IDT™ 89HPES12N3
PCI Express® Switch

User Manual

June 2006

6024 Silver Creek Valley Road, San Jose, California 95138
Telephone: (800) 345-7015 • (408) 284-8200 • FAX: (408) 284-2775
Printed in U.S.A.

GENERAL DISCLAIMER

Integrated Device Technology, Inc. reserves the right to make changes to its products or specifications at any time, without notice, in order to improve design or performance and to supply the best possible product. IDT does not assume any responsibility for use of any circuitry described other than the circuitry embodied in an IDT product. The Company makes no representations that circuitry described herein is free from patent infringement or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent, patent rights or other rights, of Integrated Device Technology, Inc.

CODE DISCLAIMER

Code examples provided by IDT are for illustrative purposes only and should not be relied upon for developing applications. Any use of the code examples below is completely at your own risk. IDT MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE NONINFRINGEMENT, QUALITY, SAFETY OR SUITABILITY OF THE CODE, EITHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. FURTHER, IDT MAKES NO REPRESENTATIONS OR WARRANTIES AS TO THE TRUTH, ACCURACY OR COMPLETENESS OF ANY STATEMENTS, INFORMATION OR MATERIALS CONCERNING CODE EXAMPLES CONTAINED IN ANY IDT PUBLICATION OR PUBLIC DISCLOSURE OR THAT IS CONTAINED ON ANY IDT INTERNET SITE. IN NO EVENT WILL IDT BE LIABLE FOR ANY DIRECT, CONSEQUENTIAL, INCIDENTAL, INDIRECT, PUNITIVE OR SPECIAL DAMAGES, HOWEVER THEY MAY ARISE, AND EVEN IF IDT HAS BEEN PREVIOUSLY ADVISED ABOUT THE POSSIBILITY OF SUCH DAMAGES. The code examples also may be subject to United States export control laws and may be subject to the export or import laws of other countries and it is your responsibility to comply with any applicable laws or regulations.

LIFE SUPPORT POLICY

Integrated Device Technology's products are not authorized for use as critical components in life support devices or systems unless a specific written agreement pertaining to such intended use is executed between the manufacturer and an officer of IDT.

1. Life support devices or systems are devices or systems which (a) are intended for surgical implant into the body or (b) support or sustain life and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any components of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

About This Manual

Notes

Introduction

This user manual includes hardware and software information on the 89HPES12N3, a member of IDT's PRECISE™ family of PCI Express® switching solutions offering the next-generation I/O interconnect standard.

Finding Additional Information

Information not included in this manual such as mechanicals, package pin-outs, and electrical characteristics can be found in the data sheet for this device, which is available from the IDT website **(www.idt.com)** as well as through your local IDT sales representative.

Content Summary

Chapter 1, "PES12N3 Device Overview," provides a complete introduction to the performance capabilities of the 89HPES12N3. Included in this chapter is a summary of features for the device as well as a system block diagram and pin description.

Chapter 2, "Clocking, Reset, and Initialization," provides a description of the two differential reference clock inputs that are used internally to generate all of the clocks required by the internal switch logic and the SerDes.

Chapter 3, "Link Operation," describes the operation of the link feature including polarity inversion, link width negotiation, and lane reversal.

Chapter 4, "Switch Operation," discusses the procedure for forwarding PCIe® TLPs between switch ports.

Chapter 5, "Power Management," describes the power management capability structure located in the configuration space of each PCI-PCI bridge in the PES12N3.

Chapter 6, "Hot-Plug and Hot-Swap," describes the behavior of the hot-plug and hot-swap features in the PES12N3.

Chapter 7, "SMBus Interfaces," describes the operation of the 2 SMBus interfaces on the PES12N3.

Chapter 8, "General Purpose I/O," describes how the eight General Purpose I/O (GPIO) pins may be individually configured as general purpose inputs, general purpose outputs, or alternate functions

Chapter 9, "Transparent Mode Operation," describes how the PES12N3 can be configured during a fundamental reset to operate in transparent mode or transparent mode with serial EEPROM initialization.

Chapter 10, "Test and Debug," discusses the six test modes, in addition to the normal operating mode, associated with the PES12N3.

Chapter 11, "JTAG Boundary Scan," discusses an enhanced JTAG interface, including a system logic TAP controller, signal definitions, a test data register, an instruction register, and usage considerations.

Signal Nomenclature

To avoid confusion when dealing with a mixture of "active-low" and "active-high" signals, the terms assertion and negation are used. The term assert or assertion is used to indicate that a signal is active or true, independent of whether that level is represented by a high or low voltage. The term negate or negation is used to indicate that a signal is inactive or false.

Notes

To define the active polarity of a signal, a suffix will be used. Signals ending with an 'N' should be interpreted as being active, or asserted, when at a logic zero (low) level. All other signals (including clocks, buses and select lines) will be interpreted as being active, or asserted when at a logic one (high) level. To define buses, the most significant bit (MSB) will be on the left and least significant bit (LSB) will be on the right. No leading zeros will be included.

Throughout this manual, when describing signal transitions, the following terminology is used. Rising edge indicates a low-to-high (0 to 1) transition. Falling edge indicates a high-to-low (1 to 0) transition. These terms are illustrated in Figure 1.

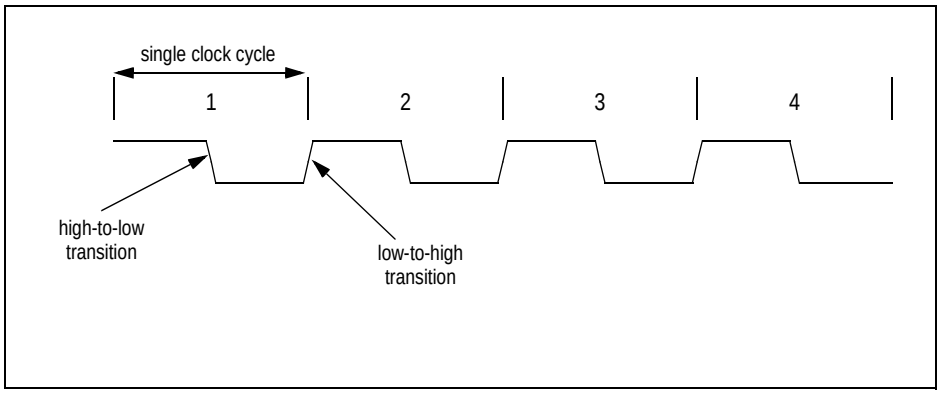

Figure 1 Signal Transitions

Numeric Representations

To represent numerical values, either decimal, binary, or hexadecimal formats will be used. The binary format is as follows: 0bDDD, where "D" represents either 0 or 1; the hexadecimal format is as follows: 0xDD, where "D" represents the hexadecimal digit(s); otherwise, it is decimal.

The compressed notation ABC[x|y|z]D refers to ABCxD, ABCyD, and ABCzD.

The compressed notation ABC[x..y]D refers to ABCxD, ABC(x+1)D, ABC(x+2)D,... ABCyD.

Data Units

The following data unit terminology is used in this document.

Term	Words	Bytes	Bits
Byte	1/2	1	8
Word	1	2	16
Doubleword (Dword)	2	4	32
Quadword (Qword)	4	8	64

Table 1 Data Unit Terminology

In quadwords, bit 63 is always the most significant bit and bit 0 is the least significant bit. In doublewords, bit 31 is always the most significant bit and bit 0 is the least significant bit. In words, bit 15 is always the most significant bit and bit 0 is the least significant bit. In bytes, bit 7 is always the most significant bit and bit 0 is the least significant bit.

The ordering of bytes within words is referred to as either "big endian" or "little endian." Big endian systems label byte zero as the most significant (leftmost) byte of a word. Little endian systems label byte zero as the least significant (rightmost) byte of a word. See Figure 2.

Notes

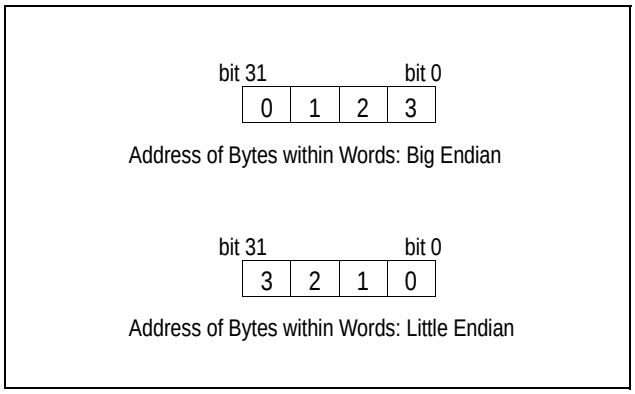

Figure 2 Example of Byte Ordering for "Big Endian" or "Little Endian" System Definition

Register Terminology

Software in the context of this register terminology refers to modifications made by PCIe root configuration writes, writes to registers made through the slave SMBus interface, or serial EEPROM register initialization. See Table 2.

Type	Abbreviation	Description
Hardware Initialized	HWINIT	Register bits are initialized by firmware or hardware mechanisms such as pin strapping or serial EEPROM. (System firmware hardware initialization is only allowed for system integrated devices.) Bits are read-only after initialization and can only be reset (for write-once by firmware) with reset.
Read Only and Clear	RC	Software can read the register/bits with this attribute. Reading the value will automatically cause the register/bit to be reset to zero. Writing to a RC location has no effect.
Read Clear and Write	RCW	Software can read the register/bits with this attribute. Reading the value will automatically cause the register/bits to be reset to zero. Writes cause the register/bits to be modified.
Reserved	Reserved	The value read from a reserved register/bit is undefined. Thus, software must deal correctly with fields that are reserved. On reads, software must use appropriate masks to extract the defined bits and not rely on reserved bits being any particular value. On writes, software must ensure that the values of reserved bit positions are preserved. That is, the values of reserved bit positions must first be read, merged with the new values for other bit positions and then written back.
Read Only	RO	Software can only read registers/bits with this attribute. Contents are hardwired. Writing to a RO location has no effect.
Read Only and set by Hardware	ROS	Software can only read registers/bits with this attribute. Contents are set by hardware and may change. Writing to a ROS location has no effect.
Read and Write	RW	Software can both read and write bits with this attribute.

Table 2 Register Terminology (Sheet 1 of 2)

Notes

Type	Abbreviation	Description
Read and Write Clear	RW1C	Software can read and write to registers/bits with this attribute. However, writing a value of zero to a bit with this attribute has no effect. A RW1C bit can only be set to a value of 1 by a hardware event. To clear a RW1C bit (i.e., change its value to zero) a value of one must be written to the location. An RW1C bit is never cleared by hardware.
Read and Write when Unlocked	RWL	Software can read the register/bits with this attribute. Writing to register/bits with this attribute will only cause the value to be modified if the REGUNLOCK bit in the SWCNTL register is set. When the REGUNLOCK bit is cleared, writes are ignored and the register/bits are effectively read-only
Zero	Zero	A zero register or bit must be written with a value of zero and returns a value of zero when read.

Table 2 Register Terminology (Sheet 2 of 2)

Use of Hypertext

In Chapter 9 there are tables which contain register names and page numbers highlighted in blue under the Register Definition column. In pdf files, users can jump from the source table directly to the registers by clicking on the register name in the source table. Each register name in the table is linked directly to the appropriate register in the register section of the chapter. To return to the source table after having jumped to the register section, click on the same register name (in blue) in the register section.

Revision History

February 8, 2006: Initial Publication.

June 7, 2006: Added revision YC information in Chapters 1 and 9.

Table of Contents

Notes

Notes

Notes

Notes

List of Tables

Notes

Notes

List of Figures

Notes

Notes

Register List

Notes

Notes

PES12N3 Device Overview

Notes

Introduction

The 89HPES12N3 is a member of IDT's PRECISE™ family of PCI Express® switching solutions offering the next-generation I/O interconnect standard. The PES12N3 is a 12 lane, 3-port peripheral chip that performs PCI Express Base switching with a feature set optimized for high performance applications such as servers, storage, and communications/networking. It provides high-performance I/O connectivity and switching functions between a PCI Express upstream port and two downstream ports or peer-to-peer switching between downstream ports.

Utilizing standard PCI Express interconnect, the PES12N3 provides the most efficient high-performance I/O connectivity solution for applications requiring high throughput, low latency, and simple board layout with a minimum number of board layers. It provides 6 GBps (48 Gbps) of aggregate switching capacity through 12 integrated serial lanes, using proven and robust IDT technology. Each lane provides 2.5 Gbps of bandwidth in both directions. The PES12N3 is fully compliant with PCI Express Base specification 1.0a.

The PES12N3 is based on a flexible and efficient layered architecture. The PCI Express layer consists of SerDes, Physical, Data Link and Transaction layers in compliance with PCI Express Base specification Revision 1.0a. The PES12N3 can operate either as a store and forward or cut-through switch and is designed to switch memory and I/O transactions. It supports eight Traffic Classes (TCs) and one Virtual Channel (VC) with sophisticated resource management. This includes system selectable algorithms such as round robin and weighted round-robin schemes guaranteeing bandwidth allocation and/or latency for critical traffic classes in applications such as high throughput 10 Gigabit I/Os, streaming media for graphics, TV tuners, and cameras.

Figure 1.1 provides a functional block diagram while Figure 1.2 illustrates the architecture of the device.

Notes

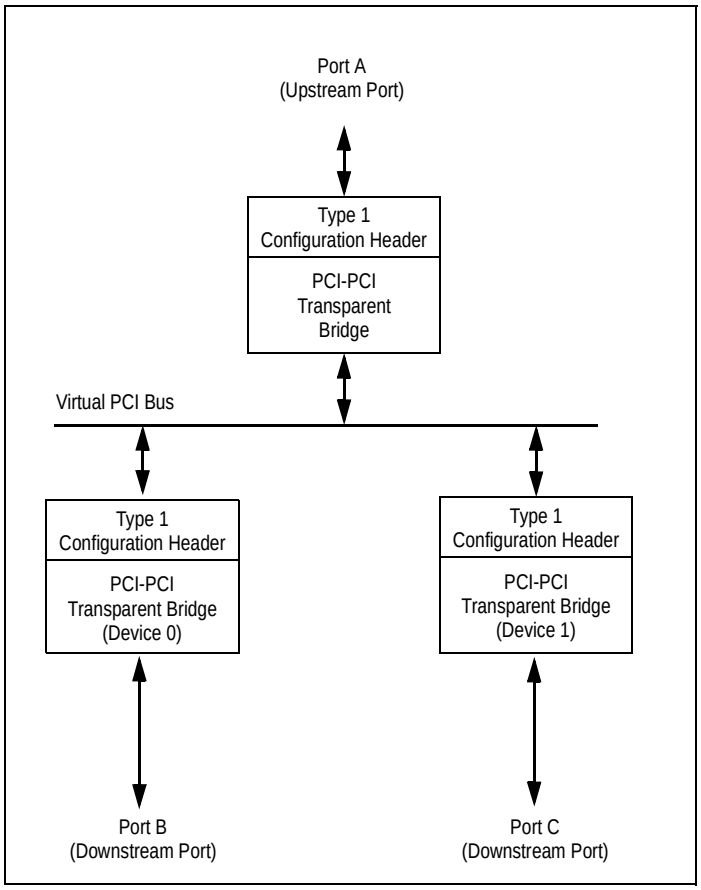

Figure 1.1 PES12N3 Functional Block Diagram

As shown in Figure 1.1, port A is configured as the upstream port and ports B and C as the downstream ports. Port B resides on the internal PCI Bus at Device 0, Function 0. Port C resides on the internal PCI Bus at Device 1, Function 0.

Notes

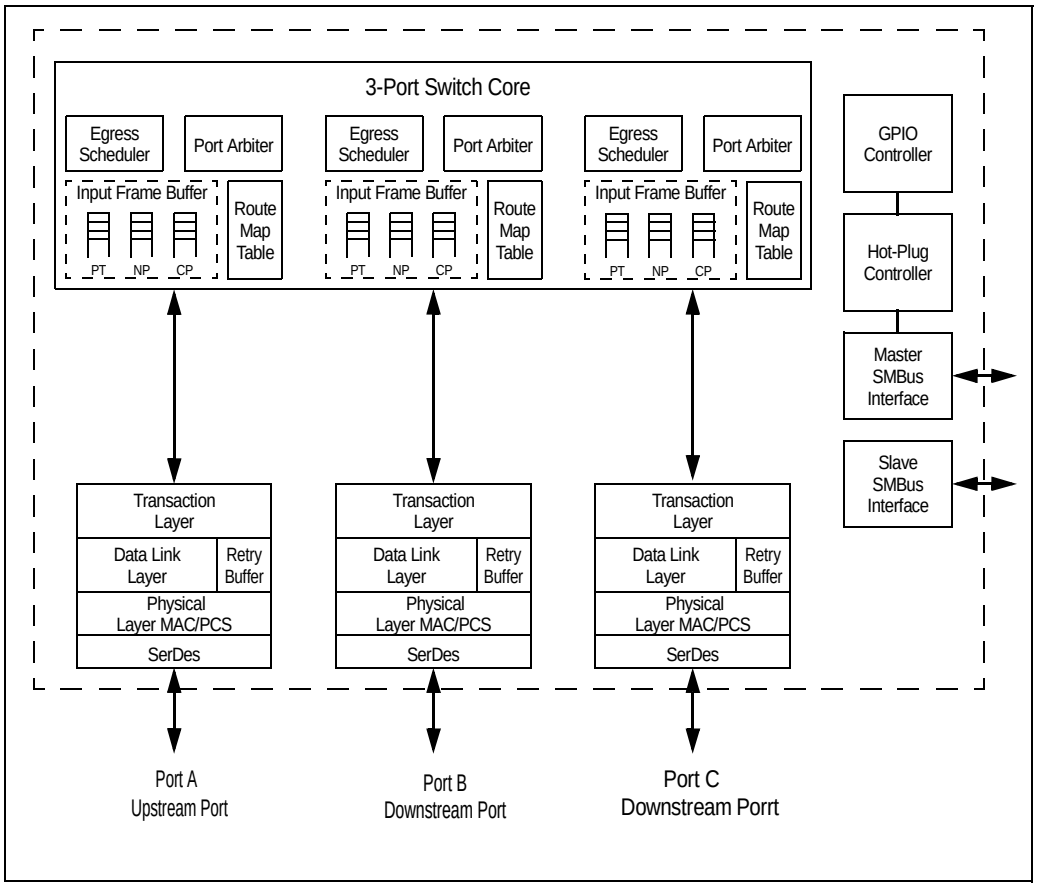

Figure 1.2 PES12N3 Architectural Block Diagram

Features

- **High Performance PCI Express Switch**
  - *Three x4 ports with 12 PCI Express lanes total*
  - *Delivers 6 GBps (48 Gbps) aggregate switching capacity*
  - *Low latency cut-through switch architecture*
  - *Supports 128 to 2048 byte maximum payload size*
  - *Supports one virtual channel*
  - *PCI Express Base specification Revision 1.0a compliant*
- **Flexible Architecture with Numerous Configuration Options**
  - *Port arbitration schemes utilizing round robin or weighted round robin algorithms*
  - *Supports automatic per port link with negotiation (x4, x2, or x1)*
  - *Supports static lane reversal on all ports*
  - *Supports polarity inversion*
  - *Supports locked transactions, allowing use with legacy software*
  - *Ability to load device configuration from serial EEPROM*
- **Highly Integrated Solution**
  - *Requires no external components*
  - *Incorporates on-chip internal memory for packet buffering and queueing*
  - *Integrates 12 2.5 Gbps embedded SerDes, 8B/10B encoder/decoder (no separate transceivers needed)*
- **Reliability, Availability, and Serviceability (RAS) Features**
  - *Internal end-to-end parity protection on all TLPs ensures data integrity even in systems that do not implement end-to-end CRC (ECRC)*

Notes

- *Supports ECRC passed through*
- *Supports PCI Express Native Hot-Plug*
  - *Compatible with Hot-Plug I/O expanders used on PC motherboards*
- *Supports Hot-Swap*

◆ **Power Management**

- *Supports PCI Express Power Management Interface specification, Revision 1.1 (PCI-PM)*
- *Unused SerDes are disabled*
- *Supports Advanced Configuration and Power Interface Specification, Revision 2.0 (ACPI) supporting active link state*

◆ **Testability and Debug Features**

- *Supports IEEE 1149.6 JTAG*
- *Built in SerDes Pseudo-Random Bit Stream (PRBS) generator*
- *Ability to read and write any internal register via the SMBus*
- *Ability to bypass link training and force any link into any mode*
- *Provides statistics and performance counters*

◆ **Two SMBus Interfaces**

- *Slave interface provides full access to all software-visible registers by an external SMBus master*
- *Master interface provides connection for an optional serial EEPROM used for initialization*
- *Master interface is also used by an external Hot-Plug I/O expander*
- *Master and slave interfaces may be tied together so the PES12N3 can act as both master and slave*

◆ **8 General Purpose Input/Output pins**

◆ **Packaged in 19x19mm 324 ball BCG with 1mm ball spacing**

System Identification

Vendor ID

All vendor IDs in the device are hardwired to 0x111D which corresponds to Integrated Device Technology, Inc.

Device ID

The device IDs for the PES12N3 are shown in Table 1.1.

PCI Device	Offset Device ID
Transparent bridge associated with Ports A, B, and C	0x8018

Table 1.1 PES12N3 Offset Device IDs

Notes

Revision ID

All revision IDs in the PES12N3 are set to the same value. The value of the revision ID is determined in one place and is easily modified during a metal mask change. The revision ID shall be incremented with each all layer or metal mask change.

Revision ID	Description
0x1	Corresponds to YA silicon
0x2	Corresponds to YB silicon
0x4	Corresponds to YC silicon

Table 1.2 PES12N3 Revision IDs

JTAG ID

The JTAG ID is:

- Version: Same value as Revision ID. See the Revision ID section above.
- Part number: Same value as base Device ID. See the Device ID section above.
- Manufacture ID: 0x33
- LSB: 0x1

Notes

Logic Diagram

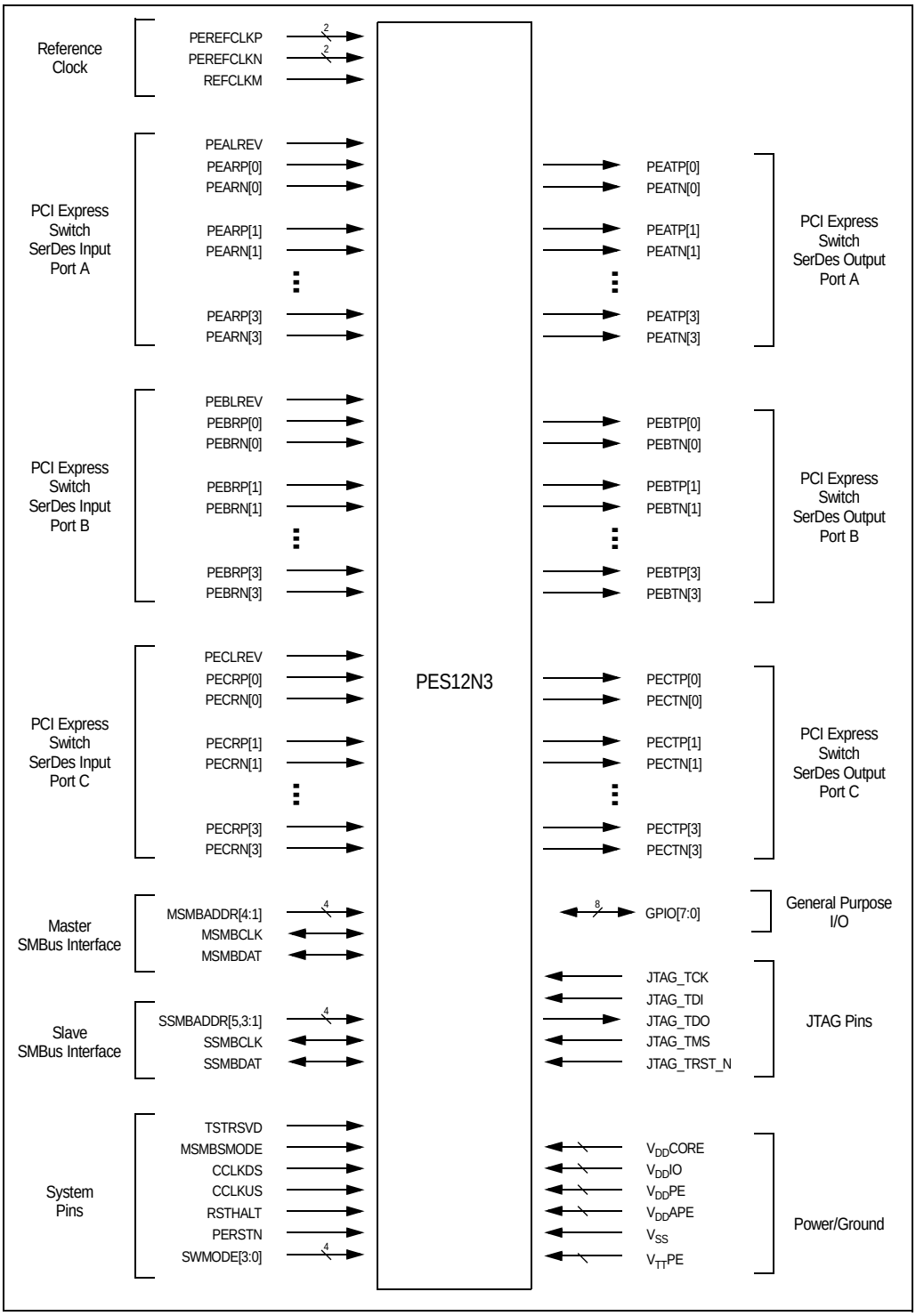

Figure 1.3 PES12N3 Logic Diagram

Notes

Pin Description

The following tables lists the functions of the pins provided on the PES12N3. Some of the functions listed may be multiplexed onto the same pin.

The active polarity of a signal is defined using a suffix. Signals ending with an “N” are defined as being active, or asserted, when at a logic zero (low) level. All other signals (including clocks, buses, and select lines) will be interpreted as being active, or asserted, when at a logic one (high) level.

Signal	Type	Name/Description
PEALREV	I	**PCI Express Port A Lane Reverse.** When this bit is asserted, the lanes of PCI Express Port A are reversed. This value may be overridden by modifying the value of the PALREV bit in the PA_SWCTL register.
PEARP[3:0] PEARN[3:0]	I	**PCI Express Port A Serial Data Receive.** Differential PCI Express receive pairs for port A.
PEATP[3:0] PEATN[3:0	O	**PCI Express Port A Serial Data Transmit.** Differential PCI Express transmit pairs for port A
PEBLREV	I	**PCI Express Port B Lane Reverse.** When this bit is asserted, the lanes of PCI Express Port B are reversed. This value may be overridden by modifying the value of the PBLREV bit in the PA_SWCTL register.
PEBRP[3:0] PEBRN[3:0]	I	**PCI Express Port B Serial Data Receive.** Differential PCI Express receive pairs for port B.
PEBTP[3:0] PEBTN[3:0]	O	**PCI Express Port B Serial Data Transmit.** Differential PCI Express transmit pairs for port B
PECLREV	I	**PCI Express Port C Lane Reverse.** When this bit is asserted, the lanes of PCI Express Port C are reversed. This value may be overridden by modifying the value of the PCLREV bit in the PA_SWCTL register.
PECRP[3:0] PECRN[3:0]	I	**PCI Express Port C Serial Data Receive.** Differential PCI Express receive pairs for port C.
PECTP[3:0] PECTN[3:0]	O	**PCI Express Port C Serial Data Transmit.** Differential PCI Express transmit pairs for port C
REFCLKP[1:0] REFCLKN[1:0]	I	**PCI Express Reference Clock.** Differential reference clock pair input. This clock is used as the reference clock by on-chip PLLs to generate the clocks required for the system logic and on-chip SerDes. The frequency of the differential reference clock is determined by the REFCLKM signal.
REFCLKM	I	**PCI Express Reference Clock Mode Select.** These signals select the frequency of the reference clock input. *0x0 - 100 MHz* *0x1 - 125 MHz*

Table 1.3 PCI Express Interface Pins

Notes

Signal	Type	Name/Description
MSMBADDR[4:1]	I	**Master SMBus Address.** These pins determine the SMBus address of the serial EEPROM from which configuration information is loaded.
MSMBCLK	I/O	**Master SMBus Clock.** This bidirectional signal is used to synchronize transfers on the master SMBus.
MSMBDAT	I/O	**Master SMBus Data.** This bidirectional signal is used for data on the master SMBus.
SSMBADDR[5,3:1]	I	**Slave SMBus Address.** These pins determine the SMBus address to which the slave SMBus interface responds.
SSMBCLK	I/O	**Slave SMBus Clock.** This bidirectional signal is used to synchronize transfers on the slave SMBus.
SSMBDAT	I/O	**Slave SMBus Data.** This bidirectional signal is used for data on the slave SMBus.

Table 1.4 SMBus Interface Pins

Signal	Type	Name/Description
GPIO[0]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin.
GPIO[1]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin.
GPIO[2]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: IOEXPINTN Alternate function pin type: Input Alternate function: Hot-Plug I/O expander interrupt input
GPIO[3]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: PAABN Alternate function pin type: Input Alternate function: Port A attention button Input
GPIO[4]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: PAAIN Alternate function pin type: Output Alternate function: Port A attention indicator output
GPIO[5]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: PAPIN Alternate function pin type: Output Alternate function: Port A power indicator output
GPIO[6]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin.
GPIO[7]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin.

Table 1.5 General Purpose I/O Pins

Notes

Signal	Type	Name/Description
CCLKDS	I	**Common Clock Downstream.** The assertion of this pin indicates that all downstream ports are using the same clock source as that provided to downstream devices.This bit is used as the initial value of the Slot Clock Configuration bit in all of the Link Status Registers for downstream ports. The value may be override by modifying the SCLK bit in the PB_PCIELSTS or PC_PCIELSTS register.
CCLKUS	I	**Common Clock Upstream.** The assertion of this pin indicates that the upstream port is using the same clock source as the upstream device. This bit is used as the initial value of the Slot Clock Configuration bit in the Link Status Register for the upstream port. The value may be overridden by modifying the SCLK bit in the PA_PCIELSTS register.
MSMBSMODE	I	**Master SMBus Slow Mode.** The assertion of this pin indicates that the master SMBus should operate at 100 KHz instead of 400 kHz. This value may not be overridden.
PERSTN	I	**Fundamental Reset.** Assertion of this signal resets all logic inside the PES12N3 and initiates a PCI Express fundamental reset.
RSTHALT	I	**Reset Halt.** When this signal is asserted during a PCI Express fundamental reset, the PES12N3 executes the reset procedure and remains in a reset state with the Master and Slave SMBuses active. This allows software to read and write registers internal to the device before normal device operation begins. The device exits the reset state when the RSTHALT bit is cleared in the PA_SWCTL register by an SMBus master.
TSTRSVD	I	**Reserved.** Reserved for future test mode. Must be tied to ground.
SWMODE[3:0]	I	**Switch Mode.** These configuration pins determine the PES12N3 switch operating mode. *0x0 - Transparent mode* *0x1 -Transparent mode with serial EEPROM initialization* *0x2 through 0x7 - Reserved* *0x8 - 10-bit loopback test mode* *0x9 - Reserved* *0xA - Internal pseudo random bit stream self-test test mode* *0xB - External pseudo random bit stream self-test test mode* *0xC - Reserved* *0xD - SerDes broadcast test mode* *0xE - 0xF Reserved*

Table 1.6 System Pins

Notes

Signal	Type	Name/Description
JTAG_TCK	I	**JTAG Clock**. This is an input test clock used to clock the shifting of data into or out of the boundary scan logic or JTAG Controller. JTAG_TCK is independent of the system clock with a nominal 50% duty cycle.
JTAG_TDI	I	**JTAG Data Input**. This is the serial data input to the boundary scan logic or JTAG Controller.
JTAG_TDO	O	**JTAG Data Output**. This is the serial data shifted out from the boundary scan logic or JTAG Controller. When no data is being shifted out, this signal is tri-stated.
JTAG_TMS	I	**JTAG Mode**. The value on this signal controls the test mode select of the boundary scan logic or JTAG Controller.
JTAG_TRST_N	I	**JTAG Reset**. This active low signal asynchronously resets the boundary scan logic and JTAG TAP Controller. An external pull-up on the board is recommended to meet the JTAG specification in cases where the tester can access this signal. However, for systems running in functional mode, one of the following should occur: 1) actively drive this signal low with control logic 2) statically drive this signal low with an external pull-down on the board

Table 1.7 Test Pins

Signal	Type	Name/Description
V_{DD}CORE	I	**Core V_{DD}.** Power supply for core logic.
V_{DD}IO	I	**I/O V_{DD}.** LVTTL I/O buffer power supply.
V_{DD}PE	I	**PCI Express Digital Power.** PCI Express digital power used by the digital power of the SerDes.
V_{DD}APE	I	**PCI Express Analog Power.** PCI Express analog power used by the PLL and bias generator.
V_{TT}PE	I	**PCI Express Termination Power.**
V_{SS}	I	Ground.

Table 1.8 Power and Ground Pins

Notes

Pin Characteristics

Some input pads of the PES12N3 do not contain internal pull-ups or pull-downs. Unused inputs should be tied off to appropriate levels. This is especially critical for unused control signal inputs which, if left floating, could adversely affect operation. Also, any input pin left floating can cause a slight increase in power consumption.

Function	Pin Name	Type	Buffer	I/O Type	Internal Resistor	Notes
PCI Express Interface	PEALREV	I	LVTTL	Input	pull-down	
	PEARN[3:0]	I	LVDS	Serial link		
	PEARP[3:0]	I	LVDS	Serial link		
	PEATN[3:0]	O	LVDS	Serial link		
	PEATP[3:0]	O	LVDS	Serial link		
	PEBLREV	I	LVTTL	Input	pull-down	
	PEBRN[3:0]	I	LVDS	Serial link		
	PEBRP[3:0]	I	LVDS	Serial link		
	PEBTN[3:0]	O	LVDS	Serial link		
	PEBTP[3:0]	O	LVDS	Serial link		
	PECLREV	I	LVTTL	Input	pull-down	
	PECRN[3:0]	I	LVDS	Serial link		
	PECRP[3:0]	I	LVDS	Serial link		
	PECTN[3:0]	O	LVDS	Serial link		
	PECTP[3:0]	O	LVDS	Serial link		
	PEREFCLKN[1:0]	I	LVDS	Serial link		
	PEREFCLKP[1:0]	I	LVDS	Serial link		
	REFCLKM	I	LVTTL	Input	pull-down	
SMBus	MSMBADDR[4:1]	I	LVTTL	Input	pull-up	
	MSMBCLK	I/O	LVTTL	STI		
	MSMBDAT	I/O	LVTTL	STI		
	SSMBADDR[5,3:1]	I	LVTTL	Input	pull-up	
	SSMBCLK	I/O	LVTTL	STI		
	SSMBDAT	I/O	LVTTL	STI		
General Purpose I/O	GPIO[7:0]	I/O	LVTTL	Input, High Drive	pull-up	

Table 1.9 Pin Characteristics (Part 1 of 2)

Notes

Function	Pin Name	Type	Buffer	I/O Type	Internal Resistor	Notes
System Pins	CCLKDS	I	LVTTL	Input	pull-up	
	CCLKUS	I	LVTTL	Input	pull-up	
	MSMBSMODE	I	LVTTL	Input	pull-down	
	PERSTN	I	LVTTL	Input		
	RSTHALT	I	LVTTL	Input	pull-down	
	TSTRSVD	I	LVTTL	Input	pull-down	External pull-down
	SWMODE[3:0]	I	LVTTL	Input	pull-up	
JTAG	JTAG_TCK	I	LVTTL	STI	pull-up	
	JTAG_TDI	I	LVTTL	STI	pull-up	
	JTAG_TDO	O	LVTTL	Low Drive		
	JTAG_TMS	I	LVTTL	STI	pull-up	
	JTAG_TRST_N	I	LVTTL	STI	pull-up	External pull-down

Table 1.9 Pin Characteristics (Part 2 of 2)

Clocking, Reset, and Initialization

Notes

Introduction

The PES12N3 has two differential reference clock inputs that are used internally to generate all of the clocks required by the internal switch logic and the SerDes. While not required, it is recommended that both reference clock input pairs be driven from a common clock source. There are no skew requirements between the reference clock inputs.The frequency of the reference clock inputs may be selected by the Reference Clock Mode Select (REFCLKM) input.

REFCLKM	Description
0	100 MHz reference clock input.
1	125 MHz reference clock input.

Table 2.1 Reference Clock Mode Encoding

Each of the reference clock differential inputs feeds six on-chip PLLs. Each PLL generates a 2.5 GHz clock which is used by four SerDes lanes and produces a 250 MHz core clock.The 250 MHz core clock output from one of the six internal PLLs is used as the system clock for internal switch logic.

Initialization

A boot configuration vector consisting of the signals listed in Table 2.2 is sampled by the PES12N3 during a fundamental reset when PERSTN is negated. The boot configuration vector defines essential parameters for switch operation.

While basic switch operation may be configured using signals in the boot configuration vector, advanced switch features require configuration via an external serial EEPROM. The external serial EEPROM allows modification of any bit in any software visible register. See Chapter 7, SMBus Interfaces, for more information on the serial EEPROM.

The external serial EEPROM and slave SMBus interface may be used to override the function of some of the signals in the boot configuration vector during a fundamental reset. The signals that may be overridden are noted in Table 2.2.

The state of all of the boot configuration signals in Table 2.2 sampled during the most recent cold reset may be determined by reading the PA_SWSTS register.

Notes

Signal	May Be Overridden	Description
CCLKDS	Y	**Common Clock Downstream.** The assertion of this pin indicates that all downstream ports are using the same clock source as that provided to downstream devices.This pin is used as the initial value of the Slot Clock Configuration bit in all of the Link Status Registers for downstream ports. The value may be overridden by modifying the SCLK bit in the PB_PCIELSTS or PC_PCIELSTS register.
CCLKUS	Y	**Common Clock Upstream.** The assertion of this pin indicates that the upstream port is using the same clock source as the upstream device. This pin is used as the initial value of the Slot Clock Configuration bit in the Link Status Register for the upstream port. The value may be overridden by modifying the SCLK bit in the PA_PCIELSTS register.
MSMBSMODE	N	**Master SMBus Slow Mode.** The assertion of this pin indicates that the master SMBus should operate at 100 KHz instead of 400 KHz.
PEALREV	Y	**PCI Express Port A Lane Reverse.** When this pin is asserted, the lanes of PCI Express Port A are reversed. This value may be overridden by modifying the value of the PALREV bit in the PA_SWCTL register.
PEBLREV	Y	**PCI Express Port B Lane Reverse.** When this pin is asserted, the lanes of PCI Express Port B are reversed. This value may be overridden by modifying the value of the PBLREV bit in the PA_SWCTL register.
PECLREV	Y	**PCI Express Port C Lane Reverse.** When this pin is asserted, the lanes of PCI Express Port C are reversed. This value may be overridden by modifying the value of the PCLREV bit in the PA_SWCTL register.
REFCLKM	N	**PCI Express Reference Clock Mode Select.** These signals select the frequency of the reference clock input. 0x0 - 100 MHz 0x1 - 125 MHz

Table 2.2 Boot Configuration Vector Signals (Part 1 of 2)

Notes

Signal	May Be Overridden	Description
RSTHALT	Y	**Reset Halt.** When this signal is asserted during a PCI Express fundamental reset, the PES12N3 executes the reset procedure and remains in a reset state with the Master and Slave SMBuses active. This allows software to read and write registers internal to the device before normal device operation begins. The device exits the reset state when the RSTHALT bit is cleared in the PA_SWCTL register through the SMBus. The value may be overridden by modifying the RSTHALT bit in the PA_SWCTL register.
TSTRSVD	N	**Reserved.** Reserved for future test mode. Must be tied to ground.
SWMODE[3:0]	N	**Switch Mode.** These configuration pins determine the PES12N3 switch operating mode. 0x0 - Transparent mode 0x1 -Transparent mode with serial EEPROM initialization 0x2 through 0x7 - Reserved 0x8 - 10-bit loopback test mode 0x9 - Reserved 0xA - Internal pseudo random bit stream self-test test mode 0xB - External pseudo random bit stream self-test test mode 0xC - Reserved 0xD - SerDes broadcast test mode 0xE - Reserved 0xF - Reserved

Table 2.2 Boot Configuration Vector Signals (Part 2 of 2)

Reset

PCI Express® defines two reset categories: fundamental reset and hot reset. A fundamental reset causes all associated logic to be returned to an initial state. A hot reset causes all associated logic to be returned to an initial state, but does not cause the state of register fields denoted as “sticky” to be modified.

