IDT® 89HPES48T12G2 PCI Express® Switch

User Manual

April 2013

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.

About This Manual

Notes

Introduction

This user manual includes hardware and software information on the 89HPES48T12G2, 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, "PES48T12G2 Device Overview," provides a complete introduction to the performance capabilities of the 89HPES48T12G2. Included in this chapter is a summary of features for the device as well as a system block diagram and pin description.

Chapter 2, "Architectural Overview," provides a high level architectural overview of the PES48T12G2 device.

Chapter 3, "Switch Core," provides a description of the PES48T12G2 switch core.

Chapter 4, "Clocking," provides a description of the PES48T12G2 clocking architecture.

Chapter 5, "Reset and Initialization," describes the PES48T12G2 reset operations and initialization procedure.

Chapter 6, "Link Operation," describes the operation of the link feature including polarity inversion, link width negotiation, and lane reversal.

Chapter 7, "SerDes," describes basic functionality and controllability associated with the Serialiazer-Deserializer (SerDes) block in PES48T12G2 ports.

Chapter 7, "Theory of Operation," describes the general operational behavior of the PES48T12G2.

Chapter 9, "Hot-Plug and Hot-Swap," describes the behavior of the hot-plug and hot-swap features in the PES48T12G2.

Chapter 10, "Power Management," describes the power management capability structure located in the configuration space of each PCI-to-PCI bridge in the PES48T12G2.

Chapter 11, "General Purpose I/O," describes how the 9 General Purpose I/O (GPIO) pins may be individually configured as general purpose inputs, general purpose outputs, or alternate functions.

Chapter 12, "SMBus Interfaces," describes the operation of the 2 SMBus interfaces on the PES48T12G2.

Chapter 13, "Multicast," describes how the multicast capability enables a single TLP to be forwarded to multiple destinations.

Chapter 14, "Register Organization," describes the organization of all the software visible registers in the PES48T12G2 and provides the address space for those registers.

Chapter 15, "PCI to PCI Bridge and Proprietary Port Specific Registers," lists the Type 1 configuration header registers in the PES48T12G2 and provides a description of each bit in those registers.

Chapter 16, "Switch Control and Status Registers," lists the switch control and status registers in the PES48T12G2 and provides a description of each bit in those registers.

Notes

Chapter 17, "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.

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.

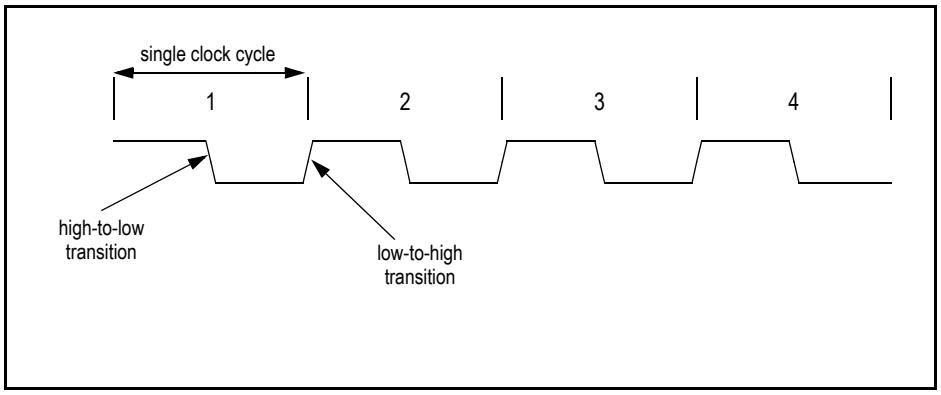


Figure 1 Signal Transitions

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

Notes

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.

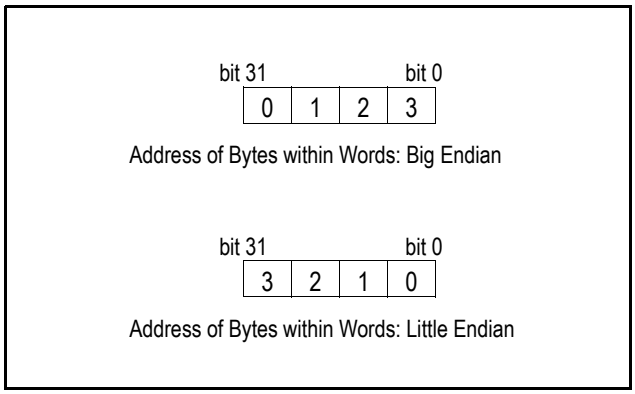


Figure 2 Example of Byte Ordering for “Big Endian” or “Little Endian” System Definition

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.

Table 2 Register Terminology (Sheet 1 of 2)

Notes

Type	Abbreviation	Description
Read Only	RO	Software can only read registers/bits with this attribute. Contents are hardwired to a constant value or are status bits that may be set and cleared by hardware. Writing to a RO location has no effect.
Read and Write	RW	Software can both read and write bits with this attribute.
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 SWCTL register is set. When the REGUNLOCK bit is cleared, writes are ignored and the register/bits are effectively read-only. Fields with this attribute are implicitly SWSticky (i.e., their value is preserved across all resets, except switch fundamental reset).
Sticky	Sticky	Register/bits with this designation take on their initial value as a result of a switch fundamental reset or fundamental reset. Other resets have no effect.
Switch Sticky	SWSticky	Register/bits with this designation take on their initial value as a result of a switch fundamental reset. Other resets have no effect.

Table 2 Register Terminology (Sheet 2 of 2)

Use of Hypertext

In Chapter 14, Tables 14.4, 14.5 and 14.6 contain register names and page numbers highlighted in blue under the Register Definition column. In pdf files, users can jump from this 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 Chapters and 16. 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.

Reference Documents

[1] PCI Express Base Specification Revision 2.0., December 20, 2006, PCI-SIG.

[2] Multicast Engineering Change Notice to [1]., May 8, 2008, PCI-SIG.

[3] Internal Error Reporting Engineering Change Notice to [1]., April 24, 2008, PCI-SIG.

[4] SMBus Specification, Version 2.0, August 3, 2000, SBS Implementers Forum.

Revision History

November 5, 2008: Initial publication of preliminary user manual.

January 12, 2009: On page 3-6, added last sentence to Port Arbitration section on page 3-6. In Table 8.10, under Description for Function in D3Hot state, changed reference to 10-1 instead of 9-1.

January 22, 2009: In Chapter 12, Table 12.15, changed the description for bit USA. In Chapter 15, PCIEDCTL register, changed the description for bit ERO.

Notes

February 9, 2009: In Chapter 1: Table 1.4, for Port 0 Serial Data Receive/Transmit signals, deleted statement that port 0 is the upstream port; Table 1.8, revised Description for SWMODE[3:0]; Table 1.10, added "3.3V is preferred" for signal V_{DD}I/O. In Chapter 6, deleted footnote in 2nd paragraph under section Software Management of Link Speed.

February 18, 2009: In Chapter 6, added a note under L2/L3 Ready in Link States section. In Chapter 15, modified Description for REG and EREG fields in the ECFGADDR register, GADDR field in the GASAADDR register, DATA field in the GASADATA register, and RSE field in the SECSTS register. In Chapters 14 and 15, added PHYLSTATE0 (0x540) register.

March 18, 2009: In Table 12.11, the address value was changed from zero to 1 for bits 3 and 5.

April 9, 2009: Changes made in register fields in Chapters 15 and 16 (Bridge and Switch Registers), to conform with device specification and validation.

April 21, 2009: In Table 1.8, deleted reference to pull-down value of 251K ohm resistor for all PxMERGEN pins. In Footnote 1 for Table 1.11, internal resistor pull-down value was changed to 91K ohms. In Chapter 16, changed "Bit x in this field corresponds to GPIO pin (x+31)" to "Bit x in this field corresponds to GPIO pin (x+32)" in the GPIOFUNC1, GPIOCFG1, and GPIOD1 registers. Changed title for Table 12.11.

April 27, 2009: ZB silicon was added to Table 1.3.

May 6, 2009: In Chapter 5, under section Switch Fundamental Reset, deleted bullet referencing SWFRST bit.

May 14, 2009: In Table 1.11, changed CML to HCSL for PCIe reference clocks.

May 28, 2009: In Chapter 6, revised Crosslink section. In Chapter 7, Tables 7.2, 7.3 and 7.4, changed column title of TX_EQ_MODE to reflect the register field used to control TX equalization depending on the operating mode of the link (e.g., TX_EQ_3DBG1). In Chapter 12, revised Introduction section and deleted references to LAERR bit in Table 12.2, Table 12.15, and Figure 12.8. In Chapter 14, added section Partial-Byte Access to Word and DWord Registers. In Chapter 16, added bit BDISCARD to the Switch Control register and changed bit 26 in the SMBus Status register from LAERR to Reserved.

June 16, 2009: In Chapters 5 and 6, added footnote explaining LTSSM reference. In Chapter 16, removed reference to RDETECT bit in Hot-Plug Configuration Control register.

June 22, 2009: In Table 1.11, System Pins section, changed GCLKFSEL to pull-down.

July 30, 2009: In Chapter 16, Switch Registers, changed bits 19:18 in the SMBus Control register from SSMBMODE to Reserved.

August 17, 2009: In Chapter 16, Switch Registers, revised the description for the RXEQZ and RXEQB fields in the SerDes x Receiver Equalization Lane Control register.

September 22, 2009: Modified Chapter 4, Clocking. In Chapter 7, SerDes, modified Table 7.2 and added Note before Figure 7.1. Modified section Transaction Layer Error Pollution in Chapter 9, Theory of Operation. In Chapter 16, Switch Registers, modified description of the LANESEL field in the SxCTL register and modified description of the RXEQZ and RXEQB fields in the SxRXEQLCTL register.

September 28, 2009: ZC silicon was added to Table 1.3.

November 6, 2009: In Chapter 3, Switch Core, modified text and figures in Operation section. In Chapter 4, Clocking, modified Introduction section. In Chapter 14, Register Organization, added new section Register Side-Effects. In Chapter 15, Bridge Registers, modified description for DVADJ bit in the Requester Metering Control register.

November 11, 2009: In Chapter 7, SerDes, deleted settings greater than 0x0F in Tables 7.7 and 7.8.

December 7, 2009: In Chapter 6, added reference in section Link width Negotiation to the MAXLNK-WDTH field in the PCI Express Link Capabilities register. In Chapter 14, added new sub-section Limitations under Register Side-Effects. In Chapter 15, modified Description for the MAXLNKWDTH field in the PCIELCAP register and added field RCVD_OVRD to the SerDes Configuration register. In Chapter 16, added field DDDNC to the Switch Control register and modified Description for the BLANK field in the SMBus Status register.

Notes

December 14, 2009: Deleted all references to support for Weighted Round Robin arbitration.

January 21, 2010: Removed Preliminary from title.

February 10, 2010: In Chapter 5, added new Port Merging section. In Table 1.8, added reference to Port Merging section in PxxMERGEN pin description.

March 31, 2010: In Chapter 14, Table 14.6, added the following register names and cross-references for ports 8, 9, 12, 13: SWPORTxCTL, SWPORTxSTS, SxCTL, SxTXLCTL0, SxTXLCTL1, SxRXEQLCTL.

December 8, 2010: In Chapter 13, corrected ports specified for I/O Expander 10 in Table 12.3. In Chapter 17, deleted PERSTN, GLK1, and SMODE from Table 17.1.

February 2, 2011: In Table 8.13, revised text in Action Taken column for ACS Source Validation. In Chapter 15, added footnote to STAS bit in PCISTS and SECSTS registers.

May 18, 2011: In Chapter 7, section Low-Swing Transmitter Voltage Mode, the reference in the first paragraph to the LSE bit being in the SerDes Control register was changed to the SerDes Configuration register.

June 28, 2011: In Chapter 16, added bit 26, TX_SLEW_C, to the SerDes x Transmitter Lane Control 0 register.

July 8, 2011: In Chapter 15, removed table footnotes from PCISTS and SECSTS registers, added Reserved bits 31:24 to AERUEM and AERUESV registers, and added last sentence to each description in the PCIESCTLIV register. In Chapter 16, added FEN and FCAPSEL fields to SWPART[x]CTL register and SWPORT[x]CTL registers, added PFAILOVER and SFAILOVER fields to SWPART[X]STS register and SWPORT[x]STS register, adjusted bit fields in GPIOCFG1 and GPIOD1 registers.

August 31, 2011: In Chapter 2, page 2-1, added bullet to explain behavior of an odd numbered port when it is merged with its even counterpart. In Chapter 6, revised text in section Link Width Negotiation in the Presence of Bad Lanes. In Chapter 7, revised Table 7.1 and text under this table, revised text in section Programmable De-emphasis Adjustment, added headings to Figures 7.1 through 7.3, and added paragraph after Figure 7.3. In Chapter 13, revised text in the Introduction section. In Chapter 15, changed type of MAXLNKSPD field in the PCIELCAP register from RWL to RO, revised Description for MAXGROUP field in MCCAP register, changed lower to upper in Description for MCBLKALLH and MCBLKUTH registers. In Chapter 17, deleted references to Failover capability from several registers.

September 9, 2011: In Chapter 7, added additional reference in last paragraph of section Driver Voltage Level and Amplitude Boost.

February 7, 2012: In Chapter 12, added footnote for RERR and WERR bits in Table 12.13.

February 23, 2012: Added paragraph after Table 12.13 to explain use of DWord addresses.

January 31, 2013: In Figure 12.8, changed No-ack to Ack between DATALM and DATAUM.

April 5, 2013: In Chapter 16, added USSBRDELAY register.

Table of Contents

Notes

Notes

Clocking

Reset and Initialization

Link Operation

SerDes

Notes

Theory of Operation

Hot-Plug and Hot-Swap

Power Management

General Purpose I/O

SMBus Interfaces

Notes

Multicast

Register Organization

PCI to PCI Bridge and Proprietary Port Specific Registers

Switch Configuration and Status Registers

Notes

JTAG Boundary Scan

Notes

List of Tables

Notes

Notes

List of Figures

Notes

Notes

Register List

Notes

Notes

Notes

Notes

PES48T12G2 Device Overview

Notes

Introduction

The 89HPES48T12G2 is a member of the IDT PRECISE™ family of PCI Express® switching solutions. The PES48T12G2 is a 48-lane, 12-port system interconnect switch optimized for PCI Express Gen2 packet switching in high-performance applications, supporting multiple simultaneous peer-to-peer traffic flows. Target applications include servers, storage, communications, embedded systems, and multi-host or intelligent I/O based systems with inter-domain communication.

Utilizing standard PCI Express interconnect, the PES48T12G2 provides the most efficient fan-out solution for applications requiring high throughput, low latency, and simple board layout with a minimum number of board layers. It provides 48 GBps (384 Gbps) of aggregated, full-duplex switching capacity through 48 integrated serial lanes, using proven and robust IDT technology. Each lane provides 5 Gbps of bandwidth in both directions and is fully compliant with PCI Express Base Specification, Revision 2.0.

Features

- **High Performance Non-Blocking Switch Architecture**
  - 48-lane 12-port PCIe switch
    - *Six x8 ports switch ports each of which can bifurcate to two x4 ports (total of twelve x4 ports)*
  - Integrated SerDes supports 5.0 GT/s Gen2 and 2.5 GT/s Gen1 operation
  - Delivers up to *48 GBps (384 Gbps)* of switching capacity
  - Supports 128 Bytes to 2 KB maximum payload size
  - Low latency cut-through architecture
  - Supports one virtual channel and eight traffic classes
- **Standards and Compatibility**
  - PCI Express Base Specification 2.0 compliant
  - Implements the following optional PCI Express features
    - *Advanced Error Reporting (AER) on all ports*
    - *End-to-End CRC (ECRC)*
    - *Access Control Services (ACS)*
    - *Power Budgeting Enhanced Capability*
    - *Device Serial Number Enhanced Capability*
    - *Sub-System ID and Sub-System Vendor ID Capability*
    - *Internal Error Reporting ECN*
    - *Multicast ECN*
    - *VGA and ISA enable*
    - *L0s and L1 ASPM*
    - *ARI ECN*
- **Port Configurability**
  - x4 and x8 ports
    - *Ability to merge adjacent x4 ports to create a x8 port*
  - Automatic per port link width negotiation (x8 → x4 → x2 → x1)
  - Crosslink support
  - Automatic lane reversal
  - Autonomous and software managed link width and speed control
  - Per lane SerDes configuration

Notes

- *De-emphasis*
- *Receive equalization*
- *Drive strength*

◆ **Initialization / Configuration**

– Supports Root (BIOS, OS, or driver), Serial EEPROM, or SMBus switch initialization
– Common switch configurations are supported with pin strapping (no external components)
– Supports in-system Serial EEPROM initialization/programming

◆ **Quality of Service (QoS)**

– Port arbitration
  - *Round robin*
  - *Weighted Round Robin (WRR)*
– Request metering
  - *IDT proprietary feature that balances bandwidth among switch ports for maximum system throughput*
– High performance switch core architecture
  - *Combined Input Output Queued (CIOQ) switch architecture with large buffers*

◆ **Multicast**

– Compliant to the PCI-SIG multicast ECN
– Supports arbitrary multicasting of Posted transactions
– Supports 64 multicast groups
– Multicast overlay mechanism support
– ECRC regeneration support

◆ **Clocking**

– Supports 100 MHz and 125 MHz reference clock frequencies
– Flexible clocking modes
  - *Common clock*
  - *Non-common clock*

◆ **Hot-Plug and Hot Swap**

– Hot-plug controller on all ports
  - *Hot-plug supported on all downstream switch ports*
– All ports support hot-plug using low-cost external I^2C I/O expanders
– Configurable presence detect supports card and cable applications
– GPE output pin for hot-plug event notification
  - *Enables SCI/SMI generation for legacy operating system support*
– Hot swap capable I/O

◆ **Power Management**

– Supports D0, D3hot and D3 power management states
– Active State Power Management (ASPM)
  - *Supports L0, L0s, L1, L2/L3 Ready and L3 link states*
  - *Configurable L0s and L1 entry timers allow performance/power-savings tuning*
– Supports PCI Express Power Budgeting Capability
– SerDes power savings
  - *Supports low swing / half-swing SerDes operation*
  - *SerDes optionally turned-off in D3hot*
  - *SerDes associated with unused ports are turned-off*
  - *SerDes associated with unused lanes are placed in a low power state*

◆ **9 General Purpose I/O**

◆ **Reliability, Availability and Serviceability (RAS)**

– ECRC support
– AER on all ports

Notes

- SECDED ECC protection on all internal RAMs
- End-to-end data path parity protection
- Checksum Serial EEPROM content protected
- Autonomous link reliability (preserves system operation in the presence of faulty links)
- Ability to generate an interrupt (INTx or MSI) on link up/down transitions

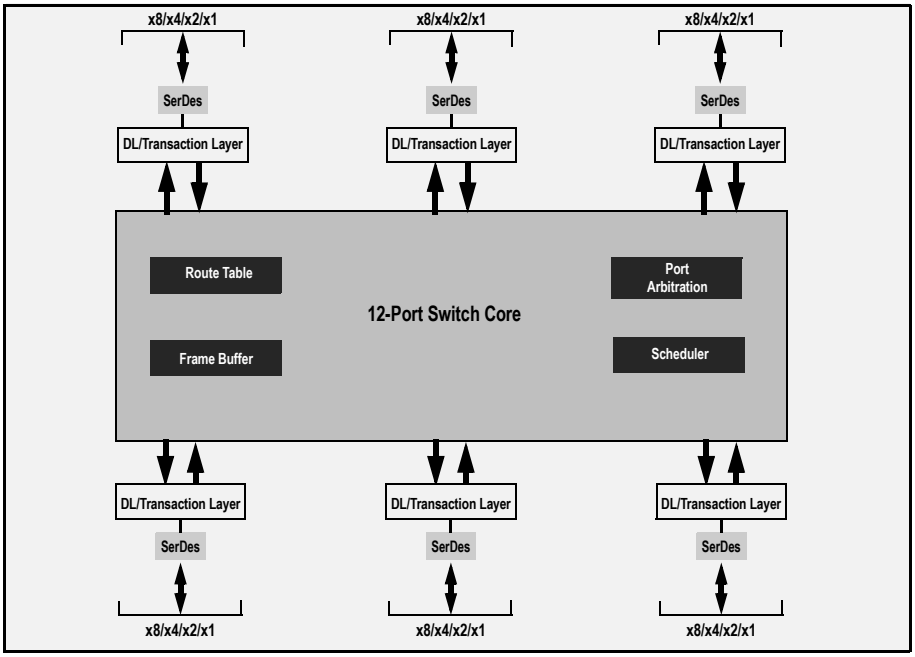
◆ **Test and Debug**

- On-chip link activity and status outputs available for Port 0 (upstream port)
- Per port link activity and status outputs available using external I^2C I/O expander for all other ports
- SerDes test modes
- Supports IEEE 1149.6 AC JTAG and IEEE 1149.1 JTAG

◆ **Power Supplies**

- Requires only two power supply voltages (1.0 V and 2.5 V)
- No power sequencing requirements

◆ **Packaged in a 27mm x 27mm 676-ball Flip Chip BGA with 1mm ball spacing**



48 PCI Express Lanes
Up to 6 x8 ports or 12 x4 Ports

Figure 1.1 PES48T12G2 Block Diagram

Port	PCIELCAP MAXLNKWDTH
0	0x4
1	0x4
2	0x4
3	0x4
4	0x4

Table 1.1 Initial Configuration Register Settings for PES48T12G2 (Part 1 of 2)

Notes

Port	PCIELCAP MAXLNKWDTH
5	0x4
6	0x4
7	0x4
8	0x4
9	0x4
12	0x4
13	0x4

Table 1.1 Initial Configuration Register Settings for PES48T12G2 (Part 2 of 2)

Note: There are no ports 10 and 11 in the PES48T12G2 device.

Logic Diagram

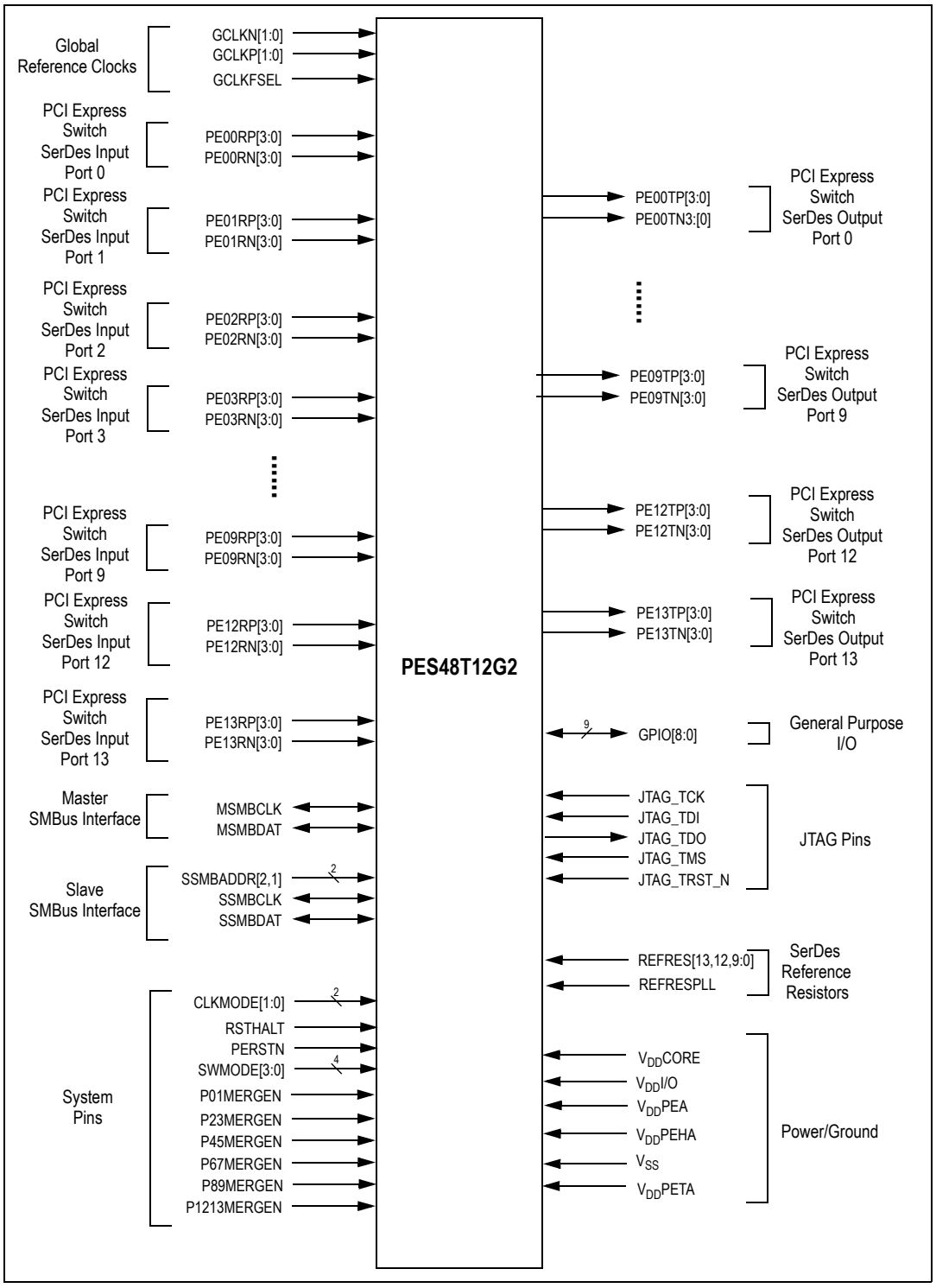


Figure 1.2 PES48T12G2 Logic Diagram

Notes

System Identification

Vendor ID

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

Device ID

The PES48T12G2 device ID is shown in Table 1.2.

PCIe Device	Device ID
0x11	0x807B

Table 1.2 PES48T12G2 Device IDs

Revision ID

The revision ID in the PES48T12G2 is set to the same value in all mode. The value of the revision ID is determined in one place and is easily modified during a metal mask change. The revision ID will start at 0x0 and will be incremented with each all-layer or metal mask change.

Revision ID	Description
0x0	Corresponds to ZA silicon
0x1	Corresponds to ZB silicon
0x2	Corresponds to ZC silicon

Table 1.3 PES48T12G2 Revision ID

JTAG ID

The JTAG ID is:

- Version: Same value as Revision ID. See Table 1.3
- Part number: Same value as base Device ID. See Table 1.2.
- Manufacture ID: 0x33
- LSB: 0x1

SSID/SSVID

The PES48T12G2 contains the mechanisms necessary to implement the PCI-to-PCI bridge Subsystem ID and Subsystem Vendor ID capability structure. However, in the default configuration the Subsystem ID and Subsystem Vendor ID capability structure is not enabled. To enable this capability, the SSID and SSVID fields in the Subsystem ID and Subsystem Vendor ID (SSIDSSVID) register must be initialized with the appropriate ID values. the Next Pointer (NXTPTR) field in one of the other enhanced capabilities should be initialized to point to this capability. Finally, the Next Pointer (NXTPTR) of this capability should be adjusted to point to the next capability if necessary.

Device Serial Number Enhanced Capability

The PES48T12G2 contains the mechanisms necessary to implement the PCI express device serial number enhanced capability. However, in the default configuration this capability structure is not enabled. To enable the device serial number enhanced capability, the Serial Number Lower Doubleword (SNUMLDW) and the Serial Number Upper Doubleword (SNUMUDW) registers should be initialized. The Next Pointer (NXTPTR) field in one of the other enhanced capabilities should be initialized to point to this capability. Finally, the Next Pointer (NXTPTR) of this capability should be adjusted to point to the next capability if necessary.

Notes

Pin Description

The following tables list the functions of the pins provided on the PES48T12G2. 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. Differential signals end with a suffix "N" or "P." The differential signal ending in "P" is the positive portion of the differential pair and the differential signal ending in "N" is the negative portion of the differential pair.

Signal	Type	Name/Description
PE00RP[3:0] PE00RN[3:0]	I	**PCI Express Port 0 Serial Data Receive.** Differential PCI Express receive pairs for port 0.
PE00TP[3:0] PE00TN[3:0]	O	**PCI Express Port 0 Serial Data Transmit.** Differential PCI Express transmit pairs for port 0.
PE01RP[3:0] PE01RN[3:0]	I	**PCI Express Port 1 Serial Data Receive.** Differential PCI Express receive pairs for port 1. When port 0 is merged with port 1, these signals become port 0 receive pairs for lanes 4 through 7.
PE01TP[3:0] PE01TN[3:0]	O	**PCI Express Port 1 Serial Data Transmit.** Differential PCI Express transmit pairs for port 1. When port 0 is merged with port 1, these signals become port 0 transmit pairs for lanes 4 through 7.
PE02RP[3:0] PE02RN[3:0]	I	**PCI Express Port 2 Serial Data Receive.** Differential PCI Express receive pairs for port 2.
PE02TP[3:0] PE02TN[3:0]	O	**PCI Express Port 2 Serial Data Transmit.** Differential PCI Express transmit pairs for port 2.
PE03RP[3:0] PE03RN[3:0]	I	**PCI Express Port 3 Serial Data Receive.** Differential PCI Express receive pairs for port 3. When port 2 is merged with port 3, these signals become port 2 receive pairs for lanes 4 through 7.
PE03TP[3:0] PE03TN[3:0]	O	**PCI Express Port 3 Serial Data Transmit.** Differential PCI Express transmit pairs for port 3. When port 2 is merged with port 3, these signals become port 2 transmit pairs for lanes 4 through 7.
PE04RP[3:0] PE04RN[3:0]	I	**PCI Express Port 4 Serial Data Receive.** Differential PCI Express receive pairs for port 4.
PE04TP[3:0] PE04TN[3:0]	O	**PCI Express Port 4 Serial Data Transmit.** Differential PCI Express transmit pairs for port 4.
PE05RP[3:0] PE05RN[3:0]	I	**PCI Express Port 5 Serial Data Receive.** Differential PCI Express receive pairs for port 5. When port 4 is merged with port 5, these signals become port 4 receive pairs for lanes 4 through 7.
PE05TP[3:0] PE05TN[3:0]	O	**PCI Express Port 5 Serial Data Transmit.** Differential PCI Express transmit pairs for port 5. When port 4 is merged with port 5, these signals become port 4 transmit pairs for lanes 4 through 7.
PE06RP[3:0] PE06RN[3:0]	I	**PCI Express Port 6 Serial Data Receive.** Differential PCI Express receive pairs for port 6.
PE06TP[3:0] PE06TN[3:0]	O	**PCI Express Port 6 Serial Data Transmit.** Differential PCI Express transmit pairs for port 6.
PE07RP[3:0] PE07RN[3:0]	I	**PCI Express Port 7 Serial Data Receive.** Differential PCI Express receive pairs for port 7. When port 6 is merged with port 7, these signals become port 6 receive pairs for lanes 4 through 7.

Table 1.4 PCI Express Interface Pins (Part 1 of 2)

Notes

Signal	Type	Name/Description
PE07TP[3:0] PE07TN[3:0]	O	**PCI Express Port 7 Serial Data Transmit.** Differential PCI Express transmit pairs for port 7. When port 6 is merged with port 7, these signals become port 6 transmit pairs for lanes 4 through 7.
PE08RP[3:0] PE08RN[3:0]	I	**PCI Express Port 8 Serial Data Receive.** Differential PCI Express receive pairs for port 8.
PE08TP[3:0] PE08TN[3:0]	O	**PCI Express Port 8 Serial Data Transmit.** Differential PCI Express transmit pairs for port 8.
PE09RP[3:0] PE09RN[3:0]	I	**PCI Express Port 9 Serial Data Receive.** Differential PCI Express receive pairs for port 9. When port 8 is merged with port 9, these signals become port 8 receive pairs for lanes 4 through 7.
PE09TP[3:0] PE09TN[3:0]	O	**PCI Express Port 9 Serial Data Transmit.** Differential PCI Express transmit pairs for port 9. When port 8 is merged with port 9, these signals become port 8 transmit pairs for lanes 4 through 7.
PE12RP[3:0] PE12RN[3:0]	I	**PCI Express Port 12 Serial Data Receive.** Differential PCI Express receive pairs for port 12.
PE12TP[3:0] PE12TN[3:0]	O	**PCI Express Port 12 Serial Data Transmit.** Differential PCI Express transmit pairs for port 12.
PE13RP[3:0] PE13RN[3:0]	I	**PCI Express Port 13 Serial Data Receive.** Differential PCI Express receive pairs for port 13. When port 12 is merged with port 13, these signals become port 12 receive pairs for lanes 4 through 7.
PE13TP[3:0] PE13TN[3:0]	O	**PCI Express Port 13 Serial Data Transmit.** Differential PCI Express transmit pairs for port 13. When port 12 is merged with port 13, these signals become port 12 transmit pairs for lanes 4 through 7.

Table 1.4 PCI Express Interface Pins (Part 2 of 2)

Signal	Type	Name/Description
GCLKN[1:0] GCLKP[1:0]	I	**Global Reference Clock.** Differential reference clock input pair. This clock is used as the reference clock by on-chip PLLs to generate the clocks required for the system logic. The frequency of the differential reference clock is determined by the GCLKFSEL signal.

Table 1.5 Reference Clock Pins

Signal	Type	Name/Description
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[2,1]	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 1.6 SMBus Interface Pins

Notes

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.
GPIO[3]	I/O	General Purpose I/O. This pin can be configured as a general purpose I/O pin.
GPIO[4]	I	General Purpose I/O. This pin can be configured as a general purpose I/O pin. 1st Alternate function — Reserved 2nd Alternate function pin name: P0LINKUPN 2nd Alternate function pin type: Output 2nd Alternate function: Port 0 Link Up Status output.
GPIO[5]	O	General Purpose I/O. This pin can be configured as a general purpose I/O pin. 1st Alternate function pin name: GPEN 1st Alternate function pin type: Output 1st Alternate function: Hot-plug general purpose even output. 2nd Alternate function pin name: P0ACTIVEN 2nd Alternate function pin type: Output 2nd Alternate function: Port 0 Link Active Status Output.
GPIO[6]	I	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.
GPIO[8]	I	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: IO expander interrupt.

Table 1.7 General Purpose I/O Pins

Signal	Type	Name/Description
CLKMODE[1:0]		**Clock Mode.** These signals determine the port clocking mode used by ports of the device.
GCLKFSEL	I	**Global Clock Frequency Select.** These signals select the frequency of the GCLKP and GCLKN signals. 0x0 100 MHz 0x1 125 MHz
P01MERGEN	I	**Port 0 and 1 Merge.** P01MERGEN is an active low signal. It is pulled low internally. When this pin is low, port 0 is merged with port 1 to form a single x8 port. The Serdes lanes associated with port 1 become lanes 4 through 7 of port 0. Refer to section Port Merging on page 5-6 for details. When this pin is high, port 0 and port 1 are not merged, and each operates as a single x4 port.

Table 1.8 System Pins (Part 1 of 3)

Notes

Signal	Type	Name/Description
P23MERGEN	I	**Port 2 and 3 Merge.** P23MERGEN is an active low signal. It is pulled low internally. When this pin is low, port 2 is merged with port 3 to form a single x8 port. The Serdes lanes associated with port 3 become lanes 4 through 7 of port 2. Refer to section Port Merging on page 5-6 for details. When this pin is high, port 2 and port 3 are not merged, and each operates as a single x4 port.
P45MERGEN	I	**Port 4 and 5 Merge.** P45MERGEN is an active low signal. It is pulled low internally. When this pin is low, port 4 is merged with port 5 to form a single x8 port. The Serdes lanes associated with port 5 become lanes 4 through 7 of port 4. Refer to section Port Merging on page 5-6 for details. When this pin is high, port 4 and port 5 are not merged, and each operates as a single x4 port.
P67MERGEN	I	**Port 6 and 7 Merge.** P67MERGEN is an active low signal. It is pulled low internally. When this pin is low, port 6 is merged with port 7 to form a single x8 port. The Serdes lanes associated with port 7 become lanes 4 through 7 of port 6. Refer to section Port Merging on page 5-6 for details. When this pin is high, port 6 and port 7 are not merged, and each operates as a single x4 port.
P89MERGEN	I	**Port 8 and 9 Merge.** P89MERGEN is an active low signal. It is pulled low internally. When this pin is low, port 8 is merged with port 9 to form a single x8 port. The Serdes lanes associated with port 9 become lanes 4 through 7 of port 8. Refer to section Port Merging on page 5-6 for details. When this pin is high, port 8 and port 9 are not merged, and each operates as a single x4 port.
P1213MERGEN	I	**Port 12 and 13 Merge.** P1213MERGEN is an active low signal. It is pulled low internally. When this pin is low, port 12 is merged with port13 to form a single x8 port. The Serdes lanes associated with port 13 become lanes 4 through 7 of port 12. Refer to section Port Merging on page 5-6 for details. When this pin is high, port 12 and port 13 are not merged, and each operates as a single x4 port.

Table 1.8 System Pins (Part 2 of 3)

Notes

Signal	Type	Name/Description
PERSTN	I	**Global Reset.** Assertion of this signal resets all logic inside PES48T12G2.
RSTHALT	I	**Reset Halt.** When this signal is asserted during a PCI Express fundamental reset, PES48T12G2 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 SWCTL register by an SMBus master.
SWMODE[3:0]	I	**Switch Mode.** These configuration pins determine the PES48T12G2 switch operating mode. **Note:** These pins should be static and not change following the negation of PERSTN. 0x0 - Normal switch mode 0x1 - Normal switch mode with Serial EEPROM initialization 0x2 through 0x7 - Reserved 0x8 - Single partition with port 0 selected as the upstream port (port 2 disabled) 0x9 - Single partition with port 2 selected as the upstream port (port 0 disabled) 0xA - Single partition with Serial EEPROM initialization and port 0 selected as the upstream port (port 2 disabled) 0xB - Single partition with Serial EEPROM initialization and port 2 selected as the upstream port (port 0 disabled) 0xE - Reserved 0xF - Reserved

Table 1.8 System Pins (Part 3 of 3)

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 1.9 Test Pins

Notes

Signal	Type	Name/Description
REFRES[13,12,9:0]	I/O	**External Reference Resistors.** Provides a reference for the SerDes bias currents and PLL calibration circuitry. A 3 kOhm +/- 1% resistor should be connected from these pins to ground.
REFRESPLL	I/O	**PLL External Reference Resistor.** Provides a reference for the PLL bias currents and PLL calibration circuitry. A 3K Ohm +/- 1% resistor should be connected from this pin to ground.
V_{DD}CORE	I	**Core V_{DD}.** Power supply for core logic (1.0V).
V_{DD}I/O	I	**I/O V_{DD}.** LVTTL I/O buffer power supply (2.5V or preferred 3.3V).
V_{DD}PEA	I	**PCI Express Analog Power.** Serdes analog power supply (1.0V).
V_{DD}PEHA	I	**PCI Express Analog High Power.** Serdes analog power supply (2.5V).
V_{DD}PETA	I	**PCI Express Transmitter Analog Voltage.** Serdes transmitter analog power supply (1.0V).
V_{SS}	I	**Ground.**

Table 1.10 Power, Ground, and SerDes Resistor Pins

Notes

Pin Characteristics

Note: Some input pads of the switch do not contain internal pull-ups or pull-downs. Unused SMBus and System 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, floating pins can cause a slight increase in power consumption. Unused Serdes (Rx and Tx) pins should be left floating. Finally, No Connection pins should not be connected.

Function	Pin Name	Type	Buffer	I/O Type	Internal Resistor[1]	Notes
PCI Express Interface	PE00RN[3:0]	I	PCIe Differential[2]	Serial Link		
	PE00RP[3:0]	I				
	PE00TN[3:0]	O				
	PE00TP[3:0]	O				
	PE01RN[3:0]	I				
	PE01RP[3:0]	I				
	PE01TN[3:0]	O				
	PE01TP[3:0]	O				
	PE02RN[3:0]	I				
	PE02RP[3:0]	I				
	PE02TN[3:0]	O				
	PE02TP[3:0]	O				
	PE03RN[3:0]	I				
	PE03RP[3:0]	I				
	PE03TN[3:0]	O				
	PE03TP[3:0]	O				
	PE04RN[3:0]	I				
	PE04RP[3:0]	I				
	PE04TN[3:0]	O				
	PE04TP[3:0]	O				
	PE05RN[3:0]	I				
	PE05RP[3:0]	I				
	PE05TN[3:0]	O				
	PE05TP[3:0]	O				
	PE06RN[3:0]	I				
	PE06RP[3:0]	I				
	PE06TN[3:0]	O				
	PE06TP[3:0]	O				
	PE07RN[3:0]	I				
	PE07RP[3:0]	I				
	PE07TN[3:0]	O				
	PE07TP[3:0]	O				
	PE08RN[3:0]	I				
	PE08RP[3:0]	I				
	PE08TN[3:0]	O				

Table 1.11 Pin Characteristics (Part 1 of 3)

Notes

Function	Pin Name	Type	Buffer	I/O Type	Internal Resistor[1]	Notes
PCI Express Interface (Cont.)	PE08TP[3:0]	O	PCIe Differential	Serial Link		
	PE09RN[3:0]	I				
	PE09RP[3:0]	I				
	PE09TN[3:0]	O				
	PE09TP[3:0]	O				
	PE12RN[3:0]	I				
	PE12RP[3:0]	I				
	PE12TN[3:0]	O				
	PE12TP[3:0]	O				
	PE13RN[3:0]	I				
	PE13RP[3:0]	I				
	PE13TN[3:0]	O				
	PE13TP[3:0]	O				
	GCLKN[1:0]	I	HCSL	Diff. Clock Input		Refer to Table 9 in the PES48T12G2 Data Sheet
	GCLKP[1:0]	I				
SMBus	MSMBCLK	I/O	LVTTL	STI[3]		pull-up on board
	MSMBDAT	I/O		STI		pull-up on board
	SSMBADDR[2:1]	I		Input	pull-up	
	SSMBCLK	I/O		STI		pull-up on board
	SSMBDAT	I/O		STI		pull-up on board
General Purpose I/O	GPIO[8:0]	I/O	LVTTL	STI, High Drive	pull-up	
System Pins	CLKMODE[1:0]	I	LVTTL	Input	pull-up	
	GCLKFSEL	I			pull-down	
	P01MERGEN	I			pull-down	
	P23MERGEN	I			pull-down	
	P45MERGEN	I			pull-down	
	P67MERGEN	I			pull-down	
	P89MERGEN	I			pull-down	
	P1213MERGEN	I			pull-down	
	PERSTN	I		STI		
	RSTHALT	I		Input	pull-down	
	SWMODE[3:0]	I			pull-down	
EJTAG / JTAG	JTAG_TCK	I	LVTTL	STI	pull-up	
	JTAG_TDI	I		STI	pull-up	
	JTAG_TDO	O				
	JTAG_TMS	I		STI	pull-up	
	JTAG_TRST_N	I		STI	pull-up	

Table 1.11 Pin Characteristics (Part 2 of 3)

Notes

Function	Pin Name	Type	Buffer	I/O Type	Internal Resistor[1]	Notes
SerDes Reference Resistors	REFRES00	I/O	Analog			
	REFRES01	I/O				
	REFRES02	I/O				
	REFRES03	I/O				
	REFRES04	I/O				
	REFRES05	I/O				
	REFRES06	I/O				
	REFRES07	I/O				
	REFRES08	I/O				
	REFRES09	I/O				
	REFRES12	I/O				
	REFRES13	I/O				
	REFRESPLL	I/O				

Table 1.11 Pin Characteristics (Part 3 of 3)

1. Internal resistor values under typical operating conditions are 92K Ω for pull-up and 91K Ω for pull-down.

2. All receiver pins set the DC common mode voltage to ground. All transmitters must be AC coupled to the media.

3. Schmitt Trigger Input (STI).

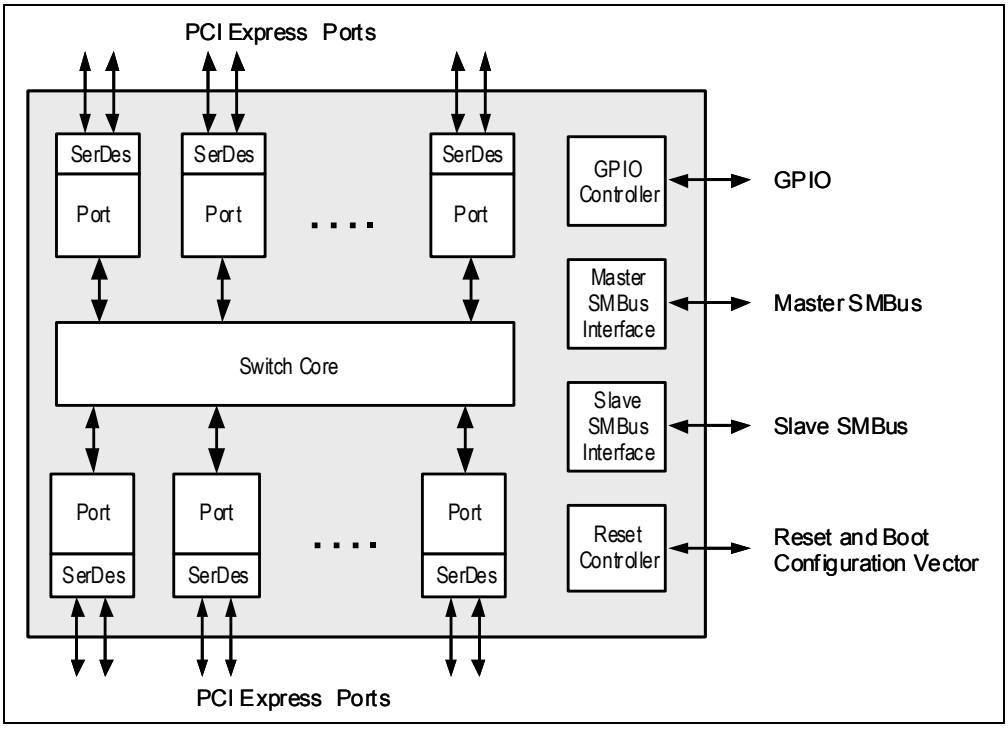
Notes

Architectural Overview

Notes

Introduction

This section provides a high level architectural overview of the PES48T12G2. An architectural block diagram of the PES48T12G2 is shown in .



