# SmartFusion2 PCIe MSS HPDMA - Libero SoC v11.7

## DG0501 Demo Guide

Microsemi
Power Matters.™

# Contents


1 Preface . . . 5
1.1 Purpose . . . 5
1.2 Intended Audience . . . 5
1.3 References . . . 5
2 PCIe Data Plane Demo Using MSS HPDMA . . . 6
2.1 Introduction . . . 6
2.2 Design Requirements . . . 7
2.3 Demo Design . . . 7
2.3.1 Introduction . . . 7
2.3.2 Demo Design Features . . . 9
2.3.3 Demo Design Description . . . 9
2.3.3.1 PC Memory to LPDDR . . . 9
2.3.3.2 LPDDR to PC Memory . . . 10
2.3.3.3 LPDDR to Fabric SRAM Block . . . 10
2.3.3.4 Fabric SRAM to LPDDR . . . 10
2.4 Setting Up the Demo Design . . . 11
2.5 Running the Demo Design . . . 12
2.5.1 Setting Up the Device . . . 13
2.5.2 Driver Installation . . . 15
2.5.3 Demo GUI . . . 18
2.5.4 Steps to Run the Design . . . 20
2.6 Summary . . . 27
3 Revision History . . . 28
4 Product Support . . . 29
4.1 Customer Service . . . 29
4.2 Customer Technical Support Center . . . 29
4.3 Technical Support . . . 29
4.4 Website . . . 29
4.5 Contacting the Customer Technical Support Center . . . 29
4.5.1 Email . . . 29
4.5.2 My Cases . . . 29
4.5.3 Outside the U.S. . . . 30
4.6 ITAR Technical Support . . . 30

# Figures


Figure 1. Demo Design Files Top-Level Structure . . . 8
Figure 2. PCIe Data Plane Demo Block Diagram . . . 8
Figure 3. FlashPro New Project . . . 12
Figure 4. FlashPro Project Configured . . . 13
Figure 5. SmartFusion2 Security Evaluation Kit Setup . . . 14
Figure 6. FlashPro Programming Passed . . . 14
Figure 7. Device Manager - PCIe Device Detection . . . 15
Figure 8. Open Command Prompt . . . 16
Figure 9. Install Jungo . . . 17
Figure 10. Jungo Driver Installation . . . 17
Figure 11. Completed Successfully . . . 18
Figure 12. GUI Installation . . . 19
Figure 13. Successful Installation of GUI . . . 19
Figure 14. Device Manager - PCIe Device Detection . . . 20
Figure 15. PCIe Demo GUI . . . 20
Figure 16. Version Information . . . 21
Figure 17. DMA Operations . . . 22
Figure 18. Transfer Size Selection . . . 23
Figure 19. Transfer Size Selection . . . 23
Figure 20. PC Memory to LPDDR Transfer Throughput . . . 24
Figure 21. LPDDR to PC Memory Transfer Throughput . . . 25
Figure 22. Fabric SRAM to LPDDR Transfer Throughput . . . 26

# Tables


Table 1. Design Requirements . . . . . 7
Table 2. SmartFusion2 Security Evaluation Kit Jumper Settings . . . . . 11
Table 3. Throughput Summary . . . . . 27

# 1 Preface

## 1.1 Purpose

This demo is for SmartFusion®2 system-on-chip (SoC) field programmable gate array (FPGA) devices. It provides instructions on how to use the corresponding reference design.

## 1.2 Intended Audience

This demo guide is intended for

- FPGA designers
- Embedded designers
- System-level designers

## 1.3 References

Refer to the following web page for a complete and up-to-date listing of SmartFusion2 device documentation: *http://www.microsemi.com/products/fpga-soc/soc-fpga/smartfusion2*

- *SmartFusion2 Microcontroller Subsystem User Guide*
- *SmartFusion2 SoC FPGA High Speed Serial Interfaces User Guide*
- *SmartFusion2 PCIe Control Plane Demo User Guide*

# 2 PCIe Data Plane Demo Using MSS HPDMA

## 2.1 Introduction

This demo describes the usage of the following embedded features of the SmartFusion2 devices:

- PCIe controller
- High-performance DMA controller
- DDR controller

The demo uses all of these embedded features and limited FPGA resources. This demo architecture is resource and power efficient, but does not demonstrate the highest performance of the PCIe link.

This demo explains the SmartFusion2 embedded PCI Express feature and how can it be used as a data plane interface using the SmartFusion2 Security Evaluation Kit. The demo design is a simple design that enables you to access the PCIe end point (EP) available in the SmartFusion2 device from a PC to initiate the DMA transactions using the PCIe interface. In this demo, SmartFusion2 High-performance DMA (HPDMA) controller has been used to perform DMA transfers. A host PC **PCIe_Demo** application is provided for setting up and initiating DMA transactions from SmartFusion2 PCIe end point to host PC. The demo also provides host PC device drivers for the SmartFusion2 PCIe Endpoint. This demo is built on top of the PCIe Control Plane demo. Refer to the *UG0456: SmartFusion2 PCIe Control Plane Demo User Guide* for more information on PCIe Control Plane demo.

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

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

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

## 2.2 Design Requirements

Table 1 lists the reference design requirements and details for running the PCIe data plane demo using microcontroller subsystem (MSS) HPDMA.

***Table 1 •*** **Design Requirements**

| Design Requirements | Description |
|---|---|
| **Hardware Requirements** | |
| SmartFusion2 Security Evaluation Kit:<br>• FlashPro4 programmer<br>• 12 V adapter<br>• PCI edge card ribbon cable | Rev D or later |
| Host PC or Laptop (8 GB RAM) | Any 64-bit Windows Operating System |
| Host PC with an available PCIe 2.0 compliant slot (x1 or greater) | – |
| **Software Requirements** | |
| Libero® System-on-Chip (SoC) | v11.7 |
| SoftConsole | v3.4 SP1* |
| FlashPro programming software | v11.7 |
| PCIe Demo GUI Installer | v11.3 or later |
| ***Note:*** **For this application, SoftConsole v4.0 is used, see the TU0546: SoftConsole v4.0 and Libero SoC v11.7 Tutorial.* | |

## 2.3 Demo Design

### 2.3.1 Introduction

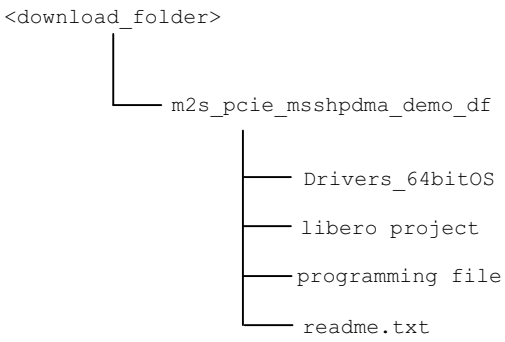
The demo design files are available for download from the following path in the Microsemi website: *http://soc.microsemi.com/download/rsc/?f=m2s_dg0501_pcie_mss_hpdma_liberov11p7_df*

The demo design files include:

- Drivers_64bitOS
- Libero Project
- Programming files
- Readme.txt file

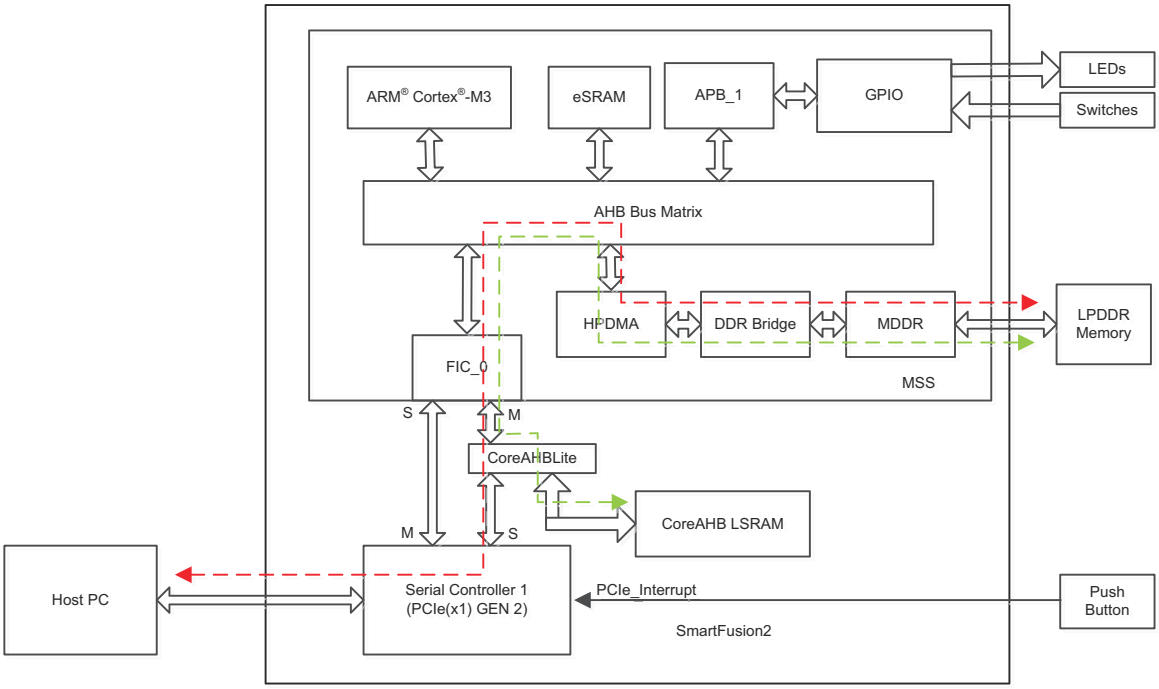
Figure 1 shows the top-level structure of the design files. For further details, refer to the `readme.txt` file.

***Figure 1 •*** **Demo Design Files Top-Level Structure**




In Figure 2, the red colored line shows the data flow between the host PC memory and the LPDDR memory and the green colored line shows the data flow between the fabric LSRAM and the LPDDR memory.

