DATA BOOK

PCI Bridge IP Core

Introduction

Unlike other PCI IP cores on the market today, this product is more than a simple PCI interface that leaves one with a lot of backend implementation issues still not addressed properly. Rather, this PCI Bridge IP Core is a real bridge, with a standardized backend that only needs to be connected to other IP cores on the WISHBONE bus and functions immediately without any additional efforts by HDL designers. When a bridge is implemented into the design, it becomes software configurable. The system designer, in conjunction with the software designer, can achieve fast operations by knowing just a few of the most important PCI operations. But even that is not a must.
If core parameters are set properly before the synthesis (like FIFO depths, number of images etc.), they allow to choose between a highly configurable with high throughput, a low cost bridge (in terms of technology specific device utilization), or a tradeoff between those two extremes.

Features

The following lists the main features of the PCI Bridge IP Core:

- Independent clock domains for PCI and WISHBONE side of the bridge
- Provision of two possible parameterized implementations – HOST (used for host bridging with the WISHBONE SoC bus as host bus) and GUEST (can be used for expansion bus bridging with WISHBONE SoC bus as expansion bus)
- WISHBONE SoC bus revision B.1 compliant with separate master and slave interfaces
- PCI 2.2 compliant 32-bit, 33 or 66-MHz initiator and target interface
- Zero Wait state burst operation
- Four synthesizable, dual port FIFOs with parameterized depth
- Parameterized number of synthesizable, fully programmable images (default one, maximum 6 images) with address translation capability and an image size of 4KB to 1GB
- Programmable image address space mapping (I/O or memory space)
- In-core realization of PCI transaction ordering requirements (bridge uses posted writes and delayed reads in either direction)
- Single delayed transaction support in either direction
- Extended configuration space, implemented for additional software programmable features of the core
- Fully transparent PCI bus command usage, controllable on an image-by-image basis with proper settings of configuration registers
- Supported initiator functions:
  - Memory Read, Memory Read Line, Memory Read Multiple, Memory Write commands
  - IO Read and Write commands
  - Configuration Read and Write commands
  - Interrupt Acknowledge command
  - Linear burst ordering supported
  - Bus Parking
  - Fully transparent WISHBONE interface operation, controllable on an image-by-image basis with proper software settings of configuration registers
- Supported target functions:
  - Memory Read, Memory Read Line, Memory Read Multiple, Memory Write, Memory Write and Invalidate commands

- IO Read and Write commands
- Configuration Read and Write commands
- Support of linear burst ordering
- Fast Back-to-Back Capable Target response

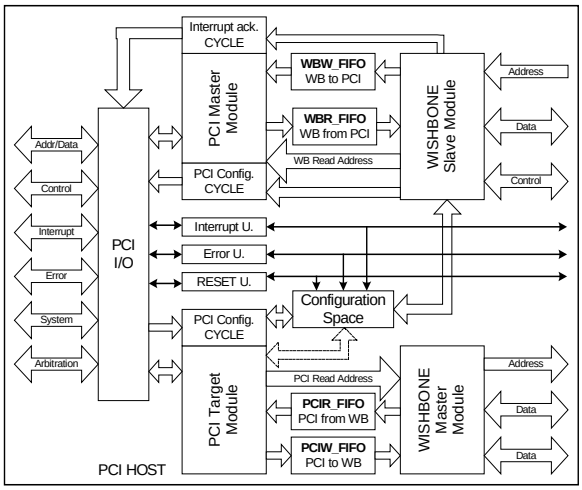

Figure 1: Core Architecture Overview

Architecture

Figure 1 shows the general architecture of the PCI Bridge IP Core. It consists of several building blocks:

- WISHBONE Slave Module
- WISHBONE Write FIFO (WBW_FIFO)
- WISHBONE Read FIFO (WBR_FIFO)
- PCI Master Module
- PCI I/O
- PCI Target Module
- PCI Write FIFO (PCIW_FIFO)
- PCI Read FIFO (PCIR_FIFO)
- WISHBONE Master Module
- Configuration Space

WISHBONE Slave Module

The slave interface is WISHBONE Rev. B.1 compliant. It accepts accesses that fall into one of the implemented image's regions and responds according to that image's configuration. Image configuration can be software programmed to perform address translation and/or use memory access optimizing commands when a burst transfer is requested. Writes are handled as posted and reads as delayed. Configuration space is also accessible through the WISHBONE slave interface with read/write capability for HOST and read only for GUEST bridges. Configuration-cycle generation is performed as delayed transaction, regardless whether it is a read or write request. An interrupt acknowledge cycle is processed as delayed read. The bridge allows only one outstanding delayed transaction request at a time in each direction.

