



APEX PCI Development Kit

Getting Started

User Guide
November 2001



101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
<http://www.altera.com>

Copyright © 2001 Altera Corporation. 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. All rights reserved.



This user guide provides comprehensive information about the Altera® APEX™ peripheral component interconnect (PCI) Development Kit.

Table 1 shows the user guide revision history.



Go to the following sources for more information:

- See “Features” on page 10 for a complete list of the kit features, including new features in this release.
- Refer to the APEX PCI Development Kit readme file on the *APEX PCI Development Kit* for late-breaking information that is not available in this user guide.

Table 1. User Guide Revision History

Date	Description
October 2001, v2.0	Updated the organization of the document. Added information on the prototyping flow and a prototyping walkthrough.
December 2000, v1.0	First version of user guide.

How to Find Information

- The Adobe Acrobat Find feature allows you to search the contents of a PDF file. Click the binoculars toolbar icon to open the Find dialog box.
- Bookmarks serve as an additional table of contents.
- Thumbnail icons, which provide miniature previews of each page, provide a link to the pages.
- Numerous links, shown in green text, allow you to jump to related information.

How to Contact Altera

For the most up-to-date information about Altera products, go to the Altera world-wide web site at <http://www.altera.com>.

For technical support on this product, go to <http://www.altera.com/mysupport>. For additional information about Altera products, consult the sources shown in Table 2.

Information Type	Access	USA & Canada	All Other Locations
Altera Literature Services	Electronic mail	lit_req@altera.com (1)	lit_req@altera.com (1)
Non-technical customer service	Telephone hotline	(800) SOS-EPLD	(408) 544-7000 (7:30 a.m. to 5:30 p.m. Pacific Time)
	Fax	(408) 544-7606	(408) 544-7606
Technical support	Telephone hotline	(800) 800-EPLD (7:30 a.m. to 5:30 p.m. Pacific Time)	(408) 544-7000 (1) (7:30 a.m. to 5:30 p.m. Pacific Time)
	Fax	(408) 544-6401	(408) 544-6401 (1)
	World-wide web site	http://www.altera.com/mysupport	http://www.altera.com/mysupport
	FTP site	ftp.altera.com	ftp.altera.com
General product information	Telephone	(408) 544-7104	(408) 544-7104 (1)
	World-wide web site	http://www.altera.com	http://www.altera.com

Note:

(1) You can also contact your local Altera sales office or sales representative.

Typographic Conventions

The *APEX PCI Development Kit Getting Started User Guide* uses the typographic conventions shown in [Table 3](#).

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} , \QuartusII directory, d: drive, chiptrip.gdf file.
Bold italic type	Book titles are shown in bold italic type with initial capital letters. Example: <i>1999 Device Data Book</i> .
Italic Type with Initial Capital Letters	Document titles are shown in italic type with initial capital letters. Example: <i>AN 75 (High-Speed Board Design)</i> .
Italic type	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: <i><file name></i> , <i><project name>.pof</i> 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 Quartus II Help topics are shown in quotation marks. Example: “Configuring a FLEX 10K or FLEX 8000 Device with the BitBlaster™ Download Cable.”
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:\quartusII\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.,...	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.



Notes:



About this User Guide	iii
How to Find Information	iii
How to Contact Altera	iv
Typographic Conventions	v
About this Kit	9
General Description	9
Features	10
Documentation	10
Getting Started	11
Before You Begin	11
Development Kit Contents	11
Inspect the Board	11
Hardware Requirements	12
Software Requirements	12
Software Installation	12
Board Installation	13
Kit Application Walkthrough	15
Example 1. PCI Target Write (Demo)	16
Example 2. PCI Master Write (Demo)	17
Example 3. PCI Master Loop (Debug)	18
Example 4. Latency Timer Configuration Register & PCI Master Loop (Debug)	20
Example 5. Address Offset & Target Loop (Debug)	21
Configuring the APEX Device	23
Serial Configuration Using Flash Memory	23
Selecting the Flash Section to Configure the APEX Device	24
Flash Programming Walkthrough	25
JTAG Configuration	27
Prototyping Overview	28
Install PCI Compiler	29
Simulate Your Application Design	29
Synthesize Your Application Design	29
Compile in the Quartus II Software & Generate Programming Files	30
Write the .rbf of Your Design into Flash	30
Hardware Test Your Application Design	31

Prototyping Walkthrough31
 Before You Begin31
 Synthesize the Reference Design33
 Compile the Reference Design in the Quartus II Software34
 Write the .rbf into the Flash Memory35
 Hardware Test the Reference Design35

General Description

The APEX PCI Development Kit provides a complete hardware platform so that designers can quickly begin hardware testing and verification. [Figure 1](#) shows the APEX PCI Development Kit contents.

Figure 1. APEX PCI Development Kit



The development kit includes:

- *APEX PCI Development Board*—You can use the 3.3-V PCI development board in a 32- or 64-bit PCI slot. It supports 33- and 66-MHz PCI interfaces as well as PCI-X interfaces. You can use the board as a desktop application with an external power supply. You can also add a PCI mezzanine card (PMC), i.e., daughter card, to the board. Refer to the *APEX PCI Development Board Data Sheet* for more information on the board.

- *Reference Design*—The kit includes a `pci_mt64` reference design that contains a DMA engine, a FIFO interface, and an SDRAM controller. The design is useful for a variety of hardware applications and lets you begin prototyping and verification quickly. For more information on the reference design, refer to *FS 10: pci_mt64 MegaCore Function Reference Design*.
- *APEX PCI Development Kit Application*—The APEX PCI Development Kit Application (hereafter called kit application) provides an interactive platform with which you can perform PCI transactions. You can also use the kit application as a starting point for developing your own custom software.

Features

- Can be used to prototype a wide variety of custom designs, coupled with Altera PCI MegaCore functions
- Allows designers to focus engineering efforts on value-added custom development, reducing time to market
- Provides a prototyping platform for both PCI form factor and standalone designs
- Supports 32- and 64-bit 33- and 66-MHz PCI designs
- Includes a 64-bit 66-MHz hardware verified, open source PCI reference design
- Includes an open source Windows application that can be used as a starting point for custom software applications
- 3.3-V PCI and PCI-X expansion card
 - PCI-BOARD/A4E development board includes an EP20K400EFC672 device that supports 33- and 66-MHz PCI interfaces
 - PCI-BOARD/A10E development board includes an EP20K1000EFC672 device that supports 33-MHz PCI interfaces

Documentation

The APEX PCI Development Kit contains the following documentation:

- *APEX PCI Development Board Data Sheet*—Describes the specifications of the board and how to load design data into the APEX device on the board.
- *PCI MegaCore Function User Guide*—Provides the specifications of Altera PCI MegaCore functions and explains how to use them.
- *FS 10: pci_mt64 MegaCore Function Reference Design*—Describes the reference design that is included with the kit.
- *Configuring PLDs with Flash Memory White Paper*—Describes how to configure APEX devices with flash memory and a MAX 3000A device.
- *APEX PCI Development Kit Configuration Controller Circuit White Paper*—Describes the flash controller program that is configured into the APEX device when the board powers up.

