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 document or to any products and services at any time without notice.

About Microsemi

Microsemi Corporation (Nasdaq: MSCC) 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. Microsemi is headquartered in Aliso Viejo, Calif., and has approximately 4,800 employees globally. Learn more at www.microsemi.com.
# Contents

1 Revision History ............................. 1
   1.1 Revision 8.0 .................................. 1
   1.2 Revision 7.0 .................................. 1
   1.3 Revision 6.0 .................................. 1
   1.4 Revision 5.0 .................................. 1
   1.5 Revision 4.0 .................................. 1
   1.6 Revision 3.0 .................................. 1
   1.7 Revision 2.0 .................................. 1
   1.8 Revision 1.0 .................................. 1

2 SmartFusion2 and IGLOO2 PCIe Data Plane Demo Using 2 Channel Fabric DMA . 2
   2.1 Demo Design ................................... 3
   2.1.1 Demo Design Features ............................. 4
   2.1.2 Demo Design Description ............................ 5
      2.1.2.1 Host PC Memory to LSRAM (Read) .................. 5
      2.1.2.2 LSRAM to Host PC Memory (Write) .................. 5
      2.1.2.3 Host PC Memory to DDR Memory (Read) .............. 5
      2.1.2.4 DDR Memory to Host PC Memory (Write) ............ 6
      2.1.2.5 LSRAM to DDR Memory (Write) ..................... 6
      2.1.2.6 DDR Memory to LSRAM (Read) ..................... 6
   2.2 Throughput Calculation .......................... 6
   2.3 Setting Up the Demo Design ....................... 7
      2.3.1 Jumper Settings for SmartFusion2 Advanced Development Kit .......... 7
      2.3.2 Jumper Settings for IGLOO2 Evaluation Kit ............. 7
      2.3.3 Programming the Device ............................ 7
      2.3.4 Connecting the Kit to Host PC PCIe Slot ............... 10
      2.3.5 Driver Installation ............................... 13
      2.3.6 PCIe_Demo Application ............................ 15
   2.4 Running the Design .............................. 17
      2.4.1 DDR DMA Operations .............................. 21
      2.4.2 LSRAM DMA Operations ............................ 23
      2.4.3 LSRAM and DDR DMA Operations ...................... 25
   2.5 Summary ........................................ 27

3 Appendix: IGLOO2 Evaluation Kit Board Setup for Laptop .................. 28

4 Appendix: Register Details ................................ 31
# Figures

| Figure 1 | Demo Design Files Top Level Structure | 3 |
| Figure 2 | PCIe Data Plane Demo Block Diagram | 4 |
| Figure 3 | FlashPro New Project | 8 |
| Figure 4 | FlashPro Project Configured | 9 |
| Figure 5 | FlashPro Programming Passed | 10 |
| Figure 6 | SmartFusion2 Advanced Development Kit Setup for Host PC | 11 |
| Figure 7 | IGLOO2 Evaluation Kit Setup | 12 |
| Figure 8 | Device Manager - PCIe Device Detection | 12 |
| Figure 9 | Update Driver Software | 13 |
| Figure 10 | Browse for Driver Software | 13 |
| Figure 11 | Browse for Driver Software Continued | 14 |
| Figure 12 | Windows Security | 14 |
| Figure 13 | Successful Driver Installation | 15 |
| Figure 14 | Installing PCIe_Demo Application | 15 |
| Figure 15 | PCIe_Demo Application Installation Steps | 16 |
| Figure 16 | Successful Installation of PCIe_Demo Application | 16 |
| Figure 17 | Device Manager - PCIe Device Detection | 17 |
| Figure 18 | PCIe Demo Application | 18 |
| Figure 19 | Device Info | 18 |
| Figure 20 | Demo Controls | 19 |
| Figure 21 | Demo Controls – Continued | 19 |
| Figure 22 | Configuration Space | 20 |
| Figure 23 | PCIe BAR1 Memory Access | 20 |
| Figure 24 | FDMA Transfer Type Selection – PC to DDR | 21 |
| Figure 25 | FDMA Transfer Type Selection – DDR to PC | 22 |
| Figure 26 | FDMA Transfer Type Selection – Both PC to and from DDR | 22 |
| Figure 27 | PC to LSRAM – FDMA Transfer Type Selection | 23 |
| Figure 28 | LSARM to PC – FDMA Transfer Type Selection | 24 |
| Figure 29 | Both PC to and from LSRAM – FDMA Transfer Type Selection | 24 |
| Figure 30 | FDMA Transfer Type Selection – DDR to LSRAM | 25 |
| Figure 31 | FDMA Transfer Type Selection – LSRAM to DDR | 26 |
| Figure 32 | FDMA Transfer Type Selection – Both DDR to LSRAM | 26 |
| Figure 33 | Lining up the IGLOO2 Evaluation Kit Board | 28 |
| Figure 34 | Inserting the IGLOO2 Evaluation Kit PCIe Connector | 29 |
| Figure 35 | IGLOO2 Evaluation Kit Connected to the Laptop | 30 |
## Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1</td>
<td>Design Requirements</td>
<td>2</td>
</tr>
<tr>
<td>Table 2</td>
<td>SmartFusion2 FPGA Advanced Kit Jumper Settings</td>
<td>7</td>
</tr>
<tr>
<td>Table 3</td>
<td>IGLOO2 FPGA Evaluation Kit Jumper Settings</td>
<td>7</td>
</tr>
<tr>
<td>Table 4</td>
<td>Throughput Summary</td>
<td>27</td>
</tr>
<tr>
<td>Table 5</td>
<td>Register Details</td>
<td>31</td>
</tr>
</tbody>
</table>
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 8.0
Updated the document for Libero SoC v11.8 software release.

