



PCI Development Kit, Cyclone II Edition

Getting Started User Guide



101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
www.altera.com
P25-11480-00

Development Kit Version: 1.0.0
Document Version: 1.0.0
Document Date: May 2005

Copyright © 2005 Altera Corporation. All rights reserved. Altera, The Programmable Solutions Company, the stylized Altera logo, specific device designations, and all other words and logos that are identified as trademarks and/or service marks are, unless noted otherwise, the trademarks and service marks of Altera Corporation in the U.S. and other countries. All other product or service names are the property of their respective holders. Altera products are protected under numerous U.S. and foreign patents and pending applications, maskwork rights, and copyrights. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera Corporation. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.



UG-C2042505-1.0

Development Kit Version 1.0.0
PCI Development Kit, Cyclone II Edition Getting Started User Guide

Altera Corporation
May 2005



About This User Guide

Revision History	v
How to Contact Altera	v
Typographic Conventions	vi

Chapter 1. About This Kit

Introduction	1-1
Kit Features	1-1
Documentation	1-3

Chapter 2. Getting Started

Introduction	2-1
Before You Begin	2-1
PCI Development Kit, Cyclone II Edition Contents	2-2
Inspect the Board	2-2
Hardware Requirements	2-2
Software Requirements	2-3
Quartus II System Requirements	2-3
Software Installation	2-4
Installing the PCI Development Kit, Cyclone II Edition CD-ROM Contents	2-4
Installing the Quartus II Software & MegaCore Functions	2-5
Setting Up Licensing	2-6
Licensing the PCI Development Kit, Cyclone II Edition	2-6
Licensing MegaCore Functions	2-7
Install the Board in Your PC	2-8
Perform PCI Transactions with the Kit's Demo Application	2-9
PCI Target Write Transaction (Demo Tab)	2-9
PCI Master Write Transaction (Demo Tab)	2-10
Latency Timer Configuration Register (Debug Tab)	2-11
PCI Master Loop (Debug Tab)	2-13
Address Offset & Target Loop (Debug Tab)	2-14
Configuring the Cyclone II Device	2-15
Serial Flash Configuration	2-15
JTAG Configuration	2-15
Programming the EPCS64 Serial Flash	2-16
Active Serial Mode	2-17
JTAG Programming via Serial Flash Loader	2-17
Restoring the Factory-Programmed Design	2-18
Using the Board as a Prototyping Platform	2-18
Prototyping Walkthrough Files	2-19
Prototyping Walkthrough	2-22



About This User Guide

Revision History The table below displays the revision history for the chapters in this User Guide.

Chapter	Date	Version	Changes Made
All	May 2005	1.0.0	First publication






How to Contact Altera

For the most up-to-date information about Altera® products, go to the Altera world-wide web site at www.altera.com. For technical support on this product, go to www.altera.com/mysupport. For additional information about Altera products, consult the sources shown below.

Information Type	USA & Canada	All Other Locations
Technical support	www.altera.com/mysupport/	www.altera.com/mysupport/
	(800) 800-EPLD (3753) (7:00 a.m. to 5:00 p.m. Pacific Time)	+1 408-544-8767 7:00 a.m. to 5:00 p.m. (GMT -8:00) Pacific Time
Product literature	www.altera.com	www.altera.com
Altera literature services	literature@altera.com	literature@altera.com
Non-technical customer service	(800) 767-3753	+ 1 408-544-7000 7:00 a.m. to 5:00 p.m. (GMT -8:00) Pacific Time
FTP site	ftp.altera.com	ftp.altera.com

Typographic Conventions

This document uses the typographic conventions shown below.

Visual Cue	Meaning
Bold Type with Initial Capital Letters	Command names, dialog box titles, checkbox options, and dialog box options are shown in bold, initial capital letters. Example: Save As dialog box.
bold type	External timing parameters, directory names, project names, disk drive names, filenames, filename extensions, and software utility names are shown in bold type. Examples: f_{MAX} , qdesigns directory, d: drive, chiptrip.gdf file.
<i>Italic Type with Initial Capital Letters</i>	Document titles are shown in italic type with initial capital letters. Example: <i>AN 75: High-Speed Board Design</i> .
<i>Italic type</i>	Internal timing parameters and variables are shown in italic type. Examples: <i>t_{PIA}</i> , <i>n + 1</i> . Variable names are enclosed in angle brackets (< >) and shown in italic type. Example: <file name>, <project name>.pof file.
Initial Capital Letters	Keyboard keys and menu names are shown with initial capital letters. Examples: Delete key, the Options menu.
“Subheading Title”	References to sections within a document and titles of on-line help topics are shown in quotation marks. Example: “Typographic Conventions.”
Courier type	Signal and port names are shown in lowercase Courier type. Examples: data1, tdi, input. Active-low signals are denoted by suffix n, e.g., resetn. Anything that must be typed exactly as it appears is shown in Courier type. For example: c:\qdesigns\tutorial\chiptrip.gdf. Also, sections of an actual file, such as a Report File, references to parts of files (e.g., the AHDL keyword SUBDESIGN), as well as logic function names (e.g., TRI) are shown in Courier.
1., 2., 3., and a., b., c., etc.	Numbered steps are used in a list of items when the sequence of the items is important, such as the steps listed in a procedure.
	Bullets are used in a list of items when the sequence of the items is not important.
	The checkmark indicates a procedure that consists of one step only.
	The hand points to information that requires special attention.
	The angled arrow indicates you should press the Enter key.
	The feet direct you to more information on a particular topic.

Introduction

The PCI Development Kit, Cyclone™ II Edition provides everything you need to develop and test a complete PCI system based on the Cyclone II device and using Altera® PCI MegaCore® functions.

This chapter briefly describes the kit's features and documentation including:

- The Cyclone II EP2C35 PCI development board
- *Quartus® II Software, Development Kit Edition (DKE)*
- PCI-to-DDR2 reference design
- MegaCore IP Library CD-ROM

Kit Features

The PCI Development Kit, Cyclone II Edition, features:

- *The Cyclone II EP2C35 PCI Development Board*—a prototyping platform that allows you to develop and prototype high-speed PCI and PCI-X bus interfaces, DDR2 SDRAM, and the 10/100 Ethernet interface.



For specific information about board components and interfaces, refer to the *Cyclone II EP2C35 PCI Development Board Reference Manual*.

- *PCI-to-DDR2 Reference Design*—This design is a 64-bit, 66-MHz, hardware verified, open-source PCI-to-DDR2 SDRAM memory reference design that uses the Altera `pci_mt64` and DDR & DDR2 SDRAM Controller MegaCore functions. The design is useful for a variety of hardware applications and lets you quickly begin prototyping and verification.
- *Cyclone II PCI Development Kit Application & Drivers*—This open-source Windows application is an interactive platform you can use to perform PCI transactions. You can also use the kit's application as a starting point for developing your own custom software.
- *PLD Applications PCI-X CORE CD-ROM*—PLD Applications' PCI-X IP core is a versatile integrated solution to interface any user application or system to 32- and 64-bit PCI-X buses. The core is fully customizable; most of its features can be enabled or disabled to suit

specific design requirements. This IP core includes a complete set of tools, including reference designs and applications targeted to the Cyclone II EP2C35 PCI development board.

