PCI Express Root Complex Lite x1 Native Demo

User's Guide

November 2010
UG40_01.0

Introduction

PCI Express is a point-to-point serial protocol that allows connectivity in-board to chip-to-chip, board-to-board, and box-to-box applications. In a PCI Express fabric, a link is established between a downstream port and an upstream port in a PCI Express hierarchy.

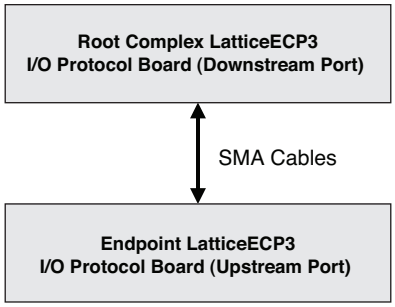
Applications connecting to the upstream port of a switch or an endpoint can use the Lattice PCI Express Root Complex Lite x1, x4 IP core for the required downstream port. The PCI Express Root Complex Lite x1 Native Demo demonstrates a PCI Express link between two LatticeEPC3™ devices.

This document assumes that the user is familiar with basic terms and concepts related to PCI Express protocol. It also assumes the user has some familiarity with the LatticeECP3 device.

Demo Overview

This user's guide describes the PCI Express Root Complex Lite x1 Native Demo. The demo uses two LatticeECP3 I/O Protocol Boards connected via SMA cables to illustrate a x1 PCI Express link between two ports. One board functions as the downstream port (Root Complex). The other functions as the upstream port (Endpoint) as shown in Figure 1.

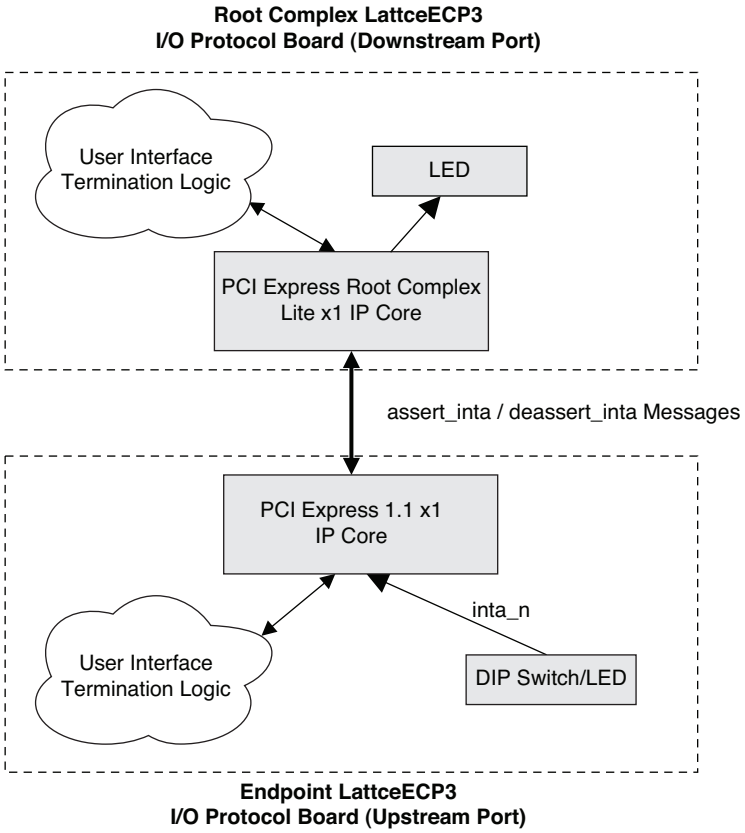
Figure 1. PCI Express Root Complex Lite x1 Native Demo Using Two LatticeECP3 I/O Protocol Boards



The LatticeECP3 I/O Protocol Board at the top of Figure 2 functions as the Root Complex Board. It is implemented in a design including the PCI Express Root Complex Lite x1 IP core, user interface termination logic, and LED logic. The user interface termination logic simply sets inputs of the interface to the correct de-assertion state. The LEDs are used to indicate various statuses of the link and downstream device including PLL Lock, Polling Inactive State, L0 state, DL Up, and INTA Assertion/Deassertion.

The LatticeECP3 I/O Protocol Board at the bottom of Figure 2 functions as the Endpoint Board. It is to be loaded with an endpoint design including the PCI Express x1 Endpoint IP core, user interface termination logic, and LED logic. The user interface termination logic simply set inputs of the interface to the correct de-assertion state. The LED logic is used to indicate various statuses for the link and endpoint device. The DIP switch is used to send interrupt messages to the Root Complex.