1.2 Revision 7.0
Updated the document for Libero SoC v11.7 software updates (SAR 77470).

1.3 Revision 6.0
Updated the document for Libero SoC v11.6 software updates (SAR 72063).

1.4 Revision 5.0
Updated the designs files links.

1.5 Revision 4.0
Updated the document for Libero SoC v11.4 software updates (SAR 59587).

1.6 Revision 3.0
The following is a summary of the changes in revision 3.0 of this document.
- Updated document for Libero SoC v11.3 and for other enhancements (SAR 56080).
- Updated throughput values in Summary, page 27 (SAR 53490).
- Throughput for Read and Write Transfer type is reversed in Throughput Summary, page 27 (SAR 53822).
- MDDR throughput displayed in Mbps (SAR 53589).

1.7 Revision 2.0
Updated Table 4, page 27 (SAR 53490).

1.8 Revision 1.0
Revision 1.0 is the first publication of this document.
2 SmartFusion2 and IGLOO2 PCIe Data Plane Demo Using 2 Channel Fabric DMA

This demo highlights the high-speed data transfer, the capability of SmartFusion®2 and IGLOO®2 devices through the PCIe interface. To achieve the high-speed data transfers, an advanced extensible interface (AXI) based direct memory access (DMA) controller is implemented in the FPGA fabric. An application, **PCle_Demo** that runs in the host PC is provided for setting up and initiating DMA transactions from the SmartFusion2 or IGLOO2 PCIe endpoint to the host PC device. Drivers for connecting the host PC to the SmartFusion2 or IGLOO2 PCIe endpoint are provided as part of the demo deliverables.

Microsemi provides three different PCIe data plane demos for SmartFusion2 devices:

- **DG0501: SmartFusion2 PCIe MSS HPDMA Demo Guide**: This demo shows the low throughput data transfers between PCIe and double data rate (DDR).
- **DG0535: SmartFusion2 PCIe Data Plane Demo using MSS HPDMA and SMC_FIC Demo Guide**: This demo shows the medium throughput data transfers between PCIe and embedded static random access memory (eSRAM).
- **DG0517: SmartFusion2 and IGLOO2 PCIe Data Plane Demo using 2 Channel Fabric DMA Demo Guide (current demo)**: This demo shows the high throughput data transfers between PCIe and large SRAM (LSRAM).

The high-speed serial interface (SERDESIF) available in the SmartFusion2 or IGLOO2 devices provides a fully hardened PCIe endpoint implementation and is compliant to the PCIe Base Specification Revision 2.0, 1.1 and 1.0. For more information, refer to the **UG0447: SmartFusion2 and IGLOO2 FPGA High Speed Serial Interfaces User Guide**.

This demo demonstrates the performance of the PCIe and DDR controller of the SmartFusion2 and IGLOO2 device families. For a tutorial design on how to develop and use the PCIe endpoint including the tools flow and simulation, see the **TU0456: SmartFusion2 SoC FPGA PCIe Control Plane Demo Tutorial**.