The PES48H12G2 contains twelve x4 ports labeled port 0 through port 13 (omitting ports 10 and 11 which do not exist in this device). An even port *n* and its odd counterpart, port *n+1*, may be merged to create a single x8 port.

– When ports are merged, the odd numbered port is not logically visible in the PCI Express hierarchy associated with the port.

All ports support 2.5 GT/s (i.e., Gen1) and 5.0 GT/s (i.e., Gen2) operation.

At a high level, the PES48H12G2 consists of ports and a switch core. A port consists of a logic that performs functions associated with the physical, data link, and transactions layers described in the PCIe base 2.0 specification. In addition, a port performs switch application layer functions such as TLP routing using route map tables, processing configuration read and write requests, etc.

The switch core is responsible for transferring TLPs between ports. Its main functions are: input buffering, maintaining per port ingress and egress flow control information, port and VC arbitration, scheduling, and forwarding TLPs between ports.

Since the PES48H12G2 represents a single architecture optimized for both fan-out and system interconnect applications, its switch core is based on a non-blocking crossbar.

Notes

Logical View

The logical view of a PCIe switch is shown in Figure 2.1. A PCI switch contains one upstream port and one or more downstream ports. Each port is associated with a PCI-to-PCI bridge. All PCI-to-PCI bridges associated with a PCIe switch are interconnected by a virtual PCI bus.

- The primary side of the upstream port's PCI-to-PCI bridge is associated with the external link, while the secondary side connects to the virtual PCI bus.
- The primary side of a downstream port's PCI-to-PCI bridge is connected to the virtual PCI bus, while the secondary side is associated with the external link.

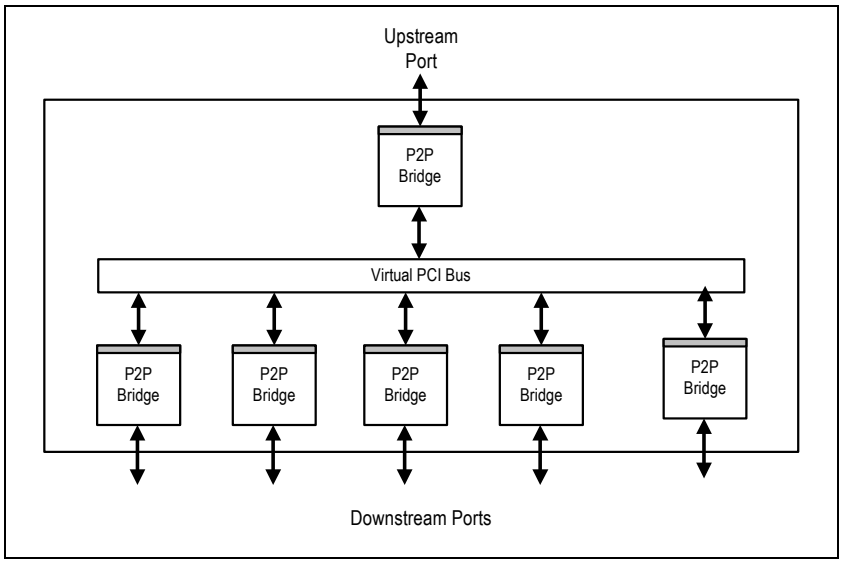


Figure 2.1 Transparent PCIe Switch

Switch Core

Notes

Introduction

This chapter provides an overview of the PES48T12G2's Switch Core. As shown in Figure 2.1 in the Architectural Overview chapter, the Switch Core interconnects switch ports. The Switch Core's main function is to transfer TLPs among these ports efficiently and reliably. In order to do so, the Switch Core provides buffering, ordering, arbitration, and error detection services.

Switch Core Architecture

The Switch Core is based on a non-blocking crossbar design optimized for system interconnect (i.e., peer-to-peer) as well as fanout (i.e., root-to-endpoint) applications. At a high level, the Switch Core is composed of ingress buffers, a crossbar fabric interconnect, and egress buffers. These blocks are complemented with ordering, arbitration, and error handling logic (not shown in the figure).

Each port has dedicated ingress and egress buffer. The ingress buffer stores data received or generated by the port. The egress buffer stores data that will be sent to the port. The crossbar interconnect is a matrix of pathways, capable of concurrently transferring data among all possible port pairs (e.g., port 0 can transfer data to port 1 at the same time port 2 transfers data to port 3).

As packets are received from the link they are stored in the corresponding ingress buffer. After undergoing ordering and arbitration, they are transferred to the corresponding egress buffer via the crossbar interconnect. The presence of egress buffers provides head-of-line-blocking (HOLB) relief when an egress port is congested. For example, a packet received on port 0 that is destined to port 1 may be transferred from port 0's ingress buffer to port 1's egress buffer even if port 1 does not have sufficient egress link credits. This transfer allows subsequent packets received on port 0 to be transmitted to their destination.

Ingress Buffer

When a packet is received from the link, the ingress port's Application Layer determines the packet's route and subjects it to TC/VC mapping. The packet is then stored in the appropriate IFB, together with its routing and handling information (i.e., the packet's descriptor). The IFB consists of three queues. These queues are the posted transaction queue (PT queue), the non-posted transaction queue (NP queue), and the completion transaction queue (CP queue).

- The queues for the IFB are implemented using a descriptor memory and a data memory.
- When two x4 ports are merged to create a x8 port, the descriptor and data memories for both x4 ports are merged.

The default size of each of these queues is shown in Table 3.1.

Port Mode	IFB Queue	Total Size and Limitations (per-port)	Advertised Data Credits	Advertised Header Credits
x4 Bifurcated	Posted	6176 Bytes and up to 64 TLPs	386	64
	Non Posted	1024 Bytes and up to 64 TLPs	64	64
	Completion	6176 Bytes and up to 64 TLPs	386	64

Table 3.1 IFB Buffer Sizes (Part 1 of 2)

Notes

Port Mode	IFB Queue	Total Size and Limitations (per-port)	Advertised Data Credits	Advertised Header Credits
x8 Merged	Posted	12352 Bytes and up to 127 TLPs	772	127
	Non Posted	2048 Bytes and up to 127 TLPs	128	127
	Completion	12352 Bytes and up to 127 TLPs	772	127

Table 3.1 IFB Buffer Sizes (Part 2 of 2)

Egress Buffer

The EFBs provide head-of-line-blocking (HOLB) relief to the IFBs by allowing packets to be stored in an egress port's EFB even if the port's link does not have sufficient credits to accept the packet. HOLB relief allows subsequent packets in the IFB to be transferred to their destinations efficiently. As packets are transferred from an IFB to an EFB, they are subjected to the egress port's TC/VC mapping and stored in the EFB.

Each EFB consists of three queues. These are the posted queue, non-posted queue, and completion queue. The use of these queues allows for packet re-ordering to improve transmission efficiency on the egress link. Refer to section Packet Ordering on page 3-3 for details.

- The queues for both EFBs are implemented using a descriptor memory and a data memory.
- When two x4 ports are merged to create a x8 port, the descriptor and data memories for both x4 ports are merged.

The default size of each of these queues is shown in Table 3.2.

Port Mode	EFB Queue	Total Size and Limitations (per-port)
x4 Bifurcated	Posted	6176 Bytes and up to 64 TLPs
	Non Posted	1024 Bytes and up to 64 TLPs
	Completion	6176 Bytes and up to 64 TLPs
x8 Merged	Posted	12352 Bytes and up to 128 TLPs
	Non Posted	2048 Bytes and up to 128 TLPs
	Completion	12352 Bytes and up to 128 TLPs

Table 3.2 EFB Buffer Sizes

In addition to providing HOLB relief, the EFB is used as a dynamically sized replay buffer. This allows for efficient use of the egress buffer space: when transmitted packets are not being acknowledged by the link partner the replay buffer grows to allow further transmission; when transmitted packets are successfully acknowledged by the link partner the replay buffer shrinks and this space is used as egress buffer space to provide maximum HOLB relief to the IFBs. Assuming a link partner issues acknowledges at the rates recommended in the PCI Express 2.0 spec, the replay buffer naturally grows to the optimal size for the port's link width and speed. Table 3.3 shows the maximum number of TLPs that may be stored in the EFB's replay buffer.

Notes

Port Mode	Replay Buffer Storage Limit
x4 Bifurcated	32 TLPs
x8 Merged	64 TLPs

Table 3.3 Replay Buffer Storage Limit

Crossbar Interconnect

The crossbar is an 12x12 matrix of pathways, capable of concurrently transferring data between a maximum of 12 port pairs. The crossbar interconnects the port ingress buffers to the egress buffers. It provides two data-interfaces per port, one for the port's ingress buffers and one for the port's egress buffers.

Figure 3.1 shows the interface between the crossbar and a port's ingress and egress buffers. The crossbar is able to support 12 simultaneous data transfers. This architecture is well suited for system interconnect applications, as it allows simultaneous full-duplex communication between up to 12 peer devices. Note there are no ports 10 and 11 in this device.

Figure 3.1 Crossbar Connection to Port Ingress and Egress Buffers

Datapaths

As mentioned earlier, the Switch Core interfaces with 12 ports. The interface between each port and the switch core can be logically divided into ingress data interface, egress data interface.

The ingress data interface transfers data received by the port from the PCIe link into the switch-core. The egress data interface transfers data from the switch-core to the port. All data paths through the ingress data interface, crossbar interconnect and egress data interface are 160-bits wide instead of the required 128-bits (i.e., a x8 Gen2 port requires a throughput of 128-bits per clock cycle). On the ingress data interface, the Switch Core receives data from the port at a rate determined by the operational mode of the port (merged or bifurcated) and the width and speed of the port's link. Packets received from the port are stored in the appropriate IFB queue. After being queued in an IFB[1] and undergoing ordering and arbitration, all data transferred through the crossbar interconnect is transferred in a continuous TLP manner (i.e., the data path is never multiplexed).

This choice of datapath width implies that the crossbar has 20% higher throughput than the throughput required to service all ports. This "over-speed" ensures that inter-port messages (i.e., internal messages exchanged by ports for switch management) do not affect the throughput of the PCIe links.

On the egress interface, data in the EFB is read by the port's data link layer (i.e., DL) when it is chosen to be transmitted on the link. If the port is in merged mode, the DL allocates all clock cycles to read data from the EFB. However, depending on the negotiated link width not all clock cycles may be used to transfer data. If the port is in bifurcated mode, the DL reads data from the appropriate EFB (i.e., each port has a dedicated EFB). Again, depending on the negotiated link width, not all clock cycles may be used to transfer data.

Packet Ordering

The PCI Express specification 2.0 contains packet ordering rules to ensure the producer/consumer model is honored across a PCIe hierarchy and to prevent deadlocks. The Switch Core performs packet ordering on a per-port basis, at the output of the ingress and egress buffers of each port (refer to Figure 3.1).

1. Please refer to section Cut-Through Routing on page 3-5 for further information on conditions for cut-through transfers to occur.

Notes

Applying ordering rules at the output of the ingress buffer (i.e., before the crossbar) is done to ensure that packets are ordered regardless of their destination port. This guarantees that the producer/consumer model is met when the data transfer involves any number of peers.

Applying ordering rules at the output of the egress buffer is done to allow packets in the EFB to be re-ordered for deadlock prevention and efficient transmission on the link without violating the PCIe ordering rules. Without this ordering logic, packets in the EFB would need to be transmitted in the order they were received by the EFB. If the oldest packet in the EFB lacked sufficient link credits for its departure, head-of-line blocking would occur at the EFB. The presence of ordering logic at the EFB reliefs potential head-of-line blocking by allowing other packets to be transmitted, as long as ordering rules are not violated.

Table 3.4 shows the ordering rules honored by the Switch Core. Note that the PES48T12G2 honors the relaxed-ordering attribute in packets as shown in the table.

Row Pass Column?		Posted Request	Non-Posted Request		Completion	
		Memory Write or Message Request	Read Request	IO or Configuration Write Request	Read Completion	IO or Configuration Write Completion
Posted Request	Memory Write or Message Request	No	Yes	Yes	Yes	Yes
Non Posted Request	Read Request	No	No	No	Yes	Yes
	IO or Configuration Write Request	No	No	No	Yes	Yes
Completion Request	Read Completion	'Yes' if packet has RO bit set; Else 'No'	Yes	Yes	No	No
	IO or Configuration Write Completion		Yes	Yes	No	No

Table 3.4 Packet Ordering Rules in the PES48T12G2

Arbitration

Packets stored in the ingress buffers are subject to arbitration as they are moved towards the egress port. The switch core performs all packet arbitration functions in the switch. Architecturally, arbitration is done at the egress ports. Each port has a dedicated arbitration configuration as programmed in the port's VC Capability Structure.

Packets undergo two levels of arbitration at an egress port:

- Port arbitration within a VC
- VC arbitration for access to the egress link

Figure 3.2 shows the architectural model of arbitration. The following sub-sections describe arbitration in detail.

Note: There are no ports 10 and 11 in this device.

Notes

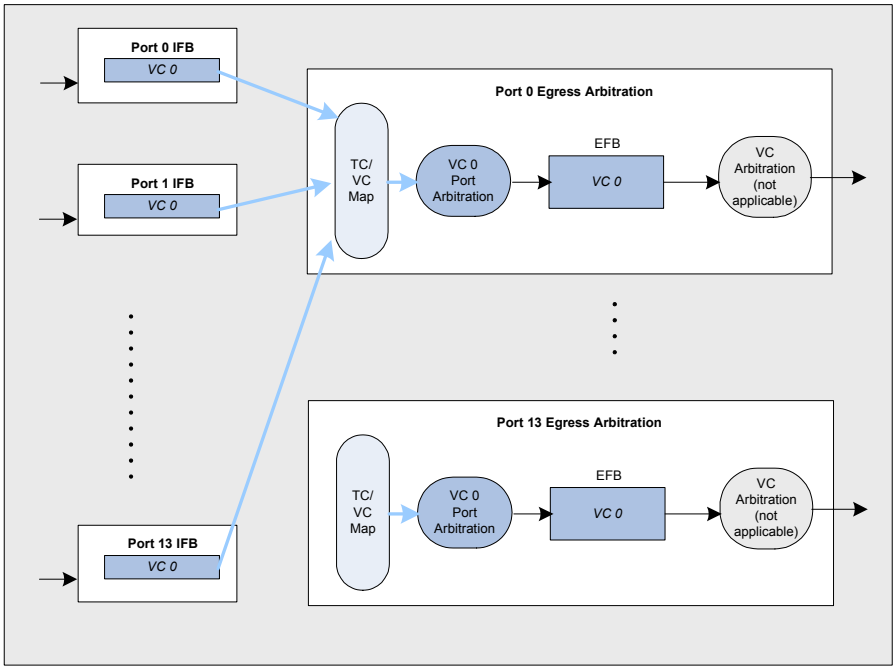


Figure 3.2 Architectural Model of Arbitration

Port Arbitration

Each egress port does port arbitration among multiple ingress ports for packets. Each egress port contains a port arbiter. Ingress port(s) that wish to transfer one or more packets to the egress port participate in arbitration.

Prior to participating in port arbitration, each ingress port does packet ordering. Based on this, each ingress port selects zero, one, or multiple packets as candidates for transfer towards the EFBs.Port arbitration is done according to the configuration of the egress port's VC Capability Structure. The PES48T12G2 ports operating in upstream switch port or downstream switch port mode support port arbitration using Hardware Fixed Round-Robin (default) and 32-phase Weighted Round-Robin (WRR) algorithms[1]. The arbitration algorithm and WRR arbitration parameters are programmed independently for each of the ports via the VC Capability Structure located in the port's configuration space.

Cut-Through Routing

The PES48T12G2 utilizes a combined input and output buffered cut-through switching architecture to forward PCIe TLPs between switch ports. Cut-through means that while a TLP is being received on an ingress link, it can be simultaneously routed across the switch and transferred on the egress link. The entire TLP need not be received and buffered prior to starting the routing process (i.e., store-and-forward). This reduces the latency experienced by packets as they are transferred across the switch.

Typically, cut-through occurs when a TLP is received on an ingress link whose bandwidth is greater than or equal to the bandwidth of the egress link. For example, a TLP received on a x4 Gen2 port and destined to a x1 Gen2 port is cut-through the switch. This rule ensures that the ingress link has enough bandwidth to prevent 'underflow' of the egress link. In addition to this, the PES48T12G2 does "adaptive cut-through", meaning that packets are cut-through even if the egress link bandwidth is greater than the ingress link bandwidth. In this case, the cut-through transfer starts when the ingress port has received enough quantity of the packet such that the packet can be sent to the egress link without underflowing this link.

1. Weighted round robin arbitration with 32-phases is implemented by converting the PCIe port arbitration table into weighted round robin counts. Therefore, over short intervals grants may not match the phase table configuration.

Notes

The ingress and egress link bandwidth is determined by the negotiated speed and width of the links. Table 3.5 shows the conditions under which cut-through and adaptive-cut-through occur. When the conditions are met, cut-through is performed across the IFB, crossbar[1], and EFB. Note that a packet undergoing a cut-through transfer across the Switch Core may be temporarily delayed by the presence of prior packets in the IFB and/or EFB. In this case, the packet starts cutting-through as soon as it becomes unblocked by prior packets.

When cut-through routing of a packet is not possible, the packet is fully buffered in the appropriate IFB prior to being transferred to the EFB and towards the egress link (i.e., store-and-forward operation). Once the packet is stored in the IFB, there is no necessity to fully store it in the EFB as it is transferred towards the egress link.

Ingress Link Speed	Ingress Link Width	Egress Link Speed	Egress Link Width	Conditions for Cut-Through
2.5 Gbps	x8	2.5	x8, x4, x2, x1	Always
		5.0	x4, x2, x1	Always
			x8	At least 50% of packet is in IFB
	x4	2.5	x4, x2, x1	Always
			x8	At least 50% of packet is in IFB
		5.0	x2, x1	Always
			x4	At least 50% of packet is in IFB
			x8	At least 75% of packet is in IFB
	x2	2.5	x2, x1	Always
			x4	At least 50% of packet is in IFB
			x8	At least 75% of packet is in IFB
		5.0	x1	Always
			x2	At least 50% of packet is in IFB
			x4	At least 75% of packet is in IFB
			x8	At least 100% of packet is in IFB
	x1	2.5	x1	Always
			x2	At least 50% of packet is in IFB
			x4	At least 75% of packet is in IFB
			x8	Never (100% of packet is in IFB)
		5.0	x1	At least 50% of packet is in IFB
			x2	At least 75% of packet is in IFB
			x4	Never (100% of packet is in IFB)
			x8	

Table 3.5 Conditions for Cut-Through Transfers (Part 1 of 2)

1. During cut-through transfers, the crossbar maintains the connection between the appropriate IFB and EFB through-out the duration of the transfer.

Notes

Ingress Link Speed	Ingress Link Width	Egress Link Speed	Egress Link Width	Conditions for Cut-Through
5.0 Gbps	x8	2.5	x8, x4, x2, x1	Always
		5.0	x8, x4, x2, x1	Always
	x4	2.5	x8, x4, x2, x1	Always
		5.0	x8	At least 50% of packet is in IFB
			x4, x2, x1	Always
	x2	2.5	x8	At least 50% of packet is in IFB
			x4, x2, x1	Always
		5.0	x8	At least 75% of packet is in IFB
			x4	At least 50% of packet is in IFB
			x2, x1	Always
	x1	2.5	x8	At least 75% of packet is in IFB
			x4	At least 50% of packet is in IFB
			x2, x1	Always
		5.0	x8	Never (100% of packet is in IFB)
			x4	At least 75% of packet is in IFB
			x2	At least 50% of packet is in IFB
			x1	Always

Table 3.5 Conditions for Cut-Through Transfers (Part 2 of 2)

Request Metering

Request metering may be used to reduce congestion in PCI express switches caused by a static rate mismatch. Request metering is available on all PES48T12G2 switch ports but is disabled by default. A static rate mismatch is a mismatch in the capacity of the path from a component injecting traffic into the fabric (e.g., a Root Complex) and the ultimate destination (e.g., an Endpoint).

An example of a static rate mismatch in a PCIe fabric is a x8 root injecting traffic destined to a x1 endpoint. PCIe fabrics are typically no more than one switch deep. Therefore, static rate mismatches typically occur within a switch due to asymmetric link rates. Figure 3.3 illustrates the effect of congestion on PCIe fabric caused by a static rate mismatch. In this example there are two endpoints issuing memory read requests to a root. Endpoint A has a x1 link to the switch, while endpoint B and the root complex have a x8 link.

Memory read request TLPs are three or four DWords in size. A single memory read request may result in up to 4 KB of completion data being returned to the requester. Depending on system architecture and configured maximum payload size, this completion data may be returned as a single completion TLP or may be returned as a series of small (e.g., 64B data) TLPs.

Consider an example where Endpoints A and B are injecting read request to the root at a high rate and the root is able to inject completion data into the fabric at a rate higher than which may be supported by endpoint A's egress link. The result is that the endpoint A's EFB and the root's IFB may become filled with queued completion data blocking completion data to endpoint B.

Notes

If read requests are injected sporadically or at a low rate, then buffering within the switch may be used to accommodate short lived contention and allow completions to endpoints to proceed without interfering. If read requests are injected at a high rate, then no amount of buffering in the switch will prevent completions from interfering.

PCIe has no end-to-end QoS mechanisms. Therefore, it is common for Endpoints to be designed to inject requests into a fabric at high rates. Request metering is a congestion avoidance mechanism that limits the request injection rate into a fabric. Although this example illustrates the effect of a static rate mismatch in an I/O connectivity application, similar situations may occur in system interconnect applications.

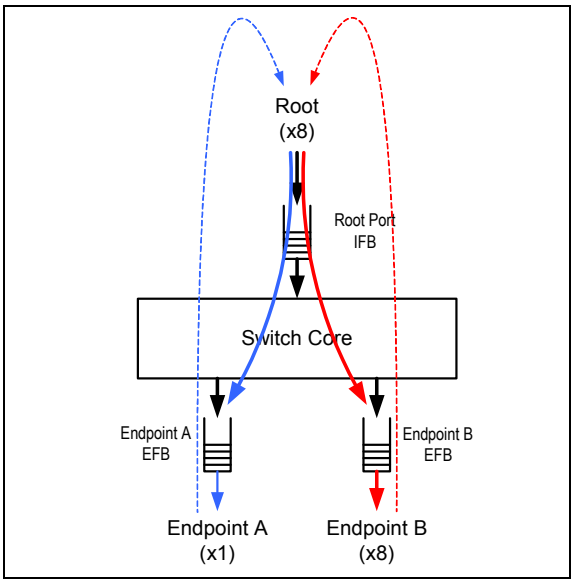


Figure 3.3 PCIe Switch Static Rate Mismatch

The request metering operation is illustrated Figure 3.4. Figure 3.4(a) shows requests injection without request metering. Figure 3.4(b) shows requests injection with request metering. Request metering is implemented by logic at the interface between the IFB and the switch core arbiter. When a request reaches the head of the non-posted IFB queue, request metering logic examines the request and estimates the amount of time that the associated completion TLPs will consume on the endpoint link (i.e., completion transfer time). The request is then allowed to proceed and a timer is initiated with the estimated completion transfer time. The next request from that IFB is not allowed to proceed until the timer has expired.

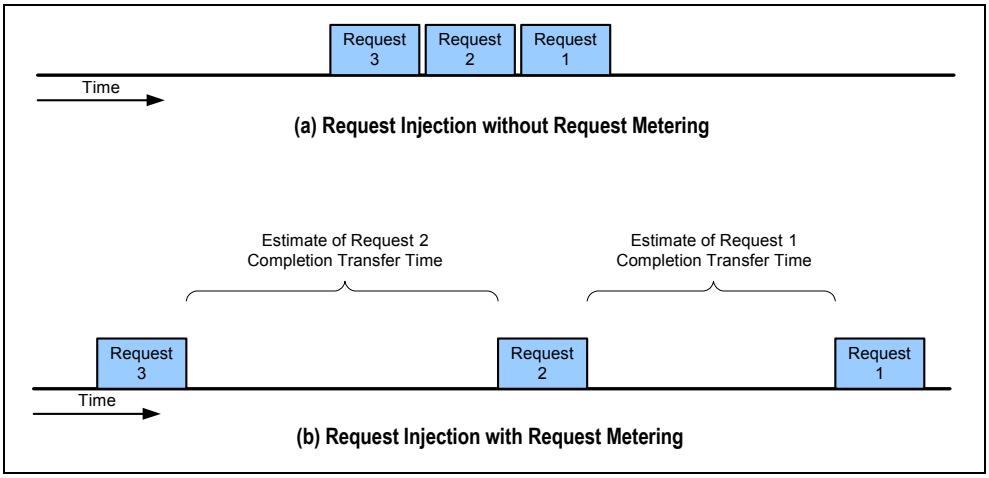


Figure 3.4 PCIe Switch Static Rate Mismatch

Notes

The request metering implementation in the PES48T12G2 makes a number of simplifying assumptions that may or may not be true in all systems. Therefore, it should be expected that some amount of parameter tuning may be required to achieve optimum performance.

Note that tuning of the request metering mechanism should take into account the completion timeout value of the associated requesters (i.e., request metering should be tuned such that a requester's completion timeout value is not violated).

Operation

The completion transfer timer is implemented using a counter. The counter is loaded with an estimate of the number of DWords that will be transferred on the link in servicing the completion and is decremented at a rate that corresponds to the number of DWords that will be transferred on the link in a 4ns period.

Request metering is enabled on an input port when the Enable (EN) bit is set in the Requester Metering Control (RMCTL) register. An non-posted request TLP is allowed to be transferred into the switch core when the request metering counter is zero.

When a request is transferred into the switch core, the request metering counter is loaded with a value that estimates the number of DWords associated with the corresponding completion(s). The method for determining this value is described in section Completion Size Estimation on page 3-11.

- The request metering counter is a 24-bit counter. The count represents a signed-magnitude fixed-point 0:13:11 number (i.e., a positive number with 13 integer bits and 11 fractional bits) but is treated by the logic as a 24-bit unsigned integer.
- The value loaded into the request metering counter for the last non-posted request is available in the Count (COUNT) field of the Request Metering Counter (RMCOUNT) register.

  The requester metering initial counter value computed as described in section Completion Size Estimation on page 3-11 is a sign-magnitude fixed point 0:13:3 number (i.e., a positive number with 13 integer bits and 3 fractional bits).

  The least significant eight fractional bits of the initial counter value are always implicitly zero.

Figure 3.5 shows the request metering count and its initial value.

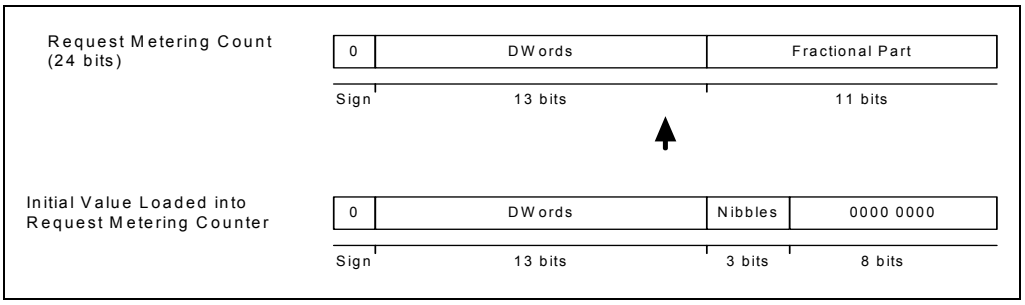


Figure 3.5 Request Metering Count and Initial Value Loaded

The request metering counter is decremented by a value that corresponds to the number of DWords transferred on the link per 4ns period. The value is equal to the sum of the decrement value plus the value of the Decrement Value Adjustment (DVADJ) field in the RMCTL register.

The decrement value is a sign-magnitude fixed-point 0:4:3 number (i.e., an positive number with 4 integer bits and 3 fractional bits), determined by the port's negotiated link width and speed as shown in Table 3.6.

- The least significant eight fractional bits of the decrement value are always implicitly zero.

Notes

The Decrement Value Adjustment (DVADJ) field represents a sign-magnitude fixed point 0:4:11 number (i.e., a positive fixed-point number with 4 integer bits and 11 fractional bits).

- DVADJ field provides fine grain programmable adjustment of the value by which the counter is decremented.
- The sign bit in the DVADJ field should not be set to negative (i.e., 0b1).

Figure 3.6 shows the decrement value and the decrement value adjustment.

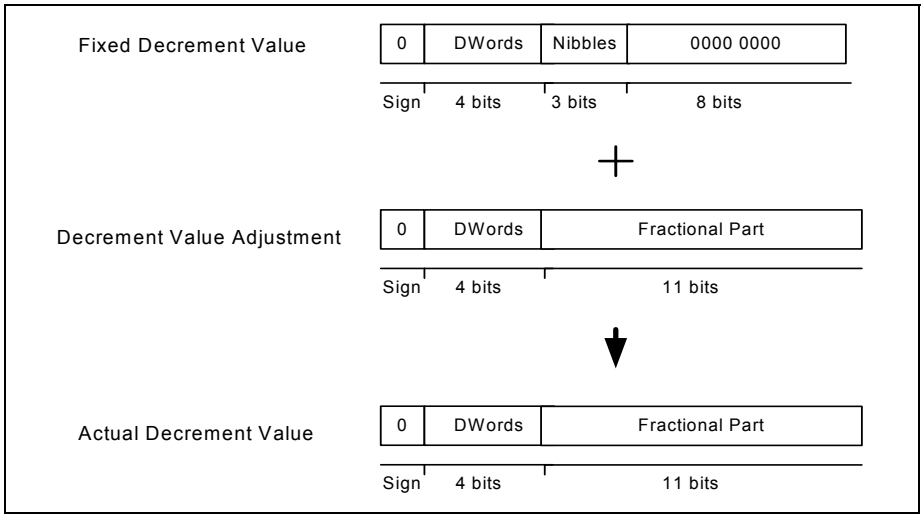


Figure 3.6 Decrement Value and Decrement Value Adjustment

The counter stops decrementing when it reaches zero or when a rollover occurs (i.e., the decrement causes it to become negative).

Link Width	Link Speed	Decrement Value	Notes
x1	Gen 1	0x02	Corresponds to 1 Byte per clock tick
x2	Gen 1	0x04	Corresponds to 2 Bytes per clock tick
x4	Gen 1	0x08	Corresponds to 4 Bytes per clock tick
x8	Gen 1	0x10	Corresponds to 8 Bytes per clock tick
x1	Gen 2	0x04	Corresponds to 2 Bytes per clock tick
x2	Gen 2	0x08	Corresponds to 4 Bytes per clock tick
x4	Gen 2	0x10	Corresponds to 8 Bytes per clock tick
x8	Gen 2	0x20	Corresponds to 16 Bytes per clock tick

Table 3.6 Request Metering Decrement Value

The computation that occurs on each clock tick by the request metering counter is shown Figure 3.7.

Notes

```
tmp = RequestMeteringCounter
RequestMeteringCounter     -=     (DecrementValue[LinkSpeed,LinkWidth]     +
RMCTL.DVADJ)
if (tmp < RequestMeteringCounter) {
    RequestMeteringCounter = 0
```

Figure 3.7 Request Metering Counter Decrement Operation

Completion Size Estimation

This section describes the value that is loaded into the request metering counter when a request is transferred into the switch core. This value is referred to as the completion size estimate. The completion size estimate is based on the type of non-posted request as described below. The request metering counter is a 24-bit counter that represents a fixed point 0:13:11 number (i.e., an unsigned number with 13 integer bits and 11 fractional bits).

The completion size estimate is a 0:13:3 number. The least significant eight fractional bits of the completion size estimate are always implicitly zero.

Non-Posted Writes

The completion size estimate is 0x0018 which corresponds to 3 DWords (3 DWord header)

Non-Posted Reads

The completion size estimate is based on the Length field in the read request header and is computed as shown in Figure 3.8. All arithmetic in this section is performed using an implicit 0:13:3 representation and all values are implicitly converted to this value.

```
DataDWords = ⌈Length / 4⌉*4
If (DataDWords == 0) {
    CompletionSizeEstimate = 3
} else if (DataDWords <= CnstLimit) {
    CompletionSizeEstimate = DataDWords + 1
} else {
    OverheadDWords = (DataDWords >> OverheadFactor)
    CompletionSizeEstimate = DataDWords + OverheadDWords
}
```

Figure 3.8 Non-Posted Read Request Completion Size Estimate Computation

The number of data DWords in a non-posted request TLP is estimated by the number of PCI Express data credits required by the corresponding completion(s). Each PCI Express data credit is 4 DWords or 16 bytes.

– The first line in Figure 3.8 computes the number of DWords required by the completion(s) using the number of required PCI Express data credits. This corresponds to PCI Express completion data credits multiplied by 4.

If the number of data DWords is zero, then the completion size is estimated to be three DWords (i.e., a 0:13:3 representation value of 0x0018).

– Otherwise, if the number of required data DWords is less than the Constant Limit (CNSTLIMIT) field in the RMCTL register, then the completion size is estimated as the number of required data DWords plus one.

Notes

– Otherwise, if the number of required data DWords is greater than CNSTLIMIT, then the completion size is estimated using OverheadDWords as described below.

OverheadDWords represents the number of DWords of link overhead. This includes the header, data link layer overhead, and physical layer overhead of the completion TLP(s) associated with this request. Ideally, OverheadDWords would be set to the number of completion TLPs associated with the request multiplied by the TLP overhead. Unfortunately, this requires a multiplication. Therefore, the following estimate may be used.

A completion header is 3 DWords. There are 2 DWords of additional overhead associated with a TLP. Therefore a reasonable estimate of the overhead is 5 DWords. In many systems, completions are 64-bytes in size (i.e., 16 DWords in size).

OverheadDWords = (Length / 16) * 5.

– This is approximately equal to OverheadDWords = (Length / 16) * 4.
– This may be simplified to (Length / 4) and may be computed as (Length >> 2).

Thus, an acceptable value for OverheadFactor in many systems is 2. The OverheadFactor value used in computing the completion size estimate is contained in the Overhead Factor (OVRFACTOR) field in the RMCTL register.

Internal Errors

Internal errors are errors associated with a PCI Express interface that occurs within a component and which may not be attributable to a packet or event on the PCI Express interface itself or on behalf of transactions initiated on PCI Express.

The PES48T12G2 classifies the following IDT proprietary switch errors as internal errors.

– Switch core time-outs
– Single and double bit internal memory ECC errors
– End-to-end data path parity protection errors

Internal errors are reported by the port in which they are detected through AER as outlined in the PCI-SIG Internal Error Reporting ECN. The reporting of internal errors may be disabled by clearing the Internal Error Reporting Enable (IERROREN) bit in the port's Internal Error Reporting Control (IERRORCTL) register. When internal error reporting is disabled, the following AER fields become read-only:

– Uncorrectable Internal Error Mask (UIE) field in the AERUEM register
– Uncorrectable Internal Error Severity (UIE) field in the AERUESV register
– Correctable Internal Error Mask (CIE) field in the AERCEM register
– Header Log Overflow Mask (HLO) field in the AERCEM register

The PES48T12G2 does not support recording of headers for uncorrectable internal errors. When an uncorrectable internal error is reported by AER, a header of all ones is recorded.

Corresponding to each possible internal error source is a status bit in the Internal Error Reporting Status (IERRORSTS) register. A bit is set in the status register when the corresponding internal error is detected. Associated with each internal error status bit in the IERRORSTS register is a mask bit in the Internal Error Reporting Mask (IERRORMSK) register. When a mask bit is set in this register, the setting of the corresponding status bit is masked from generating an internal error.

Each internal error status bit has an associated severity bit in the IERRORSEV register. When an unmasked internal error is detected, the error is reported as dictated by the corresponding severity bit (i.e., either an Uncorrectable Internal Error or a Correctable Internal Error). When an uncorrectable or correctable internal error is reported, the corresponding AER status bit is set and process as dictated by the PCIe base specification and Internal Error Reporting ECN.

To facilitate testing of software error handlers, any bit in the IERRORSTS register may be set by writing a one to the corresponding bit position in the Internal Error Test (IERRORTST) register. Once a bit is set in the ERRORSTS register, it is processed as though the actual error occurred (e.g., reported by AER).

Notes

Switch Time-Outs

The switch core discards any TLP that reaches the head of an IFB or EFB queue and is more than 64 seconds old. This includes posted, non-posted, completion and inserted TLPs. If during processing of a TLP with broadcast routing a switch core time-out occurs, then the switch core will abort processing of the TLP. This may result in the broadcast TLP being transmitted on some but not all downstream pots.

Memory SECDED ECC Protection

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 many 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 PES48T12G2 protects all memories (i.e., both data and control structures) with a Single Error Correction with Double Error Detection (SECDED) Error Correcting Code (ECC). The objective of this memory protection is to prevent silent data corruption. Single bit errors are automatically corrected and optionally reported while double bit errors are optionally reported.

Double bit errors are uncorrectable memory errors that may compromise the integrity of control and data structures. Detection of a double bit error may result in further modification of one or more memory bits in the data quantity in which the error was detected (i.e., single bit error correction is not disabled when a double bit error is detected and a double bit error may result in one or more single bit corrections).

Associated with each port are five memories: IFB control, IFB data, and EFB control, EFB data, and Replay Buffer Control. Each port contains memory error control and status registers that are used to manage memory errors associated with that port.

When a single or double bit error is detected in a memory, the status bit corresponding to the memory in which the error was detected is set in the Internal Error Reporting Status (IERRORSTS) register.

A double bit error detected by a memory associated with TLP data (i.e., IFB or EFB data) results in the TLP being nullified when it reaches the DL layer of an egress port. The TLP is nullified by inverting the computed LCRC and ending the packet with an EDB symbol. Nullified TLPs received by a link partner are discarded. Although the TLP is nullified, flow control credits associated with the egress port may not be correctly updated. Thus, double bit errors could result in a flow control credit leak.

The DL layer never replays a TLP with a sequence number different from that initially used. If a double bit error is detected during a DL layer replay, then all TLPs in the replay buffer are flushed.

If a double bit error is detected by an internal memory in a TLP that targets a function in the switch (e.g., a configuration read or write request to the PCI-to-PCI bridge function), then the TLP is discarded.

End-to-End Data Path Parity Protection

In addition to memory ECC protection, the PES48T12G2 supports end-to-end data path parity protection. Data flowing into the PES48T12G2 is protected by the LCRC. Within the Data Link (DL) layer of the switch ingress port, the LCRC is checked and a 32-bit 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

Notes

an 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 End-to-End Parity Error (E2EPE) bit is set in the Internal Error Status (IERRORSTS) register. The DL layer never replays a TLP with a sequence number different from that initially used. If a parity error is detected during a DL layer replay, then all TLPs in the replay buffer are flushed.

In addition to TLPs that flow through the switch, cases exist in which TLPs are produced and consumed by the switch (e.g., 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 by setting the E2EPE bit in the IERRORSTS register.

A parity error reported at a switch port cannot be definitively used to identify the location within the device at which the fault occurred as the fault may have occurred 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 upstream port).

Clocking

Notes

Introduction

Figure 4.1 provides a logical representation of the PES48T12G2 clocking architecture. The PES48T12G2 has a single differential global reference clock input (GCLK).

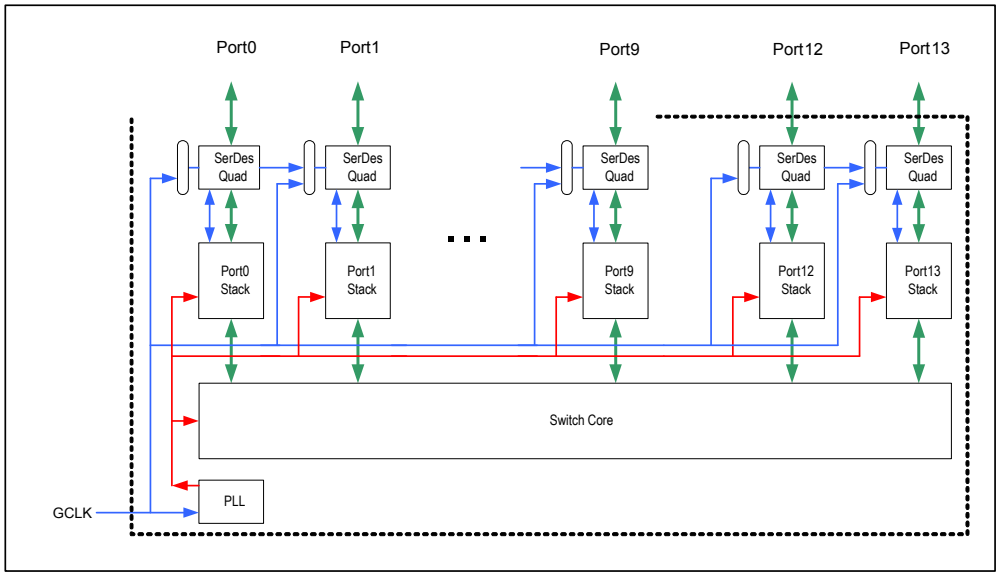


Figure 4.1 Logical Representation of the PES48T12G2 Clocking Architecture

The differential global reference clock input (GCLK) is driven into the device on the GCLKP[1:0] and GCLKN[1:0] pins.

- The nominal frequency of the global reference clock input may be selected by the Global Clock Frequency Select (GCLKFSEL) pin to be either 100 MHz or 125 MHz.
- Both global reference clock differential inputs should be driven with the same frequency. There are no skew requirements between the GCLKP[0]/GCLKN[0] and GCLKP[1]/GCLKN[1] inputs. Any constant phase difference is acceptable.
- The Global Clock supports Spread Spectrum Clocking (SSC).
- The global reference clock input is provided to each SerDes quad and to an on-chip PLL.
  - The on-chip PLL uses this clock to generate a 250 MHz core clock that is used by internal switch logic (e.g., switch core, portion of a stack, etc.).
  - The PLL within each SerDes quad generates a 5.0 GHz clock used by the SerDes analog portion (PMA) and a 250 MHz clock used by the digital portion (PCS).