- *Jungo WinDriver Development Toolkit*—Jungo’s WinDriver is a driver development toolkit that automates and simplifies the development of user mode Windows device drivers for PCI buses. WinDriver is designed to enable development of high performance, high quality user-mode device drivers, and does not require DDK knowledge or kernel-level development.
- *Quartus II Software, Development Kit Edition (DKE), version 5.0*—The Quartus II software provides a comprehensive environment for system-on-a-programmable-chip (SOPC) design. The Quartus II software integrates into nearly any design environment, with interfaces to industry-standard EDA tools. The kit includes:
 - The SOPC Builder system development tool
 - A one-year Quartus II DKE software license, Windows platform only



The Quartus II DKE software license allows you to use the product for 12 months. After 12 months, the Quartus II DKE software license expires and you must purchase another Quartus II license to continue using the Quartus II software. For more information, refer to the Altera web site at www.altera.com.

- *MegaCore IP Library CD-ROM, version 5.0*—This CD-ROM contains Altera IP MegaCore functions. You can evaluate the MegaCore functions using the OpenCore® Plus feature, which allows you to:
 - Simulate the behavior of a MegaCore function within your system
 - Verify the functionality of your design, as well as quickly and easily evaluate its size and speed
 - Generate time-limited device programming files for designs that include MegaCore functions
 - Program a device and verify your design in hardware

You only need to purchase a license for a MegaCore function when you are completely satisfied with its functionality and performance, and want to take your design to production.



The OpenCore Plus hardware evaluation feature is an evaluation tool for prototyping only. You must purchase a license to use an Altera IP function in production designs.

Documentation

The PCI Development Kit, Cyclone II Edition contains the following documents:

- *Readme file*—Contains special instructions for the kit and last minute additions to the documentation.
- *PCI Development Kit, Cyclone II Edition Getting Started User Guide*—Describes how to start using the kit (this document).
- *Cyclone II EP2C35 PCI Development Board Reference Manual*—Provides specific information about the board's components and interfaces, steps for using the board, and pin-outs and signal specifications
- *AN 390: PCI-to-DDR2 SDRAM Reference Design*—Describes the reference design that is included with the kit. The design is a typical PCI implementation of the user application that interfaces to the DDR2 Controller MegaCore function.

Introduction

The PCI Development Kit, Cyclone™ II Edition is a complete PCI and PCI-X prototyping and testing kit, based on the Cyclone II device. With this kit, you can perform various PCI transactions between the board and the host PC, as well as configure the board with either the factory-programmed or user-programmable design.

In addition to providing a PCI/PCI-X form factor development board, the kit also includes all of the hardware and software development tools, as well as the documentation and accessories you need to begin developing PCI systems using the Cyclone II device.

This user guide familiarizes you with the contents of the kit and walks you through setting up a PCI development environment.

In this guide, you will do the following:

- Inspect the contents of the kit
- Install the development tools software
- Set up licensing
- Connect the cables to the board and your PC
- Test the board using the factory-programmed design
- Install the board in your PC
- Perform example PCI transactions
- Configure the Cyclone II device
- Use the board as a prototyping platform

Before You Begin

Before using the kit or installing the software, be sure to check the kit's contents and inspect the board to verify that you received all of the items listed below. If any of the items are missing, contact Altera® before you proceed. You should also verify that your computer's hardware and software meet the kit's system requirements.

PCI Development Kit, Cyclone II Edition Contents

The *PCI Development Kit, Cyclone II Edition* contains the following items:

- Cyclone II EP2C35 PCI development board with an EP2C35F672 Cyclone II device (ordering code: PCI-DEVKIT-2C35)
- *PCI Development Kit, Cyclone II Edition CD-ROM, version 1.0.0*
 - PCI-to-DDR2 reference design
 - Cyclone II PCI development kit application and device driver
 - Board manufacturing files
- *Quartus® II Software Development Kit Edition (DKE), version 5.0*
- *Jungo WinDriver Development Toolkit*



The Jungo CD-ROM provides a free 30-day evaluation of Jungo's Driver Development Toolkit. For more information on the driver, refer to the documentation on the CD-ROM.

- *PLD Applications PCI-X CORE CD-ROM*
- *MegaCore® IP Library CD-ROM, version 5.0*, which contains the PCI Compiler and DDR & DDR2 Controller Compiler
- USB-Blaster™ download cable and USB cable
- Power supply and three separate adapter cables for North America/Japan, Europe, and the United Kingdom

Inspect the Board

Place the board on an anti-static surface and inspect it to ensure that it has not been damaged during shipment. Verify that all components are on the board and appear intact.



Without proper anti-static handling, the Cyclone II EP2C35 PCI development board can be damaged.



Refer to the *Cyclone II EP2C35 PCI Development Board Reference Manual* (available on the *PCI Development Kit, Cyclone II Edition CD-ROM*) for information on the board's components.

Hardware Requirements

The Cyclone II EP2C35 PCI development board is initially configured for installation in a conventional PCI slot.

You can use the Cyclone II EP2C35 PCI development board in either a 3.3- or 5.0-V PCI system and in either a 32- or 64-bit PCI bus slot. To use the board with the kit's demo application, you need a PC with an available PCI or PCI-X bus slot.

Software Requirements

To use the kit's demo application, you must have Windows XP, SP1.

You should install the following software before you begin developing designs for the kit:

- The Quartus II software version 5.0. See ["Installing the Quartus II Software & MegaCore Functions"](#) on page 2–5.
- Internet Explorer 5.0 or later to use Quartus II Help. You need a web browser to register the Quartus II software and request license files.

You need your network identification card (NIC) ID for licensing the Quartus II software.



Your NIC ID is a 12-digit hexadecimal number that identifies your computer to the computer that serves Quartus II licenses. Networked (or floating-node) licensing requires a NIC ID or server host ID. When obtaining a license file for network licensing, you should use the NIC ID from the PC that will issue the Quartus II licenses to distributed users over a network. You can find the NIC ID for your card by typing `ipconfig /all` at a command prompt. Your NIC ID is the number on the physical address line, without the dashes.

Quartus II System Requirements

To use the *PCI Development Kit, Cyclone II Edition CD-ROM* with the Quartus II software provided with the kit, your system must meet the Quartus II software minimum requirements.



Refer to the *Quartus II Installation & Licensing Manual for PCs* for system requirements.

Software Installation

The instructions in this section explain how to install the following:

- *PCI Development Kit, Cyclone II Edition CD-ROM*
- *The Quartus II Software, Development Kit Edition, including MegaCore functions from the MegaCore IP Library CD-ROM*

Installing the PCI Development Kit, Cyclone II Edition CD-ROM Contents

The *PCI Development Kit, Cyclone II Edition CD-ROM* contains the following items:

- *PCI Development Kit, Cyclone II Edition application and drivers*
- *PCI-to-DDR2 reference design and configuration files*
- *PCI Development Kit, Cyclone II Edition Getting Started User Guide* (this document)
- *Cyclone II EP2C35 PCI Development Board Reference Manual*



Before you can compile the PCI-to-DDR2 reference design, you must install the MegaCore IP Library CD-ROM.

- *AN 390: PCI-to-DDR2 SDRAM Reference Design*
- *Board manufacturing files*

To install the *PCI Development Kit, Cyclone II Edition CD-ROM*, perform the following steps:

1. Insert the *PCI Development Kit, Cyclone II Edition CD-ROM* into your CD-ROM drive and double click the **cycloneii_pci_kit-v1.0.0.exe** file.
2. Follow the online instructions to complete the installation process.