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



In this demo design, different blocks are configured as following:

- The SERDES_IF2_0 in the SmartFusion2 device is configured for PCIe 2.0, x1 lanes and Gen2 rate.
- The MDDR controller is configured to use the LPDDR memory on the SmartFusion2 Security Evaluation Kit.
- The CoreAHBLSRAM IP is configured to use the 128 KB of LSRAM.
- The AHB master interface of SERDES_IF2_0 is enabled and connected to the AHB slave interface of FIC_0 to access the MSS peripherals. The AHB slave interface of SERDES_IF2_0 and CoreAHBLSRAM IP are connected to the AHB master interface of FIC_0 to access the PCIe interface and the fabric LSRAM from the MSS.
- BAR0 and BAR1 are configured in 32-bit memory mapped memory mode. The AXI master window 0 is enabled and configured to map the BAR0 memory address space to MSS GPIO address space to control the MSS GPIOs. The AXI master window 1 is enabled and configured to map the BAR1 memory address space to eSRAM address space to perform read and write operations from the PCIe interface. The AXI slave window 0 is enabled and configured to map the SmartFusion2 local address space to the host PC address space.
- MSS GPIO block is enabled and configured as:
  - GPIO_0 to GPIO_7 as outputs and connected to LEDs
  - GPIO_8 to GPIO_11 as inputs and connected to DIP switches
- The MDDR controller clock is configured to 160 MHz and the M3_CLK clock and Serial Controller 1 clock base are configured to 80 MHz.

## 2.3.2 Demo Design Features

Following are the additional features that the PCIe data plane demo design has over the PCIe control plane demo:

- DMA data transfers between the host PC and the LPDDR memory using the HPDMA controller
- DMA data transfers between the LPDDR memory and the SmartFusion2 LSRAM using the HPDMA controller
- Throughput monitoring for every DMA data transfer
- Loop DMA data transfers to observe the throughput variations
- Control LEDs from both the host PC and the Cortex-M3 processor

Following are the features of PCIe control plane demo:

- Displays the PCIe link enable/disable, negotiated link width, and the link speed
- Displays the position of DIP Switches on SmartFusion2 Security Evaluation Kit
- Displays the SmartFusion2 PCIe Configuration Space
- Controls the status of LEDs on the SmartFusion2 Security Evaluation Kit according to the command from the GUI
- Enables read and writes to eSRAM
- Interrupts the Host PC, when the **Push** button is pressed. The GUI displays the count value of the number of interrupts sent from the SmartFusion2 Security Evaluation Kit

## 2.3.3 Demo Design Description

The PCIe link is configured at GEN2 rate and four lanes and is capable of throughputs greater than 320 Mbps. In this demo, the MSS using an AHB 32-bit data bus and clock rate of 80 MHz reduces the throughput of a PCIe link to the throughput of the MSS.

There are four different types of data transfers that this demo design supports. The following sections describe the process of each data transfer. Notes are provided to assist you with methods to increase system performance.

### 2.3.3.1 PC Memory to LPDDR

A data transfer from PC memory to the LPDDR device happens in the following sequence of steps:

1. HPDMA is setup over the PCIe link based on the settings in the GUI.
2. HPDMA initiates an AHB read transaction through the MSS fabric interface (FIC) and to the PCIe AHB interface.
3. The PCIe core sends a memory read (MRd) transaction layer packets (TLP) to the host PC.

4. The host PC returns with a completion with data (CplD) TLP to the PCIe link.
5. This return data completes the AHB read initiated by HPDMA controller.
6. This data is then written to the LPDDR controller.
7. HPDMA repeats this process until the transfer size set in the host PC GUI is completed. For transfers that require more than a single pass through the four buffer descriptors of HPDMA, the Cortex-M3 is used to reset the buffer descriptors and re-start the transfer process.

This read operation is effective, but not optimal for a PCIe link since the AHB bus of the MSS does not support a split-transaction.

**Note:** For optimal data transfers from PC memory to LPDDR, you are advised to use the AXI interface of the PCIe link and a fabric-based DMA controller. The AXI supports split transactions and allows the PCIe link to send a larger read request. While waiting for the read request, the AXI bus can move on to the next transaction. Multiple read requests can be initiated that allow the PCIe link to operate efficiently with read completion data returning.

### 2.3.3.2 LPDDR to PC Memory

A data transfer from the LPDDR device to PC memory happens in the following sequence of steps:

1. HPDMA is setup over the PCIe link based on the settings in the GUI.
2. HPDMA initiates an AHB read transaction of the LPDDR through the DDR controller of the MSS.
3. The data is written to the PCIe core as an AHB write transaction through the FIC.
4. The PCIe core sends a memory write (MWr) TLP to the host PC.
5. HPDMA repeats this process until the transfer size set in the host PC GUI is completed. For transfers that require more than a single pass through the four buffer descriptors of HPDMA, the Cortex-M3 is used to reset the buffer descriptors and re-start the transfer process.

This write operation is effective, but not optimal for a PCIe link since the AHB bus of the MSS does not support a burst capability to send larger MWr TLPs across the PCIe link which is more efficient.

**Note:** For optimal data transfers from LPDDR to PC memory, you are advised to use the AXI interface of the PCIe link and a fabric-based DMA controller. The AXI interface supports burst transfers to send larger MWr TLPs. The maximum TLP size supported by the PCIe core is 1 KB. This reduces the overhead of a single TLP and increases the overall efficiency of the link.

### 2.3.3.3 LPDDR to Fabric SRAM Block

The FPGA design of this demo contains a 128 KB fabric SRAM block. Data transfer from the LPDDR device to fabric SRAM happens in the following sequence of steps:

1. HPDMA is setup over the PCIe link based on the settings in the GUI.
2. HPDMA initiates an AHB read transaction of the LPDDR through the DDR controller of the MSS.
3. The data is written to the fabric SRAM as an AHB write transaction through the FIC.
4. HPDMA repeats this process until the transfer size set in the host PC GUI is completed. For transfers that require more than a single pass through the four buffer descriptors of HPDMA, the Cortex-M3 is used to reset the buffer descriptors and re-start the transfer process.

### 2.3.3.4 Fabric SRAM to LPDDR

A data transfer from the fabric SRAM to the LPDDR device happens in the following sequence of steps:

1. HPDMA is setup over the PCIe link based on the settings in the GUI.
2. HPDMA initiates an AHB read transaction through the MSS fabric interface (FIC) and to the fabric SRAM.
3. This data is written to the LPDDR controller.
4. HPDMA repeats this process until the transfer size set in the host PC GUI is completed. For transfers that require more than a single pass through the four buffer descriptors of HPDMA, the Cortex-M3 is used to reset the buffer descriptors and re-start the transfer process.

The Cortex-M3 processor initializes the HPDMA controller after receiving instructions from the PCIe demo application through the PCIe interface to perform DMA transactions between DDR, LSRAM, and the host PC. MSS timer_0 is started after initializing HPDMA and stopped on HPDMA interrupt. The timer gives the number of cycles taken to complete the data transfer. The GUI reads this value and calculates the throughput that is displayed in the application. HPDMA also interrupts the host PC application through the PCIe interface after completion of data transfer. The driver on the host PC creates a buffer and passes the physical address of that buffer to the SmartFusion2 device. HPDMA uses this address to perform the DMA transfers to host PC. The following options are available in the application for the DMA data transactions:

- Data transaction direction:
  - LPDDR to host PC
    In this DMA transfer, HPDMA reads the data from external LPDDR memory through MDDR and writes to host PC buffer through FIC_0 and PCIe.
  - Host PC to LPDDR
    In this DMA transfer, HPDMA reads the data from host PC buffer through FIC_0 and PCIe and writes to external LPDDR memory through MDDR.
  - Fabric LSRAM to LPDDR
    In this DMA transfer, HPDMA reads the data from Fabric LSRAM through FIC_0 and writes to external LPDDR memory through MDDR.
  - LPDDR to Fabric LSRAM
    In this DMA transfer, HPDMA reads the data from external LPDDR memory through MDDR and writes to Fabric LSRAM through FIC_0. The Cortex-M3 processor initializes HPDMA according to the direction of transfer.
- DMA size:
  For DMA to host PC, transfers can be selected from 64 KB to 1 MB. For DMA to LSRAM, transfers can be selected from 16 KB to 128 KB. HPDMA has four buffer descriptors and a single channel. Each buffer descriptor can perform 64 KB transfers at a time. These buffer descriptors can be initialized at a time so that the HPDMA queues the DMA operations, that is, 256 KB of data transfers can be done with a single initialization of HPDMA.
  For doing 1 MB data transfers, HPDMA is reinitialized for every 256 KB.
- Burst size of this demo is fixed to 4 bytes. The AHB interface of the MSS does not support burst therefore the PCIe TLPs only contain the 4 bytes as part of a single 32-bit AHB transaction.
- Continuous transfers:
  The DMA transfers can be done in a loop to observe the throughput changes.
- Blink LEDs from Cortex-M3 and from host PC. This option can be selected in parallel to the DMA transfers.

## 2.4 Setting Up the Demo Design

1. Connect the FlashPro4 programmer to the J5 connector of the SmartFusion2 Security Evaluation Kit board.
2. Connect the jumpers on the SmartFusion2 Security Evaluation Kit as shown in Table 2.
   **CAUTION**: While making the jumper connections, the power supply switch SW7 on the board must be in OFF position.

