PCI32 SpartanXL Master & Slave Interface

March, 1999 Data Sheet

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-377-3259
E-mail: Techsupport: hotline@xilinx.com
Feedback: logicore@xilinx.com
URL: http://www.xilinx.com/pci

Introduction

With Xilinx LogiCORE PCI32 SpartanXL Master & Slave Interface, a designer can build a cost-efficient, customizable, zero wait-state, 32-bit, 33MHz fully PCI compliant system in a SpartanXL family FPGA.

Features

- Fully PCI V 2.2 compliant 32-bit, 33MHz Interface
  - Master (Initiator/Target)
- Incorporates Xilinx Smart-IP Technology with predefined implementation for predictable timing in Xilinx SpartanXL FPGAs (see *LogiCORE Facts* for listing of supported devices)
- 3.3V and 5V operation with SpartanXL devices
- Zero wait-state burst operation
- Fully verified design
  - Tested with Xilinx internal testbench and in hardware (silicon proven)
- Configurable on-chip dual-port FIFOs can be added for maximum burst speed (see *Xilinx Documents* section)
- Programmable single-chip solution with customizable back-end functionality
- Supported Master functions
  - Memory Read, Memory Write, Memory Read Multiple, Memory Read Line commands
  - I/O Read and I/O Write commands
  - Configuration Read and Configuration Write commands
  - Special Cycles, Interrupt Acknowledge
  - Basic Host Bridging
  - Bus Parking

LogiCORE™ Facts

Core Specifics	
Device Family	SpartanXL
CLBs Used[1]	152 - 268
IOBs Used	53
System Clock f_{max}	0 – 33MHz
Device Features Used	Bi-directional data buses SelectRAM™ (optional user FIFO) Boundary scan (optional)

Supported Devices/Resources Remaining		
	I/O	**CLB[1]**
XCS20XL TQ144	60	190 - 248[2]
XCS30XL PQ208	107	308 - 424
XCS30XL PQ240	141	308 - 424
XCS40XL PQ208	107	516 - 632
XCS40XL PQ240	141	516 - 632

Provided with Core[3]	
Documentation	*PCI Design Guide* *Implementation Guide* *Conversion Guide*
Design File Formats	VHDL & Verilog Simulation Models NGO Netlist
Constraint Files	M1 User Constraint File (UCF) M1 Guide files
Verification Tools	VHDL and Verilog Testbench
Core Symbols	VHDL, Verilog
Reference designs	Synthesizable PCI Bridge Design

Design Tool Requirements	
Xilinx Core Tools	M1.5I
Entry Tools[4]	FPGA Express, FPGA Compiler, Synplicity
Verification Tools[4]	Verilog-XL, MTI

Support
Xilinx provides technical support for this LogiCORE product when used as described in the User's Guide or in the Application Notes.

Notes:
1. The exact number of CLBs depends on user configuration of the core and level of resource sharing with adjacent logic. Factors that can affect the size of the design are number and size of the BARs, and use of the latency timer.
2. The XCS20XL device only supports one BAR.
3. Available on Xilinx web site: www.xilinx.com/pci
4. See Xilinx web site for latest list of tested EDA tools and version numbers.

Features (cont.)

- Supported Target functions
  - Type 0 Configuration Space Header
  - Up to 2 Base Address Registers (memory or I/O with adjustable block size from 16 Bytes to 2 GBytes, slow decode speed)
  - Parity Generation (PAR), Parity Error Detection (PERR# and SERR#)
  - Extended Capabilities Registers (backend module)
  - Memory Read, Memory Write, Memory Read Multiple (MRM), Memory Real Line (MRL), and Memory Write & Invalidate (MWI) commands
  - I/O Read and I/O Write commands
  - Configuration Read and Configuration Write commands
  - Interrupt Acknowledge
  - 32-bit data transfers, burst transfers with linear address ordering
  - Target Abort, Target Retry, Target Disconnect
  - Full Command/Status Register
- Available for configuration and download on the web
  - Web-based configuration tool
  - Generation of proven design files
  - Instant access to new releases

Applications

- PCI add-in boards such as graphic cards, video adapters, LAN adapters, and data acquisition boards
- Embedded applications within networking, telecommunication, and industrial systems
- CompactPCI boards
- Other applications that need PCI

General Description

The LogiCORE™ PCI32 SpartanXL Master and Slave Interfaces are pre-implemented and fully tested modules for Xilinx SpartanXL FPGAs (see *LogiCORE Facts* for listing of supported devices). The pin-out and the relative placement of the internal Configurable Logic Blocks (CLBs) are pre-defined. Critical paths are controlled by TimeSpecs and guide files to ensure that timing is always met. This significantly reduces engineering time required to implement the PCI portion of your design. Resources can instead be focused on the unique back-end logic in the FPGA and the system level design. As a result, LogiCORE PCI products can minimize development time.

