

# **LogiCORE™ Endpoint PIPE v1.7 for PCI Express®**

## **Getting Started Guide**

UG168 May 17, 2007





Xilinx is disclosing this Specification to you solely for use in the development of designs to operate on Xilinx FPGAs. Except as stated herein, none of the Specification may be copied, reproduced, distributed, republished, downloaded, displayed, posted, or transmitted in any form or by any means including, but not limited to, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of Xilinx. Any unauthorized use of this Specification may violate copyright laws, trademark laws, the laws of privacy and publicity, and communications regulations and statutes.

Xilinx does not assume any liability arising out of the application or use of the Specification; nor does Xilinx convey any license under its patents, copyrights, or any rights of others. You are responsible for obtaining any rights you may require for your use or implementation of the Specification. Xilinx reserves the right to make changes, at any time, to the Specification as deemed desirable in the sole discretion of Xilinx. Xilinx assumes no obligation to correct any errors contained herein or to advise you of any correction if such be made. Xilinx will not assume any liability for the accuracy or correctness of any engineering or technical support or assistance provided to you in connection with the Specification.

THE SPECIFICATION IS PROVIDED "AS IS" WITH ALL FAULTS, AND THE ENTIRE RISK AS TO ITS FUNCTION AND IMPLEMENTATION IS WITH YOU. YOU ACKNOWLEDGE AND AGREE THAT YOU HAVE NOT RELIED ON ANY ORAL OR WRITTEN INFORMATION OR ADVICE, WHETHER GIVEN BY XILINX, OR ITS AGENTS OR EMPLOYEES. XILINX MAKES NO OTHER WARRANTIES, WHETHER EXPRESS, IMPLIED, OR STATUTORY, REGARDING THE SPECIFICATION, INCLUDING ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, AND NONINFRINGEMENT OF THIRD-PARTY RIGHTS.

IN NO EVENT WILL XILINX BE LIABLE FOR ANY CONSEQUENTIAL, INDIRECT, EXEMPLARY, SPECIAL, OR INCIDENTAL DAMAGES, INCLUDING ANY LOST DATA AND LOST PROFITS, ARISING FROM OR RELATING TO YOUR USE OF THE SPECIFICATION, EVEN IF YOU HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THE TOTAL CUMULATIVE LIABILITY OF XILINX IN CONNECTION WITH YOUR USE OF THE SPECIFICATION, WHETHER IN CONTRACT OR TORT OR OTHERWISE, WILL IN NO EVENT EXCEED THE AMOUNT OF FEES PAID BY YOU TO XILINX HEREUNDER FOR USE OF THE SPECIFICATION. YOU ACKNOWLEDGE THAT THE FEES, IF ANY, REFLECT THE ALLOCATION OF RISK SET FORTH IN THIS AGREEMENT AND THAT XILINX WOULD NOT MAKE AVAILABLE THE SPECIFICATION TO YOU WITHOUT THESE LIMITATIONS OF LIABILITY.

The Specification is not designed or intended for use in the development of on-line control equipment in hazardous environments requiring fail-safe controls, such as in the operation of nuclear facilities, aircraft navigation or communications systems, air traffic control, life support, or weapons systems ("High-Risk Applications"). Xilinx specifically disclaims any express or implied warranties of fitness for such High-Risk Applications. You represent that use of the Specification in such High-Risk Applications is fully at your risk.

© 2007 Xilinx, Inc. All rights reserved. XILINX, the Xilinx logo, and other designated brands included herein are trademarks of Xilinx, Inc. All other trademarks are the property of their respective owners.

---

---

## Revision History

The following table shows the revision history for this document.

Date	Version	Revision
04/28/05	1.1	Initial Xilinx release or revision notes.
8/31/05	1.2	Revision for delivery through CORE Generator v7.1i SP3.
01/18/06	1.3	Updated core version to 1.3, Xilinx tools version v8.1i.
7/13/06	1.4	Updated core version to 1.4, Xilinx tools to v8.2i.
9/21/06	1.5	Updated core to version 1.5.
2/15/07	1.6	Updated core to version 1.6; Xilinx tools 9.1i.
5/17/07	1.7	Updated core to version 1.7; updated for PCI-SIG compliance.

# Table of Contents

---

## Preface: About This Guide

Contents .....	5
Additional Resources .....	5
Conventions .....	6
Typographical .....	6
Online Document .....	7

## Chapter 1: Introduction

About the Core .....	9
System Requirements .....	9
Recommended Design Experience .....	9
Additional Core Resources .....	10
Technical Support .....	10
Feedback .....	10
Core .....	10
Document .....	10

## Chapter 2: Licensing the Core

Before you Begin .....	11
License Options .....	11
Simulation Only Evaluation .....	11
Full System Hardware Evaluation .....	11
Full .....	12
Obtaining Your License .....	12
Installing Your License File .....	12
Obtaining the NXP Simulation Models .....	13

## Chapter 3: Quickstart Example Design

Overview .....	15
Simulation Design Overview .....	15
Implementation Design Overview .....	16
Example Design Elements .....	16
Generating the Core .....	16
Simulating the Example Design .....	18
Setting up for Simulation .....	18
Running the Simulation .....	19
Implementing the Example Design .....	20
Directory Structure and File Contents .....	22
<project directory> .....	22
<project directory>/<component name> .....	23
<component name>/doc .....	23
<component name>/example_design .....	23
<component name>/implement .....	24
implement/results .....	24
<component name>/simulation .....	25

simulation/dsport .....	26
simulation/functional .....	26
simulation/tests .....	27
<b>NXP Simulation Models</b> .....	<b>27</b>

