LatticeECP2M PCI Express Development Kit User's Guide

Version 1.1

For use with the LatticeECP2M PCIe Solutions Board

Lattice Semiconductor Corporation
5555 NE Moore Court
Hillsboro, OR 97124
(503) 268-8000

September 4, 2008

Downloaded from Elcodis.com electronic components distributor

Copyright

Copyright © 2008 Lattice Semiconductor Corporation.

This document may not, in whole or part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form without prior written consent from Lattice Semiconductor Corporation.

Technical Support

Hotline (North America):1-800-LATTICE (North America)
Hotline (Outside North America**)**: +1-503-268-8001
e-mail: techsupport@latticesemi.com
Internet: www.latticesemi.com
Part Number: DK-PCIE-ECP2M-GUIDE

Trademarks

Lattice Semiconductor Corporation, L Lattice Semiconductor Corporation (logo), L (stylized), L (design), Lattice (design), LSC, E2CMOS, GAL, GDX, Generic Array Logic, ISP, ispATE, ispCLOCK, ispDOWNLOAD, ispGAL, ispGDS, ispGDX, ispGDXV, ispGDX2, ispGENERATOR, ispJTAG, ispLEVER, ispLEVERCORE, ispLSI, ispMACH, ispPAC, ispTRACY, ispTURBO, ispVIRTUAL MACHINE, ispVM, ispXP, ispXPGA, ispXPLD, LatticeEC, LatticeECP-DSP, LatticeECP2, LatticeXP, LatticeSC, MACH, MachXO, ORCA, PAC, PAC-Designer, PAL, Performance Analyst, Silicon Forest, Speedlocked, Speed Locking, SuperBIG, SuperCOOL, SuperFAST, SuperWIDE, sysCLOCK, sysCONFIG, sysDSP, sysHSI, sysI/O, sysMEM, LatticeMico32, The Simple Machine for Complex Design, TransFR, UltraMOS, XPIO, and specific product designations are either registered trademarks or trademarks of Lattice Semiconductor Corporation or its subsidiaries in the United States and/or other countries. ISP and Bringing the Best Together are service marks of Lattice Semiconductor Corporation.

Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

Disclaimers

NO WARRANTIES: THE INFORMATION PROVIDED IN THIS DOCUMENT IS “AS IS” WITHOUT ANY EXPRESS OR IMPLIED WARRANTY OF ANY KIND INCLUDING WARRANTIES OF ACCURACY, COMPLETENESS, MERCHANTABILITY, NONINFRINGEMENT OF INTELLECTUAL PROPERTY, OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT WILL LATTICE SEMICONDUCTOR CORPORATION (LSC) OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (WHETHER DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL, INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF THE USE OF OR INABILITY TO USE THE INFORMATION PROVIDED ON THIS SITE, EVEN IF LSC HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. BECAUSE SOME JURISDICTIONS PROHIBIT THE EXCLUSION OR LIMITATION OF CERTAIN LIABILITY, SOME OF THE ABOVE LIMITATIONS MAY NOT APPLY TO YOU.

LSC may make changes to these materials, specifications, or information, or to the products described herein, at any time without notice. LSC makes no

Downloaded from Elcodis.com electronic components distributor

commitment to update this documentation. LSC reserves the right to discontinue any product or service without notice and assumes no obligation to correct any errors contained herein or to advise any user of this document of any correction if such be made. LSC recommends its customers obtain the latest version of the relevant information to establish, before ordering, that the information being relied upon is current.

Type Conventions Used in This Document

Convention	Meaning or Use
Bold	Items in the user interface that you select or click. Text that you type into the user interface.
<*Italic*>	Variables in commands, code syntax, and path names.
Ctrl+F	The + means press the two keys at the same time.
`Courier`	Code examples. Messages, reports, and prompts from the software.
`...`	Omitted material in a line of code.
. . .	Omitted lines in code and report examples.
`[ ]`	In syntax descriptions, square brackets indicate optional items. In bus specifications, the brackets are required.
`( )`	In syntax descriptions, parentheses indicate grouped items.
|	In syntax descriptions, a vertical bar indicates a choice between items.

Downloaded from Elcodis.com electronic components distributor

Downloaded from Elcodis.com electronic components distributor

Contents

Downloaded from Elcodis.com electronic components distributor

Downloaded from Elcodis.com electronic components distributor

Downloaded from Elcodis.com electronic components distributor

Downloaded from Elcodis.com electronic components distributor

1

Introduction

Thank you for choosing the Lattice Semiconductor PCI Express (PCIe) Development Kit! This guide describes how to start using the PCIe Development Kit, a low-cost platform for demonstrating the PCI Express reference design for evaluating solutions for your own specific application.

Along with the LatticeECP2M-50 PCI Express solutions board and accessories, this kit includes all the software development tools and documentation that you need to begin developing your PCI Express solution.

About this Guide

This guide will familiarize you with the contents of the kit and walk you through the process of setting up your PCI Express development environment. The chapters are set up in a sequential manner and assumes you do not have any associated tools installed on your system. For Linux users, corresponding installation instructions are provided in Appendix A.

Using the Kit User Documentation

If you are new to ispLEVER and Lattice development kits, we recommend that you go through the chapters in this user's guide in a sequential manner and perform the tasks related to the platform you are using, either Windows or Linux.

If you are familiar with Lattice products or just want to get a quick feel of what this kit provides, use the *LatticeECP2M PCIe Demo Quick Start Guide* included with this kit to get started and use this guide to reference more detailed information. Please be aware that it is not necessary to install the ispLEVER software to set up your board and run the demos.

Downloaded from Elcodis.com electronic components distributor

Learning Objectives

After you complete the steps in this guide, you will learn how to do the following:

- Set up the solutions board properly and become familiar with its main features.
- Install all applicable development tools and the PCI Express demonstration applications.
- Establish communication with the solutions board through the PCIe link.
- Run the PCIe Basic demo, which allows you to run the preset LED light sequence, interactively light LED segments, and familiarize yourself with other features of the software.
- Run the PCIe Throughput demo, which allows you to see the performance of the Lattice PCI Express SERDES hardware and PCI Express IP core in terms of maximum data rates for writes/reads to and from your system memory.
- Run the DMA demos and observe how the Scatter-Gather DMA IP core, together with the PCIe IP core, demonstrate data transfer between the Lattice FPGA and system memory using the solutions board.
- Use what each demo teaches you about designing Lattice PCI Express solutions.
- Become familiar with an approach that will enable you to modify and rebuild the PCIe Basic demo for your own purposes.
- Become familiar with the software development tools and major design flow steps employed in this kit.
- Use other existing documentation in conjunction with this guide.

You can obtain more detailed information on specific board features by referring to the *EB33, LatticeECP2M PCI Express Evaluation Board User's Guide* document. See the “Related Documentation” section for instructions for obtaining all related documentation.

In addition to reading this guide, you should visit the LatticeECP2M PCI Express Development Kit web page on the Lattice Semiconductor Web site and familiarize yourself with the set of other documents related to PCIe and refer to other user guides and references that may provide more detail. See “Related Documentation” for details. This document assumes that you have already installed ispLEVER and are familiar with basics tasks in Project Navigator. If not, you must also refer to the ispLEVER Help system.

Downloaded from Elcodis.com electronic components distributor

Related Documentation

In addition to using this guide to help you get started developing the PCI Express solution on your device, you can refer to other documents applicable to this guide that may contain more detailed information that is beyond the scope of this guide. See the <kit_dir>\Documentation folder contents for most related documentation.

All of the following documents can be obtained on the Lattice web site at www.latticesemi.com or are contained within the kit installation:

- *LatticeECP2M PCIe Demo Quick Start Guide* – This guide, which is included in this kit, provides the basics of board setup and instructions for quickly running the demo applications without having to install the ispLEVER software. This guide is for Windows PC users.
- *EB33, LatticeECP2M PCI Express Solutions Evaluation Board User's Guide* – Describes board features, power requirements, device programming, clock management, and board schematics in detail. This document is available within the kit installation.
- *IPUG75, LatticeECP2M PCI Express 1.1 x1, x4 Endpoint IP Cores User's Guide* – Describes the features that the x1 and x4 Endpoint IP Cores support and provides a functional description of the IPs, parameters, signals, port lists, timing diagrams, memory maps, and step-by-step procedures for creating the core in IPexpress. This document is also available within the kit installation.
- *UG06, Lattice PCI Express x4 Scatter-Gather DMA Demo Verilog Source Code User's Guide* – Provides details of the Verilog code used for the Lattice PCI Express x4 Scatter-Gather DMA demo application.
- *UG07, Lattice PCI Express x4 SFIF Demo Verilog Source Code User's Guide* – Provides details of the Verilog code used for the Lattice PCI Express x4 SFIF Demo, referred to in this document as the Throughput demo.
- *UG15, Lattice PCI Express Basic Demo Verilog Source Code User's Guide* – Provides details of the Verilog code used for the Lattice PCI Express Basic demo, a block diagram of the design, and descriptions of design modules. Instructions for building the demo design in ispLEVER are also included.
- *Lattice PCI Express Software Development Resources* – An HTML page named PCIeDocIndex.html is included in this kit in the <kit_dir>\Software path. This page links to many other reference documents that provide information on the demo source code, device drivers, API library, and development environment.
- *PCI Express Development Kit for LatticeECP2M Web Page* – Visit this web page on the Lattice web site for updates to this and other related documents. You can also download kit installation files from this page as an alternative to using kit CD-ROMs. To access this web page, go to www.latticesemi.com and go to the **Products > DevKits and Hardware > Development Kits** hyperlink. Click on the **LatticeECP2M PCI Express Development Kit** hyperlink.

Downloaded from Elcodis.com electronic components distributor

- *LatticeECP2M PCI Express x4 Evaluation Board Web Page* – Links to the web page on the Lattice Semiconductor Web site for this part. On the Lattice web site, go to www.latticesemi.com/boards, click **Low-Cost FPGA Boards**, and then click **LatticeECP2M PCI Express x4 Evaluation Board**.
- *PCI Express x1 Endpoint - Optimized for LatticeECP2M Web Page* – Links to the web page on the Lattice Semiconductor Web site for this IP core. On the Lattice web site, go to the **Products > Intellectual Property > Lattice IP Cores > PCI Express x1 Endpoint** hyperlink. Be aware that the core for this design is not compatible with this kit.
- *PCI Express x4 Endpoint - Optimized for LatticeECP2M Web Page* – Links to the web page on the Lattice Semiconductor Web site for this IP core. On the Lattice web site, go to the **Products > Intellectual Property > Lattice IP Cores > PCI Express x4 Endpoint** hyperlink.
- *ispLEVER Installation Guides* – Links to the web page on the Lattice Semiconductor web site for downloading the Windows and Linux versions of this document which describes ispLEVER design software installation. On the Lattice web site, go to the **Documents > Installation Guides > ispLEVER** link.
- *ispDOWNLOAD™ Cables For Programming Lattice ISP Devices* – Describes all aspects of the download cable designed for use with this kit.

Note:
Go to the Documentation directory in the kit to access the PDF documents or access these references by going out to the Lattice web site at www.latticesemi.com. The ispLEVER Help system is available from within the software. The PDF file of this document that is included with this kit contains active hyperlinks to directly access these files.

Downloaded from Elcodis.com electronic components distributor

2

Getting Started with the Development Kit

This chapter describes all of the items that come with your Lattice PCI Express (PCIe) Development Kit, including the necessary software and all installation information for the PCIe Demo Software. We recommend that you familiarize yourself with the contents of this chapter before proceeding to board setup and running the software tools.

Before You Start

Review this section to understand what is included in this kit.

Development Kit Contents

The following items are included in your PCIe Development Kit. Please ensure that your kit contains these items listed below:

- LatticeECP2M-50 PCI Express Solutions Board (x1 and x4)
- ispDOWNLOAD^{TM} (USBN-2A) cable
- 12V DC wall adapter
- All ispLEVER 7.1 software CD-ROMs are included for Windows and Linux. This is a full-featured version with a 60-day evaluation license.
- Two kit CD-ROMs:
  - LatticeECP2M PCI Express Development Kit (Windows Version)
  - LatticeECP2M PCI Express Development Kit (Linux Version)

    These CD-ROMs include the hardware reference design, PCIe IP Demo software, IP cores, ispVM software, and any related kit documentation.

Downloaded from Elcodis.com electronic components distributor

If you are missing any of the above items, please contact technical support at at techsupport@latticesemi.com.

Development Tools

This section describes the development software contained in this kit and required licensing. The kit CD-ROM contains the following software and tools necessary to successfully use this kit:

- Lattice ispLEVER 7.1 or later design software (Windows and Linux)
- Lattice PCIe IP Demo Designs and Application Software (Windows and Linux)
- Lattice ispVM Download System Software (Windows and Linux)

The ispLEVER software is a comprehensive FPGA design environment to be used in conjunction with the PCI Express IP Demo. Using the ispLEVER software, you can develop hardware design files, synthesize your netlist, and output a bitstream file to configure your target FPGA. Use the ispLEVER design software to locate your I/O pins, apply timing constraints, and to perform timing analysis on your FPGA design. The ispVM software is bundled with ispLEVER.

Installing ispLEVER (Windows/Linux)

This kit includes the ispLEVER 7.1 FPGA design software. Separate installation CDs are included for both Windows and Linux. If you do not already have a licensed copy of ispLEVER 7.1, install the supplied 60-day evaluation version on your system.

Note:
For more information, see the ispLEVER installation notice for both Windows and Linux versions. You can access this documentation on the Lattice web site at www.latticesemi.com. Go to the **Documents > Installation Guides** navigation menu option.

To install ispLEVER for Windows:

1. Insert the ispLEVER software CD-ROM 1 or the DVD-ROM into the drive.
2. In the ispLEVER Setup window, click **Install ispLEVER Design Tools** and follow the rest of the wizard prompts to install into the desired directory location.

To install ispLEVER for Linux:

1. Insert the ispLEVER software CD-ROM 1 or DVD into the disk drive and mount the installation disk. If you are installing from a network, mount the drive by making a directory mount point and using the proper mount argument.
2. Specify a path location for installing the ispLEVER software, create a directory, and change to that directory, as in this example:

```
mkdir <path>/<install_path>
cd <path>/<install_path>
```

Downloaded from Elcodis.com electronic components distributor

3. Execute the **install.csh** script located on the disk with the path to the installation disk as its argument:

   ```
   <CD/DVD_Mount_Path>/install.csh <CD/DVD_Mount_Path>
   ```

4. Enter “**Y**” in the command line when prompted with the question, “Do you want to install all of ispLEVER Design Tools? ([Y]/N).” Make other selections based on your specific needs.

Licensing ispLEVER

You must have a licensed version of the ispLEVER FPGA design software (7.1 version or later) to work with this kit. The copy of the ispLEVER software included in this kit provides a 60-day evaluation license of the full production version of the software.

To license the kit’s ispLEVER software on the Lattice web site:

1. Go to www.latticesemi.com/license. The Software Licensing page opens.

   Note:
   If you are not logged in, you will by prompted to do so on the Lattice Account Sign In page. If you do not have a user account you will have to set one up by following the instructions. You may have to return to the Software Licensing page after online user registration is completed.

2. Under ispLEVER Design Kit, click **Request a license**.

3. In the **Software License Request Form**, provide the following information to complete licensing:

- Your full contact information, if you do not already have an account with this information.
- Development kit serial number (included on the SAVE ME card in the package)
- NIC/MAC address or server host ID. A NIC (Network Interface Card) or MAC (Media Access Controller) address is your machine’s physical address.

  On PC, enter **ipconfig -all** or **ipconfig/all** in the DOS command prompt. A machine profile appears that lists a Physical Address, a 12-digit hexadecimal number that should start with a zero, similar to 00-08-75-C6-59-01. If you get two physical addresses, do not use the one that begins with a 4. For Linux, use the **arp -a** command, look for your machine name in the ARP table listing, and your physical address will be shown in far right column.

4. After entering all required information in the form, click **Submit**.

You can also send your licensing information to the e-mail address *lic_admn@latticesemi.com* if for some reason you do not have access to this web site.

Downloaded from Elcodis.com electronic components distributor

Installing ispVM System Software

To ensure proper device configuration and support, you must upgrade your currently installed version of ispVM with the one supplied with the kit. These upgrades are also available for download on the web.

To install ispVM on Windows:

- Unzip the **ispvmsystem.zip** file on the Windows development kit CD and extract its contents into the *<isplever_install_dir>\ispvmsystem* directory to overwrite the present files.

To install ispVM on Linux:

- Unzip and untar the **ispvm_v17_2_1_linux.tar.gz** in the Linux development kit CD and extract its contents into the *<isplever_install_dir>* directory to overwrite the present files. Documentation for installation and environment variable setup is provided.

Downloading the Kit from the Web

You can download and install all of the necessary kit installation files and this document on the *PCI Express Development Kit for LatticeECP2M Web Page*. To access this web page, go to www.latticesemi.com/pciedevkit.