The installation program copies the PCI development kit files to your hard-disk, installs the software driver and application, and creates an icon in **Programs > Altera > MegaCore > PCI Development Kit, Cyclone II Edition v1.0.0** (Windows Start menu), which you can use to launch the Windows development kit application and view the kit's documentation.

When the installation is complete, the PCI Development Kit, Cyclone II Edition installation program creates the directory structure shown in [Figure 2-1](#), where <path> is the *PCI Development Kit, Cyclone II Edition CD-ROM* installation directory.

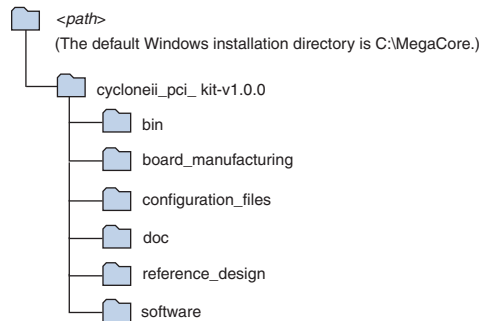
Figure 2–1. PCI Development Kit Installed Directory Structure

Table 2–1 lists the file directory names and a description of their contents.

Table 2–1. Installed File Directory Names & Description of Contents	
Directory Name	Description of Contents
bin	Contains the kit's demo application and device driver.
board_manufacturing	Contains the board design and production test files. You can use the board design files as a starting point for creating your own board.
configuration_files	Contains the configuration files for the Cyclone II EP2C35 PCI development board.
doc	Contains the documentation related to the development kit.
reference_design	Contains the PCI-to-DDR2 reference design files including open-source HDL, Quartus II synthesis and simulation files.
software	Contains the source code for the kit's demo application and device driver.

Installing the Quartus II Software & MegaCore Functions

Refer to *Installing the Quartus II Software in the Quartus II Installation & Licensing Manual for PCs* for software installation instructions. After installing the software, request and install a license to enable it. Refer to “[Setting Up Licensing](#)” for more information.



During the installation of the Quartus II software, you are given the option to install the MegaCore IP Library. When prompted to do so, choose to install the MegaCore IP Library and follow the on-screen instructions.

Setting Up Licensing

This section describes the software licensing procedures.

Licensing the PCI Development Kit, Cyclone II Edition

Before using the Quartus II software, you must obtain a license file from the Altera web site at www.altera.com and install the license file on your PC.



The Quartus II DKE software license allows you to use the product for 12 months. After 12 months, you must purchase a Fixed PC or FloatNet subscription.

To obtain a license, follow these steps:



If you have a Fixed PC or FloatNet Quartus II subscription, you can use that software instead of the Quartus II DKE software. If you intend to use your existing licensed software, you can skip the instructions below to obtain the license for the Quartus II Software Development Kit Edition included in your development kit.

1. Select the Licensing link at the top-right corner on the home page of the Altera Web site at www.altera.com.
2. Click **Cyclone II Development Kits**.
3. Follow the instructions to request your license. Altera will e-mail you a license file that enables the software. You need your network identification card (NIC) ID and the kit serial number to license the Quartus II software.



Your network interface card (NIC) ID is a 12-character hexadecimal number that uniquely identifies your computer. You can find the NIC ID for your card by typing `ipconfig /all` at a command prompt. Your NIC ID is the number on the physical address line.

The kit serial number is an 11-digit code of the form 2C35PCXXXXX where the X's represent decimal numbers. This serial number is located in three places: on the external shipping box, internal box, and Quartus II CD-ROM jacket. Refer to the serial number sticker in [Figure 2-2](#).

Figure 2–2. Serial Number Example

The serial number is the bottom-most number, which is 2C35PCXXXXX in the above example.

4. After receiving your license that Altera e-mails you, close the following software applications if they are open:
 - Quartus II software
 - MAX+PLUS II software
 - LeonardoSpectrum synthesis tool
 - Synplify synthesis software
 - ModelSim simulator software
 - Precision RTL synthesis software
5. To install your license, refer to *Specifying the License File* in the *Quartus II Design Software Installation & Licensing for PCs* manual, which is included in the *PCI Development Kit, Cyclone II Edition*.

Licensing MegaCore Functions

You only need to purchase a license for a MegaCore function when you are completely satisfied with its functionality and performance, and want to take your design to production.



The OpenCore Plus hardware evaluation feature is an evaluation tool for prototyping only. You must purchase a license to use an Altera IP function in production designs.

For details on licensing the OpenCore Plus hardware feature, refer to the application note, *AN 320: OpenCore Plus Evaluation of Megafunctions*.

To install the kit's driver, perform the following steps in the **Found New Hardware Wizard**:

1. Turn on **Install the software automatically (recommended)**.
2. Click **Next** to go to the **Completing the Found New Hardware Wizard** window.
3. Click **Finish** to complete the driver installation.

Perform PCI Transactions with the Kit's Demo Application

This section explains how to use the kit's demo application to perform the following PCI transactions:

- PCI target write (Demo tab)
- PCI master write (Demo tab)
- Latency timer configuration register & PCI Master write (Debug tab)
- PCI master loop (Debug tab)
- Address offset and target loop (Debug tab)

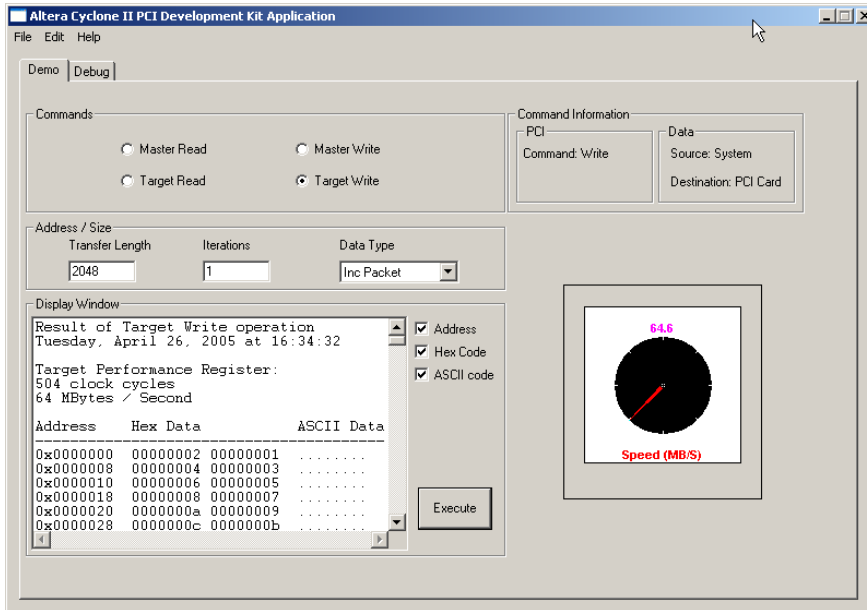
PCI Target Write Transaction (Demo Tab)

In this example, the data source is the system and the destination is the PCI board. You can verify the setup in the **Command Information** section of the kit's demo application.

