
SmartFusion2 PCIe MSS HPDMA

Demo Guide

Superseded

April 2014



Revision History

Date	Revision	Change
22 April 2014	2	Third Release
29 November 2013	1	Second Release
04 October 2013	0	First Release

Confidentiality Status

This is a non-confidential document.

Superseded

Table of Contents

Preface	4
About this document	4
Intended Audience	4
References	4
Microsemi Publications	4
1 PCIe Data Plane Demo Using MSS HPDMA	-5
Introduction	5
Tools Required	5
Demo Design	6
Introduction	6
Demo Design Features	8
Demo Design Description	8
Setting up the Demo Design	10
Running the Demo Design	11
Setting up the Device	12
Programming the Device	13
Driver Installation	15
Demo GUI	16
Steps to Run the Design	17
Summary	25
A List of Changes	-26
B Product Support	-27
Customer Service	27
Customer Technical Support Center	27
Technical Support	27
Website	27
Contacting the Customer Technical Support Center	27
Email	27
My Cases	28
Outside the U.S.	28
ITAR Technical Support	28

Preface

About this document

This demo is for SmartFusion2 SoC FPGA devices. It provides instructions on how to use the corresponding reference design.

Intended Audience

SmartFusion2 devices are used by:

- FPGA designers
- Embedded designers
- System-level designers

References

Microsemi Publications

- [SmartFusion2 Microcontroller Subsystem User Guide](#)
- [SmartFusion2 SoC FPGA High Speed Serial Interfaces User Guide](#)
- [PCIe Control Plane Demo User Guide](#)

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

PCIe Data Plane Demo Using MSS HPDMA

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 Development 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 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 [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).
- **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).
- **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 [SmartFusion2 SoC FPGA High Speed Serial Interfaces User Guide](#).

Tools Required

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

Table 1 • Reference Design Requirements and Details

Reference Design Requirements and Details	Description
Hardware Requirements	
SmartFusion2 Development Kit <ul style="list-style-type: none">• FlashPro4 programmer• 12V adapter• PCI edge card ribbon cable	Rev D or later
Host PC or Laptop (8GB RAM)	Any 64-bit Windows Operating System
Host PC with an available PCIe 2.0 compliant slot (x4 or greater)	

Table 1 • Reference Design Requirements and Details (continued)

Software Requirements	
Libero® System-on-Chip (SoC)	11.3
FlashPro programming software	11.3

Demo Design

Introduction

The demo design files are available for download from the following path in the Microsemi® website:
http://www.microsemi.com/soc/download/rsc/?f=M2S_PCIE_MSSHPDMA_DEMO_DF.

The demo design files include:

- Drivers_64bitOS
- GUI
- 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.

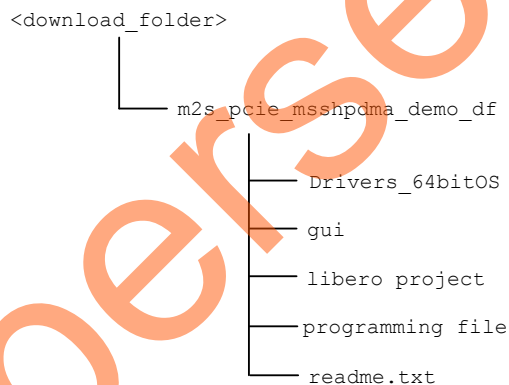


Figure 1 • Demo Design Files Top Level Structure

Figure 2 describes the demo.

In Figure 2, the red colored line shows the data flow between host PC memory and DDR3 memory and the green colored line shows the data flow between the fabric LSRAM and the DDR3 memory.

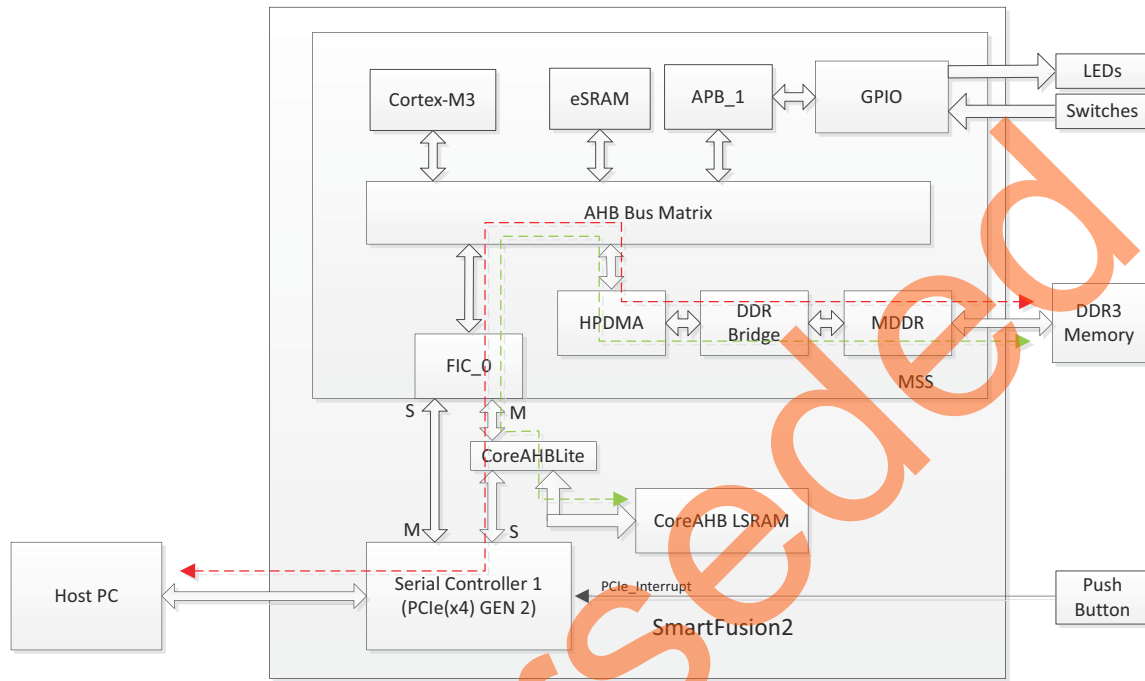


Figure 2 • PCIe Data Plane Demo Block Diagram

In this demo design different blocks are configured as following:

- The SERDES_IF_1 in the SmartFusion2 device is configured for PCIe 2.0, x4 lanes and Gen2 rate.
- The MDDR controller is configured to use the DDR3 memory on the SmartFusion2 Development Kit.
- The CoreAHBLSRAM IP is configured to use the 128 KB of LSRAM.
- The AHB master interface of SERDES_IF_1 is enabled and connected to the AHB slave interface of FIC_0 to access the MSS peripherals. The AHB slave interface of SERDES_IF_1 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 300 MHz and the M3_CLK clock and Serial Controller 1 clock base are configured to 100 MHz.