Port Clocking Mode

Port clocking refers to the clock that a port uses to receive and transmit serial data. All ports in the switch use the global reference clock (GCLK) input for receiving and transmitting serial data. The switch does not introduce any requirements on the global reference clock input beyond those imposed by PCI express. Depending on the system configuration, a port may employ the common Refclk or separate Refclk architectures defined by the PCIe Base specification.

Notes

The port clocking mode of a port is determined by the state of the CLKMODE[1:0] pins in the boot configuration vector as shown in Table 4.1. This field determines the initial value of the Slot Clock Configuration (SCLK) field in each port's PCI Express Link Status (PCIELSTS) register. The SCLK field controls the advertisement of whether or not the port uses the same reference clock source as the link partner. A one in the SCLK field indicates that the port and its link partner use the same reference clock source. This is defined as Common Clock Configuration by the PCI Express Base Specification. A zero in the SCLK field indicates that the port and its link partner do not use the same reference clock source.

CLKMODE[1:0] Value in Boot Configuration Vector	Port 0 SCLK	SCLK for Ports other than Port 0
0	0	0
1	1	0
2	0	1
3	1	1

Table 4.1 Initial Port Clocking Mode and Slot Clock Configuration State

Reset and Initialization

Notes

Introduction

This chapter describes the PES48T12G2 reset and initialization.

When multiple resets are initiated concurrently, the precedence shown in Table 6.1 is used to determine which one is acted upon.

- Reset types and causes are described in detail in the following sections.
  - A switch fundamental reset affects the entire device
  - A port reset affects only that one port
- When a high priority and low priority reset are initiated concurrently and the condition causing the high priority reset ends prior to that causing the low priority reset, then the device/port immediately transitions to the reset associated with low priority reset condition.
  - If the high priority and low priority resets share the same reset type, then the device/port remains in the corresponding reset when the high priority reset condition ends.
  - If the high priority and low priority reset have different reset types, then the device/port transitions to the low priority reset type when the high priority reset condition ends.

Priority	Reset Type	Reset cause
1 (Highest)	Switch fundamental reset	Global reset pin (PERSTN) assertion
2	Hot reset	Reception of TS1 ordered sets on upstream port indicating a hot reset
3	Hot reset	Data link layer of the upstream port transitioning to DL_Down state
4	Upstream secondary bus reset	Setting of the SRESET bit in the switch's upstream port PCI-to-PCI bridge BCTL register
5 (Lowest)	Downstream secondary bus reset	Setting of the SRESET bit in the corresponding port's PCI-to-PCI bridge BCTL register

Table 5.1 PES48T12G2 Reset Precedence

Registers and fields designated as Switch Sticky (SWSticky) or Sticky (Sticky) take on their initial value as a result of the Switch Fundamental Reset. Other resets have no effect on registers and fields with these designations.

All fields designated as Read Write when Unlocked (RWL) are implicitly SWSticky. Their value is preserved across all resets except a switch fundamental reset.

Boot Configuration Vector

A boot configuration vector consisting of the signals listed in Table 5.2 is sampled during a switch fundamental reset. Since the boot configuration vector is only sampled during a switch fundamental reset, the value of signals that make up the boot configuration vector is ignored and their state outside of a switch fundamental reset sequence has no effect on the operation of the device.

While basic switch operation may be configured using signals in the boot configuration vector, advanced switch features require more complex initialization. This initialization may be performed by an external SMBus device via the slave SMBus interface or may be performed automatically via the serial EEPROM.

See Chapter 12, SMBus Interfaces, for a description of the slave SMBus interface and serial EEPROM operation.

Notes

As noted in Table 5.2, some of the initial values specified by the boot configuration vector may be overridden by software, serial EEPROM, or an external SMBus device. The state of all of the boot configuration signals in Table 5.2 sampled during a switch fundamental reset may be determined from the Boot Configuration Status (BCVSTS) register.

Signal	May Be Overridden	Name/Description
GCLKFSEL	N	**Global Clock Frequency Select.** This pin specifies the frequency of the GCLKP and GCLKN signals.
CLKMODE[1:0]	Y	**Clock Mode.** These pins specify the clocking mode used by switch ports. See Table 4.1 for a definition of the encoding of these signals. The value of these signals may be overridden by modifying the Port Clocking Mode (PCLKMODE) register.
P01MERGEN	N	**Ports 0 and 1 Merge.** This pin specifies whether ports 0 and 1 are merged.
P23MERGEN	N	**Ports 2 and 3 Merge.** This pin specifies whether ports 2 and 3 are merged.
P45MERGEN	N	**Ports 4 and 5 Merge.** This pin specifies whether ports 4 and 5 are merged.
P67MERGEN	N	**Ports 6 and 7 Merge.** This pin specifies whether ports 6 and 7 are merged.
P89MERGEN	N	**Ports 8 and 9 Merge.** This pin specifies whether ports 6 and 8 are merged.
P1213MERGEN	N	**Ports 12 and 13 Merge.** This pin specifies whether ports 12 and 13 are merged.
RSTHALT	Y	**Reset Halt.** When this pin is asserted during a switch fundamental reset sequence, the PES48T12G2 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 SWCTL register by an SMBus master.
SSMBADDR[2,1]	N	**Slave SMBus Address.** SMBus address of the switch on the slave SMBus.
SWMODE[3:0]	N	**Switch Mode.** These pins specify the switch operating mode.

Table 5.2 Boot Configuration Vector Signals

Switch Fundamental Reset

A switch fundamental reset may be cold or warm. A cold switch fundamental reset occurs following a device being powered-on and assertion of the global reset (PERSTN) signal. A warm switch fundamental reset occurs when a switch fundamental reset is initiated while power remains applied. The PES48T12G2 behaves in the same manner regardless of whether the switch fundamental reset is cold or warm.

Notes

A switch fundamental reset may be initiated by any of the following conditions.

- A cold switch fundamental reset initiated by application of power (i.e., a power-on) followed by assertion of the global reset (PERSTN) signal.
- A warm switch fundamental reset initiated by assertion of PERSTN while power remains applied.

When a switch fundamental reset is initiated, the following sequence is executed.

1. Wait for the switch fundamental reset condition to clear (e.g., negation of PERSTN).
2. On negation of PERSTN, sample the boot configuration vector signals shown in Table 5.2.
3. If the sampled Switch Mode (SWMODE[3:0]) state corresponds to a test mode, then skip the remainder of this reset sequence and execute the reset sequence outlined in tbd.
4. All registers are initialized to their default value.
5. The Register Unlock (REGUNLOCK) bit is set in the Switch Control (SWCTL) register.
6. The PLL and SerDes are initialized (i.e., PLL lock and CDR reset).
7. Within 20 ms after the switch fundamental reset condition clears, the reset signal to the stacks is negated and link training begins on all ports. While link training takes place, execution of the reset sequence continues.
8. Within 100 ms following clearing of the switch fundamental reset condition, the stacks enter a quasi-reset state.
   - All stacks that have PCIe base specification compliant link partners have completed link training.
   - All stacks are able to receive and process TLPs.
   - Stacks respond to configuration request TLPs with a configuration request retry status completion. All other TLPs are ignored (i.e., flow control credits are returned but the TLP is discarded).
9. The master SMBus operating frequency is 100kH.
10. The slave SMBus is taken out of reset and initialized. The slave SMBus address is specified by the SSMBADDR[2,1] signals in the boot configuration vector.
11. If the sampled Switch Mode (SWMODE[3:0]) state corresponds to a mode that supports serial EEPROM initialization, then the contents of the serial EEPROM are read and appropriate switch registers are updated.
    - If a one is written by the serial EEPROM to the Full Link Retrain (FLRET) bit in any Phy Link State 0 (PHYLSTATE0) register, then link retraining is initiated on the corresponding port using the current link parameters.
    - 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 SWCTL register. Error information is recorded in the SMBUSSTS register.
    - When serial EEPROM initialization completes or when an error is detected, the EEPROM Done (EEPROMDONE) bit in the SMBUSSTS register is set.
12. All stacks remain in a quasi reset state while the Reset Halt (RSTHALT) bit is set in the SWCTL register.
    - In this state the entire device is operational except that the stacks remain in quasi-reset and respond to configuration request TLPs with a configuration request retry completion status.
    - This provides a synchronization point for a device on the slave SMBus to initialize the device. When device initialization is completed, the slave SMBus device clears the RSTHALT bit allowing the device to begin normal operation.
13. The Register Unlock (REGUNLOCK) bit is cleared in the Switch Control (SWCTL) register.
14. Normal device operation begins as dictated by the SWMODE value in the boot configuration vector.

The PCIe specification indicates that a device must respond to Configuration Request transactions within 100ms from the end of Conventional Reset (cold, warm, or hot). Additionally, the PCIe specification indicates that a device must respond to Configuration Requests with a Successful Completion within 1.0

Notes

second after Conventional Reset of a device. The reset sequence above guarantees that the switch will be ready to respond successfully to configuration requests within the 1.0 second period as long as the serial EEPROM initialization process completes within 200 ms.

- During EEPROM initialization, the switch responds to a Configuration Request with Configuration-Request-Retry-Status Completion.
- Under normal circumstances, 200 ms is more than adequate to initialize registers 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 switch fundamental reset with serial EEPROM initialization is illustrated in Figure 5.1.

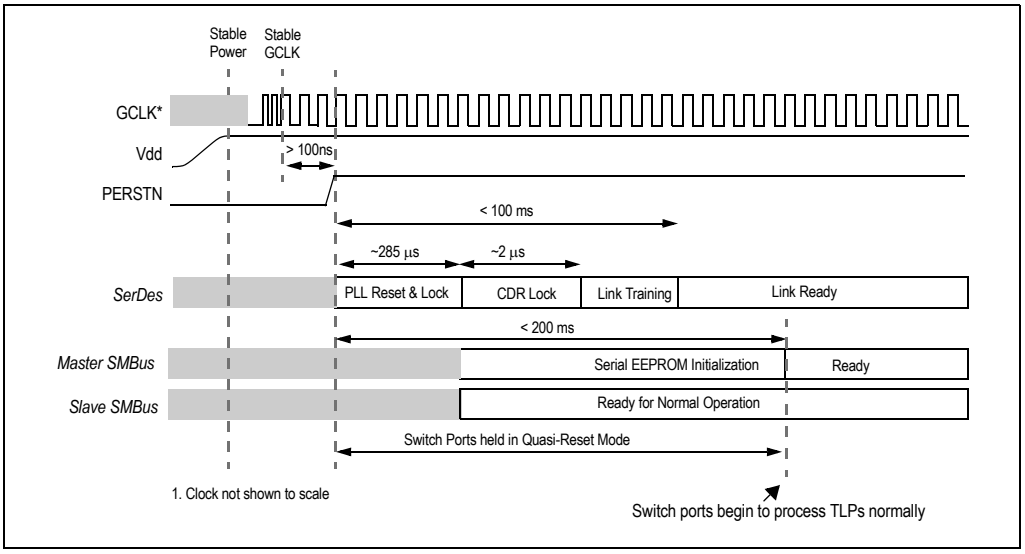


Figure 5.1 Switch Fundamental Reset with Serial EEPROM Initialization

The operation of a switch fundamental reset using RSTHALT is illustrated in Figure 5.2.

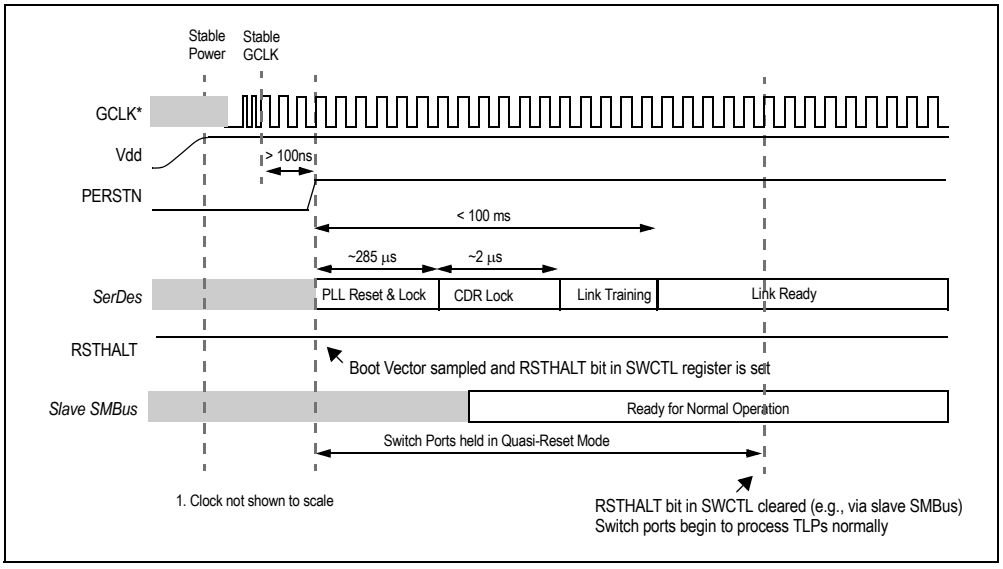


Figure 5.2 Fundamental Reset Using RSTHALT to Keep Device in Quasi-Reset State

Notes

Hot Resets

Hot resets may be subdivided into three subcategories: switch hot reset, upstream secondary bus reset, and downstream secondary bus reset. These subcategories correspond to resets defined by the PCI Express architecture.

- A fundamental reset logically causes all logic associated with the switch to take on its initial state.
- A hot reset logically causes all logic to be returned to an initial state, but does not cause the state of register fields denoted as Sticky or SWSticky to be modified.
- An upstream secondary bus reset logically causes all devices on the virtual PCI bus of the switch to be hot reset except the upstream port (e.g., upstream PCI to PCI bridge).
- A downstream secondary bus reset causes a hot reset to be propagated on the corresponding external link.

Hot Reset

A switch hot reset is initiated by any of the following events.

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

When a switch hot reset is initiated, the following sequence of actions take place.

1. The upstream port (i.e., configured as Upstream Switch Port) transitions its PHY LTSSM[1] state to the appropriate state (i.e., the Hot-Reset state on reception of TS1 ordered-sets indicating hot-reset or else the Detect state).
2. Each downstream switch port whose link is up propagates a hot reset by transmitting TS1 ordered sets with the hot reset bit set. All logic associated with the switch (i.e., switch ports, switch core, etc.) is logically reset to its initial state.
   - If the link associated with a downstream port is in the Disabled LTSSM state, then a hot reset will not be propagated out on that port. The port will instead transition to the Detect LTSSM state. Although not a hot reset, this has the same functional effect on downstream components.
3. All register fields and registers associated with the switch except those designated Sticky and SWSticky, are reset to their initial value. The value of Sticky and SWSticky registers and fields is preserved across a hot reset.
4. As long as the condition that initiated the switch hot reset persists, logic associated with the switch remains at this step.
5. Ports begin to link train and normal switch operation begins.

Upstream Secondary Bus Reset

An upstream secondary bus reset is initiated by any of the following events.

- A one is written to the Secondary Bus Reset (SRESET) bit in the switch's upstream port Bridge Control (BCTL) register[2].

When an Upstream Secondary Bus Reset occurs, the following sequence of actions take place on logic associated with the switch.

1. Each downstream port whose link is up propagates the reset by transmitting TS1 ordered sets with the hot reset bit set.
   - If the link associated with a downstream port is in the Disabled LTSSM state, then a hot reset will not be propagated out on that port. The port will instead transition to the Detect LTSSM state. Although not a hot reset, this has the same functional effect on downstream components.
2. All registers fields in all registers associated with downstream ports, except those designated Sticky and SWSticky, are reset to their initial value. The value of fields designated Sticky or SWSticky is unaffected by an Upstream Secondary Bus Reset.

---

1. The term 'LTSSM' refers to a port's Link Training and Status State Machine in the Physical Layer.
2. Note that the Bridge Control Register is only present in Type 1 Configuration Headers (i.e., PCI-to-PCI Bridge functions).

Notes

3. All TLPs received from downstream ports and queued in the switch are discarded.
4. Logic in the stack and switch core associated with the downstream ports are gracefully reset.
5. Wait for software to clear the Secondary Bus Reset (SRESET) bit in the upstream port's Bridge Control Register (BCTL).
6. Normal downstream port operation begins.

The operation of the upstream port is unaffected by a secondary bus reset. The link remains up and Type 0 configuration read and write transactions that target the upstream port complete normally.

During an Upstream Secondary Bus Reset, all TLPs destined to the secondary side of the upstream port's PCI-to-PCI bridge are treated as unsupported requests.

The operation of the slave SMBus interface is unaffected by an Upstream Secondary Bus Reset. Using the slave SMBus to access a register that is reset by an Upstream Secondary Bus Reset causes the register's default value to be returned on a read and written data to be ignored on writes.

Downstream Secondary Bus Reset

A downstream secondary bus reset may be initiated by the following condition:

– A one is written to the Secondary Bus Reset (SRESET) bit in a downstream port's Bridge Control Register (BCTL).

When a Downstream Secondary Bus Reset occurs, the following sequence of actions take place.

– If the corresponding downstream port's link is up, TS1 ordered sets with the hot reset bit set are transmitted
– All TLPs received from corresponding downstream port and queued are discarded.
– Wait for software to clear the Secondary Bus Reset (SRESET) bit in the downstream port's Bridge Control Register (BCTL).
– Normal downstream port operation begins.

The operation of the upstream port is unaffected by a downstream secondary bus reset. The operation of other downstream ports is unaffected by a downstream secondary bus reset. During a downstream secondary bus reset, Type 0 configuration read and write transactions that target the downstream port complete normally, and all TLPs destined to the secondary side of the downstream port's PCI-to-PCI bridge are treated as unsupported requests.

Port Merging

The switch allows merging of ports to form a single port whose link width is the aggregate sum of the individual port widths. Port merging is only supported between an even numbered port and its subsequent odd numbered port. The PxyMERGEN signals, sampled during switch fundamental reset, select which ports are merged. For example, if the P45MERGEN signal is driven asserted (i.e., low) at switch fundamental reset, then ports 4 and 5 are merged. It is not possible to change this port configuration until a subsequent switch fundamental reset.

When two ports are merged, the even numbered port is active and its odd-numbered pair is de-activated. For example, when ports 4 and 5 are merged, port 4 remains active and port 5 is de-activated. A de-activated port has the following behavior:

– All output signals associated with the port are placed in a negated state (e.g., link status and hot-plug signals).
  • The negated value of PxAIN, PxILOCKP, PxPEP, PxPIN, and PxRSTN is determined as shown in Table 9.2.
  • PxACTIVEN and PxLINKUPN are negated.
– All input signals associated with the port are ignored and have no effect on the operation of the device.

Notes

- The state of the following hot-plug input signals is ignored: PxAPN, PxMRLN, PxPDN, PxPFN, and PxPWRGDN.

– The port is not associated with a PCI Express link. PCI Express configuration requests targeting the port are not possible and the port is not part of the PCI Express hierarchy.

– The port is not associated with any switch partition. The port is unaffected by the state of any switch partition, and vice-versa.

– Unused logic is placed in a low power state.

– All registers associated with the port remain accessible from the global address space.[1]

– The port remains in this state regardless of the setting of the port's operating mode (i.e., via the port's SWPORTxCTL register).

[1] Refer to Chapter 14, Register Organization, for details on the switch's global address space.

Notes

Link Operation

Notes

Introduction

Link operation in the PES48T12G2 adheres to the PCI Express 2.0 Base Specification, supporting speeds of 2.5 GT/s and 5.0 GT/s. The PES48T12G2 contains sixteen x4 ports which may be merged in pairs to form x8 ports. The default link width of each port is x4 and the SerDes lanes are statically assigned to a port.

Each port supports upstream and downstream link behavior. The behavior is determined dynamically by the port's operating state (i.e., upstream switch port, downstream switch port). A full link retrain is defined as retraining of a link that transitions through the Detect LTSSM[1] state.

Polarity Inversion

Each port of the PES48T12G2 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 6 through 15 of the TS1 and TS2 ordered sets for inversion of the PExRP[n] and PExRN[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 not to be inverted.

Lane Reversal

The PCIe specification describes an optional lane reversal feature. The PES48T12G2 supports the automatic lane reversal feature outlined in the PCIe specification. The operation of lane reversal is dependent on the maximum link width determined dynamically by the PHY. The maximum link width is the minimum of:

- The value of the MAXLNKWDTH field in the port's PCI Express Link Capabilities (PCIELCAP) register.
- The number of consecutive lanes detected during the Detect state on which valid training sets are received.

Lane reversal mapping for the various non-trivial maximum link width configurations supported by the PES48T12G2 is illustrated in Figures 6.1 and 6.5.

1. The term 'LTSSM' refers to a port's Link Training and Status State Machine in the Physical Layer.

Notes

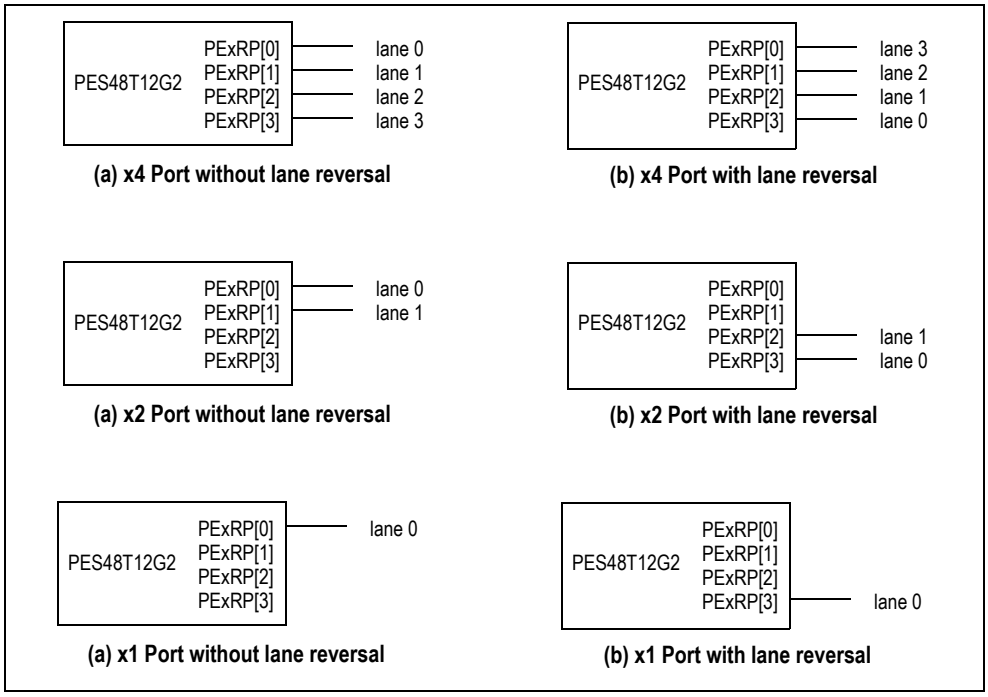


Figure 6.1 Unmerged Port Lane Reversal for Maximum Link Width of x4

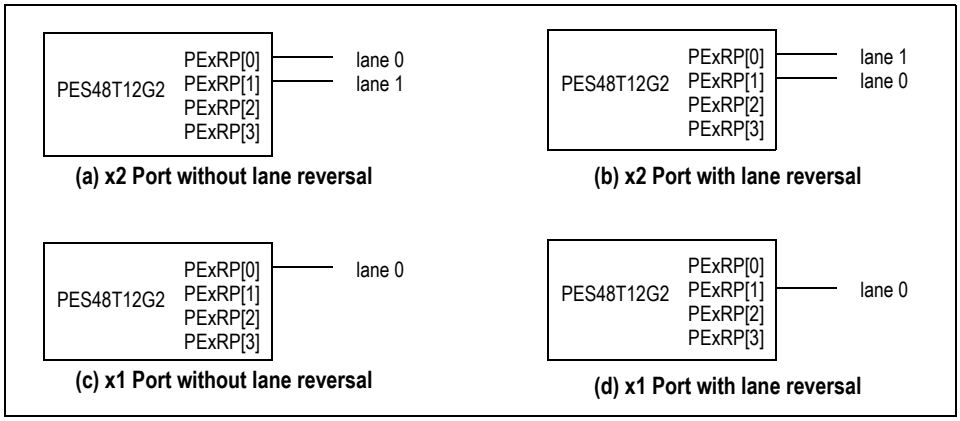


Figure 6.2 Unmerged Port Lane Reversal for Maximum Link Width of x2

Notes

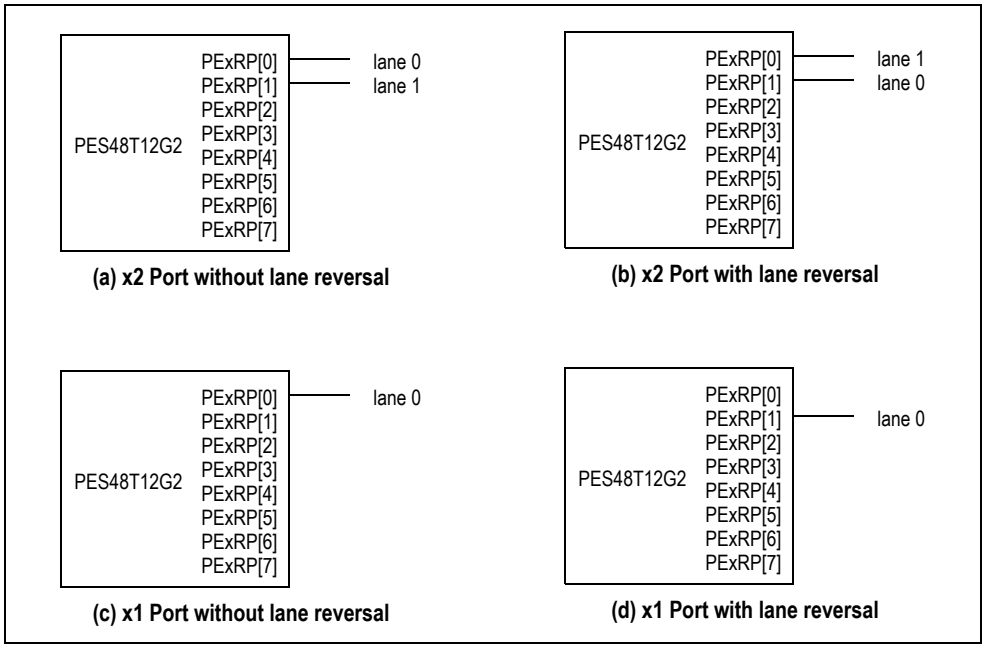


Figure 6.3 Merged Port Lane Reversal for Maximum Link Width of x2

Notes

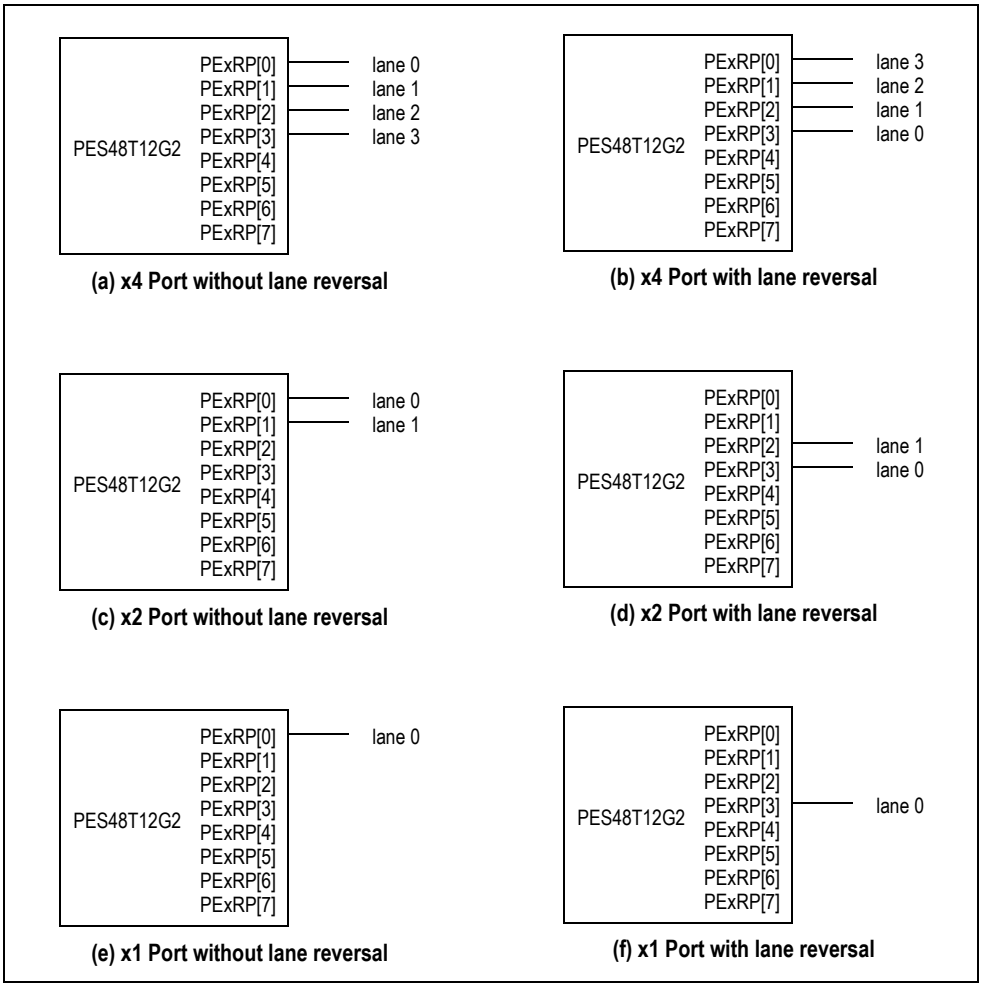


Figure 6.4 Merged Port Lane Reversal for Maximum Link Width of x4

Notes

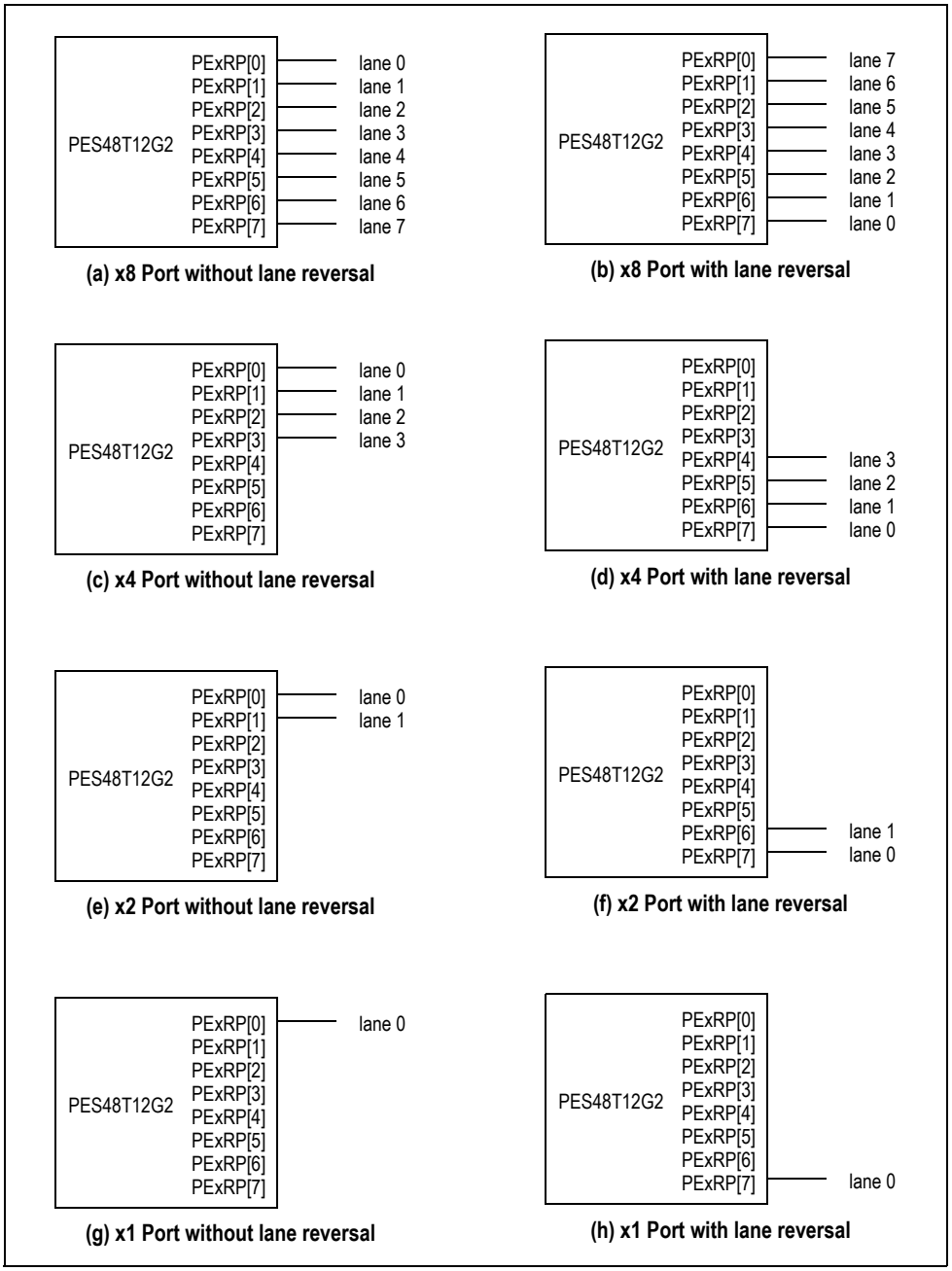


Figure 6.5 Merged Port Lane Reversal for Maximum Link Width of x8

Link Width Negotiation

The PES48T12G2 ports support the optional link variable width negotiation feature outlined in the PCI Express 2.0 specification. The Maximum Link Width (MAXLNKWDTH) field in a port's PCI Express Link Capabilities (PCIELCAP) register contains the maximum link width that the port can achieve. This field is of RWL type and may be modified when the REGUNLOCK bit is set in the 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 full link training occurs.

Notes

The actual link width is determined dynamically during link training. Ports limited to a maximum link width of x8 are capable of negotiating to a x8, x4, x2, or x1 link width. The current negotiated width of a link may be determined from the Negotiated Link Width (NLW) field in the corresponding port's PCI Express Link Status (PCIELSTS) register.

To force a link width to a smaller width than the default value, the MAXLNKWDTH field could be configured through Serial EEPROM initialization and full link retraining forced by setting the Full Link Retrain (FLRET) bit in the port's PHYLSTATE0 register.

For details, refer to the description of the MAXLNKWDTH field in the PCIELCAP register in Chapter 15.

Link Width Negotiation in the Presence of Bad Lanes

In an effort to maximize the link width when one or more lanes of a multi-lane link are not functioning correctly (i.e., reliable communication of training sets across the lane is not possible), PES48T12G2 downstream switch ports automatically attempt a lane reversed configuration when doing so has the potential to enhance the achievable link width.

For example, if lane 1 of a x4 link is not operating correctly, the device's downstream switch port attached to the link attempts a lane reversed configuration to form a x2 link using lanes 2 and 3 (Figure 6.4 (d)). If the link partner accepts the lane reversed configuration, the optimal x2 link will be formed using lanes 2 and 3. If the link partner does not accept the lane reversed configuration, but instead requests a lane configuration supported by the PES48T12G2 (e.g., x1 link using lane 0), the device accepts the configuration and forms the reduced width link. Otherwise, if the lane numbering agreement fails, the device automatically re-trains the link from the Detect state. During this re-training, the PES48T12G2 port does not reattempt a lane reversed configuration, but rather tries to form the link without reversing the lanes. As a result, a x1 link is formed using lane 0 (Figure 6.4 (e)).

Dynamic Link Width Reconfiguration

PES48T12G2 ports support dynamic link width upconfiguration and downconfiguration in response to link partner requests. This capability is honored for regular links and crosslinks.

PES48T12G2 ports do not initiate autonomous link width upconfiguration and downconfiguration of links, except for downconfiguration due to link reliability reasons. Therefore, the Hardware Autonomous Width Disable (HAWD) bit in the ports PCIELCTL register has no effect and is hardwired to 0x0. Additionally, the PES48T12G2 port's never set the 'Autonomous Change' bit in the training sets exchanged with the link partner during link training.[1]

A downstream port link partner may autonomously change link width. When this occurs, the PES48T12G2 downstream port sets the Link Autonomous Bandwidth Status (LABWSTS) bit in the PCIELSTS register.

Link Speed Negotiation

The PCI Express 2.0 specification introduces support for 5.0 GT/s data rates per lane (a.k.a., Gen2), in addition to the 2.5 GT/s data rates (a.k.a, Gen1) mandated in previous versions of the specification. Per the PCIe specification, all lanes of a link must operate at the same data rate. During full link training (i.e., from the Detect state), links initially operate at 2.5 GT/s. Once the LTSSM on both components of the link reach the L0 state and the data-link layer enters the DL_Active state, the link speed may be upgraded to 5.0 GT/s if this capability is advertised by both components. The process of upgrading the link speed does not result in a DL_Down state.

1. Note that the 'Autonomous Change' bit is located in bit 6 of the fourth symbol in the training sets. This bit has multiple meanings depending on the LTSSM state in which it is issued. PES48T12G2 never sets this bit in LTSSM states in which this bit carries the 'autonomous change' meaning.

Notes

A component advertises its supported speeds via the Data Rate Identifier bits in the TS1 and TS2 training sets transmitted to its link partner during link training. The PCIe spec permits a component to change its supported speeds dynamically. It is allowed for a component to advertise supported link speeds without necessarily changing the link speed, via the Recovery LTSSM state.

A component determines the supported speeds of its link partner by examining the Data Rate Identifier bits in the TS1/TS2 training sets received during link training, specifically in the Configuration.Complete and Recovery.RcvrCfg states. The last advertisement received overrides any previously recorded value.

Either link component may request a link speed change due to software requests or link reliability reasons (i.e., speed downgrade). Downstream components are further permitted to request link speed changes due to autonomous hardware initiated mechanisms. A component must only initiate a link speed change when it knows that its link partner supports the target speed via prior exchange of Training Sets. Gen2 support is optional while Gen1 support is mandatory.

If neither component in the link advertises support for Gen2, then the link remains operating in Gen1 speed. If one component has advertised support for Gen1 and Gen2, and the other has advertised support for Gen1 only, then the link will remain operating in Gen1 speed until the lesser speed component decides to:

- Advertise support for Gen2 via the Recovery state without modifying the link speed. The link remains operating at Gen1 speed.
- Transition the link speed to Gen2 via the Recovery.Speed state. The link will operate at Gen2 speed. In this case, the advertisement of Gen2 speed by both components is done implicitly in the Recovery substates entered while modifying the link speed.

It is the responsibility of the upstream component of the link (i.e., switch downstream ports) to keep the link at the target link speed or at the highest common speed supported by both components of the link, whichever is lower. In addition, the upstream component must initiate a link speed upgrade if it has recorded support for the higher speed by its link partner and software sets the Link Retrain bit in the PCIELCTL register with a target link speed which is not equal to the current link speed. The upstream component (i.e., switch downstream port) is capable of notifying software of link speed changes via the Link Bandwidth Notification mechanism described in the PCI Express 2.0 specification.

Link Speed Negotiation in the PES48T12G2

PES48T12G2 ports support data rates of 5.0 GT/s and 2.5 GT/s. The highest data rate of each link is determined dynamically, and depends on the following factors:

- Maximum link data rate supported by both components of the link
- The Target Link Speed set via the Link Control 2 Register (PCIELCTL2)
- The reliability of the link at 5.0 GT/s

By default, the Target Link Speed (TLS) of each port is set to 5.0 GT/s. Therefore, the PES48T12G2 ports advertise support for 2.5 GT/s and 5.0 GT/s during the link training process via training-sets.

- During normal operation, the TLS field should not be modified in an upstream port.

After a fundamental reset, each port link trains to the L0 state at 2.5 GT/s (Gen1). Once the data-link layer reaches the DL_Active state, if the Target Link Speed indicates 5.0 GT/s (default value), the switch's downstream ports automatically initiate link speed upgrade to 5.0 GT/s (Gen2) using the link speed change mechanism described in PCI Express 2.0 specification. Upstream ports do not automatically initiate link speed upgrade to Gen2.

- The Initial Link Speed Change Control (ILSCC) bit in a port's PHYLCFG0 register controls whether the port automatically initiates a speed upgrade to Gen2. If the ILSCC bit is set, the port does not automatically initiate a speed change to Gen2. Software may modify this bit to change the default behavior.
- The Link Bandwidth Management Status (LBWSTS) bit in the PCIELSTS register of downstream ports is not set since the initial link speed upgrade is not caused by a software directed link retrain or due to link reliability issues.

Notes

The current link speed of each port is reported via the Current Link Speed (CLS) field of the port's Link Status Register (PCIELSTS). The above behavior applies after full link retrain (i.e., when the LTSSM transitions through the 'Detect' state). Assuming the target link speed is set to 5.0 GT/s, a PES48T12G2 port initiates a link speed upgrade in the following cases:

- Link speed upgrade after initial link train (i.e., from the Detect state) to L0 at 2.5 GT/s, when the link partner advertised support for the higher speed.
- Link speed upgrade after full link retrain (i.e., via the Detect state) to L0 at 2.5 GT/s, when the link partner advertised support for the higher speed.

Notes

– When software sets the Link Retrain (LRET) bit in the PCIELCTL register and the PES48T12G2 port has recorded support for the higher speed by its link partner.[1]

When operating at 5.0 GT/s, a PES48T12G2 port initiates a link speed downgrade in the following cases:

– When the PHY layer cannot achieve reliable operation at the higher speed. In this case, the switch's port continues to support the higher speed in the training-sets it transmits during link training.
– When software sets the target link speed to 2.5 GT/s and sets the LRET bit in the PCIELCTL register. In this case, the switch's port removes support for the higher speed in the training-sets it transmits during link training.

Additionally, the PES48T12G2 ports always respond to link partner requests to change speed. In this case, the speed change is only successful when both components in the link advertise support the target speed. When a link speed upgrade operation fails, the PHY LTSSM reverts back to the speed before the upgrade (i.e., 2.5 GT/s) and does not autonomously initiate a subsequent link speed upgrade. In this case, the PHY continues to support Gen1 and Gen2 data rates and therefore responds to link partner requests for link speed upgrade, or to link speed upgrades triggered by software setting the LRET bit in the port's PCIELCTL register.

The PES48T12G2 ports do not have a mechanism to autonomously regulate link speed. As a result, the Hardware Autonomous Speed Disable (HASD) bit in the PCIELCTL2 register has no effect and is hardwired to 0x0. Additionally, PES48T12G2 ports never set the 'Autonomous Change' bit in the training sets exchanged with the link partner during link training[2]. Still, a link partner connected to a PES48T12G2 downstream port may autonomously change link speed. When this occurs, the PES48T12G2 downstream port sets the Link Autonomous Bandwidth Status (LABWSTS) bit in the PCIELSTS register.

A system designer may limit the maximum speed at which each port operates by changing the target link speed via software or EEPROM and forcing link retraining. Refer to section Link Retraining on page 6-10 for further details.

Software Management of Link Speed

Software can interact with the link control and status registers of downstream ports to set the link speed, as well as receive notification of link speed changes. This gives software the capability to choose the desired link speed based on system specific criteria. For example, depending on the traffic load expected on a link, software can choose to downgrade link speed to 2.5 GT/s in order to reduce power on a low-traffic link, and later upgrade the link to 5.0 GT/s when the bandwidth is required. Software may also choose to change the link speed due to link reliability reasons (i.e., a link that has reliability problems at 5.0 GT/s may be downgraded to 2.5 GT/s).

As mentioned above, the Target Link Speed (TLS) field of the port's Link Control 2 Register (PCIELCTL2) sets the preferred link speed. By default, the Target Link Speed of each PES48T12G2 port is set to 5.0 GT/s. During normal operation, the link speed of a downstream port may be modified by setting the TLS field of the port's PCIELCTL2 register to the desired speed and initiating link retraining by writing a one to the Link Retrain (LRET) bit in the Link Control (PCIELCTL) register.