1. Run the development kit's application by choosing **Programs > Altera > Megacore > PCI Development Kit, Cyclone II Edition v1.0.0 > Cyclone II PCI Kit** (Windows Start menu). The kit application opens to the **Demo** tab with a PCI target write transaction selected for one iteration of 2,048 bytes of random data. Leave the kit application running for the remaining examples.
2. Select **Target Write** from the **Commands** box.
3. Keep the following default values in the **Address/Size** box:
 - Transfer Length: 2048
 - Iterations: 1
4. Click **Execute** to begin operation.
5. Review the results in the **Display Window**.
6. In the **Data Type** list, select **Inc Packet**.
7. Click **Execute**.

- Review the transaction's speed from the performance meter and the transaction's results from the **Display Window** (see [Figure 2-4](#)).

Figure 2-4. PCI Target Write Transaction (Demo Tab)

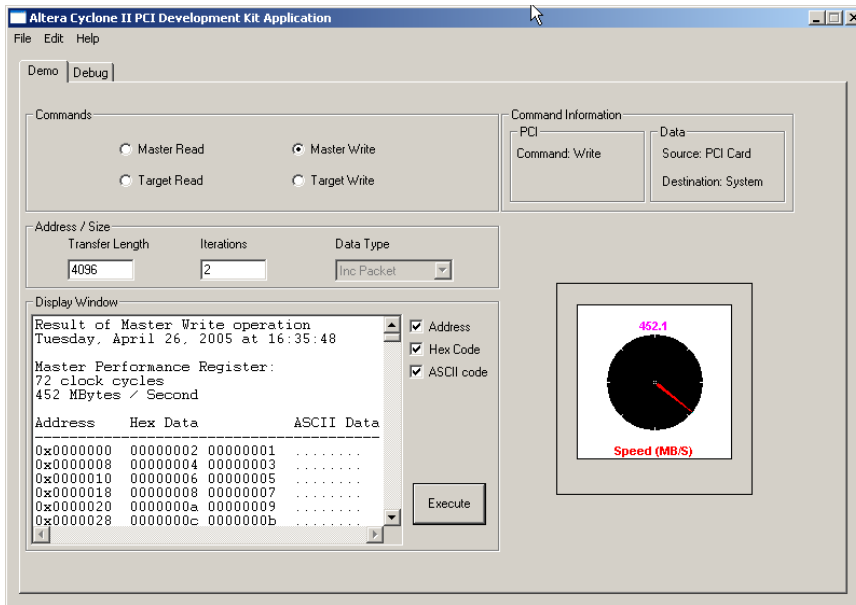


PCI Master Write Transaction (Demo Tab)

In this example, the data source is the PCI board and the destination is the system memory. You can verify this setup in the **Command Information** section of the kit's application.

- Ensure that the **Demo** tab is active.
- Select **Master Write** from the **Commands** box.
- Specify the following settings in the **Address/Size** box:
 - Transfer Length: 4096
 - Iterations: 2
- Click **Execute**.
- Review the transaction's speed from the performance meter and the transaction's results from the **Display Window** (see [Figure 2-5](#)).

Figure 2–5. PCI Master Write Transaction (Demo Tab)



Latency Timer Configuration Register (Debug Tab)

In this example, the data source is the PCI board and the destination is the system memory, employing a user-configured value for the latency timer. You can verify this setup in the Command Information section of the kit's application.

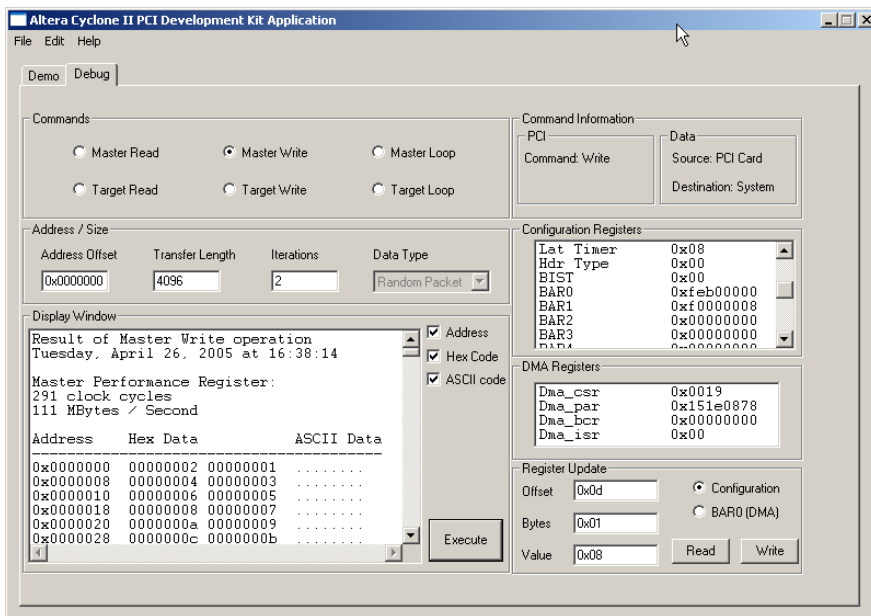
1. Click the **Debug** tab.
2. Select **Master Write** from the **Commands** box.
3. Keep the **Address Offset** setting of 0x0000000.
4. Keep the following values in the **Address/Size** box:
 - Transfer Length: 4,096
 - Iterations: 2
5. In the **Configuration Registers** list, select **Lat Timer**.



The **Lat Timer** is the default value with which the system programmed the latency timer. You can return the setting to its default value after running this example.

6. Under **Value**, type 0x08 (**Register Update** box).
7. Click **Write**.
8. Click **Execute**.
9. Review the results in the **Display Window**. Compare the results with [Figure 2-5](#), which uses the default **Lat Timer** setting of 0x40. See [Figure 2-6](#).
10. Change the latency timer register back to the default value.

Figure 2-6. Latency Timer Configuration Register & PCI Master Write (Debug Tab)

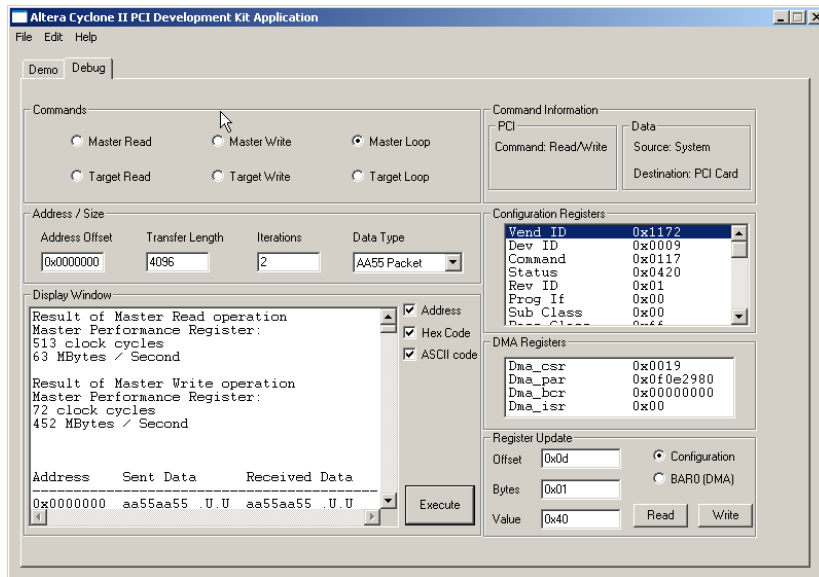


PCI Master Loop (Debug Tab)

In this example, a PCI master read is followed by a PCI master write transaction. The kit's application verifies that the data read and written by the master is the same.