Before You Begin

Before using the kit or installing the software, check the contents of the kit and inspect the board to verify that you received all of the items. If any of the items are missing, contact Altera before you proceed. You should also verify that your PC meets the kit hardware and software requirements.

Development Kit Contents

Table 1 shows the items shipped in the kit.

Table 1. APEX PCI Development Kit Contents		
Contents	Development Kit Ordering Code	
	PCI-BOARD/A4E	PCI-BOARD/A10E
APEX PCI development board with an EP20K400E device	✓	
APEX PCI development board with an EP20K1000E device		✓
PCI Development Kit CD-ROM	✓	✓
ByteBlasterMV download cable	✓	✓
Selector guides for Altera devices, tools, and IP MegaCore functions	✓	✓
Jungo WinDriver device driver CD-ROM (1)	✓	✓

Note:

- (1) The kit includes a CD-ROM containing a free 30-day evaluation of Jungo’s WinDriver device driver. Altera developed the kit application using WinDriver. For more information on the driver, refer to the documentation on the WinDriver CD-ROM.

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.



Refer to the *APEX PCI Development Board Data Sheet*—which is available on the *PCI Development Kit CD-ROM*—for information on the board components and their location.



The board can be damaged without proper anti-static handling. Therefore, you should take anti-static precautions before handling the board.

Hardware Requirements

The APEX PCI development board is a 3.3-V PCI card that should only be used in 3.3-V PCI systems. To use the board in 5.0-V PCI systems, you must use a 5.0-V to 3.3-V PCI extender card to convert the 5.0-V PCI system to a 3.3-V PCI system. Additionally you must have a PC that can accommodate a long PCI form-factor card.



Refer to the APEX PCI Development Kit readme file for a partial listing of extender card suppliers.

Software Requirements

To use the kit application, you must have the Windows 2000, Windows NT version 4.0 Service Pack 5, or Windows 98 operating system on your PC. The kit application provided with the kit has been tested on these systems only.



You must have administrative privileges to install the kit application device driver on Windows 2000 or Windows NT systems.

To create designs for the board, you must have the following software installed on your PC:

- A full version of the Quartus II software version 1.1 service pack 2 or higher (i.e., you cannot use the free web version)
- PCI Compiler version 2.1.0 or higher (you must have purchased a license for the software)

Software Installation

To install the software on the *PCI Development Kit CD-ROM*, perform the following steps.

1. Insert the *PCI Development Kit CD-ROM* into your CD-ROM drive. The installation program begins running.



If the installation program does not run automatically when you insert the CD-ROM into your CD-ROM drive, perform the following steps to run it manually.

- a. Choose **Run** (Windows Start menu).
- b. Type `<CD-ROM drive>:\pcikit.exe` in the **Open** box.
- c. Click **OK**. The installation program opens.

2. Click **Install APEX PCI Development Kit** to begin installation.
3. Follow the on-line instructions to install the APEX PCI Development Kit files, documentation, device driver, and kit application. The default installation directory is `c:\megacore\apex_pci_kit-v2.0.0`. Additionally, the installation program creates icons in **Programs > Altera > APEX PCI Development Kit** (Windows Start menu), which you can use to launch the kit application.
4. When installation finishes, remove the CD-ROM from your CD-ROM drive.

Board Installation

To install the board in your PC, perform the following steps.

1. Shut down and turn off your PC.
2. Open your PC's case.
3. Locate an empty PCI slot.



The slot must be able to accommodate a long PCI form factor.

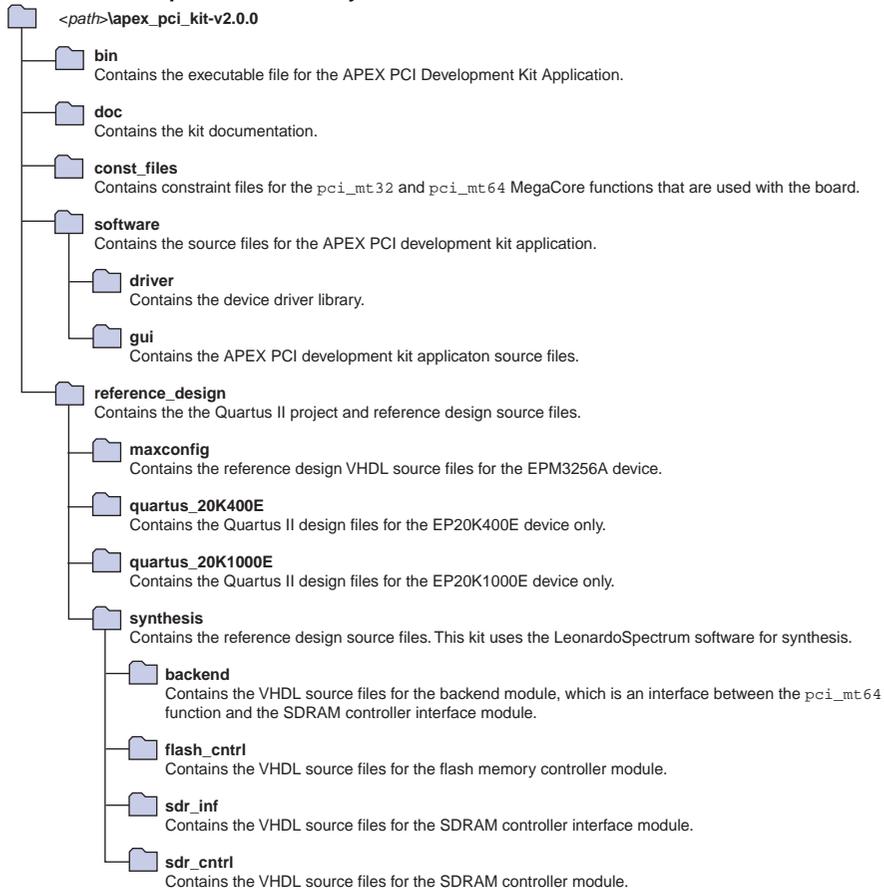
4. Insert the APEX PCI development board into the PCI slot with the RS-232 connector facing the back of the computer. Make sure the board is firmly seated. Secure the board by installing the screw on the bracket on the back of the board.
5. Turn on your PC and observe the board. LEDs 1 through 7 (LED 1 is closest to the back of the PC) show a counting pattern where LED 1 is the least significant bit and LED 7 is the most significant bit. The flashing LEDs indicate that the APEX device has been configured.
6. Close your PC's case.
7. If your PC has the Windows 2000 or Windows 98 operating system, you must install a device driver. After Windows 2000 or Windows 98 finishes loading, the operating system detects the APEX PCI development board as new hardware and prompts you to install a driver.