### Table 1 • Design Requirements

<table>
<thead>
<tr>
<th>Design Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware Requirements</strong></td>
<td></td>
</tr>
<tr>
<td>SmartFusion2 Advanced Development Kit: Rev B or later</td>
<td></td>
</tr>
<tr>
<td>12 V adapter</td>
<td></td>
</tr>
<tr>
<td>USB A to Mini-B cable</td>
<td></td>
</tr>
<tr>
<td>IGLOO2 Evaluation Kit: Rev C or later</td>
<td></td>
</tr>
<tr>
<td>FlashPro4 programmer</td>
<td></td>
</tr>
<tr>
<td>12 V adapter</td>
<td></td>
</tr>
<tr>
<td>USB A to Mini-B cable</td>
<td></td>
</tr>
<tr>
<td>Host PC (or Laptop) with 8GB RAM and PCIe 2.0 Gen1 or Gen2 Any 64-bit Windows Operating System compliant slot.</td>
<td></td>
</tr>
<tr>
<td>PCI Edge Card Ribbon Cable</td>
<td></td>
</tr>
<tr>
<td><strong>Software Requirements</strong></td>
<td>v11.8</td>
</tr>
<tr>
<td>Libero® System-on-Chip (SoC)</td>
<td></td>
</tr>
<tr>
<td>FlashPro programming software</td>
<td></td>
</tr>
<tr>
<td>PCIe_Demo Application v1.1 or later</td>
<td></td>
</tr>
</tbody>
</table>
Note:
For SmartFusion2 Kit, PCIe with ×4 or higher is required. For IGLOO2 Kit, PCIe with ×1 or higher is required.
PCI Express card slot and PCIe Express card adapter (for Laptop only).
PCIe Express card adapter is not supplied with the IGLOO2 Evaluation Kit.

2.1 Demo Design

The demo design files are available for download from the following path in the Microsemi website:
http://soc.microsemi.com/download/rsc/?f=m2s_m2gl_dg0517_PCIE_Fabric_DMA_liberov11p8_df

The demo design files include:
• PCIe Drivers
• GUI
• Libero Project
• Programming files
• Readme file

The following figure illustrates the top-level structure of the design files. For further details, see the readme.txt file.

Figure 1 • Demo Design Files Top Level Structure

The PCIe core in the SmartFusion2 or IGLOO2 devices supports both the AXI and AMBA high-performance bus (AHB) master and slave interfaces. This demo design uses the AXI master and slave interfaces to achieve maximum bandwidth. The PCIe_Demo application on the host PC initiates the DMA transfers and the embedded PCIe core in the SmartFusion2 or IGLOO2 device initiates the AXI transactions through the AXI master interface to the DMA controller in the FPGA fabric. The DMA controller has two independent channels that share the AXI read/write channels of the PCIe AXI slave interface and the MDDR AXI slave interface. The DMA controller in the FPGA fabric initiates the DMA channels depending on the type of the DMA transfer. Each channel has a timer for calculating the throughput and has 4 KB of LSRAM buffer.

DMA channel 0 handles the following DMA transfers:
• Host PC memory to LSRAM
• Host PC memory to DDR memory
• LSRAM to DDR memory

DMA channel 1 handles the following DMA transfers:
• LSRAM to host PC memory
• DDR memory to host PC memory
• DDR memory to LSRAM
The following figure illustrates the demo design.

**Figure 2 • PCIe Data Plane Demo Block Diagram**

For SmartFusion2 Advanced Development Kit, MDDR is configured for accessing DDR3 memory in x32 mode. The MDDR clock is configured to 310 MHz (620 Mbps DDR) with a 155 MHz DDR_FIC clock for an aggregate memory bandwidth of nearly 2480 MBps. The PCIe AXI interface clock, ARM Cortex-M3 clock, PCIe AXI interface clock, and fabric DMA controller clocks are configured to 155 MHz.

For IGLOO2 Evaluation Kit, MDDR is configured for accessing LPDDR memory in x16 mode. The MDDR clock is configured to 155 MHz (310 Mbps DDR) with a 155 MHz DDR_FIC clock for an aggregate memory bandwidth of nearly 620 MBps. The PCIe AXI interface clock and fabric DMA controller clocks are configured to 155 MHz.

### 2.1.1 Demo Design Features

- DMA data transfers between the host PC memory and the LSRAM
- DMA data transfers between the host PC memory and the DDR memory
- DMA data transfers between the DDR memory and the LSRAM
- Throughput for every DMA data transfer
- Enables continuous DMA transfers for observing throughput variations
- Displays the PCIe link enable/disable, negotiated link width, and the link speed on the PCIe_Demo application
- Displays the position of DIP switches on the SmartFusion2 Advanced Development Kit or IGLOO2 Evaluation Kit on the PCIe_Demo application
- Displays the PCIe Configuration Space on the PCIe_Demo application
- Controls LEDs on the board according to the command from the PCIe_Demo application
- Enables read and write operations to scratchpad register in the FPGA fabric
- Interrupts the host PC, when the Push button is pressed. The PCIe_Demo application displays the count value of the number of interrupts sent from the board.
2.1.2 Demo Design Description