– The port will only initiate a change to a higher speed if the link partner advertised support for the higher speed in its latest entry to the Configuration.Complete or Recovery.RcvrCfg states.
– If a speed change is initiated to a speed not supported by the link partner, then the port will remain at the current speed by transitioning through the Recovery state without the "Speed_Change" bit set.

1. The speed advertisement of the link partner is noted by PES48T12G2 in the latest LTSSM entry to the Configuration.Complete or Recovery.RcvrCfg sub-states.

2. Note that the 'Autonomous Change' bit is located in bit 6 of the fourth symbol in the training sets. This bit has multiple meanings depending on the LTSSM state in which it is issued. PES48T12G2 never sets this bit in LTSSM states in which this bit carries the 'autonomous change' meaning.

Notes

Notification of link speed changes if provided through the link bandwidth notification mechanism described in the PCIe specification. This mechanism is enabled by setting the Link Bandwidth Management Interrupt Enable (LBWINTEN) bit in the PCIELCTL register of switch downstream ports. For downstream ports, the Link Bandwidth Management Status (LBWSTS) bit in the PCIELSTS register is set when the link speed is changed due to the following reasons:

- Link speed downgrade initiated by a PES48T12G2 port when the PHY layer cannot achieve reliable operation at the higher speed. Note that this does not include link speed downgrading due to failure to achieve symbol lock while trying to upgrade link speed via the Recovery state.
- Link speed change initiated by the link partner that was not indicated as an autonomous change.

Also, the LBWSTS bit is set whenever software sets the LRET bit in the PCIELCTL register, even if the link speed is not changed. Note that the LBWSTS bit is not set during the initial link speed change (i.e., the speed change from Gen1 to Gen2 after fundamental reset or a full-link-retrain via the 'Detect' state). Software can verify the link speed by reading the Current Link Speed (CLS) field of the port's Link Status Register (PCIELSTS).

Link Retraining

Per the PCI Express 2.0 specification, link retraining can be done autonomously in response to link problems (i.e., repeated TLP replay attempts), or as a result of software setting the link retrain (LRET) bit in the PCI Express Link Control (PCIELCTL) register.

Writing a one to the Full Link Retrain (FLRET) bit in the Phy Link State 0 (PHYLSTATE0) register of any port forces that port's PCIe link to retrain. When this occurs, the LTSSM transitions directly to the Detect state.

Link retraining does not result in the link going down, unless the LTSSM transitions through the Detect state in its retraining attempt. The speed of the link is not necessarily changed as a result of link retraining. A link that operates at 5.0 GT/s will continue to operate at that speed if the link retraining attempt is successful at that speed. Else, the link speed is changed to 2.5 GT/s.

When link retraining results in the speed of the link being downgraded from 5.0 GT/s to 2.5 GT/s, the Link Bandwidth Management Status (LBWSTS) bit is set in the PCI Express Link Status (PCIELSTS) register. Additionally, the PHY LTSSM remains at the downgraded speed until the link partner requests a link speed upgrade, software sets the LRET bit in the PCIELCTL register, or the link is fully retained via the FLRET bit in the PHYLSTATE0 register.

In addition to link retrain (via the Recovery state), the link may be fully retrained by writing a one to the Full Link Retrain (FLRET) bit in a port's Phy Link State 0 (PHYLSTATE0) register. When this occurs the LTSSM transitions directly to the Detect state. This causes the data-link to go down (refer to the Link Down section below).

Note that the LBWSTS bit in the PCIELSTS register is not affected by a full link retrain (i.e., since the data-link transitioned to the DL_Down state).

Link Down

When an upstream port's link goes down, it triggers a hot reset, as described in section Switch Fundamental Reset on page 5-2. In addition:

- All TLPs queued in the port's ingress frame buffer (IFB) are silently discarded.
- All TLPs queued in the port's replay buffer (EFB) are silently discarded.

When a downstream port's link goes down (i.e., the data-link layer transitions to the DL_Down state), the following occurs:

- All TLPs queued in the port's ingress frame buffer (IFB) are silently discarded.
- All TLPs queued in the port's replay buffer (EFB) are silently discarded.
- Request TLPs received by other ports and destined to the logical bus number associated with the link that is down are treated as Unsupported Requests (UR).
- All other TLPs received by the other ports and destined to the logical bus number associated with the link that is down are silently discarded.

Notes

- The port handles all TLPs that target the port's function(s) normally.
  - It is possible to perform configuration read and write operations to port.

When a link comes up, flow control credits for the configured size of the port's IFB queues are advertised. A link down condition on a downstream port's link may cause the Surprise Down Error Status (SDOENERR) bit to be set in the port's AER Uncorrectable Error Status (AERUES) register. The conditions under which surprise down is reported are described in Section 3.2.1 of the PCI Express 2.0 Specification.

In addition to the exception conditions listed in Section 3.2.1of the PCI Express 2.0 specification, the SDOENERR bit in a port's AERUES register is not set in the following cases:

- The port's link is fully retrained via the FLRET bit in the PHYLSTATE0 register.
- The port's clocking mode is modified (see section Port Clocking Mode on page 4-1).

Slot Power Limit Support

The Set_Slot_Power_Limit message is used to convey a slot power limit value from a downstream switch port or root port to the upstream port of a connected device or switch.

Upstream Port

When a Set_Slot_Power_Limit message is received by an upstream port, then the fields in the message are written to the PCI Express Device Capabilities (PCIEDCAP) register of that port.

- Byte 0 bits 7:0 of the message payload are written to the Captured Slot Power Limit Scale (CSPLS) field.
- Byte 1 bits 1:0 of the message payload are written to the Captured Slot Power Limit Value (CSPLV) field.

Downstream Port

A Set_Slot_Power_Limit message is sent by downstream switch ports when either of the following events occur.

- 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., data-link layer up) state.

Link States

PES48T12G2 ports support 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 PME_Turn_Off Message.

  There is no TLP or DLLP communications over a link in this state.

  Note that in this state, the link is considered 'up'.
- L3

  Link is completely unpowered and off
- Link Down

  A transitional link down pseudo-state prior to L0. This pseudo-state is associated with the LTSSM Detect, Polling, Configuration, Disabled, Loopback and Hot-Reset states.

Notes

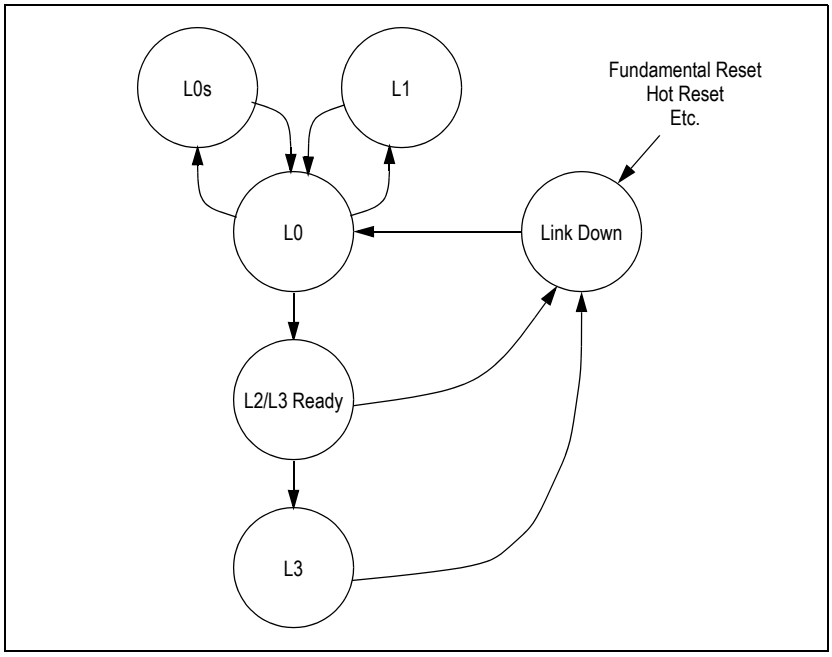


Figure 6.6 PES48T12G2 ASPM Link Sate Transitions

Active State Power Management

The operation of link Active State Power Management (ASPM) is orthogonal to device power management. Once ASPM is enabled, ASPM link state transitions are initiated by hardware without software involvement.

The PES48T12G2 ASPM supports the required receiver L0s state as well as the optional transmitter L0s and L1 states. ASPM is enabled via the ASPM field in the function's link control register (PCIELCTL).

L0s ASPM

L0s entry/exit operates independently for each direction of the link. On the receive side, the PES48T12G2 upstream and downstream ports always respond to L0s entry/exit requests from the link partner. On the transmit side, the L0s entry conditions must be met for 7us before the hardware transitions the transmit link to the L0s state.

L0s Entry Conditions

The transmit side L0s entry conditions depend on the port's operational state. A port configured in 'Upstream Switch Port' mode initiates L0s entry when all of the conditions listed below are met:

- L0s ASPM is enabled via the port's PCIELCTL register.
- The following conditions are met for the amount of time specified in the above field:
  - The receive lanes of all of the switch downstream ports which are not in a low power state (i.e., D3) and whose link is not down are in the L0s state.
  - The port has no TLPs to transmit or there are no available flow control credits to transmit a TLP.
  - The port has no DLLPs pending for transmission.

A port configured in 'Downstream Switch Port' mode initiates L0s entry when all of the conditions listed below are met:

- L0s ASPM is enabled via the port's PCIELCTL register.
- The following conditions are met for the amount of time specified in the above field:
  - The receive lanes of the switch's upstream port are in the L0s state.
  - The switch has no TLPs to transmit on the downstream port or there are no available flow control credits to transmit a TLP.
  - There are no DLLPs pending for transmission on the downstream port.

Notes

L0s Exit Conditions

The transmit side L0s exit conditions depend on the port's operational state. A port configured in 'Upstream Switch Port' mode initiates exit from L0s when any of the conditions listed below are met:

- The port has a TLP or DLLP scheduled for transmission.
- A downstream port in the switch has initiated exit from L0s.

A port configured in 'Downstream Switch Port' mode initiates exit from L0s when any of the conditions listed below are met:

- The port has a TLP or DLLP scheduled for transmission.
- The upstream port in the switch has initiated exit from L0s.

L1 ASPM

L1 entry/exit is initiated by downstream link components (i.e., upstream ports) and affects both directions of the link. Upstream link components (i.e., downstream ports) accept or reject L1 entry requests from the link partner.

L1 Entry Conditions

The PES48T12G2 downstream ports may accept or reject L1 entry requests sent by the link partner. A port configured in 'Downstream Switch Port' mode accepts L1 entry requests when all of the conditions listed below are met. Else, the L1 entry request is rejected.

- L1 ASPM is enabled via the port's PCIELCTL register.
- The port has no TLPs pending for transmission.
- The port has no ACK or NAK DLLPs pending for transmission.

The PES48T12G2 upstream ports request entry into L1 based on the criteria defined below. The L1 entry conditions must be met for 1 ms before the upstream port 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. If the link partner acknowledges the transition, then the L1 state is entered. Otherwise, L0s entry is attempted[1] A port configured in 'Upstream Switch Port' mode initiates L1 entry when all of the conditions listed below are met:

- L1 ASPM is enabled via the port's PCIELCTL register.
- All of the downstream ports which are not in a low power state (i.e., D3) and whose link is not down are in the L1 state.
- The port has no TLPs pending for transmission.

  The port's replay-buffer is empty.
- The port has no DLLPs pending for transmission.
- The port's receiver is idle (i.e., no TLPs or DLLPs are received) for the amount of time specified above.
- The port has accumulated enough flow-control header and data credits to transmit the largest possible packet of each type (i.e., posted, non-posted, or completion).

L1 Exit Conditions

The L1 exit conditions depend on the port's operational state. A port configured in 'Upstream Switch Port' mode initiates exit from L1 when any of the conditions listed below are met:

- The port has a TLP scheduled for transmission.
- A downstream port in the switch has initiated exit from L1.

  The latency between the downstream port's initiated exit from L1 and the upstream port's initiated exit from L1 must not exceed 1 µs.

A port configured in 'Downstream Switch Port' mode initiates exit from L1 when any of the conditions listed below are met:

- The port has a TLP scheduled for transmission.
- The upstream port in the switch has initiated exit from L1.

  The latency between the upstream port's initiated exit from L1 and the downstream port's initiated exit from L1 must not exceed 1 µs.

1. L0s entry is subject to the rules specified in section L0s ASPM on page 6-12.

Notes

L1 ASPM Entry Rejection Timer

When enabled by the ASPM field in the PCI Express Link Control (PCIELCTL) register, PES48T12G2 downstream ports respond to link partner requests to enter the L1 ASPM state. In order to enter the L1 ASPM link state, a downstream device (i.e., endpoint) sends continuous PM_Active_State_Request_L1 DLLPs to its link partner (i.e., a downstream switch port). This process continues until the downstream device receives an acceptance or rejection from its link partner.

A PES48T12G2 downstream port can choose to accept or reject the request, depending on a variety of conditions (refer to section L1 Entry Conditions on page 6-13). When accepting a request, the PES48T12G2 downstream port sends continuous PM_Request_Ack DLLPs until the downstream device receives these and sends an electrical idle ordered set, effectively placing the link in L1 state.

When rejecting a request, the PES48T12G2 downstream port sends a single PM_Active_State_Nak TLP. The downstream device, upon reception of this TLP, should place its transmitter into the L0s state, and exit this state prior to sending a new L1 ASPM entry request. Optionally, the downstream device may keep the link in L0 state, in which case it must wait at least 10 µs before sending a new L1 ASPM entry request.

Some endpoint devices do not meet the required 10 µs gap between consecutive L1 ASPM entry requests. A live-lock situation can develop in the following scenario:

- The Endpoint sends continuous PM_Active_State_Request_L1 DLLPs to the downstream port of a switch.
- The switch receives the request but decides to reject (i.e., due to a TLP already queued for transmission on this link). The switch sends a PM_Active_State_Nak TLP to the endpoint device.
- The endpoint device notices the rejection, waits an amount of time (i.e., 8 µs) and resumes transmission of PM_Active_State_Request_L1 DLLPs.
- The switch receives PM_Active_State_Request_L1 DLLPs, but does not recognize them as a new L1 ASPM entry request, since there was a violation of the 10 µs gap between L1 ASPM entry requests.
- The switch does not respond with an acceptance or rejection. Therefore, the endpoint keeps waiting for an acceptance or rejection. A live-lock condition develops.

To avoid this live-lock condition, PES48T12G2 downstream ports allow programmability of a timer that checks for the 10 µs gap between L1 ASPM entry requests. There is a timer per port. The Minimum Time between L1 Entry Requests (MTL1ER) field in the L1 ASPM Rejection Timer Control (Px_L1ASPMRTC) register may be programmed for this purpose.

This timer may be programmed from the nano-second range (i.e., 100 ns) up to the micro-second range (i.e., 64 µs). By default, the timer is set to 9.5 µs (refer to the Implementation note in Section 5.4.1.2 of the PCI Express 2.0 spec).

Normally, this timer starts its count after the switch downstream port issues an L1 ASPM rejection (i.e., PM_Active_State_Nak TLP), without checking activity on the link. The PES48T12G2 also provides an option to start the timer after the downstream port issues an L1 ASPM rejection (i.e., PM_Active_State_Nak TLP) and no activity is detected on the receive-lanes. The Timer Start Control (TSCTL) in the L1ASPMRTC register controls this behavior. This feature allows the PES48T12G2 downstream ports to enter L1 ASPM with a variety of endpoints, even those that don't meet the 10 µs gap between subsequent L1 ASPM entry requests.

Link Status

Associated with each PES48T12G2 port is a Port Link Up (PxLINKUPN) status output and a Port Activity (PxACTIVEN) status output. These outputs are provided on an I/O expander. The PxLINKUPN and PxACTIVEN status outputs may be used to provide a visual indication of system state and activity or for debug. The PxLINKUPN output is asserted when the port's data link layer is up (i.e., when the LTSSM is in the L0, L0s, L1 or recovery states). When the data link layer is down, this output is negated.

The PxACTIVEN output is asserted whenever any TLP, other than a vendor defined message, is transmitted or received on the corresponding port's link. Whenever a PxACTIVEN output is asserted, it remains asserted for at least 200 ms. Since an I/O expander output may change no more frequently than once every 40 ms, this translates into five I/O expander update periods.

Notes

De-emphasis Negotiation

The PCI Express 2.0 specification requires that components support the following levels of de-emphasis, depending on the link data rate:

- 2.5 GT/s (Gen1): De-emphasis = -3.5dB
- 5.0 GT/s (Gen2): De-emphasis = -3.5dB or -6.0dB

When operating at 5.0 GT/s, the de-emphasis is selected by programming the Selectable De-emphasis (SDE) field in the port's PCI Link Control 2 Register (PCIELCTL2). The chosen de-emphasis for the link is the result of a negotiation between the two components of the link. Both components must operate with the same de-emphasis across all lanes of the link.

During normal operation (i.e, not polling.compliance), de-emphasis selection is done during the Recovery state. The downstream component of the link (i.e., switch upstream port or endpoint) advertises its desired de-emphasis by transmission of training sets. The upstream component of the link (i.e., switch downstream port or root-complex port) notes its link partner desired de-emphasis, and makes a decision about the de-emphasis to be used in the link.

The PES48T12G2's upstream port physical layer advertises its desired de-emphasis based on the setting of the port's SDE field in the PCIELCTL2 register. The upstream port always accepts the link-partners decision on the de-emphasis to be used in the link. The PES48T12G2's downstream ports ignore the link partner's desired de-emphasis, and always choose the de-emphasis setting in the SDE field of the port's PCIELCTL2 register.

Crosslink

The PES48T12G2 ports support the optional crosslink capability specified in the PCI Express Base Specification 2.0. Per this specification, a crosslink is established between two downstream ports or two upstream ports. The PES48T12G2 ports are capable of establishing crosslink with any link partner, including another PES48T12G2 port.

When crosslink is formed between two ports, the physical layer of one of the ports operates as an upstream component (e.g., downstream lanes) while the physical layer of the other port operates as a downstream component (e.g., upstream lanes). The determination of which of the two ports that form the crosslink operates with upstream or downstream lanes depends on the link training process as specified in the PCI Express Base Specification.

The Link Mode (LINKMODE) field in the SWPORTxSTS register associated with the PES48T12G2 port indicates the current link mode (i.e., upstream or downstream lanes) of the port's physical layer. When two PES48T12G2 ports are crosslinked to each other, it is recommended that the crosslink connection be done among ports in different port groups, as shown in Table 6.1. In order for ports in the same port group (e.g., port 0 and port 4, port 3 and port 7, etc.) to form a crosslink, software must set the SEED field in the crosslinked port's Phy PRBS Seed (PHYPRBS) register to different values.

Port Groups			
Group 0	**Group 1**	**Group 2**	**Group 3**
0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

Table 6.1 Crosslink Port Groups

Notes

Note that when a PES48T12G2 upstream port is crosslinked to a link-partner's upstream port, neither port may automatically initiate a link speed change to Gen 2, thereby resulting in a Gen 1 link. It is possible to overcome this by setting the ILSCC bit in the PES48T12G2 upstream port's PHYLCFG0 register. By setting this bit, the PES48T12G2 upstream port will initiate the link transition to Gen 2 speed.

Crosslink is enabled by default. Crosslink may be disabled by setting the Crosslink Disable (CLINKDIS) bit in the port's Phy Link Configuration 0 (PHYLCFG0) register.

Hot Reset Operation on a Crosslink

When a PES48T12G2 port forms a crosslink, hot reset operates as follows.

- For a port operating in downstream switch port mode:
  - Regardless of the physical layer's mode of operation (i.e., upstream or downstream lanes), the physical layer responds to the reception of training sets with the hot reset bit set by transitioning to the hot reset state as specified in the PCI Express Base Specification. The hot reset does not reset the configuration registers of the port and does not affect other ports in the partition.
  - If the port's physical layer operates as 'downstream lanes' and a higher layer directs the port to the hot reset state (e.g., partition hot reset, upstream secondary hot reset, downstream secondary hot reset), the physical layer enters the recovery state and proceeds to the hot reset state, as specified in the PCI Express Base Specification.
  - If the port's physical layer operates as 'upstream lanes', the PES48T12G2 provides no higher layer mechanism to direct the physical layer to enter the hot reset state. This implies that in a crosslink formed by a PES48T12G2 downstream port whose physical layer has trained as 'upstream lanes', hot reset across the link may only be propagated by the link partner's port (which must have trained as a downstream port with downstream lanes).
- For a port operating in upstream switch port mode:
  - There is no higher layer mechanism to place the port in hot reset state.
  - Regardless of the physical layer's mode of operation (i.e., upstream or downstream lanes), the physical layer responds to the reception of training sets with the hot reset bit set by transitioning to the hot reset state. The hot reset has the effect described in section Hot Resets on page 5-5.

Link Disable Operation on a Crosslink

When a port is crosslinked, link disable operates as follows.

- For a port operating in downstream switch port mode:
  - Regardless of the port's physical layer mode of operation (i.e., downstream lanes or upstream lanes):

    If a higher layer directs the port to disable the link (i.e., the Link Disable (LDIS) bit is set in the port's PCIELCTL register), the physical layer enters the recovery state and proceeds to the disabled state, as specified in the PCI Express Base Specification.

    The physical layer responds to the reception of training sets with the disabled bit set by transitioning to the disabled state as specified in the PCI Express Base Specification.
- For a port operating in upstream switch port mode:
  - There is no higher layer mechanism to place the port's link in the disabled state.
  - Regardless of the port's physical layer mode of operation (i.e., downstream lanes or upstream lanes), the physical layer responds to the reception of training sets with the disabled bit set by transitioning to the disabled state as specified in the PCI Express Base Specification.

Gen1 Compatibility Mode

The PES48T12G2 ports may be configured to operate in 'Gen1 Compatibility Mode'. The intent of this mode is to overcome interoperability problems that arise when PCI Express 2.0 devices link train with devices that conform to the PCIe 1.1 or earlier specifications (a.k.a., Gen1 devices). Specifically, this mode overcomes the problem in which Gen1 devices react incorrectly to newly defined bits in the PCI Express 2.0 specification for the PHY training sets. Such bits include bits 2, 6, and 7 in symbol four of the TS1 and TS2 training sets.

Notes

A PES48T12G2 port is placed in Gen1 Compatibility Mode by setting the Gen1 Compatibility Mode Enable (G1CME) bit in the PHYLCFG0 register and fully retraining the link (i.e., via the FLRET bit the PHYLSTATE0 register).

When a PES48T12G2 port operates in Gen1 Compatibility Mode, the PHY does not set the following bits in Table 6.2 in the training sets that it transmits

Training Set	Symbol	Bit	PCIe 1.1 and earlier Definition	PCI Express 2.0 Definition
TS1	4	2	Reserved	5.0 GT/s Data Rate Support
		6		Multiple meanings (refer to PCI Express 2.0 Specification)
		7		Speed Change
TS2	4	2	Reserved	5.0 GT/s Data Rate Support
		6		Multiple meanings (refer to PCI Express 2.0 Specification)
		7		Speed Change

Table 6.2 Gen1 Compatibility Mode: bits cleared in training sets

A PES48T12G2 port exits Gen1 Compatibility Mode by clearing the G1CME field in the PHYLCFG0 register and fully retraining the link (i.e., via the FLRET bit the PHYLSTATE0 register). When this occurs, the training set bits listed in Table 6.2 behave per the PCI Express 2.0 definition.

Notes

SerDes

Notes

Introduction

This chapter describes the controllability of the Serializer-Deserializer (SerDes) block associated with each PES48T12G2 port. A SerDes block is composed of the serializing/deserializing logic for four PCI Express lanes (i.e., a SerDes "quad"), plus a central block that controls the quad as a whole. This central block is called CMU, and contains functionality such as a PLL to generate a high-speed clock used by each lane, initialization of the quad, etc.

In order to improve signal integrity across the high-speed PCI Express links, the PES48T12G2 allows per-lane programmability of several SerDes settings. These include the following.

- Transmitter drive level
- Transmitter de-emphasis level
- Transmitter slew rate
- Receiver equalization

In addition, the PES48T12G2 supports the optional "low-swing mode" specified by the PCI Express 2.0 specification. This mode is intended for power-sensitive applications.

This chapter describes these controls, their intended use, and the manner in which they are programmed. Before this is discussed, the topic of SerDes numbering and port association is introduced. To modify the SerDes driver and receiver settings for a port, the SerDes quad and specific lanes associated with the port must be identified as described in the next section.

SerDes Numbering and Port Association

The PES48T12G2 contains twelve SerDes quads, numbered zero to 13 (omitting ports 10 and 11 which do not exist in this device). A SerDes quad is normally associated with its corresponding numbered port (i.e., SerDes quad 0 is associated with port 0, SerDes quad 1 is associated with Port 1, and so on).

A x4 port is always associated with its corresponding SerDes quad.

A x8 (merged) port is composed of an even numbered port and its odd counterpart. This port is always associated with the two corresponding SerDes quads (i.e., a merged port 0 is associated with SerDes 0 and SerDes 1, merged port 2 is associated with SerDes 2 and SerDes 3, etc.).

SerDes Transmitter Controls

The PES48T12G2 allows programmability of SerDes transmitter voltage level, de-emphasis, and slew (i.e., signal rise and fall times), including support for the PCI Express optional low-swing mode, as well as a proprietary "amplitude boost" feature to increase the drive strength above its normal operating level (e.g., for operation across long traces).

Except for low-swing mode, which is defined by PCI Express as a per-link function, all the other controls are proprietary and provided on a per-lane basis. This allows a system designer to customize the SerDes transmitter settings for each lane independently. At the 5.0 GT/s speed (i.e., Gen2) and above, small differences in the channel characteristics among lanes may result in noticeable differences in the quality of the signal at the receiver and per-lane controllability is an important tool in improving the bit-error rate on the link.

Driver Voltage Level and Amplitude Boost

The PCI Express 2.0 specification requires that each port support the 'transmit margining' feature. This feature allows the selection of several voltage settings across the link and is intended for compliance testing and debug.

Notes

In addition to this, the PES48T12G2 offers proprietary fine grain controllability of the SerDes transmitter voltage level, across a wide range of settings. The PES48T12G2 places no restrictions on the time at which these settings can be modified (e.g., they can be modified during normal operation of the link or while the link is being tested).

By default, the SerDes transmit level can be programmed in the range from 950 mV to 120 mV, at steps of ~40 mV each[1]. In addition, there is an "amplitude-boost" control that monotonically increases the drive level by ~5% for each of four settings. By default, this control is set to the second level (out of four), meaning that the range shown above can be increased by up to 10% when the maximum boost setting is used, or decreased by 5% when the minimum boost setting is used.

Together, the controls for drive-level and amplitude boost allow the system designer to select, on a per-lane basis if desired, the appropriate drive strength for the channel. For power sensitive applications, the drive level can be reduced with fine granularity to the desired level, without compromising link reliability.

Note that the PCI Express specification requires that, at 5.0 GT/s, receivers accept incoming signals in the range 1.2 V to 0.120 V. Thus, the transmitter voltage settings may be modified without requiring any modification of the link partner's receiver settings. Refer to section Programming of SerDes Controls on page 7-3 for procedural details on modifying the default SerDes settings and to section SerDes Transmitter Control Registers on page 7-4 for details on programming the transmitter voltage level and amplitude boost controls.

De-emphasis

The PCI Express 2.0 specification supports three de-emphasis levels: -3.5 dB (at 2.5 GT/s or 5.0 GT/s speeds), -6.0 dB (only for 5.0 GT/s), and 0 dB (low-swing mode). The de-emphasis selected for the link is controlled by the Selectable De-emphasis bit in each port's PCI Express Link Control 2 register[2]. This field is set by hardware or firmware (e.g., EEPROM) during boot time and remains unchanged during normal system operation.

To allow the de-emphasis setting to be modified and customized on the link, the PES48T12G2 contains proprietary per-lane coarse and fine de-emphasis adjustment controls. Together, these controls allow the nominal de-emphasis setting (i.e., -3.5 dB or -6.0 dB) to be modified with a granularity of ~0.3 dB per setting[3]. The desired de-emphasis setting can be achieved across the range of driver level settings described in the previous section. The PES48T12G2 places no restrictions on the time at which the de-emphasis setting can be modified.

Refer to section SerDes Transmitter Control Registers on page 7-4 for details on programming the transmitter de-emphasis.

Slew Rate

In addition to transmitter drive level and de-emphasis controls, the PES48T12G2 ports also contain proprietary fine and coarse slew rate controls. These provide controllability over the rise and fall times of the transmitted differential signal, and are useful in scenarios where slow rise and fall times negatively affect the signal eye. The controls for the slew rate are provided on a per-lane basis, and the PES48T12G2 places no restrictions on the time at which they can be modified. Refer to section SerDes Transmitter Control Registers on page 7-4 for details on programming the slew rate controls.

PCI Express Low-Swing Mode

The PES48T12G2 ports support the optional low-swing transmit voltage mode defined in the PCI Express 2.0 specification. In this mode, the port's transmitter voltage level is set to approximately half the value of the full-swing (default) mode, which results in reduced power consumption in the SerDes. In addi-

1. Values are based on HSPICE simulation results, assuming typical conditions, as measured at the pin of the device.
2. In low-swing mode, de-emphasis is automatically set to 0 dB and the Selectable De-emphasis bit in the port's PCI Express Link Control 2 register is ignored.
3. Note that the PCI Express specification allows a deviation of +/- 0.5 dB from the nominal setting.

Notes

tion, signal de-emphasis is turned off. Low-swing mode is a per-link feature, meaning that all lanes of the port operate low-swing simultaneously. Refer to section Low-Swing Transmitter Voltage Mode on page 7-13 for details on enabling low-swing mode on a port.

Receiver Equalization

In addition to the transmitter controls described above, the PES48T12G2 SerDes also contains a receiver equalizer to compensate for effects of channel loss on received signal (i.e., high-speed signal degradation due to the combined effects of board traces, vias, connectors, and cables in the physical link). In general, the channel has low-pass filter characteristics, which results in the degradation of high speed signals. Receiver equalization may be used to compensate for the lossy attenuation effects of the channel on high-speed signals.

Receiver equalization can be controlled on a per-lane basis. Each SerDes lane contains a receiver equalization circuit. This circuit is a multi-stage programmable amplifier, where each stage is a peaking equalizer with a different center frequency and programmable gain. Varying amounts of gain may be applied depending on the overall frequency response of the channel loss.

For details on programming the receiver equalizer, refer to section Receiver Equalization Controls on page 7-14. The PES48T12G2 places no restrictions on the time at which the equalizer settings may be modified (e.g., the settings can be modified during normal operation of the link or while the link is being tested).

Programming of SerDes Controls

The SerDes controls described above may be programmed by accessing IDT proprietary registers within the PES48T12G2 switch. The registers may be programmed via any of the mechanisms allowed by the PES48T12G2 (i.e., via PCI Express configuration accesses from a root, via EEPROM loading at boot-time, or via the PES48T12G2's SMBus slave interface).

The following sections describe in detail the control registers associated with the SerDes and the manner in which the SerDes controls are programmed.

Programmable Voltage Margining and De-Emphasis

The PES48T12G2 contains SerDes transmitter voltage controls on a per-port, per-SerDes, and per-lane basis. There are two mechanisms to control the SerDes transmitter voltage level:

- Via the Transmit Margin (TM) field of the associated port's Link Control 2 Register (PCIELCTL2).
- Via the SerDes transmitter control registers
  - Each SerDes quad has independent transmitter control registers
  - The SerDes Lane Transmitter Control Registers (S[x]TXLCTL0 and S[x]TXLCTL1) provide transmit driver controls per-lane.[1]
  - S[0]TXLCTL0 and S[0]TXLCTL1 are associated with SerDes 0, S[1]TXLCTL0 and S[1]TXLCTL1 are associated with SerDes 1, and so on.

The selection of which of the two mechanism controls the SerDes transmit voltage is based on the setting of the TM field in the associated port's PCIELCTL2 register. The port associated with a SerDes quad is subject to the rules in section SerDes Numbering and Port Association on page 7-1.

1. The S[x]TXLCTL0 and S[x]TXLCTL1 registers are used in conjunction with the SerDes Control (S[x]CTL) register in order to apply the settings to a particular lane or all lanes of the SerDes. Please refer to the description of the S[x]CTL register for further details.

Notes

When the Transmit Margin (TM) field in the port's PCIELCTL2 register is set to 'Normal Operating Range', the transmitter voltage level for each SerDes lane of the port is controlled via the S[x]TXLCTL0 and S[x]TXLCTL1 registers. Otherwise, the TM field controls the SerDes voltage directly for all SerDes lanes of the port.

- For instance, port 0 is associated with SerDes quad 0.[1] If the TM field in the port's PCIELCTL2 register is set to 'Normal Operating Range', then the S[0]TXLCTL0 and S[0]TXLCTL1 registers control the operating voltage of the port's SerDes. If the TM field is set to another value, the SerDes voltage is set to the value in the port's PCIELCTL2.TM field.
- Also, if port 4 operates in merged mode, it is associated with SerDes quads 4 and 5. If the TM field in the port's PCIELCTL2 register is set to 'Normal Operating Range', then the S[4:5]TXLCTL0 and S[4:5]TXLCTL1 registers control the operating voltage of the port's SerDes. If the TM field is set to another value, the SerDes voltage of both SerDes quad 4 and SerDes quad 5 is set to the value in the PCIELCTL2.TM field of port 4.

De-emphasis levels may also be adjusted on a per-lane basis, using the above mentioned transmitter control registers. Nominally, de-emphasis levels are set to -3.5 dB, -6.0 dB, or 0 dB (in low-swing mode). The S[x]TXLCTL0 and S[x]TXLCTL1 registers can be used to modify the nominal values by coarse or fine steps.

SerDes Transmitter Control Registers

As described above, each SerDes quad is associated with two transmitter control registers (S[x]TXLCTL0 and S[x]TXLCTL1). Together, these registers allow full programmability of the SerDes transmitter voltage levels and de-emphasis. These registers are segmented into fields that allow programmability of the transmit driver levels under the following PHY operating modes:

- Full-Swing Mode, in Gen1 data rate, with -3.5 dB de-emphasis
- Full-Swing Mode, in Gen2 data rate, with -3.5 dB de-emphasis
- Full-Swing Mode, in Gen2 data rate, with -6.0 dB de-emphasis
- Low-Swing Mode, in Gen1 data rate (no de-emphasis)
- Low-Swing Mode, in Gen2 data rate (no de-emphasis)

The S[x]TXLCTL0 and S[x]TXLCTL1 registers have default values that select the appropriate transmit driver settings for each of the above modes. These default values may be modified to adjust the drive levels.

When the Physical layer of the port associated with the SerDes transitions dynamically across these operating modes, the appropriate driver settings are applied to the SerDes automatically. For example, when the PHY operates in Full-swing mode at Gen1 data rate with -3.5 dB de-emphasis, the SerDes transmit settings are set to the values specified in the S[x]TXLCTL0 and S[x]TXLCTL1 registers corresponding to that operating mode. As the PHY changes data rate to Gen2, the SerDes transmit settings are automatically modified to the values specified in the S[x]TXLCTL0 and S[x]TXLCTL1 registers corresponding to the new operating mode (e.g., Full-Swing mode at Gen2 data rate with -3.5 dB de-emphasis).

Table 7.1 shows the register fields that control the SerDes transmit levels for the operation modes listed above.

[1] SerDes and Port association are subject to the rules outlined in section SerDes Numbering and Port Association on page 7-1.

Notes

PHY Operation Mode			Relevant fields in S[x]TXLCTL0		Relevant fields in S[x]TXLCTL1
Voltage Swing	**Data Rate**	**De-emphasis**	**Coarse De-emphasis Control & Transmitter Equalization**	**Slew Rate Control (Course & Fine)**	**Drive Level / Fine-De-emphasis Control**
Full-Swing	2.5 GT/s	-3.5 dB	CDC_FS3DBG1 TX_EQ_3DBG1	TX_SLEW_G1 & TX_FSLEW_G1	TDVL_FS3DBG1 / FDC_FS3DBG1
Full-Swing	5.0 GT/s	-3.5 dB	CDC_FS3DBG2 TX_EQ_3DBG2	TX_SLEW_G2 & TX_FSLEWG2	TDVL_FS3DBG2 / FDC_FS3DBG2
Full-Swing	5.0 GT/s	-6.0 dB	CDC_FS6DBG2 TX_EQ_6DBG2		TDVL_FS6DBG2 / FDC_FS6DBG2
Low-Swing	2.5 GT/s	0 dB	N/A	TX_SLEW_G1 & TX_FSLEW_G1	TDVL_LSG1
Low-Swing	5.0 GT/s	0 dB	N/A	TX_SLEW_G2 & TX_FSLEW_G2	TDVL_LSG2

Table 7.1 SerDes Transmit Level Controls in the S[x]TXLCTL0 and S[x]TXLCTL1 Registers

As shown in Table 7.1, there are six parameters that may be programmed to adjust the transmitter drive levels. These are:

- Coarse Slew Rate Control (in the S[x]TXLCTL0 register).
- Transmitter Equalization Control (in the S[x]TXLCTL0 register).
- Fine Slew Rate Control (in the S[x]TXLCTL0 register).
- Coarse De-emphasis Control (in the S[x]TXLCTL0 register).
- Fine De-emphasis Control (in the S[x]TXLCTL1 register).
- Drive Level Control (in the S[x]TXLCTL1 register).

Modification of these settings take an immediate effect on the SerDes. Therefore, the link does not need to be retrained explicitly (i.e., by setting the link-retrain (LRET) bit in the PCIELCTL) in order for these settings to take effect. Still, the user must be careful when changing the transmit voltage margin while the port is in normal operating mode, as this may result in the link instability.

Table 7.2 shows a number of possible settings for the drive, de-emphasis, and slew rate controls in Gen1 mode[1]. These can be used as guidance when adjusting the SerDes transmit levels. The default setting is highlighted. Note that in Gen1 mode, de-emphasis is ideally -3.5dB with +/- 0.5dB error (refer to Section 4.3.3.5 of the PCI Express 2.0 Specification). All settings listed in the table ensure that the de-emphasis is kept within the allowable range.

1. Table values are based on simulations using the Snowbush SerDes HSPICE model and device package s-parameters. Values are sampled at the device pins. The simulation assumes typical conditions, with VddPEA = VddPETA = 1.0V, VddPEHA = 2.5V, and TX_AMPBOOST = 0x1. Please refer to the device data sheet for post-silicon device characterization data.

Notes

Transmit Levels			Settings of Relevant Fields in the S[x]TXLCTL0 & S[x]TXLCTL1 registers				
Drive Level	De-emphasis	De-emphasized Level	TDVL_FS3DBG1	TX_EQ_3DBG1	CDC_FS3DBG1	FDC_FS3DBG1	TX_SLEW_G1
959	-3.6	636	0x1C	0x2	0x3	0x3	0x2
959	-3.6	636	0x1B	0x2	0x3	0x4	0x2
958	-3.6	636	0x1A	0x2	0x3	0x4	0x2
957	-3.6	635	0x19	0x2	0x3	0x4	0x2
956	-3.6	635	0x18	0x2	0x3	0x4	0x2
949	-3.5	631	0x17	0x2	0x3	0x4	0x2
942	-3.5	628	0x16	0x2	0x3	0x4	0x2
935	-3.5	624	0x15	0x2	0x3	0x4	0x2
928	-3.5	620	0x14	0x2	0x3	0x4	0x2
902	-3.5	602	0x13	0x2	0x3	0x4	0x2
877	-3.5	584	0x12	0x2	0x3	0x4	0x2
851	-3.6	566	0x11	0x2	0x3	0x4	0x2
825	-3.6	547	0x10	0x2	0x3	0x4	0x2
789	-3.5	525	0x0F	0x2	0x3	0x3	0x2
753	-3.5	503	0x0E	0x2	0x3	0x3	0x2
716	-3.5	481	0x0D	0x2	0x3	0x3	0x2
680	-3.4	459	0x0C	0x2	0x3	0x3	0x2
638	-3.4	431	0x0B	0x2	0x3	0x2	0x2
596	-3.4	403	0x0A	0x2	0x3	0x2	0x2
554	-3.4	374	0x09	0x2	0x3	0x2	0x2
512	-3.4	346	0x08	0x2	0x3	0x2	0x2
465	-3.4	314	0x07	0x2	0x3	0x2	0x2
419	-3.5	281	0x06	0x2	0x3	0x2	0x2
372	-3.5	248	0x05	0x2	0x3	0x2	0x2
325	-3.6	216	0x04	0x2	0x3	0x2	0x2
274	-3.6	182	0x03	0x2	0x3	0x1	0x2
224	-3.6	148	0x02	0x2	0x3	0x1	0x2
173	-3.6	115	0x01	0x2	0x3	0x1	0x2
122	-3.6	81	0x00	0x2	0x3	0x1	0x2

Table 7.2 SerDes Transmit Driver Settings in Gen1 Mode

Table 7.3 shows a number of possible settings for the drive, de-emphasis, and slew rate controls in Gen2 mode with -3.5dB de-emphasis.[1] The default setting is highlighted.

Notes

Transmit Levels			Settings of Relevant Fields in the S[x]TXLCTL0 & S[x]TXLCTL1 registers				
Drive Level	De-emphasis	De-emphasized Level	TDVL_FS3DBG2	TX_EQ_3DBG2	CDC_FS3DBG2	FDC_FS3DBG2	TX_SLEW_G2
898	-3.5	603	0x1C	0x1	0x1	0x3	0x0
891	-3.5	596	0x1B	0x1	0x1	0x4	0x0
884	-3.5	589	0x1A	0x1	0x1	0x4	0x0
878	-3.6	581	0x19	0x1	0x1	0x4	0x0
871	-3.6	574	0x18	0x1	0x1	0x4	0x0
854	-3.6	562	0x17	0x1	0x1	0x4	0x0
837	-3.7	549	0x16	0x1	0x1	0x4	0x0
819	-3.7	537	0x15	0x1	0x1	0x4	0x0
802	-3.7	525	0x14	0x1	0x1	0x4	0x0
777	-3.7	509	0x13	0x1	0x1	0x3	0x0
752	-3.7	494	0x12	0x1	0x1	0x3	0x0
727	-3.6	478	0x11	0x1	0x1	0x3	0x0
702	-3.6	463	0x10	0x1	0x1	0x3	0x0
672	-3.6	446	0x0F	0x1	0x1	0x1	0x0
641	-3.5	428	0x0E	0x1	0x1	0x1	0x0
610	-3.4	411	0x0D	0x1	0x1	0x1	0x0
579	-3.4	393	0x0C	0x1	0x1	0x1	0x0
546	-3.4	370	0x0B	0x1	0x1	0x0	0x0
512	-3.4	346	0x0A	0x1	0x1	0x0	0x0
479	-3.4	323	0x09	0x1	0x1	0x0	0x0
445	-3.4	300	0x08	0x1	0x1	0x0	0x0
406	-3.4	274	0x07	0x1	0x0	0x7	0x0
367	-3.4	248	0x06	0x1	0x0	0x7	0x0
328	-3.4	222	0x05	0x1	0x0	0x7	0x0
289	-3.4	196	0x04	0x1	0x0	0x7	0x0
246	-3.5	166	0x03	0x1	0x0	0x5	0x0
203	-3.5	136	0x02	0x1	0x0	0x5	0x0
160	-3.6	107	0x01	0x1	0x0	0x5	0x0
118	-3.7	77	0x00	0x1	0x0	0x5	0x0

Table 7.3 SerDes Transmit Driver Settings in Gen2 Mode with -3.5dB de-emphasis

[1] Table values are based on simulations using the Snowbush SerDes HSPICE model and device package s-parameters. Values are sampled at the device pins. The simulation assumes typical conditions, with VddPEA = VddPETA = 1.0V, VddPEHA = 2.5V, and TX_AMPBOOST = 0x1. Please refer to the device data sheet for post-silicon device characterization data.