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 DDR3 memory using the HPDMA controller
- DMA data transfers between the DDR3 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 Development Kit
- Displays the SmartFusion2 PCIe Configuration Space
- Controls the status of LEDs on the SmartFusion2 Development 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 Development Kit

Demo Design Description

The PCIe link is configured at GEN2 rate and four lanes and is capable of throughputs greater than 700 MBps. In this demo, the microcontroller subsystem (MSS) using an AHB 32-bit data bus and clock rate of 100 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.

PC Memory to DDR3

A data transfer from PC memory to the DDR3 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 (CpID) 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 DDR3 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 DDR3, 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.

DDR3 to PC Memory

A data transfer from the DDR3 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 DDR3 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 DDR3 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.

DDR3 to Fabric SRAM Block

The FPGA design of this demo contains a 128 KB fabric SRAM block. Data transfer from the DDR3 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 DDR3 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.

Fabric SRAM to DDR3

A data transfer from the fabric SRAM to the DDR3 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 DDR3 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 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
 - DDR3 to host PC
In this DMA transfer, HPDMA reads the data from external DDR3 memory through MDDR and writes to host PC buffer through FIC_0 and PCIe.
 - Host PC to DDR3
In this DMA transfer, HPDMA reads the data from host PC buffer through FIC_0 and PCIe and writes to external DDR3 memory through MDDR.

- Fabric LSRAM to DDR3
In this DMA transfer, HPDMA reads the data from Fabric LSRAM through FIC_0 and writes to external DDR3 memory through MDDR.
- DDR3 to Fabric LSRAM
In this DMA transfer, HPDMA reads the data from external DDR3 memory through MDDR and writes to Fabric LSRAM through FIC_0.

The Cortex-M3 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
The 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.

Setting up the Demo Design

1. Connect the FlashPro4 programmer to the J59 connector of SmartFusion2 SoC FPGA Development Kit.
2. Connect the jumpers on the SmartFusion2 SoC FPGA Development Kit, as shown in [Table 2](#).
 - **Caution:** Before making the jumper connections, switch off the power supply switch, SW7.

Table 2 • SmartFusion2 SoC FPGA Development Kit Jumper Settings

Jumper	Pin (from)	Pin (to)	Comments
J70, J93, J94, J117, J123, J142, J157, J160, J167, J225, J226, J227	1	2	default
J2	1	3	default
J23	1	3	default

3. Connect the power supply to the J18 connector.

Running the Demo Design

To run the demo design follow these steps:

1. Download the demo design from:
www.microsemi.com/soc/download/rsc/?f=m2s_pcie_msshpdma_demo_df.
2. Switch on the power supply switch SW7 on the SmartFusion2 SoC FPGA Development Kit.
3. Run the FlashPro v11.3 software.
4. Click **New Project**.
5. In the **New Project** window, type the project name as PCIe_Data_Plane.

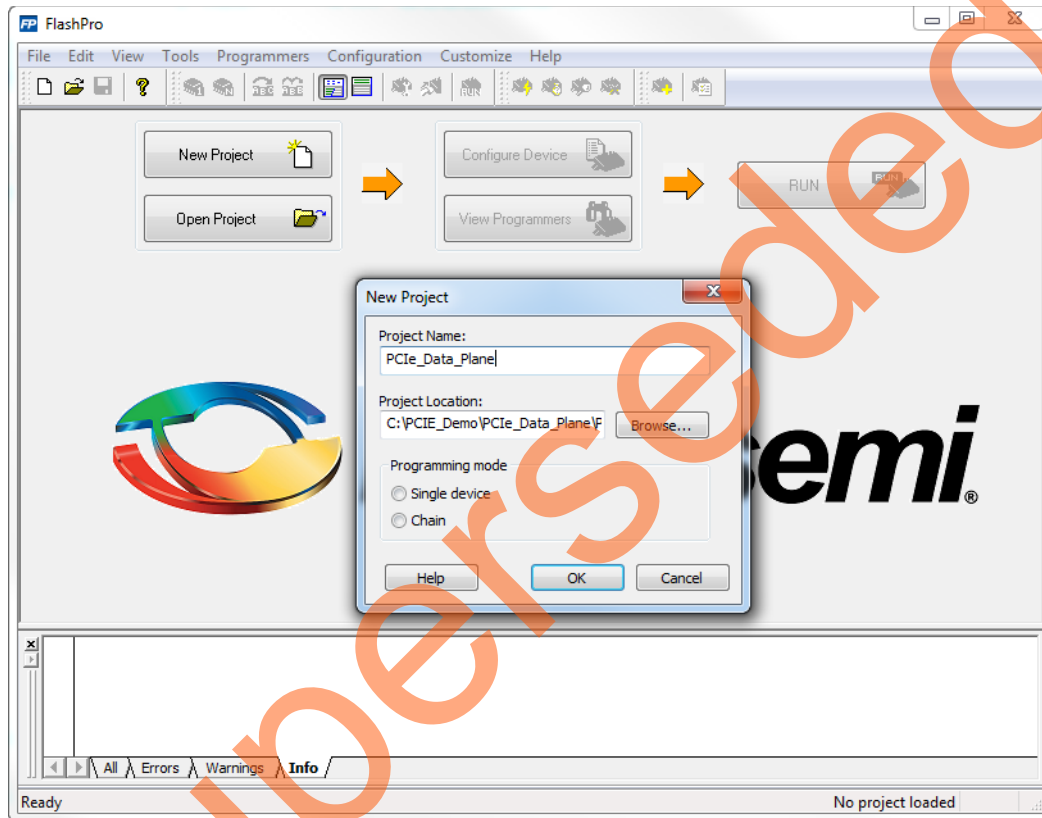


Figure 3 • FlashPro New Project

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.

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_Demo_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.