## **Appendix: Additional Design Considerations**

<b>Package Constraints</b> .....	<b>29</b>
<b>User Constraints Files</b> .....	<b>29</b>
<b>Wrapper File Usage</b> .....	<b>29</b>

## About This Guide

---

The *LogiCORE Endpoint PIPE v1.7 for PCI Express® Getting Started Guide* provides information about generating an Endpoint PIPE (PHY Interface) for PCI Express (PCIe®) core, customizing and simulating the core using the provided example design, and running the design files through implementation using the Xilinx tools.

### Contents

This guide contains the following chapters:

- [Preface, “About this Guide,”](#) introduces the organization and purpose of this guide, a list of additional resources, and the conventions used in this document.
- [Chapter 1, “Introduction,”](#) describes the core and related information, including recommended design experience, additional resources, technical support, and submitting feedback to Xilinx.
- [Chapter 2, “Licensing the Core,”](#) provides information about licensing the core and acquiring and installing the NXP simulation models.
- [Chapter 3, “Quickstart Example Design,”](#) provides instructions for quickly generating, simulating, and implementing the example design using the demonstration test bench.
- [Appendix, “Additional Design Considerations,”](#) defines additional considerations when implementing the example design.

### Additional Resources

For additional information, go to [www.xilinx.com/support](http://www.xilinx.com/support). The following table lists some of the resources you can access from this website or by using the provided URLs.

Resource	Description/URL
Tutorials	Tutorials covering Xilinx design flows, from design entry to verification and debugging. <a href="http://www.xilinx.com/support/techsup/tutorials/index.htm">www.xilinx.com/support/techsup/tutorials/index.htm</a>
Answer Browser	Database of Xilinx solution records. <a href="http://www.xilinx.com/xlnx/xil_ans_browser.jsp">www.xilinx.com/xlnx/xil_ans_browser.jsp</a>
Data Sheets	Device-specific information on Xilinx device characteristics, including readback, boundary scan, configuration, length count, and debugging. <a href="http://www.xilinx.com/xlnx/xweb/xil_publications_index.jsp">www.xilinx.com/xlnx/xweb/xil_publications_index.jsp</a>

Resource	Description/URL
Problem Solvers	Interactive tools that allow you to troubleshoot your design issues. <a href="http://www.xilinx.com/support/troubleshoot/psolvers.htm">www.xilinx.com/support/troubleshoot/psolvers.htm</a>
Tech Tips	Latest news, design tips, and patch information for the Xilinx design environment. <a href="http://www.xilinx.com/xlnx/xil_tt_home.jsp">www.xilinx.com/xlnx/xil_tt_home.jsp</a>

## Conventions

This document uses the following conventions. An example illustrates each convention.

### Typographical

The following typographical conventions are used in this document:

Convention	Meaning or Use	Example
Courier font	Messages, prompts, and program files that the system displays	speed grade: - 100
<b>Courier bold</b>	Literal commands you enter in a syntactical statement	<b>ngdbuild</b> design_name
<i>Italic font</i>	Variables in a syntax statement for which you must supply values	See the <i>Development System Reference Guide</i> for more information.
	References to other manuals	See the <i>User Guide</i> for details.
	Emphasis in text	If a wire is drawn so that it overlaps the pin of a symbol, the two nets are <i>not</i> connected.
<text in brackets>	User-defined variable for directory names.	<component_name>
Dark Shading	Items that are not supported or reserved	Unsupported feature
Square brackets [ ]	An optional entry or parameter. However, in bus specifications, such as <b>bus[7:0]</b> , they are required.	<b>ngdbuild</b> [option_name] design_name
Braces { }	A list of items from which you must choose one or more	<b>lowpwr</b> ={on off}
Vertical bar	Separates items in a list of choices	<b>lowpwr</b> ={on off}

Convention	Meaning or Use	Example
Vertical ellipsis . . .	Repetitive material that has been omitted	IOB #1: Name = QOUT' IOB #2: Name = CLKIN' . . .
Horizontal ellipsis ...	Omitted repetitive material	<b>allow block</b> block_name loc1 loc2 ... locn;
Notations	The prefix '0x' or the suffix 'h' indicate hexadecimal notation	A read of address 0x00112975 returned 45524943h.
	An '_n' means the signal is active low	usr_teof_n is active low.

## Online Document

The following linking conventions are used in this document:

Convention	Meaning or Use	Example
Blue text	Cross-reference link to a location in the current document	See the section " <a href="#">Additional Resources</a> " for details. See " <a href="#">Title Formats</a> " in <a href="#">Chapter 1</a> for details.
<a href="#">Blue, underlined text</a>	Hyperlink to a website (URL)	Go to <a href="http://www.xilinx.com">www.xilinx.com</a> for the latest speed files.



# Introduction

---

The Endpoint PIPE for PCI Express core is a high-bandwidth, scalable, and reliable serial interconnect building block for use with the Spartan™-3, Spartan-3E, and Spartan-3A FPGAs. This core supports Verilog® and VHDL. The example design described in this guide is provided in Verilog.