Figure 2. Lattice RC Lite Demo Designs



After the FPGA device is programmed, each board goes through the following sequence:

1. Performs receiver detection of the link partner.
2. Performs receiver clock data recovery.
3. Starts training the link.
4. Exchanges initial credits.
5. Enters L0, readying for TLP exchange.

Once in L0, the Endpoint and Root Complex boards behave differently:

- The Endpoint board detects the state of DIP switches SW4-8 to send an interrupt message to the Root Complex Board.
- The Root Complex Board decodes the interrupt message and lights the appropriate LED.

Design Implementation Details

Reference Clock

Both the Root Complex and Endpoint Boards are configured to use their own reference clocks. This is different from PC applications for add-in cards where the endpoint typically sources the reference clock from a slot on the motherboard. The on-board 156 MHz crystal on each board provides the differential reference clocks to the FPGA device. The SERDES Tx PLL is configured with a multiplication factor of 16 to realize a transmit bit rate of 2.5 GHz as well as other transmit clocks used internally.

Receiver Detection

PCI Express protocol requires an AC coupling capacitor to be implemented on the transmitter of each link partner. Since the AC coupling capacitors are not implemented in the LatticeECP3 I/O Protocol Board, the pcs_pipe wrappers in both the Root Complex and Endpoint designs are modified to effectively allow the respective IP core to always see its link partner receiver.

LEDs and DIP Switches

Various LEDs are used to provide the status of the Root Complex and Endpoint devices. DIP switches are also used to send interrupt assertion/deassertion messages to illustrate traffic from the Endpoint to the Root Complex Boards. The LEDs and DIP switches are described in the tables below.

Table 1. Root Complex LatticeECP3 I/O Protocol Board (Downstream Port)

Position	Color	Description
D7	Orange	On indicates the TX PLL is locking.
D8	Yellow	On indicates the LTSSM is not in Polling Compliant state.
D11	Green	On indicates the completion of flow control initialization with the link partner.
D14	Blue	On indicates the link is configured (L0) when TLP can be transferred.
D9		On indicates inta_n assertion. Off indicates inta_n de-assertion.
D6		Not used.

Table 2. Endpoint LatticeECP3 I/O Protocol Board (Upstream Port)

Position	Color	Description
D7	Orange	On indicates the TX PLL is locking.
D8	Yellow	On indicates the LTSSM not in Polling Compliant state.
D11	Green	On indicates the completion of flow control initialization with the link partner.
D14	Blue	On indicates the link is configured (L0) when TLP can be transferred.
D6	Blue	On identifies the Endpoint Board.

Table 3. DIP Switches

Position	Description
SW4-1	Must be up to force successful receiver detection results.
SW4-8	Down to send an assert_inta to the Root Complex Board. Up to send a deassert_inta to the Root Complex Board. Applicable only to the Endpoint Board.

Push-button Switch

Push-button switch PB2 can be used to reset the Endpoint and Root Complex Boards.

Power Supply

The LatticeECP3 I/O Protocol Board uses a 12V power supply. When connected, all 10 green LEDs next to the S1 DIP switch must be turned on.

Demo Hardware

- Two LatticeECP3 I/O Protocol Boards, Revision B or C
- Four SMA cables
- Two 12V wall adapter power supplies
- One USB ispVM™ programming cable

Demo Prerequisites

Before running the demo, the jumper settings on each board must be as shown in Appendix A.

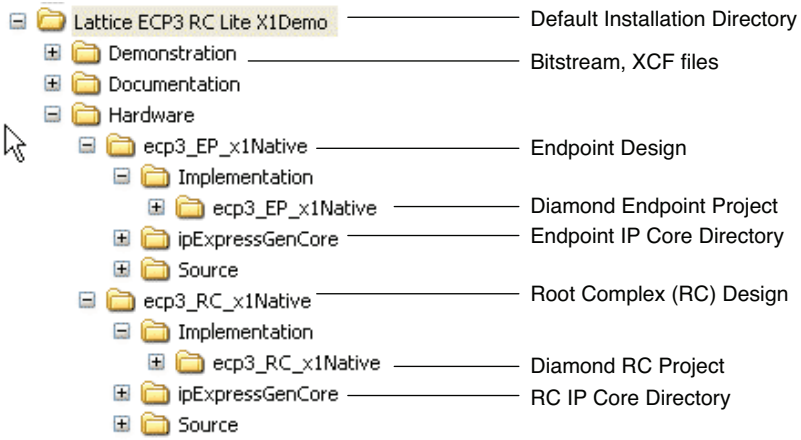
The DIP switches on each board should be set according to Table 4.