***Table 2 •*** **SmartFusion2 Security Evaluation Kit Jumper Settings**

| Jumper | Pin (From) | Pin (To) | Comments |
|---|---|---|---|
| J22, J23, J24, J8, J3 | 1 | 2 | These are the default jumper settings of the Security Evaluation Kit board. Make sure these jumpers are set accordingly. |

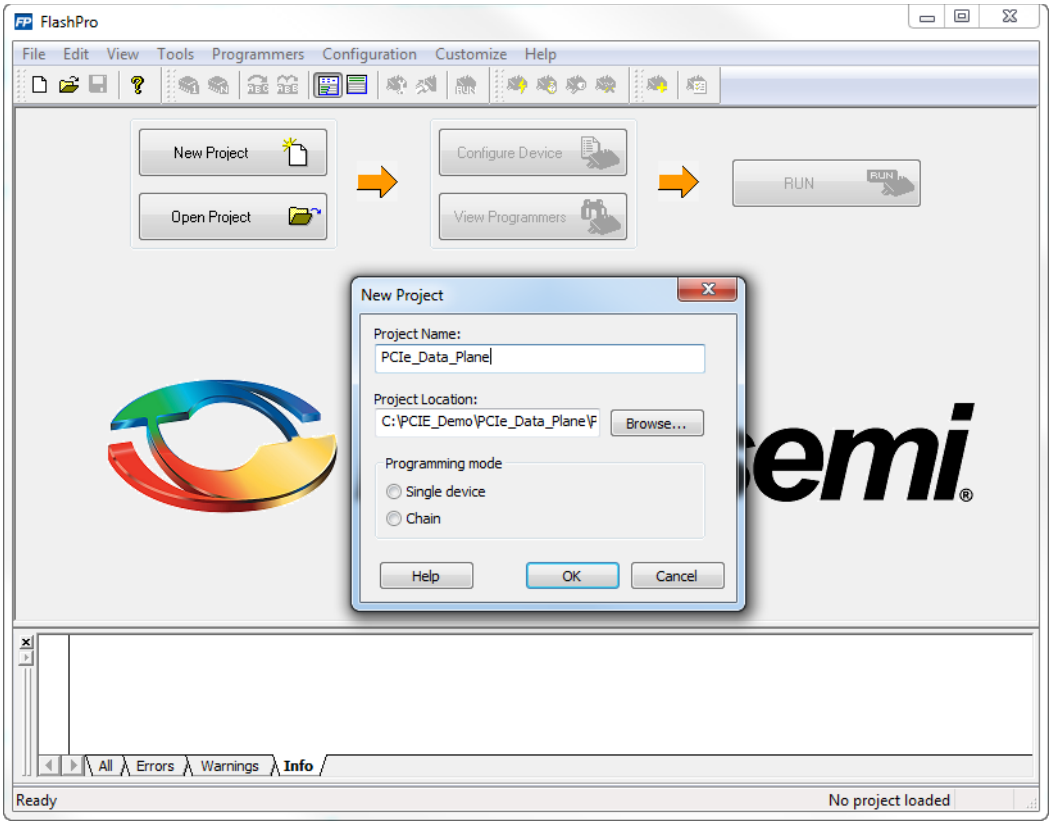
3. Connect the power supply to the J6 connector on the SmartFusion2 Security Evaluation Kit board.

## 2.5 Running the Demo Design

To run the demo design follow these steps:

1. Download the demo design from: *http://soc.microsemi.com/download/rsc/?f=m2s_dg0501_pcie_mss_hpdma_liberov11p7_df*
2. Switch on the power supply switch SW7 on the SmartFusion2 Security Evaluation Kit.
3. Run the FlashPro v11.7 software.
4. Click **New Project**.
5. In the **New Project** window, type the project name as PCIe_Data_Plane.

***Figure 3 •*** **FlashPro New Project**



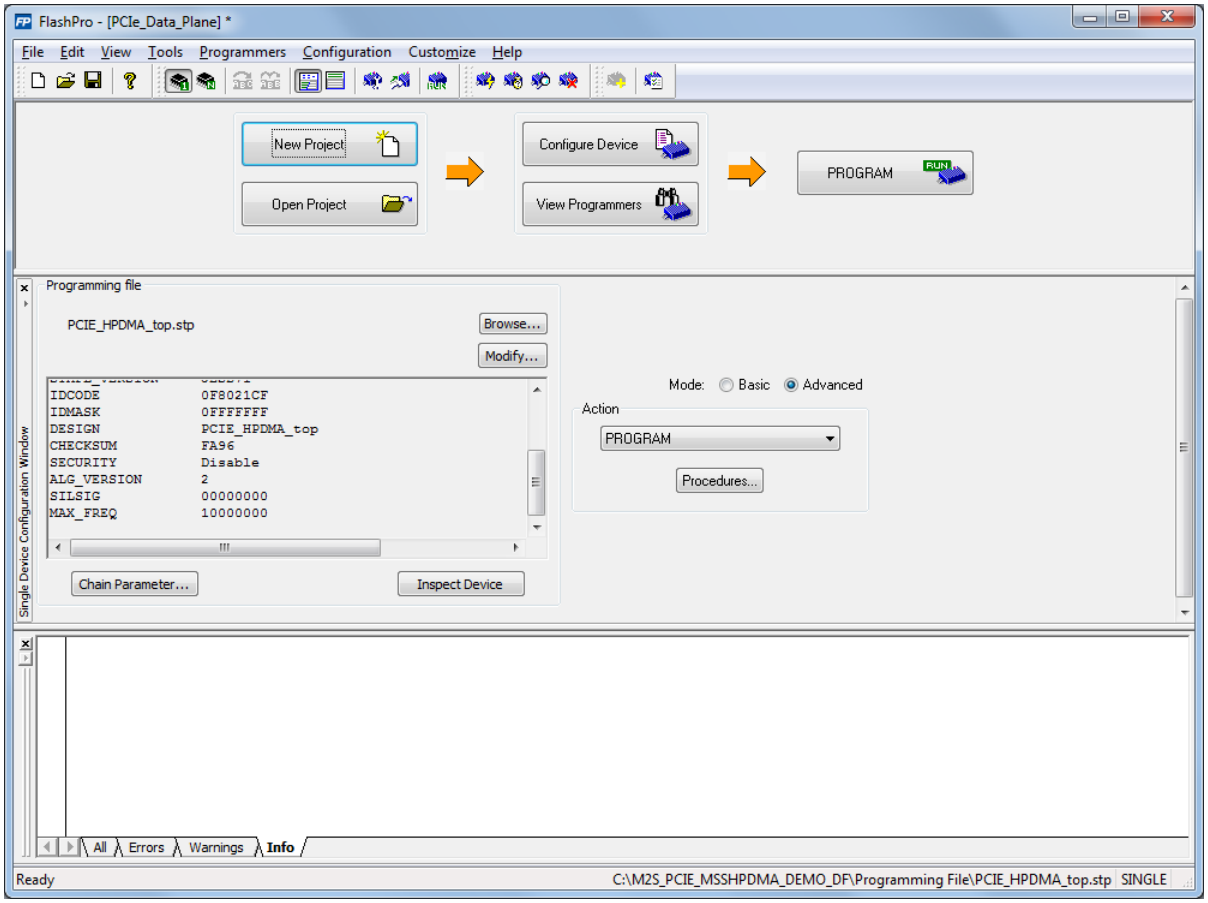

6. Click **Browse** and navigate to the location where you want to save the project.
7. Select **Single device** as the programming mode.
8. Click **OK** to save the project.

## 2.5.1 Setting Up the Device

Use the following steps to configure the device:

1. Click **Configure Device**.
2. Click **Browse** and navigate to the location where the `PCIE_HPDMA_top.stp` file is located and select the file. The default location is: *<download_folder>\M2S_PCIE_MSSHPDMA_DEMO_DF\Programming File*
3. Click **Open**. The required programming file is selected and is ready to be programmed in the device.

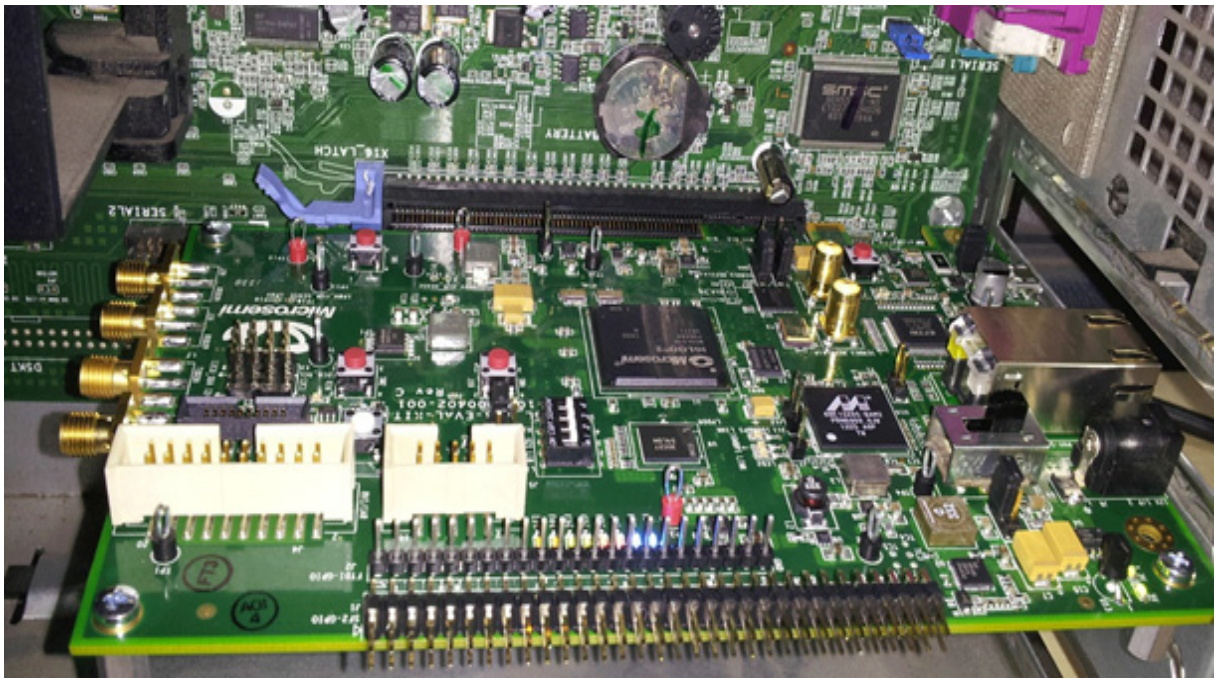
***Figure 4 •*** **FlashPro Project Configured**




4. Click **PROGRAM** to start programming the device. Wait until you get a message indicating that the **PROGRAM PASSED**.

