR4 DDR3->LSRAM LSRAM->DDR3 Both LSRAM<->DDR3 DDR3->DDR3 DDR3->DDR4 DDR4->DDR4 DDR4->DR4 150- 150- 150- 150- 100-	+ 2 (H)
Loop Count 1	50 - 0 - 1 2 3 4 5 6 No of DMA Trai DDR4 to LSRAM Avg CLSRAM DDR4 to LSRAM Avg CLSRAM	7 8 9 10 nsfers SRAM to DDR4 Avg
	Exit	

- **Note:** The AXI LSRAM in the design is configured for 4KB. This 4KB is over written if more than 4KB of DMA operation is performed on LSRAM. This option is provided to exercise the throughputs with larger DMA size.
 - 5. Click **Exit** to quit the demo.

5.3 Running the Demo Through UART

The following steps describes how to run a demo using UART if the host PC PCIe slot is not available:

Check the **Device Manager** of the host PC for UART ports. The following figure shows the example UART ports in the **Device Manager** window.

Figure 65 • Device Manager—UART Ports

Ports (COM & LPT)
 Ports (COM & LPT)
 PiashPro5 Port (COM10)
 FlashPro5 Port (COM11)
 PiashPro5 Port (COM8)
 FlashPro5 Port (COM9)

The following steps describe how to run the reference design using UART IF:

1. Go to All Programs > PolarFire_PCle_GUI > PolarFire_PCle_GUI. The PolarFire PCle Demo window is displayed as shown in the following figure.

Figure 66 • PCIe EndPoint Demo Application

s M	icrosemi	PolarFire	e PCIe D	emc	GUI	PCIeUART	Connect
Device Info	Demo Controls	Config Space	PCIe Read/\	Nrite	DMA Operatio	ns UART	
					Number of P		
C	Device Vendor ID				Number of b		
	Device Type			BAR0 A	ddress x 0	BAR0 Size	(Bytes) × 0
	Driver Version			BAR1 A	ddress x 0	BAR1 Size	(Bytes) x 0
Dr	iver Time Stamp						
	Demo Type			BAR2 A	ddress x0	BAR2 Size	(Bytes) x 0
S	upported Width			BAR3 A	ddress x 0	BAR3 Size	(Bytes) x 0
N	egotiated Width			BAR4 A	ddress ×0	BAR4 Size	(Bytes) ×0
S	Supported Speed						
N	egotiated Speed			BAR5 A	ddress ×0	BAR5 Size	(Bytes) ×0
			Exit				

2. Select UART radio button and click Connect.

The GUI application scans for UART port and after successful connection, displays the DMA Operations UART tab as shown in Figure 67, page 52.

5.3.1 UART—DMA Operations

The following instructions describe the different ways to read data through LSRAM and DDR:

- 1. Select one of the following options from the **Continuous DMA Transfer Type Selection** drop-down list:
 - LSRAM -> DDR4—to transfer the data from LSRAM to PolarFire DDR4 memory
 - **DDR4-> LSRAM**—to transfer the data from PolarFire DDR4 memory to LSRAM
 - Both LSRAM <->DDR4—to transfer the data from LSRAM to and from PolarFire DDR4 memory
- 2. Select Transfer Size (4KB to 512KB) from the drop-down lists.
- 3. Enter the Loop Count in the box.
- 4. Click **Start Transfer**. After a successful DMA operation, the GUI displays the throughput and average throughput in MBps. The following figure shows DMA throughput and average throughput from the DDR memory to the LSRAM.

Figure 67 • UART—DMA Operations

🔍 Microsemi.	PolarFire PCIe	Demo GUI	PCIeUART	Connected
UART				
DMA Operations Memory Test				
Continous DMA Transfer Type Selection	DDR4->LSRAM	990 -		+ 2 %
DDR4 to LSRAM	↓ DDR4->LSRAM LSRAM->DDR4 Both DDR4<->LSRAM	980 - 970 -		
Transfer Size(Bytes) 4K	DDR3->LSRAM LSRAM->DDR3 Both DDR3<->LSRAM	960 - 950 - 950 -		
Throughput(MBps) 0	DDR3->DDR4 DDR4->DDR3 Both DDR3<->DDR4	930 -		
Avg Thruput(MBps) 0	Avg Thruput(MBps) 0	910 -		
Loop Count 1	Start Transfer	880 - <mark>-</mark> 880 - - 0	1 No of DMA Transfe	2 rs
		DDR4 to LSRA DDR4 to LSRAM Av	M 🖳 LSRAI rg 🔼 LSRAM to	M to DDR4 📩 DDR4 Avg 🔼
	Exit			

Note: The AXI LSRAM in the design is configured for 4KB. This 4KB is over written if more than 4KB of DMA operation is performed on LSRAM. This option is provided to exercise the throughputs with larger DMA size.