There are two sub-categories of fundamental reset: cold reset and warm reset. A cold reset occurs following a device being powered on and assertion of PERSTN. A warm reset is a fundamental reset that occurs without removal of power.

A summary of reset conditions and their effect is exhibited in Table 2.3.

	Fund. Reset	Global Hot Reset to Entire Device	Global Hot Reset to Downstream Ports	Local Hot Reset
Master SMBus	Y	N	N	N
Slave SMBus	Y	N	N	N
Serial EEPROM Initialization	Y if mode requires it	N	N	N

Table 2.3 Reset Conditions and Their Effect (Part 1 of 2)

Notes

	Fund. Reset	Global Hot Reset to Entire Device	Global Hot Reset to Downstream Ports	Local Hot Reset
Switch Core	Y	Y	N (flush buffer only)	N
Port A All Registers	Y	N	N	N
Port A All Registers Except Those of Type Sticky or RWL	Y	Y	N	N
Port A Transaction Layer	Y	Y	N	N
Port A Data Link Layer	Y	Y	N	N
Port A Phy Layer	Y	Y	N	N
Port B All Registers	Y	N	N	N
Port B All Registers Except Those of Type Sticky or RWL	Y	Y	Y	N
Port B Transaction Layer	Y	Y	Y	N
Port B Data Link Layer	Y	Y	Y	Y if selected
Port B Phy Layer	Y	Y	Y	Y if selected
Port B Downstream Hot Reset Req.	N	Y	Y	Y if selected
Port C All Registers	Y	N	N	N
Port C All Registers Except Those of Type Sticky or RWL	Y	Y	Y	N
Port C Transaction Layer	Y	Y	Y	N
Port C Data Link Layer	Y	Y	Y	Y if selected
Port C Phy Layer	Y	Y	Y	Y if selected
Port C Downstream Hot Reset Req.	N	Y	Y	Y if selected

Table 2.3 Reset Conditions and Their Effect (Part 2 of 2)

Notes

Fundamental Reset

A fundamental reset of the entire device may be initiated by one of three conditions:

- A cold reset initiated by a power-on and the assertion of the PCI Express Reset (PERSTN) input pin.
- A warm reset initiated by the assertion of the PCI Express Reset (PERSTN) input pin while power is on.
- A warm reset initiated by the writing of a one to the Reset (RST) bit in the Port A Switch Control (PA_SWCTL) register.

When the device is configured to operate in a test mode, then the reset sequence described in section Device Test Modes on page 10-1 is executed. Otherwise, when configured to operate in normal mode, the following reset sequence is executed.

1. Wait for the fundamental reset condition to clear (e.g., negation of PERSTN).
2. On negation of PERSTN, sample the boot configuration signals listed in Table 2.2.
3. On negation of PERSTN, the SWMODE[3:0] signals are examined to determine the switch operating mode.
4. The PLL is initialized.
5. SerDes CDR Locking and Link training begins. While this is in progress, proceed to step 6.
6. If the Reset Halt (RSTHALT) pin is asserted, the RSTHALT bit in the PA_SWSTS register is set.
7. If the switch operating mode is not a test mode, then the reset to the PCI Express stacks and associated logic is negated but they are held in a quasi-reset state in which the following actions occur.

   All links enter an active Link Training state within 80ms of the clearing of the fundamental reset condition.

   Within 100ms of the clearing of the fundamental reset condition, all of the stacks are able to process configuration transactions and respond to these transactions with a configuration request retry status completion. All other transactions are ignored.
8. The master SMBus operating frequency is initialized.

   The state of the MSMBSMODE signal is examined. If it is asserted, then the master SMBus is initialized to operate at 100 KHz rather than 400 KHz.
9. The slave SMBus is taken out of reset and initialized.
10. The master SMBus is taken out of reset and initialized.
11. If the selected switch operating mode is one that requires initialization from the serial EEPROM, then the contents of the serial EEPROM are read and the appropriate PES12N3 registers are updated.

    If a one is written by the serial EEPROM to the Link Retrain (LRET) bit in any PCI Express Link Control (PCIELCTL) register, then link retraining is initiated on that port using the current link parameters. Note that link retraining may be forced on the upstream port when the REGUNLOCK bit is set in the PA_SWCTL register.

    If an error is detected during loading of the serial EEPROM, then loading of the serial EEPROM is aborted and the RSTHALT bit is set in the PA_SWCTL register. Error information is recorded in the PA_SMBUSSTS register.

    When serial EEPROM initialization completes or when an error is detected, the DONE bit in the

Notes

PA_SMBUSSTS register is set.

12. Wait for link training on all ports to complete or fail.[1]
13. If the Reset Halt (RSTHALT) bit is set in the PA_SWCTL register, all of the logic is held in a reset state except the master and slave SMBuses, the control/status registers, and the stacks which continue to be held in a quasi-reset state and respond to configuration transactions with a retry. The device remains in this state until the RSTHALT bit is cleared via the slave SMBus. In this mode, an external agent may read and write any internal control and status registers and may access the external serial EEPROM via the PA_EEPROMINTF register.
14. Normal device operation begins.

The PCIe® standard specifies that normal operation should begin within 1.0 second after a fundamental reset of a device. The reset sequence above guarantees that normal operation will begin within this period as long as the serial EEPROM initialization process completes within 200 ms. Under normal circumstances, 200 ms is more than adequate to initialize every register in the device even with a Master SMBus operating frequency of 100 KHz.

Serial EEPROM initialization may cause writes to register fields that initiate side effects such as link retraining. These side effects are initiated at the point at which the write occurs. Therefore, serial EEPROM initialization should be structured in a manner so as to ensure proper configuration prior to initiation of these side effects.

The operation of a fundamental reset in Transparent mode with serial EEPROM initialization (i.e., SWMODE[3:0] = 0x1) is illustrated in Figure 2.1.

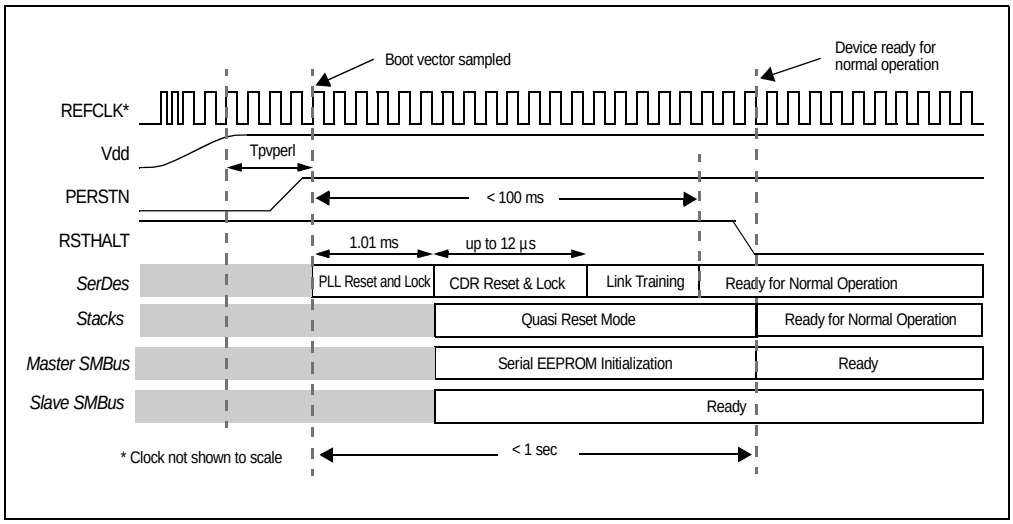

Figure 2.1 Fundamental Reset in Transparent Mode with Serial EEPROM initialization

Hot Reset

A hot reset may be initiated globally to the entire device, globally to downstream ports or locally to downstream port(s).

Globally Initiated Hot Reset To Entire Device

A hot reset is initiated globally to the entire device when any of the following conditions occur.

- Reception of TS1 ordered-sets on the upstream port indicating a hot reset.
- Data link layer of the upstream port transitions to the DL_Down state.

1. While link training is in progress, a stack responds to configuration retry requests with a configuration request retry status completion and ignores all other transactions. This process stops when link training successfully completes, a device is not detected on the link, or when link training fails 16 times.

Notes

Hot reset is only propagated downstream. TS1 ordered-sets indicating a hot reset received on a downstream port do not result in a hot reset of the downstream port or any function inside the switch.

When a globally initiated hot reset occurs, all of the logic associated with the transparent bridges, stacks and the switch core are reset except for the PLLs, SerDes, master SMBus interface, slave SMBus interface, and some registers. Regardless of the switch operating mode, a hot-reset does not result in reloading of the serial EEPROM.

The value of register fields denoted as “sticky” or as Read and Write when Unlocked (i.e., RWL) are preserved in all ports across a hot-reset. All other register fields in all ports are reset to their initial values.

When a hot reset is initiated globally, each downstream port shall send a hot-reset message to its link partner prior to being reset.

Globally Initiated Hot Reset To Downstream Ports

A hot reset is initiated globally to downstream ports when the following condition occurs

– A one is written to the Secondary Bus Reset (SRESET) bit in the upstream port’s (port A) Bridge Control Register (BCTRL). See BCTRL - Bridge Control (0x03E) on page 9-25.

When a globally initiated hot reset is initiated to downstream ports, all of the logic associated with the transparent bridges, stacks and FIFOs in the switch core associated with the downstream ports are reset except for the PLLs, SerDes, master SMBus interface, slave SMBus interface, and some registers. Regardless of the switch operating mode, it does not result in reloading of the serial EEPROM.

The value of register fields denoted as “sticky” or as Read and Write when Unlocked (i.e., RWL) in downstream ports are preserved. All other register fields are reset to their initial values.

When a hot reset is initiated globally to downstream ports, each downstream port shall send a hot-reset message to its link partner prior to being reset.

Unlike a globally initiated hot reset to the entire device, a globally initiated hot reset to downstream ports does not affect the state of the upstream port’s configuration register except those required to update port status.

Locally Initiated Hot Reset to a Downstream Port

A hot reset is initiated locally to a downstream port by writing to the SRESET bit of a downstream port’s BCTRL registers. When this occurs, a hot-reset message is sent on that port to its link partner. After the message is sent, the phy layer is effectively reset.

A locally initiated hot reset does not affect the state of any port (i.e., transparent bridge) configuration register except those required to update port status.

Notes

Link Operation

Notes

Introduction

The PES12N3 contains three ports. The default link width of each port is x4 and the SerDes lanes are statically assigned to a port.

Polarity Inversion

Each port of the PES12N3 supports automatic polarity inversion as required by the PCIe® specification. Polarity inversion is a function of the receiver and not the transmitter. The transmitter never inverts its data. During link training, the receiver examines symbols six through 16 of the TS1 and TS2 ordered sets for inversion of the PExAP[n] and PExAN[n] signals. If an inversion is detected, then logic for the receiving lane automatically inverts received data.

Polarity inversion is a lane and not a link function. Therefore, it is possible for some lanes of link to be inverted and for others to not be inverted.

Link Width Negotiation

The PES12N3 supports the option link variable width negotiation feature outlined in the PCIe specification. During link training, Each of the x4 ports is capable of negotiating to a x4, x2 or x1 link width. The negotiated width of each link may be determined from the Link Width (LW) field in the corresponding port's PCI Express® Link Status (PCIELSTS) register.

The Maximum Link Width (MAXLNKWDTH) field in a port's PCI Express Link Capabilities (PCIELCAP) register contains the maximum link width of the port. This field is of RWL type and may be modified when the REGUNLOCK bit is set in the PA_SWCTL register. Modification of this field allows the maximum link width of the port to be configured. The new link width takes effect the next time link training occurs.

To force a link width to x2 despite a link partner's ability to negotiate to x4, the MAXLNKWDTH field could be configured through Serial EEPROM initialization and link retraining forced. Assuming the link partner has a link width greater than or equal to x2 and the capability to negotiate to a width of x2, the link width will negotiate to x2.

When a link negotiates to a width less than x4, the unused lanes are put in a low power state (i.e. L1 state).

Lane Reversal

The PCIe specification describes an optional lane reversal feature. The PES12N3 does not support the automatic lane reversal feature outlined in the PCIe specification. However, it does support static lane reversal on a per port basis.

Associated with each PES12N3 switch port is a lane reversal signal. The lane reversal signal for port A is PEALREV, for port B is PEBLREV, and port C is PECLREV. The status of the lane reversal signals sampled during a fundamental reset may be determined from the PALREV, PBLREV and PCLREV fields in the PA_SWSTS register.

The port lane reversal signals are sampled during a fundamental reset and used as the initial value of the PALREV, PBLREV and PCLREV fields in the PA_SWCTL register. When these bits are set, then the lanes of the corresponding port(s) are reversed during link training.

Lane reversal mapping for the various non-trivial maximum link width configurations is illustrated in Figures 3.1 and 3.2.

Notes

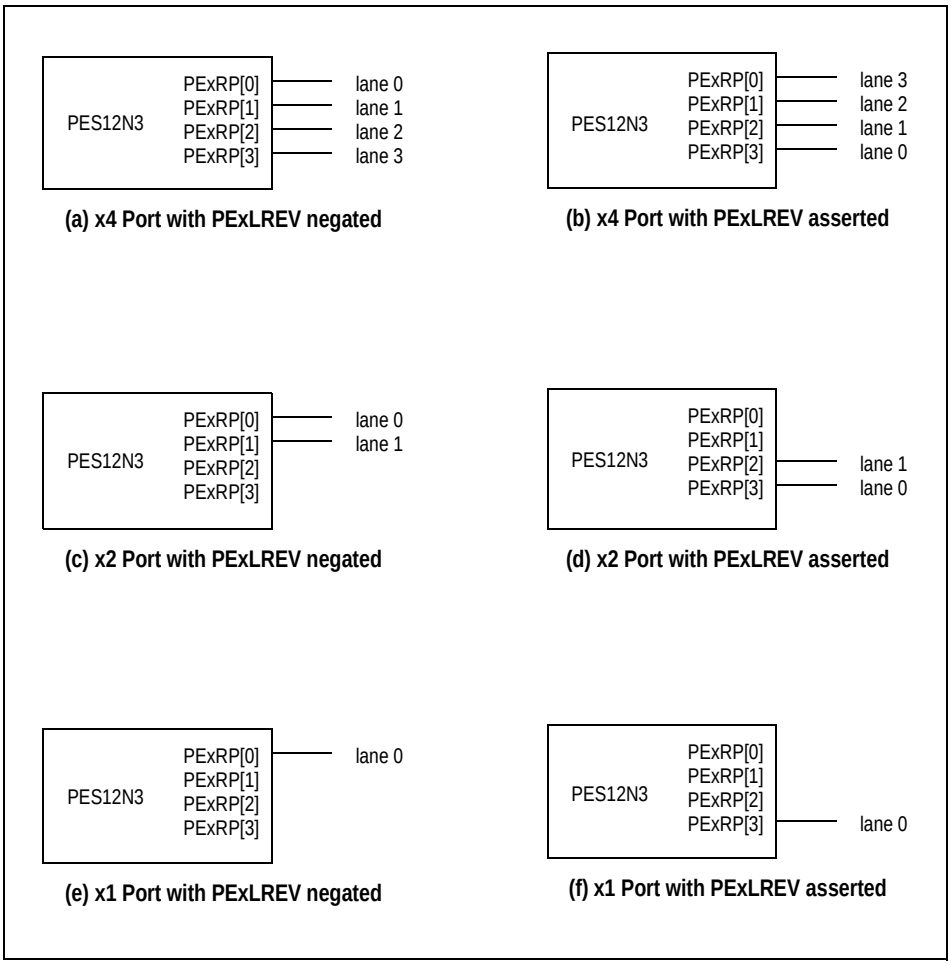

Figure 3.1 Lane Reversal for Maximum Link Width of x4 (MAXLNKWDTH[1:0]=0x2)

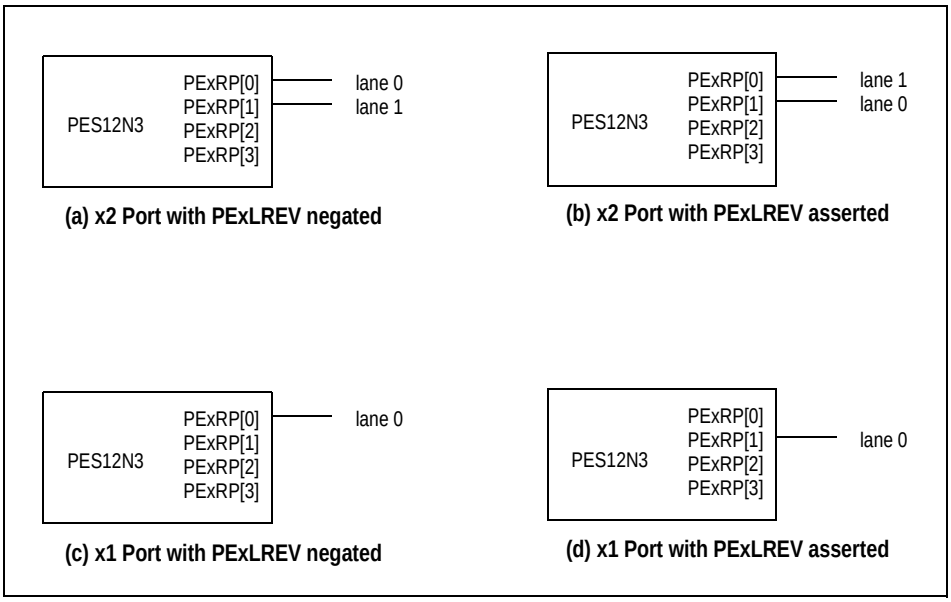

Figure 3.2 Lane Reversal for Maximum Link Width of x2 (MAXLNKWDTH[1:0]=0x1)

Notes

When link training occurs, the corresponding lane reversal bits in the PA_SWCTL register are examined. If a bit is set, then the lanes associated with that link are revered. This mechanism may be used to configure lane reversal via the serial EEPROM, slave SMBus, or root.

Link Retraining

Link retraining should not cause either a downstream component or an upstream component to reset or revert to default values.

Link Down

When a link goes down, all TLPs received by the port and queued in the switch are discarded and all TLPs received by other ports and destined to the port whose link is down are treated as Unsupported Requests (UR). While a link is down, it is possible to perform configuration read and write operations to the PCI-PCI bridge associated with the link. However, it is possible to lose configuration read or write completions when TLPs queued in the switch are discarded.[1] If this occurs, the root's completion timer associated with the transaction(s) will time-out and the transaction will be retired.

When a link comes up, flow control credits for the configured size of the IFB FIFOs are advertised.

Slot Power Limit Support

The Set_Slot_Power_Limit message is used to convey a slot power limit value from a downstream switch port (i.e., ports B or C) to the upstream port of a connected device or switch. A Set_Slot_Power_Limit message is set by downstream switch ports when either of the following events occurs:

- A configuration write is performed to the corresponding PCIESCAP register when the link associated with the downstream port is up.
- A link associated with the downstream port transitions from a non-operational state to an operational (i.e., up) state.

1. In the case of a configuration write that causes link retraining or a secondary bus reset, a completion corresponding to the configuration write is always returned and never lost.

Notes

Switch Operation

Notes

Introduction

The PES12N3 utilizes an input buffered cut-through switch to forward PCIe® TLPs between switch ports. At a high level the switch may be viewed as consisting of three PCIe stacks and a switch core. The PCIe stacks are each responsible for performing the per port Phy, data link and transaction layer functions defined in the PCIe specification. The switch core is responsible for maintaining routing information in route map tables, maintaining per port ingress and egress flow control information, buffering TLPs, and forwarding TLPs between stacks.

An architectural block diagram of the PES12N3 and switch core is provided in Figure 1.2 of Chapter 1. The buffering and data flow of the switch is graphically depicted in Figure 4.1 below.

Note that an ingress stack can transfer a TLP to its own egress stack through the switch core. This path is necessary since all transactions in the PES12N3 are routed through the switch core, even those that could be satisfied locally, due to the fact that the switch core is responsible for maintaining flow control information.

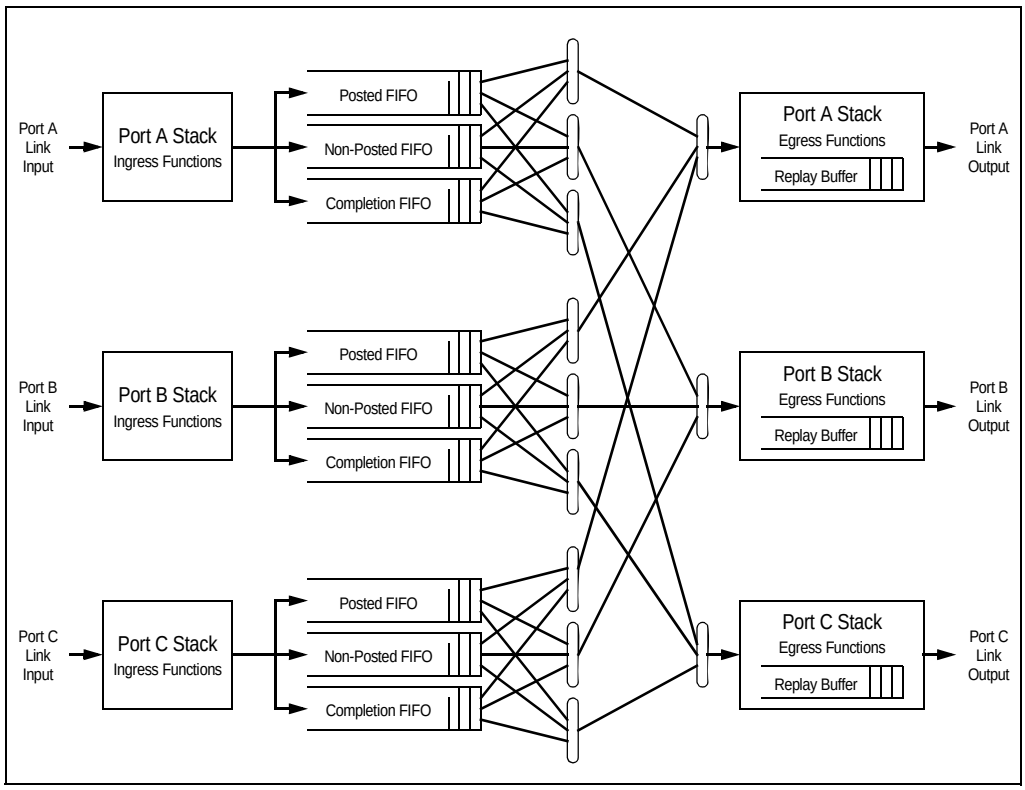

Figure 4.1 PES12N3 Switch Data Flow and Buffering

TLPs are received by a port stack and passed to the switch core. Associated with each port in the switch core are three input buffers. One for posted transactions, one for non-posted transactions and one for completions. The size of each of these buffers is shown in Table 4.1. Associated with each TLP in a buffer is a descriptor. Thus, a buffer has a limitation on the total number of TLPs that can be stored as well as on the number of bytes.

Notes

Buffer	Size and Limitations
Posted FIFO	4 KB and up to 32 TLPs
Non-posted FIFO	1 KB and up to 32 TLPs
Completions FIFO	4 KB and up to 32 TLPs
Egress Stack Replay Buffer[1]	5120 bytes and up to 15 TLPs

Table 4.1 PES12N3 Buffer Sizes

1. Stored with each TLP is a 32-bit LCRC as well as other information.

A flow control mechanism exists between the switch buffers and the transaction layer in the ingress stack to prevent overflows. This flow control mechanism forms the basis of the PCIe flow control credits advertised by the stack to the ingress port's link partner. When a TLP is sent to the switch core from an ingress stack, its header is looked-up in a routing map table and the TLP is queued in a buffer that corresponds to the TLP type (i.e., posted, non-posted or completion).

Scheduling of a TLP to be forwarded from an input buffer to an egress stack is performed by an egress scheduler and port arbiter associated with each egress stack. Thus, the PES12N3 has three egress scheduler and three port arbiters. A flow control mechanism exists between the egress scheduler and the transaction layer in the egress stack. This flow control mechanism ensures that only TLPs which may be accepted by the egress stack's link partner are forwarded through the switch.

TLPs are routed in a cut-through manner through the PES12N3 if the ingress link width is greater than or equal to the egress link width. If the ingress link width is less than the egress link width, then the entire TLP must be received before it is forwarded. The egress scheduler selects the TLP from each ingress port that may be forwarded to the associated egress port. If multiple ingress ports have TLPs which may be forwarded to the same egress port, the port scheduler selects the ingress port from which a TLP is forwarded.

Associated with each TLP in an input buffer is a timestamp. An egress scheduler always selects the TLP in the input buffer that contains the oldest timestamp. If that TLP is destined for a different egress port, then the egress scheduler makes no selection for that input port (i.e., TLPs are always forwarded from an ingress port in chronological order). TLP timestamps are also used to discard any TLP from the head of an input buffer that is more than 50 ms old. See section Switch Time-Outs on page 4-5 for additional details.

In making its selection, the egress scheduler considers the PCIe ordering rules. The PES12N3 supports relaxed ordering for requests as well as completions. When the Disable Relaxed Ordering (DRO) bit is set in the port A Switch Control (SWCTL) register, the switch strongly orders all transactions regardless of the state of the relaxed ordering bit in TLPs.

The port scheduler associated with each egress port in the PES12N3 supports hardwired round robin and weighted round robin with 32 phases. Both of these algorithms only arbitrate TLP requests and do not consider bandwidth consumption.

In addition to the input buffers in the switch core, each egress stack contains a replay FIFO. When the replay buffer fills, backpressure is provided to the switch core and no TLPs are forwarded to that egress port. Table 4.2 enumerates the default flow control credits advertised by each port of the switch core.

Notes

Flow Control Category	Default Advertised Credits	Notes
Posted Header	30 credits	Each credit represents 20 bytes (i.e., 5 doublewords) for a maximum of 600 bytes
Posted Data	204 credits	Each credit represents 16 bytes (i.e., 4 doublewords) for a maximum of 3264 bytes
Non-Posted Header	30 credits	Each credit represents 20 bytes (i.e., 5 doublewords) for a maximum of 600 bytes
Non-Posted Data	30 credits	Each credit represents 16 bytes (i.e., 4 doublewords) for a maximum of 480 bytes (note that non-posted data is assumed to consists of only one doubleword per header)
Completion Header	30 credits	Each credit represents 16 bytes (i.e., 4 doublewords) for a maximum of 480 bytes
Completion Data	204 credits	Each credit represents 16 bytes (i.e., 4 doublewords) for a maximum of 3264 bytes

Table 4.2 PES12N3 Advertised Flow Control Credits

Each header credit is allocated 20 bytes (3 double doublewords) in a FIFO regardless of whether or not a credit represents 16 or 20 bytes. Each posted and completion data credit is allocated 16 bytes (2 double doublewords) in a FIFO. Non-posted data credits are allocated 8 bytes (one double doubleword). Two header and data credits are reserved in each FIFO for TLPs generated by the switch (e.g., configuration read completions).

The header and data flow control credits advertised may be configured on a per port basis via the Flow Control Credit Posted Configuration (FCPTCFG), Flow Control Credit Non-Posted Configuration (FCNPCFG) and Flow Control Credit Completion Configuration (FCCPCFG) registers. These registers may only be modified using the serial EEPROM or during initialization via the SMBus when the RSTHALT bit is set in the PA_SWCTL register. Modifying these registers in a running system produces undefined results.

Routing

The PES12N3 supports routing of all transaction types defined in the PCIe specification. This includes routing using in specification defined transactions as well as those that may be used in vendor defined messages and in future revisions of the PCIe specifications.

Specifically, the PES12N3 supports the following type of routing:

- Address routing with 32-bit or 64-bit format
- ID based routing using bus, device and function numbers.
- Implicit routing utilizing

  Route to root

  Broadcast from root

  Local - terminate at receiver

  Gathered and routed to root
- A summary of TLP types that use the above routing methods is provided in Table 4.3.

Notes

Routing Method	TLP Type Using Routing Method
Route by Address	MRd, MrdLk, MWr, IORd, IOWr, Msg, MsgD
ID Based Routing	CfgRd0, CfgWr0, CfgRd1, CfgWr1, Cpl, CpdD, CplLk, CplDLk, Msg, MsgD
Imlicit Routing - Route to Root	Msg, MsgD
Implicit Routing - Broadcast from Root[1]	Msg, MsgD
Implicit Routing - Local	Msg, MsgD
Implicit Routing - Gathered and Routed to Root	Only supported for PME_TO_Ack messages in response to a root initiated PME_Turn_Off message.

Table 4.3 Switch Routing Methods

1. Broadcast from root messages are only accepted from the root port (i.e., port A). An unsupported request is generated if a TLP with this routing method is received from any other port.

Data Integrity

PCI Express® provides reliable hop-by-hop communication between interconnected devices, such as roots, switches, and endpoints, by utilizing a 32-bit Link CRC (LCRC), sequence numbers, and a link level retransmission protocol. While this mechanism provides reliable communication between interconnected devices, it does not protect against corruption that may occur inside of a device. PCI Express defines an optional end-to-end data integrity mechanism that consists of appending a 32-bit end-to-end CRC (ECRC) computed at the source over the invariant fields of a Transaction Layer Packet (TLP) that is checked at the ultimate destination of the TLP. While this mechanism provides end-to-end error detection, unfortunately it is an optional PCI Express feature and has not been implemented in some North bridges and endpoints. In addition, the ECRC mechanism does not cover variant fields within a TLP.

Since deep sub-micron devices are known to be susceptible to single-event-upsets, a mechanism is desired that detects errors that occur within a PCI express switch. The PES12N3 parity protects all TLPs in the switch, thus enabling corruption that may occur inside of the device to be detected and reported even in systems that do not implement ECRC.[1]

Associated with each port of the PES12N3 is a PCI-PCI bridge. Located in the switch integrity region in extended configuration space of each PCI-PCI bridge are the Switch System Integrity Control (SWSICTL) and Switch System Integrity Parity Error Count (SWSIPECNT) registers. These registers provide control and status over switch errors associated with that switch port and may be read by a root or via the slave SMBus interface.

Data flowing into the PES12N3 is protected by the LCRC. Within the Data Link (DL) layer of the switch ingress port, the LCRC is checked and 32-bit Doubleword (DWord) even parity is computed on the received TLP data. If an LCRC error is detected at this point, the link level retransmission protocol is used to recover from the error by forcing a retransmission by the link partner. As the TLP flows through the switch, its alignment or contents may be modified. In all such cases, parity is updated and not recomputed. Hence, any error that occurs is propagated and not masked by a parity regeneration. When the TLP reaches the DL layer of the switch egress port, parity is checked and in parallel a LCRC is computed. If the TLP is parity error free, then the LCRC and TLP contents are known to be correct and the LCRC is used to protect the packet through the lower portion of the DL layer, PHY layer, and link transmission.

If a parity error is detected by the DL layer of an egress port, then the TLP is nullified by inverting the computed LCRC and ending the packet with an EDB symbol. Nullified TLPs received by the link-partner are discarded. In addition to nullifying the TLP, the PES12N3 performs the following when a parity error is detected: sends an error non-fatal (ERR_NONFATAL) message (if this message reporting is enabled) to the

1. Nullified TLPs are not parity protected and no parity errors are reported for nullified TLPs since these TLPs are discarded.

Notes

root; increments the End-to-End Parity Error Count (EEPERRC) field in the SWSIPECNT register associated with the port on which the error was detected; and sets the Detected Parity Error (DPE) bit in the PCISTS register if the error was detected by a downstream port or sets DPE bit in the PCI Secondary Status (SECSTS) register if the error was detected by an upstream port.

To prevent error flooding, error messages are not sent to the root once the EEPERRC field saturates. Since PCI Express switches do not normally generate ERR_NONFATAL messages, the Silent End-to-End Parity Checking bit (SEEPC) bit in the SWSICTL register is provided to disable generation of error messages and setting of the Detected Parity Error bit when internal corruption is detected.

The default state of the switch following a fundamental reset is to enable this error reporting. (Note that the Device Control register in the PCI Express capability structure also has a bit that enables generation of ERR_NONFATAL messages and that the default value of this bit is to disable these messages.)

In addition to TLPs that flow through the switch, cases exist in which TLPs are produced and consumed by the switch (e.g., a configuration requests and responses). Whenever a TLP is produced by the switch, parity is computed as the TLP is generated. Thus, error protection is provided on produced TLPs as they flow through the switch. In addition, parity is checked on all consumed TLPs. If an error is detected, the TLP is discarded and an error is reported using the mechanism described above.

This means that a parity error reported at a switch port cannot be definitively used to identify the location at which the error occurred as the error may have occurred when parity as generated at another port, in the switch core, or may have been generated locally (i.e., for ingress TLPs to the switch core which are consumed by the port such as Type 0 configuration read requests on the root port).

Switch Time-Outs

The switch discards any TLP that reaches the head of an input buffer and is more than 50ms old.

For non-posted and completion TLPs, the requester's completion time-out mechanism will detect discarded TLPs. No similar mechanism exists in PCIe for posted TLPs. Therefore, whenever a posted TLP is discarded by the switch due to a time-out, an error non-fatal (ERR_NONFATAL) message (if this message reporting is enabled) is sent to the root.

Whenever a TLP is discarded from a posted input buffer, the Posted TLP Time-out Count (PTLPTOC) field is incremented in the Switch System Integrity Time-Out Drop Count (SWSITDCNT) register in the port on which the TLP was received. This is a saturating counter that is automatically cleared when read. Whenever a TLP is discarded from a non-posted input buffer, the Non-Posted TLP Time-out Count (NPTLPTOC) field is incremented in this register and whenever a TLP is discarded from a completion input buffer, the Completion TLP Time-out Count (NPTLPTOC) field is incremented.

To prevent error flooding, error messages are not sent to the root once the PTLPTOC counter saturates. Since PCI Express switches do not normally generate ERR_NONFATAL messages, the Silent Posted TLP Time-out (SPTLPTO) bit in the SWSICTL register is provided to disable generation of error non-fatal messages. When this bit is set, ERR_NONFATAL messages are not generated when posted transactions received on the corresponding port are discarded. The PTLPTOC field however is always updated.

Locking

The PES12N3 supports locked transactions, allowing legacy software to run without modification on PCIe. Only one locked transaction sequence may be in progress at a time. A locked transaction sequence is requested by the root by issuing a Memory Read Request - Locked (MRdLk) transaction. A lock is established when a lock request is successfully completed with a Completion with Data - Locked (CplDLk). A lock is released with an Unlock message (Msg).

Notes

When the PES12N3 receives a MRdLk transaction on its root port destined for a down-stream port, it forwards the MRdLK transaction to the downstream port and locks the downstream port so that all subsequent TLPs destined to the downstream port from ports other than the root are blocked until the lock is released.

- The MRdLK transaction obeys PCI ordering rules meaning that all queued posted requests for the downstream port are completed prior to the MRdLK being transmitted. The MRdLK is allowed by bypass queued non-posted requests and completions.
- When only the downstream port is locked, no transactions destined to any other port are blocked (e.g., transactions from the other downstream ports to the upstream port are not blocked)•

When a CplDLk is returned by the locked downstream port, the upstream port becomes locked causing all transactions destined to the upstream port from sources other than the locked downstream port (e.g., the other downstream port) to be blocked. If the lock is unsuccessful, then a CPlLk is returned by the downstream port and the upstream port does not become locked.

- The CplDLk transaction obeys PCI ordering rules[1] meaning that all queued posted requests at the locked downstream port destined to the upstream port are completed prior to the CplDLk being transmitted. The CplDLk is allowed by bypass queued non-posted requests and completions.
- When a CplDLk is returned by the locked downstream port and the upstream port becomes locked, the entire switch becomes locked. This means that only transactions between the upstream and the locked downstream port are allowed to progress. All other transactions, such as transactions from the other downstream port, hot plug messages, MSI/INT messages, etc. are blocked until the switch is unlocked.
- While the switch is locked, any register in the switch may be read or written via the SMBus
- While the switch is locked, it is illegal to read or write any of the PCIe configuration space headers in the switch since the switch can not generate a completion until the switch is unlocked. This means that the LOCKMODE field in the PA_SWSTS register can only be read via the SMBus when the switch is locked.
- The behavior of the switch is undefined when any transaction other than a MWr, MRdLk, and Unlock message is received on the upstream port when the switch is locked.
- The behavior of the switch is undefined when any transaction other than a CplLk and a CplDLk is received on the locked downstream port when the switch is locked.

Once the switch is locked, it is possible for the root to perform subsequent reads from the locked device by issuing a MRdLk requests to the locked device and receiving a CplDLk or CplLk response from the locked device. These transactions do not change the state of the switch when the switch is locked. Therefore, a CplLk completion once the switch is locked in no way “unlocks” the switch.

Once the switch is locked, it is possible for the root to perform subsequent writes to the locked device by issuing MWr requests to the locked device. These transactions in no way change the state of the switch when the switch is locked.

When an Unlock message is received on the upstream port, the switch is unlocked. This causes the Unlock message to be forwarded to the locked downstream port and the unblocking of transactions destined to the upstream and previously locked downstream port.

When a TLP from a downstream port is blocked from being forwarded due to a locked switch, then the TLP is delayed until the switch is unlocked. If the switch is locked for an extended period, this may cause TLPs to be discarded due to switch time-outs (see section Switch Time-Outs on page 4-5).

When a MRdLk TLP is received on the upstream port, then the TLP is dropped due to a lock violation and the lock drop (LOCKDROP) bit is set in the PA_SWSTS register. If error reporting is enabled, an ERR_NON_FATAL message is sent to the root when the switch is unlocked.

When the upstream port is locked with a downstream port and a TLP is received by slipstream port that is destined to the unlocked downstream port, then the TLP is dropped, the Lock Discarded (LOCKDIS) bit is set in the PA_SWSTS register. If error reporting is enabled, an ERR_NON_FATAL message is sent to the root when the switch is unlocked.

1. The relaxed ordering is ignored.

Notes

The PME Lock Error (PMELOCK) bit in the PA_SWSTS register is set and the transaction is dropped when a PME_Turn_Off message is received by a locked downstream PCI-PC I bridge (i.e., that associated with port B or C). If error reporting is enabled, an ERR_NON_FATAL message is sent to the root when the switch is unlocked.

The locked status of the switch may be determined by examining the Lock Mode (LOCK-MODE) bit in the PA_SWSTS register

Interrupts

The PES12N3 supports legacy PCI INTx emulation where x is A, B, C or D. Rather than use sideband INTx signals, PCIe defines two messages that indicate the assertion and negation of an interrupt signal. An Assert_INTx message is used to signal the assertion of an interrupt signal and an Deassert_INTx message is used to signal its negation.

The PES12N3 maintains an aggregated INTx state for each of the four interrupt signals (i.e., A through D). The value of the INTA, INTB, INTC and INTD aggregated state may be determined by examining the corresponding fields in the PA_SWSTS register. The aggregated INTx state of each port for each of the four interrupt signals (i.e., A through D) on the primary side of its PCI to PCI bridge may be determined by examining the state of the INTA, INTB, INTC and INTD fields in the corresponding port's Interrupt Status (PA_INTSTS, PB_INTSTS, and PC_INTSTS) register.

An Assert_INTx message is sent to the root by the upstream port (i.e., port A), when the aggregated state of the corresponding interrupt in the switch transitions from a negated to an asserted state. A Deassert_INTx message is sent to the root by the upstream port when the aggregated state of the corresponding interrupt transitions from an asserted to a negated state.

Table 4.4 exhibits the interrupt sources that are aggregated by the switch.