The APEX PCI Development Kit installation program automatically copies the driver files **windrvr.sys** and **wdpnp.sys** to the `\system32\drivers` directory in your Windows installation directory. Your operating system should detect the driver for the new hardware automatically.

Follow the on-screen instructions to install the driver.

The APEX PCI Development Kit installation program creates the directory structure shown in [Figure 1](#), where *<path>* is the directory in which the APEX PCI Development Kit is installed.

Figure 1. APEX PCI Development Kit Directory Structure



Kit Application Walkthrough

This section explains how to use the kit application to perform PCI transactions, including:

- Example 1. PCI Target Write (Demo)
- Example 2. PCI Master Write (Demo)
- Example 3. PCI Master Loop (Debug)
- Example 4. Latency Timer Configuration Register & PCI Master Loop (Debug)
- Example 5. Address Offset & Target Loop (Debug)

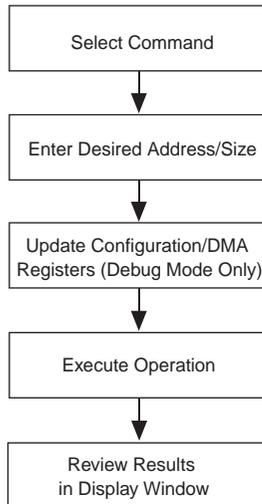
To run the kit application, choose **Programs > Altera > APEX PCI Development Kit** (Windows Start menu). The kit application opens to the **Demo** tab with a PCI target write transaction selected for one iteration of 2,048 bytes of random data.



Refer to the on-line help in the kit application for more information on kit application options and menus.

Figure 2 describes the flow when executing PCI transactions.

Figure 2. Example PCI Transaction Flow Using the Kit Application

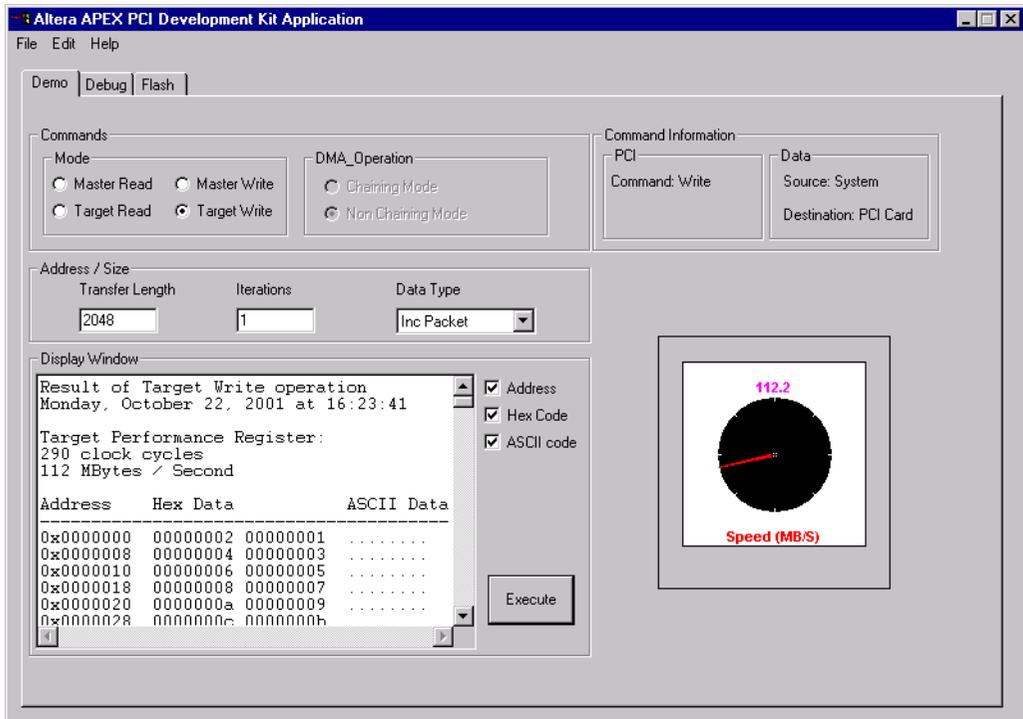


Example 1. PCI Target Write (Demo)

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

1. Run the kit application. The application opens to the **Demo** tab. Leave the kit application running for the remaining examples.
2. Choose the **Target Write** command.
3. Keep the default **Address/Size** values, i.e.,
 - **Transfer Length:** 2,048
 - **Iterations:** 1
 - **Data Type:** Inc Packet
4. Click **Execute** to begin operation.
5. Review the results in the **Display Window**.
6. Choose Incrementing from the **Data Type** drop-down list box.
7. Click **Execute**.
8. Review the speed of transaction in the performance meter window.
9. Review the results in the **Display Window**. [Figure 3](#) shows the results.