Device Configuration Software and Cable

The LatticeECP2M PCIe Solutions board is already pre-programmed with a hardware reference design that supports the PCIe Basic demo. There is no need to download anything to the device on the board to complete this demonstration but the other demos require downloading appropriate bitstream included in the kit.

When need to reconfigure the device to run a different demo, you will need to use the ispVM System software for download and the necessary download cable. See the section "Reprogramming the Solutions Board" on page 18 for details on how to reprogram the on-board device.

- You can use the Lattice ispVM software for downloading bitstreams for device configuration. This tool is integrated with the ispLEVER software. The latest version of ispVM System is always available for download from the Lattice web site at www.latticesemi.com/ispvm.
- ispDOWNLOAD^{TM} cable is used for connection to and programming of the device. See the ispDownload Cable for PCs web page on the Lattice web site for more information and to obtain the ispDOWNLOAD Cables data sheet that is available online. On the Lattice web site, go to the **Products > Dev Kits and Hardware > Programming Cables** hyperlink.

Downloaded from Elcodis.com electronic components distributor

Hardware Requirements

To install the kit design and run the demo software, a single computer with a PCIe x16, x8, x4, or x1 slot is required. You must also have a powered USB port. All of the other hardware and drivers are included in the kit.

Software Requirements

Please be aware of the following software requirements to ensure you obtain the expected results for the procedures described in this guide:

- For Windows, the Lattice PCI Express IP demo is compatible with Microsoft Windows 2000, Windows XP Professional, or Windows Server 2003 32-bit platforms.
- For Linux, the supplied files were built and tested for Red Hat Workstation 4.0, Update 4 on a 32-bit machine; however, instructions in Appendix A are provided for building necessary drivers for any kernel version you may be running. Refer to Appendix A, "Installing and Running the Development Kit in Linux" on page 57 for all Linux install instructions.
- The Verilog HDL demo design projects in this kit are built with ispLEVER 7.1. If you do not have an existing licensed copy of this version already installed on your machine, you can install the supplied ispLEVER 7.1 60-day evaluation software.
- If you are using ispVM device configuration software for your downloads, ensure you are using the latest version to ensure proper bitstream downloading results. The kit includes ispVM on the Windows and Linux kit CD-ROMs. You can also obtain this software on the Lattice web site at www.latticesemi.com/ispvm.
- To develop PCIe designs, your computer must meet minimum system requirements as described in the ispLEVER Installation Notice. You can obtain a copy on the Lattice web site at www.latticesemi.com. Choose **Documents > Installation Guides** in the main navigation bar.

Installing the PCIe Development Kit

This section provides Windows installation instructions for the PCIe development kit files and demo applications. See Appendix A for corresponding Linux kit installation instructions.

To install the development kit on Windows:

1. Load the kit CD-ROM in your CD drive and navigate to it using Windows Explorer.
2. On the kit CD-ROM, double click the **setup.exe** file.
3. If the **Install Program as Other User** dialog appears, choose to install as the current user if you have Administrator privileges or select another user with those privileges and click **OK**.
4. Click **Next** to start the installation. You must have administrative privileges to install the kit.

Downloaded from Elcodis.com electronic components distributor

5. Click **Yes** to accept the license agreement.
6. Click **Next** to install the kit in the default *C:\Lattice_DevKits* location on your hard drive or install in the desired location by using the **Browse** button.
7. If desired, click **Yes** to accept prompt to create shortcut desktop icons for the kit demo applications.
8. Click **Finish** to complete the kit installation. Figure 1 on page 10 shows the directory structure of the installed PCIe development kit.

Figure 1: Installed PCIe Development Kit Directory Structure (Windows)

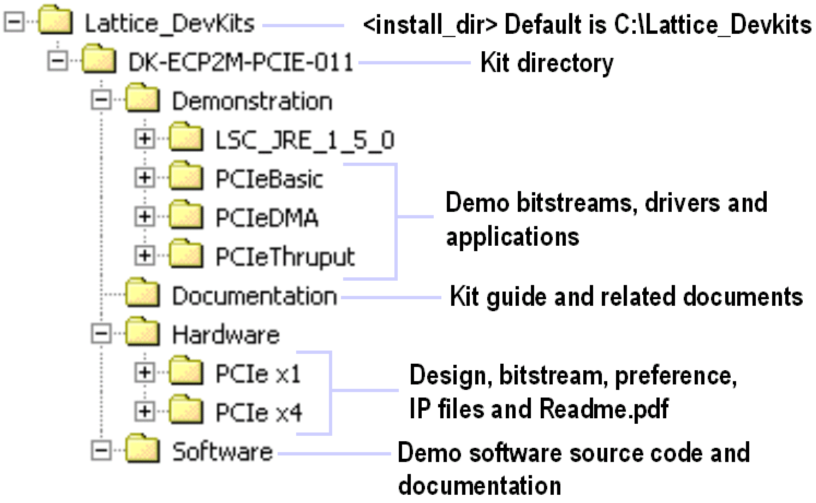

Lattice_DevKits
<install_dir> Default is C:\Lattice_Devkits
DK-ECP2M-PCIE-011
Kit directory
Demonstration
LSC_JRE_1_5_0
PCIeBasic
PCIeDMA
PCIeThruput
Demo bitstreams, drivers and applications
Documentation
Kit guide and related documents
Hardware
PCIe x1
PCIe x4
Design, bitstream, preference, IP files and Readme.pdf
Software
Demo software source code and documentation

Figure 1 above shows the default installation path for Windows. Please note that whenever the kit directory or <*kit_dir*> is referred to in this document, it refers to the <*install_dir*>*\DK-ECP2M-PCIE-011* file path, where the default <*install_dir*> path is *C:\Lattice_Devkits*.

The demo software source code is provided in the <*kit_dir*>*\Software* directory in the various demo subdirectories. Readme text and HTML files are available for demos and IP cores in the Demonstration demo subdirectories in the applicable Hardware subdirectories. Applicable bitstream files for each demo design are located in the Bitstream folder in the directory path <*kit_dir*>*\Demonstration\PCIe*<*demo_name*>, for example, the PCIeBasic folder is the first of three demo folders that contain bitstream configuration and demo application files.

After you install the development kit, on PC, you will also see Start Menu shortcuts for running demo applications if you opted to create them during installation. Please go to the following sections for further sequential instructions.

Downloaded from Elcodis.com electronic components distributor

3

PCIe Solutions Board Setup and Installation

This part of this document describes what you need to know to get started using the solutions board for the PCI Express (PCIe) Development Kit.

The PCIe Solutions Board

The LatticeECP2M-50 PCIe solutions board can be used as a basis for prototyping complex PCI Express solutions.

The PCIe solutions board is equipped with a Lattice ECP2M-50 FPGA device. The board is already pre-configured with the PCIe Basic demo reference design that is also available on the CD-ROM that comes with this kit.

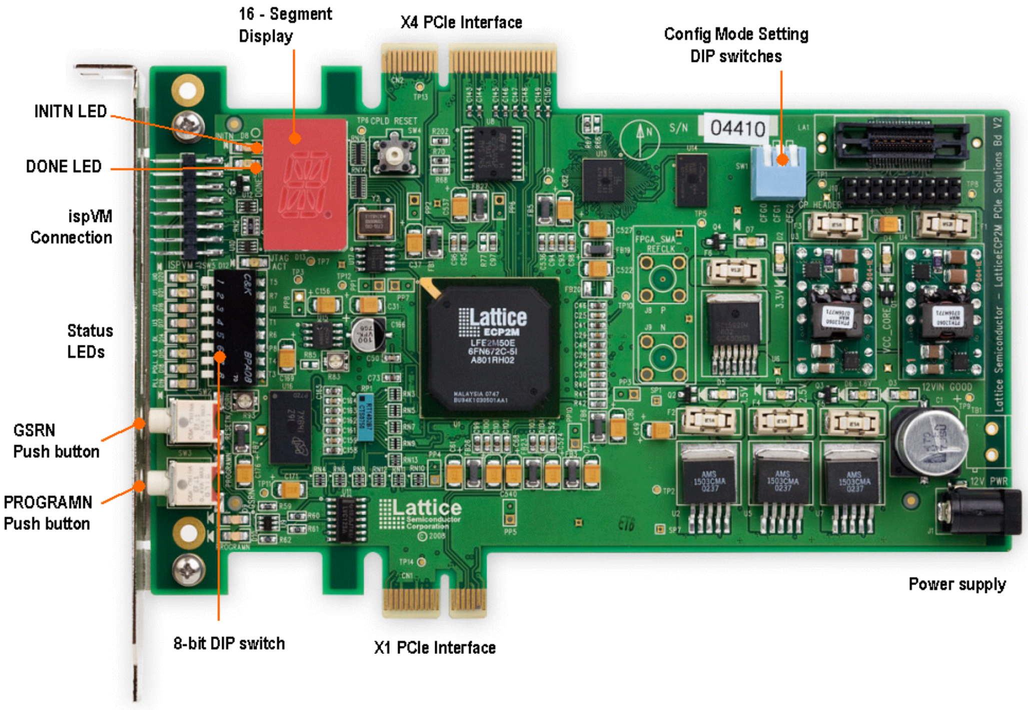
An enhanced form-factor of the PCI Express add-in card specification, the board allows for full x1 form-factor compliance. The x4 is available for demonstration purposes but with some non-standard form-factor issues. To demonstrate both x1 and x4 configurations you can simply change the mounting hardware. The board has several debugging and analysis features. See the document *EB33, LatticeECP2M PCI Express Evaluation Board User's Guide* for more details on board specifications and features outside the scope of this kit document.

Figure 2, "Interface Layout of the PCIe Solutions Board" on page 12 shows you the various LED indicators and other on-board interfaces that you will encounter during an evaluation of the solutions board. This diagram is given to simply provide a quick overview of these features.

To learn more about the details of the board design and features, refer to the *EB33, LatticeECP2M PCI Express Evaluation Board User's Guide*. The scope of that guide goes into much more detail about the hardware components and their features than the scope of this document.

Downloaded from Elcodis.com electronic components distributor

Figure 2: Interface Layout of the PCIe Solutions Board



Unpacking and Inspecting the Solutions Board

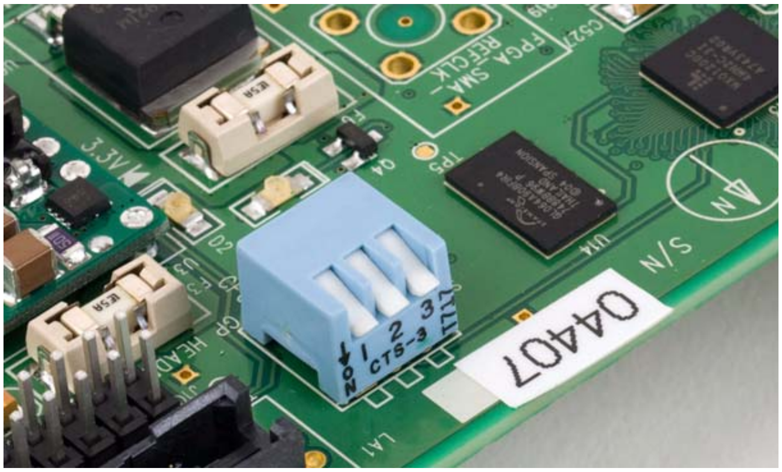
This section describes the proper steps to take to unpack the board to visually inspect its layout. There is very little setup necessary for this kit.

To unpack and inspect the Lattice PCIe solutions board:

1. Remove the PCIe solutions board and accessories from its antistatic packaging and compare the kit’s contents to the packaging list. Ensure that you avoid situations that would create electrostatic discharge (ESD) during setup or use of the board.
2. Position the board to view the x1 mode side as shown in the Figure 2 diagram and familiarize yourself with its basic layout and features. This document uses the x1 mode in procedures. You can test in the x4 mode but procedures will no longer apply in most cases.
3. Ensure that all configuration mode setting SW1 dip switches are pushed down toward the board as shown in Figure 3, "Configuration Mode Setting Dip Switches in “On” Position" on page 13.
4. Follow the procedures in the section, “Hardware Installation” which begin on page 13. You must have a PC with a PCIe slot available to proceed.

Downloaded from Elcodis.com electronic components distributor

Figure 3: Configuration Mode Setting Dip Switches in "On" Position



Hardware Installation

The procedures in this section provide step-by-step instructions for installing hardware and drivers to ensure proper board and PC communication and operation. These procedures are meant to supplement the section "Unpacking and Inspecting the Solutions Board" on page 12.

Installing the Solutions Board Hardware

After board setup, you can install the hardware. You must have administrative privileges on Windows to perform this installation. See Appendix A for Linux corresponding instructions.

Caution:
Lattice is not liable for any loss of data or damages that may result from the installation of the hardware and execution of the kit demo software tools. Do not install and operate on mission-critical systems.

To install the Lattice PCI Express Solutions Board for Windows:

1. Shut down Windows, turn off the PC and unplug the power cord.

   IMPORTANT: This step is necessary because PC power supplies have standby voltages that are present even when the PC power light and fan are turned off. Unplugging the PC is the safest way to ensure the board will not be "hot-swapped".

2. Locate an available PCI Express slot. The board can be installed in any slot that is larger than the finger edges in use, x1, x4, or x16.

3. Ensure that the solutions board is *not* connected to any external power supply before proceeding.

4. Using ESD precautions, install the LatticeECP2M PCI Express Solutions Board in the PCI Express slot in the x1 position.

Downloaded from Elcodis.com electronic components distributor

Note:
This guide's procedures do not describe the x4 implementation on the other side of the LatticeECP2M solutions board; however, you can use this in a similar fashion.

5. Power-on the PC and observe that it boots normally to the Windows login screen. If anything abnormal occurs, refer to Appendix B, "Troubleshooting" on page 63.
6. Log in as a user with administrative privileges. During the login process Windows will detect the new hardware and ask if you want to install it.

Downloaded from Elcodis.com electronic components distributor

Installing Solutions Board Drivers

This section describes installation of the solutions board device driver software on a Windows PC. This procedure pertains specifically to the PCIe Basic demo application. For the PCIe Throughput and DMA demos, you will need to load the appropriate drivers from the respective demo folders.

Also note that this procedure describes the installation on the Windows XP platform only. This may vary slightly on Windows 2000 or Windows Server 2003. See Appendix A for corresponding Linux instructions.

Note:
The **Found New Hardware** popup dialog in Windows appears when the PC is first booted with the solutions board installed. If this screen does not appear, the solutions board was not properly detected by the PC BIOS or by Windows. Refer to Appendix B, "Troubleshooting" on page 63 for more information.

To install the solutions board drivers on Windows:

1. In the Found New Hardware popup dialog, choose the **Install from specific location** option and click **Next**.
2. Use the **Browse** button to navigate to where you installed the demo package, locate the *Demonstration\PCIeBasic* directory path on the top level of the kit directory, and select the *Driver* folder.

   Note:
   For the PCIe Throughput and DMA demos used in this kit, you will have to install the appropriate drivers located in the similar directory path but in the *PCIe Thruput* and *PCIeDMA* folders*, respectively*.
3. Click **Next**. Windows now copies the driver files and will display a screen indicating this. Upon completion, a capital "I" representing initialization will be displayed on the 16-segment solutions board LED.

To verify proper driver installation and device recognition on Windows:

1. Right click on the **My Computer** icon on your Windows desktop and choose **Properties** from the popup menu. Confirm the installation and verify that Windows properly detects the solutions board hardware.
2. In the System Properties dialog, choose the **Hardware** tab and click the **Device Manager** button. The Lattice Evaluation Board (LSC_PCI Express) should be in the list of hardware devices in your system.
3. Right click on the **Lattice Eval Board** icon and select **Properties** to show the resources assigned to the device and the driver information.

   Memory ranges corresponding to the configured BAR registers will be assigned to the board. If this is all present, then the demo program is able to run and access the hardware on the solutions board.

Downloaded from Elcodis.com electronic components distributor

Installing Hardware into a Different Slot

Windows identifies PCI/PCI Express hardware devices using the bus, slot, vendor ID, and device ID fields. If you install the board into a different slot, the slot number will change. This will cause Windows to display the **Found New Hardware** popup screen when the system powers up.

The full procedure described in the previous section “Installing Solutions Board Drivers” to install the driver is unnecessary since the driver has already been installed. If the board is installed in a new slot, simply choose to allow Windows to search for the driver or the **Install the software automatically (Recommended)** option and install automatically. Windows will then associate the newly created device registry tag (bus, slot, vendor and device ID) with the *lscpcie.sys* driver and the demo GUI will work with the board in the new slot.

Now that your board is set up and hardware is installed on your computer, you can proceed on to the next chapter which describes software installation, execution, and tasks to complete the demo. The remaining sections in this chapter supplement what is described in the board setup procedure.