There are six different types of data transfers supported by this demo design. The following sections describe the process of each data transfer:

- Host PC Memory to LSRAM (Read), page 5
- LSRAM to Host PC Memory (Write), page 5
- Host PC Memory to DDR Memory (Read), page 5
- DDR Memory to Host PC Memory (Write), page 6
- LSRAM to DDR Memory (Write), page 6
- DDR Memory to LSRAM (Read), page 6

2.1.2.1 Host PC Memory to LSRAM (Read)

Data transfer from PC memory to the LSRAM occurs in the following sequence:

1. PCIe_Demo application sets up the fabric DMA controller through the PCIe link. This includes DMA direction, address, and size (4 KB).
2. Fabric DMA controller initiates a 16 beat AXI burst (that is, 128 bytes) read transaction to the PCIe AXI slave interface.
3. The PCIe core sends the memory read (MRd) transaction layer packets (TLP) to the host PC.
4. The host PC returns a completion (CplD) TLP to the PCIe link.
5. This returned data completes the AXI read initiated by the fabric DMA controller.
6. This data is stored in the LSRAM.
7. The fabric DMA controller repeats this process until the 4 KB of data transfer is completed.
8. The fabric DMA controller provides the DMA completion status and the number of clock cycles consumed to complete the DMA transaction to the PCIe_Demo application.

2.1.2.2 LSRAM to Host PC Memory (Write)

Data transfer from the LSRAM to PC memory occurs in the following sequence:

1. PCIe_Demo application sets up the fabric DMA controller through the PCIe link. This includes DMA direction, address, and size (4 KB).
2. Fabric DMA controller reads the LSRAM data and initiates an AXI 16 beat burst write transaction to PCIe AXI slave interface.
3. The PCIe core sends a memory write (MWr) TLP to the host PC.
4. The fabric DMA controller repeats this process until the 4 KB size of data transfer is completed.
5. The fabric DMA controller provides the DMA completion status and the number of clock cycles consumed to complete the DMA transaction to the PCIe_Demo application.

2.1.2.3 Host PC Memory to DDR Memory (Read)

Data transfer from the PC memory to the DDR memory occurs in the following sequence:

1. PCIe_Demo application sets up the fabric DMA controller through the PCIe link. This includes DMA direction, address and size (4 KB).
2. Fabric DMA controller initiates 16 beat AXI burst (that is, 128 bytes) read transaction to the PCIe AXI interface.
3. The PCIe core sends a memory read (MRd) transaction layer packets (TLP) to the host PC.
4. The host PC returns a completion data (CpiD) TLP to the PCIe link.
5. This returned data completes the AXI read initiated by the fabric DMA controller.
6. This data is stored in the dual port LSRAM.
7. The LSRAM data is written to the DDR controller through the AXI interface as an AXI 16 beat burst write transaction. The reads from the host PC memory and the writes to the DDR memory occur independent of each other for achieving high throughput. Empty flags are generated in the fabric DMA controller to avoid reading unknown data from the LSRAM.
8. The fabric DMA controller repeats this process until 4 KB of data transfer is completed.
9. The fabric DMA controller provides the DMA completion status and the number of clock cycles consumed to complete the DMA transaction to the PCIe_Demo application.
2.1.2.4 DDR Memory to Host PC Memory (Write)

Data transfer from the DDR memory to the PC memory occurs in the following sequence:

1. PCIe_Demo application sets up the fabric DMA controller through the PCIe link. This includes DMA direction, address and size (4 KB).
2. The fabric DMA controller initiates a 16 beat burst (that is, 128 bytes) AXI read transaction from the DDR through the MDDR controller.
3. The data is stored in the dual port LSRAM.
4. The LSRAM data is written to the PCIe core as an AXI 16 beat burst write transaction. The reads from the DDR memory and writes to the host PC memory occur independent of each other for achieving high throughput. Empty flags are generated in the fabric DMA controller to avoid reading unknown data from the LSRAM.
5. The PCIe core sends a memory write (MWr) TLP to the host PC.
6. The fabric DMA controller repeats this process until 4 KB of data transfer is completed.
7. The fabric DMA controller provides the DMA completion status and the number of clock cycles consumed to complete the DMA transaction to the PCIe_Demo application.

2.1.2.5 LSRAM to DDR Memory (Write)

Data transfer from the LSRAM to the DDR memory occurs in the following sequence:

1. PCIe_Demo application sets up the fabric DMA controller through the PCIe link. This includes DMA direction, address and size (4 KB).
2. The LSRAM data is written to the DDR controller through AXI interface as an AXI 16 beat burst write transaction.
3. The fabric DMA controller repeats this process until 4 KB of data transfer is completed.
4. The fabric DMA controller provides the DMA completion status and the number of clock cycles consumed to complete the DMA transaction to the PCIe_Demo application for display.