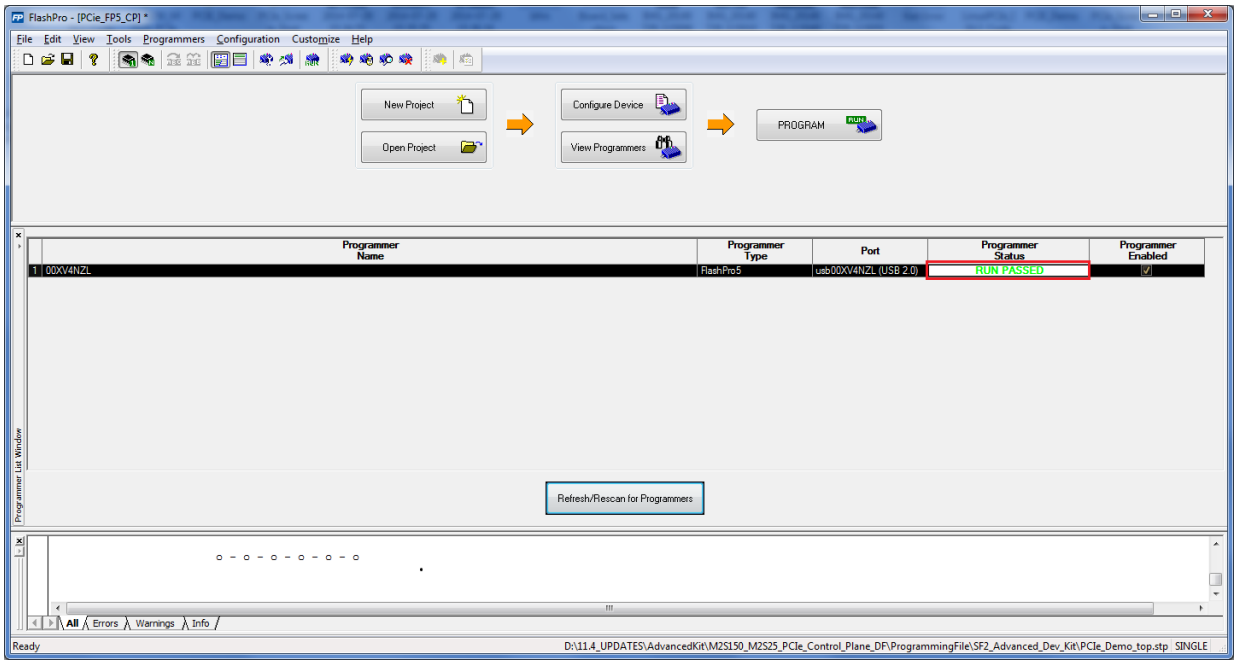
The board setup is shown in Figure 5.

***Figure 5 •*** **SmartFusion2 Security Evaluation Kit Setup**



5. After successful programming, power **OFF** the SmartFusion2 Security Evaluation Kit board and shut down the Host PC and connect the SmartFusion2 Security Evaluation Kit to the host PC.

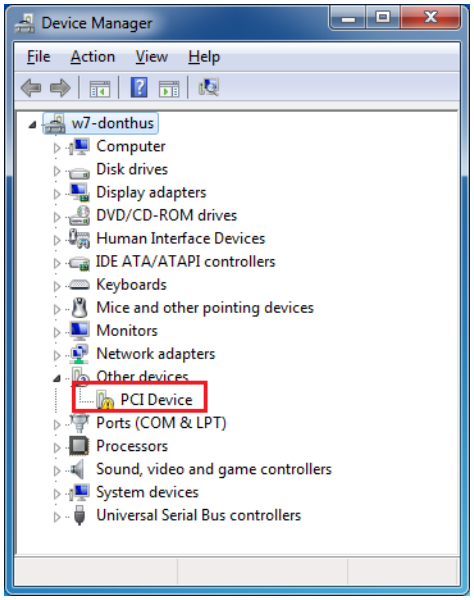
***Figure 6 •*** **FlashPro Programming Passed**




6. Switch the power supply switch **SW7** to **ON** position.
7. Power on the host PC and check the host PC's **Device Manager** for **PCIe Device**. It will be similar to Figure 7 on page 15. If the device is not detected, power cycle the SmartFusion2 Security Evaluation Kit and click "scan for hardware changes" (as shown in Figure 14 on page 20) in **Device Manager**.

8. If the host PC has any other installed drivers (previous versions of Jungo drivers) for the SmartFusion2 PCIe device, uninstall them and make sure that the PCI Device is displayed in the Device Manager window as shown in Figure 7.

***Figure 7 •*** **Device Manager - PCIe Device Detection**




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

## 2.5.2 Driver Installation

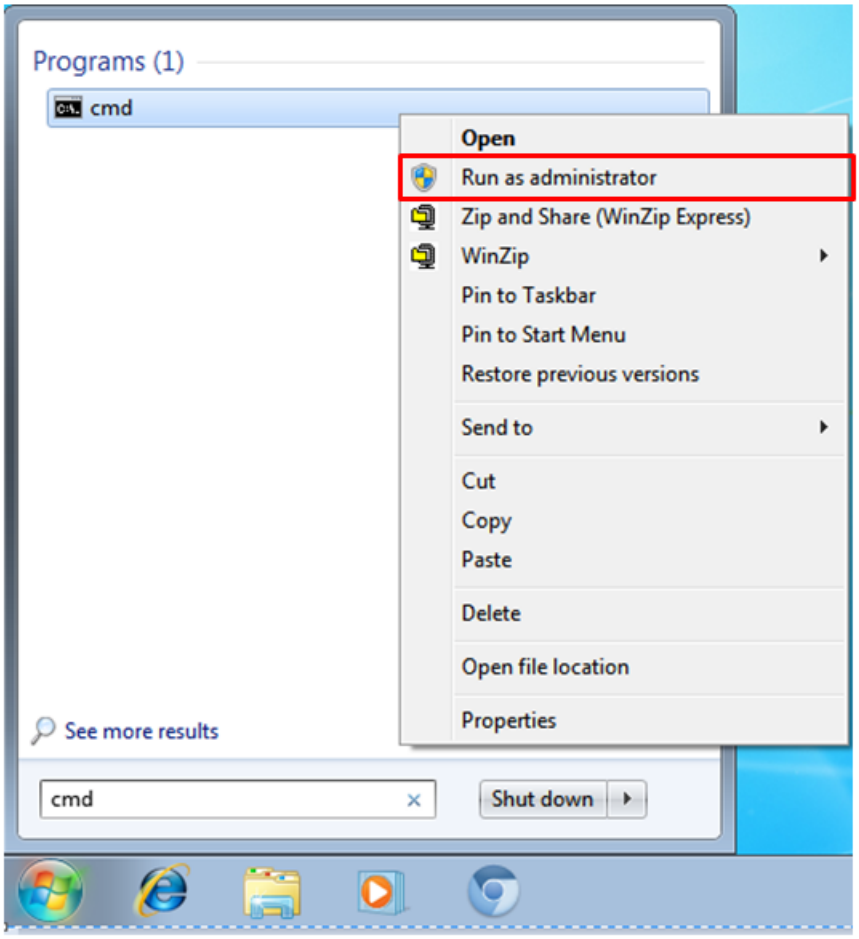
The PCIe demo uses a driver framework provided by Jungo WinDriverPro. To install the PCIe drivers on host PC for SmartFusion2 Security Evaluation Kit, run the following steps:

1. Extract the PCIe_Demo.rar to C: drive. The `PCIe_Demo.rar` is located in the provided design files:*M2S_PCIE_MSSHPDMA_DEMO_DF\Drivers_64bitOS\PCIe_Demo.rar*

**Note:** The extracted files of `PCIe_Demo.rar` must be in *C:* because the PCIe_Demo application calls these files from this location. Installing these drivers require host PC administration rights. The installation steps are described in the following.

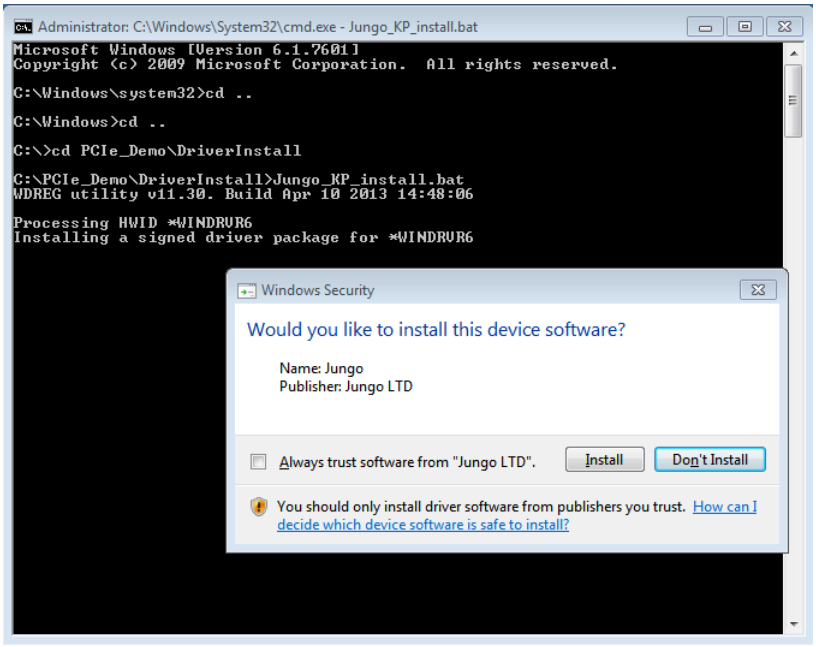
2. Run the batch file *C:\PCIe_Demo\DriverInstall\Jungo_KP_install.bat* to install the PCIe drivers for the SmartFusion2 device.
3. Open the Command Prompt as an administrator to run the batch file **C:\PCIe_Demo\DriverInstall\Jungo_KP_install.bat**, as shown in Figure 9 on page 17.