Table 4. DIP Switch Settings

DIP Switch	Description
SW1	All switch positions must be up
SW4	All switch positions must be up
SW5	All switch positions must be up for JTAG download All switch position must be down for SPI download

The demo design package is available for download from the Lattice web site. Double-click on the downloaded file to install it. Figure 3 shows the Demo package directory structure.

Figure 3. Lattice RC Lite x1 Demo Package Directory Structure



The bitstreams from the package can be used to program the FPGAs directly via JTAG in the steps described below.

ispVM System software is required for programming the FPGA device. The software can be downloaded and installed in the user system from the ispVM System web page on the Lattice web site.

Demo Procedure

1. Connect the SMA Cables according to the table below.

Source SMA – Root Complex Board	Destination SMA - Endpoint Board
J7	J34
J13	J37
J34	J7
J37	J13

2. Connect the ispVM USB download cable fly-wires to the JTAG connector (J10) on the board per the table below.

JTAG Connector J10	Description
1	3.3V
2	TDO
3	TDI
4, 5	Not used
6	TMS
7	GND
8	TCK
9, 10	Not used

3. Connect the 12V DC power supply to J56 on each board.
4. Power-on the boards.
5. Download the bitstreams included in the demo package to the boards by following the steps described in Appendix B. Two downloads are required, one to program the Endpoint Board and one to program the Root Complex Board. It is not important which board is programmed first.
6. Once both boards are programmed, they will link up with each other. A link is established when the D7, D8, D11, and D14 LEDs are lit on each board. The Endpoint Board is identified when D6 is lit.
7. Traffic is demonstrated by sending interrupt message from the Endpoint to the Root Complex Boards as follows:
   - Pressing the SW4-8 switches causes the Endpoint Board to send an assert_inta message to the Root Complex Board.
   - Successful reception and decoding of the assert_inta message turns on D9 on the Root Complex Board.
   - Pushing SW4-8 up causes the Endpoint Board to send a deassert_inta message to the Root Complex Board.
   - Successful reception and decoding of the deassert_inta message turns on D9 on the Root Complex Board.

Rebuilding the Demo Design

You can rebuild either the Root Complex or the Endpoint demo design by running the source HDL design files through the different implementation process in the Lattice Diamond™ software. All source HDL files and necessary project files are included in the demo package installation. This document assumes that you have already installed Diamond and have a basic understanding of how to use it. Refer to Figure 3 to review the locations of various files referenced in this section.

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

Implementing the Demo Design

The top.ldf Diamond project file is included in the demo installation for each design. This file contains information regarding the options to use when implementing the design. The top.lpf logical preference file specifies timing constraints and LatticeECP3 I/O pin assignments with respect to the LatticeECP3 PCI Express I/O Protocol Board. The working directory is the implementation directory.

To implement the PCI Express Root Complex Lite x1 Native Demo design using the HDL source flow:

1. Open Lattice Diamond.
2. Click **File > Open > Open Project**.
3. In the Open Project dialog, navigate to and the select the top.ldf file in the **<demo_install_dir>\Hardware\ecp3_RC_x1Native\Implementation\ecp3_RC_x1Native** directory path.
4. Click **Open**. All of the Verilog HDL files are imported into the project.
5. In the Processes window, right-click on **Build Database** and select **Properties** from the pop-up menu.
6. In **Project > Active Strategy > Translate Design Settings**, verify that **Macro Search Path** is set to the directory path **..\..\ipExpressGenCore\ecp3\pciex1Native** for Windows. Your path would have forward slashes for Linux.
7. In the **Diamond > File List** tab, verify that the device selected is **LFE3-150EA-7FN1156CES**.
8. In the **Diamond > Process** tab, double-click the **Bitstream File** process.