PCI Compatible INTx	Interrupt Sources
INTA	- External downstream port B - External downstream port C - Port B PCI-PCI bridge (hot-plug)
INTB	- External downstream port B - External downstream port C - Port C PCI-PCI bridge (hot-plug)
INTC	- External downstream port B - External downstream port C
INTD	- External downstream port B -External downstream port C

Table 4.4 PCI Compatible INTx Aggregation

PCI to PCI bridges must map interrupts on the secondary side of the bridge according to the device number of the device on the secondary side of the bridge. No mapping is performed for the PCI to PCI bridges corresponding to downstream ports as these ports only connect to device zero. A mapping is performed for the upstream port (i.e., port A).This mapping is summarized in Table 4.6 for the PES12N3.

Notes

Port A Interrupt	Interrupt Sources[1]
INTA	Port B INTA Port C INTD
INTB	Port B INTB Port C INTA
INTC	Port B INTC Port C INTB
INTD	Port B INTD Port C INTC

Table 4.5 PES12N3 Upstream Port Bridge Interrupt Mapping

1. Port X INTy corresponds to external downstream generated INTy interrupts and hot-plug INTy interrupts generated by the port.

If a Downstream Port goes to DL_Down status, the INTx virtual wires associated with that port are deasserted, and the port A aggregates are updated accordingly. This may result in the upstream port generating a Deassert_Intx message.

Switch Core Errors

This section lists error conditions that are checked by the switch core. Due to limited buffering of Unsupported Request (UR) completions, it is possible for the PES12N3 to discard UR completions if errors are generated faster than UR completions can be transmitted. Even when UR completions are discarded, error status bits are always correctly updated and an error message is generated.

Due to limited buffering, error messages may be collapsed if errors are generated faster than error messages can be transmitted. This means that multiple error conditions may result in only a single error message being generated. However, under no circumstances are error messages discarded.

Port arbitration should never be configured to starve a port. If a port arbitration table configuration results in port starvation, then TLPs generated by the port may be dropped (e.g., error messages, interrupts, configuration completions, etc.).

The following events received by the switch core from the upstream port are treated as Unsupported Requests (UR), and for non-posted transactions, result in a Unsupported Request (UR) completion being returned to the upstream port.

- Reception of a CfgRd0 or CfgWr0 TLP. All CfgRd0 and CfgWr0 TLPs should have been received and processed by the upstream stack. Therefore, the upstream stack should never pass a CfgRd0 or CfgWr0 to the switch core.
- Reception of a CfgRd1 or CfgWr1 TLP that is transformed into a CfgRd0 or CfgWr0 TLP destined to the link partner of a downstream port and in which the device number is non-zero (covers condition outlined in PCIe base 1.0a Section 7.3.1). The device number must be zero in CfgRd0 and CfgWr0 transactions to a downstream link partner.
- Reception of Msg or MsgD TLPs with route by address routing prior to initialization of the PCI-PCI bridge. Prior to initialization of the PCI-PCI bridge, no transactions should be routed to the switch core.
- Reception of route by address TLPs whose address matches an upstream port's memory or I/O base/limit pair and does not match a downstream ports' memory or I/O base/limit pair. TLPs that have no route (i.e., not destined for any upstream or downstream port) should be treated as unsupported requests.
- Reception of route by address TLPs destined to the upstream port. There are no route by address TLPs that should have been destined to the upstream port since the upstream port does not process these types of TLPs.

Notes

- Reception of TLPs that have no route (i.e., do not match an address or ID route through the switch). TLPs that have no route should be treated as unsupported requests.
- Reception of a TLP destined to a disabled downstream port (link down or MAE/IOAE bit cleared in PA_PCICMD register). TLPs destined to a disabled downstream port should be treated as unsupported requests.
- Reception of a TLP that matches a VGA region and the VGA Enable (VGAEN) bit is set in the upstream port but the TLP does not map to either downstream port (i.e., VGAEN is cleared in both downstream ports and the transaction does not map to any of the base/limit pairs associated with the downstream ports).

The following events received by the switch core from the downstream ports are treated as Unsupported Requests (UR) and for non-posted transactions, result in a Unsupported Request (UR) completion to be returned to the port on which the TLP was received.

- Reception of Msg or MsgD TLPs with route by address routing prior to initialization of the PCI-PCI bridge. Prior to initialization of the PCI-PCI bridge, no transactions should be routed to the switch core.
- Reception of Msg or MsgD TLPs with route by ID to the PCI-PCI bridge primary bus number after bus enumeration has completed. There are no entities that generate accept messages on the virtual PCI bus within the switch (i.e., the primary bus number).
- Reception of route by address TLPs whose address matches an upstream port's memory or I/O base/limit pair and does not match a downstream ports' memory or I/O base/limit pair. TLPs that have no route (i.e., not destined for any upstream or downstream port) should be treated as unsupported requests.
- Reception of TLPs that have no route (i.e., do not match an address or ID route through the switch). TLPs that have no route should be treated as unsupported requests.
- Reception of any configuration TLP. Configuration requests can only be generated by the root and received on the upstream port.
- Reception of a route by ID TLP to a port that has its primary bus number set to its secondary bus number. Such a port is uninitialized.
- Reception of a TLP that utilizes implicit routing - broadcast from root. Such a TLP can only be received by the upstream port.
- Reception of a TLP that matches a VGA region in a downstream port when the downstream port's VGA Enable (VGAEN) bit is set in its Bridge Control (BCTRL) register.
- Reception of a TLP destined to a disabled downstream port (link down or MAE/IOAE bit cleared in PCICMD register) or the upstream port when the Bus Master Enable (BME) bit is not set in the PCICMD register. TLPs destined to a disabled downstream port should be treated as unsupported requests.

Notes

Power Management

Notes

Introduction

A power management capability structure is located in the configuration space of each PCI-PCI bridge in the PES12N3. The structure associated with a PCI-PCI bridge of a downstream port only affects that port. Entering the $D3_{hot}$ state allows the link associated with the bridge to enter the L1 state.

The power management capability structure associated with the root port (i.e., port A) affects the entire device. When the root port enters a low power state and the PME_TO_Ack messages are received, then the entire device is placed into a low power state. The PES12N3 supports the following device power management states: D0 Uninitialized, D0 Active, $D3_{hot}$, and $D3_{cold}$. Transitioning a port's power management state from $D3_{hot}$ to $D0_{uninitialized}$ does not result in any logic being reset or re-initialization of register values.

A power management state transition diagram for the states supported by the PES12N3 is provided in Figure 5.1 and described in Table 5.6.

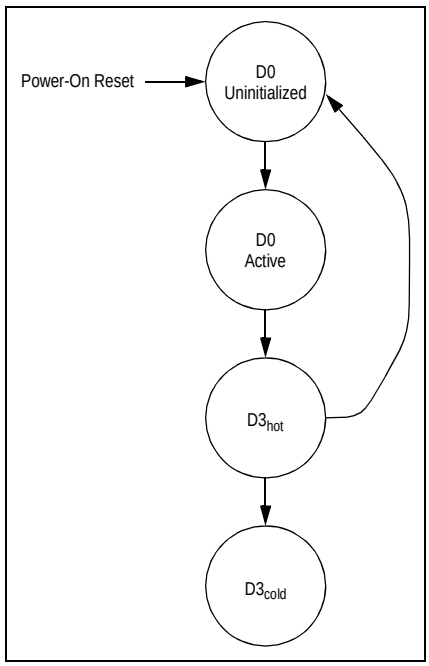

Figure 5.1 PES12N3 Power Management State Transition Diagram

Notes

From State	To State	Description
Any	D0 Uninitialized	Power-on fundamental reset.
D0 Uninitialized	D0 Active	PCI-PCI bridge configured by software
D0 Active	$D3_{hot}$	The Power Management State (PMSTATE) field in the PCI Power Management Control and Status (PMCSR) register is written with the value that corresponds to the $D3_{hot}$ state.
$D3_{hot}$	D0 Uninitialized	The Power Management State (PMSTATE) field in the PCI Power Management Control and Status (PMCSR) register is written with the value that corresponds to D0 state.
$D3_{hot}$	$D3_{cold}$	Power is removed from the device.

Table 5.6 PES12N3 Power Management State Transition Diagram

PME Messages

The PES12N3 does not support generation of PME messages from the $D3_{cold}$ state. Downstream ports (i.e., PCI-PCI bridges associated with ports B and C) support the generation of hot-plug PME events (i.e., a PM_PME power management message) from the $D3_{hot}$ state. This includes both the case when the downstream port is in the $D3_{hot}$ state or the entire switch is in the $D3_{hot}$ state. The generation of a PME message by downstream ports necessitates the implementation of a PME service time-out mechanism to ensure that PME messages are not lost.

Link States

The PES12N3 supports the following link states:

- L0 — Fully operational link state
- L0s — Automatically entered low power state with shortest exit latency
- L1 — Lower power state than L0s. May be automatically entered or directed by software by placing the device in the $D3_{hot}$ state.
- L2/L3 Ready — The L2/L3 state is entered after the acknowledgement of a PM_Turn_Off Message. There is no TLP or DLLP communications over a link in this state.
- L3 — Link is completely unpowered and off.

Link states are shown in Figure 5.2.

Notes

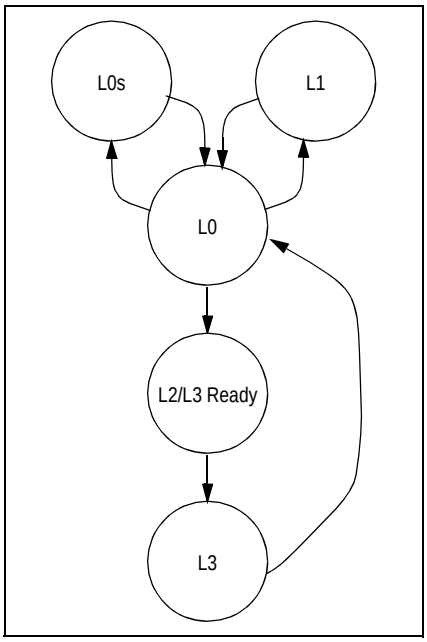

Figure 5.2 PES12N3 ASPM Link Sate Transitions

Active State Power Management

The operation of Active State Power Management (ASPM) is orthogonal to power management. Once enabled by the ASPM field in the PCI Express® Link Control (PCIELCTL) register, ASPM link state transitions are initiated by hardware without software involvement. The PES12N3 ASPM supports the required L0s state as well as the optional L1 state.

The L0s Entry Timer (L0SET) field in the PCI Power Management Proprietary Control (PMPC) register controls the amount of time L0s entry conditions must be met before the hardware transitions the link to the L0s state. The L1 Entry Timer (L1SET) field in the PCI Power Management Proprietary Control (PMPC) register controls the amount of time L1 entry conditions must be met before the hardware transitions the link to the L1 state. If these conditions are met and the link is in the L0 or L0s states, then the hardware will request a transition to the L1 state from its link partner. Note that L1 entry requests are only made by the PES12N3 upstream port. If the link partner acknowledges the transition, then the L1 state is entered. Otherwise the L0s state is entered.

Notes

Hot-Plug and Hot-Swap

Notes

Introduction

As illustrated in Figures 6.1 through 6.3, a PCIe® switch may be used in one of three hot-plug configurations.

Figure 6.1 illustrates the use of the PES12N3 in an application in which the two downstream ports are connected to slots into which add-in cards may be hot-plugged.

Figure 6.2 illustrates the use of the PES12N3 in an add-in card application. Here the two downstream ports are hardwired to devices on the add-in card and the upstream port serves as the add-in card's PCIe interface. In this application the upstream port may be hot-plugged into a slot on the main system.

Finally, Figure 6.3 illustrates the use of the PES12N3 in a carrier card application. In this application, the two downstream ports are connected to slots which may be hot-plugged and the entire assembly may be hot-plugged into a slot on the main system. Since this application requires nothing more than the functionality illustrated in both Figure 6.1 and Figure 6.2, it will not be discussed further.

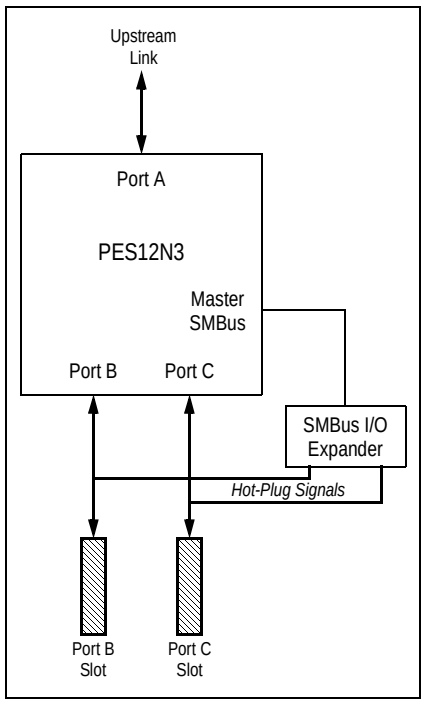

Figure 6.1 Hot-Plug on Switch Downstream Slots Application

Notes

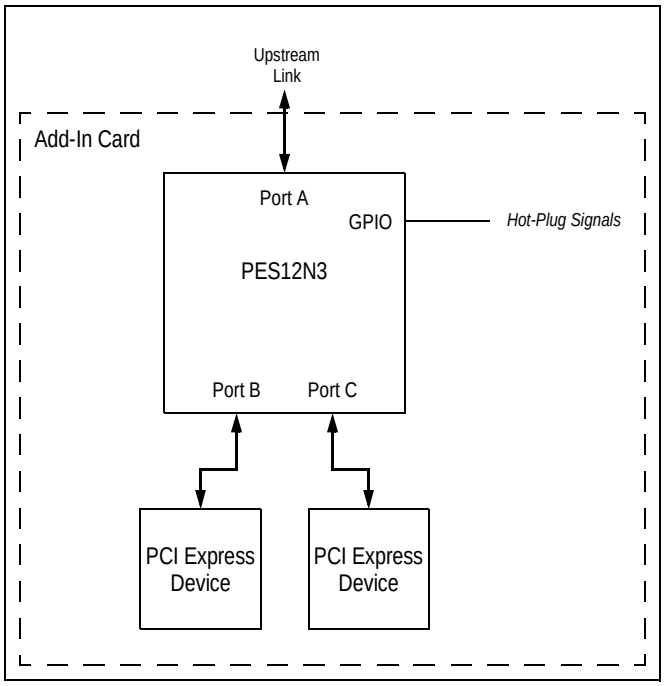

Figure 6.2 Hot-Plug with Switch on Add-In Card Application

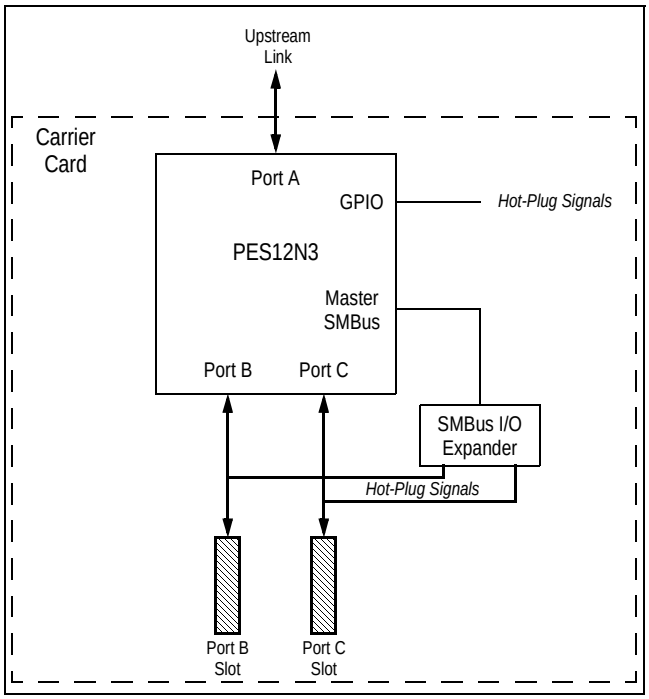

Figure 6.3 Hot-Plug with Carrier Card Application

Notes

The PCI Express® Base Specification revision 1.0a allowed a hot-plug attention indicator, power indicator, and attention button to be located on the board on which the slot is implemented or on the add-in board. When located on the add-in board, state changes are communicated between the hot-plug controller associated with the slot and the add-in card via hot-plug messages. This capability was removed in revision 1.1 of the PCI Express Base Specification.

Therefore, there are differences in the behavior of a PCIe 1.0a hot-plug capable switch and a PCIe1.1 hot-plug capable switch. The Hot Plug Mode (HPMODE) bit in the upstream port's Switch Control (PA_SWCTL) register selects the operating mode of the hot-plug functionality in the PES12N3. Differences in operation are noted in the following sections.

Hot-Plug with Downstream Port(s) Connected to a Slot

This section discusses the use of the PES12N3 in which one or both the of the downstream ports are used in an application in which an add-in card may be hot-plugged into a downstream slot. Associated with each downstream port in the PES12N3 is a hot-plug controller. The hot-plug controller may be enabled by setting the HPC bit in the PCI Express Slot Capabilities (PCIESCAP) register during configuration (e.g., via serial EEPROM).

The PES12N3 allows sensor inputs and indicator outputs to be located next to the slot or on the plug in module. When implemented on the slot, the appropriate bits should be set during configuration in the PCI Express Slot Capabilities (PCIESCAP) register. Table 6.7 lists the hot-plug inputs and outputs that may be associated with a slot. When enabled during configuration in the PCIESCAP register, these inputs and outputs are made available to external logic using an external I/O expander located on the master SMBus interface.

The PES12N3 only supports presence detect signalling via a pin assertion. It does not support in-band presence detect.

Signal	Type	Name/Description
PxAPN	I	**Port x[1] Attention Push button Input.**
PxPDN	I	**Port x Presence Detect Input.**
PxPFN	I	**Port x Power Fault Input.**
PxMRLN	I	**Port x Manually-operated Retention Latch (MRL) Input.**
PxAIN	O	**Port x Attention Indicator Output.**
PxPIN	O	**Port x Power Indicator Output.**
PxPEP	O	**Port x Power Enable Output.**
PxINTERLOCKP	O	**Port x Electromechanical Interlock.**

Table 6.7 Downstream Ports B and C Hot Plug Signals

1. x corresponds to downstream port B or C.

Hot-Plug I/O Expander

The PES12N3 utilizes an external SMBus/I2C-bus I/O expander connected to the master SMBus interface for hot-plug related signals associated with downstream ports. It is not possible to utilize EEPROM commands to toggle the I/O expander outputs due to conflicting usage of the master SMBus. SMBus writes or configuration writes can be used to control the I/O expander outputs.

The PES12N3 is designed to work with Phillips PCA9555 compatible I/O expanders. See the Phillips PCA9555 data sheet for details on the operation of this device.

The external SMBus I/O expander provides 16 bit I/O pins that may be configured as inputs or outputs. The mapping of downstream port B and C hot-plug signals listed in Table 6.7 to these I/O pins is exhibited in Table 6.8.

Notes

The port B and C electromechanical interlock outputs are only used in PCIe 1.1 mode (i.e., HPMODE bit set). These signals are driven to their negated state in PCIe 1.0a mode.

SMBus I/O Expander Bit	Type	Signal
0 (I/O-0.0)[1]	I	PBAPN
1 (I/O-0.1)	I	PBPDN
2 (I/O-0.2)	I	PBPFN
3 (I/O-0.3)	I	PBMRLN
4 (I/O-0.4)	O	PBAIN
5 (I/O-0.5)	O	PBPIN
6 (I/O-0.6)	O	PBPEP
7 (I/O-0.7)	O	PBINTERLOCKP[2]
8 (I/O-1.0)	I	PCAPN
9 (I/O-1.1)	I	PCPDN
10 (I/O-1.2)	I	PCPFN
11 (I/O-1.3)	I	PCMRLN
12 (I/O-1.4)	O	PCAIN
13 (I/O-1.5)	O	PCPIN
14 (I/O-1.6)	O	PCPEP
15 (I/O-1.7)	O	PCINTERLOCKP[2]

Table 6.8 SMBus I/O Expander Signals

1. I/O-x.y corresponds to the notation used for PCA9555 port x I/O pin y.

2. Not used in PCIe 1.0a mode (i.e., HPMODE bit cleared).

During configuration of the PES12N3, the SMBus/I2C-bus address of the hot-plug I/O expander should be written to the Hot-plug I/O Expander Master SMBus Address (IOEADDR) field in the SMBUS status (PA_SMBUSSTS) register.

SMBus write transactions are issued to the I/O expander by the PES12N3 to configure the device whenever the value of the IOEADDR field is modified. Outputs for downstream ports that are disabled are set to their negated value (e.g., the power indicator is turned off).

The I/O expander configuration sequence issued by the PES12N3 is as follows:

- write value 0x50 to I/O expander register 2
- write value 0x50 to I/O expander register 3
- write value 0x0 to I/O expander register 4 (no inversion in IO-0)
- write value 0x0 to I/O expander register 5 (no inversion in IO-1)
- write value 0x0F to I/O expander register 6 (bits 4, 5, 6 and 7 are outputs of IO-0)
- write value 0x0F to I/O expander register 7 (bits 4, 5, 6 and 7 are outputs of IO-1)
- read value of I/O expander register 0 to obtain the current state of the I/O IO-0 inputs.
- read value of I/O expander register 1 to obtain the current state of the I/O IO-I inputs.

Whenever a hot-plug output from port B or C needs to change state, a master SMBus transaction is initiated to update the state of the I/O expander. This write operation causes the I/O expander to change the state of its output(s). Port B output values are written to I/O expander register 2 and Port C values are written to I/O expander register 3.

Notes

The I/O expander has an open drain interrupt output that is asserted when a pin configured as an input changes state from the value previously read. The interrupt output from the SMBus I/O expander should be connected to GPIO[2], and GPIO[2] should be initialized during configuration to operate in alternate function mode as the Hot-plug I/O expander interrupt input. See Chapter 8, General Purpose I/O.

Whenever a input to the I/O expander changes state from the value previously read, the interrupt output of the I/O expander connected to GPIO[2] is asserted. This causes the PES12N3 to issue a master SMBus transaction to read the updated state of the I/O expander inputs.

Regardless of the state of the interrupt output of the I/O expander, the PES12N3 will not issue a master SMBus transaction to read the updated state of the I/O expander inputs more frequently than once every 40 milliseconds. This delay in sampling may be used to eliminate external debouncing circuitry. The 40 millisecond sampling frequency also applies to the Hot Plug Port A Attention Button (PAABN) GPIO alternate function.

Port B input values are read from I/O expander register 0 and Port C values are read from I/O expander register 1. The I/O expander interrupt request output is negated whenever the input values are read or when the input pin changes state back to the value previously read. Any errors detected during I/O expander SMBus read or write transactions is reflected in the status bits of the SMBus Status (PA_SMBUSSTS) register. The I/O Expander Interface (PA_IOEXPINTF) register allows direct testing and debugging of the I/O expander functionality.

The Port B Hot-Plug Signals (PBHPS) and the Port C Hot-Plug Signals (PCHPS) fields in the PA_IOEXPINTF register reflect the current state, as viewed by the PES12N3, of all of the I/O expander inputs and outputs.

Writing a one to the Reload I/O Expander Signals (RELOADIOEX) bit in the PA_IOEXPINTF register causes the PES12N3 to generate SMBus write and read transactions to the I/O expander, causing the value in the PBHPS and PCHPS fields to reflect the state of the I/O expander signals. This feature may be used to aid in debugging hot-plug operation. For example, a user who neglects to configure GPIO[2] as an alternate function may use this feature to determine that master SMBus transactions to the I/O expander function properly and that the issue is with the interrupt logic.

The I/O Expander Test Mode (IOEXTM) bit in the PA_IOEXPTINF register allows an I/O expander test mode to be entered. Normally, hot-plug outputs which are generated by ports B and C first update the PBHPS and PCHPS fields before being written to the I/O expander. When this bit is set, these hot-plug outputs are blocked from updating the fields. Instead, values written directly to the PBHPS and PCHPS fields will be sent to the I/O expander. In this mode, the PES12N3 issues a transaction to update the state of the I/O expander whenever a bit corresponding to a hot-plug controller output in these fields changes state due to a configuration write.

Hot-Plug Messages

The PCI Express Base Specification revision 1.0a allows the attention indicator, power indicator and attention button to be implemented on an add-in card instead of the board on which the slot is located. To support this, the specification defines messages that implement virtual wires between the add-in card and the hot-plug controller associated with the slot. The PCI Express Base Specification revision 1.1 removed this capability.

When the Hot Plug Mode (HPMODE) bit is set in the Switch Control (PS_SWCTL) register, the hot-plug controllers operate in PCIe revision 1.1 mode and no downstream messages are generated by the port B or C hot-plug controllers. In addition, if a hot-plug message is received on these ports in this mode, it is silently discarded.

If the HPMODE bit is cleared (default value), then hot-plug messages are generated and processed when received by the port B and C hot-plug controllers.

A downstream ATTENTION_INDICATOR_ON, ATTENTION_INDICATOR_BLINK, or ATTENTION_INDICATOR_OFF message is sent down on port B or C when the hot-plug controller associated with the port is enabled and the state of the Attention Indicator Control (AIC) field is modified in the PB_PCIESCTL or PC_PCIESCTL register.

Notes

A downstream POWER_INDICATOR_ON, POWER_INDICATOR_BLINK, or POWER_INDICATOR_OFF message is sent down on port B or C when the hot-plug controller associated with the port is enabled and the state of the Power Indicator Control (PIC) field is modified in the PB_PCIESCTL or PC_PCIESCTL register.

An attention button pressed event and the Attention Button Pressed (ABP) bit in the PCI Express Slot Status (PCIESST) register is set when the hot plug controller associated with the port is enabled and an ATTENTION_BUTTON_PRESSED message is received on port B or C. ATTENTION_BUTTON PRESSED messages are always consumed by the PES12N3 and not passed to the upstream port.

Hot-Plug Interrupts and Wake-up

The hot-plug controller associated with a downstream slot may generate an interrupt or wake up event.

Hot-plug interrupts are only generated when the Hot Plug Interrupt Enable (HPIE) bit is set in the corresponding port's PCI Express Slot Control (PCIESCTL) register.

The following bits, when set in the PCI Express Slot Status (PCIESSTS) register, generate an interrupt if not masked by the corresponding bit in the PCI Express Slot Control (PCIESCTL) register or by the HPIE bit: the Attention Button Pressed (ABP), Power Fault Detected (PFD), MRL Sensor Changed (MRLSC), Presence Detected Changed (PDC), and Command Completed (CC).

When an unmasked hot-plug interrupt is generated, the action taken is determined by the MSI Enable (EN) bit in the MSI Capability (MSICAP) register and the Interrupt Disable (INTXD) bit in the PCI Command (PCICMD) register.

When the downstream port or the entire switch is in a $D3_{hot}$ state, then the hot-plug controller generates a wake-up event using a PM_PME message instead of an interrupt if the event interrupt is not masked in the slot control (PCIESCTL) register and hot-plug interrupts are disabled by the HPIE bit. If the event interrupt is not masked and hot-plug interrupts are enabled, then both a PM_PME and an interrupt are generated. If the event interrupt is masked, then neither a PM_PME or interrupt are generated. Note that a command completed (CC bit) interrupt will not generate a wake-up event.

Hot-Plug with Switch on an Add-In Card

This section discusses the use of the PES12N3 in add-in card applications in which the upstream port may be hot-plugged into a slot.

The PCI Express Base Specification revision 1.0a allows the attention indicator, power indicator and attention button to be implemented on an add-in card instead of board containing the slot. To support this, the specification defines messages that implement virtual wires between the add-in card and the downstream port hot-plug controller. The PCI Express Base Specification revision 1.1 removed this capability.

When the Hot Plug Mode (HPMODE) bit is set in the Switch Control (PS_SWCTL) register, the hot-plug controllers operate in PCIe revision 1.1 mode. In this mode, all hot plug messages received on the upstream port are silently discarded and no hot-plug messages are sent on the upstream port.

If the HPMODE bit is cleared (default value), then hot-plug messages are generated and processed on the upstream port.

Hot plug signals associated with the upstream port of the switch are listed in Table 6.9 and are alternate functions of GPIO pins. See Chapter 8, General Purpose I/O.

Signal	Type	Name/Description
PAABN	I	**Port A Attention Button Input.**
PAAIN	O	**Port A Attention Indicator Output.**
PAPIN	O	**Port A Power Indicator Output.**

Table 6.9 Upstream Port A Hot Plug Signals

Notes

Association of a power indicator output with the upstream port of the PES12N3 is enabled by setting the Power Indicator Present (PIP) bit in the PCI Express Device Capabilities (PCIEDCAP) register of the upstream port. When this bit is set and GPIO[5] is configured as an alternate function, then the state of the PAPIN output is modified by POWER_INDICATOR_ON, POWER_INDICATOR_BLINK, and POWER_INDICATOR_OFF messages received on the upstream port. These messages are consumed by the upstream port.

Association of an attention indicator output with the upstream port of the PES12N3 is enabled by setting the Attention Indicator Present (AIP) bit in the PCI Express Device Capabilities (PCIEDCAP) register of the upstream port. When this bit is set and GPIO[4] is configured as an alternate function, then the state of the PAAIN output is modified by ATTENTION_INDICATOR_ON, ATTENTION_INDICATOR_BLINK, and ATTENTION_INDICATOR_OFF messages received on the upstream port. These messages are consumed by the upstream port.

Association of a attention push button input with the upstream port of the PES12N3 is enabled by setting the Attention Button Present (ABP) bit in the PCI Express Device Capabilities (PCIEDCAP) register of the upstream port. When this bit is set and GPIO[3] is configured as an alternate function, the assertion of the PAABN signal results in an ATTENTION_BUTTON_PRESSED message being sent on the upstream port of the switch.

The PES12N3 will not sample the state of the PAABN input more frequently than once every 40 milliseconds. This delay in sampling may be used to eliminate external debouncing circuitry.

Hot-Swap

The PES12N3 is hot-swap capable and meets the following requirements:

- All of the I/Os are tri-stated on reset (i.e., SerDes, GPIO, SMBuses, etc.)
- All I/O cells function predictably from early power. This means that the device is able to tolerate a non-monotonic ramp-up as well as a rapid ramp-up of the DC power.
- All I/O cells are able to tolerate a precharge voltage
- Since no clock is present during physical connection, the device will maintain all outputs in a high-impedance state even when no clock is present.
- The I/O cells meet VI requirements for hot-swap.
- The I/O cells respect the required leakage current limits over the entire input voltage range.

In summary, the PES12N3 meets all of the I/O requirements necessary to build a PICMG compliant hot-swap board or system. The hot-swap I/O buffers of the PES12N3 may also be used to construct proprietary hot-swap systems. For a detailed specification of I/O buffer characteristic, see the 89HPES12N3 Data Sheet on the IDT web site.

Notes

SMBus Interfaces

Notes

Introduction

The PES12N3 contains two SMBus interfaces. The slave SMBus interface provides full access to all software visible registers in the PES12N3, allowing every register in the device to be read or written by an external SMBus master. The slave SMBus may also be used to initialize the serial EEPROM used for initialization. The Master SMBus interface provides connection for an optional external serial EEPROM used for initialization and an optional I/O expander used for hot-plug signals.

Six pins make up each of the two SMBus interfaces. These pins consist of an SMBus clock pin, an SMBus data pin, and 4 SMBus address pins. As shown in Figure 7.1, the master and slave SMBuses may be used in a unified or split configuration.

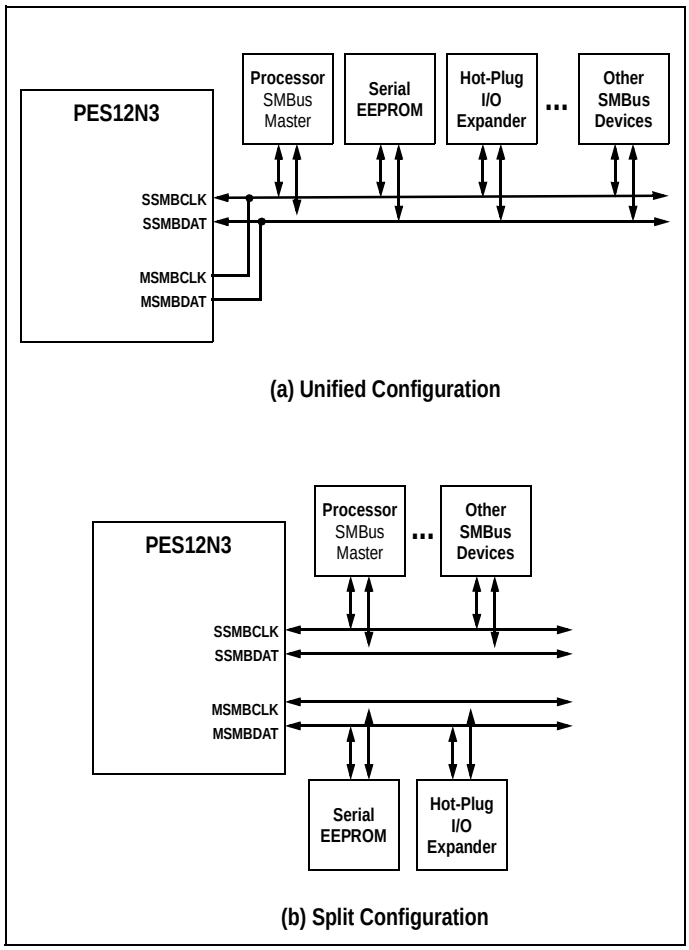

Figure 7.1 SMBus Interface Configuration Examples

In the unified configuration, shown in Figure 7.1(a), the master and slave SMBuses are tied together and the PES12N3 acts both as an SMBus master as well as an SMBus slave on this bus. This requires that the external SMBus master or processor that has access to PES12N3 registers support SMBus arbitration. In

Notes

some systems, this external SMBus master interface may be implemented using general purpose I/O pins on a processor or microcontroller, and thus may not support SMBus arbitration. To support these systems, the PES12N3 may be configured to operate in a split configuration as shown in Figure 7.1(b).

In the split configuration, the master and slave SMBuses operate as two independent buses and thus multi-master arbitration is not required.

SMBus Registers

Bit Field	Field Name	Type	Default Value	Description
0	Reserved	RO	0x0	Reserved field.
7:1	SSMBADDR	RO	HWINIT	**Slave SMBus Address.** This field contains the SMBus address assigned to the slave SMBus interface.
8	Reserved	RO	0x0	Reserved field.
15:9	MSMBADDR	RO	HWINIT	**Master SMBus Address.** This field contains the SMBus address assigned to the master SMBus interface.
16	Reserved	RO	0x0	Reserved field.
23:17	IOEADDR	RWL	0x0	**Hot-plug I/O Expander Master SMBus Address.** This field contains SMBus address assigned to the hot-plug I/O expander on the master SMBus interface.
24	EEPROM-DONE	RO	0x0	**Serial EEPROM Initialization Done.** When the switch is configured to operate in a mode in which serial EEPROM initialization occurs during a fundamental reset, this bit is set when serial EEPROM initialization completes or when an error is detected.
25	NAERR	RW1C	0x0	**No Acknowledge Error.** This bit is set if an unexpected NACK is observed during a master SMBus transaction. The setting of this bit may indicate the following: that the addressed device does not exist on the SMBus (i.e., addressing error); data is unavailable or the device is busy; an invalid command was detected by the slave; or invalid data was detected by the slave.
26	LAERR	RW1C	0x0	**Lost Arbitration Error.** When the master SMBus interface loses arbitration for the SMBus, it automatically re-arbitrates for the SMBus. If the master SMBus interface loses 16 consecutive arbitration attempts, then the transaction is aborted and this bit is set.
27	OTHERERR	RW1C	0x0	**Other Error.** This bit is set if a misplaced START or STOP condition is detected by the master SMBus interface.
28	ICSERR	RW1C	0x0	**Initialization Checksum Error.** This bit is set if an invalid checksum is computed during Serial EEPROM initialization or when a configuration done command is not found in the serial EEPROM.
29	URIA	RW1C	0x0	**Unmapped Register Initialization Attempt.** This bit is set if an attempt is made to initialize via serial EEPROM a register that is not defined in the corresponding PCI configuration space.
31:30	Reserved	RO	0x0	Reserved field.

Table 7.1 SMBUSSTS - SMBus Status

Notes

Bit Field	Field Name	Type	Default Value	Description
15:0	MSMBCP	RW	HWINIT	**Master SMBus Clock Prescalar.** This field contains a clock prescalar value used during master SMBus transactions. The prescalar clock period is equal to 32 ns multiplied by the value in this field. When the field is cleared to zero or one, the clock is stopped. The initial value of this field is 0x0139 when the master SMBus is configured to operate in slow mode (i.e., 100 KHz) in the boot configuration and to 0x0053[1] when it is configured to operate in fast mode (i.e., 400 KHz).
16	MSMBIOM	RW	0x0	**Master SMBus Ignore Other Masters.** When this bit is set, the master SMBus proceeds with transactions regardless of whether it won or lost arbitration.
17	ICHECKSUM	RW	0x0	**Ignore Checksum Errors.** When this bit is set, serial EEPROM initialization checksum errors are ignored (i.e., the checksum always passes).
19:18	SSMBMODE	RW	0x0	**Slave SMBus Mode.** The slave SMBus contains internal glitch counters on the SSMBCLK and SSMBDAT signals that wait approximately 1uS before sampling or driving these signals. This field allows the glitch counter time to be reduced or entirely removed. In some systems, this may permit high speed slave SMBus operation. 0x0 - (normal) Slave SMBus normal mode. Glitch counters operate with 1uS delay. 0x1 - (fast) Slave SMBus interface fast mode. Glitch counters operate with 100nS delay. 0x2 - (disabled) Slave SMBus interface with glitch counters disabled. Glitch counters operate with zero delay which effectively removes them. 0x3 - reserved.
21:20	MSMBMODE	RW	0x0	**Master SMBus Mode.** The master SMBus contains internal glitch counters on the MSMBCLK and MSMBDAT signals that wait approximately 1uS before sampling or driving these signals. This field allows the glitch counter time to be reduced or entirely removed. In some systems, this may permit high speed master SMBus operation. 0x0 - (normal) Master SMBus normal mode. Glitch counters operate with 1uS delay. 0x1 - (fast) Master SMBus interface fast mode. Glitch counters operate with 100nS delay. 0x2 - (disabled) Master SMBus interface with glitch counters disabled. Glitch counters operate with zero delay which effectively removes them. 0x3 - reserved.
31:22	Reserved	RO	0x0	Reserved field.

Table 7.2 SMBUSCTL - SMBus Control

1. The MSMBCLK low minimum pulse width is equal to half the period programmed in this field. The value of 0x53, which corresponds to ~373 KHz, allows the min low pulse width to be satisfied. In systems where this timing parameter is not critical, the operating frequency may be increased.

Notes

Master SMBus Interface

The master SMBus interface is used during a fundamental reset to load configuration values from an optional serial EEPROM. It is also used to support an optional I/O expander for hot-plug signals.

Initialization

Master SMBus initialization occurs during a fundamental reset (see Fundamental Reset on page 2-5).

During a fundamental reset initialization sequence, the state of the Master SMBus Slow Mode (MSMBSMODE) signal is examined. If this signal is asserted, then the Master SMBus Clock Prescalar (MSMBCP) field in the port A SMBus Control (PA_SMBUSCTL) register is initialized to support 100 KHz SMBus operation. If the signal is negated, then the MSMBCP field is initialized for 400 KHz SMBus operation.

Serial EEPROM

During a fundamental reset, an optional serial EEPROM may be used to initialize any software visible register in the device.

Serial EEPROM loading occurs if the Switch Mode (SWMODE[3:0]) field selects an operating mode that performs serial EEPROM initialization (e.g., transparent mode with serial EEPROM initialization).

The address used by the SMBus interface to access the serial EEPROM is specified by the MSMBADDR[4:1] signals as shown in Table 7.3.

Address Bit	Address Bit Value
1	MSMBADDR[1]
2	MSMBADDR[2]
3	MSMBADDR[3]
4	MSMBADDR[4]
5	1
6	0
7	1

Table 7.3 Serial EEPROM SMBus Address

Device Initialization from a Serial EEPROM

During initialization from the optional serial EEPROM, the master SMBus interface reads configuration blocks from the serial EEPROM and updates corresponding registers in the PES12N3.