Notes

Table 7.4 shows a number of possible settings for the drive, de-emphasis, and slew rate controls in Gen2 mode with -6.0dB de-emphasis[1]. The default setting is highlighted. As mentioned above, the PCI Express 2.0 spec allows an error of up to +/- 0.5dB on the de-emphasis. All settings listed in the table ensure that the de-emphasis is kept within the allowable range.

Transmit Levels			Settings of Relevant Fields in the S[x]TXLCTL0 & S[x]TXLCTL1 registers				
Drive Level	De-emphasis	De-emphasized Level	TDVL_FS6DBG2	TX_EQ_6DBG2	CDC_FS6DBG2	FDC_FS6DBG2	TX_SLEW_G2
903	-5.8	461	0x1C	0x1	0x3	0x1	0x0
901	-5.8	460	0x1B	0x1	0x3	0x2	0x0
899	-5.8	459	0x1A	0x1	0x3	0x2	0x0
897	-5.8	458	0x19	0x1	0x3	0x2	0x0
895	-5.8	456	0x18	0x1	0x3	0x2	0x0
884	-5.9	447	0x17	0x1	0x3	0x3	0x0
874	-6.0	438	0x16	0x1	0x3	0x3	0x0
864	-6.1	429	0x15	0x1	0x3	0x3	0x0
853	-6.2	420	0x14	0x1	0x3	0x3	0x0
853	-6.2	420	0x13	0x1	0x3	0x2	0x0
853	-6.2	420	0x12	0x1	0x3	0x2	0x0
853	-6.2	420	0x11	0x1	0x3	0x2	0x0
853	-6.2	420	0x10	0x1	0x3	0x2	0x0
800	-6.2	394	0x0F	0x1	0x3	0x1	0x0
748	-6.2	368	0x0E	0x1	0x3	0x1	0x0
695	-6.2	342	0x0D	0x1	0x3	0x1	0x0
642	-6.2	316	0x0C	0x1	0x3	0x1	0x0
605	-6.1	298	0x0B	0x1	0x1	0x7	0x0
568	-6.1	280	0x0A	0x1	0x1	0x7	0x0
530	-6.1	262	0x09	0x1	0x1	0x7	0x0
493	-6.1	244	0x08	0x1	0x1	0x7	0x0
450	-6.1	224	0x07	0x1	0x1	0x5	0x0
407	-6.0	203	0x06	0x1	0x1	0x5	0x0
363	-5.9	183	0x05	0x1	0x1	0x5	0x0
320	-5.9	162	0x04	0x1	0x1	0x5	0x0

Table 7.4 SerDes Transmit Driver Settings in Gen2 Mode with -6.0dB de-emphasis (Part 1 of 2)

1. Table values are based on simulations using the Snowbush SerDes HSPICE model and device package s-parameters. Values are sampled at the device pins. The simulation assumes typical conditions, with VddPEA = VddPETA = 1.0V, VddPEHA = 2.5V, and TX_AMPBOOST = 0x1. Please refer to the device data sheet for post-silicon device characterization data.

Notes

Transmit Levels			Settings of Relevant Fields in the S[x]TXLCTL0 & S[x]TXLCTL1 registers				
Drive Level	De-emphasis	De-emphasized Level	TDVL_FS6DBG2	TX_EQ_6DBG2	CDC_FS6DBG2	FDC_FS6DBG2	TX_SLEW_G2
272	-6.0	138	0x03	0x1	0x1	0x3	0x0
225	-6.1	113	0x02	0x1	0x1	0x3	0x0
177	-6.2	88	0x01	0x1	0x1	0x3	0x0
130	-6.3	63	0x00	0x1	0x1	0x3	0x0

Table 7.4 SerDes Transmit Driver Settings in Gen2 Mode with -6.0dB de-emphasis (Part 2 of 2)

When the PHY operates in low-swing mode, de-emphasis is automatically turned off. Therefore, the fine and coarse de-emphasis controls in the S[x]TXLCTL0 and S[x]TXLCTL1 registers have no effect. In this mode, the TDVL_LSG1 and TDVL_LSG2 fields in the S[x]TXLCTL1 register control the transmitter voltage swing for Gen1 and Gen2 modes respectively. Please refer to section Low-Swing Transmitter Voltage Mode on page 7-13 for further details.

In addition to the SerDes settings described above, the user may apply an amplitude boost to the drive swing by setting the TX_AMPBOOST field in the S[x]TXLCTL0 register. Amplitude boost may be applied on a per-lane basis. Amplitude boost may be applied to increase the drive swings above the values shown in Tables 7.2, 7.3, and 7.4. Refer to the description of the TX_AMPBOOST field for further details.

Programmable De-emphasis Adjustment

The tables shown in the previous section list different settings to control the SerDes drive swing while keeping the de-emphasis within the allowable range, depending on the PHY operating mode (e.g., Gen1 data rate, Gen2 data rate and -3.5 dB de-emphasis, or Gen2 data rate with -6.0 dB de-emphasis).

It is possible to modify the de-emphasis in fine or coarse increments on a per-lane basis, using the appropriate fields in the S[x]TXLCTL0 and S[x]TXLCTL1 registers. Table 7.1 shows the register fields that control fine and coarse de-emphasis for each PHY operating mode.

When using the de-emphasis controls, it is important to understand that the actual deemphasis applied on the link is a function of the de-emphasis controls, the transmitter equalization control, the transmit drive swing controls, and the data rate of the SerDes.

The coarse de-emphasis controls should generally be set as shown in Tables 7.2, 7.3, and 7.4. Note that there are separate coarse de-emphasis and transmit equalization controls per PHY operating mode. The settings shown in the above tables ensure that the de-emphasis falls within the nominal range mandated by the PCI Express Specification. The coarse de-emphasis settings shown in the above tables ensure that the de-emphasis falls within the nominal range mandated by the PCI Express specification 2.0. As shown in the tables, the coarse de-emphasis setting is dependent on the transmit drive swing setting. Therefore, modifying the transmit drive swing must be done in conjunction with modifying the coarse de-emphasis setting.

The fine de-emphasis registers allow modification of the de-emphasis in fine steps. There is a fine de-emphasis control per PHY operating mode.

When the PHY operates in Gen1 data rate with -3.5 dB de-emphasis, the fine de-emphasis control (FDC_FS3DBG1 field in the S[x]TXLCTL1 register) has the effect shown in Figure 7.1. In the figure, TXLEV[4:0] refers to the TDVL_FS3DBG1 field in the S[x]TXLCTL1 register.

When the PHY operates in Gen2 data rate with -3.5 dB de-emphasis, the fine de-emphasis control (FDC_FS3DBG2 field in the S[x]TXLCTL1 register) has the effect shown in Figure 7.2. In the figure, TXLEV[4:0] refers to the TDVL_FS3DBG2 field in the S[x]TXLCTL1 register.

Notes

When the PHY operates in Gen2 data rate with -6.0 dB de-emphasis, the fine de-emphasis control (FDC_FS6DBG2 field in the S[x]TXLCTL1 register) has the effect shown in Figure 7.3. In the figure, TXLEV[4:0] refers to the TDVL_FS6DBG2 field in the S[x]TXLCTL1 register.

As shown in these figures, the de-emphasis applied on the line varies depending on the setting of the transmit drive level field. Thus, when modifying the transmit drive level of the SerDes, the fine de-emphasis control must be adjusted appropriately to ensure that the de-emphasis on the line falls within the range mandated by the PCI Express specification.

Note: It is possible to turn off the de-emphasis (i.e., 0 db de-emphasis) by setting the coarse de-emphasis setting to a value of 0x3 and the fine de-emphasis setting to a value of 0x7.

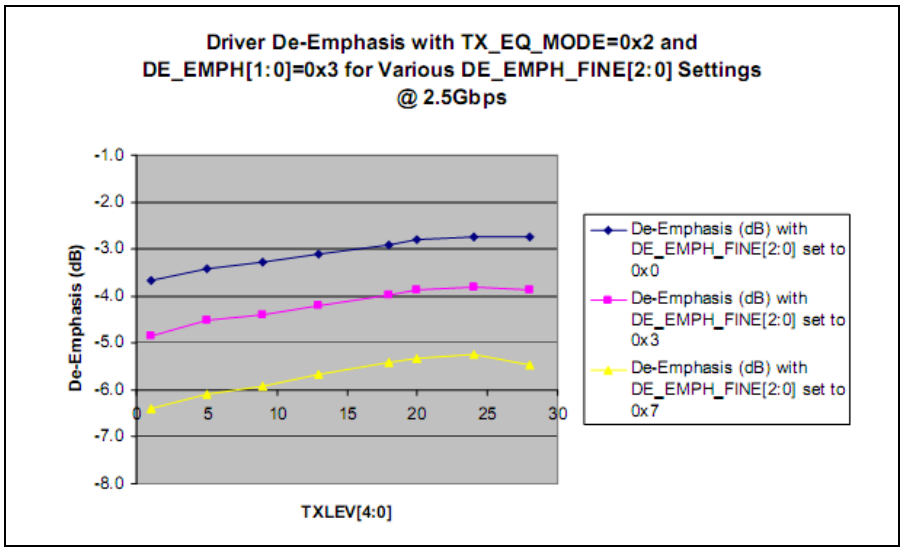


Figure 7.1 De-emphasis Applied on Link as a Function of the Fine de-emphasis and Transmit Drive Level Controls, when the PHY Operates in Gen1 Data Rate with -3.5 dB Nominal de-emphasis

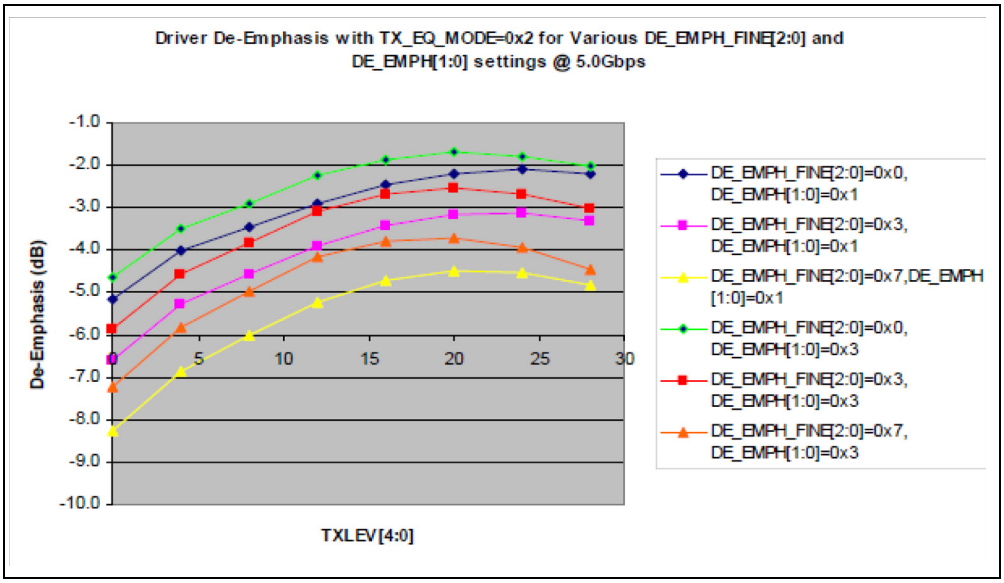


Figure 7.2 De-emphasis Applied on Link as a Function of the Fine de-emphasis and Transmit Drive Level Controls, when the PHY Operates in Gen2 Data Rate with -3.5 dB Nominal de-emphasis

Notes

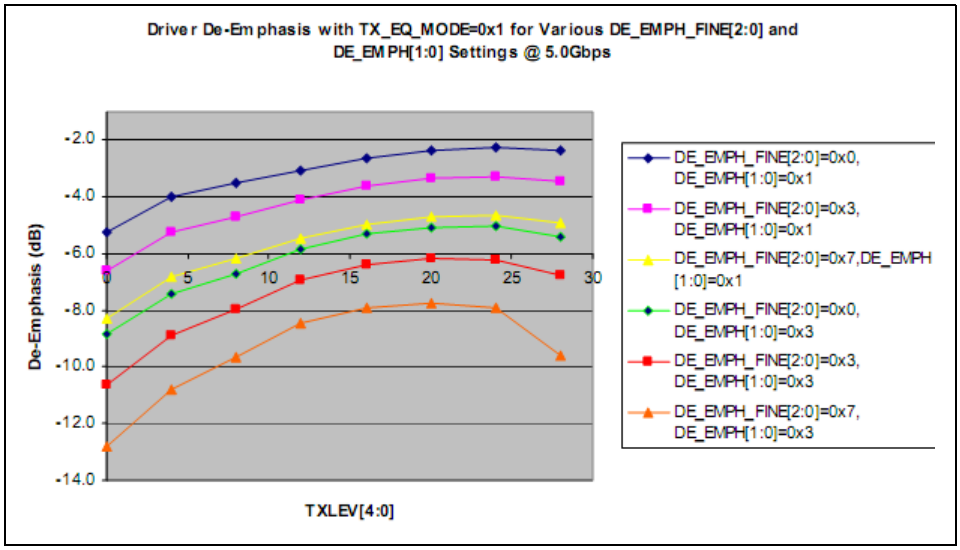


Figure 7.3 De-emphasis Applied on Link as a Function of the Fine de-emphasis and Transmit Drive Level Controls, when the PHY Operates in Gen1 Data Rate with -6.0 dB Nominal de-emphasis

Finally, note that it is possible to turn off de-emphasis (i.e., 0 dB de-emphasis) for a given PHY operating mode by setting the corresponding transmitter equalization control to 0x0, the coarse de-emphasis control to a value of 0x3, and the fine de-emphasis control to a value of 0x7.

Programmable Slew Adjustment

The transmitter's slew rate is controlled by fields in the S[x]TXLCTL0 register. It is possible to select different settings for Gen1 operation and Gen2 operation. The S[x]TXLCTL0 register contains the following slew rate controls:

At Gen1 data rate:

TX_SLEW_G1: transmit driver coarse slew control

TX_FSLEW_G1: transmit driver fine slew control

At Gen2 data rate:

TX_SLEW_G2: transmit driver coarse slew control

TX_FSLEW_G2: transmit driver fine slew control

Table 7.5 shows the slew rate settings. The values shown apply to both Gen1 and Gen2 data rates.

TX_SLEW	TX_FSLEW	Rise/Fall Times (20% to 80%) in Picoseconds
3	5	326
	4	266
	2	141
	0	89

Table 7.5 Transmitter Slew Rate Settings (Part 1 of 2)

Notes

2	5	337
	4	269
	2	120
	0	60
1	5	340
	4	272
	2	123
	0	50
0	5	342
	4	273
	2	127
	0	48

Table 7.5 Transmitter Slew Rate Settings (Part 2 of 2)

Transmit Margining using the PCI Express Link Control 2 Register

When the Transmit Margin (TM) field in the port's PCIELCTL2 register is set to a value other than 'Normal Operating Range', the transmitter voltage levels are controlled by hardware based on the setting of the TM field, and not by the S[x]TXLCTL0 and S[x]TXLCTL1 registers. Per the PCI Express 2.0 specification, transmit margining may be done in full-swing mode or in low-swing mode. Table 7.6 shows the transmit margining settings supported by the switch.[1]

Full Swing Mode (mV)	Low Swing Mode (mV)
900	500
700	400
500	300
300	200
200	100

Table 7.6 PCI Express Transmit Margining Levels supported by the PES48T12G2

Note that in compliance mode (i.e., when the associated port's PHY LTSSM is in the Polling.Compliance state), the SerDes transmit level is controlled by the TM field in the associated port's PCIELCTL2 register, and the de-emphasis setting is controlled by the LTSSM based on the rules described in Section 4.2.6.2.2 of the PCI Express 2.0 specification.

- When the LTSSM enters the Polling.Compliance state in full-swing mode, the values for full-swing margining are applied.
- When the LTSSM enters the Polling.Compliance state in low-swing mode, the values for low-swing margining are applied.

1. The TX_AMPBOOST field in the S[x]TXLCTL0 register does have an effect during transmit margining.

Notes

Finally, when the TM field is modified, the newly selected value is not applied until the PHY LTSSM transitions through the states in which it is allowed to modify the transmit margin setting on the line (e.g., Recovery.RcvrLock). Therefore, after modifying this field, it is recommended that the link be retrained by setting the LRET bit in the port's PCIELCTL register.

Low-Swing Transmitter Voltage Mode

The PES48T12G2 ports support the optional low-swing transmit voltage mode defined in the PCI Express 2.0 specification. In this mode, the transmitter's voltage level is set to approximately half the value of the full-swing (default) mode, which reduces power consumption in the SerDes. This mode is enabled by setting the Low-Swing Enable (LSE) bit in the SerDes Configuration register.

– For a merged port, the LSE bit must be set in the SerDes Configuration register of both of the SerDes associated with the port.

When Low-Swing mode is enabled, the transmitter drive level is reduced and de-emphasis is automatically turned off. Therefore, the Selectable De-emphasis (SDE) and Compliance De-emphasis (CDE) fields in the PCIELCTL2 register have no effect. Additionally, the Current De-emphasis (CDE) field in the PCIELSTS2 register becomes invalid.

The low-swing mode transmitter voltage swing may be adjusted via the TDVL_LSG1 (when operating in Gen1 mode) and TDVL_LSG2 (when operating in Gen2 mode) fields in the S[x]TXLCTL1 register. Table 7.7 shows the transmitter's drive swing for different values of TDVL_LSG1, when the port operates in low swing mode at Gen1 speed[1]. Table 7.8 shows the transmitter's drive swing for different values of TDVL_LSG2, when the port operates in low swing mode at Gen2 speed. The default setting is highlighted.

Drive Level	TDVL_LSG1
820	0x0F
785	0x0E
750	0x0D
714	0x0C
673	0x0B
632	0x0A
590	0x09
549	0x08
499	0x07
449	0x06
399	0x05
350	0x04
296	0x03
242	0x02
188	0x01
134	0x00

Table 7.7 SerDes Transmit Drive Swing in Low Swing Mode at Gen1 Speed

1. Table values are based on simulations using the Snowbush SerDes HSPICE model and device package s-parameters. The simulation assumes typical conditions, with VddPEA = VddPETA = 1.0V, VddPEHA = 2.5V, and TX_AMPBOOST = 0x2. Please refer to the device data sheet for post-silicon device characterization data.

Notes

Drive Level	TDVL_LSG2
764	0x0F
737	0x0E
709	0x0D
682	0x0C
649	0x0B
616	0x0A
583	0x09
550	0x08
506	0x07
462	0x06
418	0x05
374	0x04
320	0x03
266	0x02
212	0x01
158	0x00

Table 7.8 SerDes Transmit Drive Swing in Low Swing Mode at Gen2 Speed

When the PHY enters the Polling.Compliance state and low-swing mode is enabled, the following occurs:

- The transmit drive level is selected by the Transmit Margin (TM) field in the PCIELCTL2 register. This field has specific transmit margin levels for full-swing and low-swing mode. The values corresponding to low-swing mode are applied.
- De-emphasis is turned off.

Receiver Equalization Controls

The switch contains SerDes receiver equalization controls on a per-lane basis. The receiver equalization circuit has two controls which may be programmed via the SerDes Receiver Equalization Lane Control (S[x]RXEQLCTL) register. These are:

- Receiver Equalization Zero (RXEQZ): Increases the high-frequency gain of the equalizer.
- Receiver Equalization Boost (RXEQB): Reduces the low-frequency gain of the equalizer.

Together, RXEQZ and RXEQB provide wide programmability and fine grain control over the equalizer's boost. Refer to the definition of the S[x]RXEQLCTL register for further details on programming these controls.

Notes

SerDes Power Management

In order to maximize power savings in the SerDes, the PES48T122G2 adheres to the following guidelines. For SerDes quads that are used, their power state depends on the state of the port(s) associated with the SerDes, as described below.

1. When a port is disabled:
   - For a x4 or x8 port, the SerDes quad(s) associated with the port are placed in a deep low power state.
     - There is one SerDes quad associated with a x4 port.
     - There are two SerDes quads associated with a x8 port.
   - For a x1 or x2 port, the SerDes lanes associated with the port are placed in a deep low power state.
     - If all lanes of a SerDes quad are associated with disabled ports, the entire SerDes quad is placed in a deep low power state.
2. When a port is not disabled:
   - The SerDes quad(s) associated with the port are turned-on.
   - Unused lanes are powered down.
     - Lanes that form the initial link width (i.e., lanes on which the PHY LTSSM detected the presence of a link partner in the Detect state) are considered used. All other lanes associated with the port are unused.[1]
   - Used lanes are active and fully powered.
   - Dynamic link width downconfigure (i.e., change of link width while the link is up) is handled per the rules in the PCI Express 2.0 specification. In this case, inactive lanes place their transmitter in electrical idle and enable receiver termination[2].

It is possible to explicitly power-down a SerDes quad by setting the POWERDN bit in the corresponding SerDes Control (S[x]CTL) register. Refer to the definition of this field for further details. Powering-down a SerDes shared by multiple ports results in all such ports being affected.

1. Note that unused lanes may become used when the PHY LTSSM transitions to the Detect state and retrains the link.
2. In the PES48T12G2, these lanes are placed in the P1 power state.

Notes

Theory of Operation

Notes

Introduction

This chapter describes the PES48T12G2-specific architectural behavior of the PCI Express switch.

Transaction Routing

The switch supports routing of all transaction types defined in PCI Express Base Specification Revision 2.0. This includes routing of specification defined transactions as well as those that may be used in vendor defined messages and in future revisions of the PCI Express specification. Specifically, the PES48T12G2 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 8.1.

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

Table 8.1 Switch Routing Methods

1. The only Gathered and Routed to Root message supported is a PME_TO_Ack message received on a downstream port.

Interrupts

When an unmasked interrupt condition occurs, an MSI or interrupt message is generated by the corresponding port as described in Table 8.2. The removal of the interrupt condition occurs when unmasked status bit(s) causing the interrupt are masked or cleared. The PES48T12G2 assumes that all generated MSIs target the root and routes these transactions to the upstream port. Configuring the address contained in a downstream port's MSIADDR and MSIADDRU registers to an address that does not route to the upstream port and generating an MSI produces undefined results.

It follows that MSIs generated by the switch's ports can't fall within the multicast BAR aperture in the partition. When this occurs, the behavior is undefined.

Notes

Unmasked 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 8.2 Downstream Port Interrupts

Downstream Port Interrupts

Downstream ports support generation of legacy interrupts and MSIs. The following are sources of downstream port interrupts and MSIs.

- Downstream port's hot-plug controller
- Link bandwidth notification capability (i.e., assertion of the LBWSTS or LABWSTS bits in the PCIELSTS register when interrupt notification is enabled for these bits)

When a port is configured to generate INTx messages, only INTA is used.

Legacy Interrupt Emulation

PES48T12G2 supports legacy PCI INTx emulation. 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 PES48T12G2 maintains an aggregated INTx state for each of the four interrupt signals (i.e., A through D) at each port. An Assert_INTx message is sent to the root by the upstream port when the aggregated state of the corresponding interrupt in the upstream port 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 in the upstream port transitions from an asserted to a negated state.

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. This mapping is summarized in Table 8.3.

Notes

	Upstream Port Interrupt (Port 0)			
	INTA	**INTB**	**INTC**	**INTD**
Downstream Device[1] Interrupt	Device (N mod 4) = 0 INTA	Device (N mod 4) = 0 INTB	Device (N mod 4) = 0 INTC	Device (N mod 4) = 0 INTD
	Device (N mod 4) = 1 INTD	Device (N mod 4) = 1 INTA	Device (N mod 4) = 1 INTB	Device (N mod 4) = 1 INTC
	Device (N mod 4) = 2 INTC	Device (N mod 4) = 2 INTD	Device (N mod 4) = 2 INTA	Device (N mod 4) = 2 INTB
	Device (N mod 4) = 3 INTB	Device (N mod 4) = 3 INTC	Device (N mod 4) = 3 INTC	Device (N mod 4) = 3 INTA

Table 8.3 Downstream to Upstream Port Interrupt Routing Based on Device Number

1. Device X INTy corresponds to external downstream generated INTy interrupts and INTy interrupts generated by the port.

If a Downstream Port goes down (i.e., transition to DL_Down state) the INTx virtual wires associated with that port are negated, and the upstream port's aggregate state is updated accordingly. This may result in the upstream port generating a Deassert_Intx message.

Access Control Services

The PES48T12G2 ports support Access Control Services (ACS) as defined in the PCI Express 2.0 Specification. In the PES48T12G2, ACS functionality is only performed by ports operating in Downstream Switch Port mode. A port operating in Downstream Switch Port mode supports the following ACS operations:

- ACS Source Validation
- ACS Translation Blocking
- ACS Peer-to-Peer[1] Request Redirect
- ACS Completion Redirect
- ACS Upstream Forwarding
- ACS Peer-to-Peer Egress Control
- ACS Direct Translated Peer-to-Peer

ACS is programmed via the ACS Capability Structure in a function's configuration space. ACS checks are only applied to TLPs flowing in the upstream direction.

- For a downstream port, these are TLPs that are received from the port's link, regardless of the final destination of the TLP (e.g., regardless of whether the TLP is going to the upstream port or a peer downstream port).

When a ACS causes a TLP to be re-directed, the re-direction is implemented such that TLPs received by a port that are ACS re-directed follow the ordering rules (for more information, contact ssdhelp@idt.com). Specifically, non-relaxed-ordered completion TLPs that are re-directed towards the root-complex can't pass previously received posted TLPs re-directed in the same direction.

The following figures show examples of ACS checks applied to TLPs. Figure 8.1 shows an example of ACS source validation at a downstream port. In this case, the offending TLP is dropped and a completion with completer-abort status is generated.

1. For a port operating in Downstream Switch Port mode, 'peer-to-peer' implies traffic received by the downstream port via the PCIe link that is destined towards another downstream port.

Notes

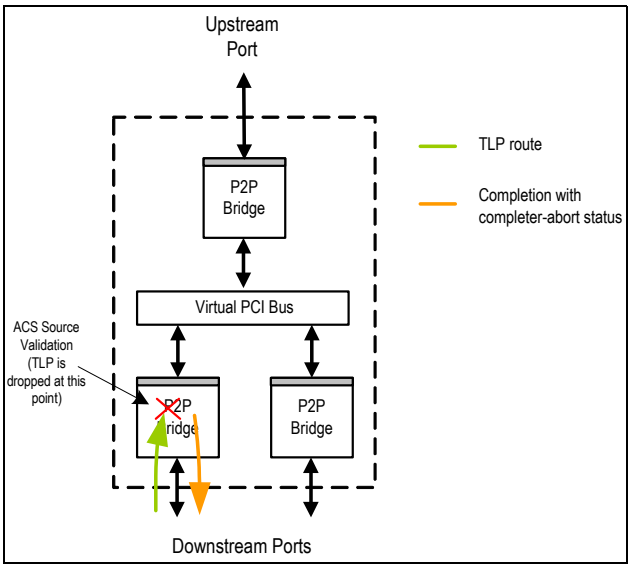


Figure 8.1 ACS Source Validation Example

Figure 8.2 shows an example of ACS peer-to-peer request re-direct at a downstream port. In this case, the offending TLP received by the downstream port is re-directed towards the root-complex.

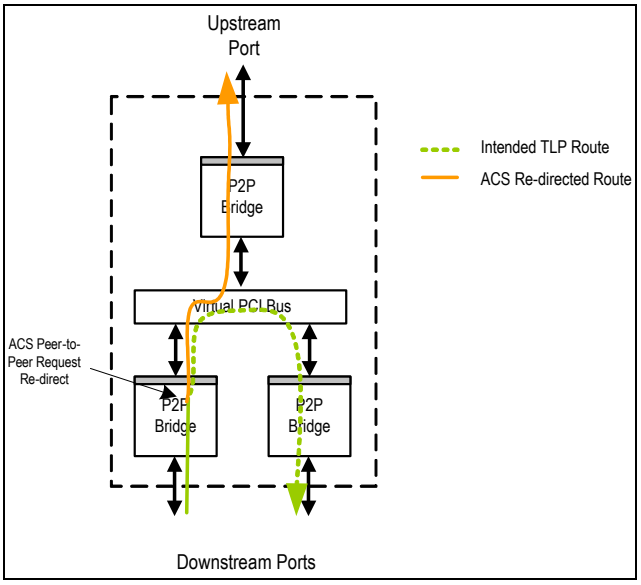


Figure 8.2 ACS Peer-to-Peer Request Re-direct at a Downstream Port

Figure 8.3 shows an example of ACS upstream forwarding at a downstream port. As with ACS Peer-to-Peer forwarding, the offending TLP received by the downstream port is re-directed towards the root-complex.

Notes

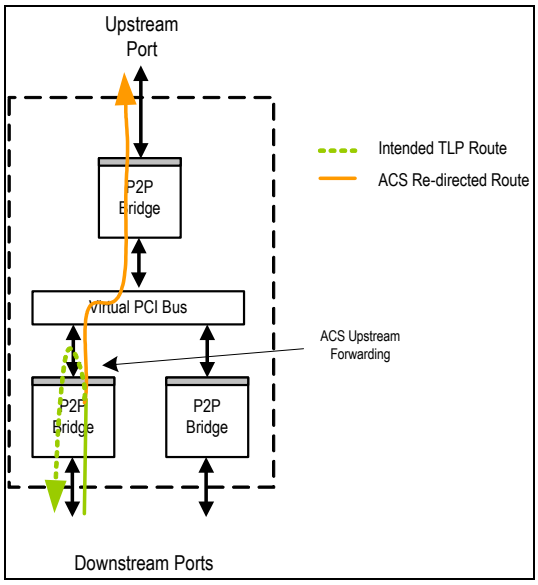


Figure 8.3 ACS Upstream Forwarding Example

When multiple ACS checks are enabled, they are prioritized as described below. Table 8.4 shows the prioritization for ACS checks associated with the reception of request TLPs. Table 8.5 shows the prioritization for ACS checks associated with the reception of completion TLPs.

ACS Check	Priority	Comment
ACS Source Validation	4 (Highest)	Applicable to request TLPs received by a downstream port on its ingress link.
ACS Translation Blocking	3	Applicable to memory request TLPs received by a downstream port on its ingress link.
ACS Upstream Forwarding	2	Applicable to request or completion TLPs received by the downstream port on its ingress link that target the port's egress link. This is not considered a peer-to-peer transfer.
ACS Peer-to-Peer Request Redirect	1 (Lowest)	Applicable to peer-to-peer request TLPs only Subject to the interaction rules in Section 6.12.3 of the PCI Express 2.0 Specification.
ACS Peer-to-Peer Egress Control		
ACS Direct Translated Peer-to-Peer		

Table 8.4 Prioritization of ACS Checks for Request TLPs

Notes

ACS Check	Priority	Comment
ACS Upstream Forwarding	2 (Highest)	Applicable to request or completion TLPs received by the downstream port on its ingress link that target the port's egress link. This is not considered a peer-to-peer transfer.
ACS Peer-to-Peer Completion Redirect	1 (Lowest)	Applicable to non-relaxed-ordered peer-to-peer completion TLPs only

Table 8.5 Prioritization of ACS Checks for Completion TLPs

ACS checks are only applicable to certain TLP types. Table 8.6 list the ACS checks supported by the PES48T12G2 and the TLP types on which they are applied.

ACS Check	Applicable to this TLP type(s)
ACS Source Validation	Request TLPs
ACS Translation Blocking	Memory Request TLPs
ACS P2P Request Re-direct	Peer-to-Peer Request TLPs
ACS P2P Completion Re-direct	Peer-to-Peer Completion TLPs
ACS Upstream Forwarding	Request or Completion TLPs that target the port's own egress link
ACS P2P Egress Control	Peer-to-Peer Request TLPs
ACS Direct Translated P2P	Peer-to-Peer Memory Request TLPs

Table 8.6 TLP Types Affected by ACS Checks

ACS violations associated with transparent operation are handled as described in section Error Detection and Handling on page 8-6.

Error Detection and Handling

This section describes error conditions associated with the switch. This includes physical, data-link, and transaction layer errors detected by the switch ports, as well as routing errors associated with the PCI-to-PCI bridge function in each switch port. Internal switch errors (i.e., parity errors, switch time-out, and internal memory errors) are not described here. Refer to section Internal Errors on page 3-12 for details on port operational modes.

Error detection and handling in the PES48T12G2 follows the requirements in the PCI Express 2.0 specification. The error checking and handling described here is performed by each PES48T12G2 port. In cases where the error condition propagates among ports (e.g., a poisoned TLP flowing from the upstream port to a downstream port), each port performs error checking and handling independently.

The errors described below are associated with specific actions to log and report the error. The terms 'uncorrectable error processing' and 'correctable error processing' refer to the processing described in Section 6.2.5 of the PCI Express 2.0 specification.

Some of the errors described below are marked as function-specific when the 'function claims the TLP'.

Notes

A PCI-to-PCI Bridge function claims a TLP in the following cases:

- Address Routed TLPs: If received on the primary side of the bridge, the TLPs address falls within the address space range(s) programmed in the base/limit registers. If received on the secondary side of the bridge, always.
- ID Routed TLPs: If received on the primary side of the bridge, the TLPs destination ID matches the bus aperture range programmed in the primary/secondary/subordinate registers or matches the bridge function's bus/device/function assignment. If received on the secondary side of the bridge, always.
- Implicit Route TLPs: Always.

Physical Layer Errors

Table 8.7 lists error checks performed by the physical layer and the action taken when an error is detected. Physical layer errors affect all functions of the port.

Error Condition	PCIe Base 2.0 Specification Section	Function-Specific Error	Action Taken
Link Errors (8b/10b, loss of symbol lock, elastic buffer overflow/underflow, lane-to-lane deskew)	4.2.4.6	No	Correctable error processing
Any TLP or DLLP framing rule violation.	4.2.2	No	Correctable error processing

Table 8.7 Physical Layer Errors

Data Link Layer Errors

Table 8.8 lists error checks performed by the data link layer and action taken when an error is detected. Data link layer errors affect all functions of the port. Per the PCI Express 2.0 specification, data link layer errors are ignored in cases where the error is associated with a received packet for which the physical layer reports an error. This prevents error pollution.

Error Condition	PCIe Base 2.0 Specification Section	Function-Specific Error	Action Taken
Bad TLP[1]	3.5.3.1	No	TLP discarded, Correctable error processing
Bad DLLP[2]	3.5.2.1	No	DLLP discarded, Correctable error processing
Replay time-out	3.5.2.1	No	Correctable error processing
REPLAY NUM rollover	3.5.2.1	No	Correctable error processing
DL Protocol Error[3]	3.5.2.1	No	DLLP discarded, Uncorrectable error processing
Surprise link down (refer to section Link Down on page 6-10).	3.5.2.1 & 3.2.1	No	Uncorrectable error processing

Table 8.8 Data Link Layer Errors

1. A Bad TLP is a TLP ending in EDB with LCRC that does not match inverted calculated LCRC, or a TLP with incorrect LCRC, or a TLP received with sequence number not equal to NEXT_RCV_SEQ and this is not a duplicate TLP)

2. A bad DLLP is a DLLP with a bad LCRC.

Notes

3. A DL protocol error occurs when an ACK or NAK DLLP is received and the sequence number specified by AckNak_Seq does not correspond to an unacknowledged TLP or to the value in ACKD_SEQ

Transaction Layer Errors

Table 8.9 lists non-ACS error checks associated with a PCI-to-PCI bridge function and the action taken when an error is detected. ACS error checks and handling are discussed in section ACS Error Handling on page 8-12. Per the PCI Express 2.0 specification, transaction layer errors are ignored in cases where the error is associated with a received packet for which the physical or data-link layers report an error. This prevents error pollution. Refer to section Transaction Layer Error Pollution on page 8-13 for details on transaction layer error pollution.

Notes

Error Condition	PCI Express Specification[1] Section	Function-Specific Error	Role Based (Advisory) Error Reporting Condition	Action Taken
Poisoned TLP received	2.7.2.2	Yes	Advisory when the corresponding error is configured as non-fatal in the AERUESV register and the function receives the TLP from the link or when function is the ultimate receiver of the TLP	Detected Parity Error (DPE) bit in the PCISTS or SECSTS register set appropriately. Non-advisory cases: uncorrectable error processing. Advisory cases: correctable error processing. TLP header logged in AER for advisory and non-advisory cases. Else, TLP header is not logged. Affected packet is forwarded across the bridge function (unless the bridge function is the ultimate receiver of the TLP, in which case the TLP is dropped by this function).
ECRC check failure	2.7.1	No	Advisory when the corresponding error is configured as non-fatal in the AERUESV register and the function is an intermediate receiver and receives the packet from the link [2]	Non-advisory cases: uncorrectable error processing. Advisory cases: correctable error processing. TLP header logged in AER. Affected packet is forwarded across the bridge (unless the bridge function is the ultimate receiver of the TLP, in which case the TLP is dropped by this function).
Unsupported request	See Table 8.10 below	Yes if a function claims the TLP. Else No.	Advisory when the corresponding error is configured as non-fatal in the AERUESV register and the request is non-posted	Non-advisory cases: uncorrectable error processing. Advisory cases: correctable error processing. TLP header logged in AER. For Non-Posted unsupported requests, the function that claims the TLP generates a completion with UR status. If the request is not claimed, then function 0 generates the completion with UR status.
Completion timeout	2.8	N/A	N/A (always non-advisory)	Not applicable (the bridge never issues requests).
Completer abort	2.3.1	N/A	N/A (always non-advisory)	Not applicable. Switch ports never issue completions with 'Completer Abort' status except for ACS violations. For the latter, the error is considered an ACS error and is not logged as a completer abort error.

Table 8.9 Transaction Layer Errors associated with the PCI-to-PCI Bridge Function (Part 1 of 2)

Notes

Error Condition	PCI Express Specification[1] Section	Function-Specific Error	Role Based (Advisory) Error Reporting Condition	Action Taken
Unexpected completion received	2.3.2	Yes if a function claims the TLP. Else No.	Advisory when the corresponding error is configured as non-fatal in the AERUESV register	Non-advisory cases: uncorrectable error processing. Advisory cases: correctable error processing. TLP header logged in AER. The unexpected completion is dropped.
Receiver overflow	2.6.1.2	No	N/A (always non-advisory)	Uncorrectable error processing TLP header **is not** logged in AER. TLP is nullified.
Flow control protocol error	2.6.1	No	N/A (always non-advisory)	Not applicable. The PES48T12G2 does not check for any flow control errors.
Malformed TLP	See Tables 8.11 and 8.12 below	No	N/A (always non-advisory)	Uncorrectable error processing TLP header logged in AER. TLP is nullified.
Multicast Blocked TLP	Multicast ECN	Yes	N/A (always non-advisory)	The Signaled Target Abort (STAS) bit is set in the PCISTS or SECSTS register if the TLP was received on the function's primary or secondary side respectively. Uncorrectable error processing TLP header logged in AER. TLP is nullified.
Internal Error	Internal Error Reporting ECN[3]	Yes	N/A (always non-advisory)	Refer to section Internal Errors on page 3-13.

Table 8.9 Transaction Layer Errors associated with the PCI-to-PCI Bridge Function (Part 2 of 2)

1. Refer to PCI Express Base 2.0 specification.

2. If the function is an intermediate receiver but does not receive the packet from the link, the error is not logged in AER.

3. Internal Error Reporting Engineering Change Notice to PCI Express 2.0 Base specification, April 24, 2008, PCI-SIG.

Note that ECRC checking is enabled for the port when at least one of the functions in the port has ECRC checking enabled. When ECRC checking is enabled for the port, the port checks the ECRC of all TLPs in which the TLP Digest (TD) bit is set in the TLP header. ECRC error logging and signaling is only applicable to port functions which have ECRC checking enabled.

Table 8.10 lists the conditions for which the PCI-to-PCI Bridge function in the PES48T12G2 ports handles requests as unsupported requests (UR).

Notes

Conditions handled as UR	Description	PCIe Specification Section
Routing Errors	Refer to section Routing Errors on page 8-16.	Numerous
Vendor Defined Type 0 message reception[1]	Vendor Defined Type 0 message which targets the PCI-to-PCI bridge function.	2.2.8.6
Messages with invalid message code	Reception of a message TLP with invalid message code that targets the switch port's PCI-to-PCI bridge function.	2.3.1
Poisoned IO request, memory write request, or message with data targeting the bridge function	Reception of a poisoned IO request, memory write request, or message with data (except Vendor Defined messages) that targets a switch port's PCI-to-PCI bridge function.	2.7.2.2
Function in D3Hot state	Refer to section Introduction on page 10-1.	5.3.1.4.1
Downstream Switch Port Link Down	TLPs flowing downstream across a downstream port's PCI-to-PCI Bridge whose link is down. Such TLPs are URed by the appropriate downstream port.	2.9.1

Table 8.10 Conditions handled as Unsupported Requests (UR) by the PCI-to-PCI Bridge Function

1. NOTE: Vendor Defined Type 1 messages which target the PCI-to-PCI bridge function are silently discarded.

Table 8.11 lists the formation error checks performed by the PCI-to-PCI bridge function on reception of TLPs. These checks are performed whenever the bridge function receives the packet from the link and are implemented on the ingress datapath of the stack's application layer.

TLP Type	Error Check
All	TLP must have a valid FMT/TYPE combination Data payload length <= Max_Payload_Size (i.e., MPS field in PCIEDCTL register)
All TLPs with data (i.e., FMT[1]=1)	LENGTH field must match actual payload data
All TLPs with ECRC (i.e., TD=1)	Actual TLP length must match calculated length (HEADER + PAYLOAD + ECRC)
I/O read or write request	LENGTH = 1 (doubleword) TC = 0 ATTR = 0 Last DWord BE[3:0] = 0b0000
Configuration read or write request	LENGTH = 1 (doubleword) TC = 0 ATTR = 0 Last DWord BE[3:0] = 0b0000

Table 8.11 Ingress TLP Formation Checks Associated with the PCI-to-PCI Bridge Function (Part 1 of 2)

Notes

TLP Type	Error Check
Message Requests interrupt message Power management message Error signalling message Unlock message Set power limit message	TC = 0
TLPs with Route to Root Complex routing.	May only be received on downstream ports
TLPs with Broadcast from Root Complex routing.	May only be received on upstream ports
TLPs with Gathered and Routed to Root Complex routing	May only be received by the downstream ports Must be a PME_TO_ACK message (all other TLP types with this routing are illegal)
Interrupt messages (INTx)	May only be received by the downstream ports
All	TLP traffic class (TC) must be mapped to VC0. TC to VC mapping is controlled by the TC/VC Map (TCVCMAP) field in the ingress port's VC Resource 0 Control (VCR0CTL) register.

Table 8.11 Ingress TLP Formation Checks Associated with the PCI-to-PCI Bridge Function (Part 2 of 2)

Table 8.12 lists the TLP formation error checks performed whenever the bridge function transmits a packet. These checks are implemented on the egress datapath of the stack's transaction layer.

TLP Type	Error Check
All	TLP traffic class (TC) must be mapped to VC0. TC to VC mapping is controlled by the TC/VC Map (TCVCMAP) field in the egress port's VC Resource 0 Control (VCR0CTL) register of the PCI-to-PCI bridge function.

Table 8.12 Egress Malformed TLP Error Checks

ACS Error Handling

As described in section Access Control Services on page 8-3, ACS checks are performed ports that operate in downstream switch port mode. All ACS checks are function-specific (i.e., are logged and handled by the function that detected the error). ACS checks may be divided into two groups: ACS checks that redirect the routing of TLPs and ACS checks that block the routing of TLPs.

- ACS re-direction of a TLP is not considered an error case and is therefore not logged in the function's AER capability structure.
- ACS blocking of a TLP is considered an error case and is logged in the function's AER capability structure. Such an error case is referred to as "ACS violation".

Table 8.13 lists ACS violation checks performed by the PCI-to-PCI bridge function of a port that operates in Downstream Switch Port mode. Note that the PES48T12G2 downstream ports do not support ACS Source Validation on message requests received by a port with 'Local - Terminate at Receiver' and 'Gathered and Routed to Root Complex' routing type (e.g., INTx, PME_TO_Ack, Vendor Defined messages).