This chapter introduces the core and provides related information, including recommended design experience, additional resources, technical support, and submitting feedback to Xilinx.

## About the Core

The Endpoint PIPE for PCIe is a Xilinx CORE Generator™ IP core, included in the latest IP Update on the Xilinx IP Center. For detailed information about the core, see the [Endpoint PIPE for PCI Express](#) product page. See [Chapter 2, “Licensing the Core,”](#) for information about choosing a license option.

## System Requirements

### Windows

- Windows® 2000 Professional with Service Pack 2-4
- Windows XP Professional with Service Pack 1-2

### Solaris/Linux

- Sun Solaris® 9/10
- Red Hat® Enterprise Linux 3.0 (32-bit and 64-bit)

### Software

- ISE™ 9.1i with applicable Service Pack

See the core release notes for the required Service Pack; ISE Service Packs can be downloaded from [www.xilinx.com/xlnx/xil\\_sw\\_updates\\_home.jsp?update=sp](http://www.xilinx.com/xlnx/xil_sw_updates_home.jsp?update=sp).

## Recommended Design Experience

Although the Endpoint PIPE for PCIe core is a fully verified solution, the challenge associated with implementing a complete design varies depending on the configuration and functionality of the application. For best results, previous experience building high performance, pipelined FPGA designs using Xilinx implementation software and User Constraints Files (UCF) is recommended.

## Additional Core Resources

For detailed information and updates about the Endpoint PIPE for PCIe core, see the following documents on the [Endpoint PIPE for PCI Express](#) product page.

**Note:** Use the password issued with the product to access the User Guide and Release Notes in the Product Lounge.

- *LogiCORE Endpoint PIPE for PCI Express Data Sheet*
- *LogiCORE Endpoint PIPE for PCI Express User Guide*
- *LogiCORE Endpoint PIPE for PCI Express Release Notes* file (available from the core directory after generating the core)

Additional information and resources relating to the PCI Express technology are available from the following web sites:

- [PCI Express at PCI-SIG](#)
- [PCI Express PHY Transceiver \(PIPE\)](#)
- [PCI Express Developer's Forum](#)

## Technical Support

For technical support, go to [www.xilinx.com/support](http://www.xilinx.com/support). Questions are routed to a team of engineers with expertise using the Endpoint PIPE for PCI Express core.

Xilinx provides technical support for use of this product as described in the *LogiCORE Endpoint PIPE for PCI Express User Guide* and the *LogiCORE Endpoint PIPE for PCI Express Getting Started Guide*. Xilinx cannot guarantee timing, functionality, or support of this product for designs that do not follow these guidelines.

## Feedback

Xilinx welcomes comments and suggestions about the Endpoint PIPE for PCIe core and the accompanying documentation.

### Core

For comments or suggestions about the Endpoint PIPE for PCIe core, please submit a WebCase from [www.xilinx.com/support](http://www.xilinx.com/support). Be sure to include the following information:

- Product name
- Core version number
- Explanation of your comments

### Document

For comments or suggestions about this document, please submit a WebCase from [www.xilinx.com/support](http://www.xilinx.com/support). Be sure to include the following information:

- Document title
- Document number
- Page number(s) to which your comments refer
- Explanation of your comments

# Licensing the Core

---

This chapter provides instructions licensing for the Endpoint PIPE for PCI Express core, which you must do before using the core in your designs. In addition, information about acquiring and installing the NXP simulation models is provided.

The Endpoint PIPE for PCIe core is provided under the terms of the [Xilinx LogiCORE Site License Agreement](#), which conforms to the terms of the [SignOnce](#) IP License standard defined by the Common License Consortium. Purchase of the core entitles you to technical support and access to updates for one year.

## Before you Begin

This chapter assumes you have installed the core using either the CORE Generator IP Software Update installer or by performing a manual installation after downloading the core from the web. For information about installing the core, see the [Endpoint PIPE for PCI Express](#) product page.

## License Options

The Endpoint PIPE for PCIe core provides three licensing options.

### Simulation Only Evaluation

The Simulation Only Evaluation license is the default license provided with the core. This license lets you generate a simulation model and simulate it using the provided demonstration test bench. Functional simulation is supported by a structural model provided by the CORE Generator.

The Simulation Only Evaluation license lets you assess the core functionality with either the provided example design or alongside your own design and demonstrates the various interfaces on the core in simulation.

### Full System Hardware Evaluation

The Full System Hardware Evaluation license is available at no cost and lets you fully integrate the core into an FPGA design, place and route the design, evaluate timing, and perform back-annotated gate-level simulation of the core using the provided demonstration test bench.

In addition, the license lets you generate a bitstream from the placed and routed design, which can then be downloaded to a supported device and tested in hardware. The core can be tested in the target device for a limited time before *timing out* (ceasing to function) at which time it can be reactivated by reconfiguring the device.

You can obtain the Full System Hardware Evaluation license in one of the following ways, depending on the core:

- By registering on the Xilinx IP Evaluation page and filling out a form to request an automatically generated evaluation license
- By contacting your local Xilinx FAE to request a Full System Hardware Evaluation license key

Click Evaluate on the core's product page for information about how to obtain a Full System Hardware Evaluation license.