Xilinx SpartanXL family FPGAs enables the design of fully PCI compliant systems. These devices meet all specifications for 3.3 V and 5 V PCI and meet all required electrical and timing parameters including AC output drive characteristics, input capacitance specifications (10pF), 7 ns setup and 0 ns hold to system clock, and 11 ns system clock to output.

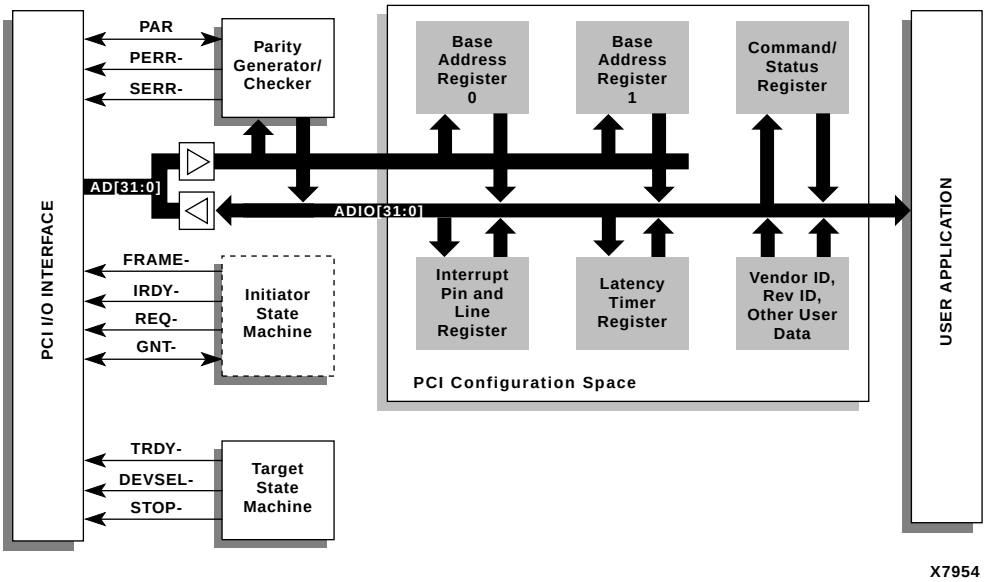


Figure 1: LogiCORE PCI32 SpartanXL Interface Block Diagram (one BAR only in XCS20XL)

The PCI Compliance Checklists, found in the Xilinx PCI Databook, have additional details. Other features that enable efficient implementation of a complete PCI system in the SpartanXL family includes:

- Select-RAM™ memory: on-chip ultra-fast RAM with synchronous write option and dual-port RAM option. Used in the PCI32 SpartanXL Interface to implement the FIFO.
- Individual output enable for each I/O
- Internal 3-state bus capability
- 8 global low-skew clock or signal distribution networks
- IEEE 1149.1-compatible boundary scan logic support

See *Spartan FPGA Data Sheet* for more details.

The module is carefully optimized for best possible performance and utilization in the SpartanXL FPGA architecture. When implemented in the XCS30, more than 50% of the FPGA's resources remain for integrating a unique back-end interface and other system functions into a fully programmable one-chip solution. When implemented in the XCS40, more than 65% of the FPGA's resources remain for integrating a unique back-end interface and other system functions into a fully programmable one-chip solution.

Smart-IP Technology

Drawing on the architectural advantages of Xilinx FPGAs, new Xilinx Smart-IP technology ensures optimal performance, predictability, reproducibility, and flexibility in PCI designs. The Smart-IP technology is incorporated in every LogiCORE PCI.