WISHBONE Write FIFO

The WISHBONE Write FIFO stores address and data information coming from the external WISHBONE master that performs a write to the address that falls within one of the image's address spaces. On the other side, the PCI master state machine takes data out and completes writes on the PCI bus. Depth is parameterized.

WISHBONE Read FIFO

This FIFO stores data from delayed reads that have already completed on the PCI bus. The PCI Master state machine performs a read when requested, stores data in the FIFO, and provides data for the external WISHBONE master when it repeats the request through the WISHBONE Slave Interface. FIFO depth is parameterized.

PCI Master Module

The PCI master module is a PCI 2.2 compliant initiator interface. It is responsible for completing posted writes stored in the WBW_FIFO or servicing delayed read, configuration read/write or interrupt acknowledge cycle requests. The bus command used on the PCI bus depends on several factors:

- In an image access it depends on the address space mapping of an image (IO or memory), i.e. if it is a burst transfer, and on the configuration of an image, i.e. if memory access optimizing commands are enabled.
- Other requests use specially dedicated commands according to the PCI specification.

PCI Target Module

The target state machine accepts configuration cycles from the PCI bus and cycles that fall within the address space of an image.
Configuration cycle accesses provide access to the Type0 predefined header portion of

configuration space only. Extended configuration space is accessible with memory read and write accesses through a special 4KB memory mapped image. Read only access to configuration space is allowed to HOST, read/write access to GUEST bridges. All but configuration accesses are passed through the bridge to the WISHBONE master interface. Each implemented image can be configured to perform address translation and/or to configure the WISHBONE master state machine to perform burst transfers even though the transfer-requesting PCI device does not use a memory access optimizing command.

PCI Write FIFO

The PCI Write FIFO stores address and data information coming from an external PCI initiator that performs a write command to an address that falls within one of the image's address spaces. On the other side, the WISHBONE master state machine takes out data and completes writes on the WISHBONE bus. Depth is parameterized.

PCI Read FIFO

The PCI Read FIFO stores data from delayed reads that already completed on the WISHBONE bus. The WISHBONE master state machine performs a read when requested, stores data in the FIFO, and provides data for the external PCI initiator when repeating the request through the PCI target module. FIFO depth is parameterized.

WISHBONE Master Module

The WISHBONE master module responds to requests received from the PCI bus and passed through the bridge. When memory-optimizing commands are used on the PCI bus, the WISHBONE master interface can boost performance by pre-fetching large bursts for reads and storing them in a read FIFO. Burst writes accepted from the PCI bus and stored in the Write FIFO are also performed as burst writes on the WISHBONE bus.

Configuration Space

4KB of configuration space is provided for software controllable features of the bridge. Besides a Type0 predefined header demanded by the PCI specification, there are registers provided for image control, address translation, interrupt control and status, error reporting etc. Read/Write access is provided for the WISHBONE bus in HOST implementations and for the PCI bus in GUEST implementations. The opposite side has read-only access to configuration space.

Clocks

PCI and WISHBONE clocks are independent of each other and have no special synchronization demands. The PCI clock is specified in the PCI specification to be 0-66 MHz, while the frequency of the WISHBONE clock is up to the designers' will and technology limitations.

Interrupts

Given certain conditions, such as Parity error, Target Abort or similar signals during posted writes, the bridge can trigger interrupts. The configuration space provides a mechanism for enabling these interrupts and an interrupt status register for reporting them. The bridge also routes and reports interrupts triggered on busses. The HOST bridge implementation triggers interrupts (if enabled) on the WISHBONE bus, the GUEST bridge implementation on the PCI bus.