1. Ensure that the **Debug** tab is active.
2. Select **Master Loop** from the **Commands** box.
3. Keep the **Address Offset** setting of 0x0000000.
4. Specify the following settings in the **Address/Size** box:
 - Transfer Length: 4096
 - Iterations: 2
5. In the **Data Type** list, select **AA55 Packet**.
6. Click **Execute**.
7. Review the results in the **Display Window** (see [Figure 2-7](#)).

Figure 2-7. PCI Master Loop (Debug Tab)

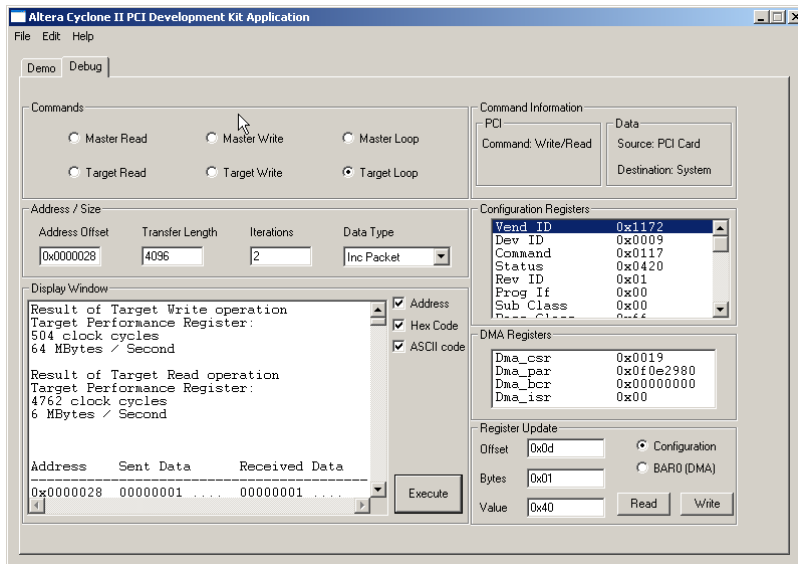


Address Offset & Target Loop (Debug Tab)

In this example, a target loop/target write transaction is followed by a target read. The address offset changes the transaction's starting address.

1. Ensure that the **Debug** tab is active.
2. Select **Target Loop** from the **Commands** box.
3. In the **Address Offset** box, type 0x0000028.
4. Keep the following values in the **Address/Size** box:
 - Transfer Length: 4096
 - Iterations: 2
5. In the **Data Type** list, select **Inc Packet**.
6. Click **Execute**.
7. Review the starting address of the data transaction in the **Display Window** (Figure 2–8).

Figure 2–8. Address Offset & Target Loop (Debug Tab)



Configuring the Cyclone II Device

The on-board Cyclone II device can be configured in one of two ways:

- Serial flash configuration
- JTAG configuration

Serial Flash Configuration

The Cyclone II FPGAs use SRAM cells to store configuration data. Because SRAM memory is volatile, configuration data must be downloaded to the Cyclone FPGAs each time power is applied to the board.

The board has a non-volatile configuration scheme that automatically configures the Cyclone II device with either a user-programmable or factory-programmed default design. A switch (J3) is used to select either the user-programmable or the factory-programmed EPCS64 device.

If the switch (J3) is not in the *Down* position (toward the back of the board), you will erase the factory-programmed default design. Refer to [“Restoring the Factory-Programmed Design” on page 2–18](#).

Configuration via User-Programmable Flash Memory

Upon power-up, the configuration circuit (comprised of the selected EPCS64 device) configures the Cyclone II device. If the switch (J3) is set for user configuration (*Down* position), the circuit attempts to load the user design. If the load is not successful, the CONF_DONE LED (D10) does not illuminate and the Cyclone II device is not configured. If the load is successful, the CONF_DONE LED illuminates.

Configuration via Factory-Programmed Design

Assuming the board is installed in a PCI slot, when the factory-programmed design is loaded into the Cyclone II device, the user LEDs blink and the CONF_DONE LED illuminates. To select the factory default design, set the switch (J3) to the *Up* position.

JTAG Configuration

The Cyclone II device can be configured after power is applied to the board. The JTAG interface permits the Quartus II software to load the Cyclone II device with a user design through the Altera USB-Blaster download cable. The user design remains in the Cyclone II device until power is removed from the board.

To configure the Cyclone II device using the Quartus II software and the USB-Blaster cable, perform the following steps:

1. Attach the USB-Blaster cable to the J8 header.



The board has two connectors that can accept the USB-Blaster cable for this operation. You must use the connector on the PCI bracket next to the Ethernet and RS-232 connectors.

2. Open the Quartus II SRAM Object File (.sof) that you want to load into the device. This step launches the Quartus II Programmer.
3. Select USB-Blaster as the hardware. For instructions, search for *Changing the Hardware Setup* in Quartus II Help.
4. Set the mode to JTAG.
5. Click **Start**.



If the board is installed into a computer's PCI slot when it is configured by the USB-Blaster™ cable, the computer system could lock up. If this happens, reset the computer. Do not shut down the computer or the configuration will be lost. Restart the computer to re-enumerate the PCI bus.

If the configuration is successful, the CONF_DONE LED (D10) illuminates.



For information on how to use the USB-Blaster cable, refer to:

- *USB-Blaster Download Cable User Guide*
- Quartus II Help

Programming the EPCS64 Serial Flash

You can program the EPCS64 device in one of two ways:

- In active serial mode
- Using the Serial FlashLoader (SFL)

Active Serial Mode

To program the on-board serial flash in active serial mode using a POF file, perform the following steps:

1. Ensure that EPCS64 device select switch (J3) is in the *Down* position (towards the back of the board).

If the switch (J3) is not in the *Down* position (i.e., toward the back of the board), you will erase the factory-programmed default design. Refer to [“Restoring the Factory-Programmed Design” on page 2–18](#).

2. Connect the USB-Blaster to the active serial interface connector (J11).
3. Start the Quartus II software.
4. Choose **Programmer** (Tools menu).
5. In the **Mode** list, select **Active Serial Programming**.
6. Click **Add File** and browse to the POF you want to program to flash memory. Click **Open**.
7. Click **Start** to program the EPCS64 device.

JTAG Programming via Serial Flash Loader

The SFL solution is an in-system programming scheme that streamlines the process of programming the EPCS64 device. For example, you do not need a separate programming interface to configure the Cyclone II device and program the serial configuration device. However, the SFL solution is slower because you need to configure the Cyclone II device before programming the serial configuration device.

To configure the Cyclone II device via the SFL solution, you need to:

- Bridge the JTAG interface with the active serial memory interface (ASMI) block of the Cyclone II device
- Convert the SOF to a JIC file



For information about converting the SOF to a JIC file, and step-by-step instructions for the SFL solution, refer to *Application Note 370: Using the Serial FlashLoader with the Quartus II Software*.

Restoring the Factory-Programmed Design

To restore the factory-programmed design stored in flash memory, program the **cpd_top_factory.pof** configuration file to the factory programmed section of flash memory using the Quartus II software. Before programming the factory-default file to flash, the switch (J3) must be in the *Up* position.