Verifying Correct Board Operation

The section lists checks you should make to ensure proper functioning of the board. Also refer to related documentation on this board described in the document *EB33, LatticeECP2M PCI Express Solutions Evaluation Board User’s Guide*. For part references, see Figure 2, "Interface Layout of the PCIe Solutions Board" on page 12.

There are four status LED lights on the board that will go through a light sequence when the device is first powered on. To verify the PCIe link is functioning properly, examine these indicators at the time of powering up. The PCIe demonstration software used later in the kit verifies board operation. In addition, you can also check that the Status LED lights are functioning at normal conditions in the sections below.

Note:
All boards leave the manufacturer fully tested. See the document *EB33, LatticeECP2M PCI Express Solutions Evaluation Board User’s Guide* for details.

LED Definitions

The Status LEDs on the LatticeECP2M PCI Express Solutions Board are located vertically along the left edge, middle portion of the board. Refer to the diagram shown in Figure 2, "Interface Layout of the PCIe Solutions Board" on page 12.

The LEDs are in the following order and have the following functions shown in the tables below.

Downloaded from Elcodis.com electronic components distributor

Table 1: LED Order and Functionality

LED Name	LED Number	Color	Usage
U0	D20 (x1) D28 (x4)	Blue	User LED. Off by default.
U1	D21(x1) D29 (x4)	Blue	User LED. Off by default.
U2	D16 (x1) D25 (x4)	Red	User LED. Off by default.
U3	D17 (x1) D24 (x4)	Red	User LED. Off by default.
DL_UP	D14 (x1) D22 (x4)	Green	Data Link up, ready for packets at Transaction Layer (PCI enumeration of config registers).
L0	D15 (x1) D23 (x4)	Green	L0 training sequences completed, PHY Layer up and ready for flow control.
POLL	D18 (x1) D26 (x4)	Yellow	Polling.
PLL	D19 (x1) D27 (x4)	Yellow	PLL locked to PCI Express 100MHz clock.
16-Segment Display	On board	Red	16-seg will display states with letter codes as described below: I – Initialization; driver is loaded and initializes the board O – Open; demo program is run and driver opens access to board. C – Close; demo exits and driver closes access to board. R – Remove; driver reads that board is removed from system, as in uninstalled device. E = Error; driver detects board hardware error during initialization.
16-Segment Display Decimal Point	On display in lower right	Red	Blinks to indicate PCIe transaction is accessing registers or memory in demo IP. Indicates PCIe bus transactions.

Downloaded from Elcodis.com electronic components distributor

Table 2: Normal x4 and x1 Link Status

PLL	POLL	L0	DL_UP
YLW	YLW	GRN	GRN

Default Switch Settings

The following table shows the default switch settings on the board.

Switch	Default Setting
SW1	All ON, (in towards circuit board). The FPGA CFG pins are set on the board for a particular programming mode via the SW1 DIP switch for the x1 board configuration.

Reprogramming the Solutions Board

The PCIe solutions board comes pre-programmed with the PCIe Basic demo bitstream. Other demos included with this kit require that you reprogram the device with the appropriate bitstream for that particular demo.

That is, each demo, including x1 and x4 variations, has bitstreams that contain different PCI Subsystem IDs that uniquely identify the demo and associates it with a particular device driver and demo application. So, if using the x1 side of the board, ensure that you use the bitstream (or XCF file) with the x1 identifier in the file name.

Important:
Only connect or disconnect the ispVM download cable from the solutions board when the power is off. Damage to the devices in the JTAG scan can occur if the board is turned on. This means that if you have a board installed in the PC running ispVM, you must shut down the PC before connecting the cable to the board. It can remain connected thereafter.

To reprogram the solutions board with new bitstream:

1. After ensuring that ispVM has been installed and configured, attach the programming cable to the board with the power off.
2. After the cable has been attached, power on the board.
3. From the Windows Start menu, click **Start > Programs > Lattice Semiconductor 7.1 > Accessories > ispVM System** to open ispVM.
4. Click **File > Open** and navigate to the appropriate XCF file. For the PCIe Basic demo, go to the *<install_dir>\Demonstration\PCIeBasic\Bitstreams* path to the file **PCIeSBx1_ECP2M50_PCIeBasic.xcf***.*

   Note:
   Ignore any messages you may encounter about the file being modified. The date and time of the file may have been changed during installation and no longer matches the XCF file's internal date.

Downloaded from Elcodis.com electronic components distributor

5. In the ispVM main window, double click the appropriate XCF file.

   Observe the selected options in the Device Information dialog. The Device Access Option should be set to **SPI Flash Programming**. Verify that the **File Name** shows the appropriate x1 configuration, (i.e., as opposed to x4) and select **Edit Device** and change the .bit file if necessary. Go to **ispVM > Options > Cable and I/O Port Setup > Cable Type** and ensure you use the **USB** cable type.

6. Click the **Go** button to initiate your download. You must wait while the flash chip on the board is programmed.

   Check the Chain Configuration status to see if the download received a **Pass** status. A processing status bar appears to show you the progress of the download. Upon completion, bitstream is programmed into SPI flash memory.

7. Push in the PROGRAM button on the board to program your device from SPI flash memory.

8. After programming the device, shut down and then reboot your PC. You must do this so Windows recognizes the new PCI registers and loads the proper driver.

9. If this is the first time this demo bitstream has been run, Windows will identify the board as new hardware and the driver for that demo needs to be loaded. The corresponding driver is in the Driver directory of each demo. See “Installing Solutions Board Drivers” on page 15 for instructions.

   Note:
   If an error about starting the device occurs during driver installation, use the Device Manager to disable and then re-enable the board. Or, you can reboot your PC.

Downloaded from Elcodis.com electronic components distributor

Downloaded from Elcodis.com electronic components distributor

4

Running the PCI Express Basic Demo

Once you have installed your PCIe solutions board in your computer and installed all necessary software, you can run the PCI Express (PCIe) Basic demo which consists of hardware, IP and software. This part of the document describes what you need to know to get started and successfully complete this demo.

Before You Begin

Before beginning this demo, you must do the following operations.

- Use ispVM to download the bitstream for this demo to the solutions board.

  You can find the x1 and x4 bitstreams and the XCF file necessary for ispVM in the *<kit_dir>\Demonstration\PCIeBasic\Bitstreams* directory path.

  For general information ispDOWNLOAD cable on ispVM configuration software usage, see the following:

  - Download procedure described in the section, “Reprogramming the Solutions Board” on page 18.
  - General introduction to device configuration with ispDOWNLOAD cable and ispVM software described in section, “Device Configuration Software and Cable” on page 8.
  - ispVM Help system

- Install the board drivers for the application.

  You can find the driver files necessary for proper demo installation in the *<kit_dir>\Demonstration\PCIeBasic\Driver* directory path. See the procedure in section, “Installing Solutions Board Drivers” on page 15 for guidance.

Downloaded from Elcodis.com electronic components distributor

Resource References

Please be aware of the specific companion documentation for supplementing your knowledge when using this demo:

Hardware Resources

The PCIe Basic Demo x1 bitstream is built from the ispLEVER project located in Hardware\PCIex1\ecp2m50_PCIeBASIC_SBx1\Implementation\ ecp2m50_PCIeBasic_SBx1 directory. The Verilog source code is located in the project Source\ directory.

The Verilog design architecture is explained in Documentation\ECP2M PCI Express Basic Reference Design User Guide\UG15.pdf. This document describes the purpose and functionality of the Verilog modules used in PCIe Basic Demo design.

Software Resources

The PCIe Basic Demo uses the lscpcie2.sys device driver. The source code for this device driver is located in Software\lscpcie2_Win2kXP\drvr. The architecture of the lscpcie2 device driver is explained in the “lscpcie2 Driver Reference Manual” which can be accessed through the Software\ PCIeDocIndex.html link.

The PCIe Basic Demo application source code is located in Software\PCIeBasic_Win2kXP\BasicGUI\DemoUI. This directory contains the Java project source code to create the user interface. The GUI also uses the PCIeAPI_Lib_Win2kXP API library.

The architecture of the PCIe Basic Demo application is explained in the “PCIe Basic Demo Reference Manual” and “PCIe API Reference Manual” which can be accessed through the Software\ PCIeDocIndex.html link.

Basic Demo Operations Overview

The PCIe Basic demo shows the capabilities of the Lattice FPGA and the PCI Express IP core functioning in a PCI Express slot in a Windows PC. The demo is easy to use and requires no test equipment.

This demo software allows you to access memory and registers on the board and provides real-time interaction with the solutions board hardware to demonstrate a functional PCI Express communications path between the application and driver software (running on the PC CPU) and the FPGA IP. Device driver and application source code are available so you can modify and extend the behavior of the tests or use them as a starting point for new PCIe designs.

If you experience any problems running this demo, please refer to Appendix B, “Troubleshooting” on page 63.

Downloaded from Elcodis.com electronic components distributor

Running the PCIe Basic Demo Software

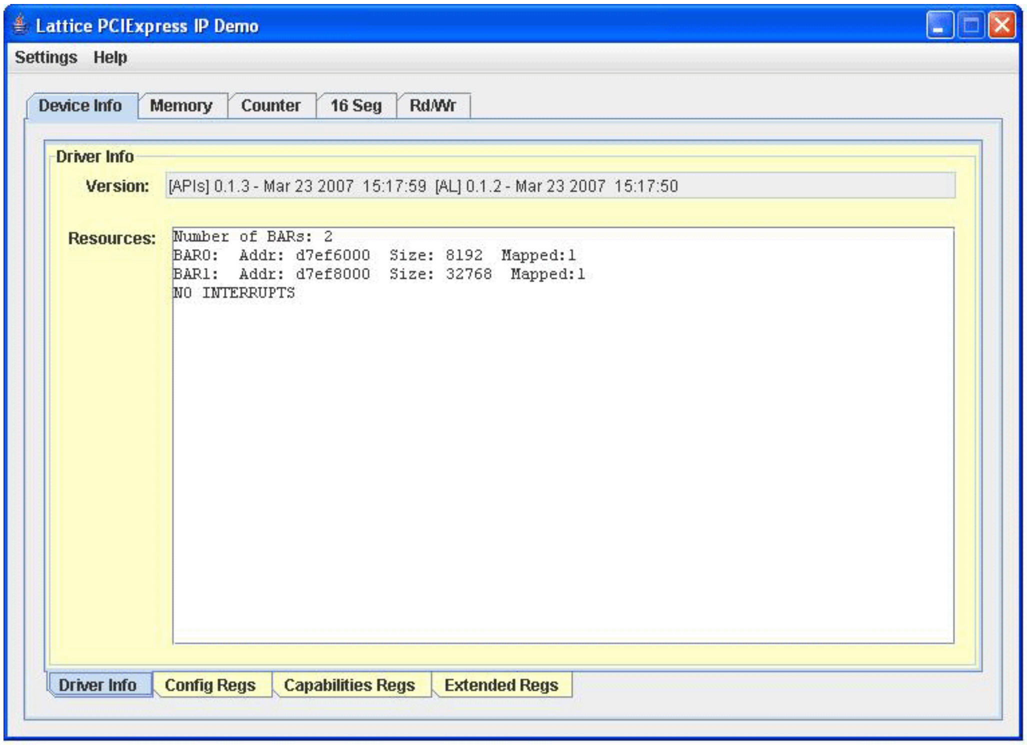
This section describes how to run the PCI Demo software after installation. You can access the PCIe Basic demo software from the Windows Start Menu.

To run the PCIe Basic demo software from your PC:

- From the Windows desktop, choose **Start > Programs > LatticeECP2M PCI Express Development Kit > PCIe Basic Demo**.

  The graphical user interface opens the PCIe Basic demo software with the Device Info tab activated as shown in Figure 4 below.

Figure 4: PCIe Demo Device Info Page



Driver Info

Version: [APIs] 0.1.3 - Mar 23 2007 15:17:59 [AL] 0.1.2 - Mar 23 2007 15:17:50

Resources:

```
Number of BARs: 2
BAR0:  Addr: d7ef6000  Size: 8192  Mapped:1
BAR1:  Addr: d7ef8000  Size: 32768  Mapped:1
NO INTERRUPTS
```

The Device Info page displays information about the device driver and the device's PCI configuration registers. The data displayed is for informational purposes only and cannot be edited. Descriptions of all of the information you can view in this page are available in the beginning of the next section "Touring the PCIe Basic Demo Interface".

Downloaded from Elcodis.com electronic components distributor

Touring the PCIe Basic Demo Interface

After opening the PCIe Basic demo software, this section will describe the pages and features in the interface.

1. In the Device Info page, click on the **Device Info sub tabs** and observe the structure of the information that is displayed in each. The following table describes what information is available for viewing by clicking each of the sub tabs that you can select and are located at the bottom of the page dialog.

Table 3: Device Info Page Sub Tab Descriptions

Sub Tab Page	Information Description
Driver Info	Obtained from the Lattice solutions board PCI Config space registers by the Lattice PCI Express driver when the demo is started. Displays Windows resources assigned to the device driver to access the solutions board.
Config Regs	Displays the standard PCI Config type 0 registers with each field annotated. Displaying this page causes the driver to issue PCI Config Type 0 read requests and re-displays the register values.
Capabilities Regs	Displays the PCI Express capabilities structures that are found in the register range 0x40 to 0xff. The applicable bit-fields of registers are parsed and displayed in readable format.
Extended Regs	Displays PCI Express extended configuration registers which are not used in this demo. These are inaccessible through the PC.

2. Now click on the 16 Seg tab to see the contents of the **16 Segment Control page**. In this page, you will be running a demonstration LED sequence and controlling the display on your board from this console. See Figure 5, "PCIe Basic Demo 16 Segment Control Page" on page 25.

   The 16-Segment Control page provides a way to interactively light segments on the display. You can preset character sequences from this page or select single characters and run them to light the display.

   The states of the LED segments are converted to a 16-bit word value (each segment is controlled by a bit) and written to the LED control register in the GPIO portion of the IP in the FPGA. This demonstrates a memory write across the PCI Express bus.

3. In the 16 Segment Control page, click the **Run** button. Notice on the board how the sequence of LED lights run in a certain pattern on the display.

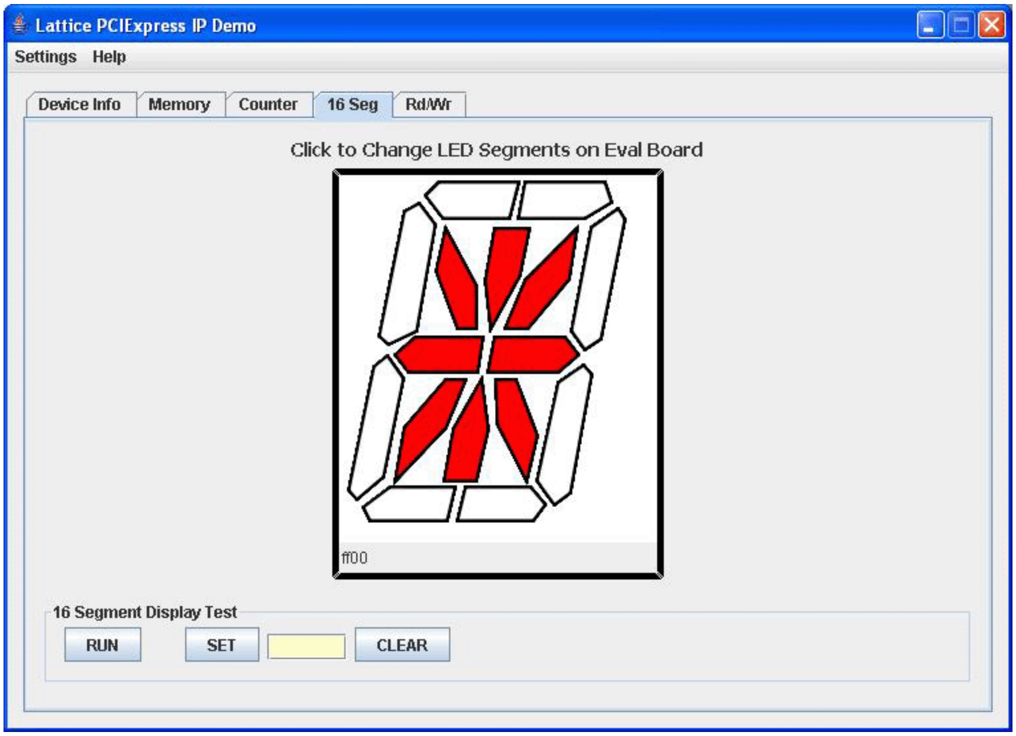
   The 16-segment display has two test modes. In the first mode demonstrated here, a pre-set sequence of segments are lit and characters are written to the display.

Downloaded from Elcodis.com electronic components distributor

This LED sequence run takes approximately 30 seconds to complete. You must observe the 16-segment LEDs to see if it is operating correctly. The correct sequence is:

- Light all segments, one at a time, around the perimeter.
- Light all inner segments in a clock-wise order.
- Turn off all inner segments in reverse order.
- Turn off all outer segments in reverse order.
- Write the characters “LATTICE*” one at a time to the display.
- The “*” will be displayed when the test ends.