Notes

ACS Check	PCI Express Specification[1] Section	Role Based (Advisory) Error Reporting Condition	Action Taken
ACS Source Validation	6.12.1.1	Advisory when the corresponding error is configured as non-fatal in the AERUESV register and an ACS violation is detected on a non-posted request	If TLP is a non-posted request, a completion with 'completer abort' status is generated. Note that this is not considered a completer abort error in AER. The Signaled Target Abort (STAS) bit is set in the SECSTS register. Non-advisory case: uncorrectable error processing. Advisory case: correctable error processing. TLP header logged in AER. The offending TLP is dropped.
ACS Translation Blocking		Advisory when the corresponding error is configured as non-fatal in the AERUESV register and an ACS violation is detected on a non-posted request	If TLP is a non-posted request, a completion with 'completer abort' status is generated and Signaled Target Abort (STAS) bit is set in the SECSTS register. Note that this is not considered a completer abort error in AER. Non-advisory case: uncorrectable error processing. Advisory cases: correctable error processing. TLP header logged in AER. The offending TLP is dropped.
ACS Peer-to-Peer Egress Control		Advisory when the corresponding error is configured as non-fatal in the AERUESV register and an ACS violation is detected on a non-posted request	If TLP is a non-posted request, a completion with 'completer abort' status is generated and Signaled Target Abort (STAS) bit is set in the SECSTS register. Note that this is not considered a completer abort error in AER. Non-advisory case: uncorrectable error processing. Advisory cases: correctable error processing. TLP header logged in AER. The offending TLP is dropped.
ACS Direct Translated Peer-to-Peer		(Refer to the next column)	Offending TLP is subject to ACS Peer-to-Peer Egress Control and ACS Peer-to-Peer Request Redirect rules.

Table 8.13 ACS Violations for Ports Operating in Downstream Switch Port Mode

1. Refer to PCI Express Base 2.0 specification.

Transaction Layer Error Pollution

Per section 6.2.3.2.3 of the PCI Express Base Specification Revision 2.0, transaction layer errors may be prioritized to prevent error pollution in AER. Error pollution rules only apply to errors associated with the reception of a TLP. Errors not associated with the reception of a TLP are logged for each occurrence of the error.

The PES48T12G2 does not apply error pollution rules to internal errors detected by the device, even when such errors are associated with the reception of a TLP. As a result, it is possible that more than one AER error be logged on reception of a TLP which causes an internal error. For example, reception of a poisoned TLP which causes an internal double-bit ECC error in a port's ingress buffer memory would result in the poisoned and internal errors logged in the ingress port's AER capability structure.

Notes

In addition, the Detected Parity Error bit (DPE) in the PCISTS and SECSTS registers is not subject to error pollution rules and is therefore set when the PCI-to-PCI bridge receives a poisoned TLP on its primary or secondary side respectively, even if error pollution rules indicate that the poisoned TLP received error is superseded by a higher priority error.

Table 8.14 shows the prioritization of transaction layer errors used by PES48T12G2 ports. All the errors listed in the table are associated with the reception of a TLP. Errors not applicable to the PES48T12G2 (e.g., completion timeout and completer-abort) are not shown. Higher priority errors have precedence over lower priority errors. Errors with the same priority are mutually exclusive (the errors can't occur simultaneously).

Error	Associated with Packet Reception	Priority
Receiver Overflow	Yes	7 (highest)
ECRC Check failure	Yes	6
Malformed TLP received	Yes	5
ACS Violation	Yes	4
Multicast Blocked TLP	Yes	3
Unsupported Request	Yes	2
Unexpected Completion received	Yes	
Poisoned TLP received	Yes	1 (lowest)

Table 8.14 Prioritization of Transaction Layer Errors

The prioritization of errors shown in Table 8.14 determines the error that is logged and reported when multiple errors are detected simultaneously for the received TLP. Higher priority errors inhibit the logging and reporting of lower priority errors in AER. Still, higher priority errors do not inhibit the checking and TLP handling action of lower priority errors, unless the higher priority error results in the TLP being consumed, dropped, or nullified by the detecting function (refer to Table 8.9 and Table 8.13).

Refer to section Routing Errors on page 8-16 which contains a decision diagram for error checking and logging on a received TLP taking into account the error pollution rules and priorities.

Notes

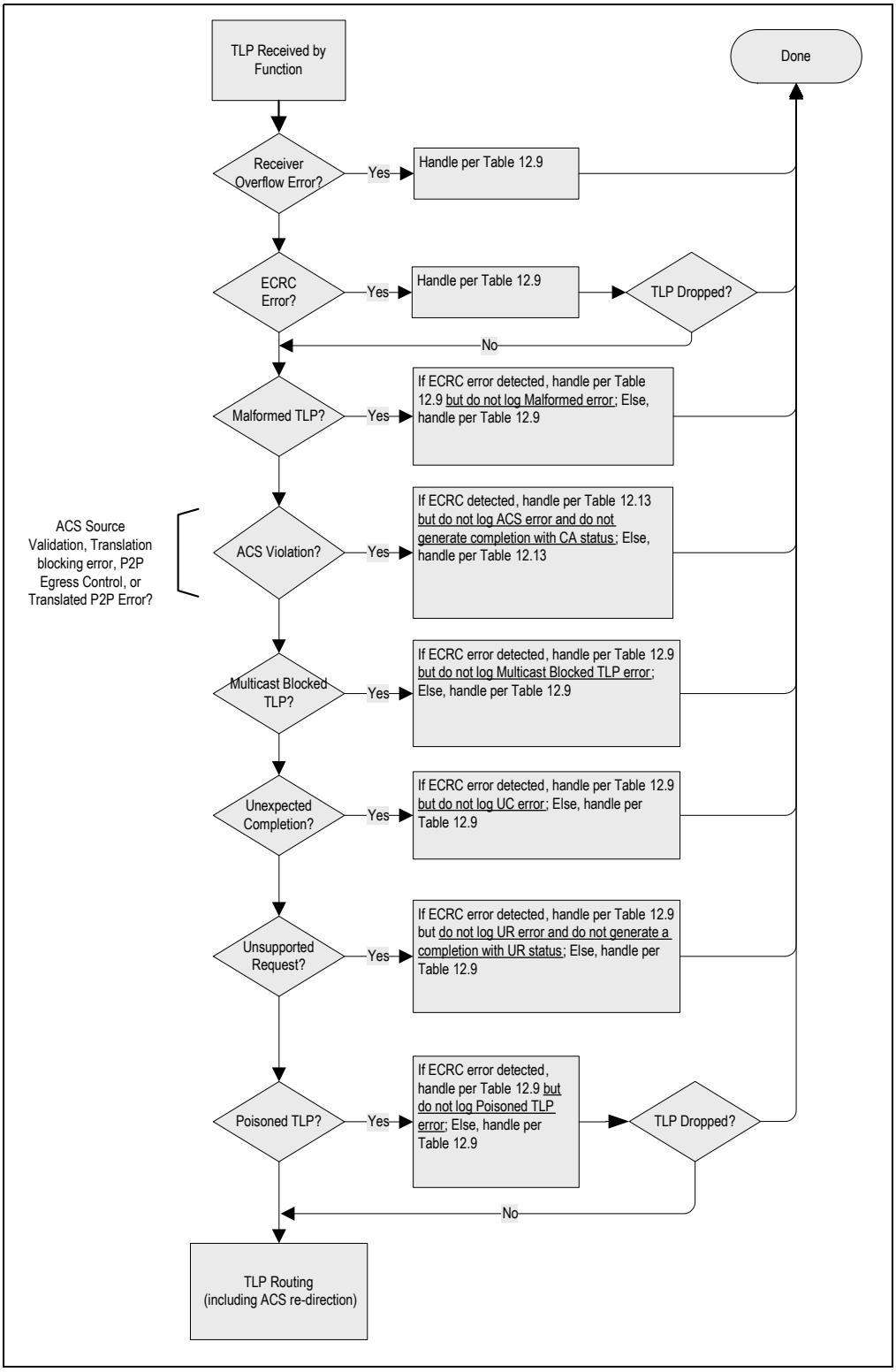


Figure 8.4 Error Checking and Logging on a Received TLP

Notes

Note the following:

- Except for ECRC and Poisoned TLP errors, all other errors detected on the received TLP cause the detecting function to consume, drop, or nullify the TLP.
- Receiver overflow errors are only checked and logged.
- ECRC errors are only checked and logged when the TLP has passed receiver overflow checks. Per the error handling rules in Table 8.9, a TLP with ECRC error may result in the TLP being forwarded across the PCI-to-PCI bridge function. Such TLPs are subject to further error checking by the receiving function.
- TLP malformation errors are only logged and reported when the TLP has passed ECRC error checking. Still, a TLP with ECRC error that is not dropped as a result of the ECRC error is subject to TLP malformation error checking since the higher priority ECRC error does not inhibit the checking of the lower-priority malformation error.
  - In case the TLP with ECRC error is malformed, the TLP is nullified (per the error handling rules in Table 8.9) but the malformed TLP error is not logged in AER.
- ACS errors violations are only logged and reported when the TLP has passed malformation and ECRC checks. Still, a TLP with ECRC error that is not dropped as a result of the higher priority errors is subject to ACS blocking or re-direction since the higher priority ECRC error does not inhibit the checking of the lower-priority ACS checks.
  - In case the TLP with ECRC error is blocked by an ACS check (e.g., ACS Source Validation), the blocking action takes place but the ACS error is not logged and a completion with 'completer-abort' status is not generated.
- Multicast blocking errors are only logged and reported when the TLP has passed ECRC, malformation, and ACS violation checks. Still, a TLP with ECRC error that is not dropped as a result of the higher priority errors is subject to multicast blocking checks since the higher priority ECRC error does not inhibit the checking for multicast blocking errors.
  - In case the TLP with ECRC error is blocked by the multicast blocking check, the blocking action takes place but a multicast blocking error is not logged.
- Unsupported request errors are only logged and reported when the request TLP has passed ECRC, malformation, ACS violation, and multicast blocking checks. Still, an unsupported request TLP that is not dropped as a result of the higher priority errors is subject to unsupported request handling.
  - In case the TLP with ECRC error is an unsupported request, the TLP is handled per the rules in Table 8.9 but the unsupported is not logged in AER and a completion TLP is not generated.
- Finally, poisoned TLP errors are only logged when the TLP has passed ECRC, malformation, ACS violation, multicast blocking, unsupported request, and unexpected completion checks (i.e., the TLP is a valid request or completion claimed by a port function).
  - Per the error handling rules in Table 8.9, poisoned TLP may be forwarded across the PCI-to-PCI bridge function.

For example, when a downstream port receives a posted memory request TLP from the link with an ECRC error[1], the port's PCI-to-PCI bridge function will handle the TLP as described in Table 8.9 and Table 8.13. Because ECRC error has higher priority than other errors, only the ECRC error is logged in the port's AER Capability Structure. If the TLP targets the receiving port (i.e., the port is the 'ultimate receiver' of the TLP), the TLP is dropped and no further checking is required. If the port that received the TLP is an intermediate receiver, the TLP is not dropped due to the ECRC error and thus the port performs lower priority error checks and takes the appropriate action. In this example, if the TLP results in an unsupported request error (e.g., the BME bit in the function's PCICMD register is cleared), the port consumes the TLP, does not log the UR error, and does not generate a completion TLP as a result of the UR error.

Routing Errors

This section lists TLP routing errors that are detected by the PCI-to-PCI bridge function in the PES48T12G2 ports. Except for completions (section Completions (Routed by ID) on page 8-17), all of these errors are treated as unsupported requests.

1. Assuming that the reception of the TLP did not cause a receiver overflow error on the port.

Notes

Address Routed TLPs

TLPs received by an upstream port that match the upstream port's address range but which do not match a downstream port's address range (i.e., TLPs that do not route through the switch).

TLPs received by a downstream port that do not match the address range of any other downstream port, but match the address range of the upstream port.

TLPs received by a downstream port whose address decoding indicates they are to route back to the port on which they were received, if ACS Upstream Forwarding is disabled on the port. When ACS Upstream Forwarding is enabled, such TLPs are not considered errors and are forwarded upstream.

TLPs received by the primary side of a port that is not enabled for such transactions.

- For prefetchable memory and non-prefetchable memory transactions the Memory Access Enable (MAE) bit must be set in the port's PCI Command (PCICMD) register.
- For I/O transactions the I/O Access Enable (IOAE) bit must be set in the port's PCI Command (PCICMD) register.

MEM or IO TLPs received on a downstream port and the port's Bus Master Enable (BME) bit in the PCICMD register is cleared. MEM or IO TLPs from downstream ports that target the upstream port and the Bus Master Enable (BME) bit is cleared in the upstream port's PCICMD register. A VGA route from a VGA enabled downstream port.

Configuration Requests (Routed by ID)

Type 0 requests that arrive on a downstream port. Type 1 requests that arrive on a downstream port.

Type 1 requests that do not route through the upstream port's PCI-to-PCI bridge. Type 1 requests that are converted to Type 0 requests at the upstream port but which do not target an enabled downstream port device number (i.e., target a PCI-to-PCI bridge device number that doesn't exist.

Type 1 requests that route through the PES48T12G2, target a downstream port's link partner (i.e., are converted to a Type 0 request at the downstream port), and which do not target device zero. Note that this check is disabled when the Alternative Routing ID (ARI) function is enabled via the ARIFEN bit in the PCIEDCTL2 register.

Completions (Routed by ID)

Completions that attempt to route back onto the link on which they were received are silently dropped, if ACS Upstream Forwarding is disabled. When ACS Upstream Forwarding is enabled, such completion TLPs are not dropped and are forwarded upstream. Completions that do not have a valid route through the PES48T12G2 are silently dropped. All completions that terminate within the PES48T12G2 (i.e., ones that target the upstream switch port bus number or any device/function on the virtual PCI bus within the switch) are treated as unexpected completions.

ID Routed Messages

Messages that attempt to route back onto the link on which they were received, if ACS Upstream Forwarding is disabled. When ACS Upstream Forwarding is enabled, such TLPs are not considered errors and are forwarded upstream. Messages that do not have a valid route through the PES48T12G2. Messages that target a downstream port device number that does not exist.

A non-Vendor Defined Type 1 message which targets an enabled PES48T12G2 port. Vendor Defined Type 1 messages received by a PES48T12G2 port are silently discarded.

A non-Vendor Defined Type 1 message which is received by the upstream port.

Bus Locking

The PES48T12G2 supports locked transactions, allowing legacy software to run without modification on PCIe. Only one locked transaction sequence may be in progress at a time.

Notes

A locked transaction sequence is requested by the root complex 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) sent by the root complex.

When the PES48T12G2 receives a MRdLk transaction on the upstream switch port, it forwards the MRdLk transaction to the appropriate downstream port and locks the downstream port so that all subsequent TLPs destined to the locked port from other ports (except the upstream port) are blocked until the lock is released. Bus locking only affects TLPs that map to VC0 at the egress port. TLPs that do not map to VC0 are not affected by the lock.

The MRdLK transaction obeys PCIe ordering rules meaning that all queued posted requests for the downstream port are transmitted prior to the MRdLK being transmitted. The MRdLK is allowed by bypass queued completions.

Locking of a downstream port does not affect transactions destined to any other port (e.g., transactions from the other downstream ports to the upstream port and peer-to-peer transactions among other downstream ports are not blocked).

When a CplDLk is received by the locked downstream port, it forwards the CplDLk transaction to the upstream port and locks the upstream port so that all subsequent TLPs destined to the locked port from other ports (except the locked downstream port) are blocked until the lock is released. Bus locking only affects TLPs that map to VC0 at the egress port. TLPs that do not map to VC0 are not affected by the lock.

The CplDLk transaction obeys PCI ordering rules 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 to bypass queued non-posted requests. When a CplDLk is returned by the locked downstream port and the upstream port becomes locked, the switch becomes 'bus-locked'. While a switch is bus-locked, the following applies:

It is illegal to read or write any of the PCIe configuration space headers in ports associated with the switch since the switch can not generate a completion until it is unlocked. The behavior of the switch is undefined when a switch's PCIe configuration space register is read while the switch is bus-locked. Any register in the ports of the switch may be read or written via the SMBus.

It is allowed for the root to perform subsequent reads from the locked device (e.g., a legacy endpoint) 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 bus-locked switch. Therefore, a CplLk completion received by the downstream port of a bus-locked switch in no way "unlocks" the switch.

It is allowed 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 bus-locked switch.

The locked upstream and downstream ports may generate messages (i.e., "insert messages"). These messages include interrupt emulation messages and error messages. The locked ports may also generate MSIs.

The behavior of a bus-locked switch is undefined when:

- Any transaction other than a MWr, MRdLk, and Unlock message is received on the upstream port.
- Any transaction other than a CplLk and a CplDLk is received on the locked downstream port.
- A MRdLk TLP is received on the switch's upstream port destined to an unlocked downstream port.
- A TLP is received by the upstream port destined to an unlocked downstream port.

When an Unlock message is received on the switch's 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 previously locked ports.

The unlock message obeys PCI ordering rules meaning that all queued posted requests from the upstream port are completed prior to the switch becoming unlocked. Unlocked ports ignore the reception of the unlock message.

Notes

Note that when a TLP received by port is blocked from being forwarded due to a bus-locked switch, 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.

Notes

Hot-Plug and Hot-Swap

Notes

Introduction

As illustrated in Figures 9.1 through 9.3, a PCIe switch may be used in one of three hot-plug configurations. Figure 9.1 illustrates the use of PES48T12G2 in an application in which two downstream ports are connected to slots into which add-in cards may be hot-plugged.

Figure 9.2 illustrates the use of PES48T12G2 in an add-in card application. Here the 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 9.3 illustrates the use of PES48T12G2 in a carrier card application. In this application, the 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 Figures 9.1 and 9.2, it will not be discussed further.

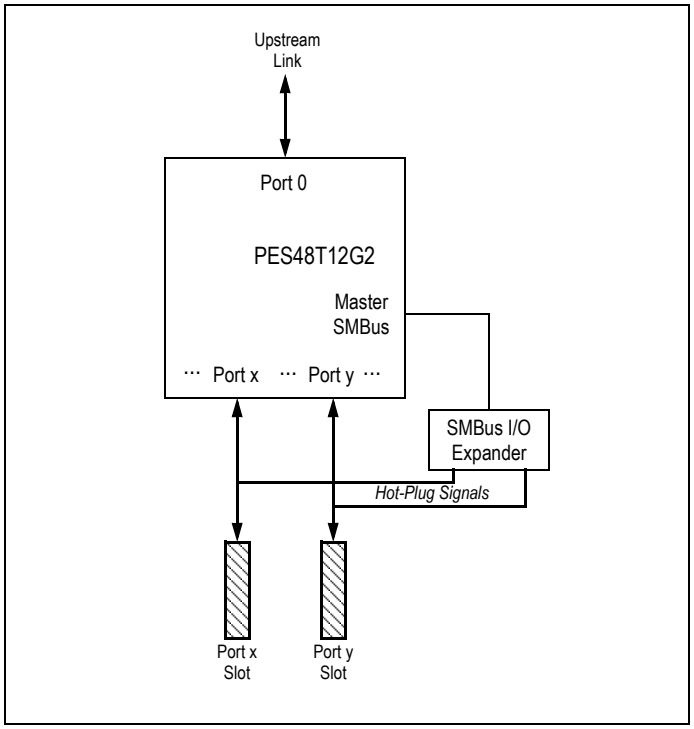


Figure 9.1 Hot-Plug on Switch Downstream Slots Application

Notes

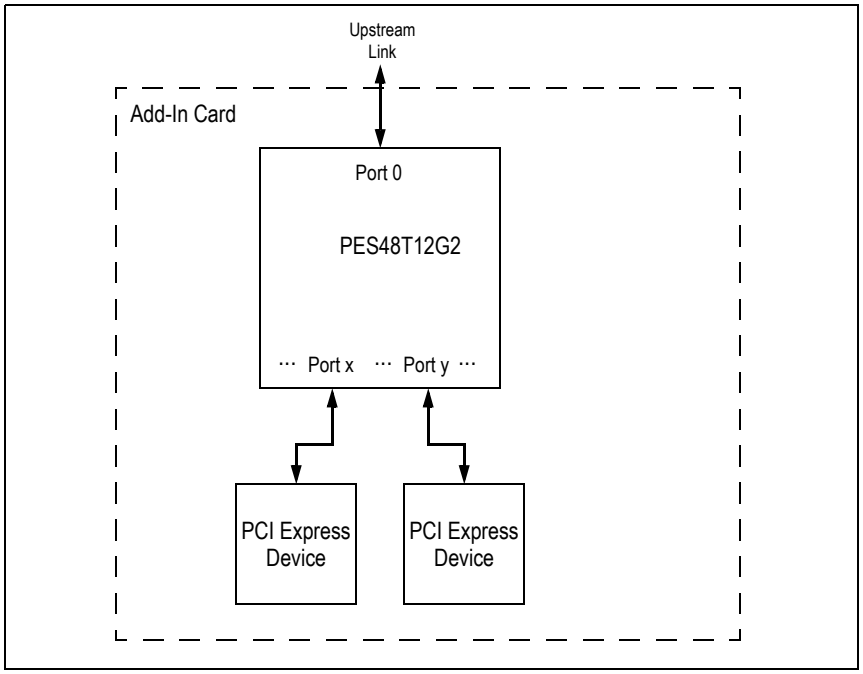


Figure 9.2 Hot-Plug with Switch on Add-In Card Application

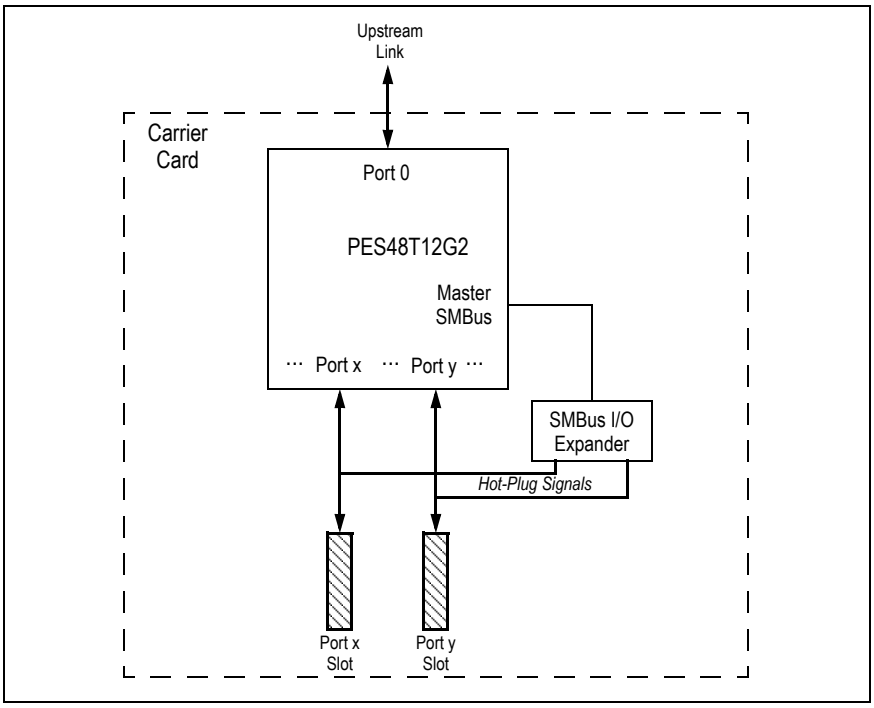


Figure 9.3 Hot-Plug with Carrier Card Application

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 and is not supported in PES48T12G2.

Notes

Associated with all PES48T12G2 ports is a hot-plug controller. However, hot-plug is only supported when a port is configured to operate in downstream switch port mode. In a port configured to operate in downstream switch port mode, the hot-plug controller may be enabled by setting the HPC bit in the PCI Express Slot Capabilities (PCIESCAP) register associated with that port.

– The HPC bit may be set at any time, but is typically set during a switch fundamental reset via serial EEPROM.

PES48T12G2 allows hot-plug sensor inputs and indicator outputs to be located next to the slot or on the plug in module. Regardless of the physical location, the indicators are controlled by PES48T12G2's downstream port.

Hot-Plug Signals

All PCIe defined hot-plug signals are supported on each port using low cost external I/O expanders. Table 9.1 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.

Signal	Type	Name/Description
PxAIN	O	**Port x[1] Attention Indicator Output.**
PxAPN	I	**Port x Attention Push button Input.**
PxILOCKP	O	**Port x Electromechanical Interlock.**
PxMRLN	I	**Port x Manually-operated Retention Latch (MRL) Input.**
PxPDN	I	**Port x Presence Detect Input.**
PxPEP	O	**Port x Power Enable Output.**
PxPFN	I	**Port x Power Fault Input.**
PxPIN	O	**Port x Power Indicator Output.**
PxPWRGDN	I	**Port x Power Good Input (asserted when slot power is good).**
PxRSTN	O	**Port x Reset Output.**

Table 9.1 Port Hot Plug Signals

1. x corresponds to port number (i.e., 0 through 7).

The negated value for an unused hot-plug I/O expander output is the value shown in Table 9.2. The value is equal to the default value as indicated by the signal name suffix (i.e., N for active low and P for active high) modified as indicated by the corresponding invert polarity bit in the Hot-Plug Configuration Control (HPCFGCTL) register.

Signal	Negated Output Value with Non-Inverted Polarity (IPXxxx = 0)	Negated Output Value with Inverted Polarity (IPXxxx = 1)
PxAIN	1 (high)	0 (low)
PxILOCKP	0 (low)	1 (high)
PxPEP	0 (low)	1 (high)
PxPIN	1 (high)	0 (low)
PxRSTN	1 (high)	0 (low)

Table 9.2 Negated Value of Unused Hot-Plug Output Signals

Notes

PES48T12G2 utilizes external SMBus/I2C-bus I/O expanders connected to the master SMBus interface for hot-plug related signals associated with downstream ports. See section I/O Expanders on page 12-5 for details on the operation of the I/O expanders and for the mapping of hot-plug signals to I/O expander inputs and outputs.

SMBus I/O expander transactions are automatically initiated when the state of a hot-plug input signal changes or a new value needs to be driven on a hot-plug output signal.

When an IO Expander is initialized (i.e., the IOEXPADDR field in the IOEXPADDR[3:0] registers is written), the hot-plug controller for the corresponding port initiates an SMBus access to configure the IO Expander and updates the status bits in the PCI Express Slot Status (PCIESSTS) register.

- During this initial access, the Presence Detect Changed (PDC) and MRL Sensor Changed (MRLSC) bits in the PCIESSTS register are not set, since this access is used to determine the initial state of the IO Expander signals.

PES48T12G2 supports presence detect signalling via assertion of the Presence Detect Input signal in the external I/O Expander module and through “in-band” presence detect. The Presence Detect Control (PDETECT) field in the Hot-Plug Configuration Control (HPCFGCTL) register may be used to control the mechanism used for presence detect.

Since the polarity of hot-plug signals has been defined differently in various specifications, each hot plug signal has a corresponding control bit in the Hot-Plug Configuration Control (HPCFGCTL) that allows the polarity of that signal to be inverted. Inversion affects the corresponding signal in all ports.

When a one is written to the EIC bit in the PCIESCTL register, then the PxILOCKP signal is pulsed for a length greater than 100 ms and less than 150 ms (i.e., it transitions from negated to asserted, maintains an asserted state for 100 to 150 ms, and then transitions back to negated). When the Toggle Electromechanical Interlock Control (TEMICTL) bit in the HPCFGCTL register is set, writing a one to the EIC bit inverts the state of the PxILOCKP signal.

When the MRL Automatic Power Off (MRLPWROFF) bit is set in the HPCFGCTL register and the Manual Retention Latch Sensor Present (MRLP) bit is set in the PCI Express Slot Capability (PCIESCAP) register, then power to the slot is automatically turned off when the MRL sensor indicates that the MRL is open. This occurs regardless of the state of the Power Controller Control (PCC) bit in the PCI Express Slot Control (PCIESCTL) register.

The state of a port’s Power Fault (PxPFN) input is not latched. For proper operation the system designer should ensure that once the PxPFN signal is asserted, it remains asserted until the power enable (PxPEP) signal is toggled. This is required adapter behavior for the PCI Express ExpressModule form factor.

The default value of hot-plug registers following a switch fundamental reset may be configured via serial EEPROM initialization. Since hot-plug I/O expander initialization occurs after serial EEPROM initialization, the Command Completed (CC) bit is not set in the PCI Express Slot Status (PCIESSTS) register as a result of serial EEPROM initialization.

Following a hot-reset, an upstream secondary bus hot-reset, or a downstream secondary bus hot-reset, each downstream port’s PHY will transition the link to the Hot-Reset state and subsequently re-train the link starting from the Detect state. When this occurs, the Hot-Plug controller for the port does not set the Presence Detect Changed (PDC) bit in the PCIESSTS register.

Port Reset Outputs

Individual port reset outputs PxRSTN are provided as I/O expander outputs.

Port reset outputs may be configured to operate in one of two modes. These modes are power enable controlled reset output and power good controlled reset output. The port reset output mode for all ports is determined by the Reset Mode (RSTMODE) field in the Hot-Plug Configuration Control (HPCFGCTL) register.

Notes

In addition to a port reset output being asserted as determined by the Reset Mode (RSTMODE) field, a port reset output is also asserted under the following circumstances.

- When the switch experiences a fundamental reset.
- When the operating mode of a port is modified and the OMA field is set to reset.

Hardware ensures that the minimum port reset output assertion pulse width is no less than 200 µS.

Power Enable Controlled Reset Output

In this mode a downstream port reset output state is controlled as a side effect of slot power being turned on or off. The operation of this mode is illustrated in Figure 9.4. A downstream port's slot power is controlled by the Power Controller Control (PCC) bit in the PCI Express Slot Control (PCIESCTL) register

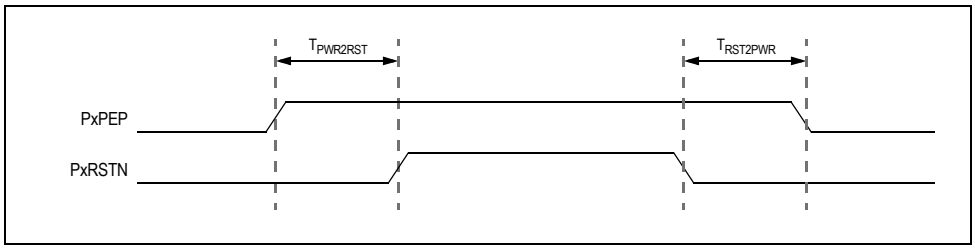


Figure 9.4 Power Enable Controlled Reset Output Mode Operation

While slot power is disabled, the corresponding downstream port reset output is asserted.

When slot power is enabled by writing a zero to the PCC bit, the Port x Power Enable Output (PxPEP) is asserted and then power to the slot is enabled and the corresponding downstream port reset output is negated. The time between the assertion of the PxPEP signal and the negation of the PxRSTN signal is controlled by the value in the Slot Power to Reset Negation (PWR2RST) field in the HPCFGCTL register.

While slot power is enabled, the corresponding downstream port reset output is negated.

When slot power is disabled by writing a one to the PCC bit, the corresponding downstream port reset output is asserted and then slot power is disabled. The time between the assertion of the PxRSTN signal and the negation of the PxPEP signal is controlled by the value in the Reset Negation to Slot Power (RST2PWR) field in the HPCFGCTL register.

Power Good Controlled Reset Output

As in the Power Enable Controlled Reset mode, in this mode a downstream port reset output state is controlled as a side effect of slot power being turned on or off. However, the timing in this mode depends on the power good state of the slot's power supply. The operation of this mode is illustrated in Figure 9.5.

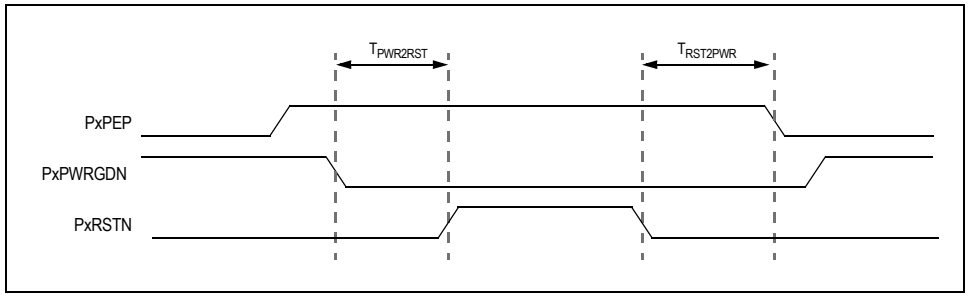


Figure 9.5 Power Good Controlled Reset Output Mode Operation

Notes

The operation of this mode is similar to that of the Power Enable Controlled Reset mode except that when power is enabled, the negation of the corresponding port reset output occurs as a result of and after assertion of the slot's Power Good (PxPWRGDN) signal is observed. The time between the assertion of the PxPWRGDN signal and the negation of the PxRSTN signal is controlled by the value in the Slot Power to Reset Negation (PWR2RST) field in the HPCFGCTL register.

When slot power is disabled by writing a one to the PCC bit, the corresponding downstream port reset output is asserted and then slot power is disabled. The time between the assertion of the PxRSTN signal and the negation of the PxPEP signal is controlled by the value in the Reset Negation to Slot Power (RST2PWR) field in the HPCFGCTL register.

If at any point while a downstream port is not being reset (i.e., PxRSTN is negated) the power good (i.e., PxPWRGDN) is negated, then the corresponding port reset output is immediately asserted.

Hot-Plug Events

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

Hot-plug interrupts are enabled 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), Command Completed (CC), and Data Link Layer Active State Change (DLLASC).

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 is in a $D3_{hot}$ state, then the hot-plug controller generates a wakeup event using a PM_PME message instead of an interrupt when the following conditions are satisfied. The status bit for an enabled hot-plug event listed below transitions from not set to set.

- Attention button pressed
- Power fault detected
- MRL sensor changed
- Presence detect changed
- Command completed event
- Data link layer state change event

The PME Enable (PMEE) bit in the PCI Power Management Control and Status (PMCSR) is set. If is not required that the Hot Plug Interrupt Enable (HPIE) bit be set in the corresponding port's PCI Express Slot Control (PCIESCTL) register in order to generate a wakeup event using a PM_PME message. Software may clear the HPIE bit to disable interrupt generation while keeping wakeup event generation enabled.

If a hot-plug event occurs while a downstream port is in D3hot and the corresponding interrupt is enabled, the port will generate an interrupt if the corresponding event's status bit is set in the PCIESCTL is set and the state of the port is transitioned from $D3_{hot}$ to D0 without a reset.

Legacy System Hot-Plug Support

Some systems require support for operating systems that lack PCIe hot-plug support. PES48T12G2 supports these systems by providing a General Purpose Event (GPEN) output as a GPIO alternate function that can be used instead of the INTx, MSI, and PME mechanisms defined by PCI Express hot-plug.

Associated with each downstream port's hot-plug controller is a bit in the General Purpose Event Control (GPECTL) register. When this bit is set, the corresponding PCIe base 2.0 hot plug event notification mechanisms are disabled for that port and INTx, MSI and PME events will not be generated by that port due to hot-plug events. Instead, hot-plug events are signaled through assertion of the GPEN signal.

Notes

GPEN is a GPIO alternate function. The GPIO pin will not be asserted when GPEN is asserted unless it is configured to operate as an alternate function. Whenever a port signals a hot-plug event through assertion of the GPEN signal, the corresponding port's status bit in the General Purpose Event Status (GPESTS) register is set. A bit in the GPESTS register can only be set if the corresponding port's hot plug controller is configured to signal hot-plug events using the general purpose event (GPEN) signal assertion mechanism.

The hot-plug event signalling mechanism is the only thing that is affected when a port is configured to use general purpose events instead of the PCIe defined hot-plug signalling mechanisms (i.e., INTx, MSI and PME). Thus, the PCIe defined capability, status and mask bits defined in the PCIe slot capabilities, status and control registers operate as normal and all other hot-plug functionality associated with the port remains unchanged. INTx, MSI and PME events from other sources are also unaffected.

The enhanced hot-plug signalling mechanism supported by the PES48T12G2 is graphically illustrated in Figure 9.6. This figure provides a conceptual summary of the enhanced hot-plug signalling mechanism in the form of a pseudo logic diagram. Logic gates in this diagram are intended for conveying general concepts, and not for direct implementation.

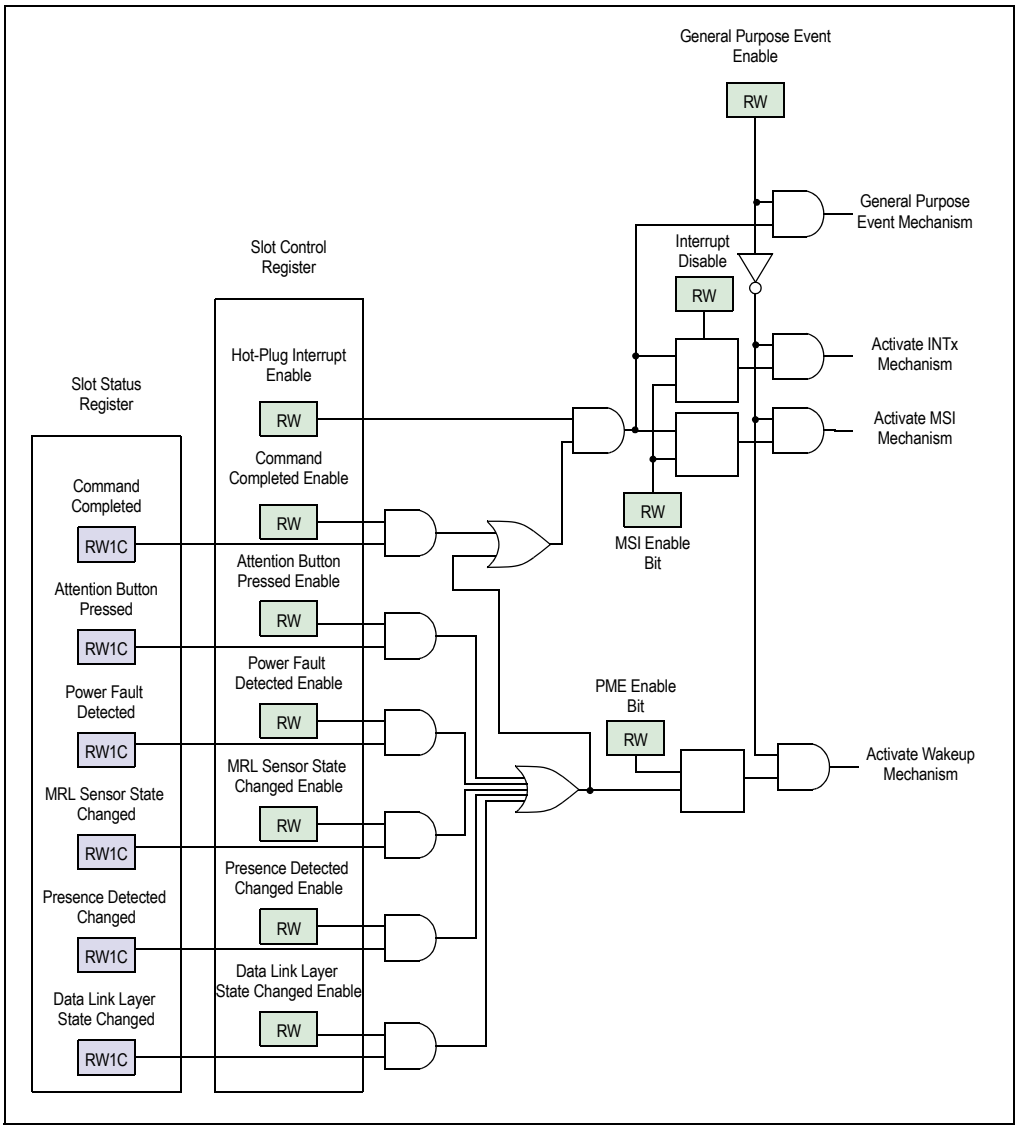


Figure 9.6 PES48T12G2 Hot-Plug Event Signalling

Notes

Hot-Swap

PES48T12G2 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, PES48T12G2 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 PES48T12G2 may also be used to construct proprietary hot-swap systems. See the PES48T12G2 Data Sheet for a detailed specification of I/O buffer characteristics.

Power Management

Notes

Introduction

Located in configuration space of each Function in the PES48T12G2 (i.e., PCI-to-PCI Bridge Function) is a power management capability structure. PES48T12G2 Functions support the following device power management states:

- D0 ($D0_{uninitialized}$ and $D0_{active}$)
- $D3_{Hot}$
- $D3_{Cold}$.

A power management state transition diagram for the states supported by the PES48T12G2 is provided in Figure 10.1 and described in Table 10.1. Transitioning a Function's power management state from $D3_{hot}$ to $D0_{uninitialized}$ does not result in any logic being reset or re-initialization of register values. Thus, the default value of the No Soft Reset (NOSOFTRST) bit in the Function's PCI Power Management Control and Status (PMCSR) register corresponds to the functional context being maintained in the $D3_{hot}$ state.

The power management capability structure associated with each Function affects the power state of that Function only. When the upstream PCI-to-PCI bridge Function enters the $D3_{Hot}$ state and the PME Turn Off protocol is completed (i.e., PME_TO_Ack messages are received), then the switch is placed into a low power state.

The link's power state is derived from the power state of the Function(s) in the port. When a Function enters the D0 state (i.e., $D0_{uninitialized}$ or $D0_{active}$), the Function transitions the port's link to the L0 state. When a Function enters the $D3_{Hot}$ state, the Function transitions the port's link to the L1 state.

- A port configured in 'Upstream Switch Port' mode initiates entry into L1 when all the Functions in the port enter the $D3_{Hot}$ state.
- A port configured in 'Downstream Switch Port' mode always accepts entry into L1 when requested by the link partner.[1]

[1] This applies when entry into L1 is a result of the link partner being placed in D3hot. This does not apply for entry into L1-ASPM, where the L1 entry request may be rejected by the downstream port. Refer to section Active State Power Management on page 6-11 for further details on L1 ASPM.

Notes

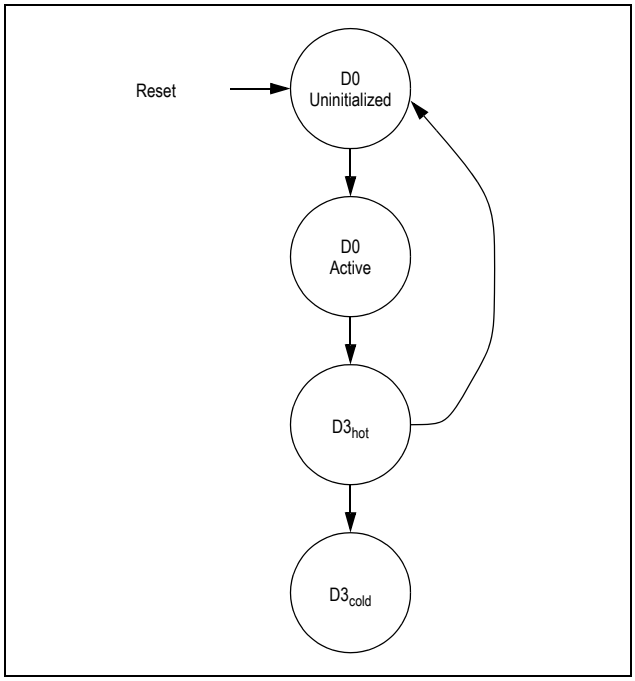


Figure 10.1 PES48T12G2 Power Management State Transition Diagram

From State	To State	Description
any	D0 Uninitialized	Switch reset (any type).
D0 Uninitialized	D0 Active	Function 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.
$D3_{hott}$	$D3_{cold}$	Power is removed from the device.

Table 10.1 PES48T12G2 Power Management State Transition Diagram

PCI-to-PCI bridge Functions have the following behavior when in the $D3_{hot}$ power management state.

