Data Sheet for PCIe to ISA Bus Controller

DOCUMENT REVISION HISTORY

Revision	Date	Change Description	Author
1.0	22nd Jan 2009	Draft version	CP
REL 1.0	16th Aug 2012	Removed implementation results	VC

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Purpose

The purpose of this document is to describe the technical specifications of PCIe to ISA bus controller implemented on the FPGA and how the interface is done with the PCIe PHY to complete the PCIe edge connector interface.

1.2 Features

The following are the main features of the PCIe to ISA Bus Controller:

- System Interface
  - 100MHz external reference clock for synchronous clocking of PCI express Interface
  - Supports interface to external active low reset signal
- PCI Express Interface
  - Compliant with the PCI Express base specification v1.1
  - Lane width supported x1
  - Link speed supported 2.5 Gbps
  - User interface width supported 32-bit
- PCI Express Application Interface
  - Target only support
  - I/O BAR0 supported for ISA I/O bus access
  - Memory BAR1 supported for ISA memory bus access.
- ISA Master Interface
  - The ISA Bridge implements a 16-bit data interface
  - Supports Bus clock of 8 MHz for ISA interface
  - Supports a 20-bit system address lines tristate, which can be latched on to the falling edge of bus address latch enable signal
  - Supports latchable address lines, these unlatched address signals give the system up to 16 MB of address ability

1.3 Acronyms and Abbreviations

Table 1: Acronyms & Abbreviations

Term	Meaning
FPGA	Field Programmable Gate Arrays
BAR	Base Address Register
PHY	Physical Layer
ISA	Industry Standard Architecture
PIO	Programmed Input Output
TLP	Transaction Layer Packet

2 PCIe to ISA Bus Controller

2.1 Block Diagram

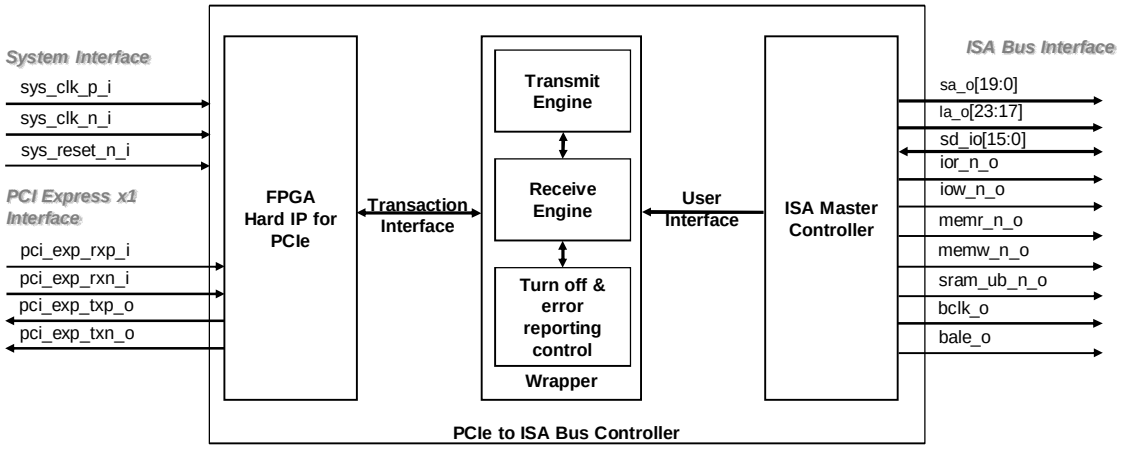

Figure 1: PCIe to ISA bus Controller Block Diagram

2.2 Description

- **PCIe Endpoint**: This is a hard macro IP from Xilinx/Altera. It implements Gen1 x1 PCIe endpoint.
- **Transmit Engine:** This module handle the Transmit TLP packet to PCIe Hard IP.
- **Receive Engine:** This module handle the Receive TLP packet from PCIe Hard IP.
- **ISA Master:** The ISA master is an ISA bus controller implemented in user interface side of the PIO design. The host processor can access the unit through I/O or memory read/writes commands. The ISA bus is a 16-bit interface, which can be used to connect peripheral components to the host CPU through ISA bus. PCIe to ISA bus controller supports the 16-bit data transfer between host CPU & peripheral components connected through ISA bus in ISA bridge side.

2.3 I/O Description

Table 2: System Interface IO Signals

Signal	I/O	Width	Description
sys_clk_p_i	I	1	Reference clock Positive: 100 MHz
sys_clk_n_i	I	1	Reference clock Negative: 100 MHz
sys_reset_n_i	I	1	Asynchronous signal. This signal is used to reset the PCI Express block, the GTP transceiver, the block RAM, and the PLL. The minimum pulse width for this input signal is 1500 ns.