***Figure 8 •*** **Open Command Prompt**



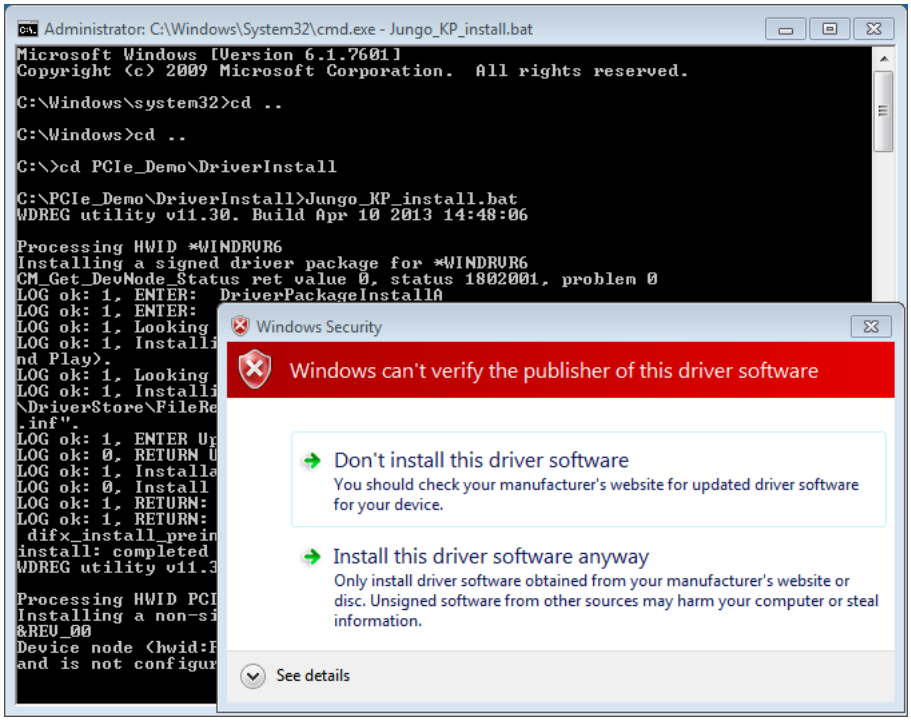

4. Navigate to the required folder in *C:* drive and execute the **Jungo_KP_install.bat**.
5. Click **Install** in the windows security dialog box, as shown in Figure 9 on page 17.

***Figure 9 •*** **Install Jungo**



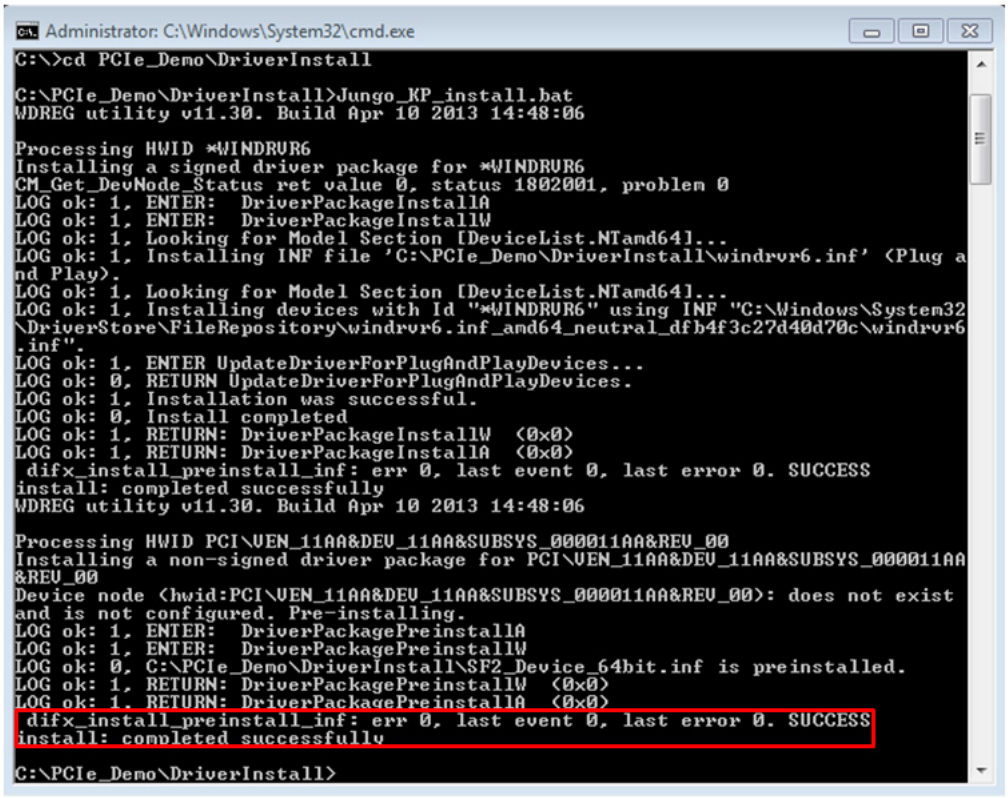

6. Click **Install this driver software anyway** if the window appears, as shown in Figure 10.

***Figure 10 •*** **Jungo Driver Installation**



The completed successfully message is displayed, as shown in Figure 11.

***Figure 11 •*** **Completed Successfully**



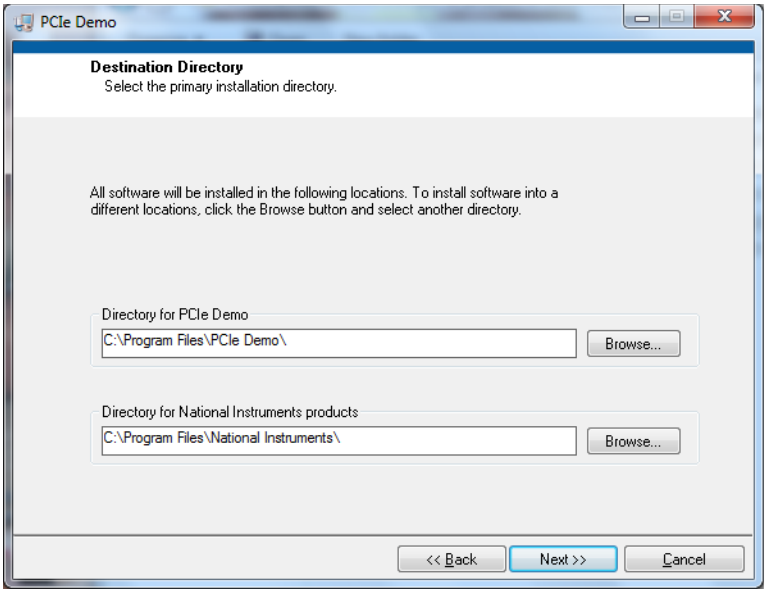

## 2.5.3 Demo GUI

SmartFusion2 PCIe Demo GUI runs on the host PC to communicate with the SmartFusion2 PCIe endpoint device. The GUI provides PCIe link status, driver information, and demo controls to perform the DMA transactions to observe the data throughputs. GUI invokes the PCIe driver installed on the host PC and provides commands to the driver according to the selection made.

To install the GUI, use the following steps:

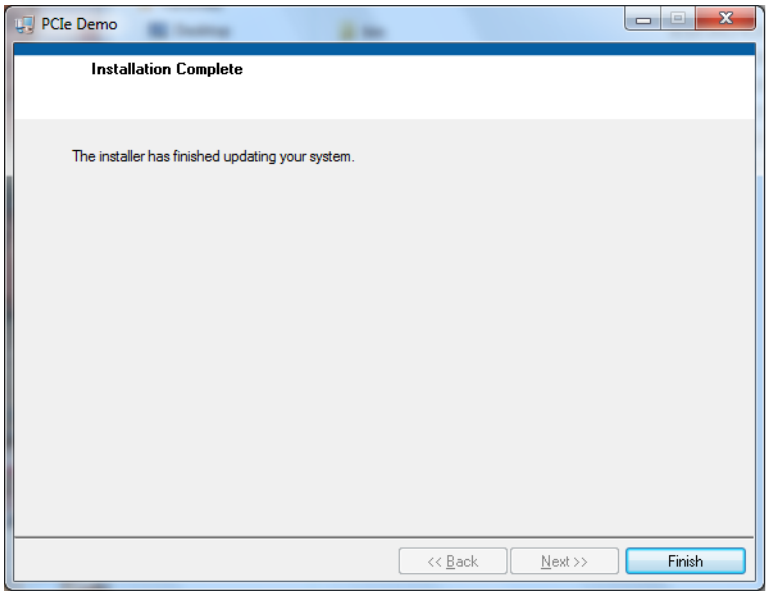
1. Download the PCIe demo GUI installer from *http://soc.microsemi.com/download/rsc/?f=PCIe_Demo_GUI_Installer*
2. Extract the **PCIe_Demo_GUI_Installer.rar**.
3. Double-click the **setup.exe** in the provided GUI installation (*PCIe_Demo_GUI_Installer/setup.exe*). Apply default options as shown in Figure 12 on page 19.

***Figure 12 •*** **GUI Installation**




4. Click **Next** and **Finish** to complete the installation. Figure 13 shows the **Successful GUI Installation** window.