Figure 5: PCIe Basic Demo 16 Segment Control Page



See the Table 4 below for details about features on the 16 Segment Control page.

Table 4: 16 Segment Control Page Features

Feature	Description
LED Display	Allows you to interactively change the LED display using mouse clicks to toggle segments on and off.
RUN	Starts an LED light sequence or command operation.

Downloaded from Elcodis.com electronic components distributor

Table 4: 16 Segment Control Page Features

Feature	Description
SET	Sets a user-defined LED light command operation based on input characters in the text box.
CLEAR	Turns off all segments in the display.

4. Now click on any segment in the interactive segment display in 16 Segment Control page. Notice that any selection will immediately cause the corresponding segment on the LED to light on your solutions board's LED display.

   Clicking on a segment will turn it on or off (toggles). The 16-bit value written to the LED register in the FPGA is shown on the bottom left.

5. Now type any character in the text box and click the **Set** button. The character will be configured in the display.

   This second mode of operation allows a single character to be sent to the display. Any printable ASCII character can be displayed (lower case is displayed as upper case). You cannot write a blank character using SET.

6. Click the **Clear** button. This turns off all segments of the LED display. Right-clicking on the background area behind the segments will clear the entire display.

   The interactive 16 Segment Control page demonstration you just performed illustrates that a functional PCI Express communications path exists between the application and driver software that is running on the CPU and the FPGA IP. Continue on to learn more about the other pages in the PCIe Basic demo software and their features.

7. Next, click on the Memory tab to open the **Memory page**. The Memory tab has various memory access tests that can be run to show that the IP is accessible from host software via the PCI Express bus. Refer to graphic in Figure 6.

   The page contains text boxes for entering data to be sent to device registers in the FPGA design. These text boxes are color coded to indicate the data format they accept. See the Table 5 below for details about these codes.

Table 5: Memory Page Text Box Color Codes

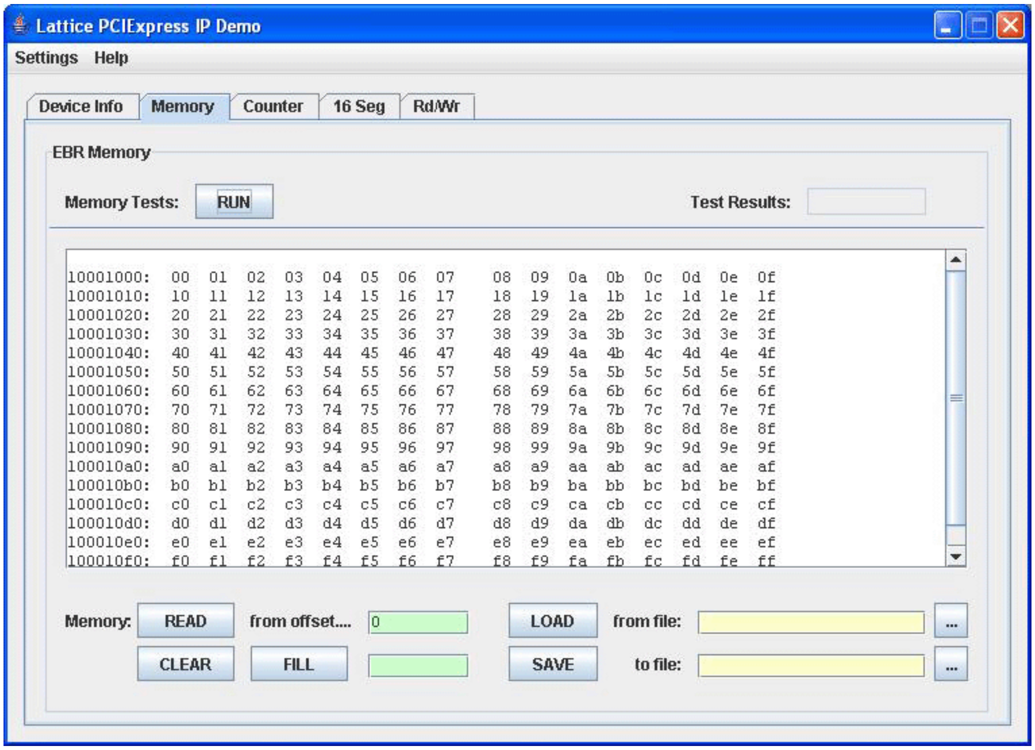
Color Code	Description
Green	Indicates hex value fields. Do not include any prefixes (0x) or suffixes (H). Only digits are allowable.
Yellow	Indicates character string fields, e.g., ones containing file names, paths, or letter values.
Blue	Indicates decimal (base 10) value fields.

The Memory Page features allow you to test the access to the 16KB of EBR internal to the FPGA. Accesses are done on a byte basis. All 16KB memory locations are accessed successively, testing the PCIe link to the

Downloaded from Elcodis.com electronic components distributor

memory interface. See Table 6, "Memory Page Features" on page 27 to see what actions can be performed in this page.

Figure 6: PCIe Basic Demo Memory Page



```
10001000:  00  01  02  03  04  05  06  07   08  09  0a  0b  0c  0d  0e  0f
10001010:  10  11  12  13  14  15  16  17   18  19  1a  1b  1c  1d  1e  1f
10001020:  20  21  22  23  24  25  26  27   28  29  2a  2b  2c  2d  2e  2f
10001030:  30  31  32  33  34  35  36  37   38  39  3a  3b  3c  3d  3e  3f
10001040:  40  41  42  43  44  45  46  47   48  49  4a  4b  4c  4d  4e  4f
10001050:  50  51  52  53  54  55  56  57   58  59  5a  5b  5c  5d  5e  5f
10001060:  60  61  62  63  64  65  66  67   68  69  6a  6b  6c  6d  6e  6f
10001070:  70  71  72  73  74  75  76  77   78  79  7a  7b  7c  7d  7e  7f
10001080:  80  81  82  83  84  85  86  87   88  89  8a  8b  8c  8d  8e  8f
10001090:  90  91  92  93  94  95  96  97   98  99  9a  9b  9c  9d  9e  9f
100010a0:  a0  a1  a2  a3  a4  a5  a6  a7   a8  a9  aa  ab  ac  ad  ae  af
100010b0:  b0  b1  b2  b3  b4  b5  b6  b7   b8  b9  ba  bb  bc  bd  be  bf
100010c0:  c0  c1  c2  c3  c4  c5  c6  c7   c8  c9  ca  cb  cc  cd  ce  cf
100010d0:  d0  d1  d2  d3  d4  d5  d6  d7   d8  d9  da  db  dc  dd  de  df
100010e0:  e0  e1  e2  e3  e4  e5  e6  e7   e8  e9  ea  eb  ec  ed  ee  ef
100010f0:  f0  f1  f2  f3  f4  f5  f6  f7   f8  f9  fa  fb  fc  fd  fe  ff
```

from offset....: 0

The following table provides descriptions of all of the Memory page features.

Table 6: Memory Page Features

Feature	Description
Pattern Tests	Pressing Run starts a test to check that all locations of the EBR can be read and written and that the contents are correct. First, all 16KB are cleared to 0 and verified. Then various patterns (AA, 55, 01, FF) are written to all locations and verified. If everything passes, PASS is displayed. If a memory location has an incorrect value the test aborts and displays ERRORS! The memory contents are left with an incrementing pattern 00 01 02... that is displayed when the test successfully finishes.
READ	The contents of the EBR memory are read from the value entered in the offset field. 256 bytes are read and displayed in the list box above.
CLEAR	Sets all 16KB to 0.
FILL	Writes the byte value entered in the field to all 16KB locations.

Downloaded from Elcodis.com electronic components distributor

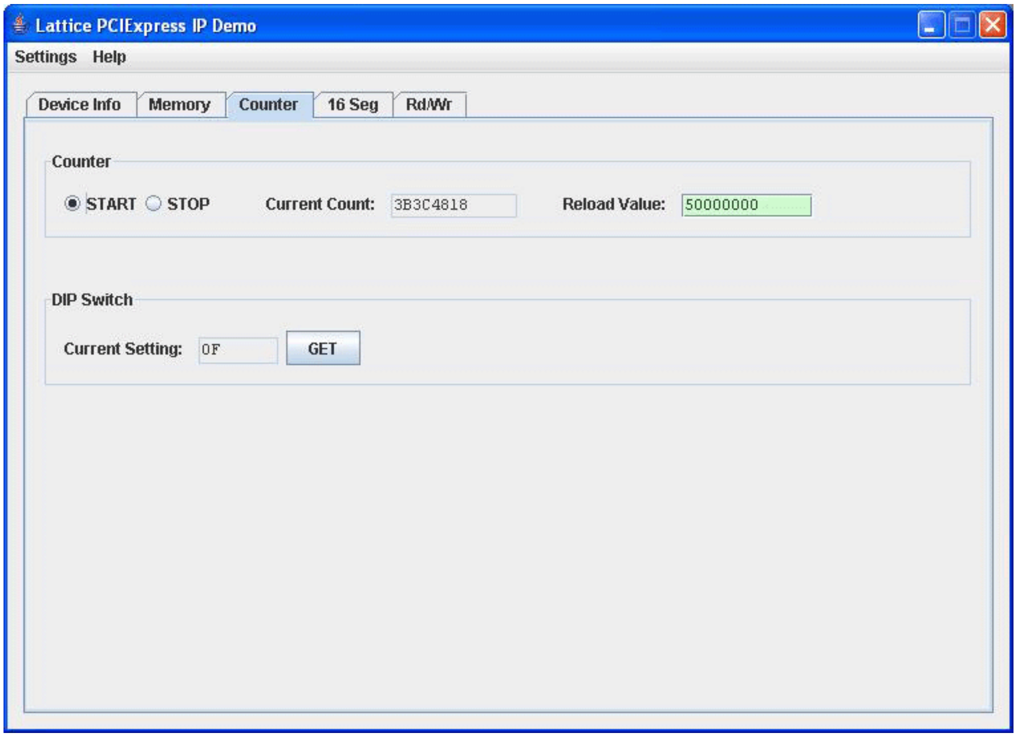
Table 6: Memory Page Features

Feature	Description
LOAD	Loads 16KB of binary data from the file specified (or as much data as is in the file) into EBR memory, starting at location 0. This can be used to load a known pattern into the EBR memory by using a file created by another tool.
SAVE	Writes all 16KB of EBR memory to the file specified. This can be used to save the contents of EBR memory for off-line processing (i.e., to verify that the pattern loaded in with LOAD is correctly saved in the EBR).

8. Next click on the Counter tab to open the **Counter page**. The Counter page allows you to control a 32-bit down counter in the FPGA hardware. The page is illustrated in Figure 7 below. Table 7, "Counter Page Features" on page 29 provides descriptions of the page's features.

   The counter is driven by the 125 MHz clock that feeds the IP. The counter is started by selecting the Start radio button. Counting begins from the value entered into the Reload Value field. The current count value is displayed in the Current Count field.

Figure 7: PCIe Basic Demo Counter Page



Lattice PCIExpress IP Demo

Counter

● START ○ STOP Current Count: 3B3C4818 Reload Value: 50000000

DIP Switch

Current Setting: 0F

Downloaded from Elcodis.com electronic components distributor

The following table provides descriptions of all of the Counter page features.

Table 7: Counter Page Features

Feature	Description
START/STOP	Starts and stops the 32-bit down counter in the FPGA hardware.
Current Count	Displays the current count value.
Reload Value	Sets the number from which counting begins.
DIP Switch	The DIP switch section shows that user changes to the switches on the solutions board are seen by the application software on the PC. The GUI polls the DIP switch register 10 times per second and displays the value read from the 8-bit DIP switch register.
Get button	Used to immediately update the value. This is active if No Polling was selected from the Settings dropdown menu.

9. Finally, click on the Rd/Wr tab to open the **Read/Write page**. The Read/Write page is used for looking at and poking at registers and EBR memory values in the application IP. Refer to Figure 8, "PCIe Basic Demo Read/Write Page" on page 30.

   The Read/Writer is primarily used for debugging and diagnosing the application IP registers. The following table provides descriptions of all of the Read/Write page features.

Table 8: Read/Write Page Features

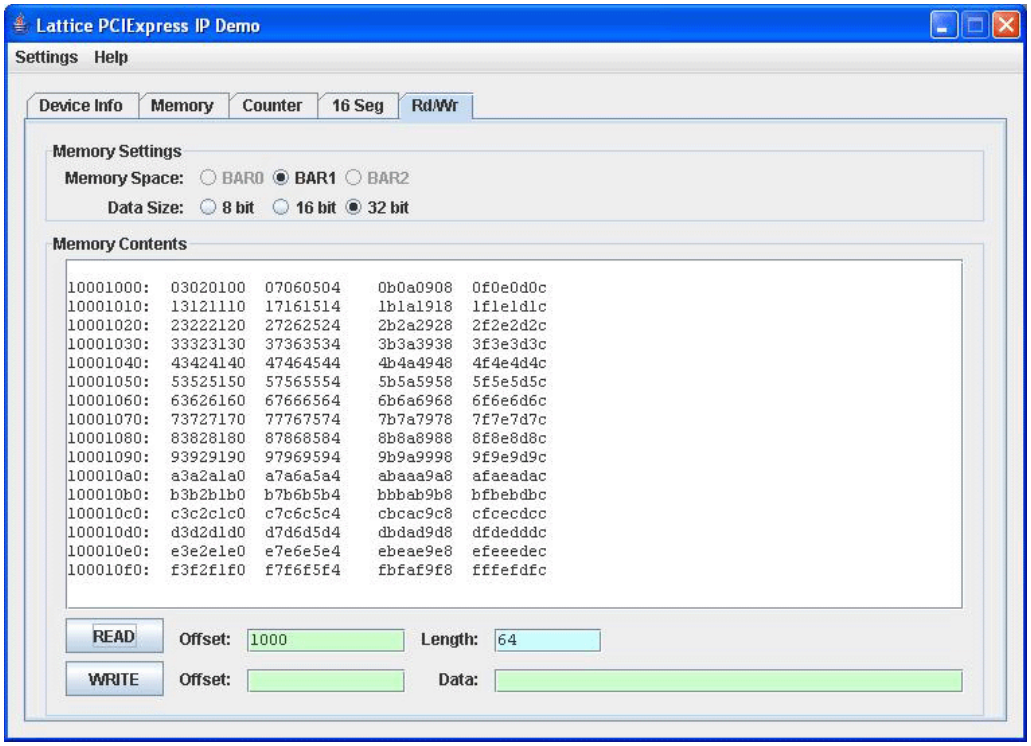
Feature	Description
Memory Space	Indicates the base address register (BAR) memory space to access.
Data Size	Indicates bit size. Options are 8-bit, 16-bit, and 32-bit.
Memory Contents	Displays memory contents.
READ	Starts a read data access based on offset and length settings.
WRITE	Starts a write data access based on offset and data settings.

Data accesses can be specified as byte, short or word operations by selecting the Data Size. Access is done to the selected Base Address Register (BAR). The memory contents are displayed in the window. In the address, the upper nibble (31:28) specifies the BAR being accessed. The following example shows reading the EBR memory (BAR 1, starting at offset 0x1000) in the application IP and displaying them in word format.

Data can be written to registers using the WRITE button. Specify the BAR Offset to start writing at and the hex data in the Data field. Separate each

Downloaded from Elcodis.com electronic components distributor

Figure 8: PCIe Basic Demo Read/Write Page



Memory Settings

Memory Space: ○ BAR0 ● BAR1 ○ BAR2

Data Size: ○ 8 bit ○ 16 bit ● 32 bit

Memory Contents

```
10001000:  03020100  07060504   0b0a0908  0f0e0d0c
10001010:  13121110  17161514   1b1a1918  1f1e1d1c
10001020:  23222120  27262524   2b2a2928  2f2e2d2c
10001030:  33323130  37363534   3b3a3938  3f3e3d3c
10001040:  43424140  47464544   4b4a4948  4f4e4d4c
10001050:  53525150  57565554   5b5a5958  5f5e5d5c
10001060:  63626160  67666564   6b6a6968  6f6e6d6c
10001070:  73727170  77767574   7b7a7978  7f7e7d7c
10001080:  83828180  87868584   8b8a8988  8f8e8d8c
10001090:  93929190  97969594   9b9a9998  9f9e9d9c
100010a0:  a3a2a1a0  a7a6a5a4   abaaa9a8  afaeadac
100010b0:  b3b2b1b0  b7b6b5b4   bbbab9b8  bfbebdbc
100010c0:  c3c2c1c0  c7c6c5c4   cbcac9c8  cfcecdcc
100010d0:  d3d2d1d0  d7d6d5d4   dbdad9d8  dfdedddc
100010e0:  e3e2e1e0  e7e6e5e4   ebeae9e8  efeeedec
100010f0:  f3f2f1f0  f7f6f5f4   fbfaf9f8  fffefdfc
```

Offset: 1000 Length: 64

Offset: Data:

value with a space. Data size should match the Data Size selected at the top of the page in Memory Settings.