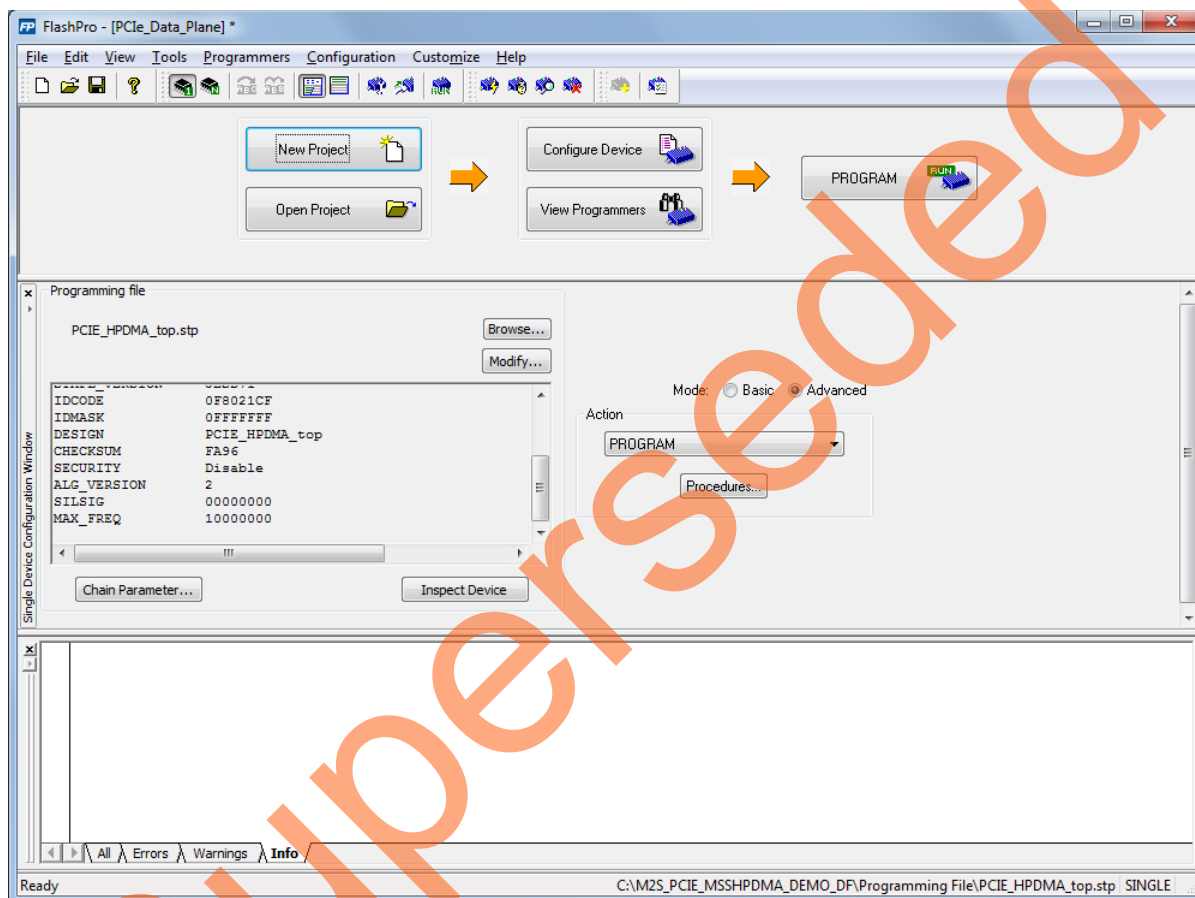


Figure 4 • FlashPro Project Configured

Programming the Device

Use the following steps to start programming the device:

1. Click **PROGRAM** to start programming the device. Wait until you get a message "PROGRAM PASSED".

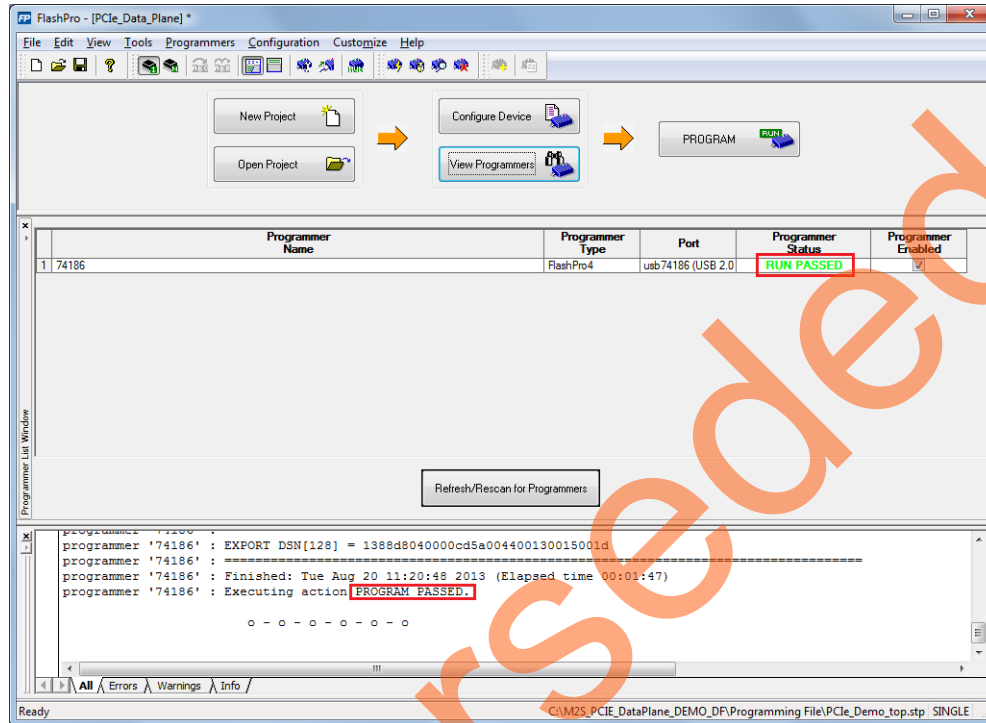


Figure 5 • FlashPro Programming Passed

2. After Successful programming, power off the SmartFusion2 Development Kit and shut down the host PC.
3. This Demo is designed to run on any PCIe Gen2 compliant slot. If your host PC does not support Gen2 compliant slot, the demo switches to Gen1 mode automatically. Connect J230 - PCIe Edge connector to the host PC's PCIe slot through PCI edge card ribbon cable.

Note: Host PC or laptop should be powered OFF while inserting the PCIe Edge Connector. If you do not power off the system, the PCIe device detection and selection of Gen1 or Gen2 mode do not occur properly. So it is recommended to power off the host PC during PCIe card insertion.