Xilinx Smart-IP technology leverages the Xilinx architectural advantages, such as look-up tables (LUTs), distributed RAM, and segmented routing, logic mapping, and relative location constraints. This Smart-IP technology provides the best physical layout, predictability, performance, and significantly reduced compile times in designing a PCI Core.

The PCI32 SpartanXL Interface can be parameterized, allowing for design flexibility in which users can create the exact PCI interface needed. PCI Cores made with Smart-IP technology are unique by maintaining their performance and predictability regardless of the device size.

Functional Description

The LogiCORE PCI32 SpartanXL Interface is partitioned into five major blocks, plus the user application, shown in Figure 1. Each block is described below.

PCI I/O Interface Block

The I/O interface block handles the physical connection to the PCI bus including all signaling, input and output synchronization, output three-state controls, and all request-grant handshaking for bus mastering.

Parity Generator/Checker

Generates/checks even parity across the AD bus, the CBE lines, and the PAR signal. Reports data parity errors via PERR- and address parity errors via SERR-.

Target State Machine

This block manages control over the PCI32 SpartanXL Interface for Target functions. The states implemented are a subset of equations defined in "Appendix B" of the *PCI Local Bus Specification*. The controller is a high-performance state machine using state-per-bit (one-hot) encoding for maximum performance. State-per-bit encoding has narrower and shallower next-state logic functions that closely match the Xilinx FPGA architecture.

Initiator State Machine

This block manages control over the PCI32 SpartanXL Interface for Initiator functions. The states implemented are a subset of equations defined in "Appendix B" of the *PCI Local Bus Specification*. The Initiator Control Logic also uses state-per-bit encoding for maximum performance.

PCI Configuration Space

This block provides the first 64 bytes of Type 0, version 2.1, Configuration Space Header (CSH) (see Table 1) to support software-driven "Plug-and Play" initialization and configuration. This includes Command, Status, and two Base Address Registers (BARs). These BARs illustrate how to implement memory- or I/O-mapped address spaces. Each BAR sets the base address for the interface and allows the system software to determine the addressable range required by the interface. Using a combination of Configurable Logic Block (CLB) flip-flops for the read/write registers and CLB look-up tables for the read-only registers results in optimized packing density and layout.

With this release, the hooks for extending configuration space has been built into the backend interface. Setting the CapPtr and bit 15 of the Status Register allows the user to implement functions such as Advanced Configuration and Power Interface (ACPI) in the backend design.

User Application with Optional Burst FIFOs

The LogiCORE PCI32 SpartanXL Interface provides a simple, general-purpose interface with a 32-bit data path and latched address for de-multiplexing the PCI address/data bus. The general-purpose user interface allows the rest of the device to be used in a wide range of applications.

Typically, the user application contains burst FIFOs to increase PCI system performance (An Application Note is available, please see the *Xilinx Documents* section). An on-chip read/write FIFO, built from the on-chip synchronous dual-port RAM (SelectRAM™) available in SpartanXL devices, supports data transfers in excess of 33 MHz.

Table 1: PCI Configuration Space Header

31	16	15	0	
Device ID		Vendor ID		00h
Status		Command		04h
Class Code			Rev ID	08h
BIST	Header Type	Latency Timer	*Cache Line Size*	0Ch
Base Address Register 0 (BAR0)				10h
Base Address Register 1 (BAR1)				14h
Base Address Register 2 (BAR2)				18h
Base Address Register 3 (BAR3)				1Ch
Base Address Register 4 (BAR5)				20h
Base Address Register 5 (BAR5)				24h
Cardbus CIS Pointer				28h
Subsystem ID		Subsystem Vendor ID		2Ch
Expansion ROM Base Address				30h
Reserved			CapPtr	34h
Reserved				38h
Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line	3Ch
Reserved				40h-FFh

Note:
Italicized address areas are not implemented in the LogiCORE PCI32 SpartanXL Interface default configuration. These locations return zero during configuration read accesses.

Interface Configuration

The LogiCORE PCI32 SpartanXL Interface can easily be configured to fit unique system requirements using Xilinx web-based PCI Configuration and Download Tool. The following customization is supported by the LogiCORE product and described in accompanying documentation.