Any PES12N3 software visible register in the upstream port or downstream port(s) may be initialized with values stored in the serial EEPROM.

Each software visible register in the PES12N3 has a CSR system address which is formed by adding the PCI configuration space offset value of the register to the base address of the configuration space in which the register is located. Configuration blocks stored in the serial EEPROM use this CSR system address shifted right two bits (i.e., configuration blocks in the serial EEPROM use doubleword CSR system addresses and not byte CSR system addresses). Base addresses for the PCI configuration spaces in the PES12N3 are listed in Table 7.4.

Notes

PCI Configuration Space	Base Address Value used to form CSR System Address
Upstream Port A	0x0000
Downstream Port B	0x1000
Downstream Port C	0x2000

Table 7.4 Base Addresses for PCI Configuration Spaces in the PES12N3

Since configuration blocks are used to store only the value of those registers that are initialized, a serial EEPROM much smaller than the total size of all of the configuration spaces may be used to initialize the device.

Any serial EEPROM compatible with those listed in Table 7.5 may be used to store PES12N3 initialization values. Some of these devices are larger than the total size of all of the PCI configuration spaces in the PES12N3 that may be initialized and thus may not be fully utilized.

Serial EEPROM	Size
24C32	4 KB
24C64	8 KB
24C128	16 KB
24C256	32 KB
24C512	64 KB

Table 7.5 PES12N3 Compatible Serial EEPROMs

During serial EEPROM initialization, the master SMBus interface begins reading bytes starting at serial EEPROM address zero. These bytes are interpreted as configuration blocks and sequential reading of the serial EEPROM continues until the end of a configuration done block is reached or the serial EEPROM address rolls over from 0xFFFF to 0x0.

All register initialization performed by the serial EEPROM is performed in double word quantities. There are three configuration block types that may be stored in the serial EEPROM. The first type is a single double word initialization sequence. A double word initialization sequence occupies six byes in the serial EEPROM and is used to initialize a single double word quantity in the PES12N3.

A single double word initialization sequence consists of three fields and its format is shown in Figure 7.2. The CSR_SYSADDR field contains the double word CSR system address of the double word to be initialized. The actual CSR system address, which is a byte address, equals this value with two lower zero bits appended. The next field is the TYPE field that indicates the type of the configuration block. For single double word initialization sequence, this value is always 0x0. The final DATA field contains the double word initialization value.

Notes

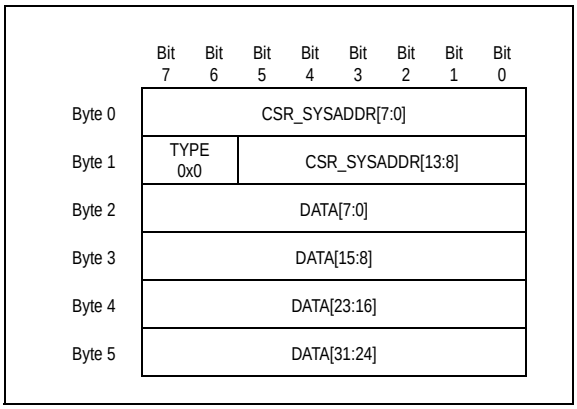

Figure 7.2 Single Double Word Initialization Sequence Format

The second type of configuration block is the sequential double word initialization sequence. It is similar to a single double word initialization sequence except that it contains a double word count that allows multiple sequential double words to be initialized in one configuration block.

A sequential double word initialization sequence consists of four required fields and one to 65535 double word initialization data fields. The format of a sequential double word initialization sequence is shown in Figure 7.3. The CSR_SYSADDR field contains the starting double word CSR system address to be initialized. The next field is the TYPE field that indicates the type of the configuration block. For sequential double word initialization sequences, this value is always 0x1. The NUMDW field specifies the number of double words initialized by the configuration block. This is followed by the number of DATA fields specified in the NUMDW field.

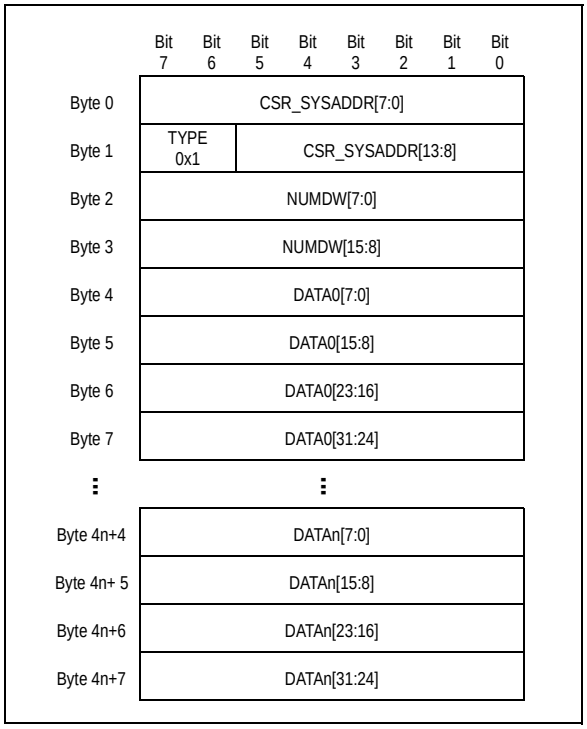

Figure 7.3 Sequential Double Word Initialization Sequence Format

The final type of configuration block is the configuration done sequence which is used to signify the end of a serial EEPROM initialization sequence.

Notes

If during serial EEPROM initialization, an attempt is made to initialize a register that is not defined in a configuration space (i.e., does not appear in Tables 9.6, 9.7, or 9.8), then the Unmapped Register Initialization Attempt (URIA) bit is set in the SMBUSSTS register and the write is ignored.

The configuration done sequence consists of two fields and its format is shown in Figure 7.3. The CHECKSUM field contains the checksum of all of the bytes in all of the fields read from the serial EEPROM from the first configuration block to the end of this done sequence. The second field is the TYPE field which is always 0x3 for configuration done sequences.

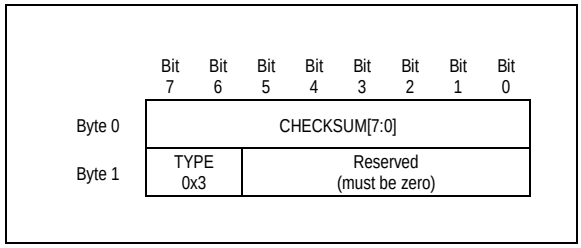

Figure 7.4 Configuration Done Sequence Format

The checksum in the configuration done sequence enables the integrity of the serial EEPROM initialization to be verified. Since uninitialized EEPROMs typically have a value of all ones, initialization from an uninitialized serial EEPROM will result in a checksum mismatch.

The checksum is computed in the following manner.An 8-bit counter is initialized to zero and the 8-bit sum is computed over the configuration bytes stored in the serial EEPROM, including the entire contents of the configuration done sequence, with the checksum field initialized to zero.[1] The 1's complement of this sum is placed in the checksum field.

The checksum is verified in the following manner. An 8-bit counter is cleared and the 8-bit sum is computed over the bytes read from the serial EEPROM, including the entire contents of the configuration done sequence.[2] The correct result should always be 0xFF (i.e., all ones). Checksum checking may be disabled by setting the Ignore Checksum Errors (ICHECKSUM) bit in the port A SMBus Control (PA_SMBUSCTL) register.

If an error is detected during loading of the serial EEPROM, then loading of the serial EEPROM is aborted and the RSTHALT bit is set in the PA_SWCTL register. This allows debugging of the error condition via the slave SMBus interface but prevents normal system operation with a potentially incorrectly initialized device. Error information is recorded in the PA_SMBUSSTS register. Once serial EEPROM initialization completes, or when an error is detected, the EEPROM Done (EEPROMDONE) bit is set in the port A SMBus Status (PA_SMBSTS) register.

A summary of possible errors during serial EEPROM initialization and specific action taken when detected is summarized in Table 7.6.

1. This includes the byte containing the TYPE field.

2. This includes the checksum byte as well as the byte that contains the type and reserved field.

Notes

Error	Action Taken
Configuration Done Sequence checksum mismatch with that computed by the PES12N3	- Set RSTHALT bit in PA_SWCTL register - ICSERR bit is set in the PA_SMBUSSTS register - Abort initialization, set DONE bit in the PA_SMBUSSTS register
Serial EERPOM address roll-over from 0xFFFF to 0x0000	- Set RSTHALT bit in PA_SWCTL register - ICSERR bit is set in the PA_SMBUSSTS register - Abort initialization, set DONE bit in the PA_SMBUSSTS register
Invalid configuration block type (only invalid type is 0x2)	- Set RSTHALT bit in PA_SWCTL register - ICSERR bit is set in the PA_SMBUSSTS register - Abort initialization, set DONE bit in the PA_SMBUSSTS register
An unexpected NACK is observed during a master SMBus transaction	- Set RSTHALT bit in PA_SWCTL register - NAERR bit is set in the PA_SMBUSSTS register - Abort initialization, set DONE bit in the PA_SMBUSSTS register
Master SMBus interface loses 16 consecutive arbitration attempts	- Set RSTHALT bit in PA_SWCTL register - LAERR bit is set in the PA_SMBUSSTS register - Abort initialization, set DONE bit in the PA_SMBUSSTS register
A misplaced START or STOP condition is detected by the master SMBus interface	- Set RSTHALT bit in PA_SWCTL register - OTHERERR bit is set in the PA_SMBUSSTS register - Abort initialization, set DONE bit in the PA_SMBUSSTS register

Table 7.6 Serial EEPROM Initialization Errors

Programming the Serial EEPROM

The serial EEPROM may be programmed prior to board assembly or in-system via the slave SMBus interface or a PCIe® root. Programming the serial EEPROM via the slave SMBus is described in section Serial EEPROM Read or Write Operation on page 7-12. A PCIe root may read and write the serial EEPROM by performing configuration read and write transactions to the port A Serial EEPROM Interface (PA_EEPROMINTF) register.

To read a byte from the serial EEPROM, the root should configure the Address (ADDR) field in the PA_EEPROMINTF register with the byte address of the serial EEPROM location to be read and the Operation (OP) field to “read.” The Busy (BUSY) bit should then be checked. If the EEPROM is not busy, then the read operation may be initiated by performing a write to the Data (DATA) field. When the serial EEPROM read operation completes, the Done (DONE) bit in the PA_EEPROMINTF register is set and the busy bit is cleared. When this occurs, the DATA field contains the byte data of the value read from the serial EEPROM.

To write a byte to the serial EEPROM, the root should configure the ADDR field with the byte address of the serial EEPROM location to be written and set the OP field to “write.” If the serial EEPROM is not busy (i.e., the BUSY bit is cleared), then the write operation may be initiated by writing the value to be written to the DATA field. When the write operation completes, the DONE bit is set and the busy bit is cleared.

Initiating a serial EEPROM read or write operation when the BUSY bit is set produces undefined results. SMBus errors may occur when accessing the serial EEPROM. If an error occurs, then it is reported in the port A SMBus Status (PA_SMBUSSTS) register. Software should check for errors before and after each serial EEPROM access.

Hot-Plug I/O Expander

The PES12N3 utilizes an external SMBus/I2C-bus I/O expander connected to the master SMBus interface for hot-plug related signals associated with downstream ports. See section Hot-Plug I/O Expander on page 6-3 for information on the operation of the hot-plug I/O expander.

Notes

Slave SMBus Interface

The slave SMBus interface provides the PES12N3 with a configuration, management and debug interface. Using the slave SMBus interface, an external master can read or write any software visible register in the device.

Initialization

Slave SMBus initialization occurs during a fundamental reset (see Fundamental Reset on page 2-5). During the fundamental reset initialization sequence, the address is specified by the SSMBADDR[5,3:1] signals as shown in Table 7.7.

Address Bit	Address Bit Value
1	SSMBADDR[1]
2	SSMBADDR[2]
3	SSMBADDR[3]
4	0
5	SSMBADDR[5]
6	1
7	1

Table 7.7 Slave SMBus Address When a Static Address is Selected.

SMBus Transactions

The slave SMBus interface responds to the following SMBus transactions initiated by an SMBus master. See the SMBus 2.0 specification for a detailed description of these transactions.

- Byte and Word Write/Read
- Block Write/Read

Initiation of any SMBus transaction other than those listed above to the slave SMBus interface produces undefined results.

Associated with each of the above transactions is a command code. The command code format for operations supported by the slave SMBus interface is shown in Figure 7.5 and described in Table 7.8.

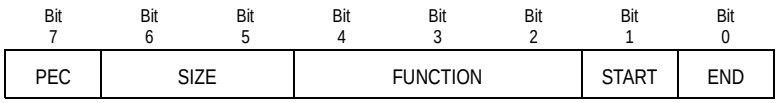

Figure 7.5 Slave SMBus Command Code Format

Notes

Bit Field	Name	Description
0	END	End of transaction indicator. Setting both START and END signifies a single transaction sequence 0 - Current transaction is not the last read or write sequence. 1 - Current transaction is the last read or write sequence.
1	START	Start of transaction indicator. Setting both START and END signifies a single transaction sequence 0 - Current transaction is not the first of a read or write sequence. 1 - Current transaction is the first of a read or write sequence.
4:2	FUNCTION	This field encodes the type of SMBus operation. 0 - CSR register read or write operation 1 - Serial EEPROM read or write operation 2 through 7 - Reserved
6:5	SIZE	This field encodes the data size of the SMBus transaction. 0 - Byte 1 - Word 2 - Block 3 - Reserved
7	PEC	This bit controls whether packet error checking is enabled for the current SMBus transaction. 0 - Packet error checking disabled for the current SMBus transaction. 1 - Packet error checking enabled for the current SMBus transaction.

Table 7.8 Slave SMBus Command Code Fields

The FUNCTION field in the command code indicates if the SMBus operation is a CSR register read/write or a serial EEPROM read/write operation. Since the format of these transactions is different. They will be described individually in the following sections.

If a command is issued while one is already in progress or if the slave is unable to supply data associated with a command, then the command is NACKed. This indicates to the master that the transaction should be retried.

CSR Register Read or Write Operation

Table 7.9 indicates the sequence of data as it is presented on the slave SMBus following the byte address of the Slave SMBus interface.

Notes

Byte Position	Field Name	Description
0	CCODE	**Command Code.** Slave Command Code field described in Table 7.8.
1	BYTCNT	**Byte Count.** The byte count field is only transmitted for block type SMBus transactions. SMBus word and byte accesses do not contain this field. The byte count field indicates the number of bytes following the byte count field when performing a write or setting up for a read. The byte count field is also used when returning data to indicate the number of following bytes (including status). *Note that the byte count field does not include the PEC byte if PEC is enabled.*
2	CMD	**Command.** This field encodes fields related to the CSR register read or write operation.
3	ADDRL	**Address Low.** Lower 8-bits of the doubleword CSR system address of register to access.
4	ADDRU	**Address Upper.** Upper 6-bits of the doubleword CSR system address of register to access. Bits 6 and 7 in the byte must be zero and are ignored by the hardware.
5	DATALL	**Data Low Low.** Bits [7:0] of data doubleword.
6	DATALM	**Data Low Middle.** Bits [15:8] of data doubleword.
7	DATAUM	**Data Upper Middle.** Bits [23:16] of data doubleword.
8	DATAUU	**Data Upper Upper.** Bits [31:24] of data doubleword.

Table 7.9 CSR Register Read or Write Operation Byte Sequence

The format of the CMD field is shown in Figure 7.6 and described in Table 7.10.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WERR	RERR	0	OP	BEUU	BEUM	BELM	BELL

Figure 7.6 CSR Register Read or Write CMD Field Format

Bit Field	Name	Type	Description
0	BELL	Read/Write	**Byte Enable Lower Lower**. When set, the byte enable for bits [7:0] of the data word is enabled.
1	BELM	Read/Write	**Byte Enable Lower Middle**. When set, the byte enable for bits [15:8] of the data word is enabled.
2	BEUM	Read/Write	**Byte Enable Upper Middle**. When set, the byte enable for bits [23:16] of the data word is enabled.
3	BEUU	Read/Write	**Byte Enable Upper Upper**. When set, the byte enable for bits [31:24] of the data word is enabled.

Table 7.10 CSR Register Read or Write CMD Field Description (Part 1 of 2)

Notes

Bit Field	Name	Type	Description
4	OP	Read/Write	CSR Operation. This field encodes the CSR operation to be performed. 0 - CSR write 1 - CSR read
5	0	0	Reserved. Must be zero
6	RERR	Read-Only and Clear	Read Error. This bit is set if the last CSR read SMBus transaction was not claimed by a device. Success indicates that the transaction was claimed and not that the operation completed without error.
7	WERR	Read-Only and Clear	Write Error. This bit is set if the last CSR write SMBus transaction was not claimed by a device. Success indicates that the transaction was claimed and not that the operation completed without error.

Table 7.10 CSR Register Read or Write CMD Field Description (Part 2 of 2)

Serial EEPROM Read or Write Operation

Table 7.11 indicates the sequence of data as it is presented on the slave SMBus following the byte address of the Slave SMBus interface.

Byte Position	Field Name	Description
0	CCODE	**Command Code.** Slave Command Code field described in Table 7.8.
1	BYTCNT	**Byte Count.** The byte count field is only transmitted for block type SMBus transactions. SMBus word and byte accesses do not contain this field. The byte count field indicates the number of bytes following the byte count field when performing a write or setting up for a read. The byte count field is also used when returning data to indicate the number of following bytes (including status).
2	CMD	**Command.** This field contains information related to the serial EEPROM transaction
3	EEADDR	**Serial EEPROM Address.** This field specifies the address of the Serial EEPROM on the Master SMBus when the USA bit is set in the CMD field. Bit zero must be zero and thus the 7-bit address must be left-justified.
4	ADDRL	**Address Low.** Lower 8-bits of the Serial EEPROM byte to access.
5	ADDRU	**Address Upper.** Upper 8-bits of the Serial EEPROM byte to access.
6	DATA	**Data.** Serial EEPROM value read or to be written.

Table 7.11 Serial EEPROM Read or Write Operation Byte Sequence

The format of the CMD field is shown in Figure 7.7 and described in Table 7.12

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0		OTHERERR	LAERR	NAERR	0	USA	OP

Figure 7.7 Serial EEPROM Read or Write CMD Field Format

Notes

Bit Field	Name	Type[1]	Description
0	OP	RW	Serial EEPROM Operation. This field encodes the serial EEPROM operation to be performed. 0 - Serial EEPROM write 1 - Serial EEPROM read
1	USA	RW	**Use Specified Address**. When this bit is set the serial EEPROM SMBus address specified in the EEADDR is used instead of that specified in the MSMBADDR field in the SMBUSSTS register.
2	Reserved		
3	NAERR	RC	No Acknowledge Error. This bit is set if an unexpected NACK is observed during a master SMBus transaction when accessing the serial EEPROM. This bit has the same function as the NAERR bit in the PA_SMBUSSTS register. The setting of this bit may indicate the following: that the addressed device does not exist on the SMBus (i.e., addressing error), data is unavailable or the device is busy, an invalid command was detected by the slave, invalid data was detected by the slave.
4	LAERR	RC	Lost Arbitration Error. This bit is set if the master SMBus interface loses 16 consecutive arbitration attempts when accessing the serial EEPROM. This bit has the same function as the LAERR bit in the PA_SMBUSSTS register.
5	OTHERERR	RC	Other Error. This bit is set if a misplaced START or STOP condition is detected by the master SMBus interface when accessing the serial EEPROM. This bit has the same function as the OTHERERR bit in the PA_SMBUSSTS register.
7:6	Reserved	0	Reserved. Must be zero

Table 7.12 Serial EEPROM Read or Write CMD Field Description

1. See Table Table 1.2 in Chapter 1 for a definition of these abbreviations.

Sample Slave SMBus Operation

This section illustrates sample Slave SMBus operations. Shaded items are driven by the PES12N3's slave SMBus interface and non-shaded items are driven by an SMBus host.

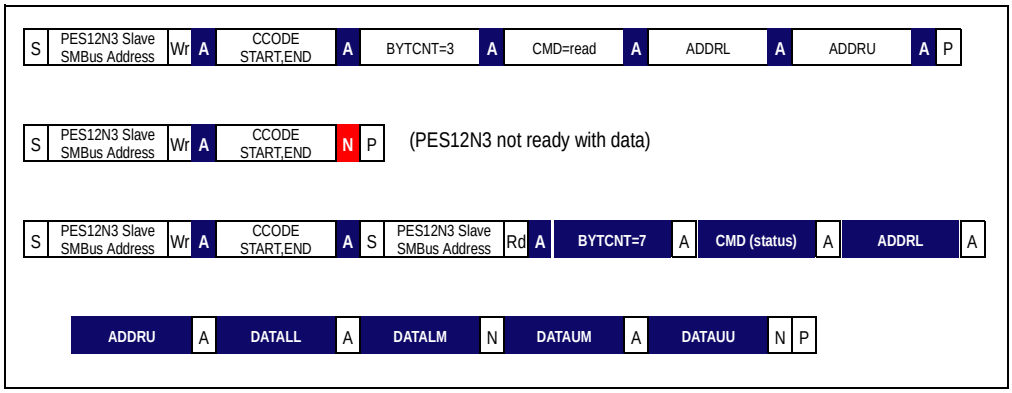

Figure 7.8 CSR Register Read Using SMBus Block Write/Read Transactions with PEC Disabled

Notes

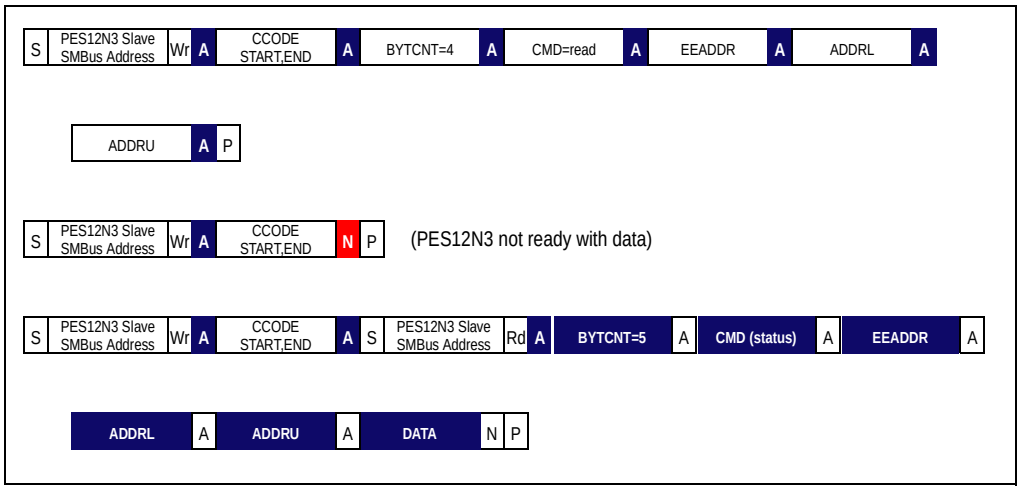

Figure 7.9 Serial EEPROM Read Using SMBus Block Write/Read Transactions with PEC Disabled

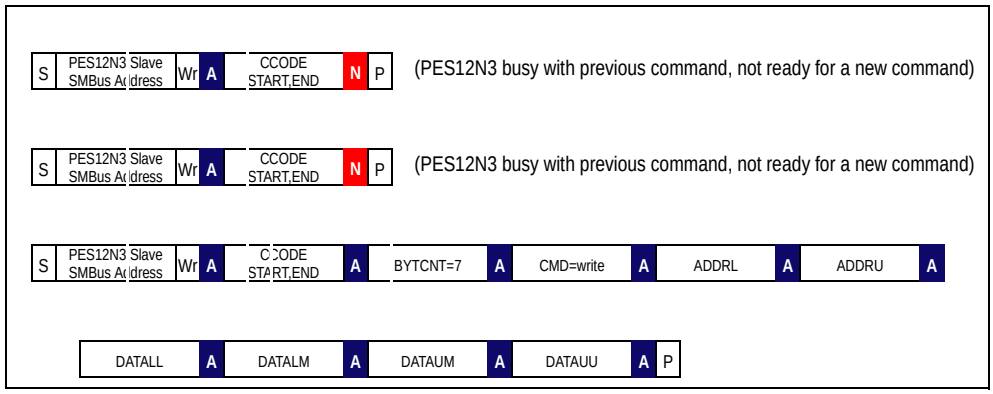

Figure 7.10 CSR Register Write Using SMBus Block Write Transactions with PEC Disabled

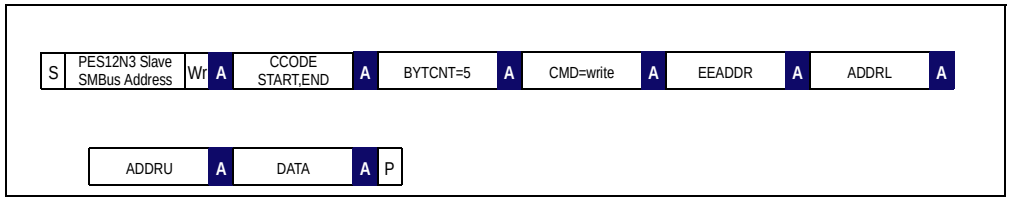

Figure 7.11 Serial EEPROM Write Using SMBus Block Write Transactions with PEC Disabled

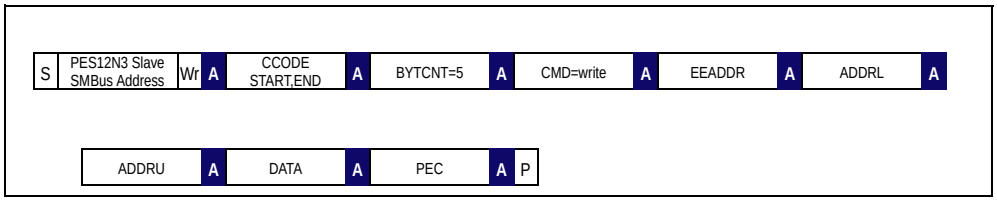

Figure 7.12 Serial EEPROM Write Using SMBus Block Write Transactions with PEC Enabled

Notes

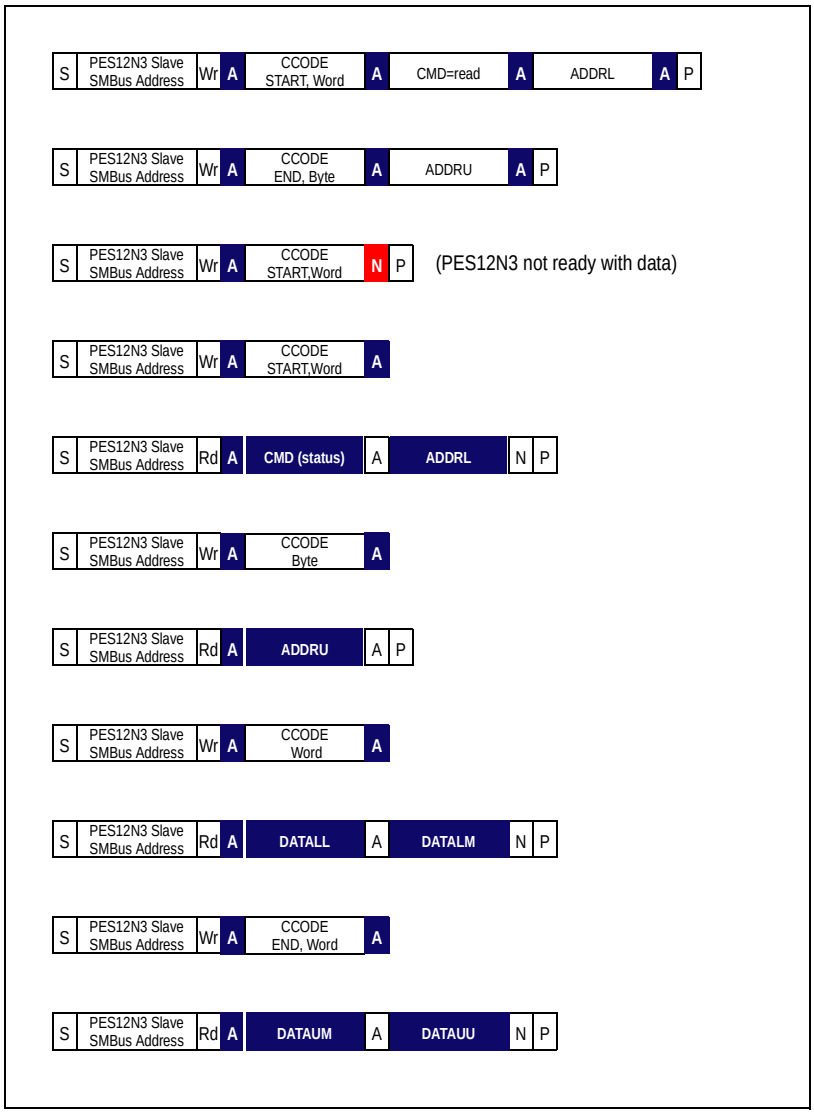

Figure 7.13 CSR Register Read Using SMBus Read and Write Transactions with PEC Disabled

Notes

General Purpose I/O

Notes

Introduction

The PES12N3 has eight General Purpose I/O (GPIO) pins that may be individually configured as: general purpose inputs, general purpose outputs, or alternate functions. GPIO pins are controlled by the General Purpose I/O Control and Status (GPIOCS) register located in upstream port A's PCI configuration space (see Table 8.1).

GPIO Registers

Bit Field	Field Name	Type	Default Value	Description
7:0	GPIOFUNC	RW	0x0	**GPIO Function.** Each bit in this field controls the corresponding GPIO pin. When set to a one, the corresponding GPIO pin operates as the alternate function as defined in Table 8.2. When a bit is cleared to a zero, the corresponding GPIO pin operates as a general purpose I/O pin.
15:8	GPIOCFG	RW	0x0	**GPIO Configuration.** Each bit in this field controls the corresponding GPIO pin. When a bit is configured as a general purpose I/O pin and the corresponding bit in this field is set, then the pin is configured as a GPIO output. When a bit is configured as a general purpose I/O pin and the corresponding bit in this field is zero, then the pin is configured as an input. When the pin is configured as an alternate function, the behavior of the pin is defined by the alternate function.
23:16	GPIOD	RW	HWINIT	**GPIO Data.** Each bit in this field controls the corresponding GPIO pin. Reading this field returns the current value of each GPIO pin regardless of GPIO pin mode (i.e., alternate function or GPIO pin). Writing a value to this field causes the corresponding pins which are configured as GPIO outputs to change state to the value written.
31:24	Reserved	RO	0x0	Reserved field.

Table 8.1 General Purpose IO Registers

As shown in Table 8.2, GPIO pins [5:0] are shared with other on-chip functions. The GPIO Function (GPIOFUNC) field in the GPIOCS register controls whether a GPIO bit operates as a general purpose I/O or as the specified alternate function.

Notes

GPIO Pin	Alternate Function Pin Name	Alternate Function Description	Alternate Function Pin Type
2	IOEXPINTN	Hot-plug I/O expander interrupt	Input
3	PAABN	Hot-plug port A attention button	Input
4	PAAIN	Hot-plug port A attention indicator output	Output
5	PAPIN	Hot-plug port A power indicator output	Output

Table 8.2 General Purpose I/O Pin Alternate Function

After reset, all GPIO pins default to the GPIO input function. GPIO pins configured as GPIO inputs are sampled no more frequently than once every 128 ns and may be treated as asynchronous inputs.

When a GPIO pin is configured to use the GPIO function, the unneeded alternate function associated with the pin is held in an inactive state by internal logic. Care should be exercised when configuring the GPIO pins as outputs since an incorrect configuration could cause damage to external components as well as the PES12N3.

GPIO Configuration

Each bit in the GPIOFUNC, GPIOCFG and GPIOD fields in the GPIOCS register is associated with the corresponding GPIO pin. Table 8.3 summarizes the configuration of GPIO pins.

GPIOFUNC	GPIOCFG	Pin Function
0	0	GPIO input
0	1	GPIO output
1	don't care	Alternate function

Table 8.3 GPIO Pin Configuration

GPIO Pin Configured as an Input

When configured as an input in the GPIOCFG field and as a GPIO function in the GPIOFUNC field, the GPIO pin is sampled and registered in the GPIOD field. The value of the input pin can be determined at any time by reading the GPIOD field. Note that the value in this field corresponds to the value of the pin irrespective of whether the pin is configured as a GPIO input, GPIO output or alternate function.

GPIO Pin Configured as an Output

When configured as an output in the GPIOCFG field and as a GPIO function in the GPIOFUNC field, the value in the corresponding bit position of the GPIOD field is driven on the pin. System designers should treat the GPIO outputs as asynchronous outputs. The actual value of the output pin can be determined by reading the GPIOD field.

GPIO Pin Configured as an Alternate Function

When configured as an alternate function in the GPIOFUNC field, the pin behaves as an described by the section associated with that function. The value of the alternate function pin can be determined at any time by reading the GPIOD field.

Transparent Mode Operation

Notes

Introduction

When the PES12N3 is configured during a fundamental reset to operate in transparent mode or transparent mode with serial EEPROM initialization, the device functionally operates as illustrated in Figure 9.1. In this mode, the PES12N3 may logically viewed as consisting of three PCI-PCI transparent bridges, one per port, and an internal virtual PCI bus.

Associated with each port is a 4 KB configuration space and a Type 1 configuration header. The organization of these configuration spaces is described in section Port Configuration Space Organization on page 9-6. The functional operation of the PES12N3 in transparent mode is fully consistent with that described in the PCI Express® Base Specification, revision 1.0a for a three port switch.

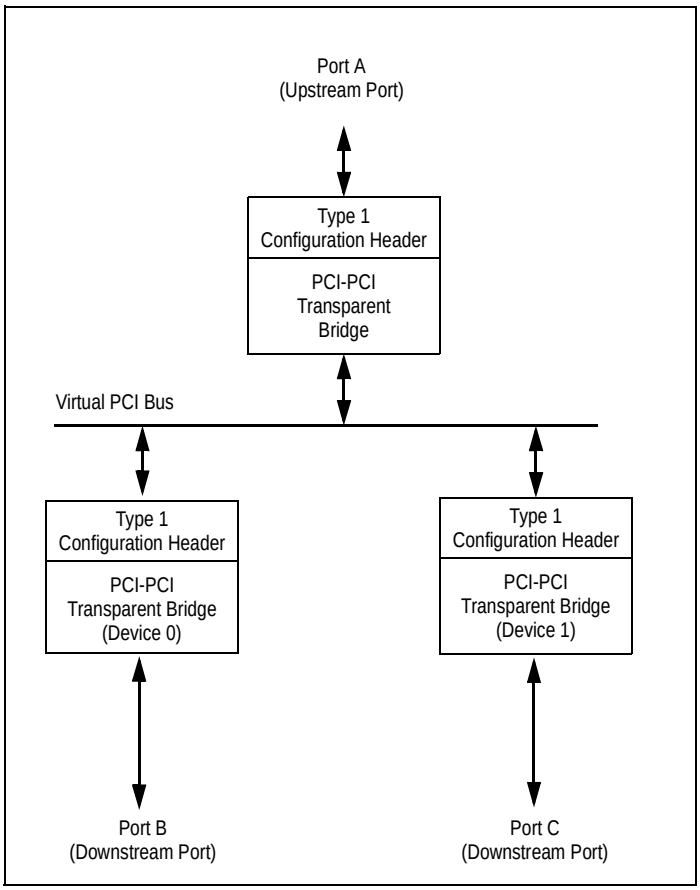

Figure 9.1 PES12N3 Functional Block Diagram in Transparent Mode

As shown in Figure 9.1, port A is configured as the upstream port and ports B and C as the downstream ports. Port B resides on the internal PCI Bus at Device 0, Function 0. Port C resides on the internal PCI Bus at Device 1, Function 0.

Notes

End-to-End CRC

PCIe® defines an optional end-to-end CRC associated with TLPs.

The PES12N3 fully supports ECRC for all TLPs that pass through the switch except for transactions utilizing gathered and routed to root complex implicit routing. For transactions received with this routing type, the ECRC is discarded and not checked and the resulting gathered message is generated without an ECRC. The only standard defined message that utilizes this method of routing is the PME_TO_Ack message.

The PES12N3 does not support ECRC for TLPs that it generates (e.g., configuration responses, INTx messages, etc.). Also, it does not support ECRC for TLPs it consumes (e.g., configuration requests). However; if a TLP is received with an ECRC, the CRC is discarded and not checked and the transaction is performed.

Interrupts

The upstream port, port A, does not support generation of legacy interrupts or MSIs. The downstream ports, ports B and C, do support generation of legacy interrupts and MSIs by their corresponding hot-plug controllers. When configured to generate INTx messages, only INTA is used.

When an unmasked hot-plug interrupt condition occurs (see section Hot-Plug Interrupts and Wake-up on page 6-6), then an MSI or interrupt message is generated by the corresponding port as described in Table 9.1. The removal of the interrupt condition occurs when unmasked hot-plug status bits are masked or cleared.

Unmasked Hot-Plug Interrupt	EN bit in MSICAP Register	INTXD bit in PCICMD Register	Action
Asserted	1	X	MSI message generated
	0	0	Assert_INTA message request generated to switch core
	0	1	None
Negated	1	X	None
	0	0	Deassert_INTA message request generated to switch core
	0	1	None

Table 9.1 Transparent Mode Port B and C Interrupts

Error Detection and Handling

This section describes error detection performed by an ingress or egress stack when the switch is configured to operate in transparent mode. Table 9.2 lists error checks performed by the physical layer and action taken when an error is detected.

Notes

Error Condition	PCIe Base 1.0a Specification Section	Action Taken
Invalid symbol or running disparity error detected.	4.2.1.3	Correctable error processing
Any TLP or DLLP framing rule violation.	4.2.2.1	Correctable error processing
8b/10b decode error	4.2.4.4	Correctable error processing
Any violation of the link initialization or training protocol	4.2.4	Uncorrectable error processing

Table 9.2 Physical Layer Errors

Table 9.3 lists error checks performed by the data link layer and action taken when an error is detected.

Error Condition	PCIe Base 1.0a Specification Section	Action Taken
TLP ending in ENDB with LCRC that does not match inverted calculated LCRC	3.5.3.1	TLP discarded
TLP received with incorrect LCRC	3.5.3.1	Correctable error processing
TLP received with sequence number not equal to NEXT_RCV_SEQ and this is not a duplicate TLP	3.5.3.1	Correctable error processing
Bad DLLP[1]	3.5.2.1	Correctable error processing
Replay time-out	3.5.2.1	Correctable error processing
REPLAY NUM rollover	3.5.2.1	Correctable error processing
Violation of flow control initialization protocol	3.3.1	Uncorrectable error processing
Sequence number specified by AckNak_Seq does not correspond to an unacknowledged TLP or to the value in ACKD_SEQ	3.5.2.1	Uncorrectable error processing

Table 9.3 Data Link Layer Errors

1. A bad DLLP is a DLP with a bad LCRC.

Table 9.4 lists error checks performed by the transaction link layer and action taken when an error is detected.

Notes

Error Condition	PCIe Base 1.0a Specification Section	Action Taken
ECRC check failure	2.7.1	None. The PES12N3 does not check ECRC for transactions that terminate in the switch.
Malformed TLP	-	Fatal error processing.
Flow control protocol error	2.6.1	Not applicable. The PES12N3 does not check for any flow control errors.
Receiver overflow	2.6.1.2	None.
Completer abort Completion time-out Unsupported request	2.3.1 2.8	Not applicable. The PES12N3 never generates non-posted transactions.
Unsupported request	2.3.1	See section Switch Core Errors on page 4-8.
Unexpected completion	2.3.2	Non-fatal error processing.

Table 9.4 Transaction Layer Errors

Table 9.5 lists the error checks performed by the transaction layer for malformed TLPs. TLP error checks are only performed when a TLP is received by the switch (i.e., by the stack associated with the port on which the switch receives the TLP). No checks are made for malformed TLPs inside the switch.