4. The board setup is shown in [Figure 6](#).

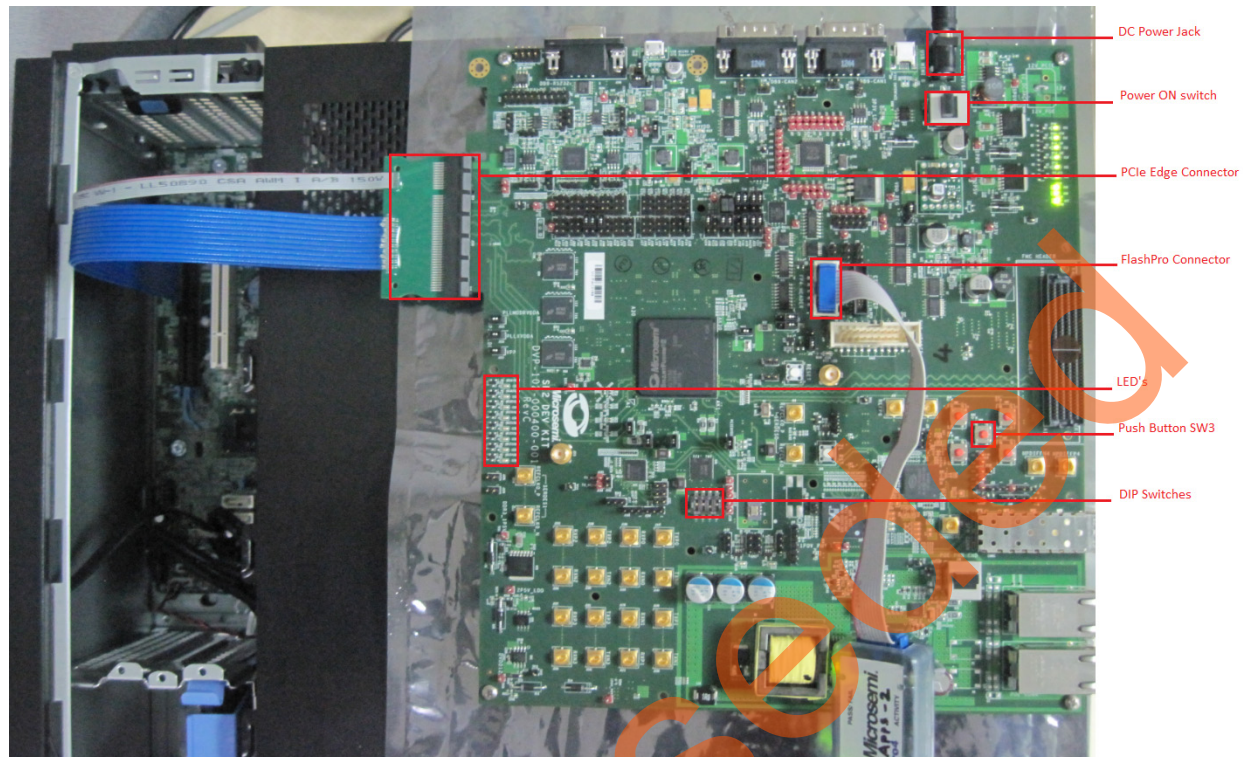


Figure 6 • SmartFusion2 Development Kit Setup

5. Switch the power supply switch **SW7** to **ON** position.
6. Power on the host PC and check the host PC's **Device Manager** for **PCIe Device**. It will be similar to [Figure 7](#). If the device is not detected, power cycle the SmartFusion2 Development Kit and click "scan for hardware changes" (as shown in [Figure 12 on page 17](#)) in **Device Manager**. 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](#).

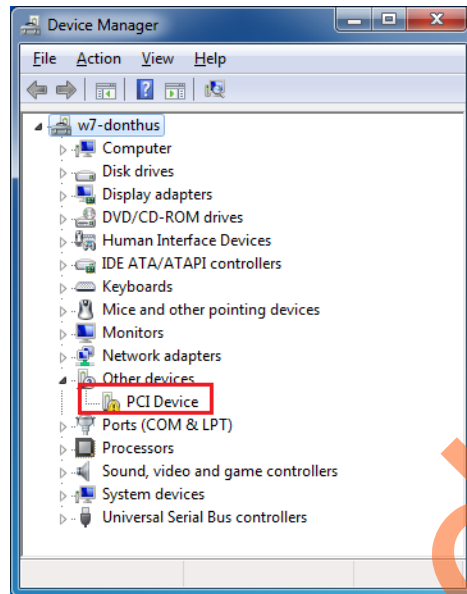


Figure 7 • Device Manager - PCIe Device Detection

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.

Driver Installation

The PCIe Demo uses a driver framework provided by Jungo WinDriverPro. To install the PCIe drivers on host PC for SmartFusion2 Development 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.

2. Run the batch file `C:\PCIe_Demo\DriverInstall\Jungo_KP_install.bat` to install the PCIe drivers for the SmartFusion2 device.

Note: Installing these drivers require host PC Administration rights.

3. In the **Windows Security** dialog, click **Install**.

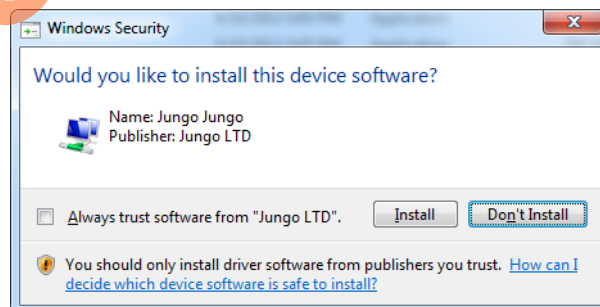


Figure 8 • Jungo Driver Installation

Note: If the installation is not in progress, run the command prompt in administrator mode and run the batch file `C:\PCIe_Demo\DriverInstall\Jungo_KP_install.bat`

4. Click **Install this driver software anyway** if the window appears as shown in Figure 9.

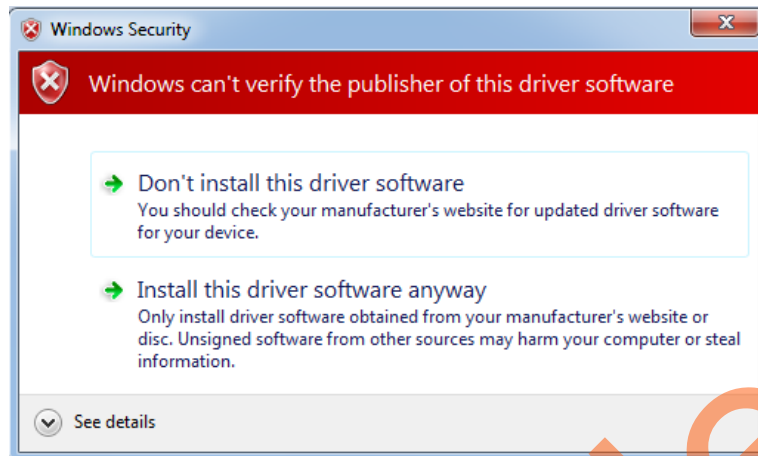


Figure 9 • Windows Security

Demo GUI

SmartFusion2 PCIe Demo GUI is a simple GUI that 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. Extract the PCIe_Demo_GUI_Installer.rar to C: drive. The PCIe_Demo_GUI_Installer.rar is located in the provided design files: *M2S_PCIE_MSSHPDMA_DEMO_DF/GUI*.
2. Double click **setup.exe** (*PCIe_Demo_GUI_Installer/setup.exe*). Apply default options and click **Next** to complete the installation as shown in Figure 10.