The factory-programmed file can be located in `<path>/cycloneii_pci_kit-v1.0.0/configuration_files` directory.

Using the Board as a Prototyping Platform

You can use the Cyclone II EP2C35 PCI development board as a prototyping platform. To prototype your design with the board, follow these steps:

1. *Create your design*—You can use the supplied reference design as a basis for your design, or create a new design according to your requirements.
2. *Simulate your design*—Extensive simulation of your design minimizes the hardware debugging effort.
3. *Compile your design in the Quartus II software*—You can use the Quartus II software to compile your design, verify timing, and generate programming files.
4. *Configure your Cyclone II device*—Configure the Cyclone II device with your programming file using the configuration method of your choice.
5. *Test your design in hardware*—This testing typically requires software and a driver that are built specifically for your application. You can use the open-source kit application as a starting point to develop your own hardware test program.

The prototyping walkthrough uses the reference design included with the *PCI Development Kit, Cyclone II Edition CD-ROM* and requires the following components:

- The Quartus II software
- *MegaCore IP Library CD-ROM, version 5.0*

Altera recommends that you license the `pci_mt64` and DDR2 SDRAM Controller MegaCore function. If you do not have these licenses, you can compile the project, perform timing analysis, and generate a time-limited SOF file from within the Quartus II software enabled by the OpenCore® Plus feature. However, you cannot generate unrestricted programming files.



Refer to “Software Installation” on page 2–4 and “Setting Up Licensing” on page 2–6 for more information.

To simulate your design using this walkthrough, you need the Model Technology ModelSim simulator version 6.0 or later, which is not included with the kit.



The reference design’s DDR2 memory simulation model is only available in Verilog HDL, and the reference design is in VHDL. Mixed VHDL and Verilog HDL simulation may require additional ModelSim licenses.

Prototyping Walkthrough Files

The prototyping walkthrough uses reference design source files, simulation files, and compilation files that you copy from the *PCI Development Kit, Cyclone II Edition CD-ROM* installation directory to your working directory.

Table 2–2 describes the reference design source files used in the walkthrough. These files are located in the `<path>\cycloneii_pci_kit-v1.0.0\reference_design\vhdl` directory.

Table 2–2. Prototyping Walkthrough Reference Design Source Files (Part 1 of 2)

File	Subdirectory (1)	Description
<code>cpd_top.vhd</code>	—	The top-level file that instantiates the PCI and DDR2 SDRAM cores and the local design. Refer to <i>AN 390: PCI-to-DDR2 SDRAM Reference Design</i> for a description of the local design.
<code>vhdl_components.vhd</code>	—	The VHDL component instantiation templates.
<code>pci_top.vhd</code>	<code>pci_mt64</code>	<p>This wrapper file implements the <code>pci_mt64</code> core. The file was generated with the PCI Compiler wizard and the following settings:</p> <ul style="list-style-type: none"> • VHDL output file • 64-bit master/target MegaCore function (<code>pci_mt64</code>) • Device ID is set to 0x0009 • Revision ID is set to 0x1 • Subsystem ID is set to 0xE102 • Subsys Vendor ID is set to 0x1172 • BAR0 is a 1-MByte memory • BAR1 is a 64-MByte prefetchable memory • All other parameters are at the default values <p>For more information, refer to the <i>PCI Compiler User Guide</i>.</p>
<code>backend.vhd</code>	<code>pci_local</code>	The top level of the local PCI interface.

Table 2–2. Prototyping Walkthrough Reference Design Source Files (Part 2 of 2)

File	Subdirectory (1)	Description
cnten.vhd datapath_fifo.vhd dma.vhd dma_reg.vhd dma_sm.vhd last_gen.vhd mstr_cntrl.vhd mstr_fifo_cntrl.vhd mstr_perf.vhd targ_cntrl.vhd targ_fifo_cntrl.vhd targ_perf.vhd	pci_local	These files are the local PCI interface. They implement target and master state machine data path controllers and FIFO buffers.
fifo_128x32.vhd fifo_128x4.vd fifo_128x64.vhd	pci_local	The reference design uses library of parameterized module (LPM) functions, which are generated using the Quartus II software. For more information on generating LPM functions, refer to Quartus II Help.
ddr_top.vhd	ddr_cntrl	This wrapper file implements the DDR2 SDRAM Controller Compiler. The file is generated with the DDR2 SDRAM Controller. Refer to the <i>DDR & DDR2 SDRAM Controller Compiler User Guide</i> for instructions on how to use the wizard.
ddr_intf.vhd	ddr_intf	The top level of the PCI-DDR2 SDRAM interface.
adr_gen.vhd clk_sync.vhd cntrl_intf.vhd mr_sm.vhd mw_sm.vhd tr_sm.vhd tw_sm.vhd	ddr_intf	These files are the PCI-DDR2 SDRAM interface. They implement target and master state machines, address logic, and clock synchronizers.

Note to [Table 2–2](#):

(1) Indicates the subdirectory of the `<path>\cycloneii_pci_kit_v-1.0.0\reference_design\vhdl` directory.

[Table 2–3](#) describes the simulation files used in the walkthrough. The files are located in the `<path>\cycloneii_pci_kit-v1.0.0\reference_design\sim` directory.

Table 2–3. Prototyping Walkthrough Simulation Files

File	Subdirectory (1)	Description
modelsim.ini	—	The ModelSim initialization settings.
sim.do	—	The ModelSim simulation script.

Table 2–3. Prototyping Walkthrough Simulation Files

File	Subdirectory (1)	Description
wave_cycloneii_pci2ddr2.do	—	The ModelSim waveform display script.
cpd_top_tb.vhd	—	The testbench wrapper file. This file instantiates the reference design, PCI bus functional model, and DDR2 SDRAM model.
trgt_tranx_mem_init.dat	—	The PCI target transactor memory initialization data file. This file provides initialization data for the PCI target transactor.
trgt_tranx.vhd	pci_bfm	The PCI target transactor. This file uses sequences of PCI host transfers to write and read the DDR2 SDRAM and flash memory controllers.
mstr_tranx.vhd	pci_bfm	The PCI master transactor. This file uses sequences of PCI host transfers to program the local DMA controller to issue PCI master reads and writes from the DDR2 SDRAM memory.
arbiter.vhd clk_gen.vhd log.vhd monitor.vhd mstr_pkg.vhd pull_up.vhd	pci_bfm	The PCI bus functional models used for simulation. These files support monitoring, and logging for the PCI bus functional model.
ddr2_mem_model.vhd	ddr_model	The DDR2 SDRAM DIMM module model used for simulation.
ddr2.v	ddr_model	The DDR2 SDRAM memory simulation model placeholder. You must replace this file with the correct simulation model before you can simulate. Refer to the kit readme file for instructions on how to obtain the appropriate DDR2 SDRAM memory model.
vhdl simulation models	altera_lib	The Altera functional simulation models needed for simulation.

Note to **Table 2–3**:

(1) Indicates the subdirectory of the `<path>\cycloneii_pci_kit-v1.0.0\reference_design\sim` directory.

Table 2–4 describes the compilation files used in the walkthrough. The files are located in the `<path>\cycloneii_pci_kit-v1.0.0\reference_design\QuartusII` directory.