***Figure 13 •*** **Successful Installation of GUI**

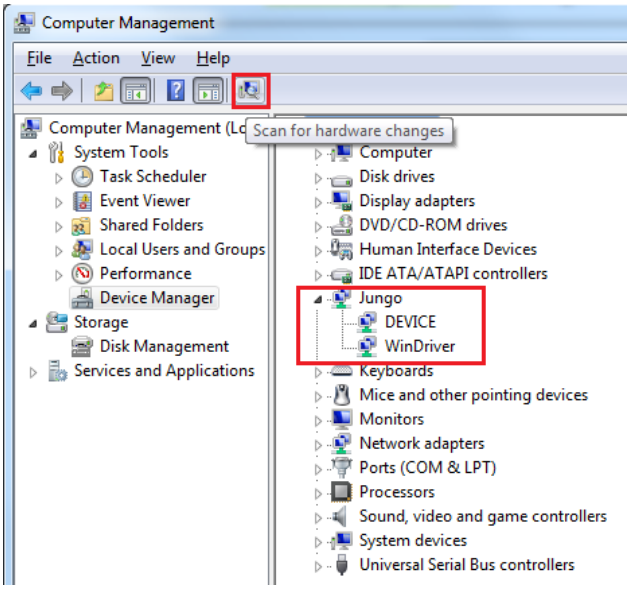



5. Shut down the host PC.
6. Power-cycle the SmartFusion2 Security Evaluation Kit board.
7. Restart the host PC.

## 2.5.4 Steps to Run the Design

1. Check the host PC **Device Manager** for the drivers. It will be similar to Figure 14. If the device is not detected, power-cycle the SmartFusion2 Security Evaluation Kit and click **scan for hardware changes** in **Device Manager.**

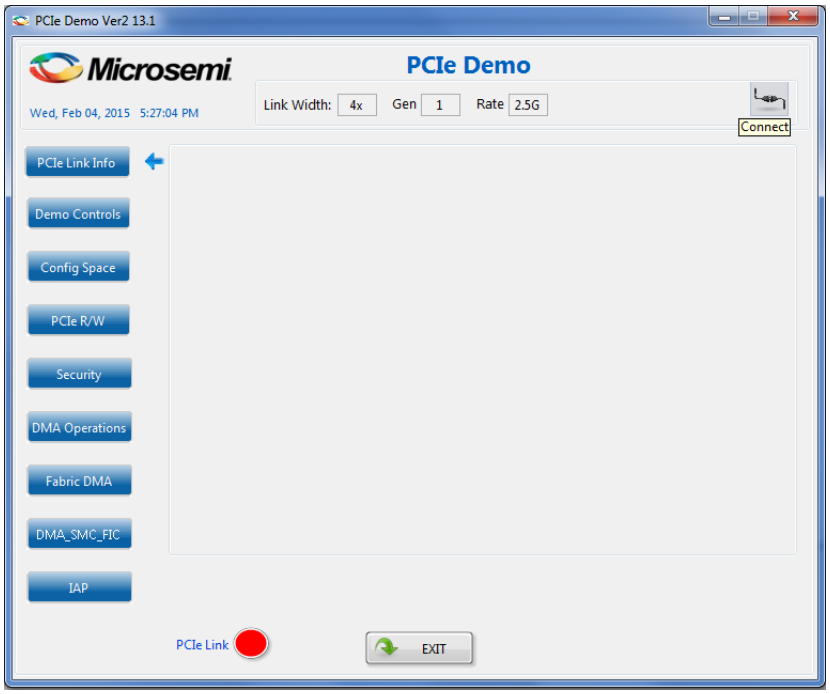
***Figure 14 •*** **Device Manager - PCIe Device Detection**




**Note:** If a warning symbol appears on the **Device** or **WinDriver** icons in **Device Manager**, uninstall them and start from step1 of driver installation.

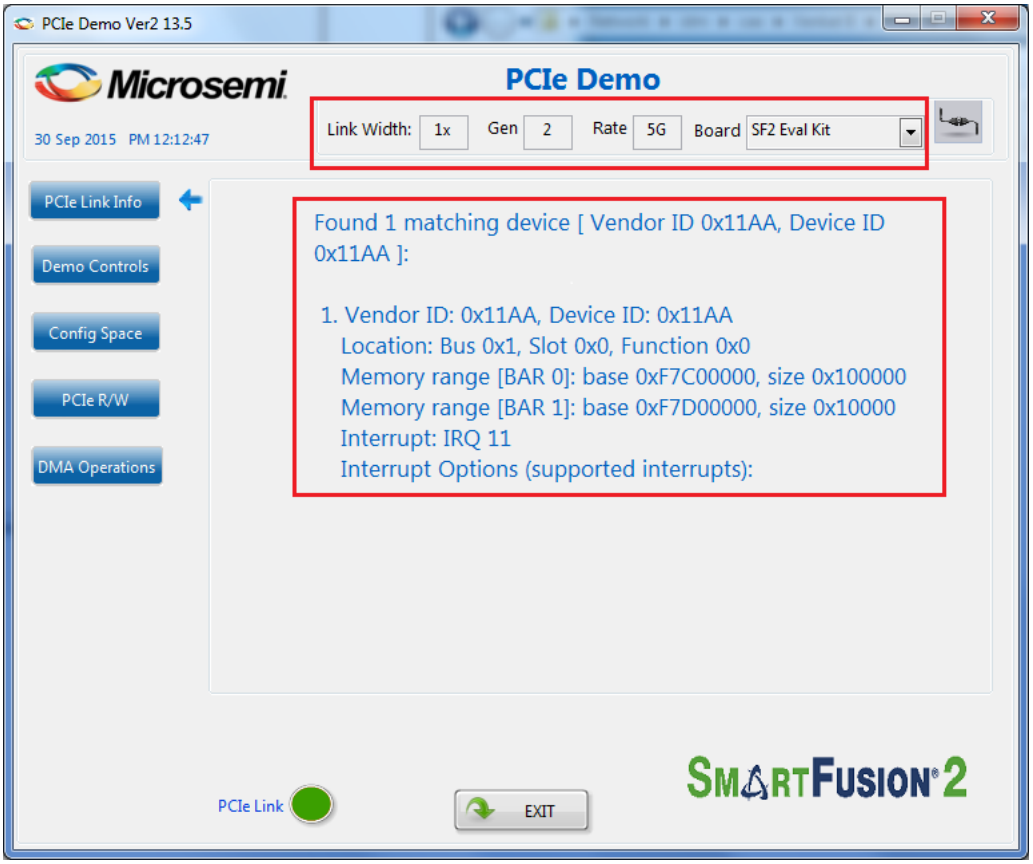
2. Invoke the GUI from **ALL Programs > PCIe_Demo > PCIe Demo GUI**. The GUI is displayed as shown in Figure 15.

***Figure 15 •*** **PCIe Demo GUI**



3. Click the **Connect** button at top right corner (highlighted below) of the GUI. The messages are displayed on the GUI as shown in Figure 16.

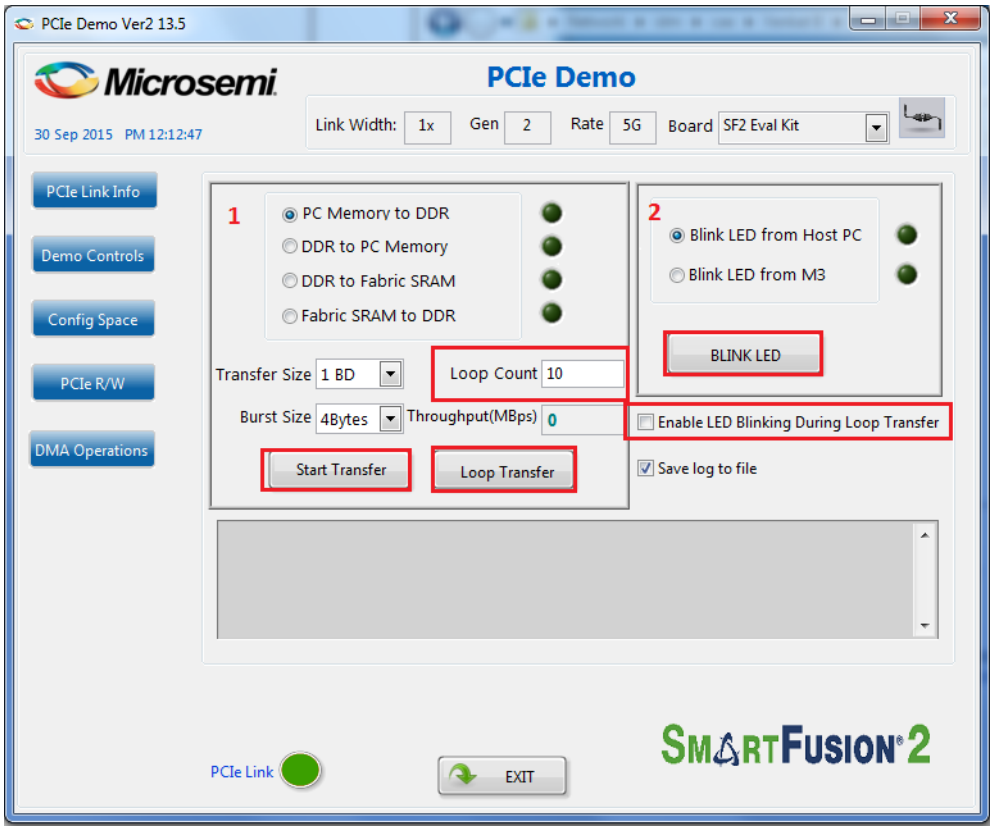
***Figure 16 •*** **Version Information**




4. The three buttons on the left side (that is **Demo Controls**, **Config Space**, and **PCIe R/W** button functions) are same as in the PCIe Control Plane Demo. For more information, refer to the *UG0456: SmartFusion2 SoC FPGA PCIe Control Plane Demo User Guide*.