TLP Type	Error Check
All	LENGTH < Max_Payload_Size (i.e., MPS field in PCIEDCTL register)
I/O read request	LENGTH = 1 (doubleword) TC = 0, ATTR = 0 The actual packet length is correct (4 doublewords when CRC is present, 3doublewords otherwise)
I/O write request	LENGTH = 1 (doubleword) TC = 0 ATTR = 0 The actual packet length is correct (5 doublewords when CRC is present, 4 doublewords otherwise)
Configuration read request	LENGTH = 1 (doubleword) TC = 0, ATTR = 0 The actual packet length is correct (4 doublewords when CRC is present, 3doublewords otherwise)
Configuration write request	LENGTH = 1 (doubleword) TC = 0 ATTR = 0 The actual packet length is correct (5 doublewords when CRC is present, 4 doublewords otherwise)

Table 9.5 Malformed TLP Error Checks (Part 1 of 2)

Notes

TLP Type	Error Check
Memory read request (32- and 64-bit address mode)	The packet length is correct. 32-bit address mode: 3 doublewords when ECRC is present, 4 doublewords otherwise 64-bit address mode: 4 doublewords when ECRC is present, 5 doublewords otherwise
Memory write request (32- and 64-bit address mode)	The packet length is correct. 32-bit address mode: - Number of doublewords received equals LENGTH + 4 when ECRC is present - Number of doublewords received equals LENGTH + 3 when a packet does not contain ECRC 64-bit address mode -Number of doublewords received equals LENGTH + 5 when ECRC is present - Number of doublewords received equals LENGTH + 4 when a packet does not contain ECRC
Completion with data	The packet length is correct. - Number of doublewords received equals LENGTH + 3 when a packet does not contain ECRC - Number of doublewords received equals LENGTH + 4 when ECRC is present - Number of doublewords received is not greater then MaxPayloadSize + 3/4
Message or message with data	The actual packet length is correct: - Number of received double words equals LENGTH + 4 when a packet does not contain ECRC - Number of received double words equals LENGTH + 5 when ECRC is present - Number of received doublewords is not greater then MaxPayloadSize + 4/5
Interrupt messages Power management messages Error signalling messages Unlock message Set slot power limit message	TC = 0 (a message of this type is considered a malformed TLP if it uses a traffic class designator other than TC0)

Table 9.5 Malformed TLP Error Checks (Part 2 of 2)

Notes

Configuration Requests

The PCI-PCI bridges associated with ports A, B and C all have configuration registers that may be accessed with Type 0 configuration read and write requests. PCIe allows multiple outstanding configuration read and write requests. The PCI-PCI bridges in the PES12N3 each support a maximum of four outstanding configuration requests. Issuing more than four outstanding configuration requests to any of these entities may result in configuration requests being dropped.

Configuration accesses to different entities within the PES12N3 may complete out-of-order and may limit the utility of multiple outstanding PCIe configuration accesses. For example, a configuration access that modifies a secondary or subordinate bus number which is immediately followed by a Type 1 configuration access that relies on the modified value for routing may result in an error when issued concurrently, but execute properly when issued in sequence. Configuration accesses to any particular entity always complete in order.

Port Configuration Space Organization

The organization of port A, B and C configuration space is shown in Figure 9.2. While all three ports share the same basic layout, only port A (the upstream port) has switch control and status registers and only ports B and C (downstream ports) have an MSI capability structure. Port A does not contain an MSI capability structure since it does not have an associated hot-plug controller that generates interrupts.

Notes

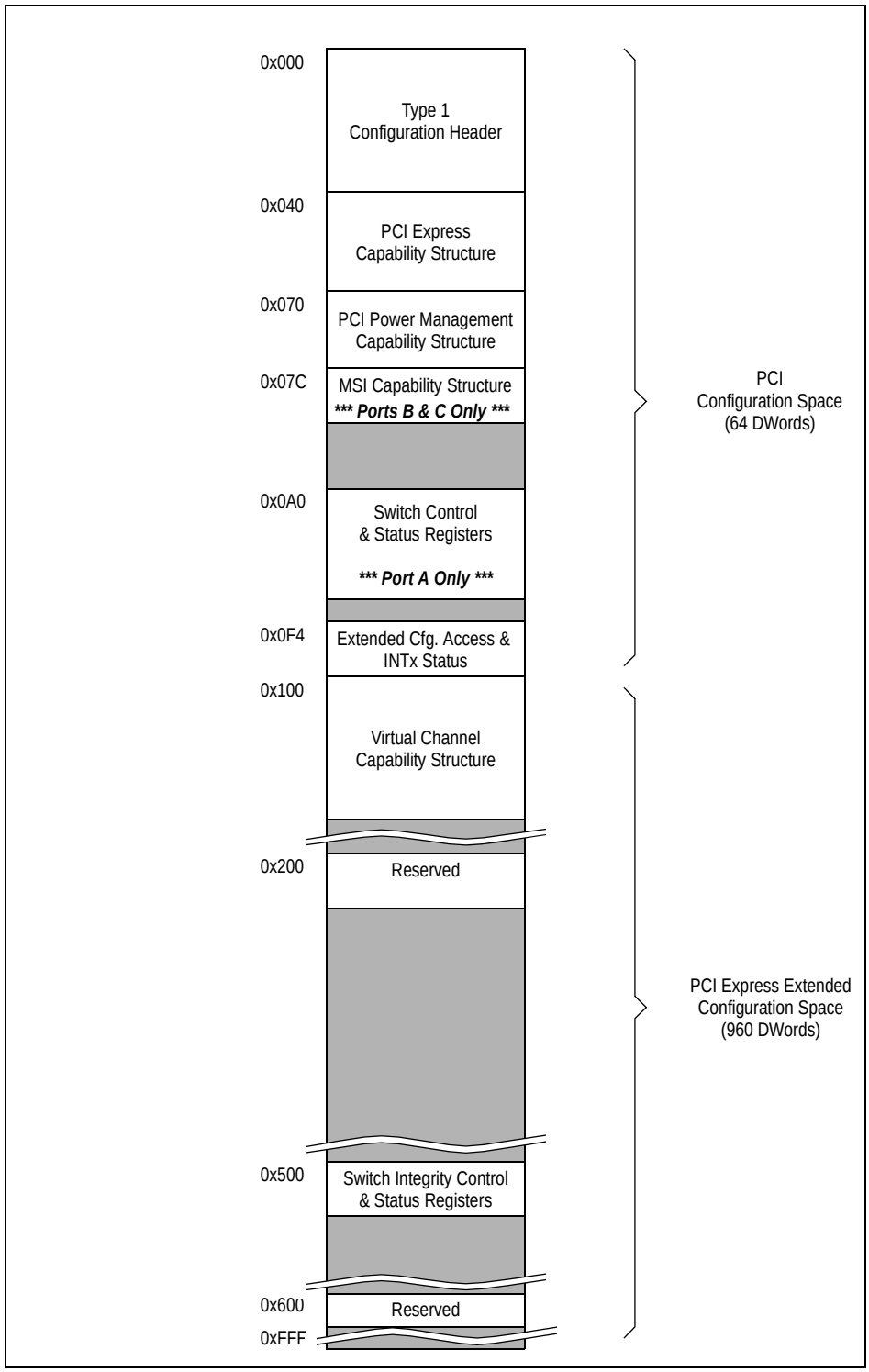

Figure 9.2 Port Configuration Space Organization

Notes

Upstream Port A Configuration Space Registers

All configuration space locations not listed in Table 9.6 return a value of zero when read. Writes to these locations are ignored and have no side-effects.

Port A configuration space registers may be read and written via the slave SMBus interface and initialized from the serial EEPROM using the CSR system address formed by adding the base address 0x0000 to the PCI configuration space offset address.

Note: In pdf format, clicking on a register name in the Register Definition column creates a jump to the appropriate register. To return to the starting place in this table, click on the same register name (in blue) in the register section.

Cfg. Offset	Size	Register Mnemonic	Register Definition
0x000	Word	PA_VID	VID - Vendor Identification (0x000) on page 9-17
0x002	Word	PA_DID	DID - Device Identification (0x002) on page 9-17
0x004	Word	PA_PCICMD	PCICMD - PCI Command (0x004) on page 9-17
0x006	Word	PA_PCISTS	PCISTS - PCI Status (0x006) on page 9-18
0x008	Byte	PA_RID	RID - Revision Identification (0x008) on page 9-19
0x009	3 Bytes	PA_CCODE	CCODE - Class Code (0x009) on page 9-19
0x00C	Byte	PA_CLS	CLS - Cache Line Size (0x00C) on page 9-20
0x00D	Byte	PA_PLTIMER	PLTIMER - Primary Latency Timer (0x00D) on page 9-20
0x00E	Byte	PA_HDR	HDR - Header Type Register (0x00E) on page 9-20
0x00F	Byte	PA_BIST	BIST - Built-in Self Test (0x00F) on page 9-20
0x010	DWord	PA_BAR0	BAR0 - Base Address Register 0 (0x010) on page 9-20
0x014	DWord	PA_BAR1	BAR1 - Base Address Register 1 (0x014) on page 9-20
0x018	Byte	PA_PBUSN	PBUSN - Primary Bus Number (0x018) on page 9-21
0x019	Byte	PA_SBUSN	SBUSN - Secondary Bus Number (0x019) on page 9-21
0x01A	Byte	PA_SUBUSN	SUBUSN - Subordinate Bus Number (0x01A) on page 9-21
0x01B	Byte	PA_SLTIMER	SLTIMER - Secondary Latency Timer (0x01B) on page 9-21
0x01C	Byte	PA_IOBASE	IOBASE - I/O Base (0x01C) on page 9-21
0x01D	Byte	PA_IOLIMIT	IOLIMIT - I/O Limit (0x01D) on page 9-22
0x01E	Word	PA_SECSTS	SECSTS - Secondary Status (0x01E) on page 9-22
0x020	Word	PA_MBASE	MBASE - Memory Base (0x020) on page 9-23
0x022	Word	PA_MLIMIT	MLIMIT - Memory Limit (0x022) on page 9-23
0x024	Word	PA_PMBASE	PMBASE - Prefetchable Memory Base (0x024) on page 9-23
0x026	Word	PA_PMLIMIT	PMLIMIT - Prefetchable Memory Limit (0x026) on page 9-24
0x028	DWord	PA_PMBASEU	PMBASEU - Prefetchable Memory Base Upper (0x028) on page 9-24
0x02C	DWord	PA_PMLIMITU	PMLIMITU - Prefetchable Memory Limit Upper (0x02C) on page 9-24
0x030	Word	PA_IOBASEU	IOBASEU - I/O Base Upper (0x030) on page 9-24

Table 9.6 Upstream Port A Configuration Space Registers (Part 1 of 3)

Notes

Cfg. Offset	Size	Register Mnemonic	Register Definition
0x032	Word	PA_IOLIMITU	IOLIMITU - I/O Limit Upper (0x032) on page 9-25
0x034	Byte	PA_CAPPTR	CAPPTR - Capabilities Pointer (0x034) on page 9-25
0x038	DWord	PA_EROMBASE	EROMBASE - Expansion ROM Base Address (0x038) on page 9-25
0x03C	Byte	PA_INTRLINE	INTRLINE - Interrupt Line (0x03C) on page 9-25
0x03D	Byte	PA_INTRPIN	INTRPIN - Interrupt PIN (0x03D) on page 9-25
0x03E	Word	PA_BCTRL	BCTRL - Bridge Control (0x03E) on page 9-26
0x040	DWord	PA_PCIECAP	PA_PCIECAP - PCI Express Capability (0x040) on page 9-10
0x044	DWord	PA_PCIEDCAP	PCIEDCAP - PCI Express Device Capabilities (0x044) on page 9-27
0x048	Word	PA_PCIEDCTL	PCIEDCTL - PCI Express Device Control (0x048) on page 9-28
0x04A	Word	PA_PCIEDSTS	PCIEDSTS - PCI Express Device Status (0x04A) on page 9-29
0x04C	DWord	PA_PCIELCAP	PCIELCAP - PCI Express Link Capabilities (0x04C) on page 9-30
0x050	Word	PA_PCIELCTL	PCIELCTL - PCI Express Link Control (0x050) on page 9-30
0x052	Word	PA_PCIELSTS	PCIELSTS - PCI Express Link Status (0x052) on page 9-31
0x070	DWord	PA_PMCAP	PMCAP - PCI Power Management Capabilities (0x070) on page 9-36
0x074	DWord	PA_PMCSR	PMCSR - PCI Power Management Control and Status (0x074) on page 9-36
0x078	DWord	PA_PMPC	PMPC - PCI Power Management Proprietary Control (0x078) on page 9-37
0x0A0	DWord	PA_SWSTS	SWSTS Switch Status (0x0A0) on page 9-40
0x0A4	DWord	PA_SWCTL	SWCTL - Switch Control (0x0A4) on page 9-42
0x0A8	DWord	PA_GPIOCS	GPIOCS - General Purpose I/O Control and Status (0x0A8) on page 9-45
0x0AC	DWord	PA_SMBUSSTS	SMBUSSTS - SMBus Status (0x0AC) on page 9-45
0x0B0	DWord	PA_SMBUSCTL	SMBUSCTL - SMBus Control (0x0B0) on page 9-46
0x0B4	DWord	PA_EEPROMINTF	EEPROMINTF - Serial EEPROM Interface (0x0B4) on page 9-47
0x0B8	DWord	PA_IOEXPINTF	IOEXPINTF - I/O Expander Interface (0x0B8) on page 9-48
0x0BC	DWord	PA_TMCTL	TMCTL - Test Mode Control (0x0BC) on page 9-55
0x0C0	DWord	PA_TMFSTS	TMFSTS - Test Mode Fail Status (0x0C0) on page 9-56
0x0C4	DWord	PA_TMSSTS	TMSSTS - Test Mode Synchronization Status (0x0C4) on page 9-56
0x0C8	DWord	PA_TMCNTCFG	TMCNTCFG - Test Mode Count Configuration (0x0C8) on page 9-57
0x0CC	DWord	PA_TMCNT0	TMCNT0 - Test Mode Count 0 (0x0CC) on page 9-59
0x0D0	DWord	PA_TMCNT1	TMCNT1 - Test Mode Count 1 (0x0D0) on page 9-59

Table 9.6 Upstream Port A Configuration Space Registers (Part 2 of 3)

Notes

Cfg. Offset	Size	Register Mnemonic	Register Definition
0x0D4	DWord	PA_TMCNT2	TMCNT2 - Test Mode Count 2 (0x0D4) on page 9-60
0x0F4	Word	PA_INTSTS	INTSTS - Interrupt Status (0x0F4) on page 9-49
0x0F8	DWord	PA_ECFGADDR	ECFGADDR - Extended Configuration Space Access Address (0x0F8) on page 9-49
0x0FC	DWord	PA_ECFGDATA	ECFGDATA - Extended Configuration Space Access Data (0x0FC) on page 9-50
0x100	DWord	PA_PCIEVCECAP	PCIEVCECAP - PCI Express Virtual Channel Enhanced Capability Header (0x100) on page 9-50
0x104	DWord	PA_PVCCAP1	PVCCAP1- Port VC Capability 1 (0x104) on page 9-50
0x110	DWord	PA_VCR0CAP	VCR0CAP- VC Resource 0 Capability (0x110) on page 9-51
0x114	DWord	PA_VCR0CTL	VCR0CTL- VC Resource 0 Control (0x114) on page 9-51
0x118	DWord	PA_VCR0STS	VCR0STS - VC Resource 0 Status (0x118) on page 9-52
0x120	DWord	PA_VCR0TBL0	VCR0TBL0 - VC Resource 0 Arbitration Table Entry 0 (0x120) on page 9-53
0x124	DWord	PA_VCR0TBL1	VCR0TBL1 - VC Resource 0 Arbitration Table Entry 1 (0x124) on page 9-54
0x200 — 0x414			Reserved
0x500	Dword	PA_SWSICTL	SWSICTL - Switch System Integrity Control (0x500) on page 9-60
0x504	Dword	PA_SWSIPECNT	SWSIPECNT - Switch System Integrity Parity Error Count (0x504) on page 9-61
0x508	Dword	PA_SWSITDCNT	SWSITDCNT - Switch System Integrity Time-Out Drop Count (0x508) on page 9-61
0x510 — 0x61C			Reserved

Table 9.6 Upstream Port A Configuration Space Registers (Part 3 of 3)

Register Specialization

PA_PCIECAP - PCI Express Capability (0x040)

Bit Field	Field Name	Type	Default Value	Description
7:0	CAPID	—	—	PCIECAP - PCI Express Capability (0x040) on page 9-26
15:8	NXTPTR	—	—	PCIECAP - PCI Express Capability (0x040) on page 9-26
19:16	VER	—	—	PCIECAP - PCI Express Capability (0x040) on page 9-26
23:20	TYPE	RO	0x5	**Port Type.** Upstream port of a PCI-Express switch.

Notes

Bit Field	Field Name	Type	Default Value	Description
24	SLOT	RWL	0x0	**Slot Implemented.** This bit is set when the PCI Express link associated with this Port is connected to a slot. Does not apply to the upstream port and should be set to zero.
29:25	IMN	—	—	PCIECAP - PCI Express Capability (0x040) on page 9-26
31:30	Reserved	RO	0x0	Reserved field.

Downstream Port B Configuration Space Registers

All configuration space locations not listed in Table 9.7 return a value of zero when read. Writes to these locations are ignored and have no side-effects. Port B configuration space registers may be read and written via the slave SMBus interface and initialized from the serial EEPROM using the CSR system address formed by adding the base address 0x1000 to the PCI configuration space offset address.

Cfg. Offset	Size	Register Mnemonic	Register Definition
0x000	Word	PB_VID	VID - Vendor Identification (0x000) on page 9-17
0x002	Word	PB_DID	DID - Device Identification (0x002) on page 9-17
0x004	Word	PB_PCICMD	PCICMD - PCI Command (0x004) on page 9-17
0x006	Word	PB_PCISTS	PCISTS - PCI Status (0x006) on page 9-18
0x008	Byte	PB_RID	RID - Revision Identification (0x008) on page 9-19
0x009	3 Bytes	PB_CCODE	CCODE - Class Code (0x009) on page 9-19
0x00C	Byte	PB_CLS	CLS - Cache Line Size (0x00C) on page 9-20
0x00D	Byte	PB_PLTIMER	PLTIMER - Primary Latency Timer (0x00D) on page 9-20
0x00E	Byte	PB_HDR	HDR - Header Type Register (0x00E) on page 9-20
0x00F	Byte	PB_BIST	BIST - Built-in Self Test (0x00F) on page 9-20
0x010	DWord	PB_BAR0	BAR0 - Base Address Register 0 (0x010) on page 9-20
0x014	DWord	PB_BAR1	BAR1 - Base Address Register 1 (0x014) on page 9-20
0x018	Byte	PB_PBUSN	PBUSN - Primary Bus Number (0x018) on page 9-21
0x019	Byte	PB_SBUSN	SBUSN - Secondary Bus Number (0x019) on page 9-21
0x01A	Byte	PB_SUBUSN	SUBUSN - Subordinate Bus Number (0x01A) on page 9-21
0x01B	Byte	PB_SLTIMER	SLTIMER - Secondary Latency Timer (0x01B) on page 9-21
0x01C	Byte	PB_IOBASE	IOBASE - I/O Base (0x01C) on page 9-21
0x01D	Byte	PB_IOLIMIT	IOLIMIT - I/O Limit (0x01D) on page 9-22
0x01E	Word	PB_SECSTS	SECSTS - Secondary Status (0x01E) on page 9-22
0x020	Word	PB_MBASE	MBASE - Memory Base (0x020) on page 9-23
0x022	Word	PB_MLIMIT	MLIMIT - Memory Limit (0x022) on page 9-23
0x024	Word	PB_PMBASE	PMBASE - Prefetchable Memory Base (0x024) on page 9-23
0x026	Word	PB_PMLIMIT	PMLIMIT - Prefetchable Memory Limit (0x026) on page 9-24

Table 9.7 Downstream Port B Configuration Space Registers (Part 1 of 3)

Notes

Cfg. Offset	Size	Register Mnemonic	Register Definition
0x028	DWord	PB_PMBASEU	PMBASEU - Prefetchable Memory Base Upper (0x028) on page 9-24
0x02C	DWord	PB_PMLIMITU	PMLIMITU - Prefetchable Memory Limit Upper (0x02C) on page 9-24
0x030	Word	PB_IOBASEU	IOBASEU - I/O Base Upper (0x030) on page 9-24
0x032	Word	PB_IOLIMITU	IOLIMITU - I/O Limit Upper (0x032) on page 9-25
0x034	Byte	PB_CAPPTR	CAPPTR - Capabilities Pointer (0x034) on page 9-25
0x038	DWord	PB_EROMBASE	EROMBASE - Expansion ROM Base Address (0x038) on page 9-25
0x03C	Byte	PB_INTRLINE	INTRLINE - Interrupt Line (0x03C) on page 9-25
0x03D	Byte	PB_INTRPIN	INTRPIN - Interrupt PIN (0x03D) on page 9-25
0x03E	Word	PB_BCTRL	BCTRL - Bridge Control (0x03E) on page 9-26
0x040	DWord	PB_PCIECAP	PB_PCIECAP - PCI Express Capability (0x040) on page 9-13
0x044	DWord	PB_PCIEDCAP	PCIEDCAP - PCI Express Device Capabilities (0x044) on page 9-27
0x048	Word	PB_PCIEDCTL	PCIEDCTL - PCI Express Device Control (0x048) on page 9-28
0x04A	Word	PB_PCIEDSTS	PCIEDSTS - PCI Express Device Status (0x04A) on page 9-29
0x04C	DWord	PB_PCIELCAP	PCIELCAP - PCI Express Link Capabilities (0x04C) on page 9-30
0x050	Word	PB_PCIELCTL	PCIELCTL - PCI Express Link Control (0x050) on page 9-30
0x052	Word	PB_PCIELSTS	PCIELSTS - PCI Express Link Status (0x052) on page 9-31
0x054	DWord	PB_PCIESCAP	PCIESCAP - PCI Express Slot Capabilities (0x054) on page 9-32
0x058	Word	PB_PCIESCTL	PCIESCTL - PCI Express Slot Control (0x058) on page 9-33
0x05A	Word	PB_PCIESSTS	PCIESSTS - PCI Express Slot Status (0x05A) on page 9-35
0x070	DWord	PB_PMCAP	PMCAP - PCI Power Management Capabilities (0x070) on page 9-36
0x074	DWord	PB_PMCSR	PMCSR - PCI Power Management Control and Status (0x074) on page 9-36
0x078	DWord	PB_PMPC	PMPC - PCI Power Management Proprietary Control (0x078) on page 9-37
0x07C	DWord	PB_MSICAP	MSICAP - Message Signaled Interrupt Capability and Control (0x07C) on page 9-39
0x080	DWord	PB_MSIADDR	MSIADDR - Message Signaled Interrupt Address (0x080) on page 9-39
0x084	DWord	PB_MSIUADDR	MSIUADDR - Message Signaled Interrupt Upper Address (0x084) on page 9-40
0x088	DWord	PB_MSIMDATA	MSIMDATA - Message Signaled Interrupt Message Data (0x088) on page 9-40
0x0F4	Word	PB_INTSTS	INTSTS - Interrupt Status (0x0F4) on page 9-49

Table 9.7 Downstream Port B Configuration Space Registers (Part 2 of 3)

Notes

Cfg. Offset	Size	Register Mnemonic	Register Definition
0x0F8	DWord	PB_ECFGADDR	ECFGADDR - Extended Configuration Space Access Address (0x0F8) on page 9-49
0x0FC	DWord	PB_ECFGDATA	ECFGDATA - Extended Configuration Space Access Data (0x0FC) on page 9-50
0x100	DWord	PB_PCIEVCECAP	PCIEVCECAP - PCI Express Virtual Channel Enhanced Capability Header (0x100) on page 9-50
0x104	DWord	PB_PVCCAP1	PVCCAP1- Port VC Capability 1 (0x104) on page 9-50
0x110	DWord	PB_VCR0CAP	VCR0CAP- VC Resource 0 Capability (0x110) on page 9-51
0x114	DWord	PB_VCR0CTL	VCR0CTL- VC Resource 0 Control (0x114) on page 9-51
0x118	DWord	PB_VCR0STS	VCR0STS - VC Resource 0 Status (0x118) on page 9-52
0x120	DWord	PB_VCR0TBL0	VCR0TBL0 - VC Resource 0 Arbitration Table Entry 0 (0x120) on page 9-53
0x124	DWord	PB_VCR0TBL1	VCR0TBL1 - VC Resource 0 Arbitration Table Entry 1 (0x124) on page 9-54
0x200 — 0x414			Reserved
0x500	Dword	PB_SWSICTL	SWSICTL - Switch System Integrity Control (0x500) on page 9-60
0x504	Dword	PB_SWSIPECNT	SWSIPECNT - Switch System Integrity Parity Error Count (0x504) on page 9-61
0x508	Dword	PB_SWSITDCNT	SWSITDCNT - Switch System Integrity Time-Out Drop Count (0x508) on page 9-61
0x510 — 0x61C			Reserved

Table 9.7 Downstream Port B Configuration Space Registers (Part 3 of 3)

Register Specialization

PB_PCIECAP - PCI Express Capability (0x040)

Bit Field	Field Name	Type	Default Value	Description
7:0	CAPID	—	—	PCIECAP - PCI Express Capability (0x040) on page 9-26
15:8	NXTPTR	—	—	PCIECAP - PCI Express Capability (0x040) on page 9-26
19:16	VER	—	—	PCIECAP - PCI Express Capability (0x040) on page 9-26
23:20	TYPE	RO	0x6	**Port Type.** Downstream port of a PCI-Express switch.
24	SLOT	RWL	0x0	**Slot Implemented.** This bit is set when the PCI Express link associated with this Port is connected to a slot.
29:25	IMN	—	—	PCIECAP - PCI Express Capability (0x040) on page 9-26
31:30	Reserved	RO	0x0	Reserved field.

Notes

Downstream Port C Configuration Space Registers

All configuration space locations not listed in Table 9.8 return a value of zero when read. Writes to these locations are ignored and have no side-effects.

Port C configuration space registers may be read and written via the slave SMBus interface and initialized from the serial EEPROM using the CSR system address formed by adding the base address 0x2000 to the PCI configuration space offset address.

Note: In pdf format, clicking on a register name in the Register Definition column creates a jump to the appropriate register. To return to the starting place in this table, click on the same register name (in blue) in the register section.

Cfg. Offset	Size	Register Mnemonic	Register Definition
0x000	Word	PC_VID	VID - Vendor Identification (0x000) on page 9-17
0x002	Word	PC_DID	DID - Device Identification (0x002) on page 9-17
0x004	Word	PC_PCICMD	PCICMD - PCI Command (0x004) on page 9-17
0x006	Word	PC_PCISTS	PCISTS - PCI Status (0x006) on page 9-18
0x008	Byte	PC_RID	RID - Revision Identification (0x008) on page 9-19
0x009	3 Bytes	PC_CCODE	CCODE - Class Code (0x009) on page 9-19
0x00C	Byte	PC_CLS	CLS - Cache Line Size (0x00C) on page 9-20
0x00D	Byte	PC_PLTIMER	PLTIMER - Primary Latency Timer (0x00D) on page 9-20
0x00E	Byte	PC_HDR	HDR - Header Type Register (0x00E) on page 9-20
0x00F	Byte	PC_BIST	BIST - Built-in Self Test (0x00F) on page 9-20
0x010	DWord	PC_BAR0	BAR0 - Base Address Register 0 (0x010) on page 9-20
0x014	DWord	PC_BAR1	BAR1 - Base Address Register 1 (0x014) on page 9-20
0x018	Byte	PC_PBUSN	PBUSN - Primary Bus Number (0x018) on page 9-21
0x019	Byte	PC_SBUSN	SBUSN - Secondary Bus Number (0x019) on page 9-21
0x01A	Byte	PC_SUBUSN	SUBUSN - Subordinate Bus Number (0x01A) on page 9-21
0x01B	Byte	PC_SLTIMER	SLTIMER - Secondary Latency Timer (0x01B) on page 9-21
0x01C	Byte	PC_IOBASE	IOBASE - I/O Base (0x01C) on page 9-21
0x01D	Byte	PC_IOLIMIT	IOLIMIT - I/O Limit (0x01D) on page 9-22
0x01E	Word	PC_SECSTS	SECSTS - Secondary Status (0x01E) on page 9-22
0x020	Word	PC_MBASE	MBASE - Memory Base (0x020) on page 9-23
0x022	Word	PC_MLIMIT	MLIMIT - Memory Limit (0x022) on page 9-23
0x024	Word	PC_PMBASE	PMBASE - Prefetchable Memory Base (0x024) on page 9-23
0x026	Word	PC_PMLIMIT	PMLIMIT - Prefetchable Memory Limit (0x026) on page 9-24
0x028	DWord	PC_PMBASEU	PMBASEU - Prefetchable Memory Base Upper (0x028) on page 9-24
0x02C	DWord	PC_PMLIMITU	PMLIMITU - Prefetchable Memory Limit Upper (0x02C) on page 9-24
0x030	Word	PC_IOBASEU	IOBASEU - I/O Base Upper (0x030) on page 9-24

Table 9.8 Downstream Port C Configuration Space Registers (Part 1 of 3)

Notes

Cfg. Offset	Size	Register Mnemonic	Register Definition
0x032	Word	PC_IOLIMITU	IOLIMITU - I/O Limit Upper (0x032) on page 9-25
0x034	Byte	PC_CAPPTR	CAPPTR - Capabilities Pointer (0x034) on page 9-25
0x038	DWord	PC_EROMBASE	EROMBASE - Expansion ROM Base Address (0x038) on page 9-25
0x03C	Byte	PC_INTRLINE	INTRLINE - Interrupt Line (0x03C) on page 9-25
0x03D	Byte	PC_INTRPIN	INTRPIN - Interrupt PIN (0x03D) on page 9-25
0x03E	Word	PC_BCTRL	BCTRL - Bridge Control (0x03E) on page 9-26
0x040	DWord	PC_PCIECAP	PC_PCIECAP - PCI Express Capability (0x040) on page 9-16
0x044	DWord	PC_PCIEDCAP	PCIEDCAP - PCI Express Device Capabilities (0x044) on page 9-27
0x048	Word	PC_PCIEDCTL	PCIEDCTL - PCI Express Device Control (0x048) on page 9-28
0x04A	Word	PC_PCIEDSTS	PCIEDSTS - PCI Express Device Status (0x04A) on page 9-29
0x04C	DWord	PC_PCIELCAP	PCIELCAP - PCI Express Link Capabilities (0x04C) on page 9-30
0x050	Word	PC_PCIELCTL	PCIELCTL - PCI Express Link Control (0x050) on page 9-30
0x052	Word	PC_PCIELSTS	PCIELSTS - PCI Express Link Status (0x052) on page 9-31
0x054	DWord	PC_PCIESCAP	PCIESCAP - PCI Express Slot Capabilities (0x054) on page 9-32
0x058	Word	PC_PCIESCTL	PCIESCTL - PCI Express Slot Control (0x058) on page 9-33
0x05A	Word	PC_PCIESSTS	PCIESSTS - PCI Express Slot Status (0x05A) on page 9-35
0x070	DWord	PC_PMCAP	PMCAP - PCI Power Management Capabilities (0x070) on page 9-36
0x074	DWord	PC_PMCSR	PMCSR - PCI Power Management Control and Status (0x074) on page 9-36
0x078	DWord	PC_PMPC	PMPC - PCI Power Management Proprietary Control (0x078) on page 9-37
0x07C	DWord	PC_MSICAP	MSICAP - Message Signaled Interrupt Capability and Control (0x07C) on page 9-39
0x080	DWord	PC_MSIADDR	MSIADDR - Message Signaled Interrupt Address (0x080) on page 9-39
0x084	DWord	PC_MSIUADDR	MSIUADDR - Message Signaled Interrupt Upper Address (0x084) on page 9-40
0x088	DWord	PC_MSIMDATA	MSIMDATA - Message Signaled Interrupt Message Data (0x088) on page 9-40
0x0F4	Word	PC_INTSTS	INTSTS - Interrupt Status (0x0F4) on page 9-49
0x0F8	DWord	PC_ECFGADDR	ECFGADDR - Extended Configuration Space Access Address (0x0F8) on page 9-49
0x0FC	DWord	PC_ECFGDATA	ECFGDATA - Extended Configuration Space Access Data (0x0FC) on page 9-50

Table 9.8 Downstream Port C Configuration Space Registers (Part 2 of 3)

Notes

Cfg. Offset	Size	Register Mnemonic	Register Definition
0x100	DWord	PC_PCIEVCECAP	PCIEVCECAP - PCI Express Virtual Channel Enhanced Capability Header (0x100) on page 9-50
0x104	DWord	PC_PVCCAP1	PVCCAP1- Port VC Capability 1 (0x104) on page 9-50
0x110	DWord	PC_VCR0CAP	VCR0CAP- VC Resource 0 Capability (0x110) on page 9-51
0x114	DWord	PC_VCR0CTL	VCR0CTL- VC Resource 0 Control (0x114) on page 9-51
0x118	DWord	PC_VCR0STS	VCR0STS - VC Resource 0 Status (0x118) on page 9-52
0x120	DWord	PC_VCR0TBL0	VCR0TBL0 - VC Resource 0 Arbitration Table Entry 0 (0x120) on page 9-53
0x124	DWord	PC_VCR0TBL1	VCR0TBL1 - VC Resource 0 Arbitration Table Entry 1 (0x124) on page 9-54
0x200 — 0x414			Reserved
0x500	Dword	PC_SWSICTL	SWSICTL - Switch System Integrity Control (0x500) on page 9-60
0x504	Dword	PC_SWSIPECNT	SWSIPECNT - Switch System Integrity Parity Error Count (0x504) on page 9-61
0x508	Dword	PC_SWSITDCNT	SWSITDCNT - Switch System Integrity Time-Out Drop Count (0x508) on page 9-61
0x510 — 0x61C			Reserved

Table 9.8 Downstream Port C Configuration Space Registers (Part 3 of 3)

Register Specialization

PC_PCIECAP - PCI Express Capability (0x040)

Bit Field	Field Name	Type	Default Value	Description
7:0	CAPID	—	—	PCIECAP - PCI Express Capability (0x040) on page 9-26
15:8	NXTPTR	—	—	PCIECAP - PCI Express Capability (0x040) on page 9-26
19:16	VER	—	—	PCIECAP - PCI Express Capability (0x040) on page 9-26
23:20	TYPE	RO	0x6	**Port Type.** Downstream port of a PCI-Express switch.
24	SLOT	RWL	0x0	**Slot Implemented.** This bit is set when the PCI Express link associated with this Port is connected to a slot.
29:25	IMN	—	—	PCIECAP - PCI Express Capability (0x040) on page 9-26.
31:30	Reserved	RO	0x0	Reserved field.

Notes

Generic PCI to PCI Bridge Register Definition

Type 1 Configuration Header Registers

VID - Vendor Identification (0x000)

Bit Field	Field Name	Type	Default Value	Description
15:0	VID	RO	0x111D	**Vendor Identification.** This field contains the 16-bit vendor ID value assigned to IDT. See section Vendor ID on page 1-4.

DID - Device Identification (0x002)

Bit Field	Field Name	Type	Default Value	Description
15:0	DID	RO	-	**Device Identification.** This field contains the 16-bit device ID assigned by IDT to this transparent bridge. See section Device ID on page 1-4.

PCICMD - PCI Command (0x004)

Bit Field	Field Name	Type	Default Value	Description
0	IOAE	RW	0x0	**I/O Access Enable.** When this bit is cleared, the bridge does not respond to I/O accesses from the primary bus specified by IOBASE and IOLIMIT. 0x0 - (disable) Disable I/O space. 0x1 - (enable) Enable I/O space.
1	MAE	RW	0x0	**Memory Access Enable.** When this bit is cleared, the bridge does not respond to memory and prefetchable memory space access from the primary bus specified by MBASE, MLIMIT, PMBASE and PMLIMIT. 0x0 - (disable) Disable memory space. 0x1 - (enable) Enable memory space.
2	BME	RW	0x0	**Bus Master Enable.** When this bit is cleared, the bridge does not issue requests (e.g., memory, I/O and MSIs since they are in-band writes) on behalf of subordinate devices and responds to non-posted transactions with a Unsupported Request (UR) completion. This bit does not affect completions in either direction or the forwarding of non memory or I/O requests. 0x0 - (disable) Disable request forwarding. 0x1 - (enable) Enable request forwarding.
3	SSE	RO	0x0	**Special Cycle Enable.** Not applicable.
4	MWI	RO	0x0	**Memory Write Invalidate.** Not applicable.
5	VGAS	RO	0x0	**VGA Palette Snoop**. Not applicable.

Notes

Bit Field	Field Name	Type	Default Value	Description
6	PERRE	RW	0x0	**Parity Error Enable.** The Master Data Parity Error bit is set in the PCI Status register (PCISTS) if this bit is set and the bridge receives a poisoned completion or a poisoned write. If this bit is cleared, then the Master Data Parity Error bit in the PCI Status register is never set. 0x0 - (disable) Disable Master Parity Error bit reporting. 0x1 - (enable) Enable Master Parity Error bit reporting.
7	ADSTEP	RO	0x0	**Address Data Stepping.** Not applicable.
8	SERRE	RW	0x0	**SERR Enable.** Non-fatal and fatal errors detected by the bridge are reported to the Root Complex when this bit is set or the bits in the PCI Express Device Control register are set (see PCIEDCTL - PCI Express Device Control (0x048)). 0x0 - (disable) Disable non-fatal and fatal error reporting if also disabled in Device Control register. 0x1 - (enable) Enable non-fatal and fatal error reporting.
9	FB2B	RO	0x0	**Fast Back-to-Back Enable.** Not applicable.
10	INTXD	RW	0x0	**INTx Disable.** Controls the ability of the PCI-PCI bridge to generate an INTx interrupt message.
15:11	Reserved	RO	0x0	Reserved field.

PCISTS - PCI Status (0x006)

Bit Field	Field Name	Type	Default Value	Description
2:0	Reserved	RO	0x0	Reserved.
3	INTS	RO	0x0	**INTx Status.** This bit is set when an INTx interrupt is pending from the device. INTx emulation interrupts forwarded by switch ports from devices downstream of the bridge are not reflected in this bit. For ports B and C, this bit is set if an interrupt has been “asserted” by the corresponding port’s hot-plug controller. For port A, this field is always zero.
4	CAPL	RO	0x1	**Capabilities List.** This bit is hardwired to one to indicate that the bridge implements an extended capability list item.
5	C66MHZ	RO	0x0	**66 MHz Capable.** Not applicable.
6	Reserved	RO	0x0	Reserved.
7	FB2B	RO	0x0	**Fast Back-to-Back (FB2B).** Not applicable.
8	MDPED	RW1C	0x0	**Master Data Parity Error Detected.** This bit is set when the PERRE bit is set in the PCI Command register and the bridge receives a poisoned completion or poisoned write request on the primary side of the bridge. 0x0 - (noerror) no error. 0x1 - (error) Poisoned write request or completion received on primary side.

Notes

Bit Field	Field Name	Type	Default Value	Description
10:9	DEVT	RO	0x0	**DEVSEL# Timing.** Not applicable.
11	STAS	RO	0x0	**Signalled Target Abort.** Not applicable since a target abort is never signalled.
12	RTAS	RO	0x0	**Received Target Abort.** Not applicable.
13	RMAS	RO	0x0	**Received Master Abort.** Not applicable.
14	SSE	RW1C	0x0	**Signalled System Error.** This bit is set when the bridge sends a ERR_FATAL or ERR_NONFATAL message and the SERR Enable (SERRE) bit is set in the PCICMD register. 0x0 - (noerror) no error. 0x1 - (error) This bit is set when a fatal or non-fatal error is signalled.
15	DPE	RW1C	0x0	**Detected Parity Error.** This bit is set by the bridge whenever it receives a poisoned TLP on the primary side regardless of the state of the PERRE bit in the PCI Command register. For downstream ports, this bit is also set when a internal switch parity error is detected. See section Data Integrity on page 4-4.

RID - Revision Identification (0x008)

Bit Field	Field Name	Type	Default Value	Description
7:0	RID	RWL	—	**Revision ID.** This field contains the revision identification number for the device. See section Revision ID on page 1-5.