Figure 3. PCI Target Write (Demo)



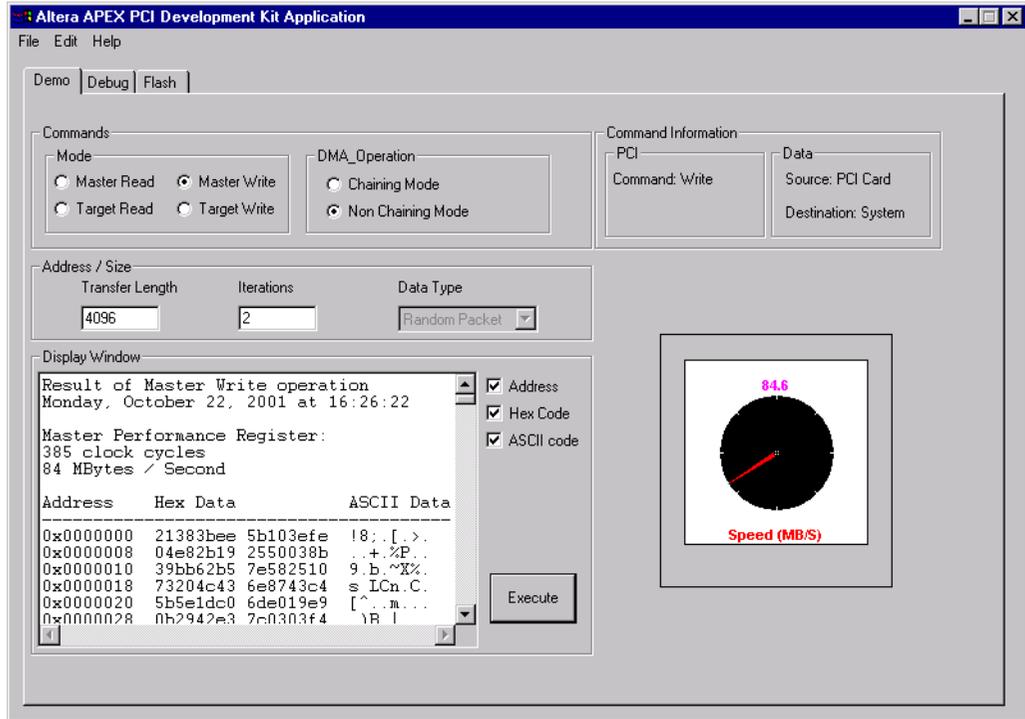
Example 2. PCI Master Write (Demo)

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

1. Select **Master Write** under **Mode**.
2. Select **DMA Non-Chaining** under **DMA Operation**.
3. Make the following **Address/Size** settings:
 - **Transfer Length:** 4,096
 - **Iterations:** 2
4. Click **Execute**.
5. Review the speed of transaction in the performance meter.

- Review the results in the **Display Window**. Figure 4 shows the results.

Figure 4. PCI Master Write (Demo)



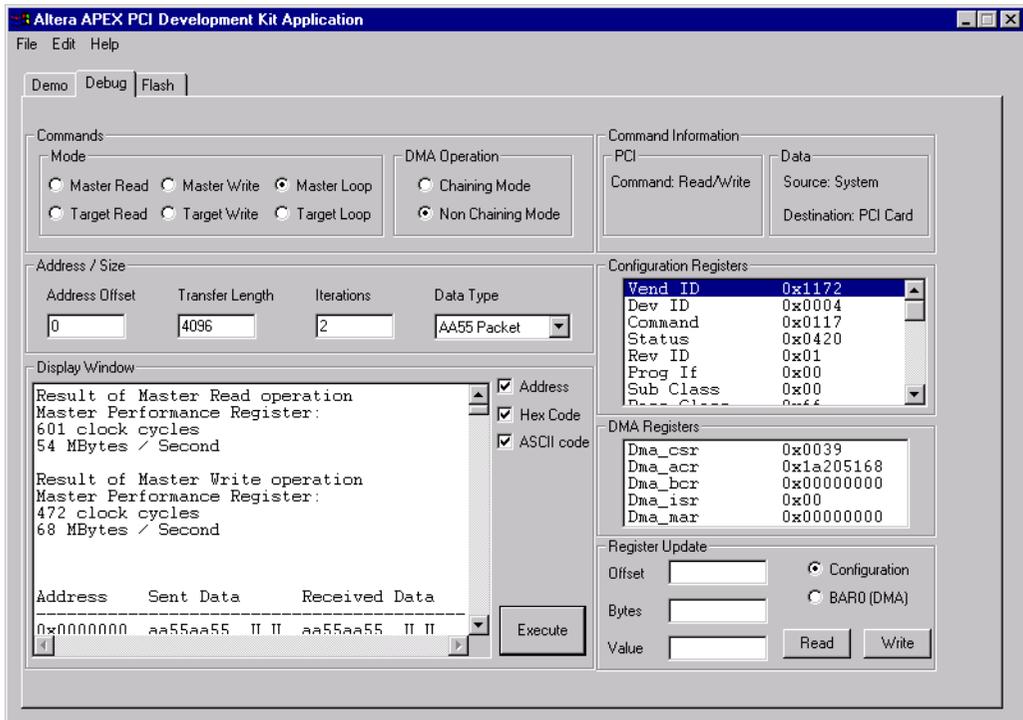
Example 3. PCI Master Loop (Debug)

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

- Click the **Debug** tab.
- Select **Master Loop** under **Mode**.
- Select **DMA Non-Chaining** under **DMA Operation**.
- Use the **Address Offset**, **Transfer Length**, and **Iterations** values that were specified in example 2.

5. Select **AA55 Packet** from the **Data Type** drop-down list box.
6. Click **Execute**.
7. Review the speed of transaction in the **Display Window**.
8. Review the results in the **Display Window**. **Figure 5** shows the results.

Figure 5. PCI Master Loop (Debug)



Example 4. Latency Timer Configuration Register & PCI Master Loop (Debug)

The operation in example 4 is the same as in example 3. By changing the **Latency Timer** setting, the PCI master device remains a master of the bus for more clock cycles, improving the throughput as the master device bursts more data.