2.1.2.6 DDR Memory to LSRAM (Read)

Data transfer from the DDR memory to the LSRAM occurs in the following sequence:

1. PCIe_Demo application sets up the fabric DMA controller through the PCIe link. This includes DMA direction, address and size (4 KB).
2. The fabric DMA controller initiates a 16 beat burst AXI read transaction of the DDR through the MDDR controller.
3. The data is stored in the dual port LSRAM.
4. The fabric DMA controller repeats this process until 4 KB of data transfer is completed.
5. The fabric DMA controller provides the DMA completion status and the number of clock cycles consumed to complete the DMA transaction to the PCIe_Demo application for display.

2.2 Throughput Calculation

This demo implements a timer to measure the throughput of DMA transfers. The throughput measured includes all of the overhead of the AXI, PCIe, and DMA controller transactions. The procedure for measuring throughput is:

1. Setup the DMA controller for the complete transfer.
2. Start a timer and the DMA controller.
3. Initiate data transfer for the requested number of bytes.
4. Wait till DMA transfer is completed.
5. Record the number of clock cycles consumed for steps 2-4.

To arrive at a realistic system performance, the throughput calculation takes into account all the overheads during a transfer. The throughput formula is as shown below:

\[
\text{Throughput} = \frac{\text{Transfer Size (Bytes)}}{\text{(Number of clock cycles taken for a transfer} \times \text{Clock Period})}
\]

EQ 1
2.3 Setting Up the Demo Design

2.3.1 Jumper Settings for SmartFusion2 Advanced Development Kit

1. Connect the jumpers on the SmartFusion2 Advanced Development Kit, as shown in Table 2, page 7. **CAUTION:** While making the jumper connections, the power supply switch SW7 must be switched OFF.

### Table 2 • SmartFusion2 FPGA Advanced Kit Jumper Settings

<table>
<thead>
<tr>
<th>Jumper</th>
<th>Pin (from)</th>
<th>Pin (to)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>J116, J353, J354, J54</td>
<td>1</td>
<td>2</td>
<td>These are the default jumper settings of the Advanced Development Kit board. Make sure these jumpers are set accordingly.</td>
</tr>
<tr>
<td>J123</td>
<td>2</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>J124, J121, J32</td>
<td>1</td>
<td>2</td>
<td>JTAG programming via FTDI</td>
</tr>
</tbody>
</table>

2. Connect the host PC to the J33 Connector using the USB A to mini-B cable. The USB to UART bridge drivers are automatically detected. Connect the power supply to the J18 connector. Switch ON the power supply switch SW7.

2.3.2 Jumper Settings for IGLOO2 Evaluation Kit

1. Connect the jumpers on the IGLOO2 Evaluation Kit, as shown in the following table. **Note:** While making the jumper connections, the power supply switch SW7 must be switched OFF.

### Table 3 • IGLOO2 FPGA Evaluation Kit Jumper Settings

<table>
<thead>
<tr>
<th>Jumper</th>
<th>Pin (from)</th>
<th>Pin (to)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>J22</td>
<td>1</td>
<td>2</td>
<td>Default</td>
</tr>
<tr>
<td>J23</td>
<td>1</td>
<td>2</td>
<td>Default</td>
</tr>
<tr>
<td>J24</td>
<td>1</td>
<td>2</td>
<td>Default</td>
</tr>
<tr>
<td>J8</td>
<td>1</td>
<td>2</td>
<td>Default</td>
</tr>
<tr>
<td>J3</td>
<td>1</td>
<td>2</td>
<td>Default</td>
</tr>
</tbody>
</table>

2. Connect the FlashPro4 programmer to the J5 connector of the IGLOO2 Evaluation Kit.
3. Connect the power supply to the J6 connector. Switch ON the power supply switch SW7.