## Full

The Full license is provided when you purchase the core; this license provides full access to all core functionality both in simulation and in hardware, including:

- Gate-level functional simulation support
- Back annotated gate-level simulation support
- Full implementation support including place and route and bitstream generation
- Full functionality in the programmed device with no time-outs

## Obtaining Your License

### Obtaining a Full System Hardware Evaluation License

To obtain a Full System Hardware Evaluation license, do the following:

- Navigate to the [Endpoint PIPE for PCI Express](#) product page.
- Click Evaluate; then click Full System Hardware Evaluation.
- Follow the onscreen instructions to obtain a license key and to satisfy an additional requirements associated with the license.

### Obtaining a Full License

To obtain a Full license, you must purchase the core. After purchase, you will receive a letter containing a serial number, which is used to register for access to the *lounge*, a secured area of the product page.

- From the [Endpoint PIPE for PCI Express](#), click Order & Register to register and request access to the lounge.
- Xilinx will review your access request and typically grants access to the lounge in 48 hours. (Contact Xilinx Customer Service if you need faster turnaround.)
- After receiving confirmation of lounge access, click Access Lounge on the [product page](#) and log in.
- Follow the instructions in the lounge to fill out the license request form; then click Submit to automatically generate the license. An e-mail containing the license and installation instructions will be sent to you immediately.

## Installing Your License File

After selecting either the Full System Hardware Evaluation or Full license option, you will receive an email containing instructions for installing your license. In addition, the email provides information about advanced licensing options and technical support.

## Obtaining the NXP Simulation Models

Acquire and install NXP PX1011A-EL1 PHY simulation models, located at [www.standardics.nxp.com/products/pcie/phys/](http://www.standardics.nxp.com/products/pcie/phys/). When simulating your design, if the NXP simulation model has not been installed or is installed in an incorrect location, a message from NXP Semiconductors appears:

Dear Customer,

In order to simulate the Xilinx Endpoint PIPE Core for PCI Express with PXPIPE interface, a simulation model of the NXP PX1011A-EL1 PCI Express PHY is required. This model is the property of NXP (formerly Philips Semiconductors), and is not included with the Xilinx product.

Various simulation models for commonly used tools are available. Please visit our NXP web page, download the no-charge end-user license agreement, sign, scan and email back to [interface.support@nxp.com](mailto:interface.support@nxp.com):

[www.standardics.nxp.com/support/models/px/](http://www.standardics.nxp.com/support/models/px/)

Be sure to include your complete contact information in your correspondence so that we can reply to your request as quickly as possible.

For data sheets, application notes, boundary scan files and other technical support documents, please visit

[www.standardics.nxp.com/products/pcie/phys/](http://www.standardics.nxp.com/products/pcie/phys/)

Thank you,

NXP Semiconductors



## Quickstart Example Design

This chapter provides an overview of the Endpoint PIPE for PCI Express example design and instructions for generating the core. It also covers simulating and implementing the example design using the provided demonstration test bench.

### Overview

The example simulation design consists of two discrete parts:

- The Downstream Port Model, a test bench that generates, consumes, and checks PCIe bus traffic.
- The Programmed Input-Output (PIO) example design, a completer application. The PIO example design responds to PCIe Read and Write requests to its memory space and can be synthesized for testing in hardware.

### Simulation Design Overview

For the example simulation design, transactions are sent from the Downstream Port Model to the core and processed by the PIO example design. [Figure 3-1](#) illustrates the simulation design provided with the core. For more information about the Downstream Port Model, see Appendix C, “Downstream Port Model Test Bench,” in the *LogiCORE Endpoint PIPE for PCI Express User Guide*.

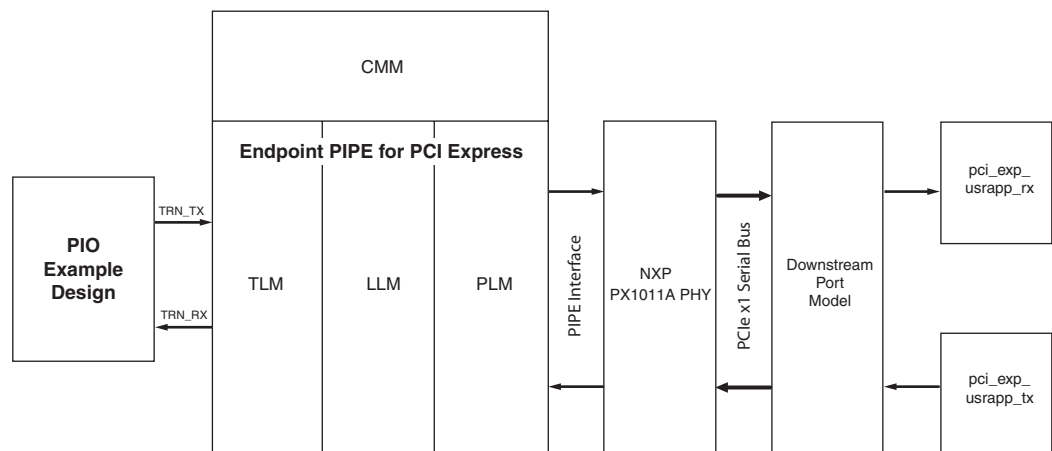


Figure 3-1: Example Simulation Design Block Diagram

## Implementation Design Overview

The example implementation design consists of a simple programmed IO (PIO) example that can accept read and write transactions and respond to requests, as illustrated in [Figure 3-2](#). Source code for this example is provided with the core. For more information about the PIO example design, see Appendix B, “Programmed Input Output Example Design,” in the *LogiCORE Endpoint PIPE for PCI Express User Guide*.