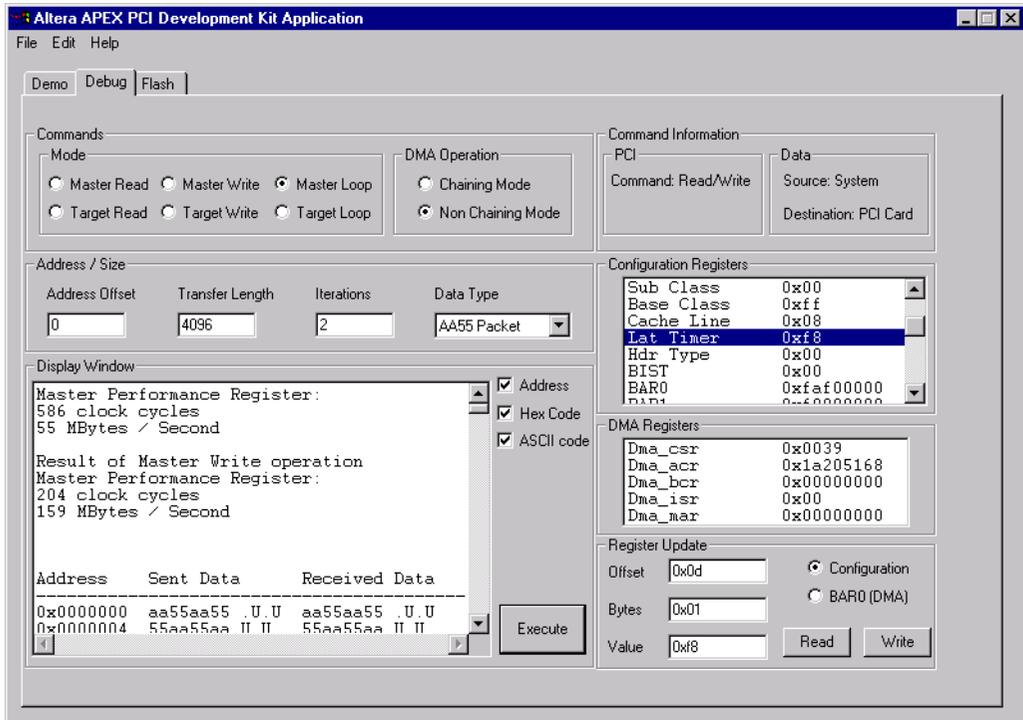
1. Use the **Command** and **Address/Size** settings you specified in example 3.
2. Select **Lat Timer** in the **Configuration Registers** box.



Make note of the default value with which the system programmed the latency timer so that you can return the setting to its default value after running this example.

3. Change the **Lat Timer** value to 0xf8.
4. Click **Write** under **Update Registers**.
5. Click **Execute**.
6. Review the results in the **Display Window**.
7. Compare the results to those of example 3, which used the default **Lat Timer** setting 0x40. See [Figure 6](#).
8. Change the latency timer register back to the default value.

Figure 6. Latency Timer & PCI Master Loop (Debug)

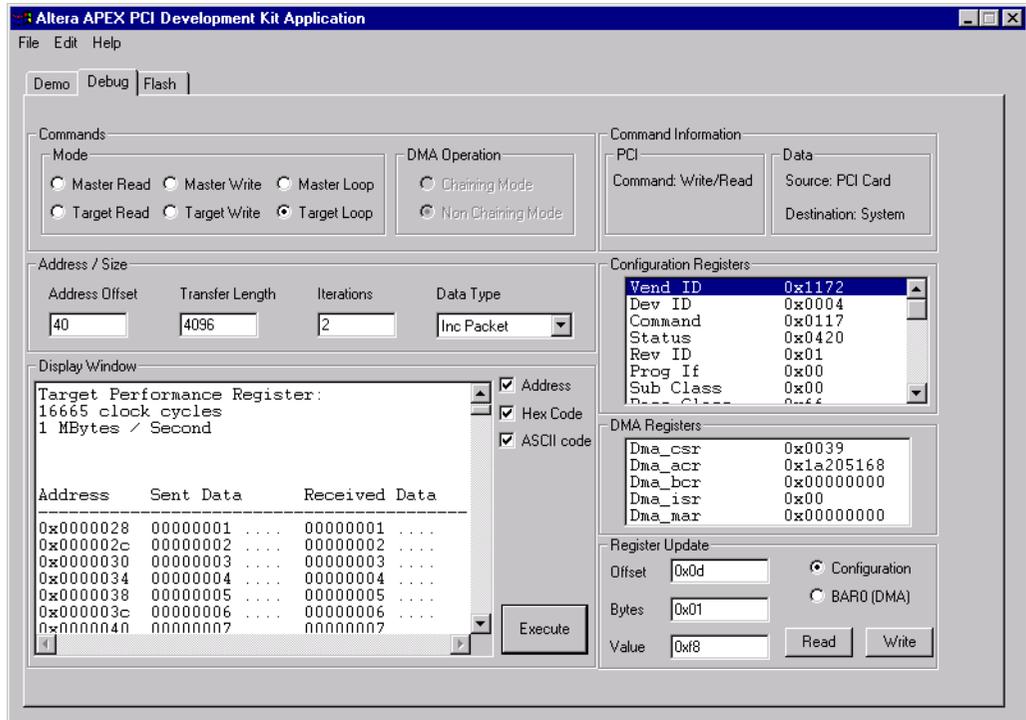


Example 5. Address Offset & Target Loop (Debug)

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

1. Enter 40 in the **Address Offset** box.
2. Select **Target Loop** under **Mode**.
3. Click **Execute**.
4. Review the starting address of the data transaction in the **Display Window**. [Figure 7](#) shows the results.

Figure 7. Address Offset & PCI Target Loop (Debug)



Configuring the APEX Device

The APEX PCI development board supports two configuration methods.

- Serial configuration from an on-board flash memory device
- JTAG configuration through a ByteBlasterMV download cable

Serial Configuration Using Flash Memory

The APEX PCI development board contains an Altera MAX[®] EPM3256A device and an on-board flash memory device. These devices implement a configuration control circuit that allows users to save several configuration files on-board. You can use the flash interface in the kit application to write a Raw Binary File (.rbf) to the on-board flash memory device.



For more information on the flash reference design, refer to the *APEX PCI Development Kit Configuration Controller Circuit White Paper*. For more information on the program that is loaded in the EPM3256A device, refer to the *Configuring PLDs with Flash Memory White Paper*. The documents are available on the *PCI Development Kit CD-ROM*.

The kit application can access several equal-sized sections of the flash memory, which can be used to store APEX device configuration files. The number and size of flash memory sections that are available is dependent on the APEX device that is on the board. The kit application detects the APEX device via the PCI subsystem vendor ID configuration register. This subsystem ID register is a read-only register defined via a parameter in the Altera PCI MegaCore function. [Table 2](#) shows the subsystem ID register's values, the defined device, and the resulting flash memory access.