- Initiator and target functionality
- Base Address Register configuration (1-2 Registers in XCS30XL and XCS40XL, 1 BAR only in XCS20XL, size and mode of BAR)
- Configuration Space Header ROM
- Initiator and target state machine (e.g., termination conditions, transaction types and request/transaction arbitration)
- Burst functionality
- User Application including FIFO (back-end design)

Table 2: PCI Bus Commands

CBE [3:0]	Command	PCI Master	PCI Slave
0000	Interrupt Acknowledge	Yes	Yes
0001	Special Cycle	Yes	Ignore
0010	I/O Read	Yes	Yes
0011	I/O Write	Yes	Yes
0100	Reserved	Ignore	Ignore
0101	Reserved	Ignore	Ignore
0110	Memory Read	Yes	Yes
0111	Memory Write	Yes	Yes
1000	Reserved	Ignore	Ignore
1001	Reserved	Ignore	Ignore
1010	Configuration Read	Yes	Yes
1011	Configuration Write	Yes	Yes
1100	Memory Read Multiple	Yes	Yes
1101	Dual Address Cycle	No[1]	Ignore
1110	Memory Read Line	Yes	Yes
1111	Memory Write Invalidate	No[1]	Yes

Note:
1. The Initiator can present these commands, however, they either require additional user-application logic to support them or have not been thoroughly tested.

Supported PCI Commands

Table 2 illustrates the PCI bus commands supported by the LogiCORE PCI32 SpartanXL Interface. The compliance checklist later in this data book have more details on supported and unsupported commands.

Burst Transfer

The PCI bus derives its performance from its ability to support burst transfers. The performance of any PCI application depends largely on the size of the burst transfer. A FIFO to support PCI burst transfer can efficiently be implemented using the SpartanXL on-chip RAM feature, SelectRAM™. Each SpartanXL CLB supports two 16x1 RAM blocks. This corresponds to 32 bits of single-ported RAM or 16 bits of dual-ported RAM, with simultaneous read/write capability.

Bandwidth

The Xilinx PCI32 SpartanXL Interface supports a sustained bandwidth of up to 132 MBytes/sec. The design can be configured to take advantage of the ability of the LogiCORE PCI32 Interface to do very long bursts. Since the FIFO does not have a fixed size, a burst can go on for as long as the chipset arbiter will allow. Furthermore, since the FIFOs and the DMA are decoupled from the proven core, a designer can modify these functions without affecting the critical PCI timing.

The flexible Xilinx backend, combined with support for many different PCI features, gives users a solution that can be used in many high-performance applications. Xilinx is able to support different depths of FIFOs as well as dual port FIFOs, synchronous or asynchronous FIFOs, and multiple FIFOs. The user is not restricted to one DMA engine, hence, a DMA that fits a specific application can be designed.

The theoretical maximum bandwidth of a 32-bit, 33 MHz PCI bus is 132 MBytes. How close you get to this maximum bandwidth will depend on several factors, including the PCI design used, PCI chipset, the processor's ability to keep up with your data stream, the maximum capability of your PCI design, and other traffic on the PCI bus. Older chipsets and processors will tend to allow less bandwidth than newer ones.

In the Zero wait-state mode, no wait-states are inserted either while sourcing data or receiving data. This allows a 100% burst transfer rate in both directions with full PCI compliance. No additional wait-states are inserted in response to a wait-state from another agent on the bus, as required by the PCI V 2.2 specification. Either IRDY or TRDY is kept asserted until the current data phase ends, as required by PCI V 2.2 Specification.

In this version of the PCI Interface, based on the Xilinx V3.0.X PCI Interface, the end of initiator transaction wait-state has been removed.

See Table 3 for PCI bus transfer rates for various operations in Zero wait-state mode.

Table 3: LogiCORE PCI32 SpartanXL Transfer Rates

Zero Wait-State Mode	
Operation	**Transfer Rate**
Initiator Write (PCI ← LogiCORE)	3-1-1-1
Initiator Read (PCI → LogiCORE)	4-1-1-1
Target Write (PCI→ LogiCORE)	5-1-1-1
Target Read (PCI ← LogiCORE)	6-1-1-1

Note: Initiator Read and Target Write operations have effectively the same bandwidth for burst transfer.

Timing Specification

The SpartanXL family, together with the LogiCORE PCI32 Interface enables design of fully compliant PCI systems. Backend design can affect the maximum speed your design is capable of. Factors in your back-end designs that can affect timing include loading of hot signals coming directly from the PCI bus, gate count and floor planning. Table 4 shows the key timing parameters for the LogiCORE PCI32 SpartanXL Interface that must be met for full PCI compliance.