5. Click **DMA Operations**. The controls are displayed as shown in Figure 17.

***Figure 17 •*** **DMA Operations**




The controls in the box marked with 1 are related to DMA operations.
The transfer size (as shown in Figure 18 on page 23) can be selected from 64 KB to 1 MB for:

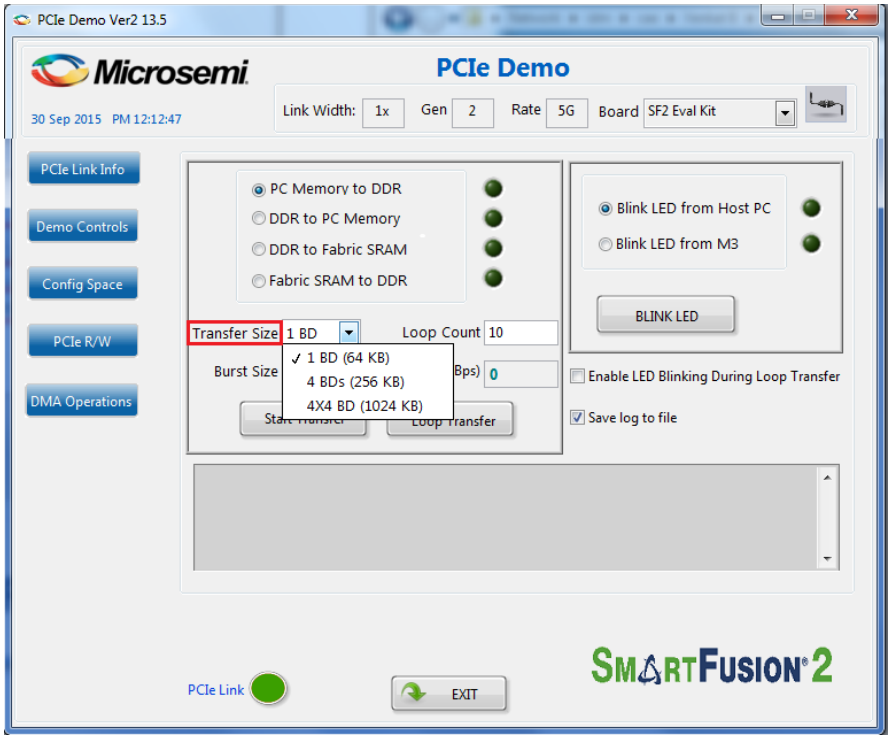
- PC memory to LPDDR
- LPDDR to PC memory

The transfer size is provided in steps of HPDMA buffer descriptor size (64 KB). For more information on buffer description, refer to the "HPDMA" chapter in the *UG0331: SmartFusion2 Microcontroller Subsystem User Guide*.

The **Transfer Size** can be selected from 16 KB to 128 KB for:

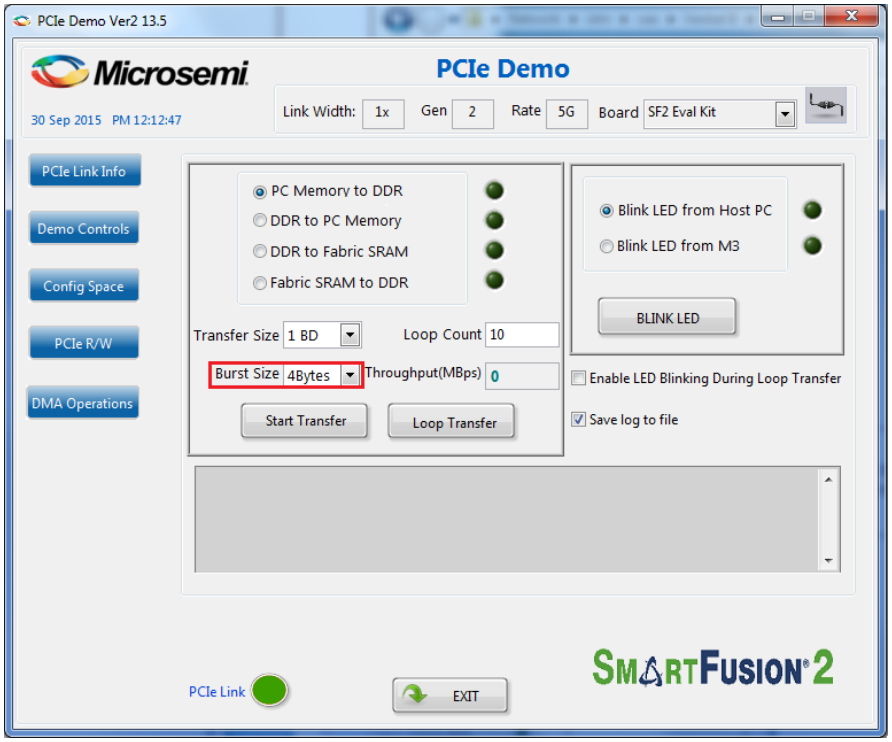
- Fabric LSRAM to LPDDR
- LPDDR to Fabric LSRAM

***Figure 18 •*** **Transfer Size Selection**



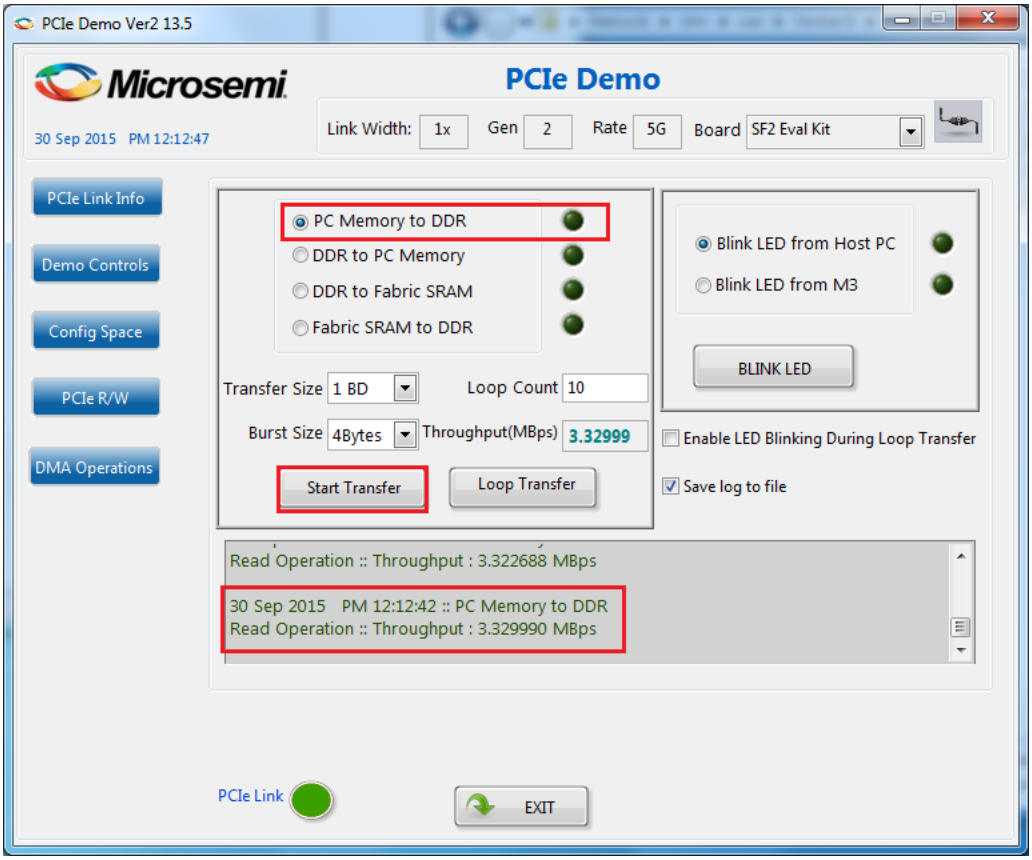

6. The **Burst Size** (TLP size) is fixed to 4 bytes as the HPDMA does not support burst and the actual size of the PCIe packet is the width of a single AHB transfer which is 4 bytes (32-bit AHB) as shown in Figure 19.

***Figure 19 •*** **Transfer Size Selection**



7. Select the type of DMA transfer as **PC Memory to SF2 LPDDR**, select any **Transfer Size** and click **Start Transfer** as shown in Figure 20. After completion of data transfer, the GUI displays the **Throughput** in Mbps.

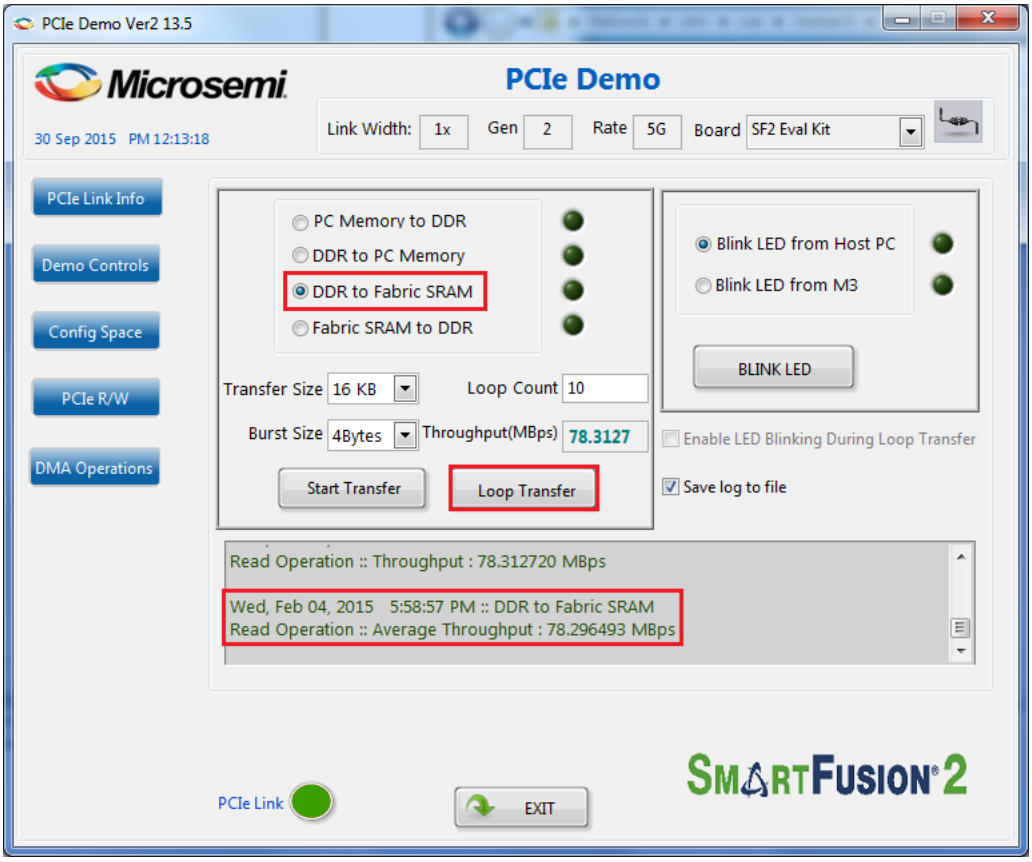
***Figure 20 •*** **PC Memory to LPDDR Transfer Throughput**




8. Enter 10 in the **Loop Count** box and click **Loop Transfer** to perform 10 sequential DMA transactions. The PCIe_Demo window displays the DMA throughput after completion of data transfer. Figure 21 on page 25 shows the throughput of DMA transactions from host PC to LPDDR. The average throughput is logged and the log file is stored at *C:\PCIe_Demo\DriverInstall* in the host PC.