CCODE - Class Code (0x009)

Bit Field	Field Name	Type	Default Value	Description
7:0	INTF	RO	0x00	**Interface.** This value indicates that the device is a PCI-PCI bridge that does not support subtractive decode.
15:8	SUB	RO	0x04	**Sub Class Code.** This value indicates that the device is a PCI-PCI bridge.
23:16	BASE	RO	0x06	**Base Class Code.** This value indicates that the device is a bridge.

Notes

CLS - Cache Line Size (0x00C)

Bit Field	Field Name	Type	Default Value	Description
7:0	CLS	RW	0x00	**Cache Line Size.** This field has no effect on the bridge's functionality but may be read and written by software. This field is implemented for compatibility with legacy software.

PLTIMER - Primary Latency Timer (0x00D)

Bit Field	Field Name	Type	Default Value	Description
7:0	PLTIMER	RO	0x00	**Primary Latency Timer.** Not applicable.

HDR - Header Type Register (0x00E)

Bit Field	Field Name	Type	Default Value	Description
7:0	HDR	RO	0x01	**Header Type.** This value indicates a type 1 header with a single function bridge layout.

BIST - Built-in Self Test (0x00F)

Bit Field	Field Name	Type	Default Value	Description
7:0	BIST	RO	0x0	**BIST.** This value indicates that the bridge does not implement BIST.

BAR0 - Base Address Register 0 (0x010)

Bit Field	Field Name	Type	Default Value	Description
31:0	BAR	RO	0x0	**Base Address Register.** Not applicable.

BAR1 - Base Address Register 1 (0x014)

Bit Field	Field Name	Type	Default Value	Description
31:0	BAR	RO	0x0	**Base Address Register.** Not applicable.

Notes

PBUSN - Primary Bus Number (0x018)

Bit Field	Field Name	Type	Default Value	Description
7:0	PBUSN	RW	0x0	**Primary Bus Number.** This field is used to record the bus number of the PCI bus segment to which the primary interface of the bridge is connected.

SBUSN - Secondary Bus Number (0x019)

Bit Field	Field Name	Type	Default Value	Description
7:0	SBUSN	RW	0x0	**Secondary Bus Number.** This field is used to record the bus number of the PCI bus segment to which the secondary interface of the bridge is connected.

SUBUSN - Subordinate Bus Number (0x01A)

Bit Field	Field Name	Type	Default Value	Description
7:0	SUBUSN	RW	0x0	**Subordinate Bus Number.** The Subordinate Bus Number register is used to record the bus number of the highest numbered PCI bus segment which is behind (or subordinate to) the bridge.

SLTIMER - Secondary Latency Timer (0x01B)

Bit Field	Field Name	Type	Default Value	Description
7:0	SLTIMER	RO	0x0	**Secondary Latency Timer.** Not applicable.

IOBASE - I/O Base (0x01C)

Bit Field	Field Name	Type	Default Value	Description
0	IOCAP	RWL	0x1	**I/O Capability.** Indicates if the bridge supports 16-bit or 32-bit I/O addressing. 0x0 - (io16) 16-bit I/O addressing. 0x1 - (io32) 32-bit I/O addressing.
3:1	Reserved	RO	0x0	
7:4	IOBASE	RW	0xF	**I/O Base.** The IOBASE and IOLIMIT registers are used to control the forwarding of I/O transactions between the primary and secondary interfaces of the bridge. This field contains A[15:12] of the lowest I/O address aligned on a 4KB boundary that is below the primary interface of the bridge.

Notes

IOLIMIT - I/O Limit (0x01D)

Bit Field	Field Name	Type	Default Value	Description
0	IOCAP	RO	0x1	**I/O Capability.** Indicates if the bridge supports 16-bit or 32-bit I/O addressing. This bit always reflects the value of the IOCAP field in the IOBASE register.
3:1	Reserved	RO	0x0	
7:4	IOLIMIT	RW	0x0	**I/O Limit.** The IOBASE and IOLIMIT registers are used to control the forwarding of I/O transactions between the primary and secondary interfaces of the bridge. This field contains A[15:12] of the highest I/O address, with A[11:0] assumed to be 0xFFF, that is below the primary interface of the bridge.

SECSTS - Secondary Status (0x01E)

Bit Field	Field Name	Type	Default Value	Description
7:0	Reserved	RO	0x0	
8	MDPED	RW1C	0x0	**Master Data Parity Error.** This bit is controlled by the Parity Error Response Enable bit in the Bridge Control register. If the Parity Response Enable bit is cleared, then this bit is never set. Otherwise, this bit is set if the secondary side of the bridge forwards a poisoned TLP received on the primary side.
10:9	DVSEL	RO	0x0	Not applicable.
11	STAS	RO	0x0	**Signalled Target Abort Status.** Not applicable.
12	RTAS	RO	0x0	**Received Target Abort Status.** Not applicable.
13	RMAS	RO	0x0	**Received Master Abort Status.** Not applicable.
14	SSE	RW1C	0x0	**Signalled System Error.** This bit is controlled by the SERR enable bit in the Bridge Control (BCTRL) register. If the SERRE bit is cleared in BCTRL, then this bit is never set. Otherwise, this bit is set if the secondary side of the bridge receives an ERR_FATAL or ERR_NONFATAL message.
15	DPE	RW1C	0x0	**Detected Parity Error.** This bit is set by the bridge whenever it receives a poisoned TLP on the secondary side regardless of the state of the PERRE bit in the PCI Command register For the upstream port, this bit is also set when a internal switch parity error is detected. See section Data Integrity on page 4-4.

Notes

MBASE - Memory Base (0x020)

Bit Field	Field Name	Type	Default Value	Description
3:0	Reserved	RO	0x0	
15:4	MBASE	RW	0xFFF	**Memory Address Base.** The MBASE and MLIMIT registers are used to control the forwarding of non-prefetchable transactions between the primary and secondary interfaces of the bridge. This field contains A[31:20] of the lowest address aligned on a 1MB boundary that is below the primary interface of the bridge.

MLIMIT - Memory Limit (0x022)

Bit Field	Field Name	Type	Default Value	Description
3:0	Reserved	RO	0x0	
15:4	MLIMIT	RW	0x0	**Memory Address Limit.** The MBASE and MLIMIT registers are used to control the forwarding of non-prefetchable transactions between the primary and secondary interfaces of the bridge. This field contains A[31:20] of the highest address, with A[19:0] assumed to be 0xF_FFFF, that is below the primary interface of the bridge.

PMBASE - Prefetchable Memory Base (0x024)

Bit Field	Field Name	Type	Default Value	Description
0	PMCAP	RWL	0x1	**Prefetchable Memory Capability.** Indicates if the bridge supports 32-bit or 64-bit prefetchable memory addressing. 0x0 - (prefmem32) 32-bit prefetchable memory addressing. 0x1 - (prefmem64) 64-bit prefetchable memory addressing.
3:1	Reserved	RO	0x0	
15:4	PMBASE	RW	0xFFF	**Prefetchable Memory Address Base.** The PMBASE, PMBASEU, PMLIMIT and PMLIMITU registers are used to control the forwarding of prefetchable transactions between the primary and secondary interfaces of the bridge. This field contains A[31:20] of the lowest memory address aligned on a 1MB boundary that is below the primary interface of the bridge. PMBASEU specifies the remaining bits.

Notes

PMLIMIT - Prefetchable Memory Limit (0x026)

Bit Field	Field Name	Type	Default Value	Description
0	PMCAP	RO	0x1	**Prefetchable Memory Capability.** Indicates if the bridge supports 32-bit or 64-bit prefetchable memory addressing. This bit always reflects the value in the PMCAP field in the PMBASE register.
3:1	Reserved	RO	0x0	
15:4	PMLIMIT	RW	0x0	**Prefetchable Memory Address Limit.** The PMBASE, PMBASEU, PMLIMIT and PMLIMITU registers are used to control the forwarding of prefetchable transactions between the primary and secondary interfaces of the bridge. This field contains A[31:20] of the highest memory address, with A[19:0] assumed to be 0xF_FFFF, that is below the primary interface of the bridge. PMLIMITU specifies the remaining bits.

PMBASEU - Prefetchable Memory Base Upper (0x028)

Bit Field	Field Name	Type	Default Value	Description
31:0	PMBASEU	RW	0xFFFF_FFFF	**Prefetchable Memory Address Base Upper.** This field specifies the upper 32-bits of PMBASE when 64-bit addressing is used. When the PMCAP field in the PMBASE register is cleared, this field becomes read-only with a value of zero.

PMLIMITU - Prefetchable Memory Limit Upper (0x02C)

Bit Field	Field Name	Type	Default Value	Description
31:0	PMLIMITU	RW	0x0	**Prefetchable Memory Address Limit Upper.** This field specifies the upper 32-bits of PMLIMIT. When the PMCAP field in the PMBASE register is cleared, this field becomes read-only with a value of zero.

IOBASEU - I/O Base Upper (0x030)

Bit Field	Field Name	Type	Default Value	Description
15:0	IOBASEU	RW	0xFFFF	**I/O Address Base Upper.** This field specifies the upper 16-bits of IOBASE. When the IOCAP field in the IOBASE register is cleared, this field becomes read-only with a value of zero.

Notes

IOLIMITU - I/O Limit Upper (0x032)

Bit Field	Field Name	Type	Default Value	Description
15:0	IOLIMITU	RW	0x0	**Prefetchable IO Limit Upper.** This field specifies the upper 16-bits of IOLIMIT. When the IOCAP field in the IOBASE register is cleared, this field becomes read-only with a value of zero.

CAPPTR - Capabilities Pointer (0x034)

Bit Field	Field Name	Type	Default Value	Description
7:0	CAPPTR	RO	0x40	**Capabilities Pointer.** This field specifies a pointer to the head of the capabilities structure.

EROMBASE - Expansion ROM Base Address (0x038)

Bit Field	Field Name	Type	Default Value	Description
31:0	EROMBASE	RO	0x0	**Expansion ROM Base Address.** The bridge does not implement an expansion ROM. Thus, this field is hardwired to zero.

INTRLINE - Interrupt Line (0x03C)

Bit Field	Field Name	Type	Default Value	Description
7:0	INTRLINE	RW	0x0	**Interrupt Line.** This register communicates interrupt line routing information. Values in this register are programmed by system software and are system architecture specific. The bridge does not use the value in this register. Legacy interrupts may be implemented by downstream ports.

INTRPIN - Interrupt PIN (0x03D)

Bit Field	Field Name	Type	Default Value	Description
7:0	INTRPIN	RWL	0x0	**Interrupt Pin.** Interrupt pin or legacy interrupt messages are not used by the bridge by default. However, they can be used for hot-plug by the downstream ports. This field should only be configured with values of 0x0 through 0x4.

Notes

BCTRL - Bridge Control (0x03E)

Bit Field	Field Name	Type	Default Value	Description
0	PERRE	RW	0x0	**Parity Error Response Enable.** This bit controls the bridges response to poisoned TLPs on the secondary interface. 0x0 - (ignore) Ignore poisoned TLPs (i.e., parity errors) on the secondary interface. 0x1 - (report) Enable poisoned TLP (i.e., parity error) detection and reporting on the secondary interface of the bridge.
1	SERRE	RW	0x0	**System Error Enable.** This bit controls forwarding of ERR_COR, ERR_NONFATAL, ERR_FATAL from the secondary interface of the bridge to the primary interface. Note that error reporting must be enabled in the Command register or PCI Express Capability structure, Device Control register for errors to be reported on the primary interface. 0x0 - (ignore) Do not forward errors from the secondary to the primary interface. 0x1 - (report) Enable forwarding of errors from secondary to the primary interface.
2	ISAEN	RO	0x0	**ISA Enable**. The PES12N3 does not support this feature.
3	VGAEN (for revisions YA and YB)	RW	0x0	**VGA Enable.** (Silicon revisions YA and YB only.) Controls the routing of processor-initiated transactions targeting VGA. 0 - (block) Do not forward VGA compatible addresses from the primary interface to the secondary interface 1 - (forward) Forward VGA compatible addresses from the primary to the secondary interface.
	VGAEN (for revision YC)	RO	0x0	Silicon revision YC does not support the VGA Enable feature.
5:4	Reserved	RO	0x0	
6	SRESET	RW	0x0	**Secondary Bus Reset.** Setting this bit triggers a hot reset down the secondary interface of the bridge.
15:7	Reserved	RO	0x0	

PCI Express Capability Structure

PCIECAP - PCI Express Capability (0x040)

Bit Field	Field Name	Type	Default Value	Description
7:0	CAPID	RO	0x10	**Capability ID**. The value of 0x10 identifies this capability as a PCI Express capability structure.

Notes

Bit Field	Field Name	Type	Default Value	Description
15:8	NXTPTR	RO	0x70	**Next Pointer**. This field contains a pointer to the next capability structure.
19:16	VER	RO	0x1	**PCI Express Capability Version.** This field indicates the PCI-SIG defined PCI Express capability structure version number.
23:20	TYPE	RO	-	Port Type.
24	SLOT	RWL	0x0	**Slot Implemented.** This bit is set when the PCI Express link associated with this Port is connected to a slot.
29:25	IMN	RO	0x0	**Interrupt Message Number.** The function is allocated only one (downstream ports) MSI or none (upstream ports). Therefore, this field is set to zero.
31:30	Reserved	RO	0x0	Reserved field.

PCIEDCAP - PCI Express Device Capabilities (0x044)

Bit Field	Field Name	Type	Default Value	Description
2:0	MPAYLOAD	RWL	0x4	**Maximum Payload Size Supported**. This field indicates the maximum payload size that the device can support for TLPs. The default value corresponds to 2048 bytes.
4:3	PFS	RO	0x0	**Phantom Functions Supported**. This field indicates the support for unclaimed function number to extend the number of outstanding transactions allowed by logically combining unclaimed function numbers. The value is hardwired to 0x0 to indicate that no function number bits are used for phantom functions.
5	ETAG	RO	0x1	**Extended Tag Field Support.** This field indicates the maximum supported size of the Tag field as a requester.
8:6	E0AL	RO	0x0	**Endpoint L0s Acceptable Latency.** This field indicates the acceptable total latency that an endpoint can withstand due to transition from the L0s state to the L0 state. The value is hardwired to 0x0 as this field does not apply to a switch.
11:9	E1AL	RO	0x0	**Endpoint L1 Acceptable Latency.** This field indicates the acceptable total latency that an endpoint can withstand due to transition from the L1 state to the L0 state. The value is hardwired to 0x0 as this field does not apply to a switch.
12	ABP	RWL	0x0	**Attention Button Present.** When set, this bit indicates that an Attention Button is implemented on the card/module. This bit should not be set on downstream ports.
13	AIP	RWL	0x0	**Attention Indicator Present.** When set, this bit indicates that an Attention Indicator is implemented on the card/module. This bit should not be set on downstream ports.

Notes

Bit Field	Field Name	Type	Default Value	Description
14	PIP	RWL	0x0	**Power Indicator Present.** When set, this bit indicates that a Power Indicator is implemented on the card/module. This bit should not be set on downstream ports.
25:18	CSPLV	RO	0x0	**Captured Slot Power Limit Value.** *(Upstream Port A only, hardwired to zero in downstream ports)* Captured Slot Power Limit Value (upstream Ports only, hardwired to zero in downstream ports) – In combination with the Slot Power Limit Scale value, specifies the upper limit on power supplied by slot.Power limit (in Watts) calculated by multiplying the value in this field by the value in the Slot Power Limit Scale field. This value is set by the Set_Slot_Power_Limit Message
27:26	CSPLS	RO	0x0	**Captured Slot Power Limit Scale.** *(Upstream Port A only, hardwired to zero in downstream ports)* This field specifies the scale used for the Slot Power Limit Value and is set via a Set_Slot_Power_Limit message. 0 - (v1) 1.0x 1 - (v1p1) 0.1x 2 - (v0p01) 0.01x 3 - (v0p001x) 0.001x
31:28	Reserved	RO	0x0	Reserved.

PCIEDCTL - PCI Express Device Control (0x048)

Bit Field	Field Name	Type	Default Value	Description
0	CEREN	RW	0x0	**Correctable Error Reporting Enable**. This bit controls reporting of correctable errors.
1	NFEREN	RW	0x0	**Non-Fatal Error Reporting Enable**. This bit controls reporting of non-fatal errors.
2	FEREN	RW	0x0	**Fatal Error Reporting Enable**. This bit controls reporting of fatal errors.
3	URREN	RW	0x0	**Unsupported Request Reporting Enable**. This bit controls reporting of unsupported requests.
4	ERO	RO	0x0	**Enable Relaxed Ordering**. When set, this bit enables relaxed ordering. The switch never sets the relaxed ordering bit in transactions it initiates as a requester.
7:5	MPS	RW	0x0	**Max Payload Size.** This field sets maximum TLP payload size for the device. 0x0 - (s128) 128 bytes max payload size 0x1 - (s256) 256 bytes max payload size 0x2 - (s512) 512 bytes max payload size 0x3 - (s1024) 1024 bytes max payload size 0x4 - (s2048) 2048 bytes max payload size 0x5 - reserved (treated as 128 bytes) 0x6 - reserved (treated as 128 bytes) 0x7 - reserved (treated as 128 bytes)

Notes

Bit Field	Field Name	Type	Default Value	Description
8	ETFEN	RW	0x0	**Extended Tag Field Enable**. Since the transparent bridge never generates a transaction that requires a completion, this bit has no functional effect on the device during normal operation.
9	PFEN	RO	0x0	**Phantom Function Enable**. The bridge does not support phantom function numbers. Therefore, this field is hard-wired to zero.
10	AUXPMEN	RO	0x0	**Auxiliary Power PM Enable**. The device does not implement this capability.
11	ENS	RO	0x0	**Enable No Snoop.** The transparent bridge does not generate transactions with the No Snoop bit set and passes transactions through the bridge with the No Snoop bit unmodified. Therefore, this field has no functional effect on the behavior of the transparent bridge.
14:12	MRRS	RO	0x0	**Maximum Read Request Size.** The transparent bridge does not generate transactions larger than 128 bytes and passes transactions through the bridge with the size unmodified. Therefore, this field has no functional effect on the behavior of the transparent bridge.
15	Reserved	RO	0x0	Reserved field.

PCIEDSTS - PCI Express Device Status (0x04A)

Bit Field	Field Name	Type	Default Value	Description
0	CED	RW1C	0x0	**Correctable Error Detected**. This bit indicates the status of correctable errors. Errors are logged in this register regardless of whether error reporting is enabled or not.
1	NFED	RW1C	0x0	**Non-Fatal Error Detected**. This bit indicates the status of correctable errors. Errors are logged in this register regardless of whether error reporting is enabled or not.
2	FED	RW1C	0x0	**Fatal Error Detected**. This bit indicates the status of Fatal errors. Errors are logged in this registers regardless of whether error reporting is enabled or not.
3	URD	RW1C	0x0	**Unsupported Request Detected**. This bit indicates the device received an Unsupported Request. Errors are logged in this register regardless of whether error reporting is enabled or not.
4	AUXPD	RO	0x0	**Aux Power Detected**. Devices that require AUX power, set this bit when AUX power is detected.This device does not require AUX power, hence the value is hardwired to zero.
5	TP	RO	0x0	**Transactions Pending.** The transparent bridge does not issue Non-Posted Requests on its own behalf. Therefore, this field is hardwired to zero.
15:6	Reserved	RO	0x0	Reserved field.

Notes

PCIELCAP - PCI Express Link Capabilities (0x04C)

Bit Field	Field Name	Type	Default Value	Description
3:0	MAXLNKSPD	RO	0x1	**Maximum Link Speed**. This field is hardwired to 0x1 to indicate 2.5 Gbps.
9:4	MAXLNKWDTH	RWL	0x8	**Maximum Link Width**. This field indicates the maximum link width of the given PCI Express link. This field may be overridden to allow the link width to be forced to a smaller value. Setting this field to a invalid or reserved value results in x1 being used. 1 - (x1) x1 link width 2 - (x2) x2 link width 4 - (x4) x4 link width 8 - (x8) x8 link width others- reserved
11:10	ASPMS	RO	0x3	**Active State Power Management (ASPM) Support**. This field is hardwired to 0x3 to indicate L0s and L1 Support.
14:12	L0SEL	RWL	see text	**L0s Exit Latency**. This field indicates the L0s exit latency for the given PCI Express link. This field depends on whether a common or separate reference clock is used. When separate clocks are used, 1 µs to 2 µs is reported with a read-only value of 0x5. When a common clock is used, 256 ns to 512 ns is reported with a read-only value of 0x3
17:15	L1EL	RWL	0x2	**L1 Exit Latency**. This field indicates the L1 exit latency for the given PCI Express link. Transitioning from L1 to L0 always requires 2.3 µs. Therefore, a value 2 µs to less than 4 µs is reported with a default value of 0x2.
23:18	Reserved	RO	0x0	Reserved field.
31:24	PORTNUM	RO	port A - 0x0 Port B - 0x1 Port C - 0x2	**Port Number**. This field indicates the PCI express port number.

PCIELCTL - PCI Express Link Control (0x050)

Bit Field	Field Name	Type	Default Value	Description
1:0	ASPM	RW	0x0	**Active State Power Management (ASPM) Control**. This field controls the level of ASPM supported by the link. The initial value corresponds to disabled. The value contained in Serial EEPROM may override this default value 0x0 - (disabled) disabled 0x1 - (l0s) L0s enable entry 0x2 - (l1) L1 enable entry 0x3 - (l0sl1) L0s and L1 enable entry
2	Reserved	RO	0x0	Reserved field.
3	RCB	RO	0x0	**Read Completion Boundary**. This field is not applicable and is hardwired to zero.

Notes

Bit Field	Field Name	Type	Default Value	Description
4	LDIS	RW	0x0	**Link Disable.** When set, this bit disables the link. This field is hardwired to 0x0 for the upstream port.
5	LRET	RW	0x0	**Link Retrain**. Writing a one to this field initiates Link retraining by directing the Physical Layer LTSSM to the Recovery state. This field always returns zero when read. For compliance with the PCIe specification, this bit has no effect on the upstream port when the REGUNLOCK bit is cleared in the PA_SWCTL register. When the REGUN-LOCK bit is set, writing a one to the LRET bit initiates link retraining on the upstream port.
6	CCLK	RW	0x0	**Common Clock Configuration**. When set, this bit indicates that this component and the component at the opposite end of the link are operating with a distributed common reference clock.
7	ESYNC	RW	0x0	**Extended Sync**. When set this bit forces transmission of 4096 FTS ordered sets in the L0s state followed by a single SKP ordered set.
15:8	Reserved	RO	0x0	Reserved field.

PCIELSTS - PCI Express Link Status (0x052)

Bit Field	Field Name	Type	Default Value	Description
3:0	LS	RO	0x1	**Link Speed**. This field is hardwired to 2.5 Gbps.
9:4	LW	ROS	HWINIT	**Link Width**. This field indicates the negotiated width of the link.
10	TERR	ROS	0x0	**Training Error.** When set, this bit indicates that a link training error has occurred.
11	LTRAIN	ROS	0x0	**Link Training**. When set, this bit indicates that link training is in progress.
12	SCLK	RWL	HWINIT	**Slot Clock Configuration**. When set, this bit indicates that the component uses the same physical reference clock that the platform provides. The initial value of this field is the state of the CCLKUS signal for port A and the CCLKDS signal for downstream ports B and C. The value contained in Serial EEPROM may override these default values.
15:13	Reserved	RO	0x0	Reserved field.

Notes

PCIESCAP - PCI Express Slot Capabilities (0x054)

Bit Field	Field Name	Type	Default Value	Description
0	ABP	RWL	0x0	**Attention Button Present**. This bit is set when the Attention Button is implemented for the port. This bit is read-only and has a value of zero when the SLOT bit in the PCIECAP register is cleared.
1	PCP	RWL	0x0	**Power Control Present**. This bit is set when a Power Controller is implemented for the port. This bit is read-only and has a value of zero when the SLOT bit in the PCIECAP register is cleared.
2	MRLP	RWL	0x0	**MRL Sensor Present**. This bit is set when an MRL Sensor is implemented for the port. This bit is read-only and has a value of zero when the SLOT bit in the PCIECAP register is cleared.
3	ATTIP	RWL	0x0	**Attention Indicator Present**. This bit is set when an Attention Indicator is implemented for the port. This bit is read-only and has a value of zero when the SLOT bit in the PCIECAP register is cleared.
4	PWRIP	RWL	0x0	**Power Indicator Present**. This bit is set when an Power Indicator is implemented for the port. This bit is read-only and has a value of zero when the SLOT bit in the PCIECAP register is cleared.
5	HPS	RWL	0x0	**Hot Plug Surprise**. When set, this bit indicates that a device present in the slot may be removed from the system without notice. This bit is read-only and has a value of zero when the SLOT bit in the PCIECAP register is cleared.
6	HPC	RWL	0x0	**Hot Plug Capable**. This bit is set if the slot corresponding to the port is capable of supporting hot-plug operations. This bit is read-only and has a value of zero when the SLOT bit in the PCIECAP register is cleared.
14:7	SPLV	RWL	0x0	**Slot Power Limit Value**. In combination with the Slot Power Limit Scale, this field specifies the upper limit on power supplied by the slot. A Set_Slot_Power_Limit message is generated using this field whenever this register is written or when the link transitions from a non DL_Up status to a DL_Up status. This bit is read-only and has a value of zero when the SLOT bit in the PCIECAP register is cleared.
16:15	SPLS	RWL	0x0	**Slot Power Limit Scale**. This field specifies the scale used for the Slot Power Limit Value (SPLV). 0x0 - (x1) 1.0x 0x1 - (xp1) 0.1x 0x2 - (xp01) 0.01x 0x3 - (xp001) 0.001x A Set_Slot_Power_Limit message is generated using this field whenever this register is written or when the link transitions from a non DL_Up status to a DL_Up status. This bit is read-only and has a value of zero when the SLOT bit in the PCIECAP register is cleared.

Notes

Bit Field	Field Name	Type	Default Value	Description
17	EIP	RWL	0x0	**Electromechanical Interlock Present**. This bit is set if an electromechanical interlock is implemented on the chassis for this slot. This bit is unused in PCIe 1.0a mode (i.e., HPMODE bit cleared) and should be set to zero. This bit is read-only and has a value of zero when the SLOT bit in the PCIECAP register is cleared or the HPMODE bit in the PA_SWCTL register is cleared.
18	NCCS	RO	0x0	**No Command Completed Support**. Software notification is always generated when an issued command is completed by the hot-plug controller. Therefore, this field is hardwired to zero. This bit is unused in PCIe 1.0a mode (i.e., HPMODE bit cleared).
31:19	PSLOTNUM	RWL	0x0	**Physical Slot Number**. This field indicates the physical slot number attached to this port. For devices interconnected on the system board, this field should be initialized to zero. This bit is read-only and has a value of zero when the SLOT bit in the PCIECAP register is cleared.

PCIESCTL - PCI Express Slot Control (0x058)

Bit Field	Field Name	Type	Default Value	Description
0	ABPE	RW	0x0	**Attention Button Pressed Enable**. This bit when set enables generation of a Hot-Plug interrupt or wake-up event on an attention button pressed event. This bit is read-only and has a value of zero when the corresponding capability is not enabled in the PCIESCAP register.
1	PFDE	RW	0x0	**Power Fault Detected Enable**. This bit when set enables the generation of a Hot-Plug interrupt or wake-up event on a power fault event. This bit is read-only and has a value of zero when the corresponding capability is not enabled in the PCIESCAP register.
2	MRLSCE	RW	0x0	**MRL Sensor Change Enable**. This bit when set enables the generation of a Hot-Plug interrupt or wake-up event on a MRL sensor change event. This bit is read-only and has a value of zero when the corresponding capability is not enabled in the PCIESCAP register.
3	PDCE	RW	0x0	**Presence Detected Changed Enable**. This bit when set enables the generation of a Hot-Plug interrupt or wake-up event on a presence detect change event. This bit is read-only and has a value of zero when the corresponding capability is not enabled in the PCIESCAP register.

Notes

Bit Field	Field Name	Type	Default Value	Description
4	CCIE	RW	0x0	**Command Complete Interrupt Enable**. This bit when set enables the generation of a Hot-Plug interrupt when a command is completed by the Hot-Plug Controller. This bit is read-only and has a value of zero when the corresponding capability is not enabled in the PCIESCAP register.
5	HPIE	RW	0x0	**Hot Plug Interrupt Enable**. This bit when set enables generation of a Hot-Plug interrupt on enabled Hot-Plug events. This bit is read-only and has a value of zero when the corresponding capability is not enabled in the PCIESCAP register.
7:6	AIC	RW	0x3	**Attention Indicator Control**. When read, this register returns the current state of the Attention Indicator. Writing to this register sets the indicator and causes the port to send the appropriate ATTENTION_INDICATOR_* message. This bit is read-only and has a value of zero when the corresponding capability is not enabled in the PCIESCAP register. This field is always zero if the ATTIP bit is cleared in the PCIESCAP register. 0x0 -(reserved) Reserved 0x1 -(on) On 0x2 -(blink) Blink 0x3 -(off) Off
9:8	PIC	RW	0x1	**Power Indicator Control**. When read, this register returns the current state of the Power Indicator. Writing to this register sets the indicator and causes the port to send the appropriate POWER_INDICATOR_* message. This bit is read-only and has a value of zero when the corresponding capability is not enabled in the PCIESCAP register. This field is always zero if the PWRIP bit is cleared in the PCIESCAP register. 0x0 -(reserved) Reserved 0x1 -(on) On 0x2 -(blink) Blink 0x3 -(off) Off This field has no effect on the upstream port.
10	PCC	RW	0x0	**Power Controller Control**. When read, this register returns the current state of the power applied to the slot. Writing to this register sets the power state of the slot. This bit is read-only and has a value of zero when the corresponding capability is not enabled in the PCIESCAP register. 0x0 -(on) Power on 0x1 -(off) Power off

Notes

Bit Field	Field Name	Type	Default Value	Description
11	EIC	RW	0x0	**Electromechanical Interlock Control**. This field always returns a value of zero when read. If an electromechanical interlock is implemented, a write of a one to this field causes the state of the interlock to toggle and a write of a zero has no effect. This bit is read-only and has a value of zero when the corresponding capability is not enabled in the PCIESCAP register or when the HPMODE bit in the PA_SWCTL register is cleared. This bit is unused and has no effect in PCIe 1.0a mode (i.e., HPMODE bit cleared).
15:12	Reserved	RO	0x0	Reserved field.

PCIESSTS - PCI Express Slot Status (0x05A)

Bit Field	Field Name	Type	Default Value	Description
0	ABP	RW1C	0x0	**Attention Button Pressed**. Set when the attention button is pressed.
1	PFD	RW1C	0x0	**Power Fault Detected**. Set when the Power Controller detects a power fault.
2	MRLSC	RW1C	0x0	**MRL Sensor Changed**. Set when an MRL Sensor state change is detected.
3	PSD	RW1C	0x0	**Presence Detected Changed**. Set when a Presence Detected change is detected.
4	CC	RW1C	0x0	**Command Completed**. This bit is set when the Hot-Plug Controller completes an issued command.
5	MRLSS	ROS	0x0	**MRL Sensor State**. This field enclosed the current state of the MRL sensor. 0x0 -(closed) MRL closed 0x1 -(open) MRL open
6	PDS	ROS	0x1	**Presence Detect State**. This bit indicates the presence of a card in the slot corresponding to the port and reflects the state of the Presence Detect status. 0x0 -(empty) Slot empty 0x1 -(present) Card present
7	EIS	ROS	0x0	**Electromechanical Interlock Status**. When an electromechanical interlock is implemented, this bit indicates the current status of the interlock. 0x0 - (disengaged) Electromechanical interlock disengaged 0x1 - (engaged) Electromechanical interlock engaged This bit is unused and is always zero in PCIe 1.0a mode (i.e., HPMODE bit cleared).
15:8	Reserved	RO	0x0	Reserved field.

Notes

Power Management Capability Structure

PMCAP - PCI Power Management Capabilities (0x070)

Bit Field	Field Name	Type	Default Value	Description
7:0	CAPID	RO	0x1	**Capability ID**. The value of 0x1 identifies this capability as a PCI power management capability structure.
15:8	NXTPTR	RO	0x7C for ports B and C 0x0 for port A	**Next Pointer**. This field contains a pointer to the next capability structure. For port A the value of this field is 0x0 indicating that it is the last capability. For ports B and C, this field is 0x7C and points to the MSI capability structure.
18:16	VER	RO	0x2	**Power Management Capability Version.** This field indicates compliance with version two of the specification.
19	PMECLK	RO	0x0	**PME Clock.** Does not apply to PCI Express.
20	Reserved	RO	0x0	
21	DEVSP	RWL	0x0	**Device Specific Initialization.** The value of zero indicates that no device specific initialization is required.
24:22	AUXI	RO	0x0	**AUX Current.** not used
25	D1	RO	0x0	**D1 Support.** This field indicates that the PES12N3 does not support D1.
26	D2	RO	0x0	**D2 Support.** This field indicates that the PES12N3 does not support D2.
31:27	PME	RO	0b11001	**PME Support.** This field indicates the power states in which the port may generate a PME. Bits 27, 30 and 31 are set to indicate that the bridge will forward PME messages. The switch does not forward PME messages in $D3_{cold}$. This functionality may be supported in the system by routing WAKE# around the switch.

PMCSR - PCI Power Management Control and Status (0x074)

Bit Field	Field Name	Type	Default Value	Description
1:0	PSTATE	RW	0x0	**Power State**. This field is used to determine the current power state and to set a new power state. 0x0 - (d0) D0 state 0x1 - (d1) D1 state (not supported by PES12N3 and reserved) 0x2- (d2) D2 state (not supported by PES12N3 and reserved) 0x3 - (d3) $D3_{hot}$ state
7:2	Reserved	RO	0x0	

Notes

Bit Field	Field Name	Type	Default Value	Description
8	PMEE	RW	0x0 **Sticky** for ports B & C RO 0x0 for port A	**PME Enable.** When this bit is set, PME message generation is enabled for the port. If a hot plug wake-up event is desired when exiting the $D3_{cold}$ state, then this bit should be set during serial EEPROM initialization. Port A never generates PM_PME messages. Therefore, this bit is hardwired to zero for that port. A hot reset does not result in modification of this field.
12:9	DSEL	RO	0x0	**Data Select.** The optional data register is not implemented.
14:13	DSCALE	RO	0x0	**Data Scale.** The optional data register is not implemented.
15	PMES	RW1C	0x0 **Sticky** for ports B & C RO 0x0 for port A	**PME Status.** This bit is set if a PME is generated by the port even if the PMEE bit is cleared. This bit is not set when the bridge is propagating a PME message but the port is not itself generating a PME.
21:16	Reserved	RO	0x0	
22	B2B3	RO	0x0	**B2/B3 Support.** Does not apply to PCI Express.
23	BPCCE	RO	0x0	**Bus Power/Clock Control Enable.** Does not apply to PCI Express.
31:24	DATA	RO	0x0	**Data.** This optional field is not implemented.

PMPC - PCI Power Management Proprietary Control (0x078)

Mode1

Bit Field	Field Name	Type	Default Value	Description
15:0	L1ET	RW	0x3E8	**L1 Entry Timer.** This field specifies the L1 entry timer value for the related port transmitter. If all L1 entry conditions are met for the specified amount of time, then the transmitter port enters L1. This field is used by the upstream ports and is ignored by downstream ports. The timer value is specified in the number 1 µS clock cycles. The default value corresponds to 1 mS
27:16	L0ET	RW	0x6D6	**L0s Entry Timer**. This field specifies the L0s entry time value for the related port transmitter. If all L0s entry conditions are met for the specified amount of time, then the transmitter port enters L0s. The timer value is specified in the number 4ns clock cycles. The default value corresponds to the PCI Express value of 7µS.
30:28	Reserved	RO	0x0	

Notes

Bit Field	Field Name	Type	Default Value	Description
31	MCS	RW	0x0	**Mode Configuration Switch.** When this bit is set to zero, the PMPC register is configured as Mode1, shown in this table. When this bit is set to one, the register is configured as Mode2, shown in the Mode2 table.

Mode2

Bit Field	Field Name	Type	Default Value	Description
5:0	PMCS	RO	0x3F	**Values for PM current state.** This field denotes the current value of the internal PM State Machine. Note that the SMBus is generally used to read this field rather than Configuration Reads. L0 = 6'd0 L0s_entry = 6'd1 L0s = 6'd2 L1pcipm_0 = 6'd3 L1pcipm_1 = 6'd4 L1pcipm_2 = 6'd5 L1pcipm_3 = 6'd6 L1pcipm_4 = 6'd7 L1pcipm_5 = 6'd8 L1pcipm_6 = 6'd9 L1pcipm_7 = 6'd10 L1comply_0 = 6'd11 L1comply_1 = 6'd12 L1comply_2 = 6'd13 L1accept_0 = 6'd14 L1accept_1 = 6'd15 L1accept_2 = 6'd16 L1reject_0 = 6'd17 L1timed_0 = 6'd18 L1timed_1 = 6'd19 L1timed_2 = 6'd20 L1timed_3 = 6'd21 L1timed_nak = 6'd22 L1_entry_1 = 6'd23 L1_entry_2 = 6'd24 L1_entry_3 = 6'd25 L1 = 6'd26 L23ready__entry_1 = 6'd29 L23ready__entry_2 = 6'd30 L23ready__entry_3 = 6'd31 L23ready__entry_4 = 6'd32 L23ready__entry_5 = 6'd33 L23ready__entry_6 = 6'd34 L23ready = 6'd35 LDown = 6'd63
30:6	Reserved	RO	0x0	**Reserved**

Notes

Bit Field	Field Name	Type	Default Value	Description
31	MCS	RW	0x0	**Mode Configuration Switch.** When this bit is set to zero, the PMPC register is configured as Mode1, shown in the Mode1 table. When this bit is set to one, the register is configured as Mode2, shown in this table.

Message Signaled Interrupt Capability Structure

MSICAP - Message Signaled Interrupt Capability and Control (0x07C)

Bit Field	Field Name	Type	Default Value	Description
7:0	CAPID	RO	0x5	**Capability ID**. The value of 0x5 identifies this capability as a MSI capability structure.
15:8	NXTPTR	RO	0x0	**Next Pointer**. This field contains a pointer to the next capability structure. This field is set to 0x0 indicating that it is the last capability.
16	EN	RW	0x0	**Enable.** This bit enables MSI. 0x0 - (disable) disabled 0x1 - (enable) enabled
19:17	MMC	RO	0x0	**Multiple Message Capable.** This field contains the number of requested messages. The transparent bridge requests one message.
22:20	MME	RO	0x0	**Multiple Message Enable.** Hardwired to one message.
23	A64	RO	0x1	**64-bit Address Capable.** The transparent bridge is capable of generating messages using a 64-bit address.
31:24	Reserved	RO	0x0	Reserved.

MSIADDR - Message Signaled Interrupt Address (0x080)

Bit Field	Field Name	Type	Default Value	Description
1:0	Reserved	RO	0x0	Reserved.
31:2	ADDR	RW	0x0	**Message Address**. This field specifies the lower portion of the DWORD address of the MSI memory write transaction.

Notes

MSIUADDR - Message Signaled Interrupt Upper Address (0x084)

Bit Field	Field Name	Type	Default Value	Description
31:0	UADDR	RW	0x0	**Upper Message Address**. This field specifies the upper portion of the DWORD address of the MSI memory write transaction. If the contents of this field are non-zero, then 64-bit address is used in the MSI memory write transaction. If the contents of this field are zero, then the 32-bit address specified in the MSIADDR field is used.

MSIMDATA - Message Signaled Interrupt Message Data (0x088)

Bit Field	Field Name	Type	Default Value	Description
15:0	MDATA	RW	0x0	**Message Data**. This field contains the lower 16-bits of data that are written when a MSI is signalled.
31:16	Reserved	RO	0x0	Reserved.