Table 2. Subsystem ID Registers

Device	Subsystem ID Default Value (1)	Number of Flash Memory Sections	Size of Flash Memory Section (KBytes)
EP20K400E	0xA400	7	512
EP20K1000E	0xAA00	2	1,152

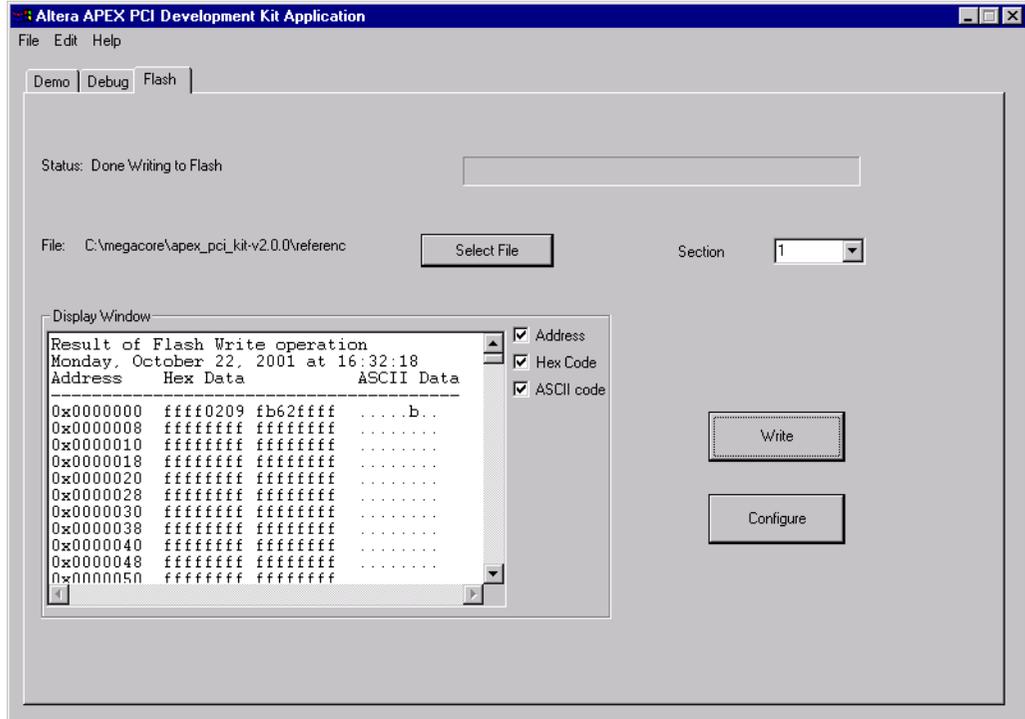
Note:

- (1) The two least significant Hex digits can have values from 00 to FF.

The flash memory includes a default configuration file for the APEX PCI Development Kit reference design. To avoid overwriting the default configuration file, the kit application does not permit you to access the default configuration section (section 0) of the flash memory.

After you write a configuration file to the flash memory using the commands in the **Flash** tab in the kit application, you can click the **Configure** button to reconfigure the APEX device from the newly written flash memory. [Figure 8 on page 24](#) shows the **Flash** programming tab.

Figure 8. Flash Programming Tab



The **Display Window** reports data transferred during on-board flash memory operations.



You can set board jumpers to indicate which flash section to use to configure the APEX device on power up.

Selecting the Flash Section to Configure the APEX Device

After writing APEX device configuration files to the flash memory, you can use dip-switch S1 to select which flash memory section is used to configure the APEX device.



Refer to the *APEX PCI Development Board Data Sheet* for the location of dip-switch S1.

Tables 3 and 4 define the dip-switch settings used to access different sections of the flash memory for the development board that has the EP20K400E device or EP20K1000E device, respectively.

Table 3. APEX EP20K400E Configuration File Dip-Switch Settings

Section	Factory Defined or User Defined	Dip-Switch Settings				Section Size (KBytes)
		def_flash pgm[3]	def_flash pgm[2]	def_flash pgm[1]	def_flash pgm[0]	
0	Factory	On	On	On	On	512
1	User	On	On	Off	On	512
2	User	On	Off	On	On	512
3	User	On	Off	Off	On	512
4	User	Off	On	On	On	512
5	User	Off	On	Off	On	512
6	User	Off	Off	On	On	512
7	User	Off	Off	Off	On	512

Table 4. APEX EP20K1000E Configuration File Dip-Switch Settings

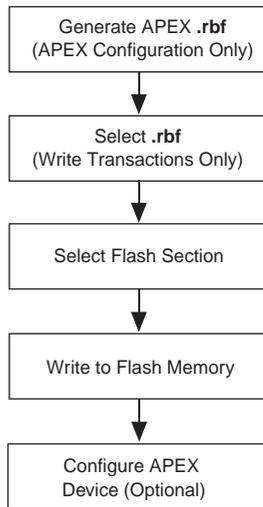
Section	Factory Defined or User Defined	Dip-Switch Settings				Section Size (KBytes)
		def_flash pgm[3]	def_flash pgm[2]	def_flash pgm[1]	def_flash pgm[0]	
0	Factory	On	On	On	On	1,152
1	User	On	Off	On	On	1,152
2	User	Off	On	On	Off	1,152



For more information, see the *APEX PCI Development Board Data Sheet*.

Flash Programming Walkthrough

This section explains how to use the kit application to write a Raw Binary File (.rbf) into the flash. Figure 9 shows the flow for executing flash transactions with the kit application.

Figure 9. Example Flash Programming Flow

To configure the APEX device from flash memory, you must use an **.rbf**. Perform the following steps to write an **.rbf** to section 1 of the flash memory and configure the APEX device from this section using the kit application.



Refer to Quartus II Help for instructions on generating an **.rbf** of your project.

1. Run the kit application.
2. Click the **Flash** tab. Refer back to [Figure 8 on page 24](#), which shows the **Flash** tab interface.
3. Click **Select File** to select the **.rbf** to program into the flash. This walkthrough uses the **.rbf** for the APEX PCI reference design as an example.
4. Browse to the `<installation path>\apex_pci_kit-v2.0.0\reference_design\quartus_20K400E` directory for the EP20K400E board or the `<installation path>\apex_pci_kit-v2.0.0\reference_design\quartus_20K1000E` directory for the EP20K1000C board.
5. Select the file **apex_brd_example.rbf**.
6. Click **Open**.

7. Choose **1** from the **Section** drop-down list box. The **.rbf** will be written into this flash memory section.
8. Click **Write**. The write progress is displayed next to **Status**. The **Display Window** shows the data written into the flash memory.
9. The APEX device can be configured from section 1 of the flash memory in one of the following two ways:
 - Click **Configure**. You must perform a soft reboot of your PC for the configuration to take effect.



Some systems do not re-enumerate the PCI bus during a soft reboot. If you have such system, you cannot use this reboot method. If, after you perform a soft reboot, the APEX PCI board is not recognized when you launch the kit application, the PCI bus was not re-enumerated.