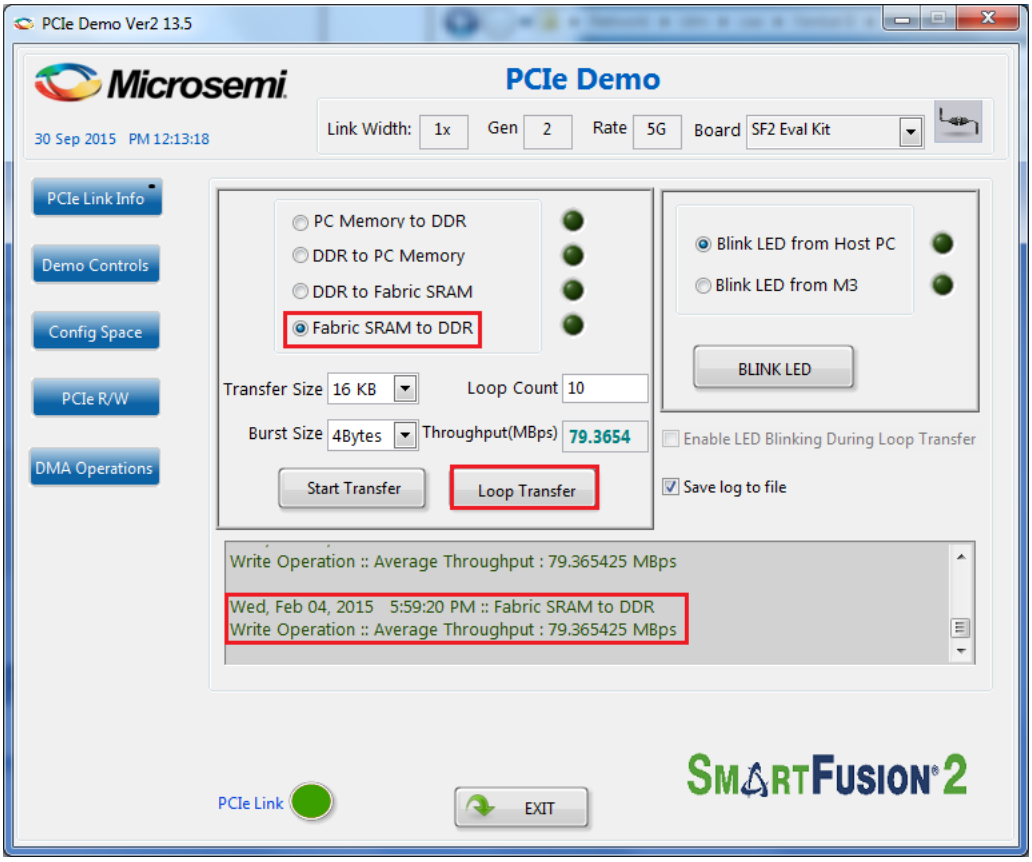
9. Select the type of DMA transfer as LPDDR **to Fabric SRAM**, select any **Transfer Size** and click **Loop Transfer** as shown in Figure 21.

***Figure 21 •*** **LPDDR to PC Memory Transfer Throughput**



10. Select the type of DMA transfer as **Fabric SRAM to LPDDR**, select any **Transfer Size** and click **Loop Transfer** as shown in Figure 22.

***Figure 22 •*** **Fabric SRAM to LPDDR Transfer Throughput**




11. The LEDs on the board can be blinked in parallel to the DMA operations by using the LED controls on the right side of GUI. The **Do DMA transfer and LED toggle** check box need to be selected to do the LED blinking from host PC and DMA transfers.
12. Click **Exit** to quit the demo.

## 2.6 Summary

This demo shows how to implement a PCIe data plane design using MSS HPDMA. Data transfer occurs between PCIe, SmartFusion2 LSRAM, and SmartFusion2 DDR. Throughput for data transfers is dependent on host PC system configuration, type of PCIe slots used.

Table 3 shows the throughput values observed on the HP Workstation Z220 - CMT.

***Table 3 •*** **Throughput Summary**

| DMA Transfer Type | DMA Transfer Size | Throughput (M Bytes Per Second) | | | |
|---|---|---|---|---|---|
| | | Gen 1 | | Gen 2 | |
| | | Single Transfer | Loop Transfer | Single Transfer | Loop Transfer |
| Host PC Memory to LPDDR | 1 BD (64KB)/ 4 BD (256KB)/ 4x4 BD (1MB) | 2.3 | 2.1 | 2.4 | 2.1 |
| LPDDR to Host PC Memory | 1 BD (64KB)/ 4 BD (256KB)/ 4x4 BD (1MB) | 22 | 22 | 32 | 32 |
| LPDDR to Fabric SRAM | 16KB/ 32KB/ 64KB/ 128KB | 78 | 78 | 78 | 78 |
| Fabric SRAM to LPDDR | 16KB/ 32KB/ 64KB/ 128KB | 79 | 79 | 79 | 79 |

# 3 Revision History

The following table shows important changes made in this document for each revision.

| Revision | Changes |
|---|---|
| Revision 6<br>(May 2016) | Updated the document for Libero v11.7 software release (SAR 77405). |
| Revision 5<br>(October 2015) | Updated the document for Libero v11.6 software release (SAR 71815). |
| Revision 4<br>(February 2015) | Updated the document for Libero v11.5 software release (SAR 64507). |
| Revision 3<br>(August 2014) | Updated the document for Libero v11.4 software release (SAR 59739). |
| Revision 2<br>(April 2014) | Updated the document for Libero v11.3 software release (SAR 56582). |
| Revision 1<br>(November 2013) | Updated the document for Libero v11.2 software release (SAR 52901). |
| Revision 0<br>(September 2013) | Initial Release |

# 4 Product Support

Microsemi SoC Products Group backs its products with various support services, including Customer Service, Customer Technical Support Center, a website, electronic mail, and worldwide sales offices. This appendix contains information about contacting Microsemi SoC Products Group and using these support services.

## 4.1 Customer Service

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

From North America, call 800.262.1060
From the rest of the world, call 650.318.4460
Fax, from anywhere in the world, 408.643.6913

## 4.2 Customer Technical Support Center

Microsemi SoC Products Group staffs its Customer Technical Support Center with highly skilled engineers who can help answer your hardware, software, and design questions about Microsemi SoC Products. The Customer Technical Support Center spends a great deal of time creating application notes, answers to common design cycle questions, documentation of known issues, and various FAQs. So, before you contact us, please visit our online resources. It is very likely we have already answered your questions.

## 4.3 Technical Support

For Microsemi SoC Products Support, visit
http://www.microsemi.com/products/fpga-soc/design-support/fpga-soc-support.

## 4.4 Website

You can browse a variety of technical and non-technical information on the Microsemi SoC Products Group home page, at http://www.microsemi.com/products/fpga-soc/fpga-and-soc.

## 4.5 Contacting the Customer Technical Support Center

Highly skilled engineers staff the Technical Support Center. The Technical Support Center can be contacted by email or through the Microsemi SoC Products Group website.

### 4.5.1 Email

You can communicate your technical questions to our email address and receive answers back by email, fax, or phone. Also, if you have design problems, you can email your design files to receive assistance. We constantly monitor the email account throughout the day. When sending your request to us, please be sure to include your full name, company name, and your contact information for efficient processing of your request.

The technical support email address is soc_tech@microsemi.com.

### 4.5.2 My Cases

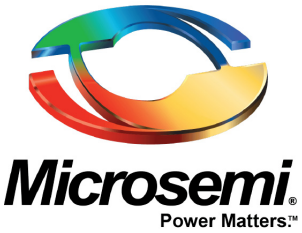
Microsemi SoC Products Group customers may submit and track technical cases online by going to My Cases.

### 4.5.3 Outside the U.S.

Customers needing assistance outside the US time zones can either contact technical support via email (soc_tech@microsemi.com) or contact a local sales office. Visit About Us for sales office listings and corporate contacts.

## 4.6 ITAR Technical Support

For technical support on RH and RT FPGAs that are regulated by International Traffic in Arms Regulations (ITAR), contact us via soc_tech@microsemi.com. Alternatively, within My Cases, select **Yes** in the ITAR drop-down list. For a complete list of ITAR-regulated Microsemi FPGAs, visit the ITAR web page.




**Microsemi Corporate 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
E-mail: sales.support@microsemi.com

© 2016 Microsemi Corporation. All rights reserved. Microsemi and the Microsemi logo are trademarks of Microsemi Corporation. All other trademarks and service marks are the property of their respective owners.

Microsemi Corporation (Nasdaq: MSCC) offers a comprehensive portfolio of semiconductor and system solutions for communications, defense & security, aerospace 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**.

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.

50200501-6/5.16