2.3.3 Programming the Device

Download the demo design from: [http://soc.microsemi.com/download/rsc/?f=m2s_m2gl_dg0517_pcie_fabric_dma_liberov11p8_df](http://soc.microsemi.com/download/rsc/?f=m2s_m2gl_dg0517_pcie_fabric_dma_liberov11p8_df)

1. Launch the FlashPro software.
2. Click **New Project**.
3. In the **New Project** window, type the project name.
Figure 3 • FlashPro New Project

4. Click **Browse** and navigate to the location where you want to save the project.
5. Select **Single device** as the Programming mode.
6. Click **OK** to save the project.
7. Click **Configure Device**.
8. Click **Browse** and navigate to the location where the PCIe_SB_top.stp file is located and select the file. The default location is:
   - SmartFusion2: `<download_folder>\M2S_M2GL_PCIE_FabricDMA_demo_DF\Programming-File\SF2`
   - IGLOO2: `<download_folder>\M2S_M2GL_PCIE_FabricDMA_demo_DF\ProgrammingFile\IGL2`
9. Click **Open**. The required programming file is selected and is ready to be programmed in the device.
10. Click **PROGRAM** to start programming the device. Wait until you get a message indicating that the program passed.
2.3.4 Connecting the Kit to Host PC PCIe Slot

1. After successful programming, switch OFF the SmartFusion2 Advanced Development Kit or the IGLOO2 Evaluation Kit and shut down the host PC.

2. This demo is designed to run in any PCIe Gen 2 compliant slot. If the host PC does not support Gen 2 compliant slot the demo switches to Gen 1 mode. Connect the CON1 - PCIe Edge connector of the SmartFusion2 Advanced Development Kit to the host PC's PCIe slot through the PCI Edge card ribbon cable.

   OR

   Connect the CON1 - PCIe Edge connector of the IGLOO2 Evaluation Kit to the PCIe slot of the host PC or connect the CON1-PCIe Edge connector to the laptop PCIe slot using the Express card adapter. If using a laptop, the Express card adapters support only Gen 1 and the demo works in Gen 1 mode.

   Note: Power OFF the host PC (or laptop) 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 not occur properly. The device detection and selection are dependent on the host PC (or laptop) PCIe configuration.

3. The following figure shows the board setup for the host PC in which SmartFusion2 Advanced Development Kit is connected to the host PC PCIe slot.
Figure 6 • SmartFusion2 Advanced Development Kit Setup for Host PC
The following figure shows the board setup for the host PC in which the IGLOO2 Evaluation Kit is connected to the host PC PCIe slot. To connect the IGLOO2 Evaluation Kit to the laptop using the Express card adapter, see Appendix: IGLOO2 Evaluation Kit Board Setup for Laptop, page 28.

**Figure 7 • IGLOO2 Evaluation Kit Setup**

4. Switch ON the power supply switch SW7.
5. Switch ON the host PC and check the **Device Manager of the Host PC** for PCIe Device. The following figure shows the example **Device Manager** window. If the device is not detected, power cycle the SmartFusion2 Advanced Development Kit or the IGLOO2 Evaluation Kit and click **scan for hardware changes** option in the **Device Manager**.

**Figure 8 • Device Manager - PCIe Device Detection**

**Note:** If the device is still not detected, check whether or not the BIOS version in the host PC is latest, and if PCI is enabled in the host PC BIOS.
2.3.5 **Driver Installation**

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

1. Right-click **PCI Device** in Device Manager and select **Update Driver Software**... See the following figure.

*Figure 9 • Update Driver Software*

2. In the **Update Driver Software - PCIe Device** window, select the **Browse my computer for driver software** option. See the following figure.

*Figure 10 • Browse for Driver Software*
3. Browse the drivers folder: M2S_M2GL_PCIE_FabricDMA_demo_DF_PCIE
drivers\Win_64bit_PCIE_Drivers and click Next. See the following figure.

   **Figure 11 • Browse for Driver Software Continued**

![Browse for driver software on your computer](image)

4. The Windows Security dialog box is displayed and click Install. See the following figure. After successful driver installation, a message window appears. See Figure 13, page 15.

   **Figure 12 • Windows Security**

![Windows Security dialog box](image)
2.3.6 PCIe_Demo Application

The PCIe_Demo application is a simple graphic user interface that runs on the host PC to communicate with the SmartFusion2 or IGLOO2 PCIe endpoint device. It provides PCIe link status, driver information and demo controls. The PCIe_Demo application invokes the PCIe driver installed on the host PC and provides commands to the driver according to the selection made. To install the PCIe_Demo application:

1. Extract the PCIe_Demo_GUI_Installer.rar and locate the files at M2S_M2GL_PCIe_FabricDMA_demo_DF\GUI.
2. Double-click the setup.exe in the provided GUI installation (PCIe_Demo_GUI_Installer\setup.exe).
3. Apply default options, as shown in the following figure.