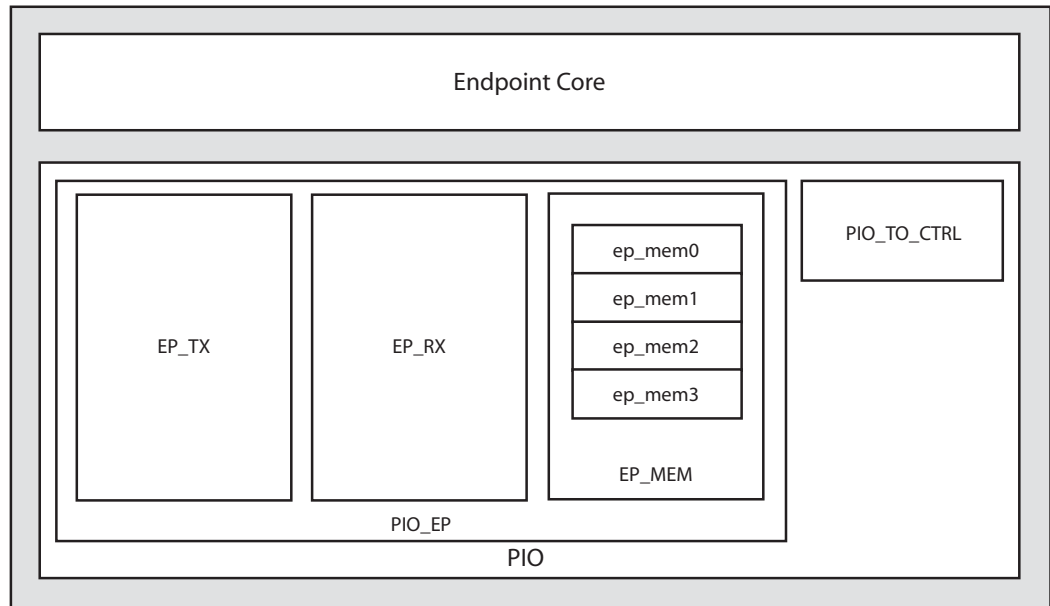


Figure 3-2: Example Implementation Design Block Diagram

## Example Design Elements

The PIO example design elements include the following:

- Core netlists
- Core simulation models
- An example Verilog HDL wrapper (instantiates the core and example design)
- A customizable demonstration test bench to simulate the example design

The example design has been tested with Xilinx ISE v9.1i and the following simulators:

- Cadence™ IUS (VNC) 5.5
- Synopsys® VCS X2006.06
- Mentor Graphics® ModelSim® v6.1e

## Generating the Core

To generate an Endpoint PIPE for PCIe core using the default values in the CORE Generator graphical user Interface (GUI), do the following:

1. Start the CORE Generator.

For help starting and using the CORE Generator, see the Xilinx CORE Generator Guide, available from the [ISE documentation](#) web page.

- Choose File > New Project. The New Project dialog box appears.

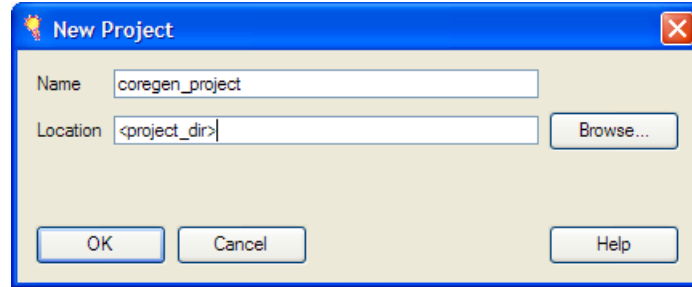


Figure 3-3: New Project Dialog Box

- Enter a project name and location; then click OK. <project\_dir> is used in this example. The Project Options dialog box appears.

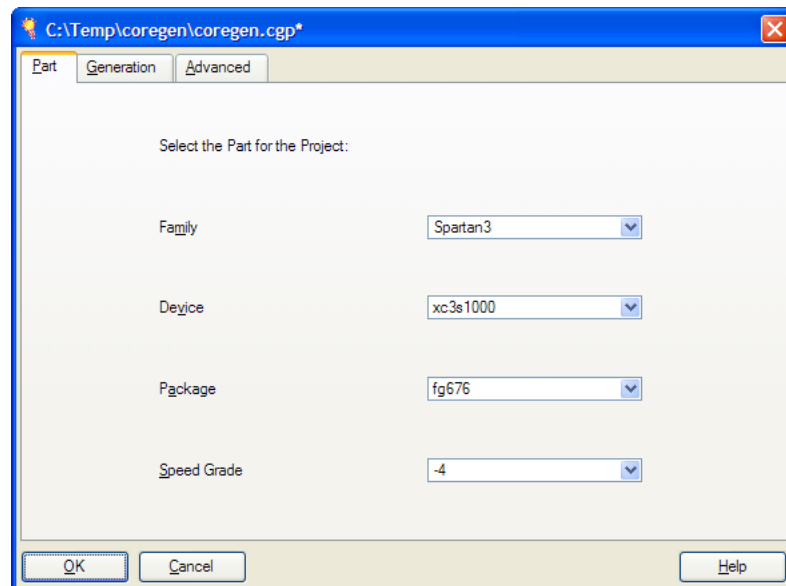


Figure 3-4: Project Options

- Set the project options:

From the Part tab, select the following options:

- **Family:** Spartan3
- **Device:** xc3s1000
- **Package:** fg676
- **Speed Grade:** -4

**Note:** If an unsupported silicon family is selected, the core is not available for customization and is dimmed in the list of cores.