Only BAR1 is accessible in the current demo. See the IP Register Memory Map section of *UG08 - Lattice PCI Express Demo User's Guide* for a list of valid device addresses. This document is available on the Lattice web site.

Rebuilding the PCIe Basic Demo Design

You can rebuild the PCIe Basic demo IP reference design by running the source HDL design files through a design flow in the ispLEVER software. All source HDL files and necessary project files are included in the kit installation. This document assumes that you have already installed ispLEVER and are familiar with basics tasks in Project Navigator. Refer to the figure, "Installed PCIe Development Kit Directory Structure (Windows)" on page 10 to understand where various files referenced in this section are located.

We recommend that you copy the files from the install location to a new working location. This allows you to quickly move back to the original configuration without re-installing this kit.

Downloaded from Elcodis.com electronic components distributor

Implementing the PCIe Basic Demo Design

The **top.syn** ispLEVER project file is included in this kit. This file contains information regarding options to use when implementing the Demo design. The **top.lpf** logical preference file specifies timing constraints and ECP2M I/O pin assignments with respect to the ECP2M Standard PCIe board. The working directory is the Implementation directory.

To implement the demo design using the EDIF/NGO flow:

1. Open ispLEVER Project Navigator.
2. Click **File > Open Project**.
3. In the Open Project dialog, navigate to and the select the **top.syn** file in the *<kit_dir>\Hardware\PCIe x1\ecp2m50_PCIeBasic_SBx1\Implementation\ecp2m50_PCIeBasic_SBx1* directory path.
4. Click **Open**. All of the HDL files are imported into the project.
5. In the Processes window, right click on **Build Database** and select Properties from the popup menu.
6. In the Properties dialog, verify that **Macro Search Path** is set to the directory path, **..\ispLeverGenCore\ecp2m\pciex4d1\** for Windows. Your path would have forward slashes for Linux.
7. In ispLEVER Project Navigator, verify that the device select is a LFE2M50E-6F672C.
8. In Project Navigator, double click the **Generate Bitstream Data** process.

Modifying the PCIe Basic Demo Design

This section provides a very simple alteration to the HDL to demonstrate a change in the behavior of the function of the LED light in the demo display. It will involve a small change in the HDL code in the source file.

To modify the PCIe Basic demo design:

1. Open the **top_basic.v** file with an ASCII editing tool or the internal ASCII editing tool in Project Navigator. This file is located in the *<kit_dir>\Hardware\Source\ecp2m* directory path, where *<kit_dir>* represents the path, *<install_dir>\DK-ECP2M-PCIE-011*.
2. On or about the line 390 as shown below, delete the **tilde** character (~) that appears before the (**led_out_int)** parameter.

   ```
   led_out        <= ~(led_out_i);
   ```

3. After making this small change, click **File > Save** and close the editor.

   This modification to the code will cause the 16-segment LED to operate in reverse, that is, all of the lights will be on when the demo starts instead of off.

4. Open the **top.syn** project file in ispLEVER's Project Navigator. This file is located in the *ecp2m50_PCIeBasic_SBx1* folder in the *Hardware\PCIe x1\ecp2m50_PCIeBasic_SBx1\Implementation\ecp2m50_PCIeBasic_SBx1* path.

Downloaded from Elcodis.com electronic components distributor

5. Double click the **Generate Bitstream Data** process in the Processes window to generate a **top.bit** file in the directory *<kit_dir>\Hardware\Implementation*.
6. Start ispVM and perform the steps described in the section "Reprogramming the Solutions Board" on page 18 to download the new bitstream to the board's SPI flash memory.
7. Push in the PROGRAM button on the board to program your device from SPI flash memory.
8. Reboot the PC so that the BIOS recognizes the new PCIe endpoint device configuration.
9. Verify that the status LEDs are correct and note that all the LED segments are now on.
10. Rerun the LED test described in the section, "Touring the PCIe Basic Demo Interface" on page 24. Notice that the state of the 16-segment LED on the board is the inverse of what is displayed in the GUI.

You have completed the Lattice PCIe Basic demo and have successfully completed all of the learning objectives of this kit.

Downloaded from Elcodis.com electronic components distributor

5

Running the PCI Express Throughput Demo

This chapter describes the Lattice PCI Express (PCIe) Throughput demo that you can run within this kit on a Windows system (Microsoft Windows2000, Windows XP SP2, Server2003).

Before You Begin

Before beginning this demo, you must do the following operations.

- Use ispVM to download the bitstream for this demo to the solutions board.

  You can find the x1 and x4 bitstreams and the XCF file necessary for ispVM in the *<kit_dir>\Demonstration\PCIeThruput\Bitstreams* directory path.

  For general information ispDOWNLOAD cable on ispVM configuration software usage, see the following:

  - Download procedure described in the section, “Reprogramming the Solutions Board” on page 18.
  - General introduction to device configuration with ispDOWNLOAD cable and ispVM software described in section, “Device Configuration Software and Cable” on page 8.
  - ispVM Help system

- Install the board drivers for the application.

  You can find the driver files necessary for proper demo installation in the *<kit_dir>\Demonstration\PCIeThruput\Driver* directory path. See the procedure in section, “Installing Solutions Board Drivers” on page 15 for guidance.

Downloaded from Elcodis.com electronic components distributor

Resource References

Please be aware of the specific companion documentation for supplementing your knowledge when using this demo:

Hardware Resources

The PCIe Throughput Demo x1 bitstream is built from the ispLEVER project located in *Hardware\PCIe x1\ecp2m50_PCIeThruput_SBx1\Implementation \ecp2m50_PCIeThruput_SBx1*. The Verilog source code is located in the project Source\ directory.

The Verilog design architecture is explained in Documentation\ ECP2M PCI Express Throughput Reference Design User Guide\UG07.pdf. This document describes the purpose and functionality of the Verilog modules used in PCIe Throughput Demo design.

Software Resources

The PCIe Throughput Demo uses the lscpcie2.sys device driver. The source code for this device driver is located in Software\lscpcie2_Win2kXP\drvr. The architecture of the lscpcie2 device driver is explained in the "lscpcie2 Driver Reference Manual" which can be accessed through the Software\ PCIeDocIndex.html link.

The PCIe Throughput Demo application source code is located in *Software\PCIeSFIF_Win2kXP\SFIF_GUI\SFIF_UI*. This directory contains the Java project source code to create the user interface. The GUI also uses the PCIeAPI_Lib_Win2kXP API library.

The architecture of the PCIe Throughput Demo application is explained in the "PCIe Thruput Demo Reference Manual" and "PCIe API Reference Manual" which can be accessed through the Software\ PCIeDocIndex.html link.

Throughput Demo Operations Overview

The purpose of this demo is to show the performance of the Lattice PCI Express SERDES hardware and PCI Express IP core when operating in a PC PCI Express expansion slot. The data rates for writes to the PC system memory and reads from the PC system memory are measured and displayed in a graphical user interface.

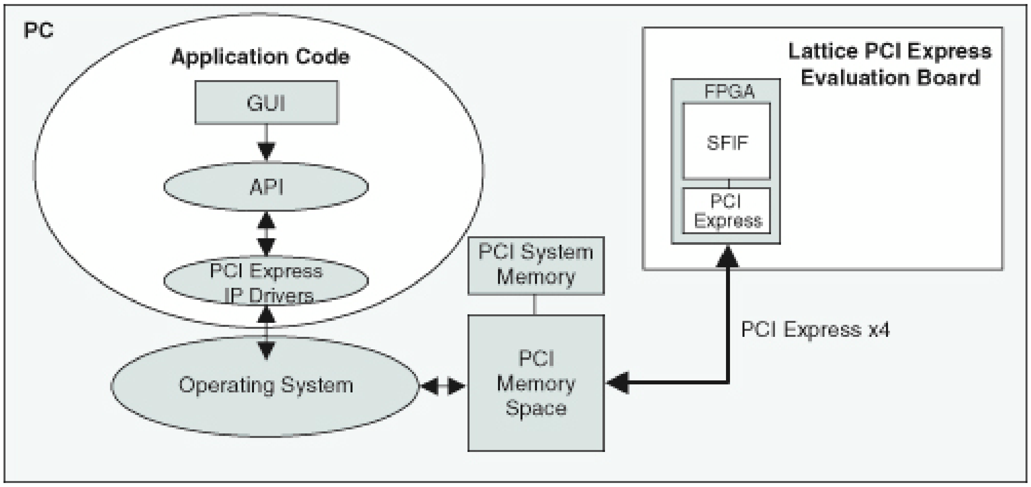
The demo performs Direct Memory Access (DMA) operations by transferring data directly to and from the PC memory. The demo uses an IP block named the SFIF (Stored FIFO InterFace) to generate read and write Transaction Layer Packets (TLPs) that will access the PC system memory. The SFIF exercises the PCI Express IP core and link with low overhead so the true performance of the PCI Express core and link can be measured.

The PCI Express interface is used for both control plane and data plane traffic. The control plane loads the SFIF memory and sets up the transfer. The

Downloaded from Elcodis.com electronic components distributor

data plane transfers the data from the SFIF to the PC memory. Figure 9 below shows the relationship of the hardware and software components of the demo. For more details on SFIF IP, register mapping and related topics, see the documents IPUG75 and UG07 referred to in the section, “Related Documentation” on page 3.

Figure 9: PCI Express Throughput Block Diagram



The Throughput demo software allows you to set up different types of data transfers to understand the PCI Express link. You can select the type of transfer to perform (e.g., write, read, and write/read) as well as how many bytes of data to transfer. You also have the option of selecting the size of the TLP in which to perform the transfer.

Note:
The PCIe Throughput demo design requires at least 16 posted credits to use 128-byte write TLPs. This requirement is to optimize the throughput of the PCIe link. You can determine the amount of posted credits for the given slot in the GUI. If the posted credits are less than 32, then 64-byte write TLPs are the largest size supported.

If you experience any problems running this demo, please refer to Appendix B, “Troubleshooting” on page 63.

Running the Throughput Demo

This section describes how to run the PCIe Throughput demo after installation. You can access the demo from the Windows Start Menu.

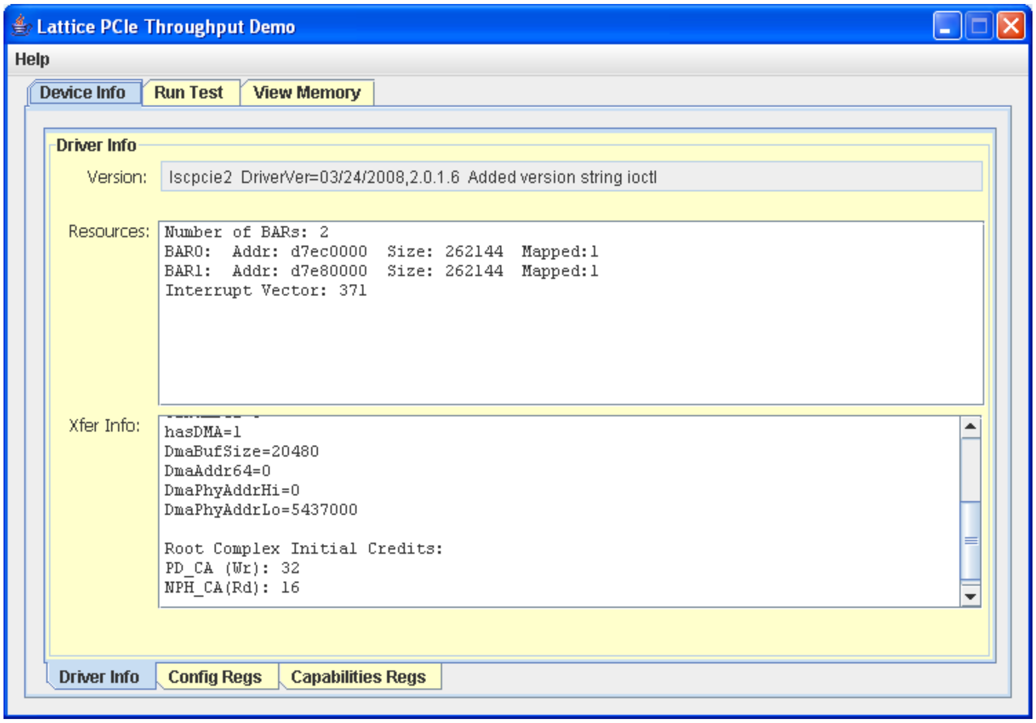
To run the PCIe Throughput demo from you PC:

- From the Windows desktop, choose **Start > Programs > LatticeECP2M PCI Express Development Kit > ECP2M Thruput**.

  The graphical user interface opens the PCIe Throughput demo software with the Device Info tab activated as shown in Figure 10 below.

Downloaded from Elcodis.com electronic components distributor

Figure 10: Throughput Demo Device Info Page



Version: lscpcie2 DriverVer=03/24/2008,2.0.1.6 Added version string ioctl

Resources:

```
Number of BARs: 2
BAR0:  Addr: d7ec0000  Size: 262144  Mapped:1
BAR1:  Addr: d7e80000  Size: 262144  Mapped:1
Interrupt Vector: 371
```

Xfer Info:

```
hasDMA=1
DmaBufSize=20480
DmaAddr64=0
DmaPhyAddrHi=0
DmaPhyAddrLo=5437000

Root Complex Initial Credits:
PD_CA (Wr): 32
NPH_CA(Rd): 16
```

The Device Info page displays information about the board's PCI config space registers, PCI Express capabilities, and root complex buffer sizes. Descriptions of all of the information you can view in this page are available in the beginning of the next section "Touring the PCIe Throughput Demo Interface".

Touring the PCIe Throughput Demo Interface

After opening the PCIe Throughput demo software, this section will describe the pages and features in the interface.

1. In the Device Info page, click on the **Device Info sub tabs** and observe the structure of the information that is displayed in each. The following table describes what information is available for viewing by clicking each

Downloaded from Elcodis.com electronic components distributor

of the sub tabs that you can select and are located at the bottom of the page dialog.

Table 9: Device Info Page Sub Tab Descriptions

Sub Tab Page	Information Description
Driver Info	Provides information about the device driver including the version, the resources used, and the transfer information. The demo design uses the lscpci2 driver. The demo requests two BARs (Base Address Registers) and a single interrupt vector. The Xfer Info box provides the buffer sizes for the root complex for Posted and Non-Posted TLPs. This information is important when considering the amount of credit waiting the demo design demonstrates when running a transfer. A root complex with larger buffers will provide better performance when running the demo since it will not have to release credits as quickly to allow the next TLP.
Config Regs	Provides the standard PCI Type0 space configuration register contents. Things such as Device ID and Vendor ID are displayed and the assigned BARs.
Capabilities Regs	Provides the link list of capability structures and their contents. Key information found in this box is the maximum TLP size supported by the root complex and the negotiated link width.

2. Now click on the Run Test tab to see the contents of the **Run Test page**. This page operates the demo design. In this page, you will be running the demonstration to compute the throughput of the PCI Express link and display the transfer rates with bar graphs. You can select read, write and read-write throughput tests. See Figure 11, "Throughput Run Test Info Page" on page 38.

3. On the Run Test page, under Setup options, select the following:

   Test Mode: Thruput

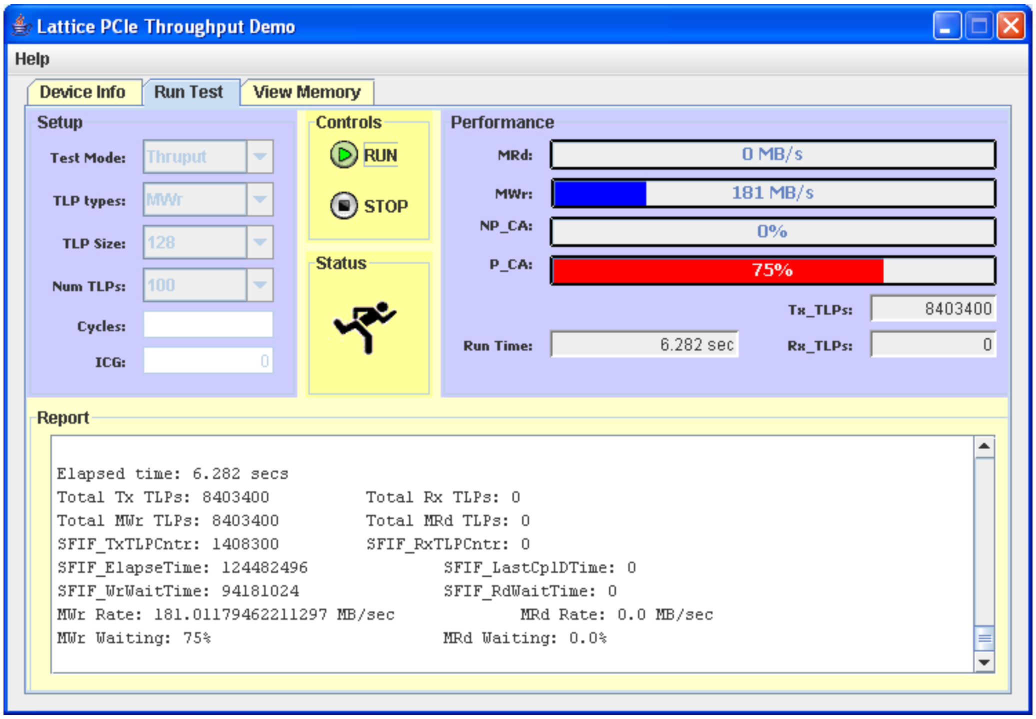
   TLP Type: MWr

   For the rest of the options, take the defaults.