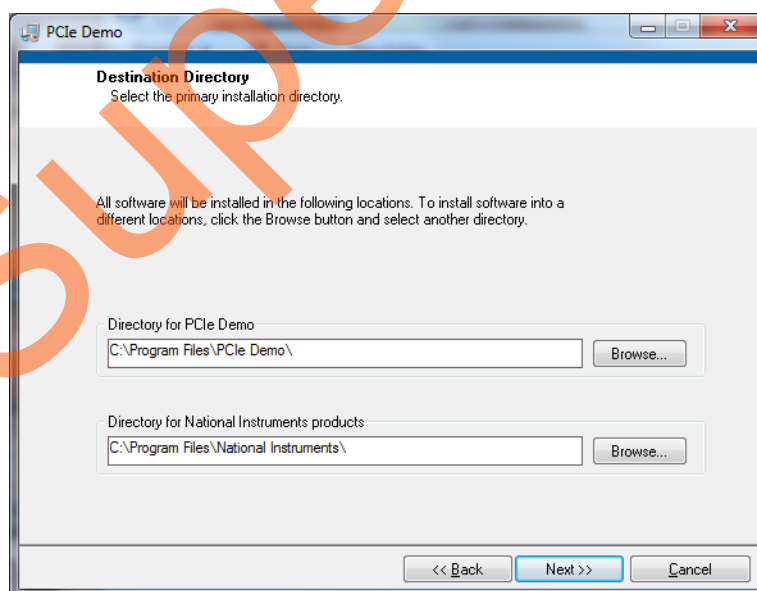


Figure 10 • GUI Installation

3. After successful installation the following window is displayed.

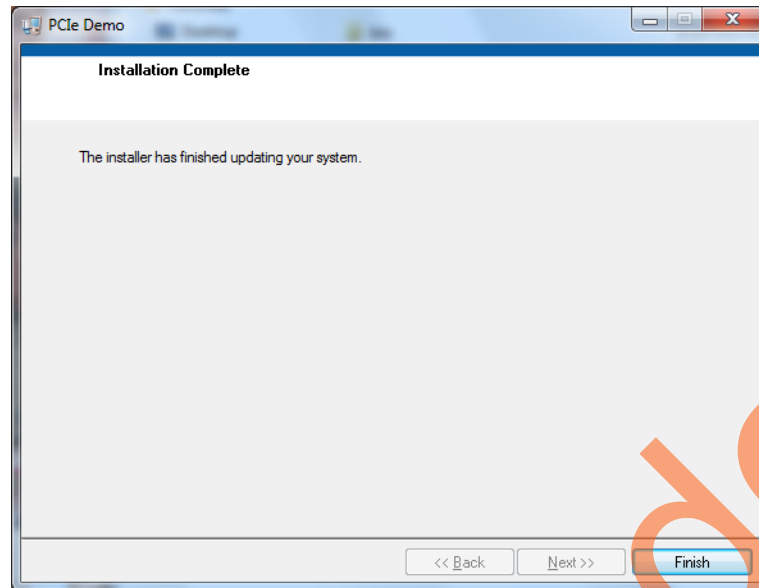


Figure 11 • Successful Installation of GUI

4. Shut down the host PC.
5. Power-Cycle the SmartFusion2 Development Kit board.
6. Restart the host PC.

Steps to Run the Design

1. Check the host PC **Device Manager** for the drivers. It will be similar to Figure 12. If the device is not detected, power cycle the SmartFusion2 Development Kit and click “scan for hardware changes” in **Device Manager**.

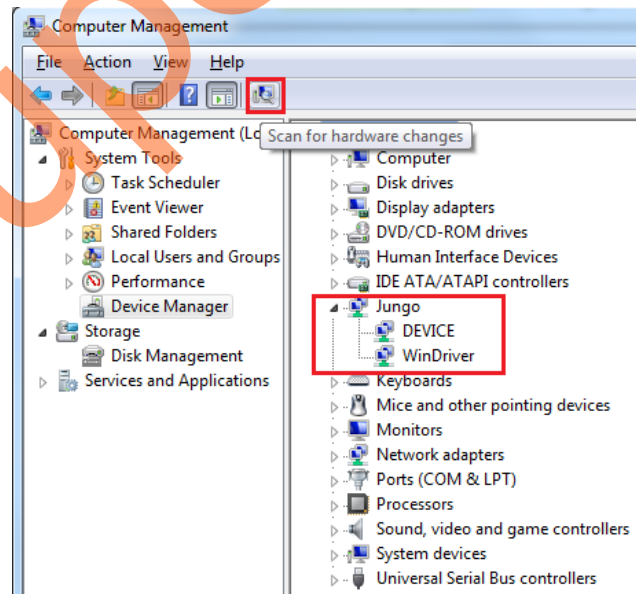


Figure 12 • Device Manager - PCIe Device Detection

- 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 13**.

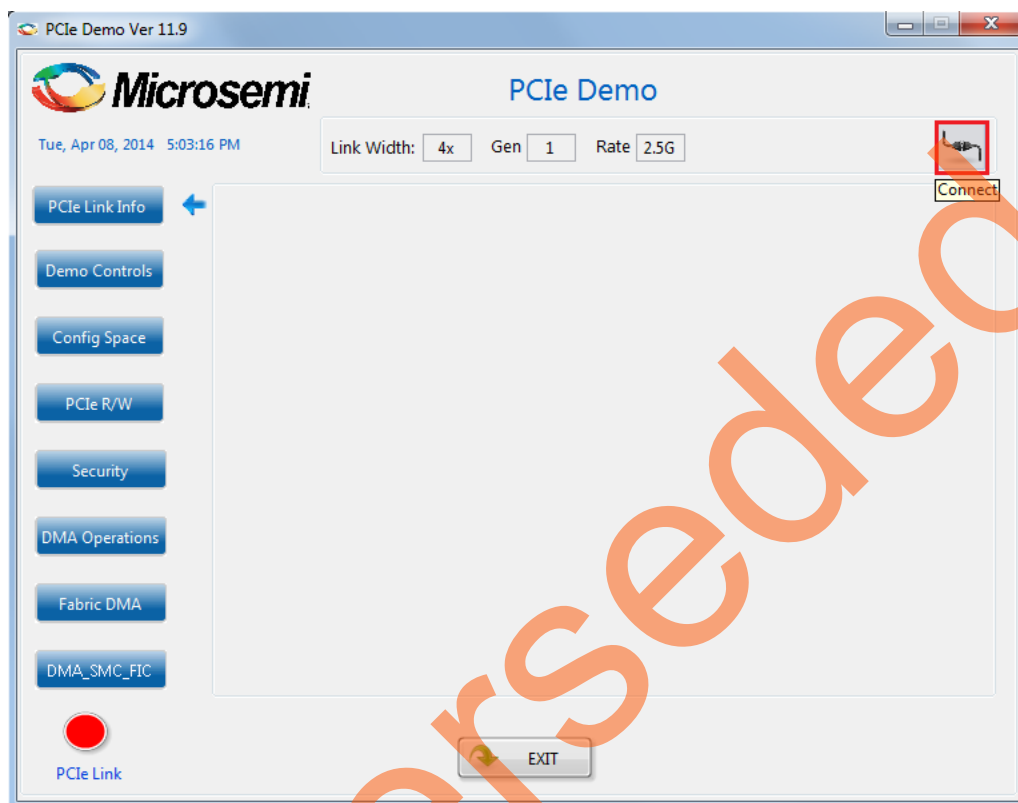


Figure 13 • PCIe Demo GUI

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 14.