Table 2–4. Prototyping Walkthrough Compilation Files

File	Description
cpd_top.qpf	The Quartus II project file.

Table 2–4. Prototyping Walkthrough Compilation Files

File	Description
cpd_top.qsf	The board's constraint file. This is the ready-to-use reference design constraint file, which includes all of the board's pin assignments, MegaCore functions, and device-specific settings.
cpd_top.sof cpd_top.pof	The programming files for the board. The cpd_top.sof file can be loaded into the Cyclone II device with the USB-Blaster cable. The cpd_top.pof can be loaded into the on-board serial flash memory using Quartus II software. These files were generated from the reference design and have exactly the same behavior as the factory-default configuration, except that the value stored in the subsystem ID register is 0xE102 instead of 0xE101.

Prototyping Walkthrough

The walkthrough has the following steps:

- Create your design
- Simulate your design
- Compile your design in the Quartus II software
- Configure your Cyclone II device
- Test your design in hardware

Create Your Design

The walkthrough uses the kit's reference design file. To create a copy of the reference design, follow these steps:

1. Create a working directory and use it to simulate your design with the ModelSim simulator and compile the design with the Quartus II software. These instructions assume your working directory is **c:\pci_prototype_example**.
2. Copy all of the files in the **c:\MegaCore\cycloneii_pci_kit-v1.0.0\reference_design** directory and its subdirectories to your working directory, maintaining the kit's installation directory structure. These instructions assume you installed the *PCI Development Kit, Cyclone II Edition CD-ROM* prototyping source files into the **c:\MegaCore\cycloneii_pci_kit-v1.0.0** directory. [Tables 2–2](#), [2–3](#), and [2–4](#) describe the design source files.

When you are finished with the walkthrough, modify your copy of the reference design in the **c:\pci_prototype_example** directory to create a new design according to your requirements.



When creating your design, you can use the Cyclone II EP2C35 PCI development board's LEDs to indicate design status.

Simulate Your Design

The kit includes a testbench and PCI behavioral models that you use to simulate your design's PCI transactions in a third-party simulation tool.

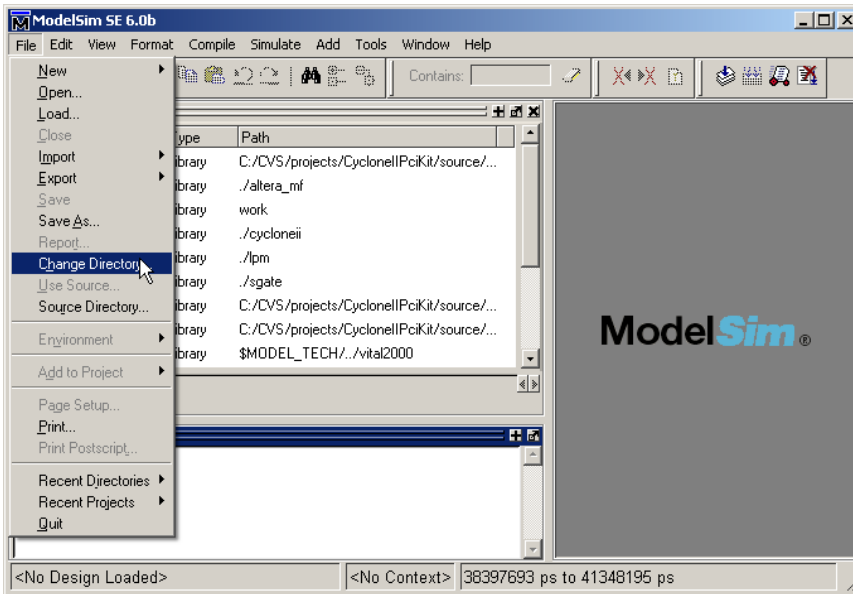


This walkthrough uses the Model Technology ModelSim SE simulator version 6.0 or later, which is not included with the kit.

To simulate the reference design, perform the following steps:

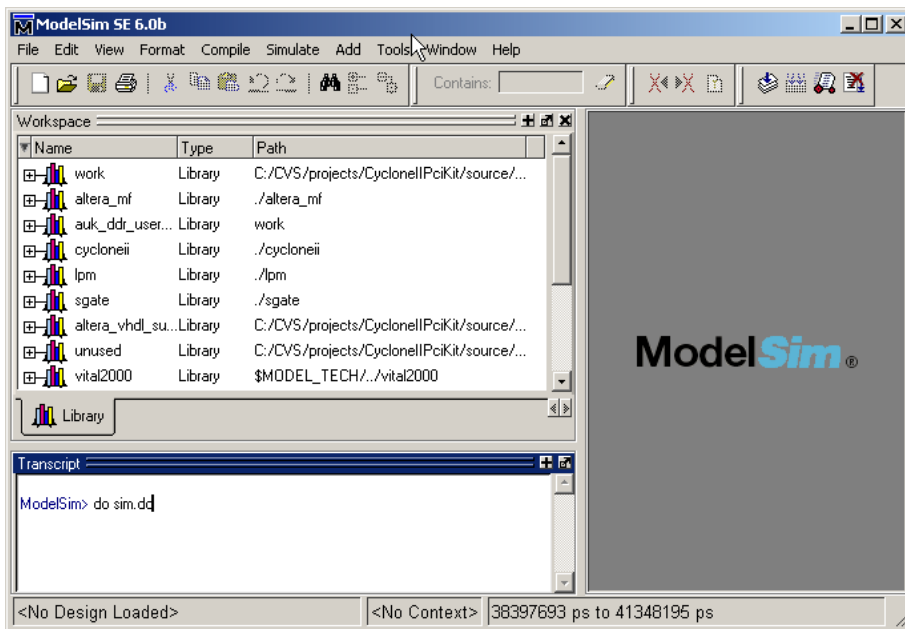
1. Replace the DDR2 SDRAM simulation model file in the `c:\pci_prototype_example\sim\ddr_model` directory with the correct simulation model before you simulate. Refer to the kit's **readme** file for instructions on how to obtain the correct model.
 - a. Download the DDR2 SDRAM simulation model package from the Micron Technology, Inc. web site, www.micron.com.
 - b. Extract the downloaded package and override any existing files in `c:\pci_prototype_example\sim\ddr_model`.
 - c. Edit the downloaded `ddr2.v` and replace "include `ddr2_parameters.v`" with "include `./ddr_model/ddr2_parameters.v`" and then save the edited file.
2. Run the ModelSim software.
3. Choose **Change Directory** (File menu). See [Figure 2-9](#).

Figure 2–9. Simulate the Design in ModelSim



4. Browse to the `c:\pci_prototype_example\sim` directory.
5. In the **Transcript** box, type `do sim.do` at the ModelSim command prompt. See [Figure 2–10](#).