4. On the Run Test page, click the **RUN** button. After running your test, notice the status indicators in the Performance section at right. The two top progress indicator bars for MRd (memory read TLPs), and MWr (memory write TLPs) will contain a percentage of blue which indicates throughput. The two progress indicator bars below that show the wait time for the root complex to accept TLPs over the entire time spent running.

Downloaded from Elcodis.com electronic components distributor

Figure 11: Throughput Run Test Info Page



Test Mode: Thruput

TLP types: MWr

TLP Size: 128

Num TLPs: 100

Cycles:

ICG: 0

MRd: 0 MB/s

MWr: 181 MB/s

NP_CA: 0%

P_CA: 75%

Tx_TLPs: 8403400

Run Time: 6.282 sec

Rx_TLPs: 0

```
Elapsed time: 6.282 secs
Total Tx TLPs: 8403400          Total Rx TLPs: 0
Total MWr TLPs: 8403400         Total MRd TLPs: 0
SFIF_TxTLPCntr: 1408300         SFIF_RxTLPCntr: 0
SFIF_ElapseTime: 124482496              SFIF_LastCplDTime: 0
SFIF_WrWaitTime: 94181024               SFIF_RdWaitTime: 0
MWr Rate: 181.01179462211297 MB/sec             MRd Rate: 0.0 MB/sec
MWr Waiting: 75%                        MRd Waiting: 0.0%
```

Downloaded from Elcodis.com electronic components distributor

See the Table 10, “Run Test Page Feature Descriptions” below for details about features on the 16 Segment Control page.

Table 10: Run Test Page Feature Descriptions

Feature	Information Description
Setup	Sets the configuration for the specific test.
Test Mode	There are two modes of operation, cycles and throughput. ◆ Throughput - This mode allows the test to run continuously looping through the tx_fifo and updating the performance numbers every second. This test will run until the you click the STOP button. ◆ Cycles - This mode allows you to set up a specific number of times the tx_fifo will be looped. Once complete, the test will stop automatically and the performance numbers will be displayed based on the entire run. The cycle consists of all MRd TLPs or all MWr TLPs. The purpose is to validate that the correct number of TLPs was sent/received using the counters and View Memory page. See TLP Types in this table for descriptions. The key difference between the two modes is how the performance data is displayed. A throughput test will provide new performance data every second. A cycle test will provide performance data after the number of cycles completes, or one second, whichever comes first. Note that cycles will not run longer than one second. Cycles tests run once; throughput tests run continuously until stopped.
TLP Types	There are four types of TLP types which impact the type of “traffic” sent over the PCI Express link. ◆ MWr – Memory Write TLPs to write data from the endpoint to the PC system memory. ◆ MRd – Memory Read TLPs to read data from PC system memory to the endpoint. ◆ MRd+MWr – Both Memory Read and Memory Write TLPs are sent to the root complex. ◆ R+W+Ctl – Read, Write, and Control data are present on the PCI Express link. The Read and Write TLPs are sent from the SFIF while the PC is also modifying the GPIO 16-segment display LEDs. This TLP type shows both data and control plane TLPs sharing the PCI Express link.

Downloaded from Elcodis.com electronic components distributor

Table 10: Run Test Page Feature Descriptions

Feature	Information Description
TLP Size	The TLP size controls allows you to select the size of the TLPs to be sent from the SFIF. The maximum size of the TLP will be dependent on the root complex. In MRd mode, the maximum TLP size is limited by Max Read Request size (512 bytes). In MWr mode, the maximum TLP size is limited by Max TLP Size (128 bytes). In Read/Write mode the following sizes are available for MRd TLP and MWr TLP combinations. ◆ 512,128 – 512-byte read requests with 128-byte write TLPs ◆ 256,128 – 256-byte read requests with 128-byte write TLPs ◆ 128,128 – 128-byte read requests with 128-byte write TLPs ◆ 64,64 – 64-byte read requests with 64-byte write TLP ◆ 32,32 – 32-byte read requests with 32-byte write TLPs ◆ 16,16 – 16-byte read requests with 16-byte write TLPs
Num TLPs	This control allows the user to select the ratio of read requests to write TLPs. ◆ 1Rd,1Wr – This ratio results in one Read Request and 1 Write TLP back-to-back. The completion data will need to be received before another Read Request can be made. ◆ 1Rd,4Wr – This ratio results in one Read Request and four Write TLPs. The completion data must be received before another Read Request can be made. This results in much greater bandwidth since read requests are four times the size of a write TLP. This ratio balances the PCI Express link, but still waits for read data. ◆ 4Rd,16Wr – This ratio results in four Read Requests and 16 Write TLPs. This ratio allows for four read requests to be outstanding (TAGs). This ratio is only recommended on server class motherboards due to the high bandwidth required. With four reads outstanding the root complex can better utilize the read data. ◆ 16Rd,64Wr – This ratio results in 16 Read Requests and 64 Write TLPs. This ratio allows for 16 read requests to be outstanding (TAGs). This ratio is only recommended on server class motherboards due to the high bandwidth required. With 16 read outstanding the root complex can better utilize the read data.

Downloaded from Elcodis.com electronic components distributor

Table 10: Run Test Page Feature Descriptions

Feature	Information Description
Cycles	This control is only available when the Cycle mode has been selected. This controls the number of times the tx_fifo is looped before ending the test. The software starts the SFIF and waits one second while the SFIF transfers data (number of cycles). After one second, the software stops the SFIF and displays the performance. This has the effect of limiting cycles tests to a maximum of one second of operation. The cycles value cannot be larger than 65535 (it is a 16-bit counter). In Throughput mode this control is not used. In Throughput mode the SFIF is looping the tx_fifo continuously until the user presses the STOP button.
ICG	(Inter Cycle Gap) This control sets the number of 125MHz clock cycles between cycles. You can use this control to model TLP traffic patterns that may be appropriate for your system. The ICG value cannot be larger than 65535 (it is a 16-bit counter).
Controls	Stops and starts test. Status shows an image of a running man to indicate test is in progress.
Performance	Displays the current data rates and other statistics. Data rates are displayed as progress bars, with the rate (MB/sec) displayed in the bar. The bars are updated every second when running in Throughput mode or upon completion in Cycle mode. The rates are computed from the hardware counters in the SFIF. ◆ Write rates are computed from the following SFIF hardware counters: Tx TLP Count and Elapsed Count. ◆ Write Rate (MB/sec) = (Tx TLP Count * TLP Size) / (Elapsed Count * 8ns) ◆ Read rates are computed from the following SFIF hardware counters: Rx TLP Count and CplD Timestamp. ◆ Read Rate (MB/sec) = (Rx TLP Count * RCB_Size) / (Elapsed Count * 8ns) ◆ RCB_Size is the size in bytes of a CplD. In Throughput mode performance is recalculated every second and the counters are reset. In Cycles mode the performance is calculated once at the end of the run displaying the results for the entire transfer.

Downloaded from Elcodis.com electronic components distributor

Table 10: Run Test Page Feature Descriptions

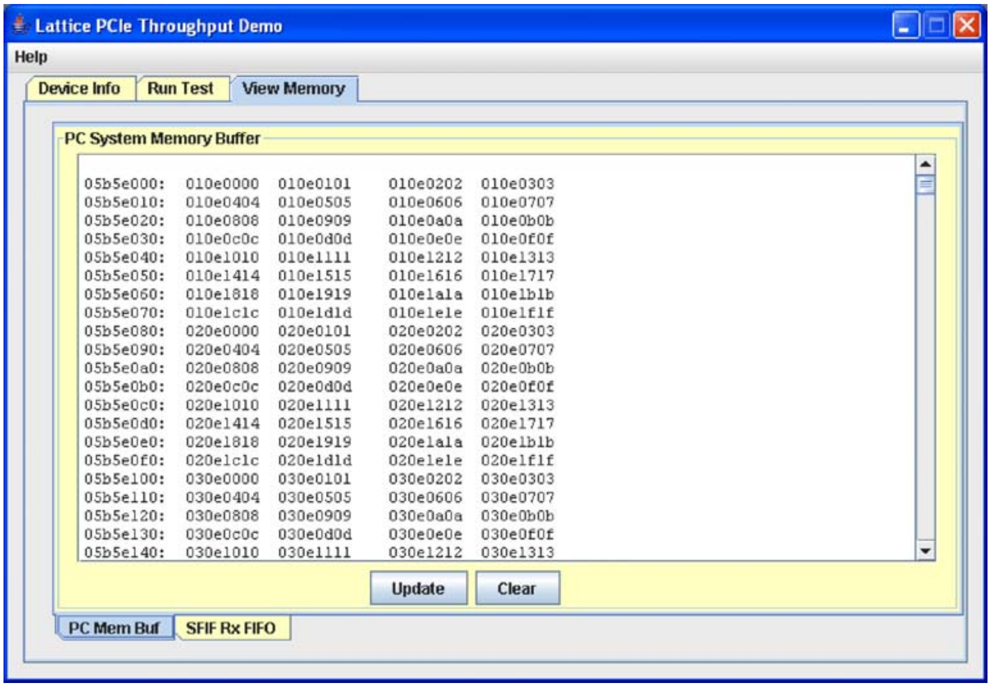
Feature	Information Description
NP_CA and P_CA	Time spent waiting for the root complex to accept TLPs is computed and displayed as a bar graph in percentage of time waiting over time spent running. Normal efficient operation should show a small percentage of time spent waiting for credits and more time spent sending TLPs. Counters record when the SFIF wants to send a MRd but the credit available ports of the PCI Express core indicates the root complex has not yet processed the read requests. The PCI Express core is waiting to accept an UpdateFC-NP freeing up Non-Posted credits to send another MRd TLP. Counters record when the SFIF wants to send a MWr but the credit available ports of the PCI Express core indicates the root complex has not yet processed the sent write TLPs. The PCI Express core is waiting to accept an UpdateFC-P freeing up Posted credits to send another MWr TLP.
Report	Logs all of the test information. The Report box provides details about the test. In the Throughput mode this report will be updated every second up to 10 seconds. After 10 seconds the data is no longer updated in the report box to prevent system load and excessive resource usage during long duration tests (over night). In the Cycles mode the report window is updated when the test is complete.

5. Next, click on the View Memory tab to open the **View Memory page**. Refer to graphic in Figure 12. Notice that this page allows you to inspect the memory contents of the PC’s system memory buffer and the SFIF rx_fifo to check for data integrity. See the descriptions of these sub tabs in Table 11.

   After inspecting this page, you can move on to the next chapter which describes the PCIe DMA demos.

Downloaded from Elcodis.com electronic components distributor

Figure 12: Throughput Demo View Memory Page



```
05b5e000:  010e0000  010e0101   010e0202  010e0303
05b5e010:  010e0404  010e0505   010e0606  010e0707
05b5e020:  010e0808  010e0909   010e0a0a  010e0b0b
05b5e030:  010e0c0c  010e0d0d   010e0e0e  010e0f0f
05b5e040:  010e1010  010e1111   010e1212  010e1313
05b5e050:  010e1414  010e1515   010e1616  010e1717
05b5e060:  010e1818  010e1919   010e1a1a  010e1b1b
05b5e070:  010e1c1c  010e1d1d   010e1e1e  010e1f1f
05b5e080:  020e0000  020e0101   020e0202  020e0303
05b5e090:  020e0404  020e0505   020e0606  020e0707
05b5e0a0:  020e0808  020e0909   020e0a0a  020e0b0b
05b5e0b0:  020e0c0c  020e0d0d   020e0e0e  020e0f0f
05b5e0c0:  020e1010  020e1111   020e1212  020e1313
05b5e0d0:  020e1414  020e1515   020e1616  020e1717
05b5e0e0:  020e1818  020e1919   020e1a1a  020e1b1b
05b5e0f0:  020e1c1c  020e1d1d   020e1e1e  020e1f1f
05b5e100:  030e0000  030e0101   030e0202  030e0303
05b5e110:  030e0404  030e0505   030e0606  030e0707
05b5e120:  030e0808  030e0909   030e0a0a  030e0b0b
05b5e130:  030e0c0c  030e0d0d   030e0e0e  030e0f0f
05b5e140:  030e1010  030e1111   030e1212  030e1313
```

See the Table 11 below for details about features on the View Memory page:

Table 11: View Memory Page Sub Tab Descriptions

Sub Tab Page	Information Description
PC Mem Buf	The PC system memory buffer sub tab allows you to inspect the contents of the PC's system memory buffer allocated in the kernel space by the driver and is used for the source of MRd requests and destination for MWr TLPs. This can also be used to verify that the MWr TLPs have worked and that the data was transferred from the PCI Express solutions board into system memory.
SFIF Rx FIFO	The SFIF Rx FIFO sub tab displays the parsed and formatted contents of the SFIF rx_fifo. This can be used to verify that a small burst of MRd TLPs have returned the proper data to the solutions board. The TLPs are parsed and time stamped.

Known Issues

Currently there is only one known issue that should be considered when exercising the Throughput demo. The Throughput reference design has a difficult time meeting timing constraints when using a -6 speed grade. If you rebuild the Throughput reference design, a -7 speed grade is recommended.

Downloaded from Elcodis.com electronic components distributor

Downloaded from Elcodis.com electronic components distributor

6

Running the PCI Express DMA Demos

This chapter describes the Lattice PCIe Direct Memory Access (DMA) demos that you can run within this kit on a Windows system (Microsoft Windows 2000, Windows XP SP2, Windows Server 2003).

The DMA demos illustrate Lattice PCIe and Scatter-Gather DMA (SGDMA) IP cores working together to transfering data over the PCIe bus. The SGDMA operates as a Master DMA, reading and writing data to PC system memory.

One demo illustrates moving large amounts of image data from the solutions board to PC system memory and display on the screen. The other demo implements a simple hardware image processor, in which pixel data from a source image on the screen is read by the board, modified by the hardware and written back and redisplayed. A test program is also provided that checks all the driver and IP functionality.

Before You Begin

Before beginning this demo, you must do the following operations.

- Use ispVM to download the bitstream for this demo to the solutions board.

  You can find the x1 and x4 bitstreams and the XCF file necessary for ispVM in the *<kit_dir>\Demonstration\PCIeDMA\Bitstreams* directory path.

  For general information ispDOWNLOAD cable on ispVM configuration software usage, see the following:

  - Download procedure described in the section, “Reprogramming the Solutions Board” on page 18.

Downloaded from Elcodis.com electronic components distributor

  - General introduction to device configuration with ispDOWNLOAD cable and ispVM software described in section, “Device Configuration Software and Cable” on page 8.
  - ispVM Help system

- Install the board drivers for the application.

  You can find the driver files necessary for proper demo installation in the *<kit_dir>\Demonstration\PCIeDMA\Driver* directory path. See the procedure in section, “Installing Solutions Board Drivers” on page 15 for guidance.

Resource References

Please be aware of the specific companion documentation for supplementing your knowledge when using these demos:

Hardware Resources

The PCIe DMA Demo x1 bitstream is built from the ispLEVER project located in Hardware\PCIe x1\ecp2m50_PCIeSGDMA_SBx1\Implementation\ ecp2m50_PCIeSGDMA_SBx1. The Verilog source code is located in the project Source\ directory.

The Verilog design architecture is explained in Documentation\ ECP2M PCI Express SG-DMA Reference Design User Guide\UG06.pdf. This document describes the purpose and functionality of the Verilog modules used in PCIe DMA Demo design.

Software Resources

The PCIe DMA Demo uses the lscdma.sys device driver. The source code for this device driver is located in Software\lscdma_Win2kXP\drvr. The architecture of the lscdma device driver is explained in the “lscdma Driver Reference Manual” which can be accessed through the Software\ PCIeDocIndex.html link.

The PCIe DMA Demo application source code is located in two directories corresponding to the two demos: Software\ PCIeDMA_Win2kXP\ColorBars and Software\PCIeDMA_Win2kXP\ImageMove. These applications use the OpenGL APIs for displaying the image data. The demos also use the PCIeAPI_Lib_Win2kXP API library.

The architecture of the PCIe DMA Demo application is explained in the “PCIe DMA Demo Reference Manual” and “PCIe API Reference Manual” which can be accessed through the Software\ PCIeDocIndex.html link.