4. To start the installation, click Next.
5. Click Finish to complete the installation.

6. Shut down the host PC.
7. Power cycle the SmartFusion2 Advanced Development Kit.
8. Restart the host PC.
2.4 Running the Design

The following steps describe how to run the demo design:

1. Expand the Microsemi PCIe device in the host PC Device Manager. See the following figure.

   Figure 17  Device Manager - PCIe Device Detection

   ![Device Manager - PCIe Device Detection](image)

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

2. Go to All Programs > Microsemi_PCIE_Demo > Microsemi_PCIE_GUI. The PCIe Demo GUI window is displayed. See the following figure.
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 Width, Negotiated Width, Supported Speed, Negotiated Speed, Number of Bars, and BAR Address. See the following figure.

**Figure 19 • Device Info**
4. Click the **Demo Controls** tab to display the **LED Controls**, **DIP Switch Status**, and **Interrupt Counters**. See the following figure.

*Figure 20 • Demo Controls*

5. Click **Start LED ON/OFF Walk**, **Enable DIP SW Session**, and **Enable Interrupt Session** to view controlling LEDs, getting the DIP switch status, and monitoring the interrupts simultaneously. See the following figure.

*Figure 21 • Demo Controls – Continued*
6. Click the **Config Space** tab to view the details about the PCIe configuration space. See the following figure.

*Figure 22 • Configuration Space*

7. Click the **PCIe Read/Write** tab to perform read and write to LSRAM using BAR1 space.
8. Click **Read** to read the 4 KB memory mapped to BAR1 space. See the following figure.

*Figure 23 • PCIe BAR1 Memory Access*

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

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

1. Select one of the following options from the **FDMA Transfer Type Selection** drop-down list:
   - **PC -> DDR**—to transfer the data from host PC to SmartFusion2/IGLOO2 DDR memory
   - **DDR-> PC**—to transfer the data from SmartFusion2/IGLOO2 DDR memory to host PC
   - **Both PC <-> DDR**—to transfer the data from host PC to and from SmartFusion2/IGLOO2 DDR memory

2. Enter the **Loop Count** in the box.

3. Click **Start Transfer**. After a successful DMA operation, the GUI displays the throughput in Mbps and average throughput in Mbps. See the following figures.

*Figure 24 • FDMA Transfer Type Selection – PC to DDR*
Figure 25 • FDMA Transfer Type Selection – DDR to PC

Figure 26 • FDMA Transfer Type Selection – Both PC to and from DDR
2.4.2 LSRAM DMA Operations

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

1. Select one of the following options from the FDMA Transfer Type Selection drop-down list:
   - **PC -> LSRAM**—to transfer the data from host PC to SmartFusion2/IGLOO2 LSRAM memory
   - **LSRAM -> PC**—to transfer the data from SmartFusion2/IGLOO2 LSRAM memory to host PC
   - **Both PC <--> LSRAM**—to transfer the data from host PC to and from SmartFusion2/IGLOO2 LSRAM memory
2. Enter the Loop Count in the box.
3. Click Start Transfer. After a successful DMA operation, the GUI displays the throughput in Mbps and average throughput in Mbps. See the following figures.

*Figure 27* • PC to LSRAM – FDMA Transfer Type Selection
**Figure 28**  •  LSARM to PC – FDMA Transfer Type Selection

![Schematic](Image)

**Figure 29**  •  Both PC to and from LSRAM – FDMA Transfer Type Selection

![Schematic](Image)
2.4.3 **LSRAM and DDR 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 **FDMA Transfer Type Selection** drop-down list:
   - **DDR -> LSRAM**—to transfer the data from DDR to SmartFusion2/IGLOO2 LSRAM memory.
   - **LSRAM-> DDR**—to transfer the data from SmartFusion2/IGLOO2 LSRAM memory to DDR.
   - **Both DDR <-> LSRAM**—to transfer the data from DDR to and from SmartFusion2/IGLOO2 LSRAM memory
2. Enter the **Loop Count** in the box.
3. Click **Start Transfer**. After a successful DMA operation, the GUI displays the throughput in Mbps and average throughput in Mbps. See the following figures.

*Figure 30 • FDMA Transfer Type Selection – DDR to LSRAM*
4. Click **Exit** to quit the demo.
2.5 **Summary**

This demo shows how to implement a PCIe data plane design using the AXI-based fabric DMA controller. Throughput for data transfers is dependent on the host PC system configuration and the type of PCIe slots used.

The following table lists the throughput values observed on the HP workstation Z220 PCIe slot 4.