- Modify the APEX PCI board dip-switch settings (S1) to make section 1 the default configuration at power up. See [“Selecting the Flash Section to Configure the APEX Device”](#) on page 24 for details.

JTAG Configuration

All components on the APEX PCI development board that support JTAG boundary-scan testing (BST) are connected in a JTAG chain. You can use the JTAG chain to either program/configure the programmable logic devices on the board or for BST of the connections between board components. You can generate programming files to program/configure Altera devices in a JTAG chain using the Quartus II software.



Refer to Quartus II Help for more information on JTAG chain setup.

The following board components are included in the JTAG chain:

- APEX 20K device
- MAX EPM3256A device
- Two EPC4 devices
- PMC headers

The APEX PCI development board has special jumpers to configure the JTAG chain, depending on the devices installed on the board. The jumper options allow you to bypass—or include—some of the JTAG components in the chain.



Refer to the *APEX PCI Development Board Data Sheet* for the location of the jumpers and information on the JTAG chain circuit.

Table 5 shows the jumper settings that you must use to configure the APEX device via the JTAG interface.

<i>Table 5. Jumper Settings for JTAG Options</i>		
Connect Pin Numbers	State	Function
1-2	Off	The EPM3256A device is installed on the board and is included in the chain. (1)
3-4	On	The EPC4 device #1 is not installed; therefore, it is bypassed.
5-6	On	The EPC4 device #0 is not installed; therefore, it is bypassed.
7-8	On	The PMC is not installed; therefore, it is bypassed.

Note:

- (1) The EPM3256A device cannot be bypassed if the device is installed on the board.

Prototyping Overview

This section describes the procedure you must follow to prototype your own design using the kit.



You must have a valid license to compile and generate programming files for designs that include Altera PCI MegaCore functions. The kit does not include licenses for any of the PCI MegaCore functions. You can purchase a license from the Altera web site at <http://www.altera.com/IPmegastore> or from your local Altera sales representative

Prototyping your design involves the following steps:

1. Install PCI Compiler
2. Simulate Your Application Design
3. Synthesize Your Application Design
4. Compile in the Quartus II Software & Generate Programming Files
5. Write the **.rbf** of Your Design into Flash
6. Hardware Test Your Application Design

Install PCI Compiler

If you have not already done so, install the PCI compiler version 2.1.0 from the *PCI Development Kit CD-ROM*.



For more information on the PCI Compiler, refer to:

- *PCI Compiler Data Sheet*
- *PCI MegaCore Function User Guide*

Simulate Your Application Design

Use the PCI behavioral models that are provided with the PCI Compiler version 2.1.0 to simulate the PCI transactions of your application design in a third-party simulation tool. Extensive simulation of your application design minimizes the hardware debugging effort.



For more information on simulating in third-party tools, refer to:

- *AN 169: Simulating the PCI Behavioral Models*
- *Simulating Visual IP Models with the ModelSim Simulator for PCs White Paper*
- *Simulating Visual IP Models with the NC-Verilog, Verilog-XL, VCS, or ModelSim (UNIX) Simulators White Paper*

Synthesize Your Application Design

You can synthesize your design in a third-party synthesis tool; the Altera PCI MegaCore function is treated as a black box. Refer to the following tips as you are building your design:

- If you plan to use the flash memory for configuration, Altera recommends that you include the flash programming circuit from the PCI reference design in your application design. This circuit lets you easily update the flash memory without reverting back to default configuration settings. Additionally, you should not alter the contents of section 0 of the flash memory so that you can use the default configuration if needed. The flash program works together with the APEX PCI Development Kit application; therefore, you must make the appropriate changes in your application to account for flash programming.
- Consider using the on-board LEDs to indicate the health of the design. For example, define one LED to indicate that the device was configured, another LED for the SDRAM initialization, etc.

- Before generating an EDIF Input File (**.edf**) or Verilog Quartus Mapping File (**.vqm**) in a third-party synthesis tool, you must turn off the **Insert I/O Pads** option. Refer to the help for your third-party synthesis tool for more information about this option.

Compile in the Quartus II Software & Generate Programming Files

Compile the **.vqm** or **.edf** generated from the third-party synthesis tool in the Quartus II software to generate an SRAM Object File (**.sof**) and **.rbf** for your project.



Refer to Quartus II Help for more information on generating a **.sof** and **.rbf**.

To meet PCI timing, Altera provides constraint files for `pci_mt64` and `pci_mt32` MegaCore functions with the APEX PCI Development Kit. The files are installed into the `<path>\apex_pci_kit-v2.0.0\const_files` directory. Refer to the *PCI MegaCore Function User Guide* for instructions on how to annotate the PCI constraint files into your project.



The constraint files provided with the APEX PCI Development Kit are made for the PCI compiler version 2.1.0 using the Quartus II software version 1.1 service pack 2 or higher and can only be used with these versions of the products. If you use a different version of the PCI compiler or Quartus II software, you may receive timing violations on the PCI signals.

Additionally, the pin locations in the constraint files provided with the kit are specific to the APEX PCI development board; therefore, you should use the constraint file provided with the APEX PCI Development Kit for your own board design.

Write the .rbf of Your Design into Flash

You can use the reference design and the APEX PCI Development Kit application to write an **.rbf** into any available section of the flash memory.



For more information on writing your design into the flash memory and configuring the APEX device, refer to [“Serial Configuration Using Flash Memory”](#) on page 23.

Prototyping Walkthrough

Hardware Test Your Application Design

You can use the open source APEX PCI Development Kit application as a starting point to develop your own program. You can also modify the kit application to perform hardware testing of your application design.

This section uses the reference design included with the APEX PCI Development Kit to illustrate the steps needed to prototype your own design. The following software should be installed on your PC:

- APEX PCI Development Kit version 2.0.0
- PCI Compiler version 2.1.0
- Quartus II software version 1.1 service pack 2 or higher
- LeonardoSpectrum-Altera version 2001.1d or higher (optional)



You can use a different synthesis tool; however, this walkthrough uses LeonardoSpectrum-Altera.

Additionally, Altera recommends that you have a license for the `pci_mt64` MegaCore function. If you do not have a license, you cannot generate programming files; however, you can follow the walkthrough to compile the project in the Quartus II software and perform timing analysis.