Downloaded from Elcodis.com electronic components distributor

DMA Demo Operations Overview

Direct Memory Access (DMA) is a method of transferring data from one memory mapped device to another. The data is transferred by a dedicated device that performs the bus cycle (memory reads and writes). The CPU is not involved in the actual data movement.

Using a dedicated DMA device frees the CPU to do other operations and also shortens the transfer time. If the CPU had to move the data, it would be done in a software loop which requires fetching, decoding and executing each instruction involved in the loop. This could easily expand to 10 or more instruction cycles per datum moved. A DMA engine could perform the same datum move operation in 1 to 3 bus clocks (depending on bus architecture).

In modern PC systems the DMA engine, the device responsible for performing the bus cycles to implement the transfer, is located on the add-in card. This is known as Bus Master DMA and is the preferred method of operation. The PCI bus is being phased out and replaced with the PCIe bus. To take advantage of the high bandwidth that PCIe offers, DMA is used to transfer the data between the add-in card and the system memory. The Lattice SGDMAC IP works in conjunction with the Lattice PCIe IP core to transport the data.

The Scatter-Gather DMA (SGDMA) IP core, together with the PCIe IP core, demonstrates moving data between the Lattice FPGA and PC system memory using a Lattice PCI Express solutions board. The board uses the PCI Express link as both control (setup and operation of the core) and data path (DMA to/from PC system memory). The PC provides the test platform (power, run-time environment) and the user interface.

A PC platform is used because currently PCs are the only readily available, economical and standard platform utilizing PCIe. A Windows device driver provides the interface to the board's register and memory space. Application software uses the driver to setup and configure the DMA engine, execute it, and verify the results. The demo system is illustrated in the block diagram in Figure 13 on page 48.

The demo hardware has the following objectives:

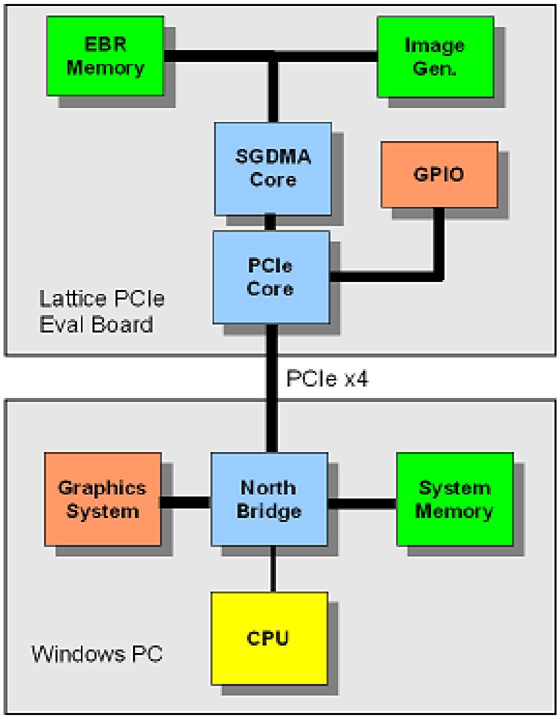
- Acts as a reference design for using the PCIe and SGDMA IP cores
- Performs actual DMA transfers over the PCIe bus at an optimal rate
- Provides counters and timers to measure performance
- Provides a platform for demonstration and experimentation

The demo application software has the following objectives:

- Demonstrates accessing, configuring and operating the PCIe and SGDMA IP cores
- Verifies proper operation (ensures all DMA data is transferred from source to destination un-corrupted)

Downloaded from Elcodis.com electronic components distributor

Figure 13: DMA Demo Block Diagram



- Demonstrate Windows driver and system programming so users can extend software for their own particular system needs
- System memory allocation (Memory Descriptor Lists)
- Interrupt handling - ISRs and DPCs

Scatter-Gather DMA Overview

Hardware devices perform Direct Memory Access (DMA) by initiating read/write bus transactions. DMA means transferring data to and from system memory directly, without involving the CPU.

Bus Master DMA means the device (the PCIe Core on the board) is controlling the bus and doing the data transfers. In order to perform the transfer, an address is needed and a length. The SGDMA is configured by the software driver. The addresses known to software for describing a buffer's location in memory are only relevant in the domain of the CPU. The CPU (and software) view memory as virtual 2 GB address spaces per process - the DMA needs physical memory addresses.

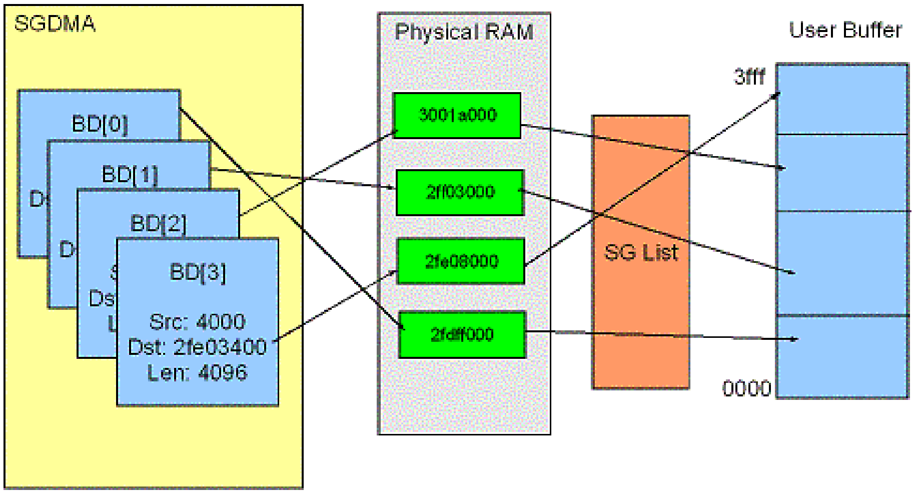
When software allocates a large buffer of memory, the memory manager finds the number of required free pages (4KB per page) in system memory and makes them appear contiguous to software via virtual memory translation tables in hardware. A 1MB buffer alocated by user software appears contiguous to the software, but in reality is scattered throughout physical system memory in discontinuous 4KB chunks. The magic of virtual memory makes it appear contiguous to software.

Downloaded from Elcodis.com electronic components distributor

The SGDMA needs physical addresses to put on the bus and needs contiguous memory. In a simple flat memory architecture, the SGDMA could just take a starting address and a length of 1MB and transfer all data in one continuous operation. In virtual memory machines, the kernel and memory manager need to be enlisted at the driver level to create a map of the virtual memory to physical pages. In Windows, this mapping is known as a Memory Descriptor List (MDL).

The MDL is a Scatter-Gather List that maps virtual memory to physical page addresses. The device driver uses the MDL entries to program the buffer descriptors. Each buffer descriptor is programmed with the physical memory address and the length (usually one page, 4096 bytes). When the SGDMA channel is activated it reads the linked list of buffer descriptors and moves the data to that address, and then moves to the next buffer descriptor and next address until the end of the list is reached. The following figure illustrates this operation.

Figure 14: SGDMA Buffer Address Mapping



The buffer descriptors (BD[1, 2,..], shown on the left in Figure 14 have their destination addresses programmed to the start of the physical pages in memory. These pages in Physical RAM to the right of the buffer descriptors may not be contiguous or sequential in memory. The memory manager in the PC hardware uses the MDL or Scatter Gather List (SG List) to make this set of pages appear contiguous to the application running in user space (Virtual Memory mode).

The SGDMA offloads the processor and kernel by having the ability to perform this scattering of contiguous data (memory on the solutions board) to arbitrary memory pages, or for reading, to gather a set of discontinuous memory pages into a contiguous memory on the solutions board.

Downloaded from Elcodis.com electronic components distributor

See the code links to the DMATest.cpp, ColorBars.cpp and ImageMove.cpp files included with this kit. The source code is the best documentation of what is really going on behind the scenes in the demos. To access this documentation, go to the kit Software directory and open the **PCIeDocIndex.html** document. Under the Documentation section click the hyperlink, **PCIe DMA Demo Reference Manual**. Click on the **File List** book in the navigation pane at left or the **Files** tab in the main pane on the page.

Note:
The PCIe SGDMA demo design requires at least 16 posted credits. This requirement is to optimize the throughput of the PCIe link. You can determine the amount of posted credits for the given slot using the PCIe Throughput demo and GUI. If the posted credits are less than 32, then the PCIe SGDMA demo will not be able to run in the given slot.

If you experience any problems running this demo, please refer to Appendix B, “Troubleshooting” on page 63.

Running the DMA Demos

This section describes how to start running the DMA demos and refers you to documentation on the demos that describes what these applications demonstrate.

To run the ColorBars graphical DMA demo,

- From the Windows desktop, choose **Start > Programs > LatticeECP2M PCI Express Development Kit > ColorBars**.

  In this demo, image data is transferred from the solutions board to PC memory and displayed. The ColorBars window displays a series of vertical colored bars in a gradient manner. See Figure 15 below. For details on this demo, see the section “PCIe DMA ColorBars Demo” on page 52.

To run the ImageMove graphical DMA demo,

- From the Windows desktop, choose **Start > Programs > LatticeECP2M PCI Express Development Kit > ImageMove**.

  In this demo, image data is transferred from the PC to the solutions board and then back to the software, which then displays a modified image on the screen. See Figure 16 below. For details on this demo, see the section “PCIe DMA ImageMove Demo” on page 54

The 16-segment LED displays the real-time interrupt processing during execution of the ColorBars and ImageMove. The inner eight segments are the lower eight bits of the ISR routine counter. The outer eight segments are the lower eight bits of the DPC routine counter, where real processing is done. All segments (inner and outer) should be changing at a rapid rate during demo operation (interrupts after each DMA transfer) indicating that the hardware is operating and interrupts are being serviced.

Downloaded from Elcodis.com electronic components distributor

Figure 15: ColorBars Demo Window

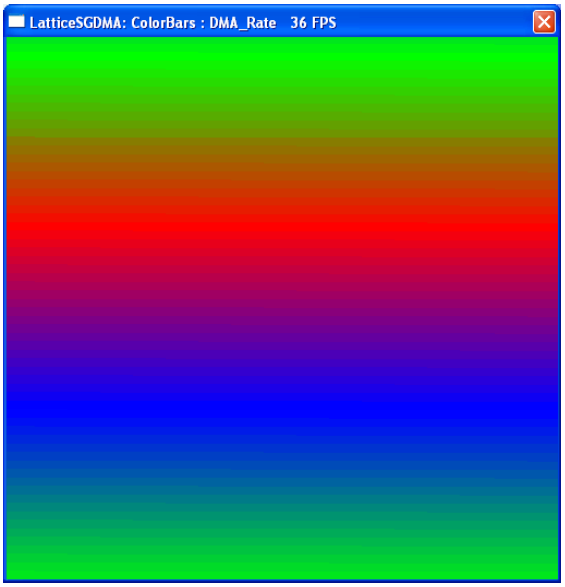
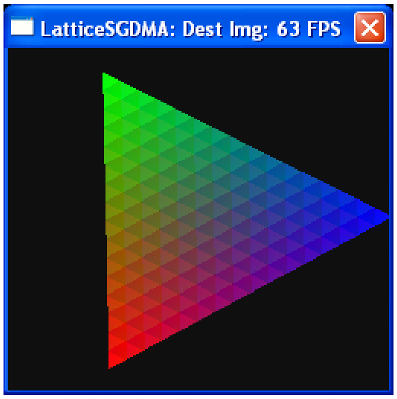


Figure 16: ImageMove Demo Window



A demo can error out displaying an error dialog under the following circumstances:

- The board is not recognized by hardware or the OS.
- The driver is not loaded (bitstream not PCIe demo).
- The PCIe link is not a x4.
- Driver can not access registers.
- Application or driver can not verify IP register IDs.
- Another demo is running.

Downloaded from Elcodis.com electronic components distributor

Running Multiple DMA Demos

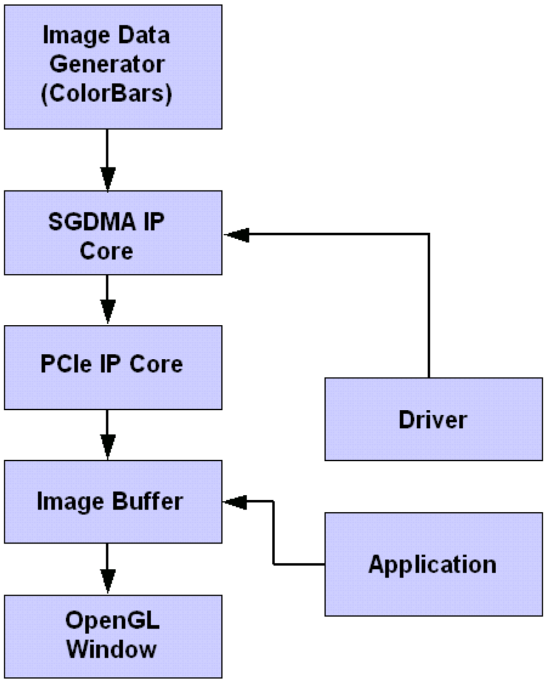
Do not run more than one demo at a time. The ImageMove and ColorBars demos can not be run at the same time because they are mutually exclusive. Each needs DMA channels in the SGDMA. The driver marks channels as in-use once a demo "opens" the channels. Starting another demo will fail when it attempts to open the same channels.

PCIe DMA ColorBars Demo

This program demonstrates the Lattice PCIe IP core and the SGDMA IP core operating on a PCIe Solutions Board. It transfers image data from the solutions board to PC memory and software, which then displays it on the screen.

The image source is a block of IP operating as a FIFO. The IP tracks how many reads have been requested, and after eight complete rows have been read, it changes the color data provided with the next eight rows.

Figure 17: ColorBars Program Operation Flow



The image is displayed using OpenGL calls. The display rate is therefore also dependent on the OpenGL library and graphics subsystem hardware. Displaying an image is a quick way to illustrate that data has been moved. It would not be practical to display 1MB to the screen in a text dump, or save it to a file. An image provides a quick, visual way to observe a large transfer of data, and it can run continuously.

Downloaded from Elcodis.com electronic components distributor

The image data is 1MB in size. Each DMA Read request is 1MB in size. After the hardware has transferred the pixel data, the API call returns and the software displays the image. This loop is repeated over and over. The data rate (frame rate) is displayed in the window title bar. The frame rate is roughly the throughput rate in MB/sec. (each frame = 1 MB). Frame rate is governed by the video refresh rate. Most video systems will not draw frames into video memory faster than the frame rate (waste of operations).

To see what key commands are available for the ColorBars demo, refer to the table below:

Table 12: DMA ColorBars Demo Keyboard Commands

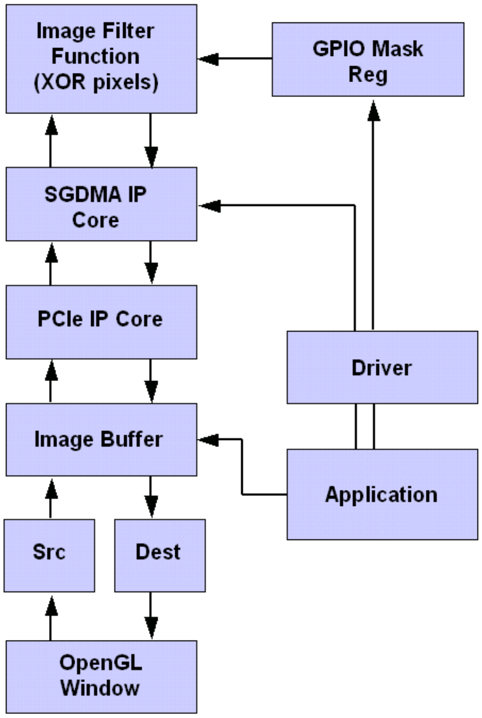
Key command	Description
<Esc>	Terminates program and closes the window.
<F1>	Draw blank image buffer only (do not generate data). This is the fastest rate,
<F2>	Generate ColorBars data with the software. This will usually be slowest data rate.
<F3>	Get image data from solutions board from DMA transfer.
<F4>	Draw a frame each second (slowly) so it can be viewed and the changes are visible.
<Space>	Pause/resume image transfer.

Downloaded from Elcodis.com electronic components distributor

PCIe DMA ImageMove Demo

This file demonstrates the Lattice PCIe IP core and the SGDMA IP core operating on a PCIe solutions board. It transfers image data from the PC to the solutions board and then back to the software, which then displays a modified image on the screen.

Figure 18: ImageMove Program Operation Flow



The image is displayed using OpenGL calls. The display rate is therefore also dependent on the OpenGL library and graphics subsystem hardware. Displaying an image is a quick way to illustrate that data has been moved. An image provides a quick, visual way to observe a large transfer of data. Each image is 256KB in size.