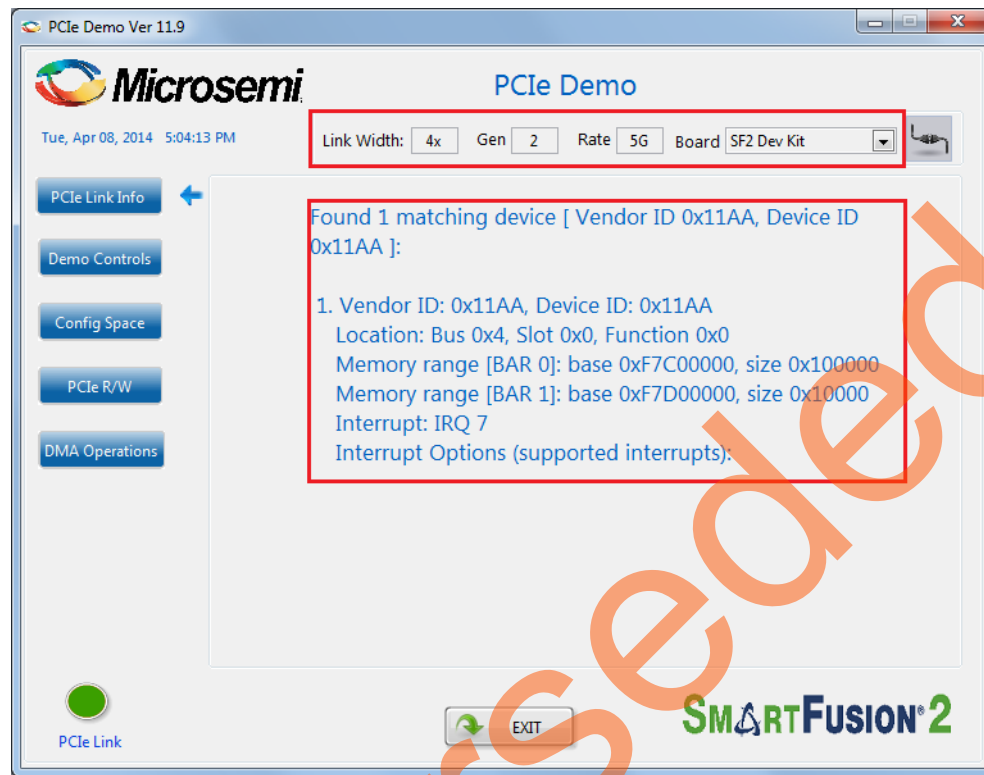


Figure 14 • Version Information

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 [PCIe Control Plane Demo User Guide](#).

5. Click **DMA Operations**. The controls are displayed as shown in [Figure 15](#).

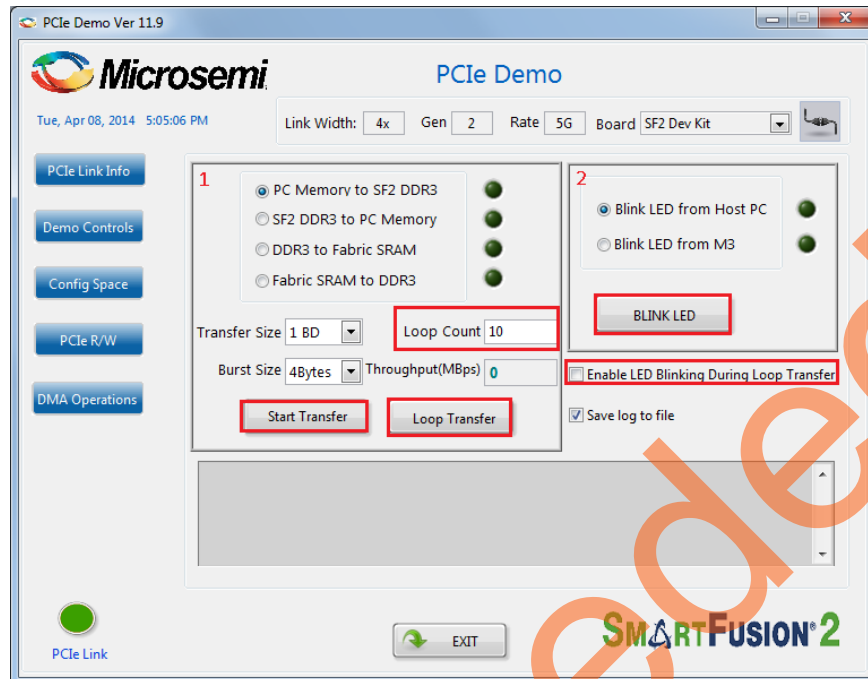


Figure 15 • DMA Operations

The controls in the box marked with 1 are related to DMA operations.

The transfer size (as shown in [Figure 16 on page 21](#)) can be selected from 64 KB to 1 MB for:

- PC memory to DDR3
- DDR3 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 [SmartFusion2 Microcontroller Subsystem User Guide](#).

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

- Fabric LSRAM to DDR3
- DDR3 to Fabric LSRAM

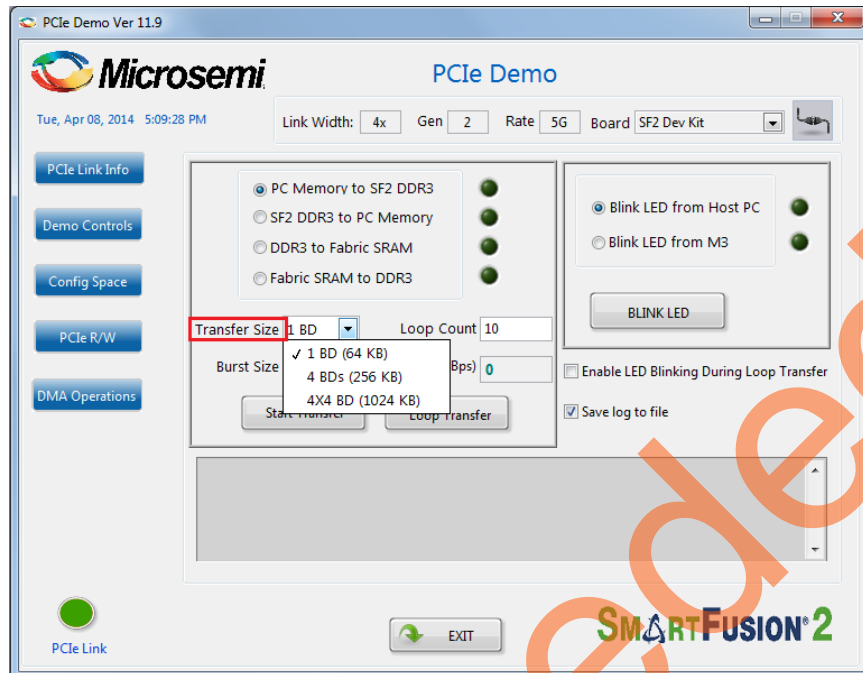


Figure 16 • Transfer Size Selection

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 17.