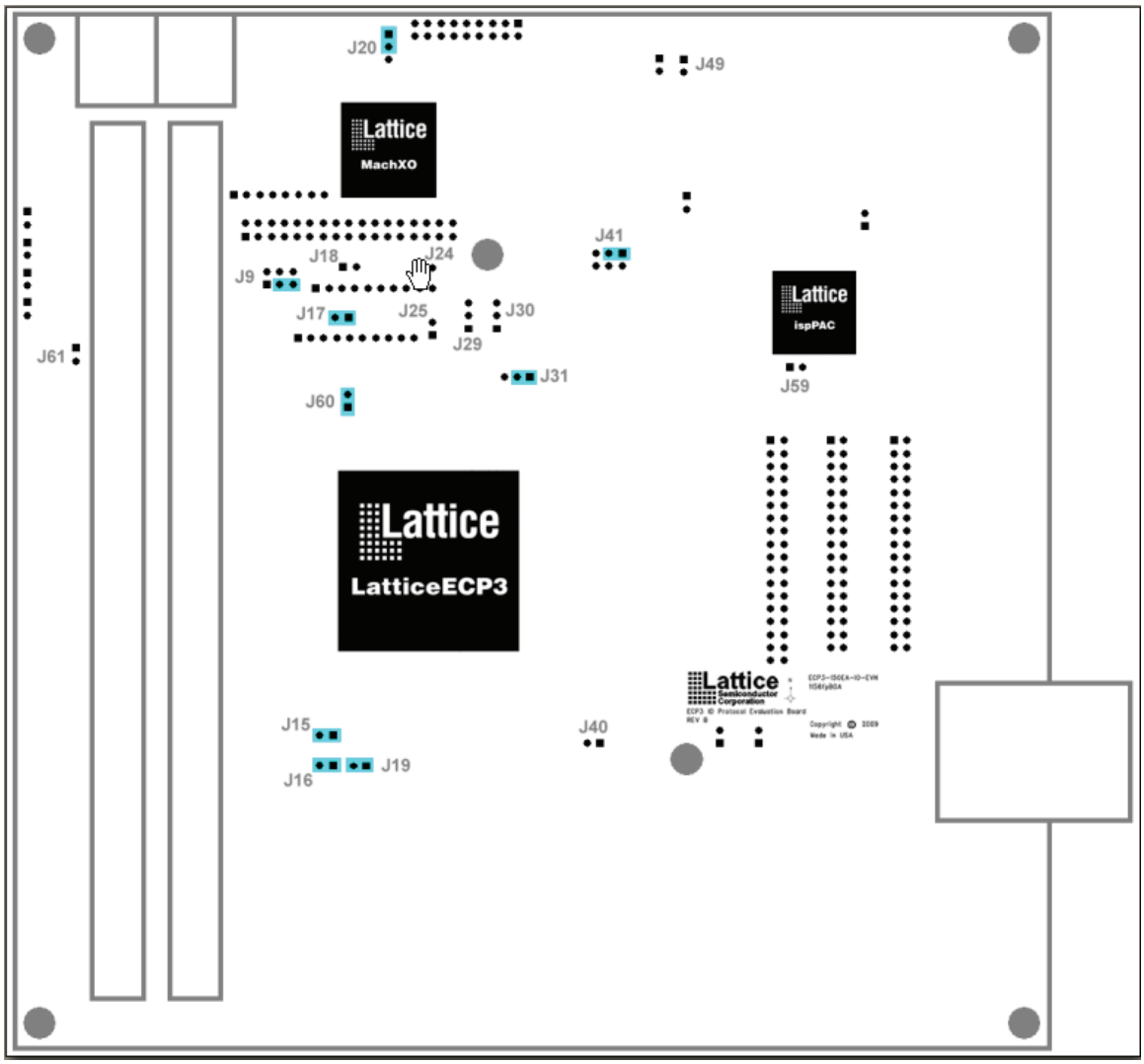
Technical Support Assistance

Hotline: 1-800-LATTICE (North America)
+1-503-268-8001 (Outside North America)
e-mail: techsupport@latticesemi.com
Internet: www.latticesemi.com

Revision History

Date	Version	Change Summary
November 2010	01.0	Initial release.

Appendix A. LatticeECP3 I/O Protocol Board Default Jumper Settings



Appendix B. FPGA Programming

This appendix describes the steps to program the FPGA directly using the ispVM USB download cable.

1. From the Windows Start menu, click **Start > Programs > Lattice Diamond 1.0 > Accessories > ispVM System** to open ispVM.
2. Click **File > Open** and navigate to the XCF file under **<demo_pkg_install_dir>\Demonstration\Bitstreams**

   Select the **EP_x1Native_JTAG.xcf** for programming the FPGA on the Endpoint (upstream) board.

   Select the **RC_x1Native_JTAG.xcf** for programming the FPGA on the Root Complex (downstream) board.

   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 match the XCF file's internal date.
3. Go to **ispVM > Options > Cable and I/O Port Setup > Cable Type** and ensure that you are using the USB cable type.
4. Click the **Go** button to initiate your download. You must wait while the FPGA device on the board is programmed.
5. 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.

Appendix C. Troubleshooting

This appendix outlines some debug procedures to follow when experiencing trouble running the demo.

- Make sure that none of the ten LEDs next to DIP switch S1 are blinking or are turned off. Otherwise, the supply associated with the LED is out of range. This could cause the device to not work properly.
- Make sure that the DONE LED (D5) is lighted. This LED indicates that the FPGA has been successfully programmed.
- Make sure the SMA cables are properly connected as listed in Step 1 of the Demo Procedure.

Note: The polarity on one end can be swapped since this is automatically handled by the designs.