5.3.1.1 UART—Memory Test

The following instructions describe running Memory Test between PC and DDR4/LSRAM:

- 1. Select Transfer Size (4KB to 1MB) from the drop-down list.
- 2. Select **Pattern Selection** from the drop-down list—Increment, Decrement, Fill with Zeros, Fill with Ones, Fill with all A's, and Fill with all 5's. For successful Memory test operation, the **Patter Type for Mem Init** and **Patter Type for Mem Test** should be same.
- 3. Click Memory Test.
 - GUI sends command to fabric logic to initiate the LSRAM/DDR4 memory
 - · GUI sends command to fabric logic to read and compare LSRAM/DDR4 memory

The following figure shows UART—Memory Test tab.

Figure 68 • UART—Memory Test

Microsemi. PolarFire PCIe Demo GUI					
UART DMA Operations Memory Test					
Memory Test	Mer Memory Type	mory View	55		
Memory Size(Bytes) 4K	LSRAM DDR3 DDR4	× 0	0x8	View Memory	
Pattern Type for Mem Init Incremental	Ox000			E	
Incremental Offset Data x0	ОК				
Memory Type SRAM DDR3 DDR4	0x070 0x080 0x090 0x0A0				
Offset Address x0	0x0B0 0x0C0 0x0D0 0x0E0				
Memory Test PASS	0x0F0 0x100			.	
	Exit				

Note: Change the Offset Address and click View Memory to read the RAM memory content.

- 4. Click View Memory. It shows 1KB of RAM memory content.
- 5. Click OK.
- 6. Click Exit to quit the demo.

The following table lists the throughput values observed in Continuous DMA mode.

DMA Transfer Type	DMA Size	Throughput (MBps)	Average Throughput (MBps)
PC to LSRAM	64 K	1122	1086
LSRAM to PC		1019	1141
Both PC to and from LSRAM		1019/1128	1074/1125
PC to DDR4	64 K	1070	1049
DDR4 to PC		527 ¹	528
Both PC to and from DDR4		1075/527	1050/528

Table 5 • PolarFire Throughput Summary—Continuous DMA Mode

1. The PCIe DMA performs maximum of 32 beat AXI burst transactions (not AXI4's maximum of 256 beat), which causes low read performance of DDR4.

The following table lists the throughput values observed in SGDMA Mode.

Table 6 • PolarFire Throughput Summary—SGDMA Mode

DMA Transfer Type	DMA Size	Throughput (MBps)	Average Throughput (MBps)
PC to DDR4	1 MB	1031	1028
DDR4 to PC	-	530 ¹	530
Both PC to and from DDR4	_	1038/530	1020/530

1. The PCIe DMA performs maximum of 32 beat AXI burst transactions (not AXI4's maximum of 256 beat), which causes low read performance of DDR4.

The following table lists the throughput values observed in Core DMA mode.

Table 7 • PolarFire Throughput Summary—Core DMA Mode

DMA Transfer Type	DMA Size	Throughput (MBps)	Average Throughput (MBps)
LSRAM to DDR4	1 MB	1470	1470
DDR4 to LSRAM	-	1205	1205
Both LSRAM to and from DDR4	-	1470/1205	1470/1205

6 Appendix: References

This section lists documents that provide more information about the PCIe EndPoint and IP cores used in the reference design.

- For more information about PolarFire transceiver blocks, PF_TX_PLL, and PF_XCVR_REF_CLK, see UG0677: PolarFire FPGA Transceiver User Guide.
- For more information about PF_PCIE, see UG0685: PolarFire FPGA PCI Express User Guide.
- Fore more information about PF_CCC, see UG0684: PolarFire FPGA Clocking Resources User Guide.
- Fore more information about DDR4 memory, see UG0676: PolarFire FPGA DDR Memory Controller User Guide.
- For more information about Libero, ModelSim, and Synplify, see the *Microsemi Libero SoC PolarFire* web page.
- For more information about PolarFire FPGA Splash Kit, see UG0786: PolarFire FPGA Splash Kit User Guide.
- For more information about CoreAHBLite, see CoreAHBLite Handbook.
- For more information about CoreAHBtoAPB3, see CoreAHBtoAPB3 Handbook.
- For more information about CoreUART, see CoreUART Handbook.