- A bridge accepts, processes and completes all type 0 configuration read and write requests.
- Accepts and processes all message requests that target the bridge.
- All requests received by the bridge on the primary interface, except as noted above, are treated as unsupported requests (UR).
- Any error message resulting from the receipt of a TLP is reported in the same manner as when the bridge is not in $D3_{hot}$ (e.g, generation of an ERR_NONFATAL message to the root).
  - This requires transitioning the link to the L0 state when error reporting is enabled and the link is not in L0.
- Error messages resulting from any event other than the receipt of a TLP are discarded (i.e., no error message is generated).
- All completions that target the bridge are treated as unexpected completions (UC).
- Completions flowing in either direction through the bridge are routed as normal. This behavior of the bridge does not differ from that of the bridge when it is in the D0 power management state.

Notes

- This requires transitioning the link to the L0 state when the completion needs to be transmitted on the link by the bridge Function and the link is not in L0.

– All request TLPs received on the secondary interface are treated as unsupported requests (UR).

PME Messages

PES48T12G2 does not support generation of PME messages from the $D3_{cold}$ state. PME message generation is only supported by downstream ports (i.e., the PCI-to-PCI bridge Function associated with an upstream port does not support PME message generation).

Downstream ports (i.e., PCI-to-PCI bridges associated with downstream ports) 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.

PCI Express Power Management Fence Protocol

The root complex takes the following steps to turn off power to a system.

– The root places all devices in the D3 state

– Upon entry to D3, all devices transition their links to the L1 state

– The root broadcasts a PME_Turn_Off message.

- The links temporarily transition to L0 in order to transfer the message.

– Devices acknowledge the PME_Turn_Off message by returning a PME_TO_ACK message

- After transmitting a PME_TO_ACK, a device places its link in L2/L3-Ready state.

The PME_Turn_Off / PME_TO_Ack protocol may be initiated by the root when the switch Function's are in any power management state. The port's handling of the power management fence protocol depends on its operating mode as described below.

Upstream Switch Port or Downstream Switch Port Mode

– When a port configured in 'Upstream Switch Port' mode receives a PME_Turn_Off message, it broadcasts the PME_Turn_Off message on all active downstream ports. The upstream port transmits a PME_TO_Ack message and transitions its link state to L2/L3 Ready after it has received a PME_TO_Ack message on each of its active downstream ports. This process is called PME_TO_Ack aggregation.

– The aggregation of PME_TO_Ack messages on downstream ports is abandoned by the upstream switch port when this port receives a TLP after having previously received a PME_Turn_Off message but before having responded with a PME_TO_Ack message. Once a PME_TO_Ack message has been scheduled for transmission on the upstream switch port and the PME_TO_Ack aggregation process has completed, received TLPs at that point are discarded.

– If the TLP that causes PME_TO_Ack aggregation to be abandoned targets a PES48T12G2 Function, then the targeted Function responds to the TLP normally. If the TLP that causes aggregation to be abandoned is routed to a downstream port's link and the link is in L0, then the TLP is transmitted on the downstream port. If the downstream port's link is not in L0 (i.e., it is in L2/L3 Ready), then the port transitions the link to Detect and then to L0. Once the link is reaches L0, the TLP is transmitted on the downstream port.

– When PME_TO_Ack aggregation is abandoned, the PES48T12G2 makes no attempt to abandon the PME_Turn_Off and PME_TO_Ack protocol on downstream ports. Devices downstream of PES48T12G2 are allowed to respond with a PME_TO_Ack and transition to L2/L3 Ready. When a TLP is received that needs to be routed to the downstream port's link, then the switch transitions the link to Detect and then to L0. Once the link reaches L0, the TLP is transmitted on the downstream port.

Notes

Power Budgeting Capability

PES48T12G2 contains the mechanisms necessary to implement the PCI-Express power budgeting enhanced capability. However, by default, these mechanisms are not enabled. To enable the power budgeting capability, registers in this capability should be initialized and the Next Pointer (NXTPTR) field in one of the other enhanced capabilities should be initialized to point to the power budgeting capability. The Next Pointer (NXTPTR) of the power budgeting capability should be adjusted if necessary.

Each PES48T12G2 Function contains a power budgeting capability structure. This structure consists of the four power budgeting capability registers defined in the PCI Express 2.0 base specification and eight general purpose read-write registers.

The Power Budgeting Capabilities (PWRBCAP) register contains the PCI-Express enhanced capability header for the power budgeting capability. By default, this register has an initial read-only value of zero. To enable the power budgeting capability, this register should be initialized via the serial EEPROM. The Power Budgeting Data Value [7:0] (PWRBDV[7:0]) registers are used to hold the power budgeting information for that Function in a particular operating condition.

The PWRBDV registers may be read and written when the Power Budgeting Data Value Unlock (PWRBDVUL) bit is set in the Switch Control (SWCTL) register. When the PWRBDVUL bit is cleared, these registers are read-only and writes to these registers are ignored. To enable the power budgeting capability, the PWRBDV registers should be initialized with power budgeting information via the serial EEPROM.

General Purpose I/O

Notes

Introduction

The PES48T12G2 has 9 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 Function [1:0] (GPIOFUNCx), General Purpose I/O Configuration [1:0] (GPIOCFGx), General Purpose I/O Data [1:0] (GPIODx), and General Purpose I/O Alternate Function Select [1:0] (GPIOAFSELx) registers. After a switch fundamental reset, all GPIO pins default to a GPIO input function.

GPIO pins configured as GPIO inputs are double-clocked and sampled no more frequently than once every 128 ns. Thus, they may be treated as asynchronous inputs. Associated with each GPIO pin are alternate functions. Care should be exercised when configuring GPIO pins as outputs since an incorrect configuration could cause damage to the PES48T12G2 and external components.

GPIO Configuration

Associated with each GPIO pin is a bit in the GPIOFUNC, GPIOCFG and GPIOD registers. Table 11.1 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 11.1 GPIO Pin Configuration

Configured as an Input

When configured as an input in the GPIOCFGx register and as a GPIO function in the GPIOFUNCx register, the GPIO pin is sampled and registered in the GPIODx register.

The value of the input pin can be determined at any time by reading the GPIODx register. The value in this register corresponds to the value of the pin irrespective of whether the pin is configured as a GPIO input, GPIO output, or alternate function.

Configured as an Output

When configured as an output in the GPIOCFGx register and as a GPIO function in the GPIOFUNCx register, the value in the corresponding bit position of the GPIODx register is driven on the pin. System designers should treat the GPIO outputs as asynchronous outputs. The actual value of the output pin may be determined by reading the GPIODx register.

Configured as an Alternate Function

Most GPIO pins have an alternate function. The alternate function associated with a specific GPIO pin is selected by the corresponding field in the GPIOAFSEL register. The alternate functions associated with each GPIO pin are listed in Table 11.2.

Notes

GPIO Pin	Alternate Function 0	Alternate Function 1
4	—	P0LINKUPN
5	GPEN	P0ACTIVEN
8	IOEXPINTN	—

Table 11.2 General Purpose I/O Pin Alternate Function

Alternate function signals are described in Table 11.3.

Signal	Type	Name/Description
GPE	O	**General Purpose Event.** Hot-plug general purpose event output
IOEXPINTN	I	**I/O Expander x Interrupt Input.** I/O expander interrupt
PxACTIVEN	O	**Port x Link active status output.**
PxLINKUPN	O	**Port x link up status output.**

Table 11.3 GPIO Alternate Function Pins

When configured as an alternate function in the GPIOFUNCx register, a pin behaves as the alternate function signal selected by the GPIOAFSEL register. If the alternate function signal is an input, the GPIO pin behaves as an input. If the alternate function signal is an output, the GPIO pin behaves as an output.

The value of the alternate function pin may be determined at any time by reading the corresponding GPIODx register. When an alternate function input signal is not enabled on any GPIO (or I/O expander for hot-plug signals), the alternate function signal is internally held in an inactive state.

SMBus Interfaces

Notes

Introduction

PES48T12G2 has two SMBus interfaces. The slave SMBus interface provides full access to all software visible registers, allowing every register in the device to be read or written by an external SMBus master. The slave SMBus may also be used to program the serial EEPROM used for initialization. The Master SMBus interface provides connection for an optional external serial EEPROM used for initialization and optional external I/O expanders. 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 12.1, the master and slave SMBuses may only be used in a split configuration.

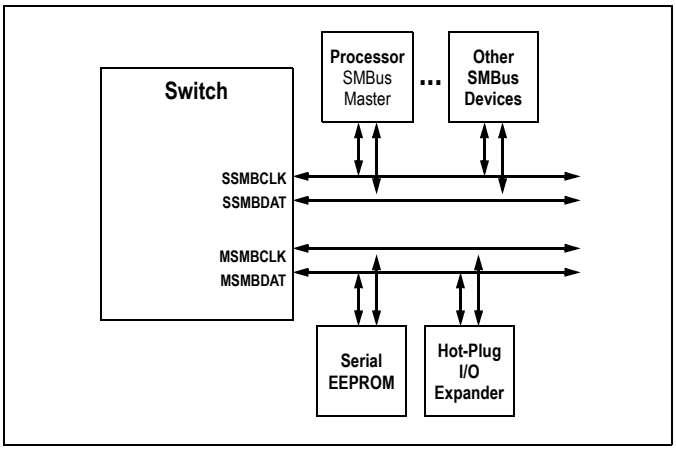


Figure 12.1 Split SMBus Interface Configuration

The switch's SMBus master interface does not support SMBus arbitration. As a result, the switch's SMBus master must be the only master in the SMBus lines that connect to the serial EEPROM and I/O expander slaves. In the split configuration, the master and slave SMBuses operate as two independent buses; thus, multi-master arbitration is not required.

Master SMBus Interface

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

Initialization

The Master SMBus initialization occurs during a switch fundamental reset (see section Switch Fundamental Reset on page 5-2). The Master SMBus Clock Prescalar (MSMBCP) field in the SMBus Control (SMBUSCTL) register is configured to support 100 KHz SMBus operation.

Serial EEPROM

During a switch 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) signal selects an operating mode that performs serial EEPROM initialization. The address used by the SMBus interface is set to default 101000b.

Notes

Initialization from 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 PES48T12G2. Any software visible register in the device may be initialized with values stored in the serial EEPROM.

All software visible registers have a system address. Configuration blocks stored in the serial EEPROM use this system address shifted right two bits (i.e., configuration blocks in the serial EEPROM use DWord addresses and not byte addresses).

– See Chapter 14, Register Organization, for the details on the system address and the address map.

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 12.1 may be used to store initialization values.

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

Table 12.1 PES48T12G2 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. When a serial EERPROM address roll over is detected, loading of the serial EEPROM is aborted, the Serial EEPROM Rollover (ROLLOVER) bit is set in the SMBus Status (SMBUSSTS) register, and the RSTHALT bit is set in the SWCTL register.

A blank serial EEPROM contains 0xFF in all data bytes. When the PES48T12G2 is configured to initialize from serial EEPROM and the first 256 bytes read from the EEPROM all contain the value 0xFF, then loading of the serial EEPROM is aborted, the computed checksum is ignored, the Blank Serial EEPROM (BLANK) bit is set in the SMBus Status (SMBUSSTS) register, and normal device operation begins (i.e., the device operates in the same manner as though it were not configured to initialize from the serial EEPROM).

This behavior allows a board manufacturing flow that utilizes uninitialized serial EEPROMs. See section Programming the Serial EEPROM on page 12-5 for information on in-system initialization of the serial EEPROM.

All register initialization performed by the serial EEPROM is performed in DWord quantities.

– Byte values may be modified by writing the entire DWord.

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 seven bytes in the serial EEPROM and is used to initialize a single double word register quantity.

A single double word initialization sequence consists of three fields and its format is shown in Figure 12.2. The TYPE field indicates the type of the configuration block. For single double word initialization sequence, this value is always 0x0. The SYSADDR field contains the upper 16-bits of the global system address of the double word to be initialized. The actual global system address, which is a byte address, equals this value with two lower zero bits appended. The final DATA field contains the double word initialization value.

Notes

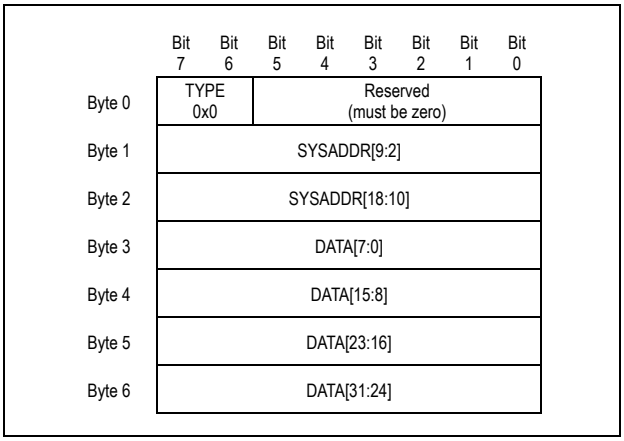


Figure 12.2 Single Double Word Initialization Sequence Format

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 12.3. The TYPE field indicates the type of the configuration block. For sequential double word initialization sequences, this value is always 0x1. The SYSADDR field contains the starting double word system address to be initialized. 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.

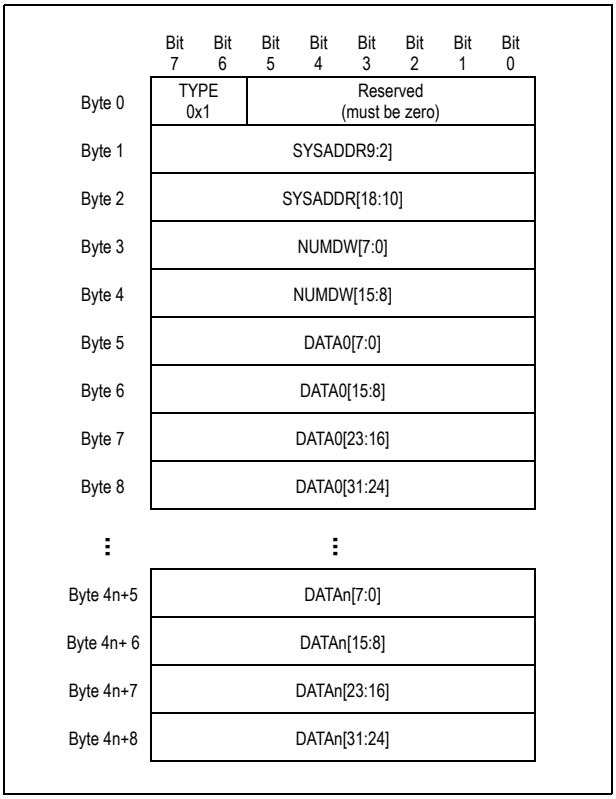


Figure 12.3 Sequential Double Word Initialization Sequence Format

Notes

The final type of configuration block is the configuration done sequence which is used to signify the end of a serial EEPROM initialization sequence. If during serial EEPROM initialization, an attempt is made to initialize a register that is not defined in a configuration space (i.e., not defined in Chapter 14), 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 12.4. The TYPE field is always 0x3 for configuration done sequences. 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.

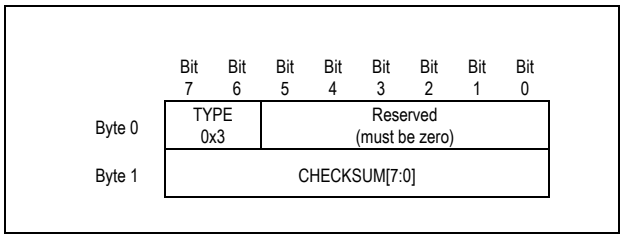


Figure 12.4 Configuration Done Sequence Format

The checksum in the configuration done sequence enables the integrity of the serial EEPROM initialization to be verified. 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 SMBus Control (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 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 SMBUSSTS register. Once serial EEPROM initialization completes, is aborted, or when an error is detected, the EEPROM Done (EEPROMDONE) bit is set in the SMBus Status (SMBUSSTS) register.

A summary of possible errors during serial EEPROM initialization and specific action taken when detected is summarized in Table 12.2.

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	- Set RSTHALT bit in SWCTL register - ICSERR bit is set in the SMBUSSTS register - Abort initialization, set DONE bit in the SMBUSSTS register
Invalid configuration block type (only invalid type is 0x2)	- Set RSTHALT bit in SWCTL register - ICSERR bit is set in the SMBUSSTS register - Abort initialization, set DONE bit in the SMBUSSTS register
An unexpected NACK is observed during a master SMBus transaction	- Set RSTHALT bit in SWCTL register - NAERR bit is set in the SMBUSSTS register - Abort initialization, set DONE bit in the SMBUSSTS register
A misplaced START or STOP condition is detected by the master SMBus interface	- Set RSTHALT bit in SWCTL register - OTHERERR bit is set in the SMBUSSTS register - Abort initialization, set DONE bit in the SMBUSSTS register
Serial EEPROM address rollover error detected	- Set RSTHALT bit in SWCTL register - ROLLOVER bit is set in the SMBUSSTS register - Abort initialization, set DONE bit in the SMBUSSTS register
Blank serial EEPROM detected	- BLANK bit is set in the SMBUSSTS register - Abort initialization, set DONE bit in the SMBUSSTS register

Table 12.2 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 12-16. A PCIe root may read and write the serial EEPROM by performing configuration read and write transactions to the Serial EEPROM Interface (EEPROMINTF) register.

To read a byte from the serial EEPROM, the root should configure the Address (ADDR) field in the 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 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 SMBus Status (SMBUSSTS) register. Software should check for errors before and after each serial EEPROM access.

I/O Expanders

PES48T12G2 utilizes external SMBus/I2C-bus I/O expanders connected to the master SMBus interface for hot-plug and port status signals. PES48T12G2 is designed to work with Phillips PCA9555 compatible I/O expanders (i.e., PCA9555, PCA9535, and PCA9539). See the Phillips PCA9555 data sheet for details on the operation of this device. An external SMBus I/O expander provides 16 bit I/O pins that may be configured as inputs or outputs.

Notes

PES48T12G2 supports up to 14 external I/O expanders. Table 12.3 summarizes the allocation of functions to I/O expanders. I/O expander signals associated with LED control (i.e., link status and activity) are active low (i.e., driven low when an LED should be turned on). I/O expander signals associated with hot-plug signals are not inverted.

SMBus I/O Expander	Section	Function
0	Lower	Port 0 hot-plug
	Upper	Port 2 hot-plug
1	Lower	Port 1 hot-plug
	Upper	Port 3 hot-plug
2	Lower	Port 4 hot-plug
	Upper	Port 6 hot-plug
3	Lower	Port 5 hot-plug
	Upper	Port 7 hot-plug
4	Lower	Port 8 hot-plug
	Upper	Unused
5	Lower	Port 9 hot-plug
	Upper	Unused
6	Lower	Port 12 hot-plug
	Upper	Unused
7	Lower	Port 13 hot-plug
	Upper	Unused
8	Lower / Upper	Hot-plug MRL inputs
9	Lower / Upper	Ports 0 through 7 hot-plug electromechanical interlock
10	Lower / Upper	Ports 8, 9, 12, 13 hot-plug electromechanical interlock
11	Lower / Upper	Fundamental reset inputs
12	Lower / Upper	Link status
13	Lower / Upper	Link activity

Table 12.3 I/O Expander Function Allocation

During PES48T12G2 initialization, the SMBus/I2C-bus address allocated to each I/O expander used in that system configuration should be written to the corresponding IO Expander Address (IOE[13:0]ADDR) field. The IOExADDR fields are contained in the IO Expander Address (IOEXPADDR[3:0]) registers.

Hot-plug outputs and I/O expanders may be initialized via serial EEPROM. Since the I/O expanders and serial EEPROM both utilize the master SMBus, no I/O expander transactions are initiated until serial EEPROM initialization completes. Since no I/O expander transactions are initiated until serial EEPROM initialization completes, it is not possible to toggle a hot-plug output through serial EEPROM initialization (i.e., it is not possible to cause a 0 -> 1 -> 0 transition or a 1 -> 0 -> 1 transition).

Whenever the value of an IOEXPADDR field is written, SMBus write transactions are issued to the corresponding I/O expander by PES48T12G2 to configure the device. This configuration initializes the direction of each I/O expander signal and sets outputs to their default value.

Notes

Outputs for ports that are disabled, mapped to GPIO alternate functions, or are not implemented in that configuration, are set to their negated value (e.g., the power indicator is turned off, the link is down, there is no activity, etc.). The default value of I/O expander outputs is shown in Table 12.4. Note that this default value may be modified via serial EEPROM or SMBus configuration prior to SMBus initialization by changing the state of the PCI Express Slot Control Register (PCIESCTL) or Hot-Plug Configuration Control (HPCFGCTL).

Hot-Plug Signal	Description	Default Value
PxAIN	Attention indicator output (off)	1
PxPIN	Power indicator output (on)	0
PxPEP	Power enable output (on)	1
PxILOCKP	Electromechanical interlock (negated - off)	0

Table 12.4 I/O Expander Default Output Signal Value

The following I/O expander configuration sequence is issued by PES48T12G2 to I/O expanders 0 through 7, 9 and 10 (i.e., the ones that contain general port hot-plug signals and electromechanical interlock signals).

- Write the default value of the outputs bits on the lower eight I/O expander pins (i.e., I/O-0.0 through I/O-0.7) to I/O expander register 2.
- Write the default value of the outputs bits on the upper eight I/O expander pins (i.e., I/O-1.0 through I/O-1.7) 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 the configuration value to select inputs/outputs in the lower eight I/O expander bits (i.e., I/O-0.0 through I/O-0.7) to I/O expander register 6.
- Write the configuration value to select inputs/outputs in the upper eight I/O expander bits (i.e., I/O-1.0 through I/O-1.7) to I/O expander register 7.
- Read value of I/O expander register 0 to obtain the current state of the lower eight I/O expander bits (i.e., I/O-0.0 through I/O-0.7)
- read value of I/O expander register 1 to obtain the current state of the upper eight I/O expander bits (i.e., I/O-1.0 through I/O-1.7)

The following I/O expander configuration sequence is issued by PES48T12G2 to I/O expanders 8 and 11 (i.e., the ones that contain MRL and fundamental reset inputs).

- 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 the configuration value to select all inputs in the lower eight I/O expander bits (i.e., I/O-0.0 through I/O-0.7) to I/O expander register 6.
- Write the configuration value to select all inputs in the upper eight I/O expander bits (i.e., I/O-1.0 through I/O-1.7) to I/O expander register 7.
- Read value of I/O expander register 0 to obtain the current state of the lower eight I/O expander bits (i.e., I/O-0.0 through I/O-0.7)
- read value of I/O expander register 1 to obtain the current state of the upper eight I/O expander bits (i.e., I/O-1.0 through I/O-1.7)

Notes

The following I/O expander configuration sequence is issued by PES48T12G2 to I/O expanders 12 and 13 (i.e., the one that contains link up and link activity status).

- Write link up status for all ports to the lower eight I/O expander pins (i.e., I/O-0.0 through I/O-0.7) to I/O expander register 2.
- Write link activity status for all ports to the upper eight I/O expander pins (i.e., I/O-1.0 through I/O-1.7) 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 the configuration value to select all outputs in the lower eight I/O expander bits (i.e., I/O-0.0 through I/O-0.7) to I/O expander register 6.
- Write the configuration value to select all outputs in the upper eight I/O expander bits (i.e., I/O-1.0 through I/O-1.7) to I/O expander register 7.

While the I/O expander is enabled, PES48T12G2 maintains the I/O bus expander signals and the PES48T12G2 internal view of the hot-plug signals in a consistent state. This means that whenever that I/O bus expander state and the internal view of the signal state differs, an SMBus transaction is initiated to resolve the state conflict.

- An example of an event that may lead to a state conflict is a hot reset. When a hot reset occurs, one or more hot-plug register control fields may be re-initialized to its default value. When this occurs, the internal state of the hot-plug signals is in conflict with the state of I/O expander hot-plug output signals. In such a situation, PES48T12G2 will initiate an SMBus transaction to modify the state of the I/O expander hot-plug outputs.

PES48T12G2 has one combined I/O expander interrupt input, labeled IOEXPINTN, which is a GPIO alternate function. Associated with each I/O expander is an open drain interrupt output that is asserted when an I/O expander input pin changes state. The open drain I/O expander interrupt output of all I/O expanders should be tied together on the board and connected to the appropriate GPIO. Whenever IOEXPINTN is asserted, PES48T12G2 reads the state of all I/O expanders. Since the I/O expander interrupt input is a GPIO alternate function, the corresponding GPIO should be initialized during configuration to operate in alternate function mode (for further information, see Chapter 12, General Purpose I/O.)

Whenever PES48T12G2 needs to change the state of an I/O expander signal output, a master SMBus transaction is initiated to update the state of the I/O expander. This write operation causes the corresponding I/O expander to change the state of its output(s). PES48T12G2 will not update the state of an I/O expander output more frequently than once every 40 milliseconds. This 40 millisecond time interval is referred to as the I/O expander update period.

Whenever an input to the I/O expander changes state from the value previously read, the interrupt output of the I/O expander is asserted. This causes PES48T12G2 to issue a master SMBus transaction to read the updated state of all I/O expander inputs. Regardless of the state of the interrupt output of an I/O expander, PES48T12G2 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 (i.e., the I/O expander update period). This delay in sampling may be used to eliminate external debounce circuitry.

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. PES48T12G2 ensures that I/O expander transactions are initiated on the master SMBus in a fair manner. This guarantees that all I/O expanders have equal service latencies. Any errors detected during I/O expander SMBus read or write transactions is reflected in the status bits of the SMBus Status (SMBUSSTS) register.

Notes

The following are system design recommendations:

- I/O expander addresses and default output values may be configured during serial EEPROM initialization. If I/O expander addresses are configured via the serial EEPROM, then PES48T12G2 will initialize the I/O expanders when normal device operation begins following the completion of the fundamental reset sequence.
- If the I/O expanders are initialized via serial EEPROM, then the data value for output signals during the SMBus initialization sequence will correspond to those at the time the SMBus transactions are initiated. It is not possible to toggle SMBus I/O expander outputs by modifying data values during serial EEPROM initialization.
- During a fundamental reset and before the I/O expander outputs are initialized, all I/O expander output signals default to inputs. Therefore, pull-up or pull-down resistors should be placed on outputs to ensure that they are held in the desired state during this period.
- All hot-plug data value modifications that correspond to hot-plug outputs result in SMBus transactions. This includes modifications due to upstream secondary bus resets and hot-resets.
- I/O expander outputs are not modified when the device transitions from normal operation to a fundamental reset. In systems where I/O expander output values must be reset during a fundamental reset, a PCA9539 I/O expander should be used.

Hot-Plug I/O Expanders 0 through 7

SMBus I/O Expander Bit	Type	Signal	Description
0 (I/O-0.0)[1]	I	PxAPN	Port *x* attention push button input
1 (I/O-0.1)	I	PxPDN	Port *x* presence detect input
2 (I/O-0.2)	I	PxPFN	Port *x* power fault input
3 (I/O-0.3)	I	PxPWRGDN	Port *x* power good input
4 (I/O-0.4)	O	PxAIN	Port *x* attention indicator output
5 (I/O-0.5)	O	PxPIN	Port *x* power indicator output
6 (I/O-0.6)	O	PxPEP	Port *x* power enable output
7 (I/O-0.7)	O	PxRSTN	Port *x* reset output
8 (I/O-1.0)	I	P*(x+2)*APN	Port *(x+2)* attention push button input
9 (I/O-1.1)	I	P*(x+2)*PDN	Port *(x+2)* presence detect input
10 (I/O-1.2)	I	P*(x+2)*PFN	Port *(x+2)* power fault input
11 (I/O-1.3)	I	P(x+2)PWRGDN	Port (x+2) power good input
12 (I/O-1.4)	O	P*(x+2)*AIN	Port *(x+2)* attention indicator output
13 (I/O-1.5)	O	P*(x+2)*PIN	Port *(x+2)* power indicator output
14 (I/O-1.6)	O	P*(x+2)*PEP	Port *(x+2)* power enable output
15 (I/O-1.7)	O	P(x+2)RSTN	Port *(x+2)* reset output

Table 12.5 Pin Mapping for I/O Expanders 0 through 7

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

Notes

I/O Expander 8

SMBus I/O Expander Bit	Type	Signal	Description
0 (I/O-0.0)[1]	I	P0MRLN	Port 0 manually operated retention latch (MRL) input
1 (I/O-0.1)	I	P1MRLN	Port 1 manually operated retention latch (MRL) input
2 (I/O-0.2)	I	P2MRLN	Port 2 manually operated retention latch (MRL) input
3 (I/O-0.3)	I	P3MRLN	Port 3 manually operated retention latch (MRL) input
4 (I/O-0.4)	I	P4MRLN	Port 4 manually operated retention latch (MRL) input
5 (I/O-0.5)	I	P5MRLN	Port 5 manually operated retention latch (MRL) input
6 (I/O-0.6)	I	P6MRLN	Port 6 manually operated retention latch (MRL) input
7 (I/O-0.7)	I	P7MRLN	Port 7 manually operated retention latch (MRL) input
8 (I/O-1.0)	I	P8MRLN	Port 8 manually operated retention latch (MRL) input
9 (I/O-1.1)	I	P9MRLN	Port 9 manually operated retention latch (MRL) input
10 (I/O-1.2)	I		Unused
11 (I/O-1.3)	I		Unused
12 (I/O-1.4)	I	P12MRLN	Port 12 manually operated retention latch (MRL) input
13 (I/O-1.5)	I	P13MRLN	Port 13 manually operated retention latch (MRL) input
14 (I/O-1.6)	I		Unused
15 (I/O-1.7)	I		Unused

Table 12.6 Pin Mapping I/O Expander 8

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

I/O Expander 9

SMBus I/O Expander Bit	Type	Signal	Description
0 (I/O-0.0)[1]	I	P0ILOCKST	Port 0 electromechanical interlock state input
1 (I/O-0.1)	I	P1ILOCKST	Port 1 electromechanical interlock state input
2 (I/O-0.2)	I	P2ILOCKST	Port 2 electromechanical interlock state input

Table 12.7 Pin Mapping I/O Expander 9 (Part 1 of 2)

Notes

SMBus I/O Expander Bit	Type	Signal	Description
3 (I/O-0.3)	I	P3ILOCKST	Port 3 electromechanical interlock state input
4 (I/O-0.4)	I	P4ILOCKST	Port 4 electromechanical interlock state input
5 (I/O-0.5)	I	P5ILOCKST	Port 5 electromechanical interlock state input
6 (I/O-0.6)	I	P6ILOCKST	Port 6 electromechanical interlock state input
7 (I/O-0.7)	I	P7ILOCKST	Port 7 electromechanical interlock state input
8 (I/O-1.0)	O	P0ILOCKP	Port 0 electromechanical interlock output
9 (I/O-1.1)	O	P1ILOCKP	Port 1 electromechanical interlock output
10 (I/O-1.2)	O	P2ILOCKP	Port 2 electromechanical interlock output
11 (I/O-1.3)	O	P3ILOCKP	Port 3 electromechanical interlock output
12 (I/O-1.4)	O	P4ILOCKP	Port 4 electromechanical interlock output
13 (I/O-1.5)	O	P5ILOCKP	Port 5 electromechanical interlock output
14 (I/O-1.6)	O	P6ILOCKP	Port 6 electromechanical interlock output
15 (I/O-1.7)	O	P7ILOCKP	Port 7 electromechanical interlock output

Table 12.7 Pin Mapping I/O Expander 9 (Part 2 of 2)

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

I/O Expander 10

SMBus I/O Expander Bit	Type	Signal	Description
0 (I/O-0.0)[1]	I	P8ILOCKST	Port 8 electromechanical interlock state input
1 (I/O-0.1)	I	P9ILOCKST	Port 9 electromechanical interlock state input
2 (I/O-0.2)	I		Unused
3 (I/O-0.3)	I		Unused
4 (I/O-0.4)	I	P12ILOCKST	Port 12 electromechanical interlock state input
5 (I/O-0.5)	I	P13ILOCKST	Port 13 electromechanical interlock state input
6 (I/O-0.6)	I		Unused
7 (I/O-0.7)	I		Unused
8 (I/O-1.0)	O	P8ILOCKP	Port 8 electromechanical interlock output
9 (I/O-1.1)	O	P9ILOCKP	Port 9 electromechanical interlock output
10 (I/O-1.2)	O		Unused
11 (I/O-1.3)	O		Unused
12 (I/O-1.4)	O	P12ILOCKP	Port 12 electromechanical interlock output
13 (I/O-1.5)	O	P13ILOCKP	Port 13 electromechanical interlock output
14 (I/O-1.6)	O		Unused
15 (I/O-1.7)	O		Unused

Table 12.8 Pin Mapping I/O Expander 10

Notes

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

I/O Expander 12

SMBus I/O Expander Bit	Type	Signal	Description
0 (I/O-0.0)[1]	O	P0LINKUPN	Port 0 link up status output
1 (I/O-0.1)	O	P1LINKUPN	Port 1 link up status output
2 (I/O-0.2)	O	P2LINKUPN	Port 2 link up status output
3 (I/O-0.3)	O	P3LINKUPN	Port 3 link up status output
4 (I/O-0.4)	O	P4LINKUPN	Port 4 link up status output
5 (I/O-0.5)	O	P5LINKUPN	Port 5 link up status output
6 (I/O-0.6)	O	P6LINKUPN	Port 6 link up status output
7 (I/O-0.7)	O	P7LINKUPN	Port 7 link up status output
8 (I/O-1.0)	O	P8LINKUPN	Port 8 link up status output
9 (I/O-1.1)	O	P9LINKUPN	Port 9 link up status output
10 (I/O-1.2)	O		Unused
11 (I/O-1.3)	O		Unused
12 (I/O-1.4)	O	P12LINKUPN	Port 12 link up status output
13 (I/O-1.5)	O	P13LINKUPN	Port 13 link up status output
14 (I/O-1.6)	O		Unused
15 (I/O-1.7)	O		Unused

Table 12.9 I/O Expander 12 - Link Up Status

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

I/O Expander 13

SMBus I/O Expander Bit	Type	Signal	Description
0 (I/O-0.0)[1]	O	P0ACTIVEN	Port 0 Link active status output
1 (I/O-0.1)	O	P1ACTIVEN	Port 1 Link active status output
2 (I/O-0.2)	O	P2ACTIVEN	Port 2 Link active status output
3 (I/O-0.3)	O	P3ACTIVEN	Port 3 Link active status output
4 (I/O-0.4)	O	P4ACTIVEN	Port 4 Link active status output
5 (I/O-0.5)	O	P5ACTIVEN	Port 5 Link active status output
6 (I/O-0.6)	O	P6ACTIVEN	Port 6 Link active status output
7 (I/O-0.7)	O	P7ACTIVEN	Port 7 Link active status output
8 (I/O-1.0)	O	P8ACTIVEN	Port 8 Link active status output
9 (I/O-1.1)	O	P9ACTIVEN	Port 9 Link active status output

Table 12.10 I/O Expander 13 - Link Activity Status (Part 1 of 2)

Notes

SMBus I/O Expander Bit	Type	Signal	Description
10 (I/O-1.2)	O		Unused
11 (I/O-1.3)	O		Unused
12 (I/O-1.4)	O	P12ACTIVEN	Port 12 Link active status output
13 (I/O-1.5)	O	P13ACTIVEN	Port 13 Link active status output
14 (I/O-1.6)	O		Unused
15 (I/O-1.7)	O		Unused

Table 12.10 I/O Expander 13 - Link Activity Status (Part 2 of 2)

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

Slave SMBus Interface

The slave SMBus interface provides PES48T12G2 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 switch fundamental reset. During the switch fundamental reset initialization sequence, the slave SMBus address is initialized. The address is specified by the SSMBADDR[2,1] signals as shown in Table 12.11.

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

Table 12.11 Slave SMBus Address

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 12.5 and described in Table 12.12.

Notes

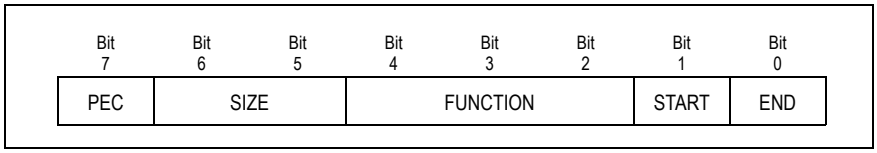


Figure 12.5 Slave SMBus Command Code Format

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 12.12 Slave SMBus Command Code Fields

The FUNCTION field in the command code indicates if the SMBus operation is a system address 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 12.13 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 12.12.
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 system address of register to access.
4	ADDRU	**Address Upper.** Upper 8-bits of the doubleword system address of register to access.
5	DATALL	**Data Lower.** Bits [7:0] of data doubleword.
6	DATALM	**Data Lower Middle.** Bits [15:8] of data doubleword.
7	DATAUM	**Data Upper Middle.** Bits [23:16] of data doubleword.
8	DATAUU	**Data Upper.** Bits [31:24] of data doubleword.

Table 12.13 CSR Register Read or Write Operation Byte Sequence

Table 12.13 indicates the sequence of data as it is presented on the slave SMBus following the byte address of the Slave SMBus interface. Dword addresses and not byte addresses must be used to access all visible software registers. ADDRL and ADDUL represent the lower 8-bit of the doubleword system address and upper 6-bit doubleword system address, respectively. For example, use ADDRU = x00 and ADDRL = 0x00 to access system address 0x00000 (port 0's Vendor/Device ID register). Use ADDRU = x00 and ADDRL = 0x01 to access system address 0x00004 (port 0's Command/Status register).

The format of the CMD field is shown in Figure 12.6 and described in Table 12.14.

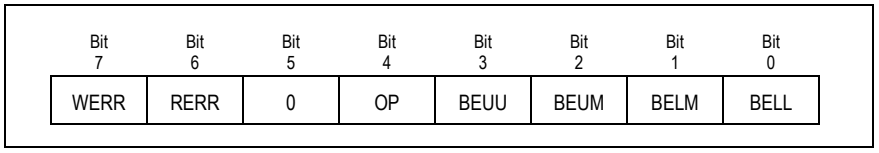


Figure 12.6 CSR Register Read or Write CMD Field Format

Notes

Bit Field	Name	Type	Description
0	BELL	Read/Write	**Byte Enable 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.** When set, the byte enable for bits [31:24] of the data word is enabled.
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[1]	Read-Only and Clear	**Read Error**. This bit is set if the last CSR read SMBus transaction was not claimed by the device. Success indicates that the transaction was claimed, not necessarily that the operation completed without error.
7	WERR[1]	Read-Only and Clear	**Write Error**. This bit is set if the last CSR write SMBus transaction was not claimed by the device. Success indicates that the transaction was claimed, not necessarily that the operation completed without error.

Table 12.14 CSR Register Read or Write CMD Field Description

1. The RERR and WERR bits are driven by the switch as status bits that indicate whether or not the switch's SMBus slave interface accepted the register read/write command (the switch accepts the access if it has the correct byte sequence). When a byte sequence refers to a register offset that is not listed or is regarded as a reserve register, the RERR and WERR bits will be set after a read or write operation is performed.

Serial EEPROM Read or Write Operation

Table 12.15 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 12.12.
1	BYTCNT	**Byte Count.** The byte count field is only transmitted for block type SMBus transactions. SMBus word and byte accesses to 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

Table 12.15 Serial EEPROM Read or Write Operation Byte Sequence (Part 1 of 2)

Notes

Byte Position	Field Name	Description
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 12.15 Serial EEPROM Read or Write Operation Byte Sequence (Part 2 of 2)

The format of the CMD field is shown in Figure 12.7 and described in Table 12.16.

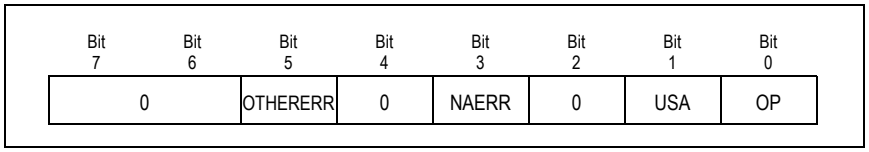


Figure 12.7 Serial EEPROM Read or Write CMD Field Format

Bit Field	Name	Type	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		Reserved field.
3	NAERR	RW1C	**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 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	Reserved		Reserved field.
5	OTHERERR	RW1C	**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 SMBUSSTS register.
7:6	Reserved	0	Reserved field. Must be zero.

Table 12.16 Serial EEPROM Read or Write CMD Field Description

Notes

Sample Slave SMBus Operation

This section illustrates sample Slave SMBus operations. Shaded items are driven by PES48T12G2's slave SMBus interface and non-shaded items are driven by an SMBus host.