Table 3: PCIe Interface IO Signals

Signal	Dir	Width	Description
pci_exp_txp_o	O	1	PCI Express Transmit Positive: Serial Differential Output 0 (+)
pci_exp_txn_o	O	1	PCI Express Transmit Negative: Serial Differential Output 0 (–)
pci_exp_rxp_i	I	1	PCI Express Receive Positive: Serial Differential Input 0 (+)
pci_exp_rxn_i	I	1	PCI Express Receive Negative: Serial Differential Input 0 (-)

Table 4: ISA Bus Interface Signal Description

Signal	Dir	Width	Description
sa_o [19:0]	O	20	System address bits
la_o [23:17]	O	7	Latchable address lines combine with the lower address lines to form a 24-bit address space
sbhe_n_o	O	1	Active low system Bus High Enable
sd_io [15:0]	O	16	System bi-directional data lines
ior_n_o	O	1	Active low IO read enable signal
iow_n_o	O	1	Active low IO write enable signal
memr_n_o	O	1	Active low memory read enable signal

Signal	Dir	Width	Description
memw_n_o	O	1	Active low memory write enable signal
bclk_o	O	1	Bus clock
bale_o	O	1	Bus address latch enable signals

3 Timing Waveforms

3.1 ISA Interface Waveforms

3.1.1 Memory write

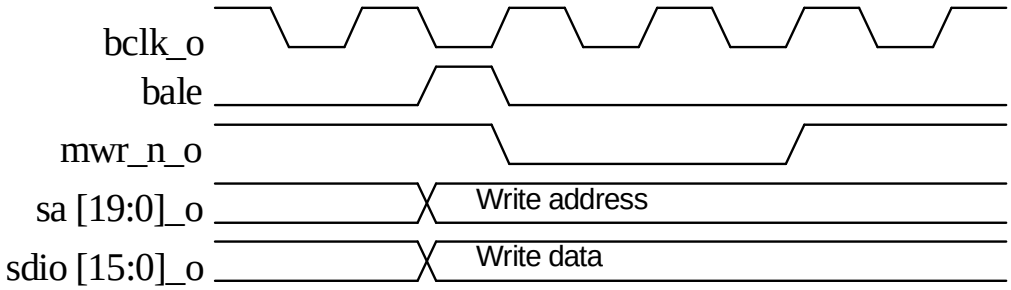

Figure 2: Memory write

3.1.2 IO write

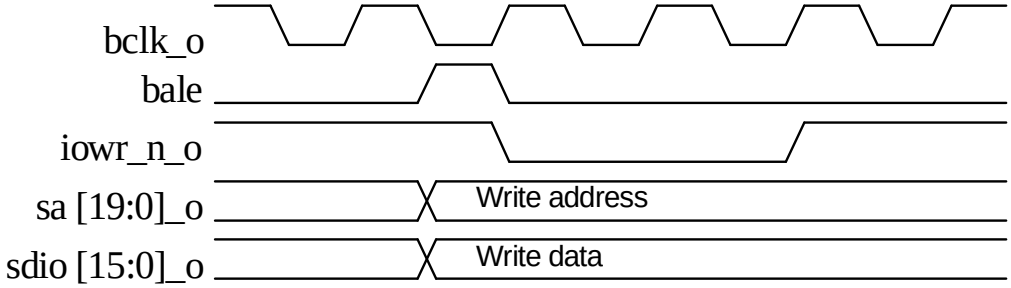

Figure 3: IO write

3.1.3 Memory read

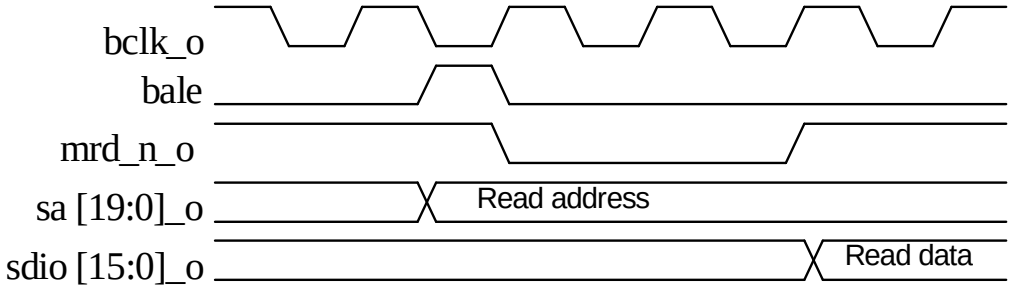

Figure 4: Memory read

3.1.4 IO read

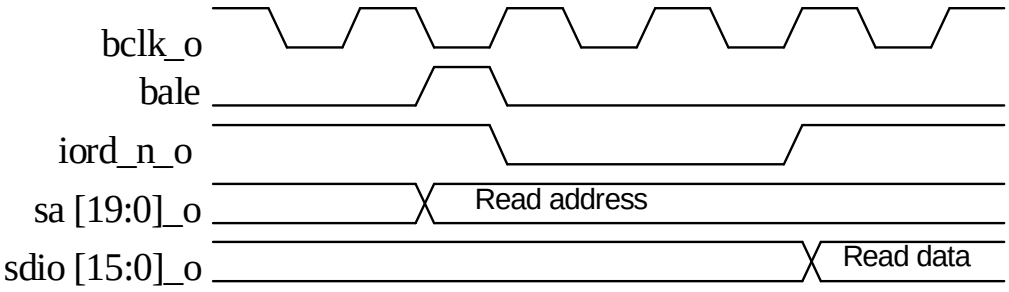

Figure 5: IO read