The walkthrough involves the following procedures:

- Before You Begin
- Synthesize the Reference Design
- Compile the Reference Design in the Quartus II Software
- Write the `.rbf` into Flash Memory
- Hardware Test the Reference Design

Before You Begin

Before you begin the walkthrough, perform the following steps:

1. Create a working directory for the walkthrough. You will use the directory to compile the reference design project in the Quartus II software version 1.1 service pack 2. These instructions use `c:\pci_prototype_example`.
2. Copy all of the files from the `<path>\apex_pci_kit-v2.0.0\reference_design\quartus_<device>` directory (where `<device>` is the APEX device installed on your board) to your working directory so that you do not change the original source files while you perform the walkthrough.

Table 6 describes the Quartus II files used in the walkthrough. The files are located in the `<path>\apex_pci_kit-v2.0.0\reference_design\quartus_<device>` directory (where `<device>` is the APEX device installed on your board).

Table 6. Quartus II Files Used in Walkthrough	
File	Description
my_pci.vhd	<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>) ■ BAR0 is memory mapped with a 1-MByte address range ■ BAR1 is memory mapped with a 16-MByte address range ■ Subsystem ID is set to <code>A402</code> for the EP20K400E device and <code>AA02</code> for the EP20K1000E device ■ All other parameters are at the default values <p>Refer to the PCI MegaCore Function User Guide for instructions on how to use the wizard.</p>
my_cntr.vhd ben_fifo_128x8.vhd dma_fifo_256x32.vhd p2s_fifo_128x32.vhd s2p_fifo_128x64.vhd	<p>These LPM functions—used by the reference design—were generated using the Quartus II software. Refer to Quartus II Help for more information on generating LPM functions. The files are treated as black boxes during synthesis. Refer to the reference design source code for more information on these files.</p>
apex_pll.vhd	<p>This file is used to source the clock to the SDRAM modules. It is treated as a black box during synthesis.</p>
apex_brd.csf apex_brd.esf	<p>Constraint files for the board. These files are generated using the PCI wizard. Generic constraint Tcl scripts are located in the <code><path>\apex_pci_kit-v2.0.0\const_files</code> directory. Refer to the PCI MegaCore Function User Guide for information on how to annotate the PCI constraint files into your project.</p>
apex_brd_example.sof apex_brd_example.rbf	<p>The programming files for the board.</p>

Table 7 describes the synthesis files used in the walkthrough. The files are located in the directory `<path>\apex_pci_kit-v2.0.0\reference_design\synthesis`.

File/Directory	Description
apex_brd.vhd	This top-level project file instantiates my_pci.vhd and all of the reference design modules.
backend flash_cntrl (flash controller) sdr_inf (SDRAM interface) sdr_cntrl (SDRAM controller)	These sub-directories contain the reference design source files. For more information on these modules, refer to <i>FS10: pci_mt64 MegaCore function Reference Design</i> and the <i>APEX PCI Development Kit Configuration Controller Circuit White Paper</i> .

To execute the walkthrough, perform the steps in the following sections.

Synthesize the Reference Design

To synthesize the design, perform the following steps.

1. Run the LeonardoSpectrum-Altera software.
2. Choose **FlowTabs** (Tools menu).
3. Choose the device on your board in the **Technology** tab:
 - **Device family:** APEX 20KE
 - **Device:** EP20K400EFC672 or EP20K1000EFC672
4. Click the **Input** tab.
5. Enter the walkthrough working directory, `c:\pci_prototype_example`, in the **Working Directory** box.
6. Click **Open files**.
7. Select all of the files in the `<path>\apex_pci_kit-v2.0.0\reference_design\synthesis\backend` directory.
8. Click **Open**.
9. Perform steps 6 through 8 for the **flash_cntrl**, **sdr_inf**, and **sdr_cntrl** directories, which are located in the `<path>\apex_pci_kit-v2.0.0\reference_design\synthesis` directory.

10. Select the top-level file `<path>\apex_pci_kit-v2.0.0\reference_design\synthesis\apex_brd.vhd` and click **Open**.
11. Click the **Optimize** tab.
12. Turn off the **Add I/O pads** option.
13. Click the **Output** tab.
14. Make sure that `c:\pci_prototype_example\apex_brd.edf` file is listed in the **Filename** box.
15. Leave all other options at the defaults.
16. Click **Run Flow** to generate the `apex_brd.edf` file.

Compile the Reference Design in the Quartus II Software

After synthesis, use the following procedure to generate programming files by compiling the design in the Quartus II software.

1. Run the Quartus II software.
2. Choose **Open** (File menu).
3. Browse to the `c:\pci_prototype_example` directory.
4. Select `apex_brd.edf`.
5. Click **Open**.
6. Create a Quartus II project.
 - a. Choose **New Project Wizard** (File menu).
 - b. Click **Next** in the introduction if you have not previously turned it off.
 - c. Type `c:\pci_prototype_example` as the working directory.
 - d. Type `apex_brd` as the project.
 - e. Click **Next**.
 - f. Click **No** if you receive a message asking if you want to select a different top-level design entity name.

- g. Click **User Library Pathnames**.
 - h. Type `<path>\PCI_Compiler-v2.1.0\lib` in the **Library name** box.
 - i. Click **Add**.
 - j. Click **OK**.
 - k. Click **Next**.
 - l. Click **Finish** to create the Quartus II project `apex_brd`.
7. Set the EDA tool settings to LeonardoSpectrum.
 - a. Choose **EDA Tool Settings** (Project menu).
 - b. Choose **Leonardo Spectrum** in the **Design entry/synthesis tool** settings drop-down list box.
 - c. Click **OK**.
 8. Choose **Start Compilation** (Processing menu) or use the keystrokes Ctrl+L to compile the project and generate the `apex_brd.sof` and `apex_brd.rbf` files. If you do not have a valid license for the `pci_mt64` MegaCore function, you will not be able to generate programming files. In this case, use the file `apex_brd_example.rbf` from the walkthrough working directory to complete the steps in this walkthrough.

Write the .rbf into the Flash Memory

To write the `.rbf` into section 1 of the flash memory, perform the steps in [“Write the .rbf of Your Design into Flash” on page 30](#).



When plugged into a PCI slot, the board configures from the factory-defined flash section. You can note the subsystem ID of the reference design in the kit application's **Debug** tab.

Hardware Test the Reference Design

The following steps describe how to perform hardware testing.

1. Shut down your PC.
2. Remove the APEX PCI development board from your PC.

3. Change the board's dip-switch settings to configure from section 1 of the flash memory.
4. Reinstall the APEX PCI development board in your PC.
5. Reboot the PC.
6. Run the kit application.
7. Click the **Debug** tab and verify that the subsystem ID has the same value as that specified in the `my_pci.vhd` wrapper file.

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