Switch Control and Status Registers

SWSTS Switch Status (0x0A0)

Bit Field	Field Name	Type	Default Value	Description
3:0	SWMODE	RO	HWINIT	**Switch Mode.** The value of this field encodes the switch mode sampled on the Switch Mode (SWMODE[3:0]) signals when exiting reset. 0x0 - (transparent) Transparent mode 0x1 - (transaprentinit) Transparent mode with serial EEPROM initialization 0x2 through 0x7 - Reserved 0x8 - (loopback10) 10-bit loopback test mode 0x9 - Reserved 0xA - (intprbs) Internal pseudo random bit stream self-test test mode 0xB - (extprbs) External pseudo random bit stream self-test test mode 0xC - Reserved 0xD - (sbroadcast) SerDes broadcast test mode 0xE - 0xF Reserved
4	CCLKDS	RO	HWINIT	**Common Clock Downstream.** This bit reflects the value of the CCLKDS signal sampled during the fundamental reset.
5	CCLKUS	RO	HWINIT	**Common Clock Upstream.** This bit reflects the value of the CCLKUS signal sampled during the fundamental reset.
6	MSMBSMODE	RO	HWINIT	**Master SMBus Slow Mode.** This bit reflects the value of the MSMBSMODE signal sampled during the fundamental reset.

Notes

Bit Field	Field Name	Type	Default Value	Description
7	PALREV	RO	HWINIT	**PCI Express Port A Lane Reverse.** This bit reflects the value of the PALREV signal sampled during the fundamental reset.
8	PBLREV	RO	HWINIT	**PCI Express Port B Lane Reverse.** This bit reflects the value of the PBLREV signal sampled during the fundamental reset.
9	PCLREV	RO	HWINIT	**PCI Express Port C Lane Reverse.** This bit reflects the value of the PCLREV signal sampled during the fundamental reset.
10	REFCLKM	RO	HWINIT	**PCI Express Reference Clock Mode Select.** This bit reflects the value of the REFCLKM signal sampled during the fundamental reset.
11	RSTHALT	RO	HWINIT	**Reset Halt.** This bit reflects the value of the RSTHALT signal sampled during the fundamental reset.
12	TSTRSVD	RO	HWINIT	**Reserved.** Reserved for future test mode. Must be tied to ground.
16:13	MARKER	RW	0x0 **Sticky**	**Marker.** This field is preserved across a hot reset and is available for general software use. A hot reset does not result in modification of this field.
18:17	LOCKMODE	RO	0x0	**Lock Mode.** This field reflects the current locked status of the switch. 0x0 - (unlocked) switch is unlocked 0x1 - (lockedab) port A is locked with downstream port B 0x2 - (lockedac) port A is locked with downstream port C 0x3 - reserved
19	LOCKDROP	RW1C	0x0	**Locked Dropped.** When the switch is locked and the upstream port may become deadlocked due to a TLP being received which cannot be forwarded due to the lock, then the TLP is dropped and this bit is set. This bit is also set and the transaction is dropped if the switch is locked and a Memory Read Request - Locked (MRdLK) transaction is received from a requester other than the one which had locked the switch.
20	INTA	RO	0x0	**INTA Aggregated State.** Aggregated switch state for INTA. 0x0 - (negated) INTA negated 0x1 - (asserted) INTA asserted
21	INTB	RO	0x0	**INTB Aggregated State.** Aggregated switch state for INTB. 0x0 - (negated) INTB negated 0x1 - (asserted) INTB asserted
22	INTC	RO	0x0	**INTC Aggregated State.** Aggregated switch state for INTC. 0x0 - (negated) INTC negated 0x1 - (asserted) INTC asserted

Notes

Bit Field	Field Name	Type	Default Value	Description
23	INTD	RO	0x0	**INTD Aggregated State.** Aggregated switch state for INTD. 0x0 - (negated) INTD negated 0x1 - (asserted) INTD asserted
24	LOCKDIS	RW1C	0x0	**Lock Discard.** When the upstream port is locked with a downstream port and a TLP is received by the upstream port that is destined to the unlocked downstream port, then the TLP is dropped, this bit is set, and if error reporting is enabled an ERR_NON_FATAL message is sent to the root.
25	PMELOCK	RW1C	0x0	**PME Lock Error.** This bit is set when a PME_Turn_Off message is received by a locked downstream PCI-PCI bridge (i.e., that associated with port B or C). When this occurs and error reporting is enabled, an ERR_NON_FATAL message is sent to the root.
29:26	Reserved	RO	0x0	
31:30	MAXLNKWDTH	RO	0x8	**Maximum Link Width.**

SWCTL - Switch Control (0x0A4)

Bit Field	Field Name	Type	Default Value	Description
0	RST	RW	0x0	**Reset.** Writing a one to this bit initiates a fundamental reset. Writing a zero has no effect. This field always returns a value of zero when read.
1	RSTHALT	RW	HWINIT	**Reset Halt.** When this bit is set, all of the switch logic except the SMBus interface remains in a reset state. In this state, registers in the device may be initialized by the slave SMBus interface. When this bit is cleared, normal operation ensues. Setting or clearing this bit has no effect following a reset operation. This bit may be set by asserting the RSTHALT signal during a reset operation or through initialization by the serial EEPROM.
2	REGUNLOCK	RW	0x0	**Register Unlock.** When this bit is set, the contents of registers and fields of type Read and Write when Unlocked (RWL) are modified when written to. When this bit is cleared, all registers and fields denoted as RWL become read-only. While the initial value of this field is cleared, it is set during a reset operation, thus allowing serial EEPROM initialization to modify the contents of RWL fields.
3	DRO	RW	0x0	**Disable Relaxed Ordering.** The switch implements relaxed ordering for TLPs with the relaxed ordering bit set. When the DRO bit is set, the switch strongly orders all transactions regardless of the state of the relaxed ordering bit in TLPs.

Notes

Bit Field	Field Name	Type	Default Value	Description
4	PALREV	RW	HWINIT	**Port A Lane Reversal.** When this bit is set, the lanes associated with port A are reversed. The initial value of this register corresponds to the state of the PALREV pin. However, this value may be overridden by the serial EEPROM, SMBus, or PCIe configuration write. Modifications to this bit take effect the next time link training occurs.
5	PBLREV	RW	HWINIT	**Port B Lane Reversal.** When this bit is set, the lanes associated with port B are reversed. The initial value of this register corresponds to the state of the PBLREV pin. However, this value may be overridden by the serial EEPROM, SMBus, or PCIe configuration write. Modifications to this bit take effect the next time link training occurs.
6	PCLREV	RW	HWINIT	**Port C Lane Reversal.** When this bit is set, the lanes associated with port C are reversed. The initial value of this register corresponds to the state of the PCLREV pin. However, this value may be overridden by the serial EEPROM, SMBus, or PCIe configuration write. Modifications to this bit take effect the next time link training occurs.
7	HPMODE	RW	0x0 **Sticky**	**Hot Plug Mode.** This bit controls the hot-plug compatibility mode. This field may only be modified via the serial EEPROM or during initialization via the SMBus when the RSTHALT bit is set in the PA_SWCTL register. Modifying this bit in a running system produces undefined results. 0x0 - (hp1p0a) PCI-Express 1.0a hot-plug 0x1 - (hp1p1) PCI-Express 1.1 hot-plug
8	Reserved	RW	0x0	Reserved
9	RSTSAFEM	RW	0x0	Reserved
10	PHYSAFEM	RW	0x0	Reserved
11	PHYNCSAFEM	RW	0x0	Reserved
12	DLSAFEM	RW	0x0	Reserved

Notes

Bit Field	Field Name	Type	Default Value	Description
17:13	TSTCLK0SEL	RW	0x0	**Test Clock 0 Output Select.** This field selects the clock output driven on the TSTCLK0 pin (GPIO[6] alternate function). 0x0 - (lane0) Lane 0 recovered clock divided by 40.[1] 0x1 - (lane1) Lane 1 recovered clock divided by 40. 0x2 - (lane2) Lane 2 recovered clock divided by 40. 0x3 - (lane3) Lane 3 recovered clock divided by 40. 0x4 - 0x7 Reserved. 0x8 - (lane8) Lane 8 recovered clock divided by 40. 0x9 - (lane9) Lane 9 recovered clock divided by 40. 0xA - (lane10) Lane 10 recovered clock divided by 40. 0xB - (lane11) Lane 11 recovered clock divided by 40. 0xC - 0xF Reserved. 0x10 - (lane16) Lane 16 recovered clock divided by 40. 0x11 - (lane17) Lane 17 recovered clock divided by 40. 0x12 - (lane18) Lane 18 recovered clock divided by 40. 0x13 - (lane19) Lane 19 recovered clock divided by 40. 0x14 - 0x17 Reserved. 0x18 - (serdes0) SerDes 0 250 MHz clock output divided by four. This SerDes is used to generate the core clock. 0x19 - Reserved. 0x1A - (serdes2) SerDes 2 250 MHz clock output divided by four. 0x1B - Reserved. 0x1C - (serdes4) SerDes 4 250 MHz clock output divided by four. 0x1D - 0x1F Reserved.
22:18	TSTCLK1SEL	RW	0x0	**Test Clock 1 Output Select.** This field selects the clock output driven on the TSTCLK1 pin (GPIO[7] alternate function). See TSTCLK0SEL (above) for encoding of this field.
24:23	Reserved	RO	0x0	
26:25	REFCLKTERM	RW	0x0	**PCI Express Reference Clock Termination** This field controls the termination resistor value between the REFCLKP[1:0] and REFCLKN[1:0] inputs. 0x0 - (ohms110) 110 ohms. 0x1 - (ohms100) 100 ohms. 0x2 - (ohms96) 96 ohms. 0x3 - (ohms90) 90 ohms.
31:27	Reserved	RO	0x0	

1. Should be 62.5 MHz.

Notes

GPIOCS - General Purpose I/O Control and Status (0x0A8)

Bit Field	Field Name	Type	Default Value	Description
7:0	GPIOFUNC	RW	0x0	**GPIO Function.** Each bit in this field controls the corresponding GPIO pin. When set to a one, the corresponding GPIO pin operates as the alternate function as defined in Chapter 8, General Purpose I/O. When a bit is cleared to a zero, the corresponding GPIO pin operates as a general purpose I/O pin.
15:8	GPIOCFG	RW	0x0	**GPIO Configuration.** Each bit in this field controls the corresponding GPIO pin. When a bit is configured as a general purpose I/O pin and the corresponding bit in this field is set, then the pin is configured as a GPIO output. When a bit is configured as a general purpose I/O pin and the corresponding bit in this field is zero, then the pin is configured as an input. When the pin is configured as an alternate function, the behavior of the pin is defined by the alternate function.
23:16	GPIOD	RW	HWINIT	**GPIO Data.** Each bit in this field controls the corresponding GPIO pin. Reading this field returns the current value of each GPIO pin regardless of GPIO pin mode (i.e., alternate function or GPIO pin). Writing a value to this field causes the corresponding pins which are configured as GPIO outputs to change state to the value written.
31:24	Reserved	RO	0x0	Reserved field.

SMBUSSTS - SMBus Status (0x0AC)

Bit Field	Field Name	Type	Default Value	Description
0	Reserved	RO	0x0	Reserved field.
7:1	SSMBADDR	RO	HWINIT	**Slave SMBus Address.** This field contains the SMBus address assigned to the slave SMBus interface.
8	Reserved	RO	0x0	Reserved field.
15:9	MSMBADDR	RO	HWINIT	**Master SMBus Address.** This field contains the SMBus address assigned to the master SMBus interface.
16	Reserved	RO	0x0	Reserved field.
23:17	IOEADDR	RWL	0x0	**Hot-plug I/O Expander Master SMBus Address.** This field contains SMBus address assigned to the hot-plug I/O expander on the master SMBus interface.
24	EEPROMDONE	RO	0x0	**Serial EEPROM Initialization Done.** When the switch is configured to operate in a mode in which serial EEPROM initialization occurs during a fundamental reset, this bit is set when serial EEPROM initialization completes or when an error is detected.

Notes

Bit Field	Field Name	Type	Default Value	Description
25	NAERR	RW1C	0x0	**No Acknowledge Error.** This bit is set if an unexpected NACK is observed during a master SMBus transaction. The setting of this bit may indicate the following: that the addressed device does not exist on the SMBus (i.e., addressing error); data is unavailable or the device is busy; an invalid command was detected by the slave; or invalid data was detected by the slave.
26	LAERR	RW1C	0x0	**Lost Arbitration Error.** When the master SMBus interface loses arbitration for the SMBus, it automatically re-arbitrates for the SMBus. If the master SMBus interface loses 16 consecutive arbitration attempts, then the transaction is aborted and this bit is set.
27	OTHERERR	RW1C	0x0	**Other Error.** This bit is set if a misplaced START or STOP condition is detected by the master SMBus interface.
28	ICSERR	RW1C	0x0	**Initialization Checksum Error.** This bit is set if an invalid checksum is computed during Serial EEPROM initialization or when a configuration done command is not found in the serial EEPROM.
29	URIA	RW1C	0x0	**Unmapped Register Initialization Attempt.** This bit is set if an attempt is made to initialize via serial EEPROM a register that is not defined in the corresponding PCI configuration space.
31:30	Reserved	RO	0x0	Reserved field.

SMBUSCTL - SMBus Control (0x0B0)

Bit Field	Field Name	Type	Default Value	Description
15:0	MSMBCP	RW	HWINIT	**Master SMBus Clock Prescalar.** This field contains a clock prescalar value used during master SMBus transactions. The prescalar clock period is equal to 32 ns multiplied by the value in this field. When the field is cleared to zero or one, the clock is stopped. The initial value of this field is 0x0139 when the master SMBus is configured to operate in slow mode (i.e., 100 KHz) in the boot configuration and to 0x0053[1] when it is configured to operate in fast mode (i.e., 400 KHz).
16	MSMBIOM	RW	0x0	**Master SMBus Ignore Other Masters.** When this bit is set, the master SMBus proceeds with transactions regardless of whether it won or lost arbitration.
17	ICHECKSUM	RW	0x0	**Ignore Checksum Errors.** When this bit is set, serial EEPROM initialization checksum errors are ignored (i.e., the checksum always passes).

Notes

Bit Field	Field Name	Type	Default Value	Description
19:18	SSMBMODE	RW	0x0	**Slave SMBus Mode.** The slave SMBus contains internal glitch counters on the SSMBCLK and SSMBDAT signals that wait approximately 1uS before sampling or driving these signals. This field allows the glitch counter time to be reduced or entirely removed. In some systems, this may permit high speed slave SMBus operation. 0x0 -(normal) Slave SMBus normal mode. Glitch counters operate with 1uS delay. 0x1 -(fast) Slave SMBus interface fast mode. Glitch counters operate with 100nS delay. 0x2 -(disabled) Slave SMBus interface with glitch counters disabled. Glitch counters operate with zero delay which effectively removes them. 0x3 -reserved.
21:20	MSMBMODE	RW	0x0	**Master SMBus Mode.** The master SMBus contains internal glitch counters on the MSMBCLK and MSMBDAT signals that wait approximately 1uS before sampling or driving these signals. This field allows the glitch counter time to be reduced or entirely removed. In some systems, this may permit high speed master SMBus operation. 0x0 -(normal) Master SMBus normal mode. Glitch counters operate with 1uS delay. 0x1 -(fast) Master SMBus interface fast mode. Glitch counters operate with 100nS delay. 0x2 -(disabled) Master SMBus interface with glitch counters disabled. Glitch counters operate with zero delay which effectively removes them. 0x3 -reserved.
31:22	Reserved	RO	0x0	Reserved field.

[1] The MSMBCLK low minimum pulse width is equal to half the period programmed in this field. The value of 0x53, which corresponds to~373 KHz, allows the min low pulse width to be satisfied. In systems where this timing parameter is not critical, the operating frequency may be increased.

EEPROMINTF - Serial EEPROM Interface (0x0B4)

Bit Field	Field Name	Type	Default Value	Description
15:0	ADDR	RW	0x0	**EEPROM Address.** This field contains the byte address in the Serial EEPROM to be read or written.
23:16	DATA	RW	0x0	**EEPROM Data.** A write to this field will initiates a serial EEPROM read or write operation, as selected by the OP field, to the address specified in the ADDR field. When a write operation is selected, the value written to this field is the value written to the serial EEPROM. When a read operation is selected, the value written to this field is ignored and the value read from the serial EEPROM may be read from this field when the DONE bit is set.

Notes

Bit Field	Field Name	Type	Default Value	Description
24	BUSY	RO	0x0	**EEPROM Busy.** This bit is set when a serial EEPROM read or write operation is in progress. 0x0 - (idle) serial EEPROM interface idle 0x1 - (busy) serial EEPROM interface operation in progress
25	DONE	RW1C	0x0	**EEPROM Operation Completed.** This bit is set when a serial EEPROM operation has completed. 0x0 - (notdone) interface is idle or operation in progress 0x1 - (done) operation completed
26	OP	RW	0x0	**EEPROM Operation Select.** This field selects the type of EEPROM operation to be performed when the DATA field is written 0x0 - (write) serial EEPROM write 0x1 - (read) serial EEPROM read
31:27	Reserved	RO	0x0	Reserved field.

IOEXPINTF - I/O Expander Interface (0x0B8)

‘

Bit Field	Field Name	Type	Default Value	Description
7:0	PBHPS	RW	0x0	**Port B Hot-Plug Signals.** Each bit in this field corresponds to a port B hot-plug signal. Reading this field returns the current value of the corresponding hot-plug signals (i.e., the input values last read from the I/O expander and output values supplied to the I/O expander). Writes to this field are ignored unless the I/O Expander Test Mode (IOEXTM) bit is set. When the IOEXTM bit is set, the value for outputs supplied to the I/O expander corresponds to the value written to this field instead of the value supplied by internal logic. Bits in this field which correspond to inputs are always read-only, even when the IOEXTM bit is set.
15:8	PCHPS	RW	0x0	**Port C Hot-Plug Signals.** Each bit in this field corresponds to a port C hot-plug signal. Reading this field returns the current value of the corresponding hot-plug signals (i.e., the input values last read from the I/O expander and output values supplied to the I/O expander). Writes to this field are ignored unless the I/O Expander Test Mode (IOEXTM) bit is set. When the IOEXTM bit is set, the value for outputs supplied to the I/O expander correspond to the value written to this field instead of the value supplied by internal logic. Bits in this field which correspond to inputs are always read-only, even when the IOEXTM bit is set.
23:16	Reserved	RO	0x0	Reserved field.
24	RELOADIOEX	RW	0x0	**Reload I/O Expander Signals.** Writing a one to this field results in an I/O expander SMBus transaction that refreshes all I/O expander input and output signal values. This bit always returns a zero when read.

Notes

Bit Field	Field Name	Type	Default Value	Description
25	IOEXTM	RW	0x0	**I/O Expander Test Mode.** Setting this bit puts the I/O expander into a test mode. In this test mode, Port B and Port C hot-plug output signals generated by the core are ignored and values supplied to the I/O expander correspond to value written to the PBHPS and PCHPS fields.
30:26	Reserved	RO	0x0	Reserved field.
31	DONE	RW1C	0x0	**I/O Expander Operation Done.** This bit is set when the RELOADI-OEX bit in this register or the IOEADDR field in the SMBUSSTS register is written and the corresponding I/O expander SMBus transaction completes; or when the I/O expander is in test mode (i.e., IOEXTM bit set), the PBHPS or PCHPS fields are written, and the corresponding I/O expander SMBus transaction updating the I/O expander outputs completes.

Extended Configuration Space Access and INTx Status Registers

INTSTS - Interrupt Status (0x0F4)

Bit Field	Field Name	Type	Default Value	Description
0	INTA	RO	0x0	**INTA Aggregated State.** Aggregated port state for INTA. 0x0 - (negated) INTA negated 0x1 - (asserted) INTA asserted
1	INTB	RO	0x0	**INTB Aggregated State.** Aggregated port state for INTB. 0x0 - (negated) INTB negated 0x1 - (asserted) INTB asserted
2	INTC	RO	0x0	**INTC Aggregated State.** Aggregated port state for INTC. 0x0 - (negated) INTC negated 0x1 - (asserted) INTC asserted
3	INTD	RO	0x0	**INTD Aggregated State.** Aggregated port state for INTD. 0x0 - (negated) INTD negated 0x1 - (asserted) INTD asserted
31:4	Reserved	RO	0x0	Reserved.

ECFGADDR - Extended Configuration Space Access Address (0x0F8)

Bit Field	Field Name	Type	Default Value	Description
0:1	Reserved	RO	0x0	Reserved.
7:2	REG	RW	0x0	**Register Number.** This field selects the configuration register number as defined by Section 7.2.2 of the PCI Express Base Specification, Rev. 1.0a
11:8	EREG	RW	0x0	**Extended Register Number.** This field selects the extended configuration register number as defined by Section 7.2.2 of the PCI Express Base Specification, Rev. 1.0a

Notes

Bit Field	Field Name	Type	Default Value	Description
31:12	Reserved	RO	0x0	Reserved.

ECFGDATA - Extended Configuration Space Access Data (0x0FC)

Bit Field	Field Name	Type	Default Value	Description
31:0	DATA	RW	0x0	**Configuration Data.** A read from this field will return the configuration space register value pointed to by the ECFGADDR register. A write to this field will update the contents of the configuration space register pointed to by the ECFGADDR register with the value written. For both reads and writes, the byte enables correspond to those used to access this field. When the ECFGADDR register points to the ECFGDATA register, then reads from ECFGDATA return zero and writes are ignored. When the ECFGADDR register points to itself, writes to the ECFGDATA register modify the contents of the ECFGADDR register. SMBus reads of this field return a value of zero and SMBus writes have no effect.

PCI Express Virtual Channel Capability

PCIEVCECAP - PCI Express Virtual Channel Enhanced Capability Header (0x100)

Bit Field	Field Name	Type	Default Value	Description
15:0	CAPID	RO	0x2	**Capability ID.** The value of 0x2. indicates a virtual channel capability structure.
19:16	CAPVER	RO	0x1	**Capability Version.** The value of 0x1. indicates compatibility with version 1 of the specification.
31:20	NXTPTR	RO	0x0	**Next Pointer.** The value of 0x0 indicates that there are no extended capabilities.

PVCCAP1- Port VC Capability 1 (0x104)

Bit Field	Field Name	Type	Default Value	Description
2:0	EVCCNT	RO	0x0	**Extended VC Count.** The value 0x0 indicates only implementation of the default VC.
6:4	LPEVCCNT	RO	0x0	**Low Priority Extended VC Count.** The value of 0x0 indicates only implementation of the default VC.
9:8	REFCLK	RO	0x0	**Reference Clock.** Time-based WRR is not implemented.

Notes

Bit Field	Field Name	Type	Default Value	Description
11:10	PATBLSIZ	RO	0x1	**Port Arbitration Table Entry Size.** This field indicates the size of the port arbitration table in the device. The value in the PES12N3 is set to 0x1 to indicate a 2-bit table. 0x0 - (bit1) Port arbitration table is 1-bit 0x1 - (bit2) Port arbitration table is 2-bits 0x2 - (bit4) Port arbitration table is 4-bits 0x3 - (bit8) Port arbitration table is 8-bits
31:12	Reserved	RO	0x0	

VCR0CAP- VC Resource 0 Capability (0x110)

Bit Field	Field Name	Type	Default Value	Description
7:0	PARBC	RO	0x3	**Port Arbitration Capability.** This field indicates the type of port arbitration supported by the VC. Each bit corresponds to a Port Arbitration capability. When more than one arbitration scheme is supported, multiple bits may be set. The PES12N3 supports hardware fixed round robin and weighted round robin with 32 phases. bit 0 - hardware fixed round robin bit 1 - weighted round robin with 32 phases bit 2 - N/A bit 3 - N/A bit 4 - N/A bit 5 - N/A
13:8	Reserved	RO	0x0	
14	APS	RO	0x0	**Advanced Packet Switching.** Not supported.
15	RJST	RO	0x0	**Reject Snoop Transactions.** Not supported for switch ports.
22:16	MAXTS	RO	0x0	**Maximum Time Slots.** Since this VC does not support time-based WRR, this field is not valid.
23	Reserved	RO	0x0	
31:24	PATBLOFFSET	RO	0x2	**Port Arbitration Table Offset.** This field contains the offset of the port arbitration table from the base address of the Virtual Channel Capability structure in double quad words (16 bytes).

VCR0CTL- VC Resource 0 Control (0x114)

Bit Field	Field Name	Type	Default Value	Description
7:0	TCVCMAP	bit 0 RO bits 1 through 7 RW	0xFF	**TC/VC Map.** This field indicates the TCs that are mapped to the VC resource. Each bit corresponds to a TC. When a bit is set, the corresponding TC is mapped to the VC.

Notes

Bit Field	Field Name	Type	Default Value	Description
15:8	Reserved	RO	0x0	
16	LPAT	RW	0x0	**Load Port Arbitration Table.** This bit, when set, updates the Port Arbitration logic from the Port Arbitration Table for the VC resource. In addition, this field is only valid when the Port Arbitration Table is used by the selected Port Arbitration scheme (that is indicated by a set bit in the Port Arbitration Capability field selected by Port Arbitration Select). Software sets this bit to signal hardware to update Port Arbitration logic with new values stored in Port Arbitration Table; clearing this bit has no effect. Software uses the Port Arbitration Table Status bit to confirm whether the new values of Port Arbitration Table are completely latched by the arbitration logic. This bit always returns 0 when read.
19:17	PARBSEL	RW	0x0	**Port Arbitration Select.** This field configures the VC resource to provide a particular Port Arbitration service. The permissible values of this field is a number that corresponds to one of the asserted bits in the Port Arbitration Capability field of the VC resource.
23:20	Reserved	RO	0x0	
26:24	VCID	RO	0x0	**VC ID.** This field assigns a VC ID to the VC resource. Since the PES12N3 implements only a single VC, this field is hardwired to zero.
30:27	Reserved	RO	0x0	
31	VCEN	RO	0x1	**VC Enable.** This field, when set, enables a virtual channel. Since the PES12N3 implements only a single VC, this field is hardwired to one (enabled).

VCR0STS - VC Resource 0 Status (0x118)

Bit Field	Field Name	Type	Default Value	Description
0	PATS	RO	0x0	**Port Arbitration Table Status.** This bit indicates the coherency status of the port arbitration table associated with the VC resource and is valid only when the port arbitration table is used by the selected arbitration algorithm. This bit is set when any entry of the port arbitration table is written by software and remains set until hardware finishes loading the value after software sets the LPAT field in the VCR0CTL register.
1	VCNEG	RO	0x0	**VC Negotiation Pending.** Since the PES12N3 implements only a single VC (i.e., the default VC) this field indicates the status of the process of flow control initialization.
30:2	Reserved	RO	0x0	

Notes

VCR0TBL0 - VC Resource 0 Arbitration Table Entry 0 (0x120)

Bit Field	Field Name	Type	Default Value	Description
1:0	PHASE0	RW	0x0	**Phase 0.** This field contains the port ID for the corresponding port arbitration period. Selecting an invalid port ID results in the entry being skipped without delay. The port arbitration behavior when this field contains an illegal value (i.e., reserved or the egress port ID) is undefined. 0x0 - (port_a) Port A (upstream port) 0x1 - (port_b) Port B 0x2 - (port_c) Port C 0x3 - reserved
3:2	PHASE1	RW	0x0	**Phase 1.** This field contains the port ID for the corresponding port arbitration period.
5:4	PHASE2	RW	0x0	**Phase 2.** This field contains the port ID for the corresponding port arbitration period.
7:6	PHASE3	RW	0x0	**Phase 3.** This field contains the port ID for the corresponding port arbitration period.
9:8	PHASE4	RW	0x0	**Phase 4.** This field contains the port ID for the corresponding port arbitration period.
11:10	PHASE5	RW	0x0	**Phase 5.** This field contains the port ID for the corresponding port arbitration period.
13:12	PHASE6	RW	0x0	**Phase 6.** This field contains the port ID for the corresponding port arbitration period.
15:14	PHASE7	RW	0x0	**Phase 7.** This field contains the port ID for the corresponding port arbitration period.
17:16	PHASE8	RW	0x0	**Phase 8.** This field contains the port ID for the corresponding port arbitration period.
19:18	PHASE9	RW	0x0	**Phase 9.** This field contains the port ID for the corresponding port arbitration period.
21:20	PHASE10	RW	0x0	**Phase 10.** This field contains the port ID for the corresponding port arbitration period.
23:22	PHASE11	RW	0x0	**Phase 11.** This field contains the port ID for the corresponding port arbitration period.
25:24	PHASE12	RW	0x0	**Phase 12.** This field contains the port ID for the corresponding port arbitration period.
27:26	PHASE13	RW	0x0	**Phase 13.** This field contains the port ID for the corresponding port arbitration period.
29:28	PHASE14	RW	0x0	**Phase 14.** This field contains the port ID for the corresponding port arbitration period.
31:30	PHASE15	RW	0x0	**Phase 15.** This field contains the port ID for the corresponding port arbitration period.

Notes

VCR0TBL1 - VC Resource 0 Arbitration Table Entry 1 (0x124)

Bit Field	Field Name	Type	Default Value	Description
1:0	PHASE16	RW	0x0	**Phase 16.** This field contains the port ID for the corresponding port arbitration period. Selecting an invalid port ID results in the entry being skipped without delay. 0x0 - (port_a) Port A (upstream port) 0x1 - (port_b) Port B 0x2 - (port_c) Port C 0x3 - reserved
3:2	PHASE17	RW	0x0	**Phase 17.** This field contains the port ID for the corresponding port arbitration period.
5:4	PHASE18	RW	0x0	**Phase 18.** This field contains the port ID for the corresponding port arbitration period.
7:6	PHASE19	RW	0x0	**Phase 19.** This field contains the port ID for the corresponding port arbitration period.
9:8	PHASE20	RW	0x0	**Phase 20.** This field contains the port ID for the corresponding port arbitration period.
11:10	PHASE21	RW	0x0	**Phase 21.** This field contains the port ID for the corresponding port arbitration period.
13:12	PHASE22	RW	0x0	**Phase 22.** This field contains the port ID for the corresponding port arbitration period.
15:14	PHASE23	RW	0x0	**Phase 23.** This field contains the port ID for the corresponding port arbitration period.
17:16	PHASE24	RW	0x0	**Phase 24.** This field contains the port ID for the corresponding port arbitration period.
19:18	PHASE25	RW	0x0	**Phase 25.** This field contains the port ID for the corresponding port arbitration period.
21:20	PHASE26	RW	0x0	**Phase 26.** This field contains the port ID for the corresponding port arbitration period.
23:22	PHASE27	RW	0x0	**Phase 27.** This field contains the port ID for the corresponding port arbitration period.
25:24	PHASE28	RW	0x0	**Phase 28.** This field contains the port ID for the corresponding port arbitration period.
27:26	PHASE29	RW	0x0	**Phase 29.** This field contains the port ID for the corresponding port arbitration period.
29:28	PHASE30	RW	0x0	**Phase 30.** This field contains the port ID for the corresponding port arbitration period.
31:30	PHASE31	RW	0x0	**Phase 31.** This field contains the port ID for the corresponding port arbitration period.

Notes

Test Mode Registers

TMCTL - Test Mode Control (0x0BC)

Bit Field	Field Name	Type	Default Value	Description
7:0	PAEN	RW	0xFF	**Port A Enable.** Each bit in this field corresponds to a port lane (e.g., bit zero corresponds to lane zero of the port). When the switch is in a SerDes test mode and a bit in this field is set, then the corresponding port lane participates in the test mode. When the switch is in a SerDes test mode and a bit in this field is cleared, then the corresponding lane does not participate in the test mode and the value received on the SerDes input is ignored and the SerDes output is held in a quiescent state. This field has no effect in a non-test mode switch setting. 0x0 - (disabled) corresponding port lane is disabled and does not participate in test mode. 0x1 - (enabled) corresponding port lane is enabled and participates in test mode.
15:8	PBEN	RW	0xFF	**Port B Enable.** Each bit in this field corresponds to a port lane (e.g., bit zero corresponds to lane zero of the port). When the switch is in a SerDes test mode and a bit in this field is set, then the corresponding port lane participates in the test mode. When the switch is in a SerDes test mode and a bit in this field is cleared, then the corresponding lane does not participate in the test mode and the value received on the SerDes input is ignored and the SerDes output is held in a quiescent state. This field has no effect in a non-test mode switch setting. 0x0 - (disabled) corresponding port lane is disabled and does not participate in test mode. 0x1 - (enabled) corresponding port lane is enabled and participates in test mode.
23:16	PCEN	RW	0xFF	**Port C Enable.** Each bit in this field corresponds to a port lane (e.g., bit zero corresponds to lane zero of the port). When the switch is in a SerDes test mode and a bit in this field is set, then the corresponding port lane participates in the test mode. When the switch is in a SerDes test mode and a bit in this field is cleared, then the corresponding lane does not participate in the test mode and the value received on the SerDes input is ignored and the SerDes output is held in a quiescent state. This field has no effect in a non-test mode switch setting. 0x0 - (disabled) corresponding port lane is disabled and does not participate in test mode. 0x1 - (enabled) corresponding port lane is enabled and participates in test mode.
24	IBS	RW	0x0	**Invert-Bit Stream**. When this bit is set, the transmitted bit-stream selected by the TMCNTLSEL0 and TMCNTPSEL0 fields is inverted. This occurs in normal as well as test modes.

Notes

Bit Field	Field Name	Type	Default Value	Description
25	TXRS	RW	0x0	**Transmit Re-Sync.** Writing a one to this bit position while the PES12N3 is in a test mode that requires synchronization, causes the PRBS or pattern generator on each lane to start a synchronization sequence. This field always returns a value of zero when read.
26	RXRS	RW	0x0	**Receive Re-Sync.** Writing a one to this bit position while the PES12N3 is in a test mode that requires synchronization, causes the PRBS or pattern checker on each lane to start a synchronization sequence. This field always returns a value of zero when read.
31:27	Reserved	RO	0x0	Reserved field.

TMFSTS - Test Mode Fail Status (0x0C0)

Bit Field	Field Name	Type	Default Value	Description
7:0	PATF	RW1C	0x0	**Port A Test Failure.** Each bit in this field corresponds to a port lane (e.g., bit zero corresponds to lane zero of the port). When the switch is configured to operate in a SerDes test mode, the lane is enabled in the TMCTL register, and an error is detected on a lane, then the corresponding bit in this field is set.
15:8	PBTF	RW1C	0x0	**Port B Test Failure.** Each bit in this field corresponds to a port lane (e.g., bit zero corresponds to lane zero of the port). When the switch is configured to operate in a SerDes test mode, the lane is enabled in the TMCTL register, and an error is detected on a lane, then the corresponding bit in this field is set.
23:16	PCTF	RW1C	0x0	**Port C Test Failure.** Each bit in this field corresponds to a port lane (e.g., bit zero corresponds to lane zero of the port). When the switch is configured to operate in a SerDes test mode, the lane is enabled in the TMCTL register, and an error is detected on a lane, then the corresponding bit in this field is set.
31:24	Reserved	RO	0x0	Reserved field.

TMSSTS - Test Mode Synchronization Status (0x0C4)

Bit Field	Field Name	Type	Default Value	Description
7:0	PASA	RW1C	0x0	**Port A Synchronization Achieved.** Each bit in this field corresponds to a port lane (e.g., bit zero corresponds to lane zero of the port). When the lane is enabled in the TMCTL register, and synchronization has been achieved, then the corresponding bit in this field is set.

Notes

Bit Field	Field Name	Type	Default Value	Description
15:8	PBSA	RW1C	0x0	**Port B Synchronization Achieved.** Each bit in this field corresponds to a port lane (e.g., bit zero corresponds to lane zero of the port). When the lane is enabled in the TMCTL register, and synchronization has been achieved, then the corresponding bit in this field is set.
23:16	PCSA	RW1C	0x0	**Port C Synchronization Achieved.** Each bit in this field corresponds to a port lane (e.g., bit zero corresponds to lane zero of the port). When the lane is enabled in the TMCTL register, and synchronization has been achieved, then the corresponding bit in this field is set.
31:24	Reserved	RO	0x0	Reserved field.

TMCNTCFG - Test Mode Count Configuration (0x0C8)

Bit Field	Field Name	Type	Default Value	Description
2:0	TMCNTLSEL0	RW	0x0	**Test Mode Count Lane Select 0.** In SerDes test mode, this field selects which SerDes lane of the port specified in the TMCNTPSEL0 field for which failures are counted in the Test Mode Error Count 0 (TMERRCNT0) field of the Test Mode Count 0 (TMCNT0) register.
4:3	TMCNTPSEL0	RW	0x0	**Test Mode Count Port Select 0.** In SerDes test mode, this field selects the port for which SerDes lane failures are counted in the the Test Mode Error Count 0 (TMERRCNT0) field of the Test Mode Count 0 (TMCNT0) register. 0x0 - (porta) port A 0x1 - (portb) port B 0x2 - (portc) port C 0x3 - (reserved) reserved
7:5	TMCNTLSEL1	RW	0x0	**Test Mode Count Lane Select 1.** In SerDes test mode, this field selects which SerDes lane of the port specified in the TMCNTPSEL1 field for which failures are counted in the Test Mode Error Count 1 (TMERRCNT1) field of the Test Mode Count 0 (TMCNT0) register.
9:8	TMCNTPSEL1	RW	0x0	**Test Mode Count Port Select 1.** In SerDes test mode, this field selects the port for which SerDes lane failures are counted in the the Test Mode Error Count 1 (TMERRCNT1) field of the Test Mode Count 0 (TMCNT0) register. 0x0 - (porta) port A 0x1 - (portb) port B 0x2 - (portc) port C 0x3 - (reserved) reserved
12:10	TMCNTLSEL2	RW	0x0	**Test Mode Count Lane Select 2.** In SerDes test mode, this field selects which SerDes lane of the port specified in the TMCNTPSEL2 field for which failures are counted in the Test Mode Error Count 2 (TMERRCNT2) field of the Test Mode Count 1 (TMCNT1) register.

Notes

Bit Field	Field Name	Type	Default Value	Description
14:13	TMCNTPSEL2	RW	0x0	**Test Mode Count Port Select 2.** In SerDes test mode, this field selects the port for which SerDes lane failures are counted in the the Test Mode Error Count 2 (TMERRCNT2) field of the Test Mode Count 1 (TMCNT1) register. 0x0 - (porta) port A 0x1 - (portb) port B 0x2 - (portc) port C 0x3 - (reserved) reserved
17:15	TMCNTLSEL3	RW	0x0	**Test Mode Count Lane Select 3.** In SerDes test mode, this field selects which SerDes lane of the port specified in the TMCNTPSEL3 field for which failures are counted in the Test Mode Error Count 3 (TMERRCNT3) field of the Test Mode Count 1 (TMCNT1) register.
19:18	TMCNTPSEL3	RW	0x0	**Test Mode Count Port Select 3.** In SerDes test mode, this field selects the port for which SerDes lane failures are counted in the the Test Mode Error Count 3 (TMERRCNT3) field of the Test Mode Count 1 (TMCNT1) register. 0x0 - (porta) port A 0x1 - (portb) port B 0x2 - (portc) port C 0x3 - (reserved) reserved
22:20	TMCNTLSEL4	RW	0x0	**Test Mode Count Lane Select 4.** In SerDes test mode, this field selects which SerDes lane of the port specified in the TMCNTPSEL4 field for which failures are counted in the Test Mode Error Count 4 (TMERRCNT4) field of the Test Mode Count 2 (TMCNT2) register.
24:23	TMCNTPSEL4	RW	0x0	**Test Mode Count Port Select 4.** In SerDes test mode, this field selects the port for which SerDes lane failures are counted in the the Test Mode Error Count 4 (TMERRCNT4) field of the Test Mode 2 Count (TMCNT2) register. 0x0 - (porta) port A 0x1 - (portb) port B 0x2 - (portc) port C 0x3 - (reserved) reserved
27:25	TMCNTLSEL5	RW	0x0	**Test Mode Count Lane Select 5.** In SerDes test mode, this field selects which SerDes lane of the port specified in the TMCNTPSEL5 field for which failures are counted in the Test Mode Error Count 5 (TMERRCNT5) field of the Test Mode Count 2 (TMCNT2) register.
29:28	TMCNTPSEL5	RW	0x0	**Test Mode Count Port Select 5.** In SerDes test mode, this field selects the port for which SerDes lane failures are counted in the the Test Mode Error Count 5 (TMERRCNT5) field of the Test Mode Count 2 (TMCNT2) register. 0x0 - (porta) port A 0x1 - (portb) port B 0x2 - (portc) port C 0x3 - (reserved) reserved

Notes

Bit Field	Field Name	Type	Default Value	Description
31:30	Reserved	RO	0x0	Reserved field.