From the Generation tab, select the following parameters:

- **Design Entry:** Select either VHDL or Verilog. (Note that the example design and test bench are provided for Verilog only.)
- **Vendor:** Select Synplicity® or ISE (for XST).

5. Click OK.
6. Locate the Endpoint PIPE for PCI Express core in the core selection tree under Standard Bus Interfaces/PCI Express. Double-click the core name to display the main GUI screen. (Figure 3-5).

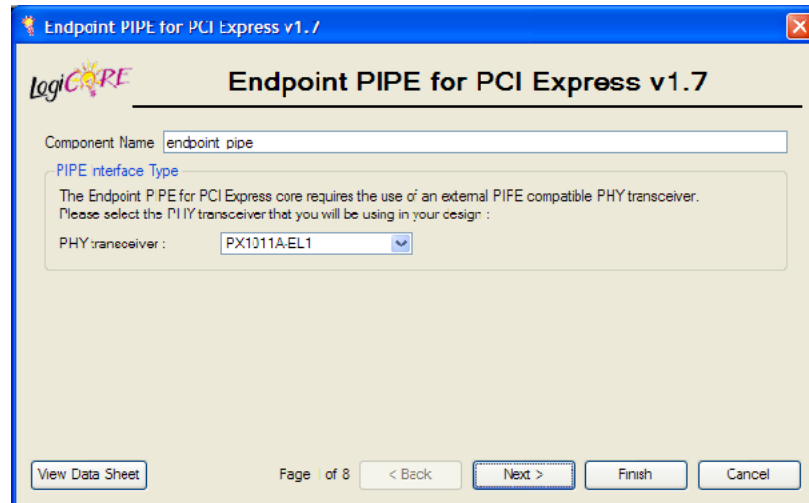


Figure 3-5: Endpoint PIPE for PCI Express Main Screen

7. In the Component Name field, enter a name for the core. <component\_name> is used to represent the project name in this example.
8. Click Finish to generate the core using the default parameters. The core and its supporting files, including the PIO example design and Downstream Port Model test bench, are generated in the project directory.

For detailed information about the example design directories and files, see “[Directory Structure and File Contents](#),” page 22.

## Simulating the Example Design

The example design provides a quick way to simulate and observe the behavior of the core. The simulation environment provided with Endpoint PIPE for PCIe core performs simple memory access tests on the PIO example design. Transactions are generated by the Downstream Port Model and responded to by the PIO example design.

- PCI Express Transaction Layer Packets (TLPs) are generated by the test bench transmit user application (`pci_exp_usrapp_tx`). As it transmits TLPs it also generates a log, `tx.dat`.
- PCI Express TLPs are received by the test bench receive user application (`pci_exp_usrapp_rx`). As the user application receives the TLPs, it generates a log file, `rx.dat`.

For more information about the test bench, see Appendix C, “Downstream Port Model Test Bench,” in the *LogiCORE Endpoint PIPE for PCI Express User Guide*.

## Setting up for Simulation

Simulation scripts are provided for Cadence IUS (VNC), Synopsys VCS, and Mentor Graphics ModelSim. Set your environment to run the simulation tool of your choice.

## Running the Simulation

The test bench provided with the example design supports pre-implementation mode (RTL) simulations:

- The test bench, along with RTL model of the example design
- The Verilog HDL model of the Endpoint PIPE for PCIe core, created by the CORE Generator
- The PCI Express PHY models (which must be obtained from NXP and installed) are simulated. For additional information about obtaining the NXP model, see [“Obtaining the NXP Simulation Models,”](#) page 13.

1. To run the simulation, go to the following directory:

```
<project_dir>/<component_name>/simulation/functional
```

2. Run the script that corresponds to your simulation tool using one of the following:

- **VCS:** `simulate_vcs.sh`
- **Verilog-NC:** `simulate_ncsim.sh`
- **ModelSim:** `simulate_mti.do`

## Implementing the Example Design

After generating the core, the netlists and the example design can be processed by the Xilinx implementation tools. The generated output files include scripts to assist the user in running the Xilinx software.

To implement the example design, open a command prompt or terminal window and type the following commands:

### Windows

```
ms-dos> cd <project_dir>\<component_name>\implement
ms-dos> implement.bat
```

### UNIX

```
unix-shell% cd <project_dir>/<component_name>/implement
unix-shell% ./implement.sh
```

These commands execute a script that synthesizes, builds, maps, and place-and-routes the example design. The script then generates a post-par simulation model for use in timing simulation. The resulting files are placed in the `results` directory and execute the following processes:

1. Remove data files from the previous runs.
2. Synthesizes the example design using either Synplicity Synplify or XST.
  - The core is instanced as a black box within the example design.
3. `ngdbuild` builds a Xilinx design database for the example design.