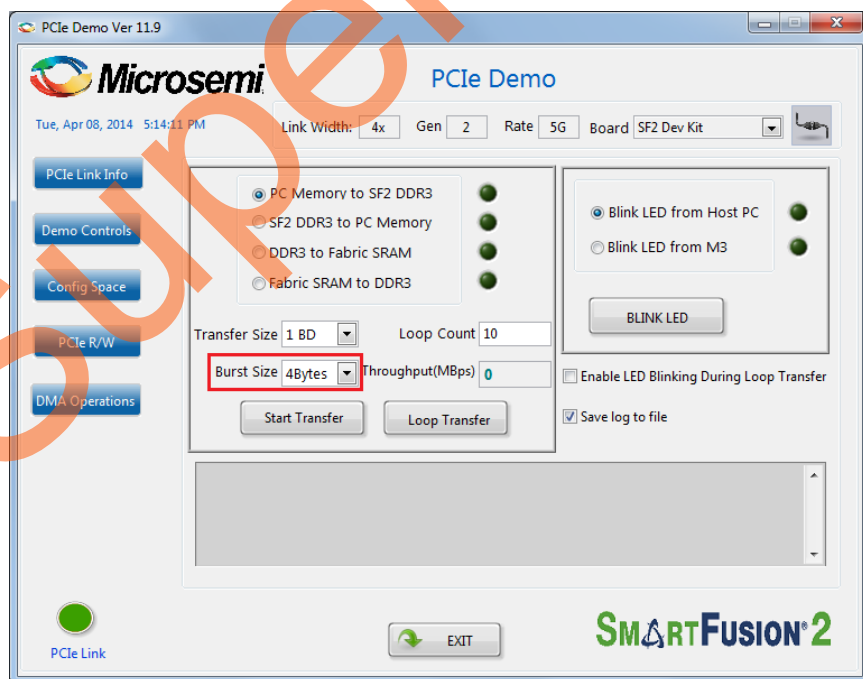


Figure 17 • Transfer Size Selection

7. Select the type of DMA transfer as **PC Memory to SF2 DDR3**, select any **Transfer Size** and click **Start Transfer** as shown in Figure 18. After completion of data transfer, the GUI displays the **Throughput** in MBps.

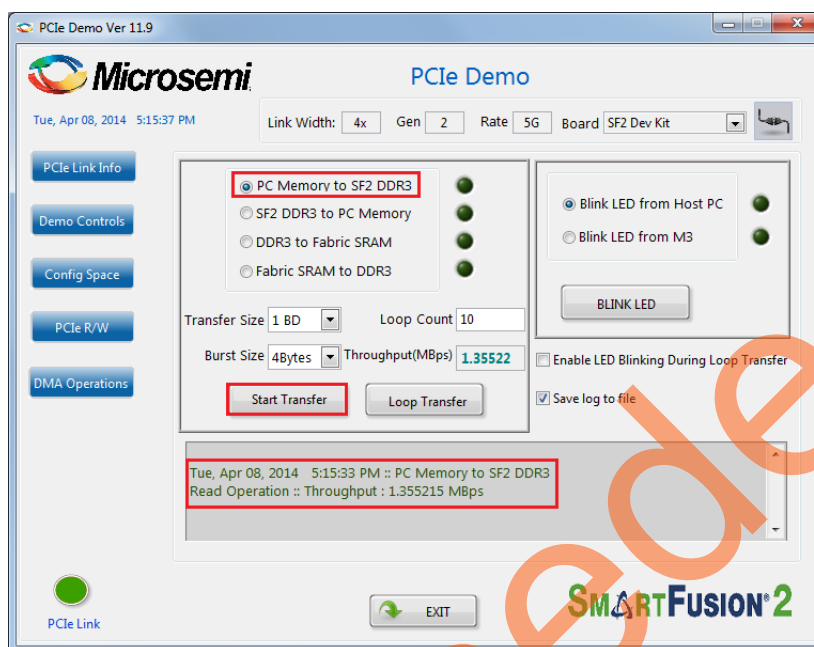


Figure 18 • PC Memory to DDR3 Transfer Throughput

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 19 shows the throughput of DMA transactions from host PC to DDR3. The average throughput is logged and the log file is stored at C:\PCIe_Demo\DriverInstall in the host PC.

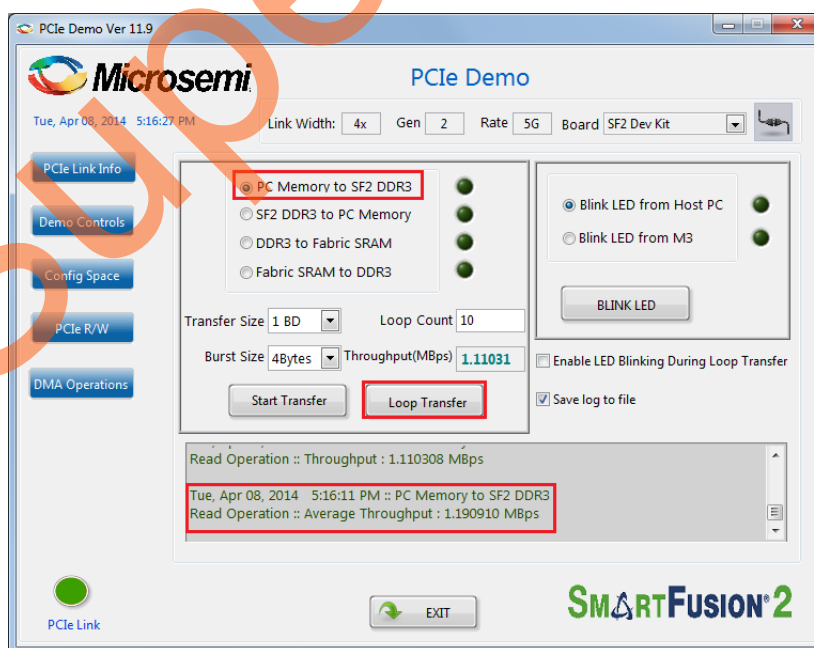


Figure 19 • Host PC to DDR3 Throughput

9. Select the type of DMA transfer as **SF2 DDR3 to PC Memory**, select any **Transfer Size** and click **Loop Transfer** as shown in Figure 20.

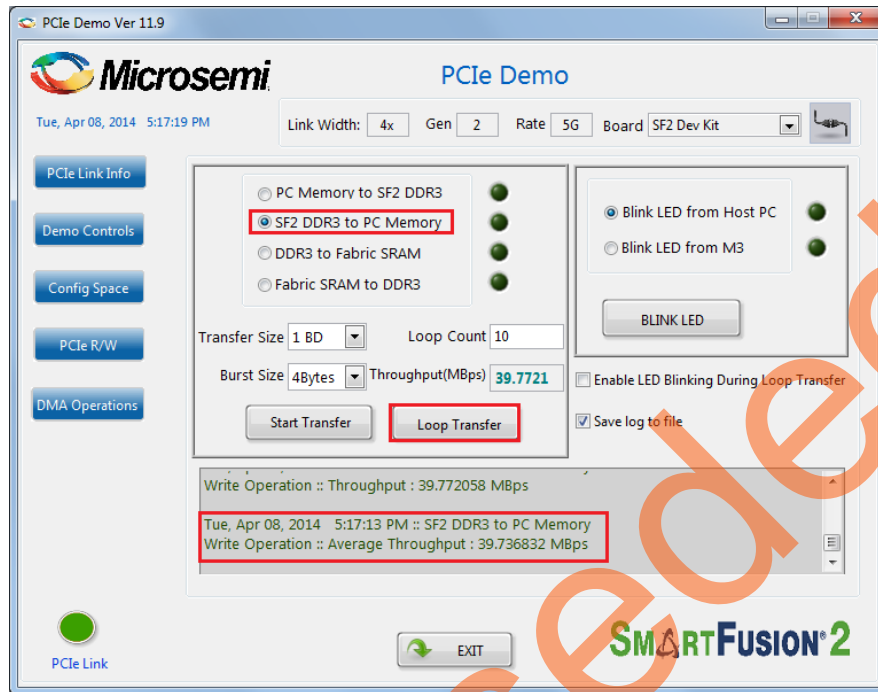


Figure 20 • DDR3 to PC Memory Transfer Throughput

10. Select the type of DMA transfer as **DDR3 to Fabric SRAM**, select any **Transfer Size** and click **Loop Transfer** as shown in Figure 21.