TMCNT0 - Test Mode Count 0 (0x0CC)

Bit Field	Field Name	Type	Default Value	Description
15:0	TMERRCNT0	RCW	0x0	**Test Mode Error Count 0.** When the switch is configured to operate in a SerDes test mode, and an error is detected in port and lane specified by the TMCNTPSEL0 and TMCNTLSEL0 fields in the TMCNTCFG register, then the value in this field in incremented. This counter saturates at its maximum value. This field is atomically cleared when read.
31:16	TMERRCNT1	RCW	0x0	**Test Mode Error Count 1.** When the switch is configured to operate in a SerDes test mode, and an error is detected in port and lane specified by the TMCNTPSEL1 and TMCNTLSEL1 fields in the TMCNTCFG register, then the value in this field in incremented. This counter saturates at its maximum value. This field is atomically cleared when read.

TMCNT1 - Test Mode Count 1 (0x0D0)

Bit Field	Field Name	Type	Default Value	Description
15:0	TMERRCNT2	RCW	0x0	**Test Mode Error Count 2.** When the switch is configured to operate in a SerDes test mode, and an error is detected in port and lane specified by the TMCNTPSEL2 and TMCNTLSEL2 fields in the TMCNTCFG register, then the value in this field in incremented. This counter saturates at its maximum value. This field is atomically cleared when read.
31:16	TMERRCNT3	RCW	0x0	**Test Mode Error Count 3.** When the switch is configured to operate in a SerDes test mode, and an error is detected in port and lane specified by the TMCNTPSEL3 and TMCNTLSEL3 fields in the TMCNTCFG register, then the value in this field in incremented. This counter saturates at its maximum value. This field is atomically cleared when read.

Notes

TMCNT2 - Test Mode Count 2 (0x0D4)

Bit Field	Field Name	Type	Default Value	Description
15:0	TMERRCNT4	RCW	0x0	**Test Mode Error Count 4.** When the switch is configured to operate in a SerDes test mode, and an error is detected in port and lane specified by the TMCNTPSEL4 and TMCNTLSEL4 fields in the TMCNTCFG register, then the value in this field in incremented. This counter saturates at its maximum value. This field is atomically cleared when read.
31:16	TMERRCNT5	RCW	0x0	**Test Mode Error Count 5.** When the switch is configured to operate in a SerDes test mode, and an error is detected in port and lane specified by the TMCNTPSEL5 and TMCNTLSEL5 fields in the TMCNTCFG register, then the value in this field in incremented. This counter saturates at its maximum value. This field is atomically cleared when read.

System Integrity

SWSICTL - Switch System Integrity Control (0x500)

Bit Field	Field Name	Type	Default Value	Description
0	DEEPC	RW	0x0	**Disable End-to-End Parity Checking.** When this bit is set, end-to-end parity is not checked by the port and errors are never generated. End-to-end parity is always computed for data sent by the port to the switch core and cannot be disabled.
1	SEEPC	RW	0x0	**Silent End-to-End Parity Checking.** When this bit is set and end-to-end parity checking is enabled, end-to-end parity errors detected at the port are logged in the EEPERRC field in the SWSIPECNT register but no other action is taken (i.e., TLP is not nullified, error non-fatal is not generated, and the DPE bit in the PCISTS register is not set)
2	GBEEP	RW	0x0	**Generate Bad End-to-End Parity.** When this bit is set, bad parity is generated for all double words in TLPs emitted to the switch core (i.e., those received on the ingress port or generated by the port) whose TLP header Length field (i.e., bits seven through zero of byte zero of the TLP header) match the value in the Error Match Length (Length) field in this register
3	SPTLPTO	RW	0x0	**Silent Posted TLP Time-out.** When this bit is set and a TLP is dropped since it has been in a posted input buffer for more than 50ms, then the error is logged in the PTLPTOC field in the SWSITDCNT register but no other action is taken (i.e., error non-fatal is not generated).
7:4	Reserved	RO	0x0	

Notes

Bit Field	Field Name	Type	Default Value	Description
15:8	LENGTH	RW	0x0	**Error Match Length.** When the GBEEP bit is set, bad parity is generated for all double words in TLPs emitted to the switch core (i.e., those received on the ingress port or generated by the stack) whose TLP header Length field (i.e., bits seven through zero of byte zero of the TLP header) matches the value in this field.
31:16	Reserved	RO	0x0	

SWSIPECNT - Switch System Integrity Parity Error Count (0x504)

Bit Field	Field Name	Type	Default Value	Description
7:0	EEPERRC	RCW	0x0	**End-to-End Parity Error Count.** This field is incremented each time an end-to-end parity error is detected at the port until it saturates at its maximum count value (i.e., it does not roll over from 0xFF to 0x00). To prevent error flooding, ERR_NONFATAL messages are inhibited from being sent when this field is at its maximum value of 0xFF. This counter saturates at its maximum value. Reading this field causes it to be cleared.
31:8	Reserved	RO	0x0	

SWSITDCNT - Switch System Integrity Time-Out Drop Count (0x508)

Bit Field	Field Name	Type	Default Value	Description
7:0	PTLPTOC	RCW	0x0	**Posted TLP Time-Out Count.** This field is incremented each time a TLP is discarded from a posted input buffer because it is more than 50 ms old. To prevent error flooding, ERR_NONFATAL messages are inhibited from being sent when this field is at its maximum value of 0xFF. This counter saturates at its maximum value. Reading this field causes it to be cleared.
15:8	NPTLPTOC	RCW	0x0	**Non-Posted TLP Time-Out Count.** This field is incremented each time a TLP is discarded from a non-posted input buffer because it is more than 50 ms old. This counter saturates at its maximum value. Reading this field causes it to be cleared.
24:16	CTLPTOC	RCW	0x0	**Completion TLP Time-Out Count.** This field is incremented each time a TLP is discarded from a completion input buffer because it is more than 50 ms old. This counter saturates at its maximum value. Reading this field causes it to be cleared.
31:8	Reserved	RO	0x0	

Notes

Notes

Device Test Modes

In addition to normal operating mode, the PES12N3 has four test modes. These test modes are selected by asserting the appropriate test mode value on the Switch Mode (SWMODE[3:0]) pins during a fundamental reset.

Switch modes are summarized in Chapter 2, Table 2.2. When one of the switch test modes described below is selected, the normal fundamental reset sequence, described in section Fundamental Reset on page 2-5, is bypassed and the following reset sequence is executed.

1. Wait for the fundamental reset condition to clear (e.g., negation of PERSTN).
2. The SerDes is initialized.
3. Sample the boot configuration signals listed in Table 2.2.
4. The SWMODE[3:0] signals are examined to determine the switch operating mode.
5. The slave SMBus is taken out of reset and initialized.
6. Device test mode operation begins.

While the device is in test mode, the slave SMBus is operational and may be used to read and write any register in the device.

The Invert Bit-Stream (IBS) bit in the port A Test Mode Control (PA_TMCTL) allows the transmitted bit stream of any of the 24 lanes to be inverted. This capability may be used to validate the correct operation of the following test modes (i.e., validate that a test mode does in fact correctly check and report errors).

All test modes that externally loop back data transmitted by the PES12N3 to a PES12N3 input SerDes may be configured to perform this loopback on-chip by setting the appropriate bit(s) in the ILBE field and configuring the LBDRVOP bit, both of which are in a port's SERDESCFG register.

10-bit Loopback Test Mode (SWMODE[3:0] = 0x8)

In 10-bit loopback test mode, data received on a SerDes lane is looped back to its output. This loopback occurs in the SerDes and is graphically illustrated in Figure 10.1. After reset, all 24-lanes of the device enter this test mode. However, individual ports may be enabled and disabled. A port is enabled when all of the bits in the PAEN, PBEN or PCEN field corresponding to the port are set in the Port A Test Mode Control (PA_TMCTL) register. If any bit in one of these fields is cleared, then the corresponding port is disabled.

The incoming serial bit stream must be synchronous to the PES12N3 reference clock. This test mode has no failure or synchronization conditions.

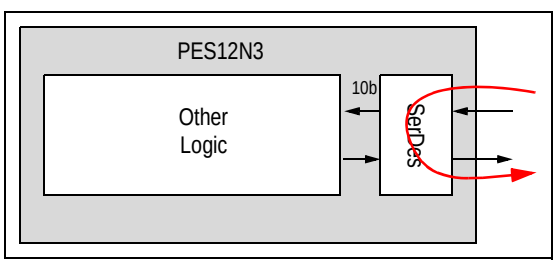

Figure 10.1 10-bit Loopback Test Mode

Notes

Internal Pseudo Random Bit Stream Self-Test Test Mode (SWMODE[3:0] = 0xA)

In the Internal Pseudo Random Bit Stream Self-test Test mode, a pseudo random bit stream is generated, looped back internally through the SerDes, and checked. This mode is graphically illustrated in Figure 10.2.

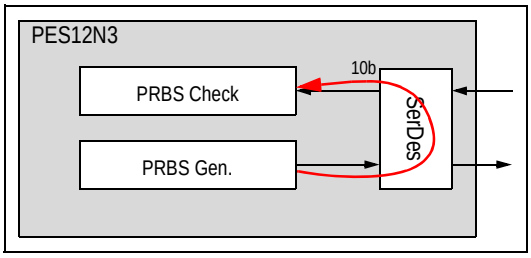

Figure 10.2 Internal Pseudo Random Bit Stream Self-Test

After reset, all 24-lanes of the device enter this test mode. However, individual lanes may be enabled and disabled by setting/clearing bits in the PAEN, PBEN and PCEN fields in the port A Test Mode Control (PA_TMCTL) register. When this test mode is entered, the PRBS generator sends 1024 STP symbols followed by a 127 bit PRBS sequence. Once the sequence completes, the entire process repeats.

The PRBS sequence is generated by a 7-bit LFSR with primitive polynomial $x^6 + x + 1$ initialized to all ones. The operation of the PRBS is equivalent to a PRBS operating at the SerDes bit clock followed by a 10-bit shift register. The 10-bit symbol produced is the output of this shift register every 10 bit clocks. The first 10 symbols produced in the PRBS sequence are shown in Table 10.1.

Symbol Number	Symbol
1	0b0000001000
2	0b0011000010
3	0b1000111100
4	0b1000101100
5	0b1110101001
6	0b1111010000
7	0b1110001001
8	0b0011011010
9	0b1101111011
10	0b0001101001

Table 10.1 PRBS LFSR Symbol Output

When this test mode is entered, the PRBS checker receives and discards STP symbols. This is referred to as PRBS synchronization. After 64 STP symbols are received, the PRBS checker is said to have achieved synchronization and the corresponding bit in the PASA, PBSA or PCSA field in the port A Test Mode Synchronization Status (PA_TMSSTS) register is set.

The first non-STP symbol received after synchronization has been achieved is assumed to be the first symbol of the PRBS sequence. This value is checked against the expected value. If a mismatch is found, the corresponding bit in the PATF, PBTF or PCTF field of the port A Test Mode Fail Status (PA_TMFSTS) register is set.

Notes

Errors on up to six lanes may be concurrently counted using the six error counters. When a lane is selected by a Test Mode Count Lane Select (TMCNTLSEL[0..5]) and the corresponding Test Mode Count Port Select (TMCNTPSEL[0..5]) fields in the port A Test Mode Count Configuration (TMCNTCTL) register, then the corresponding Test Mode Error Count [0..5] (TMERRCNT[0..5]) field is incremented in the appropriate port A Test Mode Count (PA_TMCNT[0..3]) register whenever an error is detected.

Checking continues until the last symbol of the PRBS sequence is received. Once this occurs, checking continues at the start of the PRBS sequence.

When a one is written to the Transmit Re-Sync (TXRS) bit in the PA_TMCTL register, the PRBS generator repeats the synchronization process described above. When a one is written to the Receive Re-Sync (RXRS) bit in the PA_TMCTL register, the PRBS checker repeats the synchronization process described above.

External Pseudo Random Bit Stream Self-Test Test Mode (SWMODE[3:0] = 0xB)

The operation of external pseudo random bit stream self-test mode is exactly the same as internal pseudo random bit stream self-test test mode, except that the bit stream is transmitted on the SerDes lane and is assumed to be externally looped back. This mode is graphically illustrated in Figure 10.3.

The recovered SerDes receive clock is used by the PRBS checker.

No frequency compensation is performed in this test mode. Thus, if an external bit stream generator is used to generate the receive pattern, the same reference clock should be provided to the PES12N3 as to the bit stream generator.

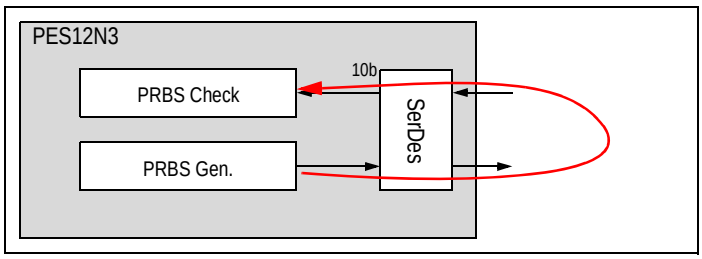

Figure 10.3 External Pseudo Random Bit Stream Self-Test

SerDes Broadcast Test Mode (SWMODE[3:0] = 0xD)

In SerDes Broadcast Test Mode, the symbols received on port C lane zero (i.e., PECRP[0] and PECRN[0]) are broadcast, with a one 250 MHz clock cycle delay between lanes, on all lanes of the device. This test mode uses data after 8b-10b conversion. Thus, the same reference clock should be provided to the PES12N3 as to the bit stream generator.

After reset, all 24-lanes of the device enter this test mode. However, individual lanes may be enabled and disabled by setting/clearing bits in the PAEN, PBEN and PCEN fields in the port A Test Mode Control (PA_TMCTL) register. The delay between the received pattern on port C lane zero and that transmitted on the SerDes lanes of the device is shown in Table 10.2.

250 MHz Clock Cycle	Symbol Received on Port C Lane 0	Symbol Transmitted																							
		Port C Lane								Port A Lane								Port B Lane							
		0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
0	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1	B	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2	C	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3	D	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4	E	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
5	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
6	G	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
7	H	B	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
8	I	C	B	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
9	J	D	C	B	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
10	K	E	D	C	B	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
11	L	F	E	D	C	B	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
12	M	G	F	E	D	C	B	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
13	N	H	G	F	E	D	C	B	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
14	O	I	H	G	F	E	D	C	B	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
15	P	J	I	H	G	F	E	D	C	B	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-
16	Q	K	J	I	H	G	F	E	D	C	B	A	-	-	-	-	-	-	-	-	-	-	-	-	-
17	R	L	K	J	I	H	G	F	E	D	C	B	A	-	-	-	-	-	-	-	-	-	-	-	-
18	S	M	L	K	J	I	H	G	F	E	D	C	B	A	-	-	-	-	-	-	-	-	-	-	-
19	T	N	M	L	K	J	I	H	G	F	E	D	C	B	A	-	-	-	-	-	-	-	-	-	-
20	U	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A	-	-	-	-	-	-	-	-	-
21	V	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A	-	-	-	-	-	-	-	-
22	W	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A	-	-	-	-	-	-	-
23	X	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A	-	-	-	-	-	-
24	Y	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A	-	-	-	-	-
25	-	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A	-	-	-	-
26	-	U	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A	-	-	-
27	-	V	U	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A	-	-
28	-	W	V	U	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A	-
29	-	X	W	V	U	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A
30	-	Y	X	W	V	U	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B

Table 10.2 SerDes Broadcast Test Mode Operation Transmit Delay Operation

Notes

SerDes Test Clock

Each of the six on-chip PLLs generates a 250 MHz clock. The output of any of these PLL clocks divided by four (i.e., a 62.5 MHz clock) may be output on GPIO alternate function pins. In addition, the 2.5 GHz recovered SerDes receive clock from any of the 24 lanes divided by 40 (i.e., 62.5 MHz) may be selected.

SerDes Test Clock 0 (TSTCLK0) is an alternate function of GPIO[6]. The clock output on this alternate function is selected by the Test Clock 0 Select (TSTCLK0SEL) field in the SWCTL register. SerDes Test Clock 1 (TSTCLK1) is an alternate function of GPIO[7]. The clock output on this alternate function is selected by the Test Clock 1 Select (TSTCLK1SEL) field in the SWCTL register.

JTAG Boundary Scan

Notes

Introduction

The JTAG Boundary Scan interface provides a way to test the interconnections between integrated circuit pins after they have been assembled onto a circuit board.

There are two pin types present in the PES12N3: AC-coupled and DC-coupled (also called AC and DC pins). The Boundary Scan interface in the PES12N3 is both IEEE 1149.1 and 1149.6 compliant to allow simultaneous testing of both AC and DC pins. The DC pins are those “normal” pins that do not require AC-coupling and are tested in the PES12N3 using the IEEE 1149.1 standard.

The presence of AC-coupling capacitors on some of the PES12N3 pins prevents DC values from being driven between a driver and receiver. An AC Boundary Scan methodology, as described in IEEE 1149.6, is available to provide a time-varying signal to pass through the AC-coupling when in AC test mode.

Test Access Point

The system logic utilizes a 16-state, six-bit TAP controller, a four-bit instruction register, and five dedicated pins to perform a variety of functions. The primary use of the JTAG TAP Controller state machine is to allow the five external JTAG control pins to control and access the PES12N3's many external signal pins. The JTAG TAP Controller can also be used for identifying the device part number. The JTAG logic of the PES12N3 is depicted in Figure 11.1.

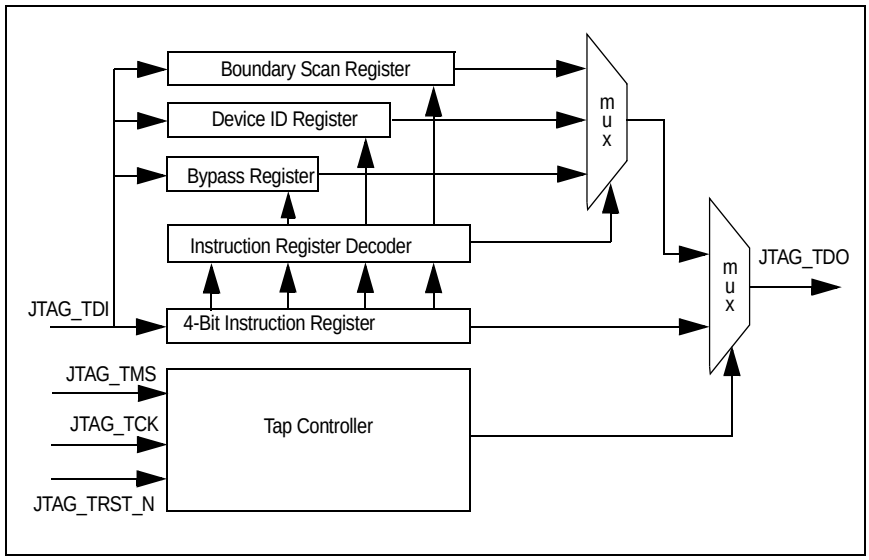

Figure 11.1 Diagram of the JTAG Logic

Refer to the IEEE 1149.1 and 1149.6 documents for a complete operational description of the Boundary Scan and TAP controller.

Signal Definitions

JTAG operations such as reset, state-transition control, and clock sampling are handled through the signals listed in Table 11.1. A functional overview of the TAP Controller and Boundary Scan registers is provided in the sections following the table.

Notes

Pin Name	Type	Description
JTAG_TRST_N	Input	JTAG RESET Asynchronous reset for JTAG TAP controller (internal pull-up)
JTAG_TCK	Input	JTAG Clock Test logic clock. JTAG_TMS and JTAG_TDI are sampled on the rising edge. JTAG_TDO is output on the falling edge.
JTAG_TMS	Input	JTAG Mode Select. Requires an external pull-up. Controls the state transitions for the TAP controller state machine (internal pull-up)
JTAG_TDI	Input	JTAG Input Serial data input for BSC chain, Instruction Register, IDCODE register, and BYPASS register (internal pull-up)
JTAG_TDO	Output	JTAG Output Serial data out. Tri-stated except when shifting while in Shift-DR and SHIFT-IR TAP controller states.

Table 11.1 JTAG Pin Descriptions

The TAP controller transitions from state to state, according to the value present on JTAG_TMS, as sampled on the rising edge of JTAG_TCK. The Test-Logic Reset state can be reached either by asserting JTAG_TRST_N or by applying a 1 to JTAG_TMS for five consecutive cycles of JTAG_TCK. A state diagram for the TAP controller appears in Figure 11.2. The value next to state represent the value that must be applied to JTAG_TMS on the next rising edge of JTAG_TCK, to transition in the direction of the associated arrow.

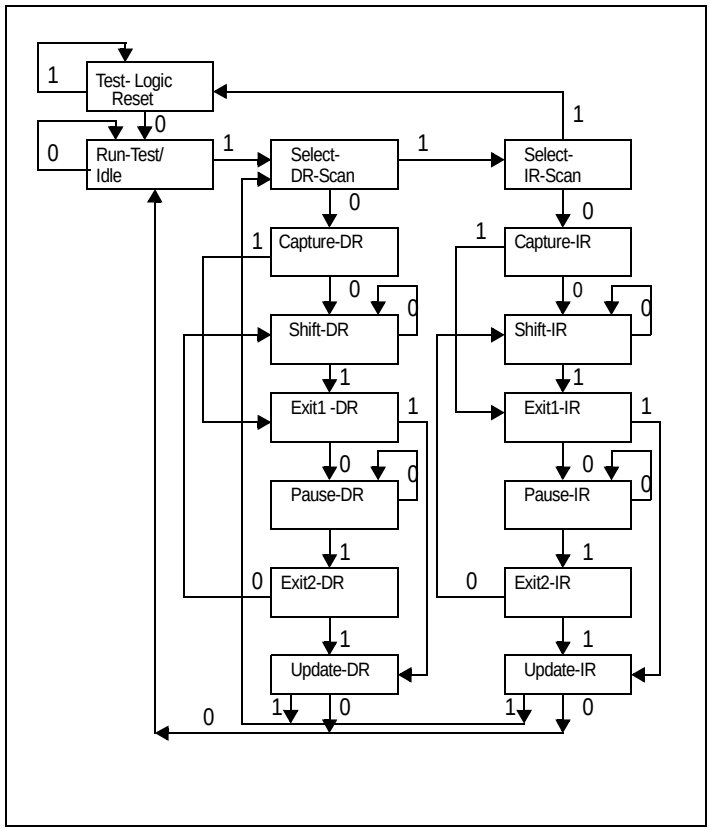

Figure 11.2 State Diagram of PES12N3's TAP Controller

Notes

Boundary Scan Chain

Function	Pin Name	Type	Boundary Cell
PCI Express Interface	PEALREV	I	
	PEARN[7:0]	I	
	PEARP[7:0]	I	
	PEATN[7:0]	O	
	PEATP[7:0]	O	
	PEBLREV	I	
	PEBRN[7:0]	I	
	PEBRP[7:0]	I	
	PEBTN[7:0]	O	
	PEBTP[7:0]	O	
	PECLREV	I	
	PECRN[7:0]	I	
	PECRP[7:0]	I	
	PECTN[7:0]	O	
	PECTP[7:0]	O	
	PEREFCLKN[1:0]	I	
	PEREFCLKP[1:0]	I	
	REFCLKM	I	
	V_{TT}PE	I	
SMBus	MSMBADDR[4:1]	I	
	MSMBCLK	I/O	
	MSMBDAT	I/O	
	SSMBADDR[5,3:1]	I	
	SSMBCLK	I/O	
	SSMBDAT	I/O	
General Purpose I/O	GPIO[7:0]	I/O	
System Pins	CCLKDS	I	
	CCLKUS	I	
	MSMBSMODE	I	
	PERSTN	I	
	RSTHALT	I	
	SWMODE[3:0]	I	
JTAG	JTAG_TCK	I	
	JTAG_TDI	I	
	JTAG_TDO	O	
	JTAG_TMS	I	
	JTAG_TRST_N	I	

Table 11.2 Boundary Scan Chain

Notes

Test Data Register (DR)

The Test Data register contains the following:

- *Bypass register*
- *Boundary Scan registers*
- *Device ID register*

These registers are connected in parallel between a common serial input and a common serial data output and are described in the following sections. For more detailed descriptions, refer to IEEE Standard Test Access Port (IEEE Std. 1149.1).

Boundary Scan Registers

This boundary scan chain is connected between JTAG_TDI and JTAG_TDO when EXTEST or SAMPLE/PRELOAD instructions are selected. Once EXTEST is selected and the TAP controller passes through the UPDATE-IR state, whatever value that is currently held in the boundary scan register's output latches is immediately transferred to the corresponding outputs or output enables.

Therefore, the SAMPLE/PRELOAD instruction must first be used to load suitable values into the boundary scan cells, so that inappropriate values are not driven out onto the system pins. All of the boundary scan cells feature a negative edge latch, which guarantees that clock skew cannot cause incorrect data to be latched into a cell. The input cells are sample-only cells. The simplified logic configuration is shown in Figure 11.3.

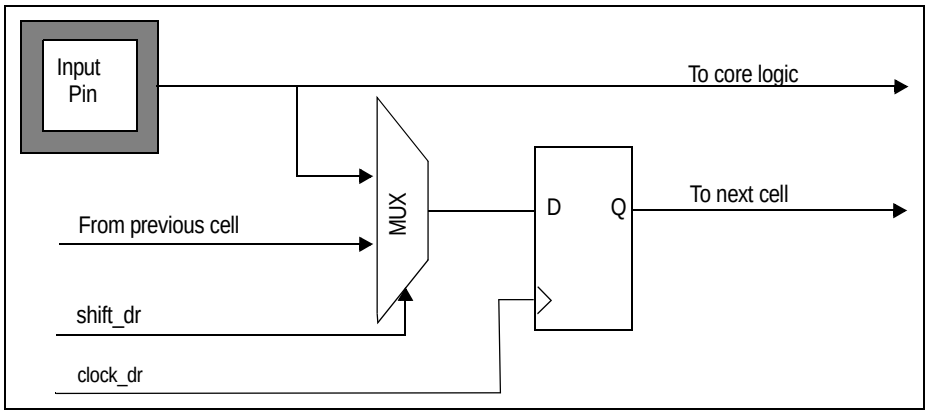

Figure 11.3 Diagram of Observe-only Input Cell

The simplified logic configuration of the output cells is shown in Figure 11.4.

Notes

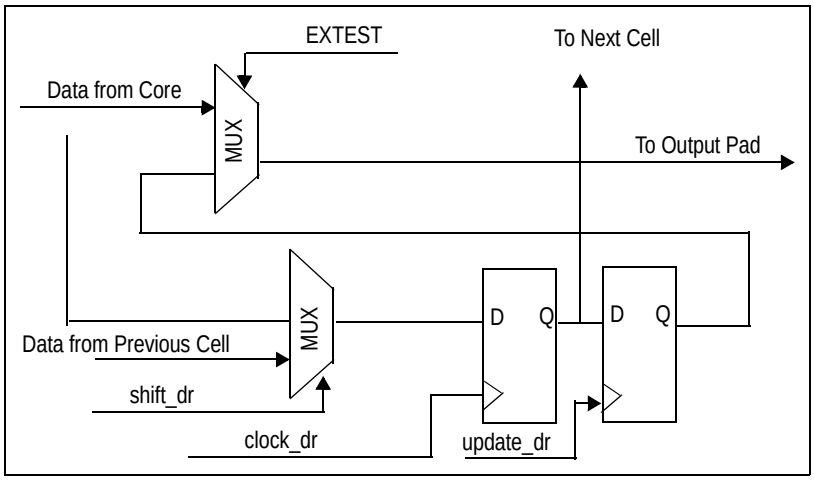

Figure 11.4 Diagram of Output Cell

The output enable cells are also output cells. The simplified logic is shown in Figure 11.5.

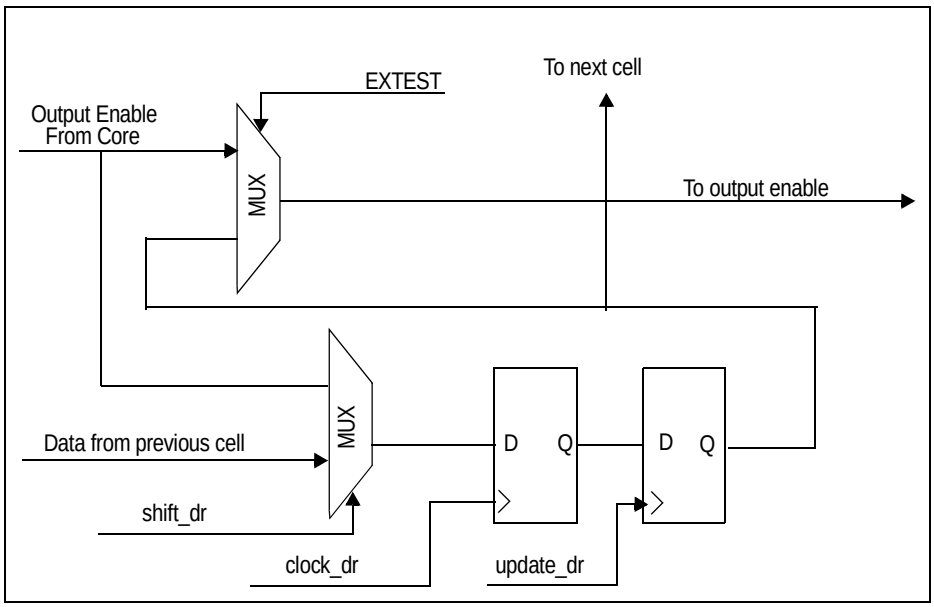

Figure 11.5 Diagram of Output Enable Cell

The bidirectional cells are composed of only two boundary scan cells. They contain one output enable cell and one capture cell, which contains only one register. The input to this single register is selected via a mux that is selected by the output enable cell when EXTEST is disabled. When the Output Enable Cell is driving a high out to the pad (which enables the pad for output) and EXTEST is disabled, the Capture Cell will be configured to capture output data from the core to the pad.

However, in the case where the Output Enable Cell is low (signifying a tri-state condition at the pad) or EXTEST is enabled, the Capture Cell will capture input data from the pad to the core. The configuration is shown graphically in Figure 11.6.

Notes

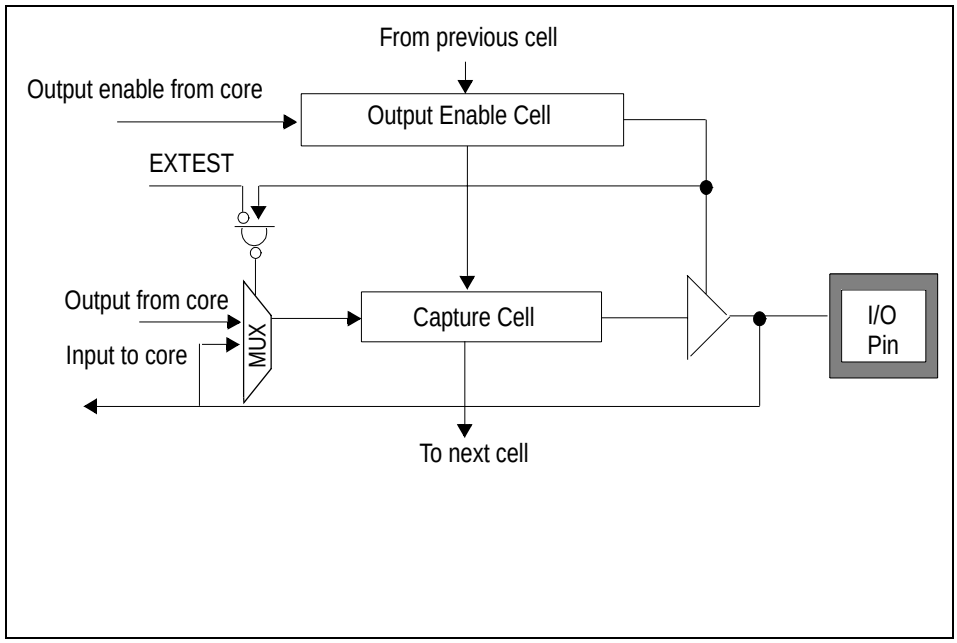

Figure 11.6 Diagram of Bidirectional Cell

Instruction Register (IR)

The Instruction register allows an instruction to be shifted serially into the processor at the rising edge of JTAG_TCK. The instruction is then used to select the test to be performed or the test register to be accessed, or both. The instruction shifted into the register is latched at the completion of the shifting process, when the TAP controller is at the Update-IR state.

The Instruction register contains six shift-register-based cells that can hold instruction data. This register is decoded to perform the following functions:

- To select test data registers that may operate while the instruction is current. The other test data registers should not interfere with chip operation and selected data registers.
- To define the serial test data register path used to shift data between JTAG_TDI and JTAG_TDO during data register scanning.

The Instruction register is comprised of 6 bits to decode instructions, as shown in Table 11.3.

Instruction	Definition	Opcode
EXTEST	Mandatory instruction allowing the testing of board level interconnections. Data is typically loaded onto the latched parallel outputs of the boundary scan shift register using the SAMPLE/PRELOAD instruction prior to use of the EXTEST instruction. EXTEST will then hold these values on the outputs while being executed. Also see the CLAMP instruction for similar capability.	000000
SAMPLE/ PRELOAD	Mandatory instruction that allows data values to be loaded onto the latched parallel output of the boundary scan shift register prior to selection of the other boundary scan test instruction. The Sample instruction allows a snapshot of data flowing from the system pins to the on-chip logic or vice versa.	000001
DEVICE_ID	Provided to select Device Identification to read out manufacturer's identity, part, and version number.	000010

Table 11.3 Instructions Supported by PES12N3's JTAG Boundary Scan (Part 1 of 2)

Notes

Instruction	Definition	Opcode
HIGHZ	Tri-states all output and bidirectional boundary scan cells.	000011
RESERVED	Behaviorally equivalent to the BYPASS instruction as per the IEEE Std. 1149.1 specification. However, the user is advised to use the explicit BTPASS instruction.	000100 — 100011
UNUSED	The unused instructions are behaviorally equivalent to the BYPASS instruction as per the IEEE Std. 1149.1 specification. However, the user is advised to use the explicit BYPASS instruction, as the internal usage of these currently unused instructions could possibly vary in future implementations of the device.	100100 — 101100
VALIDATE	Automatically loaded into the instruction register whenever the TAP controller passes through the CAPTURE-IR state. The lower two bits '01' are mandated by the IEEE Std. 1149.1 specification.	101101
UNUSED	Same as the other UNUSED instruction above.	101110 — 111100
RESERVED	Same as the other RESERVED instruction above.	111101
CLAMP	Provides JTAG users with the option to bypass the part's JTAG controller while keeping the part outputs controlled similar to EXTEST.	111110
BYPASS	The BYPASS instruction is used to truncate the boundary scan register as a single bit in length.	111111

Table 11.3 Instructions Supported by PES12N3's JTAG Boundary Scan (Part 2 of 2)

EXTEST

The external test (EXTEST) instruction is used to control the boundary scan register, once it has been initialized using the SAMPLE/PRELOAD instruction. Using EXTEST, the user can then sample inputs from or load values onto the external pins of the PES12N3. Once this instruction is selected, the user then uses the SHIFT-DR TAP controller state to shift values into the boundary scan chain. When the TAP controller passes through the UPDATE-DR state, these values will be latched onto the output pins or into the output enables.

SAMPLE/PRELOAD

The sample/preload instruction has a dual use. The primary use of this instruction is for preloading the boundary scan register prior to enabling the EXTEST instruction. Failure to preload will result in unknown random data being driven onto the output pins when EXTEST is selected. The secondary function of SAMPLE/PRELOAD is for sampling the system state at a particular moment. Using the SAMPLE function, the user can halt the device at a certain state and shift out the status of all of the pins and output enables at that time.

BYPASS

The BYPASS instruction is used to truncate the boundary scan register to a single bit in length. During system level use of the JTAG, the boundary scan chains of all the devices on the board are connected in series. In order to facilitate rapid testing of a given device, all other devices are put into BYPASS mode. Therefore, instead of having to shift 499 times to get a value through the PES12N3, the user only needs to shift one time to get the value from JTAG_TDI to JTAG_TDO. When the TAP controller passes through the CAPTURE-DR state, the value in the BYPASS register is updated to be 0.

If the device being used does not have a DEVICE_ID register, then the BYPASS instruction will automatically be selected into the instruction register whenever the TAP controller is reset. Therefore, the first value that will be shifted out of a device without a DEVICE_ID register is always 0. Devices such as the PES12N3 that include a DEVICE_ID register will automatically load the DEVICE_ID instruction when the TAP controller is reset, and they will shift out an initial value of 1. This is done to allow the user to easily distinguish between devices having DEVICE_ID registers and those that do not.

Notes

CLAMP

This instruction, listed as optional in the IEEE 1149.1 JTAG Specifications, allows the boundary scan chain outputs to be clamped to fixed values. When the clamp instruction is issued, the scan chain will bypass the PES12N3 and pass through to devices further down the scan chain.

DEVICEID

The DEVICEID instruction is automatically loaded when the TAP controller state machine is reset either by the use of the JTAG_TRST_N signal or by the application of a '1' on JTAG_TMS for five or more cycles of JTAG_TCK as per the IEEE Std. 1149.1 specification. The least significant bit of this value must always be 1. Therefore, if a device has a DEVICE_ID register, it will shift out a 1 on the first shift if it is brought directly to the SHIFT-DR TAP controller state after the TAP controller is reset. The board- level tester can then examine this bit and determine if the device contains a DEVICE_ID register (the first bit is a 1), or if the device only contains a BYPASS register (the first bit is 0).

However, even if the device contains a DEVICE_ID register, it must also contain a BYPASS register. The only difference is that the BYPASS register will not be the default register selected during the TAP controller reset. When the DEVICE_ID instruction is active and the TAP controller is in the Shift-DR state, the thirty-two bit value that will be shifted out of the device-ID register is 0x00022067.

Bit(s)	Mnemonic	Description	R/W	Reset
0	reserved	Reserved	R	0x1
11:1	Manuf_ID	**Manufacturer Identity** (11 bits) This field identifies the manufacturer as IDT.	R	0x33
27:12	Part_number	**Part Number** (16 bits) This field identifies the silicon as PES12N3.	R	0x800C
31:28	Version	**Version** (4 bits) This field identifies the silicon revision of the PES12N3.	R	silicon-dependent

Table 11.4 System Controller Device Identification Register

Version	Part Number	Mnfg. ID	LSB
xxxx	1000|0000|0000|1100	0000|0110|011	1

Figure 11.7 Device ID Register Format

VALIDATE

The VALIDATE instruction is automatically loaded into the instruction register whenever the TAP controller passes through the CAPTURE-IR state. The lower two bits '01' are mandated by the IEEE Std. 1149.1 specification.

RESERVED

Reserved instructions implement various test modes used in the device manufacturing process. The user should not enable these instructions.

UNUSED[1]

The unused instructions are behaviorally equivalent to the BYPASS instruction as per the IEEE Std. 1149.1 specification. However, the user is advised to use the explicit BYPASS instruction as the internal usage of these currently unused instructions could possibly vary in future implementations of the device.

[1] Any unused instruction is defaulted to the BYPASS instruction

Notes

Usage Considerations

As previously stated, there are internal pull-ups on JTAG_TRST_N, JTAG_TMS, and JTAG_TDI. However, JTAG_TCK also needs to be driven to a known value. It is best to either drive a zero on the JTAG_TCK pin when it is not being used or to use an external pull-down resistor. In order to guarantee that the JTAG does not interfere with normal system operation, the TAP controller should be forced into the Test-Logic-Reset controller state by continuously holding JTAG_TRST_N low and/or JTAG_TMS high when the chip is in normal operation. If JTAG will not be used, externally pull-down JTAG_TRST_N low to disable it.

Notes

IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD-PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers who are designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only to develop an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third-party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising from your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Disclaimer Rev.1.01 Jan 2024)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit www.renesas.com/contact-us/.

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.