*Table 4* • Throughput Summary

<table>
<thead>
<tr>
<th>DMA Transfer Type</th>
<th>Throughput in Mbytes/Sec</th>
<th>SmartFusion2 (X4 Lane)</th>
<th>IGLOO2 (X1 Lane)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Gen1</td>
<td>Gen2</td>
<td>Gen1</td>
</tr>
<tr>
<td>Single xfer (50)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Read</td>
<td>574</td>
<td>509</td>
<td>710</td>
</tr>
<tr>
<td>Write</td>
<td>819</td>
<td>820</td>
<td>1096</td>
</tr>
<tr>
<td>R/W</td>
<td>529/817</td>
<td>508/820</td>
<td>690/1096</td>
</tr>
<tr>
<td>Loop xfer (50)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Read</td>
<td>465</td>
<td>456</td>
<td>422</td>
</tr>
<tr>
<td>Write</td>
<td>577</td>
<td>566</td>
<td>577</td>
</tr>
<tr>
<td>R/W</td>
<td>334/570</td>
<td>323/488</td>
<td>351/510</td>
</tr>
<tr>
<td>DDR to LSRAM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Read</td>
<td>606</td>
<td>599</td>
<td>606</td>
</tr>
<tr>
<td>Write</td>
<td>606</td>
<td>590</td>
<td>606</td>
</tr>
<tr>
<td>R/W</td>
<td>606/606</td>
<td>600/592</td>
<td>606/606</td>
</tr>
<tr>
<td>R/W</td>
<td>606/606</td>
<td>600/592</td>
<td>606/606</td>
</tr>
</tbody>
</table>

Throughput values are observed as Mbytes/sec for single and loop transfers.
3 Appendix: IGLOO2 Evaluation Kit Board Setup for Laptop

The following figure shows how to line up the IGLOO2 Evaluation Kit PCIe connector with the adapter card slot.

Figure 33 • Lining up the IGLOO2 Evaluation Kit Board

Note: The notch (highlighted in red) does not go into the adapter card.
The following figure shows the IGLOO2 Evaluation Kit PCIe connector inserted into the PCIe adapter card slot.

*Figure 34 • Inserting the IGLOO2 Evaluation Kit PCIe Connector*
The following figure shows the PCIe adapter card and the IGLOO2 Evaluation Kit connected to the laptop.

*Figure 35* • IGLOO2 Evaluation Kit Connected to the Laptop
The following table lists the registers used to interface with the fabric DMA Controller. These registers are in BAR1 address space.

**Table 5 • Register Details**

<table>
<thead>
<tr>
<th>Register Name</th>
<th>Register Address</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PC_BASE_ADDR</td>
<td>$0 \times 8028$</td>
<td>Host PC memory base address provided by the driver.</td>
</tr>
<tr>
<td>DMA_DIR</td>
<td>$0 \times 8008$</td>
<td>DMA direction:</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Direction</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1. PCIe $\rightarrow$ DDR memory $0 \times 11AA0001$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. DDR $\rightarrow$ PCIe memory $0 \times 11AA0002$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. LSRAM $\rightarrow$ DDR memory $0 \times 11AA0003$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. DDR $\rightarrow$ LSRAM memory $0 \times 11AA0004$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5. PCIe $\rightarrow$ LSRAM memory $0 \times 11AA0005$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6. LSRAM $\rightarrow$ PCIe memory $0 \times 11AA0006$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To reset the DMA, the register value is $0 \times 11AA0007$.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Before initiating DMA transactions, reset the DMA with the register value, $0\times11AA0007$.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The DMA transactions 1 and 2, 3 and 4, or 5 and 6 can be performed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>simultaneously by writing the corresponding values one after other.</td>
</tr>
<tr>
<td>DMA_CH0_STATUS</td>
<td>$0 \times 8100$</td>
<td>DMA Channel-0 status</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DMA_CH0_STATUS[31] $=DMA$ operation completed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0 $=DMA$ operation not completed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DMA_CH0_STATUS[15:0] $=CLK$ count</td>
</tr>
<tr>
<td>DMA_CH1_STATUS</td>
<td>$0 \times 8108$</td>
<td>DMA Channel-1 status</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DMA_CH1_STATUS[31] $=DMA$ operation completed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0 $=DMA$ operation not completed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DMA_CH1_STATUS[15:0] $=CLK$ count</td>
</tr>
<tr>
<td>RW_REG</td>
<td>$0 \times 0$</td>
<td>Scratchpad register for PCIe R/W.</td>
</tr>
<tr>
<td>LED_CTRL</td>
<td>$0 \times A0$</td>
<td>LEDs control register.</td>
</tr>
<tr>
<td>SWITCH_STATUS</td>
<td>$0 \times 90$</td>
<td>DIP switch status.</td>
</tr>
<tr>
<td>MEMORY</td>
<td>$0 \times 9000$</td>
<td>4 k memory to access from BAR1</td>
</tr>
</tbody>
</table>

**Note:** For the DDR memory, the source memory address is fixed as 0x0100_0000 and the destination memory address is fixed as 0 × 0000_0000.