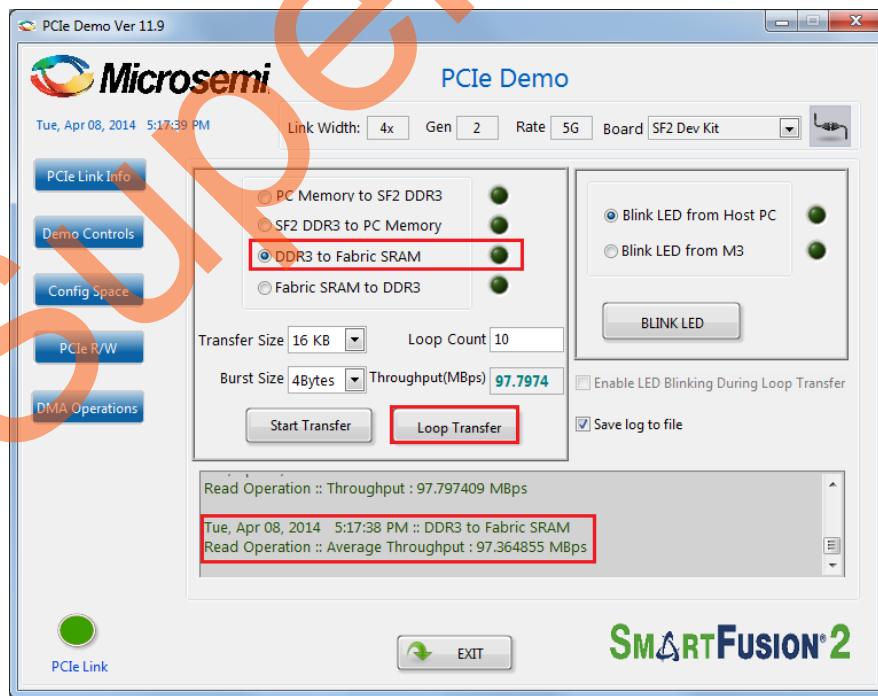


Figure 21 • DDR3 to Fabric SRAM Transfer Throughput

11. Select the type of DMA transfer as **Fabric SRAM to DDR3**, select any **Transfer Size** and click **Loop Transfer** as shown in Figure 22.

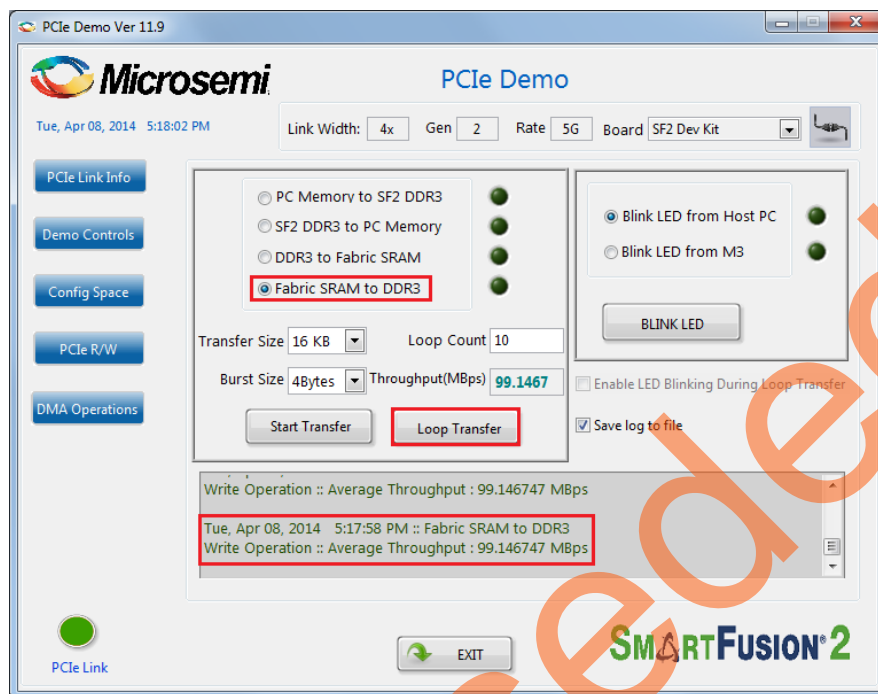


Figure 22 • Fabric SRAM to DDR3 Transfer Throughput

12. 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.
13. Click **Exit** to quit the demo.

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 DDR3	1 BD (64KB)/ 4 BD (256KB)/ 4x4 BD (1MB)	2.3	2.1	2.4	2.1
DDR3to Host PC Memory	1 BD (64KB)/ 4 BD (256KB)/ 4x4 BD (1MB)	25	25	39	39
DDR3 to Fabric SRAM	16KB/ 32KB/ 64KB/ 128KB	97	97	97	97
Fabric SRAM to DDR3	16KB/ 32KB/ 64KB/ 128KB	99	99	99	99

A – List of Changes

The following table lists the critical changes that were made in each revision of the chapter in the demo guide.

Date	Changes	Page
Revision 2 (April 2014)	Updated the document for Libero v11.3 software release (SAR 56582).	NA
Revision 1 (November 2013)	Updated the document for Libero v11.2 software release (SAR 52901).	NA
Revision 0 (September 2013)	Initial Release	NA

Superseded

B – 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.

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

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.

Technical Support

Visit the Customer Support website (www.microsemi.com/soc/support/search/default.aspx) for more information and support. Many answers available on the searchable web resource include diagrams, illustrations, and links to other resources on the website.

Website

You can browse a variety of technical and non-technical information on the SoC home page, at www.microsemi.com/soc.

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.

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.

My Cases

Microsemi SoC Products Group customers may submit and track technical cases online by going to [My Cases](#).

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. [Sales office listings](#) can be found at www.microsemi.com/soc/company/contact/default.aspx.

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_itar@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.

Superseded



Microsemi®

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

Microsemi Corporation (Nasdaq: MSCC) offers a comprehensive portfolio of semiconductor and system solutions for communications, defense and 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; security technologies and scalable anti-tamper products; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, Calif. and has approximately 3,400 employees globally. Learn more at www.microsemi.com.

© 2014 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.