Inputs:

**Part-Package-Speed Grade selection:**

XC3S1000-FG676-4

**Example design UCF:**

`xilinx_1_lane_epipe_ep-XC3S1000-FG676-4.ucf`

4. `map`: Maps the design to the selected FPGA using the constraints provided.
5. `par`: Places cells onto FPGA resources and routes connectivity.
6. `trce`: Performs static timing analysis on design using specified constraints.
7. `netgen`: Generates a logical Verilog HDL representation of the design and an SDF file, for post-layout verification.
8. `bitgen`: Generates a bitstream file for programming the FPGA.

The following FPGA implementation related files are generated in the results directory:

- `routed.bit`  
FPGA configuration information.
- `routed.v`  
Verilog functional Model.
- `routed.sdf`  
Timing model Standard Delay File.
- `mapped.mrp`  
Xilinx map report.
- `routed.par`  
Xilinx place and route report.

- `routed.twr`  
Xilinx timing analysis report.

The script file starts from an EDIF/NGC file and results in a bitstream file. Although it is possible to use the Xilinx ISE GUI to implement the example design, the GUI flow is not included in this document.

## Directory Structure and File Contents

The PCI Express PIPE core directories and their associated files are defined in the sections that follow. Click a directory name to go to the desired directory and its associated files.

- [<project directory>](#)  
 Top-level project directory; name is user-defined
- [<project directory>/<component name>](#)  
 Core release notes file
  - [<component name>/implement](#)  
 Core implementation script files
  - [<component name>/doc](#)  
 Supporting core documentation
  - [<component name>/example\\_design](#)  
 Verilog design files and UCF
  - [<component name>/implement](#)  
 Implementation script files
    - [implement/results](#)  
 Results directory, created after implementation scripts are run, and contains implement script results
  - [<component name>/simulation](#)  
 Simulation scripts
    - [simulation/dsport](#)  
 Simulation files
    - [simulation/functional](#)  
 Functional simulation files
    - [simulation/tests](#)  
 Test command files

### <project directory>

The project directory contains all the CORE Generator project files.

**Table 3-1: Project Directory**

Name	Description
<project_dir>	
<component_name>.ngc	Top-level netlist.
<component_name>.v [hd]	Verilog or VHDL simulation model.
<component_name>.xco	CORE Generator project-specific option file; can be used as an input to the CORE Generator.
<component_name>_flist.txt	List of files delivered with core.

Table 3-1: Project Directory (Continued)

Name	Description
<component_name>.{veo vho}	VHDL or Verilog instantiation template.

[Back to Top](#)

## <project\_directory>/<component\_name>

The component name directory contains the release notes file provided with the core, which may include last-minute changes and/or updates.

Table 3-2: Component Name Directory

Name	Description
<project_dir>/<component_name>	
pci_express_pipe_release_notes.txt	Release notes file for the Endpoint PIPE for PCIe core.

[Back to Top](#)

## <component\_name>/doc

The doc directory contains the PDF documentation provided with the core.

Table 3-3: Doc Directory

Name	Description
<project_dir>/<component_name>/doc	
pcie_pipe_ds321.pdf	<i>LogiCORE Endpoint PIPE for PCI Express Data Sheet</i>
pcie_pipe_gsg168.pdf	<i>LogiCORE Endpoint PIPE for PCI Express Getting Started Guide</i>
pcie_pipe_ug167.pdf	<i>LogiCORE Endpoint PIPE for PCI Express User Guide</i>

[Back to Top](#)

## <component\_name>/example\_design

The example design directory contains the example design files provided with the core.

Table 3-4: Example Design Directory

Name	Description
<project_dir>/<component_name>/example_design	
pci_exp_1_lane_epipe_ep.v	Verilog top-level example design.
xilinx_pci_exp_1_lane_epipe_ep.v	Example design wrapper file.
xilinx_pci_exp_1_lane_epipe_endpoint_product.v	Enables Endpoint PIPE for PCIe version of the test bench.

**Table 3-4: Example Design Directory (Continued)**

Name	Description
<filename>.ucf	Example design UCF. Filename varies by family, part, and package selected.
EP_MEM.v impl_test.v pci_exp_32b_app.v PIO.v PIO_32.v PIO_32_RX_ENGINE.v PIO_32_TX_ENGINE.v PIO_EP.v PIO_EP_MEM_ACCESS.v PIO_TO_CTRL.v	PIO example design files.

[Back to Top](#)

## <component name>/implement

The implement directory contains the core implementation script files.

**Table 3-5: Implement Directory**

Name	Description
<project_dir>/<component_name>/implement	
implement.bat implement.sh	DOS and UNIX/Linux implementation scripts.
synplify.prj	Synplify synthesis script.
xilinx_pci_exp_1_lane_epipe_ep_inc.xst	XST project file.
xst.scr	XST synthesis script.

[Back to Top](#)

## implement/results

The results directory is created by the implement script, after which the implement script results are placed in the results directory.

**Table 3-6: Results Directory**

Name	Description
<project_dir>/<component_name>/implement/results	
The results directory is created by the implement script, after which the implement script results are placed in the results directory.	

[Back to Top](#)

## <component name>/simulation

The simulation directory contains the source files for the provided test bench.