Verification Methods

Xilinx has developed a testbench with numerous vectors to test the Xilinx PCI design; this is included with the LogiCORE PCI32 SpartanXL Master and Slave Interfaces. A version of this testbench is also used internally by the Xilinx PCI team to verify the PCI32 Interfaces. Additionally, the PCI32 Interfaces have been tested in hardware for electrical, functional and timing compliance.

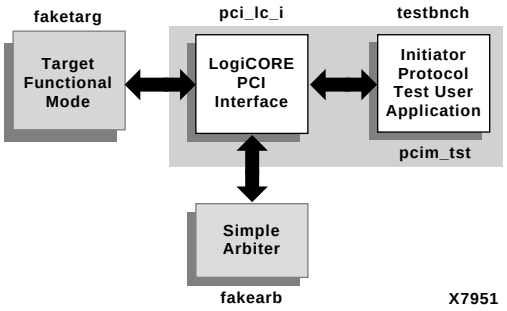
Table 4. Advanced Timing Parameters [ns]

Parameter	Ref.	PCI Spec.		LogiCORE PCI32, XCSXL-4	
		Min	Max	Min	Max
CLK Cycle Time		30	∞	30[1]	∞
CLK High Time		11		11	
CLK Low Time		11		11	
CLK to Bus Signals Valid[3]	T_{ICK-OF}	2	11	2[2]	9.6
CLK to REQ# and GNT# Valid[3]	T_{ICK-OF}	2	12	2[2]	9.6
Tri-state to Active		2		2[2]	
CLK to Tri-state			28		28[1]
Bus Signal Setup to CLK (IOB)	T_{PSU}		7		7
Bus Signal Setup to CLK (CLB)			7		7[1]
GNT# Setup to CLK	T_{PSU}		10		5.2
Input Hold Time After CLK (IOB)	T_{PH}		0		0
Input Hold Time After CLK (CLB)			0		0[2]
RST# to Tri-state			40		40[2]

Notes:
1. Controlled by TIMESPECs, included in product
2. Verified by analysis and bench-testing
3. IOB configured for Fast slew rate

The testbench shipped with the interface verifies the PCI interface functions according to the test scenarios specified in *PCI Compliance Checklist, V 2.1*; see Figure 2. This testbench consists of 28 test scenarios, each designed to test a specific PCI bus operation. Refer to the checklists chapter in this databook for a complete list of scenarios.

Figure 2. PCI Protocol Testbench



Ping Reference Design

The Xilinx LogiCORE PCI “PING” Application Example, delivered in VHDL and Verilog, has been developed to provide an easy-to-understand example which demonstrates many of the principles and techniques required to successfully use a LogiCORE PCI32 Spartan Interface in a System-on-a-Chip solution.

Synthesizable PCI Bridge Design Example

Synthesizable PCI bridge design examples, delivered in Verilog and VHDL, are available to demonstrate how to interface with the LogiCORE PCI32 Spartan Interface and provide a modular foundation upon which to base other designs. See separate data sheet for details.

Device Utilization

The Target-Only and Target/Initiator options require a variable amount of CLB resources for the PCI32 Spartan Interface. The core includes a switch to force the entire deletion of unused Base Address Registers.

Utilization can vary widely, depending on the configuration choices made by the designer. Options that can affect the size of the core are:

- Number of Base Address Registers Used. Turning off any unused BARs will save on resources.
- Size of the BARs. Setting the BAR to a smaller size requires more flip-flops. A smaller address space requires more flip-flops to decode.
- Latency timer. Disabling the latency timer will save a few resources. It must be enabled for bursting.

Recommended Design Experience

The LogiCORE PCI32 Spartan Interface is pre-implemented allowing engineering focus at the unique back-end functions of a PCI design. Regardless, PCI is a high-performance system that is challenging to implement in any technology, ASIC or FPGA. Therefore, we recommend previous experience with building high-performance, pipelined FPGA designs using Xilinx implementation software, TIMESPECs, and guide files. The challenge to implement a complete PCI design including back-end functions varies depending on configuration and functionality of your application. Contact your local Xilinx representative for details on your specific design requirements.