

## Introduction

The Endpoint PIPE (PHY Interface) for PCI Express® 1-lane core is a high-bandwidth scalable and reliable serial interconnect intellectual property building block for use with the Spartan™-3, Spartan-3E, and Spartan-3A FPGAs in conjunction with an external PHY device. This solution, compliant with the *PCI Express Base Specification v1.1*, is a flexible low-cost chipset that can be used in a wide variety of high-volume applications including add-in cards, host bus adapters, and high-end server and graphics cards.

PCI Express (PCIe®) offers a serial architecture that alleviates some of the limitations of parallel bus architectures by using clock data recovery (CDR) and differential signaling. Using CDR (as opposed to source synchronous clocking) lowers pin count, enables superior frequency scalability, and makes data synchronization easier. The layered architecture of PCIe provides for future attachment to copper, optical, or emerging physical signaling media. PCI Express technology, adopted by the PCI-SIG as the next generation PCI, is backward-compatible to the existing PCI software model.

The Xilinx solutions for PCI Express set the industry standard for a high-performance and cost-efficient third-generation I/O solution by providing higher bandwidth per pin, low overhead, low latency, reduced signal integrity issues, and CDR architecture.

Xilinx Endpoint solutions for PCI Express are compatible with industry standard application form factors such as *PCI Express Card Electromechanical (CEM) v1.1* and *PCI Industrial Computer Manufacturers Group (PICMG) 3.4* specifications.

LogiCORE™ Facts				
Core Specifics				
Supported Device Families	Spartan-3, Spartan-3E, Spartan-3A			
Minimum Device Requirement	Spartan-3	XC3S1000-4		
	Spartan-3E	XC3S500E-4		
	Spartan-3A	XC3S700A-4		
Resources Used <sup>1</sup> , All Devices	Product	LUT	FF	Block RAM
	1-Lane Endpoint PIPE	5880-6150 <sup>2</sup>	4650-4790 <sup>2</sup>	8
Special Features	Digital Clock Manager block RAM			
Supported PHY	NXP PX1011A-EL1			
Provided With Core				
Documentation	Product Specification Getting Started Guide User Guide Instantiation Template			
Design Files	Verilog® and VHDL Simulation Models Xilinx Generic Netlist Format (ngo netlist) Verilog Example Test Bench Verilog Example Design			
Constraints File	User Constraints File (UCF)			
Design Tool Support <sup>3</sup>				
HDL Synthesis Tool (Verilog only)	Synplicity® Synplify®, Xilinx XST			
Xilinx Implementation Tools	Xilinx ISE™ v9.1i			
Verification Tools	Cadence™ IUS, Synopsys® VCS, Mentor Graphics® ModelSim®			
Support				
Provided by Xilinx @ <a href="http://www.xilinx.com/support">www.xilinx.com/support</a> .				

1. The precise number of slices depends on the user configuration of the interface and the level of resource sharing with adjacent logic.
2. This range indicates resources used for a 2BAR–7BAR implementation.
3. See the *LogiCORE Endpoint PIPE Getting Started Guide* or product release notes for additional supported versions.

## Features

- High-performance, highly flexible, scalable, reliable, and general purpose I/O core
  - Compliant with the *PCI Express Base Specification v1.1*
  - Compatible with conventional PCI software model
- Fully compliant with PCI Express transaction ordering rules
- Six individually programmable/configurable BARs and expansion ROM BAR
- Supports MSI and INTX emulation
- 32-bit internal data path
- Supports removal of corrupt packets for error detection and recovery
- Compatible with PCI/PCI Express power management functions
  - Active state power management (ASPM)
  - Programmed power management (PPM)
- Used in conjunction with NXP PX1011A PCI Express standalone PHY to achieve high transceiver capability
  - 2.5 Gbps line speed
  - Elastic buffers and clock compensation
  - Automatic clock and data recovery
  - 8b/10b encode and decode
- Offers standardized easy-to-use Xilinx LocalLink interface
  - Packet-based full-duplex communication
  - Back-to-back transactions enable greater link bandwidth utilization
  - Enables flow control of data and discontinuance of an in-process transaction in the transmit direction
  - Enables flow control of data in the receive direction
  - Automatically decodes and removes error forwarding packet indicator from received data
- Supports a maximum transaction payload of up to 512 bytes
- Fully configurable using the Xilinx CORE Generator™ v9.1i
- Design verified using a Xilinx proprietary test bench

## Applications

The Endpoint PIPE for PCI Express core architecture enables a broad range of computing and consumer communications target applications, emphasizing performance, cost, scalability, feature extensibility and mission-critical reliability. Typical applications include:

- Test equipment
- Consumer graphics boards
- Medical imaging equipment
- Data communications networks
- Telecommunications networks
- Broadband deployments
- Cross-connects
- Workstation and mainframe backbones
- Network interface cards
- Chip-to-chip and backplane interconnect
- Crossbar switches
- Wireless base stations
- High-bandwidth digital video
- High-bandwidth server applications

## Functional Description

The Endpoint PIPE for PCI Express is organized into four main modules based on the three discrete logical layers defined by the *PCI Express Base Specification v1.1*. The logic modules, which manage all the system-level functions, include the following:

- Physical Layer Module (PLM)
- Data Link Layer Module (LLM)
- Transaction Layer Module (TLM)
- Configuration Management Module (CMM)

Each module is further partitioned into the Receive and the Transmit sections. The Receive section processes the inbound information, and the Transmit section processes the outbound information. **Figure 1** illustrates the main modules interfacing with one another and with the User Application using a set of four interfaces:

- System (SYS) interface
- PIPE (PXPIPE) interface
- Configuration (CFG) interface
- Transaction (TRN) interface

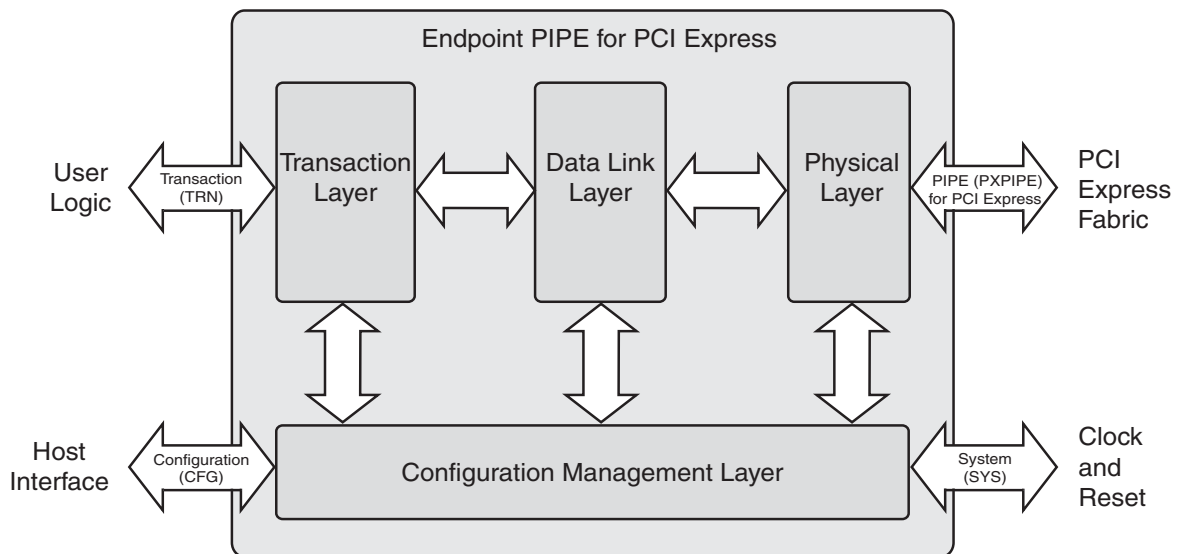


Figure 1: **Top-level Functional Blocks and Interfaces**

The core uses packets to exchange information between the various modules. Packets are formed in the Transaction and Data Link Layers to carry information from the transmitting component to the receiving component. Necessary information is added to the packet being transmitted, which is required to handle the packet at those layers. At the receiving end, each layer of the receiving element processes the incoming packet, strips the relevant information and forwards the packet to the next layer. As a result, the received packets get transformed from their Physical Layer representation to their Data Link Layer representation and the Transaction Layer representation. The main logic modules comprising the Endpoint PIPE for PCI Express and their interfaces are described in the following sections.

## Logic Modules

The logic modules are responsible for handling the functionality related to each of the layers defined by the *PCI Express Base Specification v1.1*. The functions of these modules include generation and processing of Transaction Layer Packets (TLPs), flow control management, initialization and power management functions, data protection, error checking and retry functions, physical link interface initialization, maintenance and status tracking, serialization, de-serialization and other circuitry for interface operation. These modules and their functionality are described below.

### Physical Layer Module

The Physical Layer exchanges information with the Data Link Layer in an implementation-specific format. This layer is responsible for converting information received from the Data Link Layer into an appropriate format and transmitting it across the PXPIPE interface at a frequency and width compatible with the external PHY.

**Note:** PXPIPE is the *NXP PHY Specification*, an extended version of the PIPE specification.

### Data Link Layer Module

The Data Link Layer acts as an intermediate stage between the Transaction Layer and the Physical Layer. Its primary responsibility is to provide a reliable mechanism for the exchange of TLPs between the two Components on a Link.





