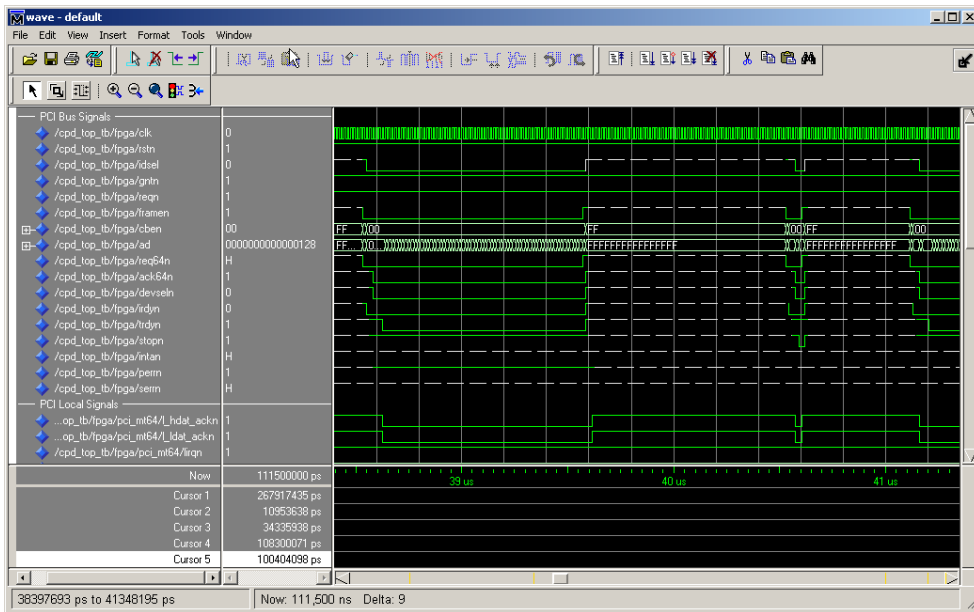
Figure 2–10. ModelSim Transcript Box



ModelSim compiles and simulates the design and displays the resulting waveforms. The waveform window allows you to check the following design signals (see Figure 2–11):

- *PCI Bus Signals*—Examine these waveforms to confirm that the PCI bus functional model is generating PCI host transactions from the **mstr_tranx.vhd** file.
- *pci_mt64 User Interface Signals*—Examine these waveforms to confirm that the PCI-to-DDR2 reference design is responding to the PCI bus transactions.
- *DDR2 Memory Signals*—Examine these waveforms to confirm that the DDR2 SDRAM memory is operating properly.

Figure 2–11. ModelSim Waveform Window



For more information on the reference design, refer to AN 390: *PCI-to-DDR2 Reference Design*.

Compile Your Design in the Quartus II Software

To generate programming files by synthesizing and compiling your design in the Quartus II software, perform the following steps:

1. Run the Quartus II software.
2. Choose **Open Project** (File menu).
3. Browse to the `c:\pci_prototype_example\QuartusII` directory.
4. Select `cpd_top.qpf`.
5. Click **Open** to select the project file.
6. If you did not install the DDR2 SDRAM Controller Compiler in the default directory `c:\MegaCore\ddr_ddr2_sdr-am-v3.2.0`, you must modify the library directory in the Quartus II project.

- a. Choose **Add/Remove Files in Project** (Project menu)
 - Replace all design files pointing to the **C:\MegaCore\ddr_ddr2_sdram-v3.2.0\lib** directory with identical design files from the directory where you install the MegaCore function files.
 - b. Select **User Libraries** under **Category**.
 - c. Type *<DDR2 SDRAM Controller Compiler path>***\lib** in the **Library name** box.
 - d. Click **Add** to add the DDR2 SDRAM library.
 - e. Click **OK**.
7. If you did not install PCI Compiler in the default directory **c:\MegaCore\pci_compiler-v4.0.0**, you must modify the library directory in the Quartus II project.
- a. Choose **Add/Remove Files in Project** (Project menu)
 - b. Select **User Libraries** under **Category**.
 - c. Type *<PCI Compiler path>***\lib** in the **Library name** box.
 - d. Click **Add** to add the PCI Compiler library.
 - e. Click **OK**.
8. If you are prototyping your own design (i.e., other than the Altera-provided reference design), you must add the following constraints to your design:
- a. Locations, I/O standards, and timing for the PCI MegaCore functions. Refer to the *PCI Compiler User Guide* for instructions on how to annotate the PCI constraint file into your project.
 - b. Locations, I/O standards, and timing for the DDR2 SDRAM Controller MegaCore function. Refer to the *DDR & DDR2 SDRAM Controller Compiler User Guide* for instructions on how to annotate the DDR2 SDRAM Controller constraint file into your project.
 - c. Locations, I/O standards, timing, and general constraints for the remainder of your design, including, but not limited to dip switch settings, push buttons, and clock oscillators.



To ensure that the prototype pin assignments match the board's pin assignments, you should use the provided constraint file (**cpd_top.qsf**) for pin-out reference. Unless otherwise indicated, you should assume that constraint files not shipped with the kit contain different pin assignments from that required by the board.

For this walkthrough, the steps are omitted and you use the provided ready-to-use constraint files **cpd_top.qsf** in the **c:\pci_prototype_example\QuartusII** directory.



The kit's constraint file (**.qsf**) is made for the PCI Compiler version 4.0.0 and the DDR & DDR2 SDRAM Controller Compiler version 3.2.0 using the Quartus II software version 5.0; therefore, the constraint files should be used with the same MegaCore function and software versions. If you use a different MegaCore function or software version, you may experience timing violations or compilation errors.

9. Choose **Start Compilation** (Processing menu) to synthesize and compile the project and generate the **cpd_top.sof** and **cdp_top.pof** files.

If you do not have licenses for the **pci_mt64** MegaCore function or the DDR & DDR2 SDRAM Controller Compiler, you will not be able to generate unrestricted programming files. In this case, use the file **cpd_top.pof** from your working directory to complete the steps in this walkthrough.

10. Choose **Compilation Report** (Processing menu) and expand the **Timing Analyzer** section to check the timing of your compiled design.
11. Choose **Compilation Report** (Processing menu) and select **Pin-Out File** to make sure all of your pins are assigned properly.

Configure Your Cyclone II Device

You can configure the Cyclone II device using either the SOF or POF created in the previous section via the JTAG interface or by programming the on-board serial flash memory. Follow the steps outlined in [“Configuring the Cyclone II Device” on page 2–15](#), depending on your preferred configuration method. The next step in this walkthrough assumes that you have programmed your POF into the serial flash memory and you are ready to test your new configuration in hardware.

Test Your Design in Hardware

This step is usually application specific. You should run any required tests to ensure that the application's design requirements are met. This walkthrough uses the Cyclone II PCI development kit's demo application to verify that the steps you have previously performed are successful.

To modify the board so that the configuration data is read from the user EPCS64 flash memory (not the default factory-programmed EPCS64 flash memory), complete the following steps.

1. Shut down your PC.
2. Install the board in your PC, and ensure that the J3 switch is in the *Down* position (for access to the user EPCS64 flash memory).
3. Turn on your PC.
4. The operating system might indicate that the new configuration of the Cyclone II EP2C35 PCI development board is new hardware and prompt you to install a driver. Refer to ["Install the Board in Your PC" on page 2–8](#) for instructions on installing the driver using the **Found New Hardware Wizard**.
5. Run the Cyclone II PCI development kit demo application.
6. Confirm that the copy of the reference design you compiled is unique. Click the **Debug** tab and verify that the subsystem ID has the same value as that specified in the **pci_top.vhd** wrapper file.

You have completed all of the steps necessary to prototype your design with the Cyclone II EP2C35 PCI development board. Perform additional testing and verification as needed for your design.