**Table 3-7: Simulation Directory**

Name	Description
<project_dir>/<component_name>/simulation	
board_common.v	Contains test bench definitions.
board.v	Top-level simulation module and loop-back.
px1011a_pli.v	NXP PX1011A-EL1 PCI Express Standalone PHY dummy model for Synopsys VCS and ModelSim simulator. This file must be replaced with the actual model acquired from the PHY vendor. See <a href="#">“Obtaining the NXP Simulation Models,” page 13</a> for information.
px1011a.v	NXP PX1011A-EL1 PCI Express Standalone PHY dummy model, for Cadence IUS simulator. This file must be replaced with the actual model acquired from the PHY vendor. See <a href="#">“Obtaining the NXP Simulation Models,” page 13</a> for information.
sys_clk_gen_ds.v	System differential clock source.
sys_clk_gen.v	System clock source.
xilinx_pci_exp_32bit_cor_ep.v	Endpoint PIPE for PCIe application module. This module instances FPGA example design and PCI Express PHY, along with application models.
xilinx_pci_exp_cor_ep.f	List of files comprising the design being tested.
xilinx_pci_exp_defines.v	Endpoint PIPE for PCIe application macro definitions.

[Back to Top](#)

## simulation/dsport

The dsport directory contains files for the Downstream Port Model test bench.

**Table 3-8: dsport Directory**

Name	Description
<project_dir>/<component_name>/simulation/dsport	
dsport_cfg.v	Downstream Port Model files.
pci_exp_usrapp_cfg.v	
pci_exp_usrapp_tx.v	
pci_exp_usrapp_rx.v	
pci_exp_usrapp_com.v	
pci_exp_1_lane_64b_dsport.v	
xilinx_pci_exp_downstream_port.v	
pci_exp_expect_tasks.v	
xilinx_pci_exp_dsport.v	

[Back to Top](#)

## simulation/functional

The functional directory contains input files for functional simulations.

**Table 3-9: Functional Directory**

Name	Description
<project_dir>/<component_name>/simulation/functional	
board_rtl_x01_epipe.f board_rtl_x01_epipe_ncv.f	List of files for RTL simulations.
simulate_ncsim.sh	Cadence IUS simulation script.
simulate_vcs.sh	Synopsys VCS simulation script.
simulate_mti.do	ModelSim simulation script
wave_ncsim.sv	Cadence IUS waveform script.
wave_mti.do	ModelSim waveform script.
xilinx_lib.f xilinx_lib_mti_v4fx.f xilinxn_lib_vcs_v4fs.f	Xilinx libraries; requires modification to point to the user location of Xilinx libraries.

[Back to Top](#)

## simulation/tests

The tests directory contains sample tests to drive the downstream port model

Table 3-10: **dsport Directory**

Name	Description
<code>&lt;project_dir&gt;/&lt;component_name&gt;/simulation/tests</code>	
sample_tests1.v	Sample tests to drive the Downstream Port Model.
tests.v	
pio_tests.v	

[Back to Top](#)

## NXP Simulation Models

To simulate the Endpoint PIPE for PCI Express core with the PXPIPE interface, a simulation model of the NXP PX1011A-EL1 PCI Express PHY is required. This model is the property of NXP and is not included with this Xilinx product.

Various simulation models for commonly used tools are available. Please visit the NXP web page at [www.standardics.nxp.com/support/models/px/](http://www.standardics.nxp.com/support/models/px/) to obtain a license agreement and register for the simulation models. Be sure to include your contact information in your request so that NXP can respond to you as quickly as possible.



## Additional Design Considerations

---

### Package Constraints

This section discusses design considerations specific to the Endpoint PIPE for PCI Express targeting Spartan-3, Spartan-3E, and Spartan-3A devices. [Table A-1](#) defines the smallest supported device and interface combinations for the core.

*Table A-1: Supported Device and Interface Combinations*

Smallest Supported Device/Part Number	Data Bus Width/Speed	Wrapper File
XC3S1000 FG676-4	Width: 32-bit Port Speed: 62.5 MHz	xilinx_pci_exp_endpoint.v
XC3S500E CP132-4	Width: 32-bit Port Speed: 62.5 MHz	xilinx_pci_exp_endpoint.v
XC3S700A FG400-4	Width: 32-bit Port Speed: 62.5 MHz	xilinx_pci_exp_endpoint.v

### User Constraints Files

The UCF contains various constraints required for the Endpoint PIPE for PCIe core. The UCF is specific to the target device and must always be used while processing a design.

#### Spartan-3

```
<project_dir>/<component_name>/example_design/xilinx_1_lane_epipe_ep_XC3S1000-FG676-4.ucf
```

#### Spartan-3E

```
<project_dir>/<component_name>/example_design/xilinx_1_lane_epipe_ep_XC3S500E-CP132-4.ucf
```

#### Spartan-3A

```
<project_dir>/<component_name>/example_design/xilinx_1_lane_epipe_ep-XC3S1400A-FG484-4.ucf
```

### Wrapper File Usage

The wrapper contains an instance of the Endpoint PIPE for PCI Express core. When starting a new design, modify this wrapper to include all I/O elements and modules.

```
<project_dir>/<component_name>/example_design/xilinx_pci_exp_1_lane_epipe_ep.v
```