Below is the sequence of events as this demo image undergoes processing:

1. The image source is generated by rotating the triangle shape using OpenGL transform matrix. The resulting image is displayed on the screen.
2. The source image is read from the screen into the source buffer.
3. The source buffer is sent to the Image Filter memory on the solutions board. The memory is only 64KB in size, so the the image is sent in four chunks.
4. After a 64KB chunk is transfered to the board, the 64KB chunk is read back, with the pixels modified by the XOR function in the read path.

Downloaded from Elcodis.com electronic components distributor

5. After four write/read chunks, the desitination buffer contains the modified image and it is displayed on the screen.

Table 13: DMA ImageMove Demo Keyboard Commands

Key command	Description
<Esc>	Terminates program and closes the window ·
<F1>	XOR filter set to 0xcc33aa55 (changes resulting display)
<F2>	XOR filter set to 0xf0f0f0f0 (changes resulting display)
<F3>	XOR filter set to 0x0f0f0f0f (changes resulting display)
<F4>	No filter is applied
<space>	Pause/remove image display

Downloaded from Elcodis.com electronic components distributor

Downloaded from Elcodis.com electronic components distributor

A

Installing and Running the Development Kit in Linux

This appendix describes the PCIe development kit installation and all of the necessary installation to run the PCIe software demo in the Linux environment. You should have already gone through Chapter 2, “Getting Started with the Development Kit” which includes important Linux ispLEVER instructions.

Once you have successfully installed and opened the demo in Linux, you should refer back to Chapter 4 “Running the PCIe Basic Demo Software” on page 23 for instructions on using the demo. The demo graphical user interface will be the same on Linux as it appears in Windows for PC.

Installing the PCIe Development Kit on Linux

This section provides Linux installation instructions for the PCIe development kit. Refer back to “Installing ispLEVER (Windows/Linux)” on page 6 if you have not properly installed a compatible version of the ispLEVER software.

To install the development kit on your Linux machine:

1. Load the kit CD-ROM in your disk drive.
2. In a Terminal window, mount the installation disk. If you are installing from a network, mount the drive by making a directory mount point and using the proper mount argument.
3. Specify a path location for installing the kit, create a directory for Lattice development kits, and change to that directory.

   ```
   mkdir <path>/<install_path>
   cd <path>/<install_path>
   ```

4. Untar the **DK-ECP2M-PCIE-011.tar** file on the CD-ROM to your current working directory.

Downloaded from Elcodis.com electronic components distributor

After you untar the tar file, the directory structure is as follows:

```
DK-ECP2M-PCIE-011
    Demonstration/
    Documentation/
    Hardware/
    Software/
```

The kit directory structure for Linux is the same as Windows with the exception of the top-level, Windows default directory “Lattice_DevKits” which can be user-defined for either platform. Figure 1, “Installed PCIe Development Kit Directory Structure (Windows)” on page 10 illustrates the structure and describes contents in the directories.

Installing the PCIe Demo Package

This section discusses the installation of the demo package, including software installation and the solutions board hardware. Please read this section completely before attempting to install the package so that you understand the steps involved and how they apply to your situation.

The Lattice PCI Express demo package is released as a compressed Linux tar file. The package includes the Linux driver, FPGA bitstreams, Java GUI, and all demo source code. The file must first be unzipped/untarred and then extracted into the final destination directory. A script can then be run to install the device drivers and demo files. Once this is complete, you are ready to run the demo executables.

Before You Begin

This demo assumes that you are familiar with basic PCI Express technology and are comfortable installing new hardware and software packages on Linux. Some experience in these areas is helpful when installing the solutions board and software.

Notes on Installing Linux Drivers

The Linux demonstration components and drivers supplied with this kit are built for Red Hat Workstation 4.0, Update 4 (32-bit kernel 2.6.9-42 i686) and successfully verified on this kernel version. Please be aware that the files provided in the *<kit_dir>/Demonstration/Software* path are only tested for compatibility with this Linux kernel version. Results on other kernel distributions or 64-bit kernel versions have not been fully validated.

You should run the following command initially to determine if you have to rebuild the driver binaries:

```
uname -r -i -m
```

If this command does not show the following information line output,

```
2.6.9-42.ELsmp i686 i386
```

then you will probably have to rebuild the drivers. If you have a 64-bit version of Linux then you will probably also need to rebuild the demos as well.

Downloaded from Elcodis.com electronic components distributor

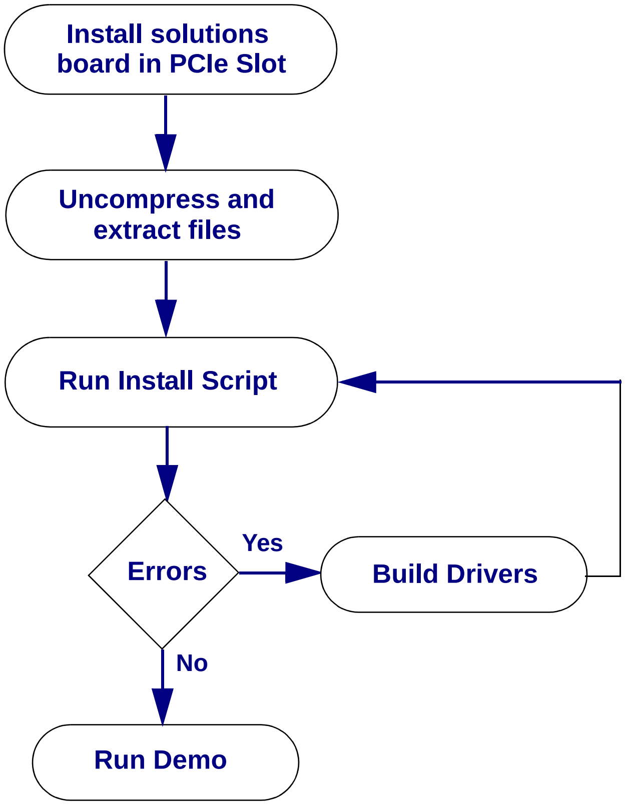
Generally, if you do not have the specific kernel version employed in this kit, you may need to rebuild the driver (and possibly demo application) from the source as described in the section, *"Building Demo Binaries from Source Code" on page 61.* The kernel header files must be installed on your system and you should be familiar with building kernel drivers from source code. Installing the driver requires the root password.

If you need more information on this topic, a good resource is *Linux Device Drivers*, by A. Rubini and J. Corbet. This book details all aspects of Linux driver development.

Installation Overview

The following steps are taken to install and run the demo.

Figure 19: Linux Demo Installation Flow


Install solutions board in PCIe Slot
Uncompress and extract files
Run Install Script
Errors
Yes
Build Drivers
No
Run Demo

Installing Solutions Board Hardware

After board setup, you can install the hardware. You must have root privileges to perform this installation.

To install the Lattice PCI Express Solutions Board on Linux:

1. Shut down your Linux operating system, turn off the machine and unplug the power cord.

Downloaded from Elcodis.com electronic components distributor

IMPORTANT: This step is necessary because power supplies have standby voltages that are present even when your machine's power light and fan are turned off. Unplugging your machine is the safest way to ensure the board will not be "hot-swapped".

2. Locate an available PCI Express slot. The board can be installed in any slot that is larger than the finger edges in use, x1, x4, or x16.
3. Ensure that the solutions board is *not* connected any external power supply before proceeding.
4. Using ESD precautions, install the LatticeECP2M PCI Express Solutions Board in the PCI Express slot in the x1 position.
5. Power-on your machine and observe that it boots normally to the login screen.
6. Log in as a user with root privileges.

Caution:
Lattice is not liable for any loss of date or damages that may result from the installation of the hardware and execution of the kit demo software tools. Do not install and operate on mission-critical systems.

Demo Package Software Installation

This section describes how to install the demo applications. You must have the solutions board installed or the demos will not run. The drivers will not have anything to open, and the demo applications will not have anything to communicate with.

To install the demos and drivers:

1. Change to the Demonstration/ directory.
2. Execute the driver installation script to install the kernel object driver files and demos using the following command.

   ```
   ./install.sh
   ```

   Note that if you get an error message such as, "insmod error inserting... Invalid module format," refer to the section, "Building Demo Binaries from Source Code" on page 61.
3. (Optional) Verify the driver installed by issuing any of the following commands:
   - `ls -l /dev/lsc*` (displays a list of installed Lattice Eval Boards)
   - `cat /proc/modules | grep lsc*` (verifies drivers are installed)
   - `cat /proc/driver/lsc*` (displays information about the solutions boards installed and what BAR resources they have been assigned)
   - `dmesg` shows messages from driver installing and finding Lattice solutions boards

To run the demos:

- Click on the appropriate Quick Launch icon on the desktop for the desired demo. Note that the solutions board must be installed with corresponding bitstream.

Downloaded from Elcodis.com electronic components distributor

- Or, run the desired demo from the command line in the appropriate Demonstration directory.

  ```
  ./rundemo.sh
  ```

  After starting the demo GUI application, go back to Chapter 4, “Running the PCI Express Basic Demo” on page 21 and follow the instructions. If the demo binaries will not run, see the section, “Building Demo Binaries from Source Code” in this appendix to produce the demo binaries.

Building Demo Binaries from Source Code

In the event that the binaries in this release are incompatible with the installed Linux distribution, the binaries can be built from the source code.

Prerequisites:

- GCC and other compiler tools
- Kernel source header files

To build the demo binaries from source code:

1. Change to the Software directory.

   ```
   cd <PCIeDemo_dir>/Software
   ```

2. To ensure your environment is free of previously compiled object files that might conflict with any newly built ones, run the following command,

   ```
   make clean
   ```

3. Build the source files into the executable demos.

   ```
   make demos
   ```

4. Build the drivers.

   ```
   make drivers
   ```

5. Install the files with the install script in the Demonstration directory.

   ```
   cd ../Demonstration
   ./install.sh
   ```

6. If previous versions were installed, then first run the remove script.

   ```
   ./remove.sh
   ```

Installing a Java Runtime Environment

If you are working in a kernel other than the one tested and supplied with this kit, you may have to install a compatible JRE to execute the demo software.

1. Go to the Sun Microsystems web site and download the JRE 5 version at www.java.com/en/download/manual.jsp.

2. Choose a Linux self-extracting file compatible with version jre-1_5_0_14-linux for your machine. Do not choose the RPM. Read the installation instructions located there. This file must match the system kernel and a different file may be required.

3. Save the JRE installation file into the `<demo>/bin` directory (or install globally on the machine).

Downloaded from Elcodis.com electronic components distributor

4. Change permissions on the jre*.bin file.

   ```
   chmod 777
   ```

5. Run the JRE installation file to install.

   ```
   ./jre*.bin
   ```

6. Tell the demo GUI where to find the newly installed Java JRE:

   ```
   export DEMO_JRE=”/usr/local./jre1.5.0_14”
   ```

   Replace the directory argument for DEMO_JRE with specific directory if different than one shown above.

Uninstalling the PCIe Demo Software

If you want to remove the software from your system (i.e., to perform a clean installation of a new version), simply run the `remove.sh` script in the Demonstration/ directory and delete the directory that the development kit was extracted into.

Downloaded from Elcodis.com electronic components distributor

B

Troubleshooting

This appendix outlines some debug procedures to follow when experiencing trouble installing or running a demo on a Windows PC.

Troubleshooting Demo Software Installation

- The most likely installation issue that may arise for the kit demo software will be related to permissions. Depending upon the system security policies, you may need to have Administrator privileges to install into certain directories, for example, the **Program Files** directory in Windows.

Troubleshooting the Solutions Board

- Ensure the board is installed into a PCI Express slot. The board only fits physically into a PCI slot. The board or PC can be damaged if power is applied when you attempt to fit the board into the wrong type of slot.
- Ensure the board has a valid PCI Express bitstream loaded in the SPI flash and for LatticeECP2M that the Mode DIP switches are set to program from SPI flash.
- Ensure the four Status LEDs are on, indicating the board is seen as a PCI Express endpoint. If the two yellow LEDs and two green LEDs are not on, the board is not being recognized by the PC BIOS or Windows. You can try installing in a different PCI Express slot to see if that fixes the link-up problem. You can also try pressing the solutions board's reset button immediately after the PC boots.
- Ensure the board is seen by Windows. From the Windows desktop, right-click on the **My Computer** icon and choose **Properties**. Then, select the **Hardware** tab and click the **Device Manager** button. Note that this may vary on other operating systems. In the Device Manager dialog, verify that the LSC_PCIe driver and solutions board are shown in the list.

Downloaded from Elcodis.com electronic components distributor

If they are not listed, shut down the system and try another slot. If the board is present, check its Properties and the Resource tab to verify memory was assigned to it. Also verify the Vendor ID and Device ID are valid, as seen be Windows Plug-n-Play. If the values are invalid, the bitstream may be corrupt and may need to be reloaded into SPI flash.

Troubleshooting Driver Installation

- The solutions board must be connected to the PC and recognized by Windows for the driver to be successfully installed. If you do not see the "Found New Hardware" message when logging in after installing the board, check the board LEDs. Try a different PCI Express slot.
- Make sure you specify the search location for the driver during installation. Specify that Windows should install from the *Demonstation\<demo_name>\Drivers* directory.
- You must have Administrator privileges to install device driver files.

Troubleshooting Demo Operation

- The solutions board must be installed inthe PC and recognized by Windows for the driver to be successfully installed/loaded. The driver must be loaded by Windows to run the demo. Verify that Windows sees the board and has loaded a driver for it. See section "Troubleshooting the Solutions Board" on page 63.
- If the GUI displays the error message, "ERROR LOADING LIBRARY:Cpp_Jni - running in View Only mode" when executed, then the driver was not found or loaded. There are two causes:
  - The driver was never loaded (or solutions board is not installed)
  - The board failed to be detected by Windows.

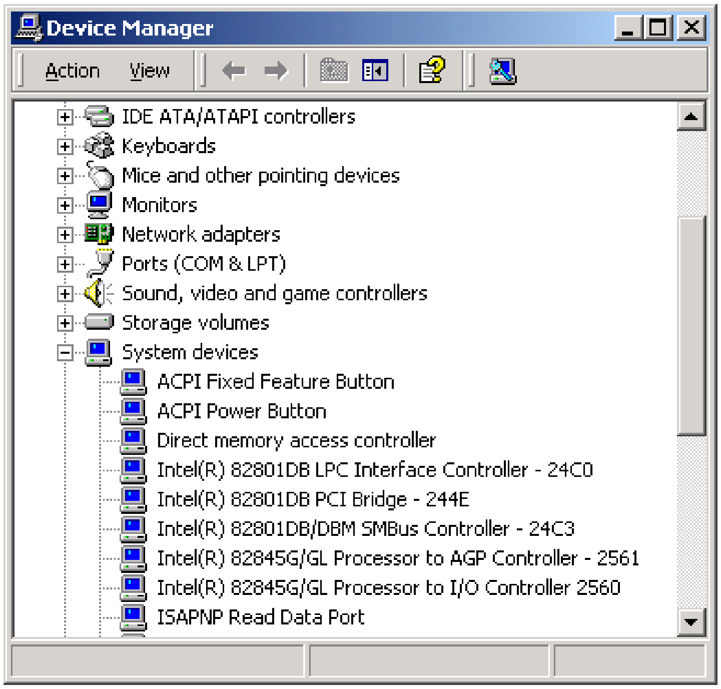
  In either case, the board needs to be installed and seen by Windows and the LSC_PCIe driver needs to be associated with the hardware.

Downloaded from Elcodis.com electronic components distributor

Using Device Manager to Debug Installation

Use Device Manager to get basic information on what hardware you have installed. To access Device Manager, right-click on the **My Computer** desktop icon and select **Properties**. In the System Properties dialog, select the **Hardware** tab and **Device Manager** button.

Figure 20: Device Manager



- IDE ATA/ATAPI controllers
- Keyboards
- Mice and other pointing devices
- Monitors
- Network adapters
- Ports (COM & LPT)
- Sound, video and game controllers
- Storage volumes
- System devices
  - ACPI Fixed Feature Button
  - ACPI Power Button
  - Direct memory access controller
  - Intel(R) 82801DB LPC Interface Controller - 24C0
  - Intel(R) 82801DB PCI Bridge - 244E
  - Intel(R) 82801DB/DBM SMBus Controller - 24C3
  - Intel(R) 82845G/GL Processor to AGP Controller - 2561
  - Intel(R) 82845G/GL Processor to I/O Controller 2560
  - ISAPNP Read Data Port

The Device Manager provides the same basic set of software driver information as in the Computer Management window. The Hardware Wizard allows you to install and remove drivers. You must to have administrator privileges to run the Hardware Wizard and install/remove drivers. Again, the most useful thing is to verify that the **lscpcie** and **lscvpci** drivers (if enabled) have been installed.

Downloaded from Elcodis.com electronic components distributor

Downloaded from Elcodis.com electronic components distributor

Index

Downloaded from Elcodis.com electronic components distributor

Downloaded from Elcodis.com electronic components distributor

Downloaded from Elcodis.com electronic components distributor