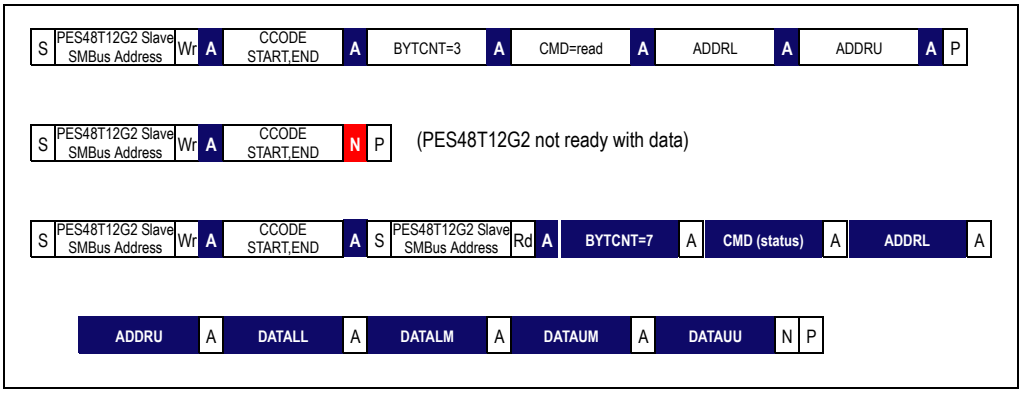


Figure 12.8 CSR Register Read Using SMBus Block Write/Read Transactions with PEC Disabled

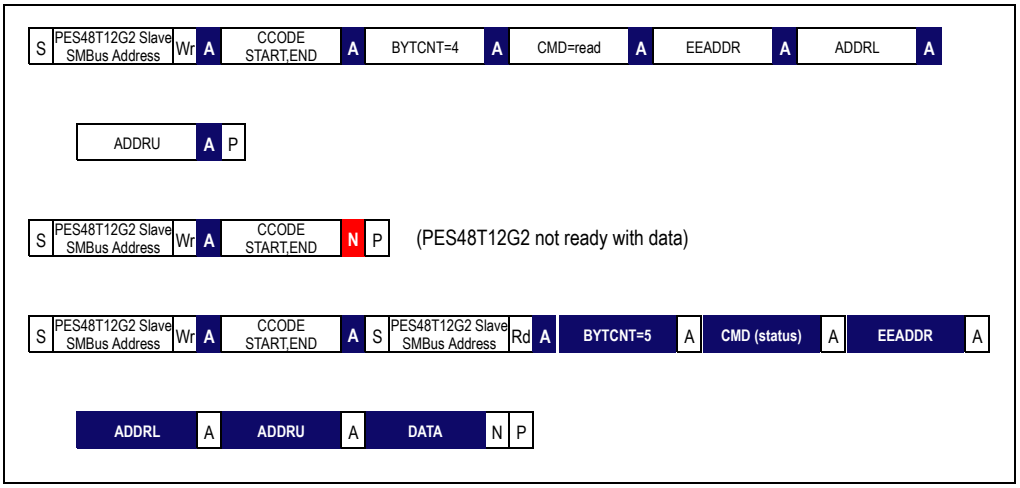


Figure 12.9 Serial EEPROM Read Using SMBus Block Write/Read Transactions with PEC Disabled

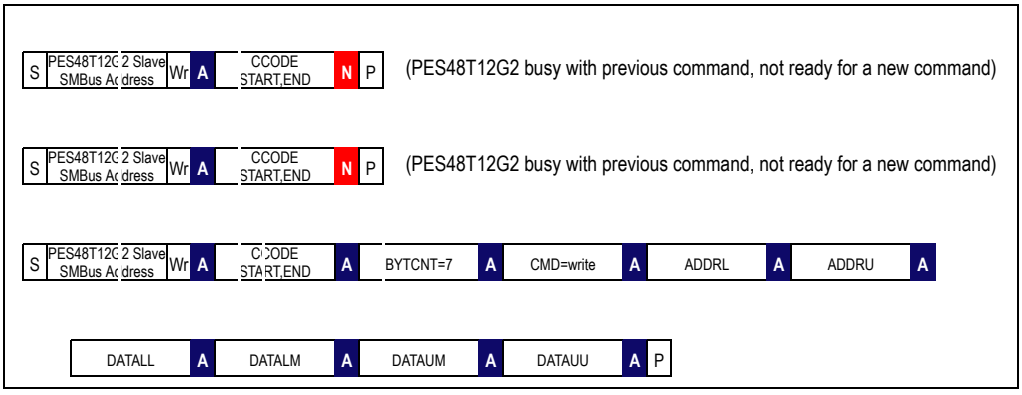


Figure 12.10 CSR Register Write Using SMBus Block Write Transactions with PEC Disabled

Notes

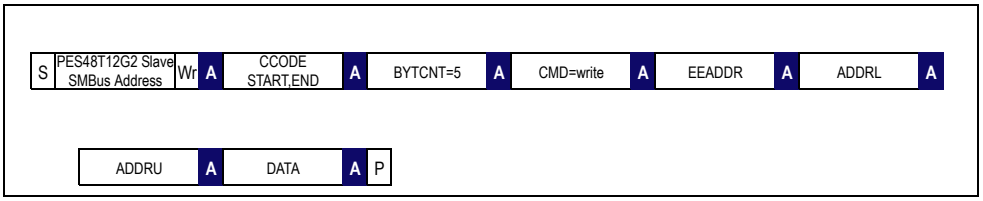


Figure 12.11 Serial EEPROM Write Using SMBus Block Write Transactions with PEC Disabled

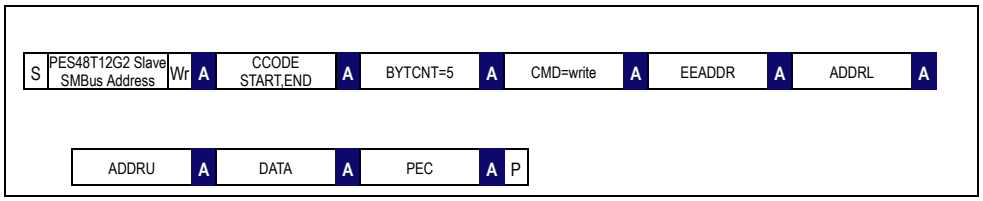


Figure 12.12 Serial EEPROM Write Using SMBus Block Write Transactions with PEC Enabled

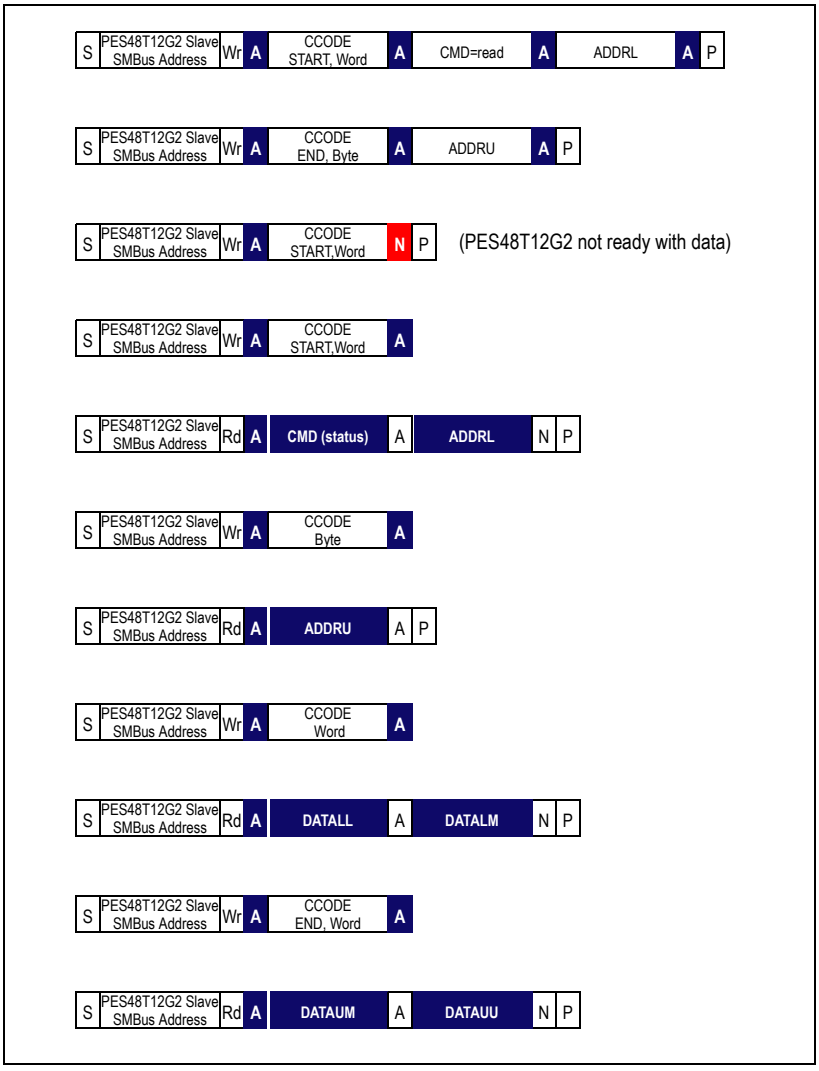


Figure 12.13 CSR Register Read Using SMBus Read and Write Transactions with PEC Disabled

Notes

Multicast

Notes

Introduction

This IDT PCIe Switch implements multicast as defined by the PCI-SIG Multicast ECN. The multicast capability enables a single TLP to be forwarded to multiple destinations. The destinations to which a multicast TLP is forwarded are referred to as a multicast group.

– A multicast group may contain zero or more destinations.

The PES48T12G2 supports up to 64 multicast groups -- this is the maximum allowed by the PCIe standard.

– The number of supported groups is advertised via the MAXGROUP field in the MCCAP register. The default value of this field corresponds to 32 groups. This field may be re-programmed during initial switch configuration (e.g., EEPROM) to advertise up to 64 groups.

A function need not be a member of a multicast group in order generate a multicast TLP that is forwarded to a multicast group. For example, any endpoint or root may generate a multicast TLP by transmitting a posted TLP with an address that maps to a multicast group.

Multicast is compatible with legacy PCIe roots and endpoints.

Addressing and Routing

Multicast addressing and routing may be partitioned into the task of determining that a TLP is a multicast TLP, routing a multicast TLP to functions (e.g., PCI-to-PCI bridges associated with egress ports), and multicast egress processing performed at each function. These tasks are described in the following sections.

Multicast TLP Determination

The determination of whether or not a TLP is a multicast TLP is made by functions that receive the TLP. All functions associated with the switch are expected to have identical multicast routing configuration. Thus, multicast TLP determination may be made using register values associated with the capability structure of any function in the switch.

– Modification of multicast routing fields requires that multicast traffic be quiesed.

The following multicast register fields must be configured to the same value in all functions in the switch. Violating this requirement results in undefined behavior on receipt of a multicast TLP. Non-multicast TLPs are not affected.

Register: MCCTL, field: NUMGROUP

Register: MCBARL, fields: INDEXPOS, MCBARL

Register: MCBARH, field: MCBARH

Unless otherwise noted, TLP processing associated with a multicast TLP is the same as that for any other TLP. For example, malformed checks are the same, poison bit processing is the same, ECRC checking and error reporting is the same, etc. When the Multicast Enable (MEN) bit is cleared in the Multicast Control (MCCTL) register, multicast is disabled and no TLP received on the link is a multicast TLP. A TLP determined not to be a multicast TLP is routed using traditional unicast PCIe routing rules. Thus, unroutable “multicast TLPs” are handled in the same manner as any other unroutable TLP.

Notes

Only posted memory write TLPs and address routed message TLPs can be multicast TLPs. The primary determinant of whether or not a memory write or address routed message TLP is a multicast TLP is its address and the address associated with multicast address regions. A multicast address region may overlap a non-multicast address region.

– Multicast TLPs that target a multicast address region are routed to all multicast group members while other TLPs, such as non-posted reads, may be routed to only one, possibly different, destination.

Multicast TLPs are posted TLPs and have the same ordering requirements as other posted TLPs. There are no new multicast TLP ordering rules.

The maximum number of multicast groups supported in an implementation is contained in the Max Multicast Groups (MAXGROUP) field in the Multicast Capability (MCCAP) register. The number of multicast groups that are actually enabled is determined by the value in the Number of Multicast Groups (NUMGROUP) field in the Multicast Control (MCCTL) register.

As illustrated in Figure 13.1, multicast TLP group membership is determined by address. Associated with each multicast group is an address region. Posted memory write and address routed message TLPs whose address is equal to that associated with a multicast group when the Multicast Enable (MEN) bit is set are defined to be multicast TLPs associated with that group.

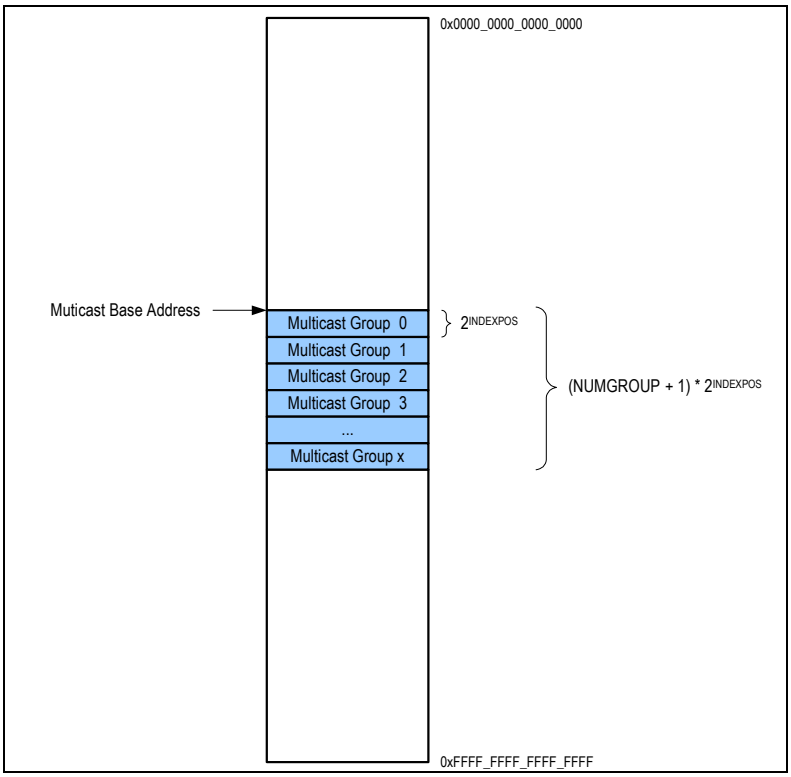


Figure 13.1 Multicast Group Address Ranges

The multicast address region associated with a TLP is determined as follows. Multicast group address regions are laid out contiguously in memory from low to high starting with multicast group zero. The number of regions is determined by the value of the NUMGROUP field in the MCCTL register.

– There are no address regions allocated for multicast group numbers that are greater than that enabled by the NUMGROUP field in the MCCTL register.

The size of each multicast group address region is determined by the value of the Index Position (INDEXPOS) field in the Multicast Base Address Low (MCBARL) register. The size of each multicast group region is equal to $2^{INDEXPOS}$.

Notes

The starting address of the region associated multicast group zero is equal to the multicast base address defined by the Multicast Base Address Low (MCBARL) field in the MCBARL register and the Multicast Base Address High (MCBARH) field in the Multicast Base Address High (MCBARH) register.

– The multicast base address is a 64-bit quantity and may start anywhere in memory.

In general, the starting address of the region associated with multicast group *n* is equal to the multicast base address plus the quantity ($n * 2^{INDEXPOS}$). Since bits in the multicast base address that correspond to the multicast group number or that are less than the multicast index position (i.e., INDEXPOS) must be zero, the multicast group ID associated with a TLP may be determined as shown in Figure 13.2.

For the purpose of multicast TLP determination, address bits in the TLP address less than the multicast index position and bits associated with the multicast group ID are zeroed. This address is then compared to the multicast base address. If the two are not equal, then the TLP is not a multicast TLP.

If the two addresses are equal, then the multicast group ID is extracted from the address bits that correspond to enabled multicast groups. The number of bits is equal to $\lceil \log_2(NUMGROUP + 1) \rceil$ and start with the address bit corresponding to the value of INDEXPOS. If the multicast group ID is greater than the number of enabled multicast groups (i.e., NUMGROUP + 1), then the TLP is not a multicast TLP. Otherwise, the TLP is a multicast TLP.

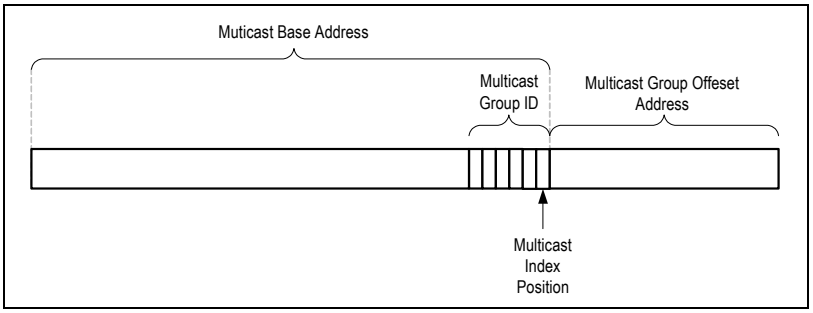


Figure 13.2 Multicast Group Address Region Determination

Once a TLP has been determined to be a multicast TLP and the multicast group ID has been determined, the following error checks are performed. If the multicast TLP fails the source validation ACS check, then it is handled as specified by ACS. No other ACS checks are performed on multicast TLPs. Associated with each multicast group is a "block all" bit. If the block all bit corresponding to the multicast group ID associated with a received multicast TLP is set in the ingress port, then the multicast TLP is treated as a blocked multicast TLP.

– The "block all" bits are contained in the ingress port's Multicast Block All (MCBLKALL) fields of the Multicast Block All Low (MCBLKALLL) and Multicast Block All High (MCBLKALLH) registers.

Associated with each multicast group is a "block untranslated" bit. If the "block untranslated" bit corresponding to the multicast group ID associated with a received multicast TLP is set in the ingress port and the TLP is untranslated, as determined by the Address Type (AT) field in the TLP header, then the multicast TLP is treated as a blocked multicast TLP.

– The "block untranslated" bits are contained in the ingress port's Multicast Block Untranslated (MCBLKUT) fields of the Multicast Block Untranslated Low (MCBLKUTL) and Multicast Block Untranslated High (MCBLKUTH) registers.

A blocked multicast TLP is treated in the following manner.

– The TLP is dropped and flow control credits are returned.
– The error is reported to AER as a MC Blocked TLP error and the header is logged.
– In an upstream port, the Signaled Target Abort (STAS) bit is set in the PCI Status (PCISTS) register.
– In a downstream port, the Signaled Target Abort (STAS) bit is set in the Secondary Status (SECSTS) register.

Notes

Note that the “block all” and “block untranslated” functions are performed at the ingress port on which the multicast TLP was received. A received multicast TLP without errors is forwarded to egress ports as described in the next section.

Multicast TLP Routing

A multicast TLP received without error by a function is forwarded as described in this section. Traditional unicast routing rules do not apply to multicast TLPs. Unlike unicast routing rules that depend on whether the TLP was received on the primary or secondary side of a PCI-to-PCI bridge and are thus different for upstream and downstream ports, multicast TLP routing is symmetric. The same multicast routing rules apply to all functions. A multicast TLP received by a function is forwarded to the virtual PCI bus in the switch. All functions connected to the virtual PCI bus examine the multicast group ID associated with the multicast TLP and perform the following actions:

- The function on which the multicast TLP was received ignores the multicast TLP. If the multicast enable (MCEN) bit is cleared, then the function ignores the multicast TLP. Associated each function is a multicast receive vector that contains a bit corresponding to each multicast group. If the MCEN bit is set and the bit corresponding to the multicast group ID associated with the multicast TLP is set in the multicast receive vector, then the multicast TLP is accepted by the function.
  - The multicast receive vector is contained in the Multicast Receive (MCRCV) fields of the Multicast Receive Low (MCRCVL) and Multicast Receive High (MCRCVH) registers.
- A function that accepts a multicast TLP forwards the TLP after multicast egress processing is performed.
  - For a PCI-to-PCI bridge, forwarding a TLP means transmitting the TLP on the link associated with the switch port corresponding to the PCI-to-PCI bridge.
- If no function accepts a multicast TLP, then the TLP is silently discarded. This is not an error.

Note: This section described multicast TLP routing from a functional perspective to aid in understanding. This functional definition does not represent the actual multicast routing implementation in the switch.

Multicast Egress Processing

Each switch function implements multicast overlay processing. When the Overlay Size (OVRSIZE) field in the Multicast Overlay Base Address Low (MCOVRBARL) register is set to zero, multicast overlay processing is disabled and multicast TLPs are forwarded without modification. When the OVRSIZE field is non-zero, multicast overlay processing is performed on all multicast TLPs accepted by the function as described below.

Address bits in the accepted multicast TLP with bit positions greater than or equal to OVRSIZE are replaced by the corresponding address bits in the multicast overlay base address.

- The multicast overlay base address is contained in the Multicast Overlay BAR Low (BARLOW) field in the Multicast Overlay Base Address Low (MCOVRBARL) register and the Multicast Overlay BAR High (MCOVRBARH) field in the Multicast Overlay Base Address High (MCOVRBARH) register.

Address bits less than OVRSIZE are not modified. As a result of multicast overlay processing, a multicast TLP with an original address above 4 GB may be translated into a multicast TLP with address below 4 GB, and vice-versa. Thus, address translation may change the size of a multicast TLP header (e.g., from 4 DWords to 3 Dwords).

Multicast overlay processing is performed independently on all functions. Therefore, it is possible to enable this capability in some functions and not others. The overlay base address associated with different functions will likely have different values. This capability is available on both upstream and downstream switch ports and operates in the same manner regardless of port type.

Notes

A side-effect of modifying the address due to multicast overlay processing is that the ECRC associated with the original TLP may not be correct for the new modified TLP. Therefore, functions perform the following ECRC processing:

- If multicast overlay processing is disabled, then no ECRC processing is performed as part of multicast egress processing. The function continues to check and report ECRC errors for received TLPs and this behavior is unaffected in any way by multicast.
- If a multicast TLP does not contain an ECRC, then no ECRC processing is performed as part of multicast egress processing. If a multicast TLP contains an ECRC and multicast overlay processing is enabled, then the following actions are performed.
- The ECRC of the original multicast TLP is checked while simultaneously the ECRC for the new modified TLP is computed or "regenerated." This is implemented in the same pipeline stage such that there is virtually no possibility of silent data corruption (e.g., a TLP bit flip that does not result in a computed ECRC error in the original or regenerated ECRC).
- If no error is detected in the ECRC associated with the original TLP, then the modified TLP is forwarded with the regenerated ECRC. If an error is detected in the ECRC associated with the original TLP, then the modified TLP is forwarded with inverted regenerated ECRC (i.e., the computed ECRC of the modified TLP is inverted). No errors are reported due to multicast egress processing.

Notes

Register Organization

Notes

Introduction

All software visible registers in the PES48T12G2 are contained in a 256 KB global address space. The address of a register in this address range is referred to as the system address of the register.

– The system address is 19-bits in size.

System addresses are used for serial EEPROM initialization and slave SMBus register access.

– The global address range is partitioned into regions as shown in Table 14.1.
– There is a 4 KB region for PCI-to-PCI bridge registers associated with each port.
– There is one 8 KB region for switch configuration and status registers.

Base Address	Address Range
0x00000	Port 0 PCI-to-PCI Bridge Registers
0x02000	Port 1 PCI-to-PCI Bridge Registers
0x04000	Port 2 PCI-to-PCI Bridge Registers
0x06000	Port 3 PCI-to-PCI Bridge Registers
0x08000	Port 4 PCI-to-PCI Bridge Registers
0x0A000	Port 5 PCI-to-PCI Bridge Registers
0x0C000	Port 6 PCI-to-PCI Bridge Registers
0x0E000	Port 7 PCI-to-PCI Bridge Registers
0x10000 - 0x3DFFF	Reserved
0x3E000 - 0x3FFFF	Switch Configuration and Status Registers
Other	Reserved

Table 14.1 Global Address Space Organization

PCI-to-PCI bridge registers correspond to the configuration registers associated with an upstream or downstream switch port.

– These registers are accessible as function 0 to PCI configuration requests when the port is configured to operate in upstream switch port or downstream switch port.
– The PCI configuration or extended configuration space address of a PCI-to-PCI bridge register is equal to the offset address of the register within the global address region. The offset address map for PCI-to-PCI bridge registers is defined in section PCI-to-PCI Bridge Registers on page 14-2.

The switch configuration and status register region contains registers that control general operation of the switch or that are proprietary in nature.

– The offset address switch configuration and status registers is defined in section Switch Configuration and Status Registers on page 14-12.

Notes

The entire PES48T12G2 global address space may be accessed using PCI configuration requests from any PES48T12G2 PCI function.

- Located in each PCI function is a Global Address Space Access Address (GASAADDR) and Global Address Space Access Data (GASADATA) register.
- The DWord address of the global address space register to be accessed is written to the Address (ADDR) field in the GASAADDR register. When a read is performed to the Data (DATA) field in the GASADATA register, the value of the corresponding global address space register selected by the ADDR field is returned. When a write is performed to the DATA field, the value of the corresponding global address space register selected by the ADDR field is updated with the value written.
- Any register in the entire device may be accessed using the GASAADDR and GASADATA registers.

Access to the global address space registers may be done via PCI configuration accesses, via SMBus, or via serial EEPROM. SMBus or serial EEPROM accesses are not affected by the global address space protection register.

Partial-Byte Access to Word and DWord Registers

Configuration registers in the switch have different sizes (e.g., Byte, Word, DWord). Registers should be accessed with byte-enables that correspond to their native size or a size of one DWord. For example, a Byte register should be read or written with only one byte enable set, or with all four byte enables set. A DWord register should be read or written with all the byte-enables set.

Register Side-Effects

There are software visible configuration registers that have a side-effect action when written and this side-effect action may affect the ability of the switch to respond with a completion. A configuration write to such a register always returns a completion to the link partner before the side-effect action is performed.

This is implemented by delaying the side-effect action by 1ms following generation of the completion. If the completion is not accepted by the link partner in this time interval, then the completion will be lost.

The following registers, when written, have a side-effect action delay.

- PCI-to-PCI Bridge Function Registers
  - PHYLSTATE0.FLRET
- Switch Configuration and Status Registers
  - SWPORTxCTL.MODE

Limitations

Due to a switch design limitation, a PCI Express configuration request must never set the FLRET bit in the PHYLSTATE0 register of a partition's upstream port that is crosslinked. This operation is not supported and will result in the completion associated with that request to be lost.

Address Maps

This section describes the address maps for regions of the global address space outlined in Table 14.1. Reserved address ranges are outlined in Table 14.1. Reading from a reserved address range returns and undefined value. Writes to a reserved address range complete successfully and have an undefined behavior.

PCI-to-PCI Bridge Registers

This section outlines the configuration space associated with PCI-to-PCI bridges. These registers are accessible as function 0 when the port is configured in the following modes.

- Upstream switch port
- Downstream switch port

Notes

These registers are always accessible regardless of the port mode using global address space access registers (i.e., GASAADDR and GASADATA), SMBus, or serial EEPROM. Access to the Extended Configuration Space Address Registers (ECFGADDR and ECFGDATA) located in the PCI-to-PCI Bridge function is not allowed via the GASAADDR and GASADATA registers.

Figure 14.1 shows the organization of the configuration space.

- Registers with offsets between 0x000 and 0x0FF are associated with PCI configuration space.
- Registers with offsets between 0x100 and 0x3FF are associated with PCI Express extended configuration space
- Registers with offsets between 0x400 and 0xFFF are associated with PCI Express extended configuration space but are used to hold IDT proprietary port specific registers.

In order to facilitate access to the PCI Express extended configuration space by legacy PCI software, the PCI-to-PCI bridge configuration space contains the Extended Configuration Space Access Address and Data registers (ECFGADDR and ECFGDATA). Refer to the definition of these registers for further details.

Offset addresses for PCI-to-PCI bridge registers are listed in Table 14.4 and register definitions are provided in Chapter 15. Registers in this address range are referenced as PxP2P_REGNAME where x represents the PES48T12G2 port number and REGNAME represents the register name in Table 14.4.

Reading from a reserved address or region returns a value of zero. Writes to a reserved address complete successfully but modify no data and have no other effect.

The port operating mode (e.g., upstream switch port or downstream switch port) determines the presence of configuration registers within the PCI-to-PCI bridge function's configuration space. For example, the slot capability, slot control, and slot status registers are only present in the configuration space of a bridge function in a downstream port.

Table 14.4 has two columns indicating the presence of each register within the PCI-to-PCI bridge function's space depending on the port operating mode. Column 'US' refers to a port in upstream switch mode, and column 'DS' refers to a port in downstream switch mode. A mark of 'N' in the column indicates that the corresponding register is not present in the configuration space. Otherwise, the register is present in the configuration space. Registers that are not present in the configuration space are considered reserved when the port operates in the corresponding operating mode.

Capability Structures

The PCI-to-PCI bridge function contains a number of PCI capability structures and PCI Express extended capability structures. Following a fundamental reset, some of these capabilities are linked by default and visible to software while others need to be explicitly linked to be made visible (e.g., via firmware, serial EEPROM, or SMBus). Table 14.2 lists PCI Express capability structures and their default linkage (i.e., the default value of each capability's Next Pointer (NXTPTR) field). The default linkage is dependent on a port's operating mode. A value of 0x0 in the NXTPTR field terminates the list.

PCI Capability Structure		Default Value of Next Pointer Field (NXTPTR)	
Name	Offset in Configuration Space	US	DS
PCI Express Capability (PCIECAP)	0x040	0x0C0	0x0C0

Table 14.2 Default PCI Capability List Linkage (Part 1 of 2)

Notes

PCI Capability Structure		Default Value of Next Pointer Field (NXTPTR)	
Name	Offset in Configuration Space	US	DS
PCI Power Management Capability (PMCAP)	0x0C0	0x0	0x0D0
Message Signaled Interrupt Capability (MSICAP)	0x0D0	0x0	0x0
Subsystem ID and Subsystem Vendor Capability (SSIDSSVIDCAP)	0x0F0	0x0	0x0

Table 14.2 Default PCI Capability List Linkage (Part 2 of 2)

Table 14.3 lists PCI Express extended capability structures and their default linkage (i.e., the default value of each capability's Next Pointer (NXTPTR) field). The default linkage is dependent on a port's operating mode. A value of 0x0 in the NXTPTR field terminates the list.

PCI Capability Structure		Default Value of Next Pointer field (NXTPTR)	
Name	Offset in Configuration Space	US	DS
Advanced Error Reporting Capability (AERCAP)	0x100	0x200	0x200
Device Serial Number Extended Capability (SNUMCAP)	0x180	0x0	0x0
PCI Express Virtual Channel Capability (PCIEVCECAP)	0x200	0x330	0x320
Power Budgeting Extended Capability (PWRBCAP)	0x280	0x0	0x0
ACS Extended Capability (ACSECAPH)	0x320	0x330	0x330
Multicast Extended Capability (MCCAPH)	0x330	0x0	0x0

Table 14.3 Default PCI Express Capability List Linkage

Notes

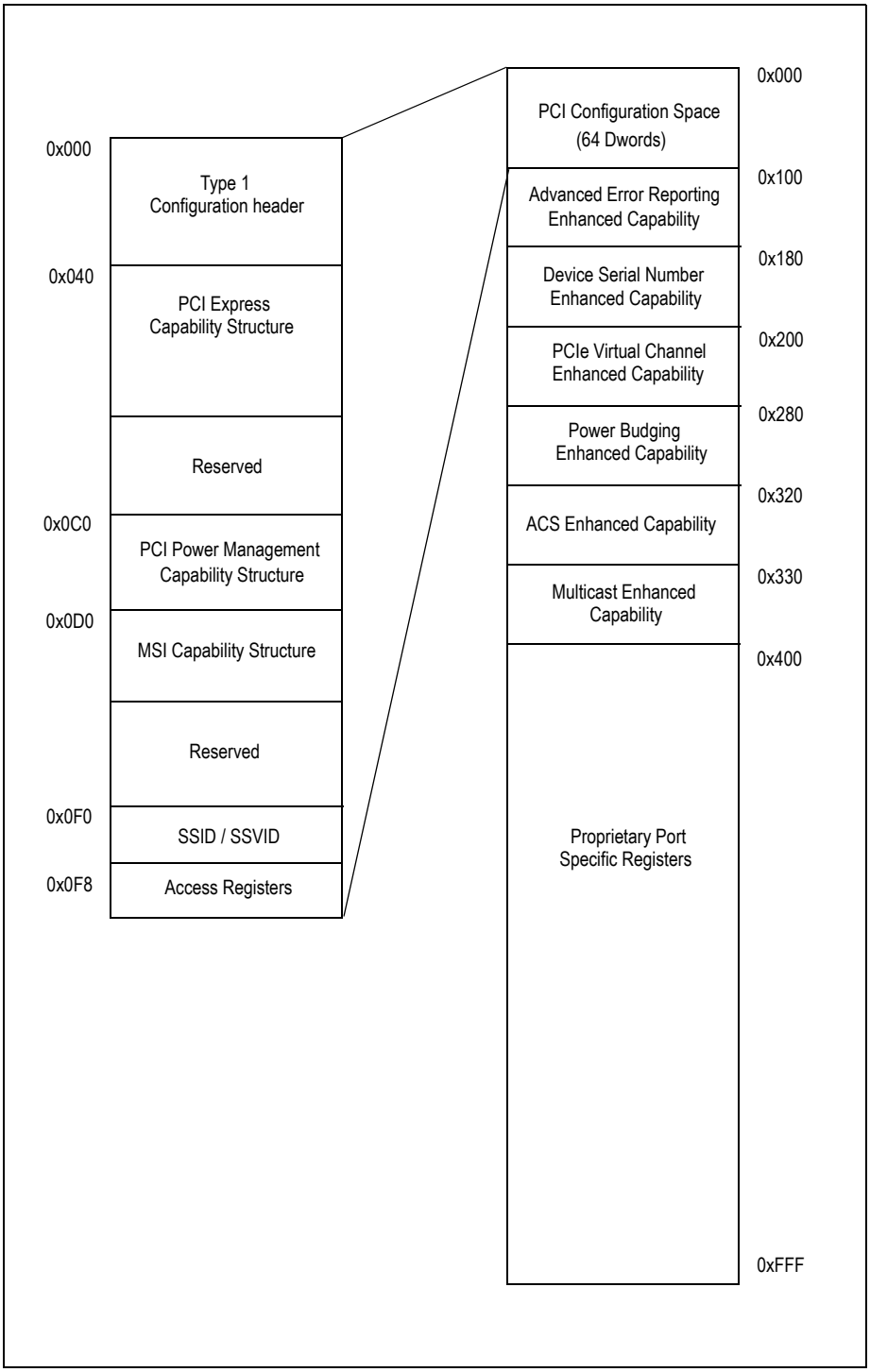


Figure 14.1 PCI-to-PCI Bridge Configuration Space Organization

Cfg. Offset	Size	Register Mnemonic	Register Definition	US	DS
0x000	Word	VID	VID - Vendor Identification Register (0x000) on page 15-1		
0x002	Word	DID	DID - Device Identification Register (0x002) on page 15-1		
0x004	Word	PCICMD	PCICMD - PCI Command Register (0x004) on page 15-1		
0x006	Word	PCISTS	PCISTS - PCI Status Register (0x006) on page 15-2		
0x008	Byte	RID	RID - Revision Identification Register (0x008) on page 15-3		
0x009	3 Bytes	CCODE	CCODE - Class Code Register (0x009) on page 15-3		
0x00C	Byte	CLS	CLS - Cache Line Size Register (0x00C) on page 15-4		
0x00D	Byte	PLTIMER	PLTIMER - Primary Latency Timer (0x00D) on page 15-4		
0x00E	Byte	HDR	HDR - Header Type Register (0x00E) on page 15-4		
0x00F	Byte	BIST	BIST - Built-in Self Test Register (0x00F) on page 15-4		
0x010	DWord	BAR0	BAR0 - Base Address Register 0 (0x010) on page 15-4		
0x014	DWord	BAR1	BAR1 - Base Address Register 1 (0x014) on page 15-4		
0x018	Byte	PBUSN	PBUSN - Primary Bus Number Register (0x018) on page 15-5		
0x019	Byte	SBUSN	SBUSN - Secondary Bus Number Register (0x019) on page 15-5		
0x01A	Byte	SUBUSN	SUBUSN - Subordinate Bus Number Register (0x01A) on page 15-5		
0x01B	Byte	SLTIMER	SLTIMER - Secondary Latency Timer Register (0x01B) on page 15-5		
0x01C	Byte	IOBASE	IOBASE - I/O Base Register (0x01C) on page 15-5		
0x01D	Byte	IOLIMIT	IOLIMIT - I/O Limit Register (0x01D) on page 15-6		
0x01E	Word	SECSTS	SECSTS - Secondary Status Register (0x01E) on page 15-6		
0x020	Word	MBASE	MBASE - Memory Base Register (0x020) on page 15-7		
0x022	Word	MLIMIT	MLIMIT - Memory Limit Register (0x022) on page 15-7		
0x024	Word	PMBASE	PMBASE - Prefetchable Memory Base Register (0x024) on page 15-7		
0x026	Word	PMLIMIT	PMLIMIT - Prefetchable Memory Limit Register (0x026) on page 15-7		
0x028	DWord	PMBASEU	PMBASEU - Prefetchable Memory Base Upper Register (0x028) on page 15-8		
0x02C	DWord	PMLIMITU	PMLIMITU - Prefetchable Memory Limit Upper Register (0x02C) on page 15-8		
0x030	Word	IOBASEU	IOBASEU - I/O Base Upper Register (0x030) on page 15-8		
0x032	Word	IOLIMITU	IOLIMITU - I/O Limit Upper Register (0x032) on page 15-8		
0x034	Byte	CAPPTR	CAPPTR - Capabilities Pointer Register (0x034) on page 15-9		
0x038	DWord	EROMBASE	EROMBASE - Expansion ROM Base Address Register (0x038) on page 15-9		
0x03C	Byte	INTRLINE	INTRLINE - Interrupt Line Register (0x03C) on page 15-9		
0x03D	Byte	INTRPIN	INTRPIN - Interrupt PIN Register (0x03D) on page 15-9		
0x03E	Word	BCTL	BCTL - Bridge Control Register (0x03E) on page 15-10		
0x040	DWord	PCIECAP	PCIECAP - PCI Express Capability (0x040) on page 15-11		

Table 14.4 PCI-to-PCI Bridge Configuration Space Registers (Part 1 of 4)

Cfg. Offset	Size	Register Mnemonic	Register Definition	US	DS
0x044	DWord	PCIEDCAP	PCIEDCAP - PCI Express Device Capabilities (0x044) on page 15-11		
0x048	Word	PCIEDCTL	PCIEDCTL - PCI Express Device Control (0x048) on page 15-13		
0x04A	Word	PCIEDSTS	PCIEDSTS - PCI Express Device Status (0x04A) on page 15-14		
0x04C	DWord	PCIELCAP	PCIELCAP - PCI Express Link Capabilities (0x04C) on page 15-14		
0x050	Word	PCIELCTL	PCIELCTL - PCI Express Link Control (0x050) on page 15-16		
0x052	Word	PCIELSTS	PCIELSTS - PCI Express Link Status (0x052) on page 15-17		
0x054	DWord	PCIESCAP	PCIESCAP - PCI Express Slot Capabilities (0x054) on page 15-19	N	
0x058	Word	PCIESCTL	PCIESCTL - PCI Express Slot Control (0x058) on page 15-20	N	
0x05A	Word	PCIESSTS	PCIESSTS - PCI Express Slot Status (0x05A) on page 15-22	N	
0x064	DWord	PCIEDCAP2	PCIEDCAP2 - PCI Express Device Capabilities 2 (0x064) on page 15-24		
0x068	Word	PCIEDCTL2	PCIEDCTL2 - PCI Express Device Control 2 (0x068) on page 15-24		
0x06A	Word	PCIEDSTS2	PCIEDSTS2 - PCI Express Device Status 2 (0x06A) on page 15-24		
0x06C	DWord	PCIELCAP2	PCIELCAP2 - PCI Express Link Capabilities 2 (0x06C) on page 15-24		
0x070	Word	PCIELCTL2	PCIELCTL2 - PCI Express Link Control 2 (0x070) on page 15-25		
0x072	Word	PCIELSTS2	PCIELSTS2 - PCI Express Link Status 2 (0x072) on page 15-27		
0x074	DWord	PCIESCAP2	PCIESCAP2 - PCI Express Slot Capabilities 2 (0x074) on page 15-27	N	
0x078	Word	PCIESCTL2	PCIESCTL2 - PCI Express Slot Control 2 (0x078) on page 15-27	N	
0x07A	Word	PCIESSTS2	PCIESSTS2 - PCI Express Slot Status 2 (0x07A) on page 15-27	N	
0x0C0	DWord	PMCAP	PMCAP - PCI Power Management Capabilities (0x0C0) on page 15-27		
0x0C4	DWord	PMCSR	PMCSR - PCI Power Management Control and Status (0x0C4) on page 15-28		
0x0D0	DWord	MSICAP	MSICAP - Message Signaled Interrupt Capability and Control (0x0D0) on page 15-29	N	
0x0D4	DWord	MSIADDR	MSIADDR - Message Signaled Interrupt Address (0x0D4) on page 15-30	N	
0x0D8	DWord	MSIUADDR	MSIUADDR - Message Signaled Interrupt Upper Address (0x0D8) on page 15-30	N	
0x0DC	DWord	MSIMDATA	MSIMDATA - Message Signaled Interrupt Message Data (0x0DC) on page 15-30	N	
0x0F0	Dword	SSIDSSVIDCAP	SSIDSSVIDCAP - Subsystem ID and Subsystem Vendor ID Capability (0x0F0) on page 15-31		
0x0F4	Dword	SSIDSSVID	SSIDSSVID - Subsystem ID and Subsystem Vendor ID (0x0F4) on page 15-31		
0x0F8	Dword	ECFGADDR	ECFGADDR - Extended Configuration Space Access Address (0x0F8) on page 15-31		
0x0FC	Dword	ECFGDATA	ECFGDATA - Extended Configuration Space Access Data (0x0FC) on page 15-32		
0x100	Dword	AERCAP	AERCAP - AER Capabilities (0x100) on page 15-32		
0x104	Dword	AERUES	AERUES - AER Uncorrectable Error Status (0x104) on page 15-32		
0x108	Dword	AERUEM	AERUEM - AER Uncorrectable Error Mask (0x108) on page 15-34		

Table 14.4 PCI-to-PCI Bridge Configuration Space Registers (Part 2 of 4)

Cfg. Offset	Size	Register Mnemonic	Register Definition	US	DS
0x10C	Dword	AERUESV	AERUESV - AER Uncorrectable Error Severity (0x10C) on page 15-36		
0x110	Dword	AERCES	AERCES - AER Correctable Error Status (0x110) on page 15-37		
0x114	Dword	AERCEM	AERCEM - AER Correctable Error Mask (0x114) on page 15-38		
0x118	Dword	AERCTL	AERCTL - AER Control (0x118) on page 15-40		
0x11C	Dword	AERHL1DW	AERHL1DW - AER Header Log 1st Doubleword (0x11C) on page 15-40		
0x120	Dword	AERHL2DW	AERHL2DW - AER Header Log 2nd Doubleword (0x120) on page 15-40		
0x124	Dword	AERHL3DW	AERHL3DW - AER Header Log 3rd Doubleword (0x124) on page 15-40		
0x128	Dword	AERHL4DW	AERHL4DW - AER Header Log 4th Doubleword (0x128) on page 15-41		
0x180	Dword	SNUMCAP	SNUMCAP - Serial Number Capabilities (0x180) on page 15-41		
0x184	Dword	SNUMLDW	SNUMLDW - Serial Number Lower Doubleword (0x184) on page 15-41		
0x188	Dword	SNUMUDW	SNUMUDW - Serial Number Upper Doubleword (0x188) on page 15-41		
0x200	DWord	PCIEVCECAP	PCIEVCECAP - PCI Express VC Enhanced Capability Header (0x200) on page 15-42		
0x204	DWord	PVCCAP1	PVCCAP1- Port VC Capability 1 (0x204) on page 15-42		
0x208	DWord	PVCCAP2	PVCCAP2- Port VC Capability 2 (0x208) on page 15-42		
0x20C	Word	PVCCTL	PVCCTL - Port VC Control (0x20C) on page 15-43		
0x20E	Word	PVCSTS	PVCSTS - Port VC Status (0x20E) on page 15-43		
0x210	DWord	VCR0CAP	VCR0CAP- VC Resource 0 Capability (0x210) on page 15-43		
0x214	DWord	VCR0CTL	VCR0CTL- VC Resource 0 Control (0x214) on page 15-44		
0x218	DWord	VCR0STS	VCR0STS - VC Resource 0 Status (0x218) on page 15-44		
0x230	DWord	VCTBL0	VCR0TBL0 - VC Resource 0 Port Arbitration Table Entry 0 (0x240) on page 15-45		
0x240	DWord	VCR0TBL0	VCR0TBL0 - VC Resource 0 Port Arbitration Table Entry 0 (0x240) on page 15-45		
0x244	DWord	VCR0TBL1	VCR0TBL1 - VC Resource 0 Port Arbitration Table Entry 1 (0x244) on page 15-46		
0x248	DWord	VCR0TBL2	VCR0TBL2 - VC Resource 0 Port Arbitration Table Entry 2 (0x248) on page 15-46		
0x24C	DWord	VCR0TBL3	VCR0TBL3 - VC Resource 0 Port Arbitration Table Entry 3 (0x24C) on page 15-47		
0x280	Dword	PWRBCAP	PWRBCAP - Power Budgeting Capabilities (0x280) on page 15-47		
0x284	Dword	PWRBDSEL	PWRBDSEL - Power Budgeting Data Select (0x284) on page 15-48		
0x288	Dword	PWRBD	PWRBD - Power Budgeting Data (0x288) on page 15-48		
0x28C	Dword	PWRBPBC	PWRBPBC - Power Budgeting Power Budget Capability (0x28C) on page 15-48		
0x300	Dword	PWRBDV0	PWRBDV[7:0] - Power Budgeting Data Value [7:0] (0x300 - 0x31C) on page 15-48		
0x304	Dword	PWRBDV1	PWRBDV[7:0] - Power Budgeting Data Value [7:0] (0x300 - 0x31C) on page 15-48		

Table 14.4 PCI-to-PCI Bridge Configuration Space Registers (Part 3 of 4)

Cfg. Offset	Size	Register Mnemonic	Register Definition	US	DS
0x308	Dword	PWRBDV2	PWRBDV[7:0] - Power Budgeting Data Value [7:0] (0x300 - 0x31C) on page 15-48		
0x30C	Dword	PWRBDV3	PWRBDV[7:0] - Power Budgeting Data Value [7:0] (0x300 - 0x31C) on page 15-48		
0x310	Dword	PWRBDV4	PWRBDV[7:0] - Power Budgeting Data Value [7:0] (0x300 - 0x31C) on page 15-48		
0x314	Dword	PWRBDV5	PWRBDV[7:0] - Power Budgeting Data Value [7:0] (0x300 - 0x31C) on page 15-48		
0x318	Dword	PWRBDV6	PWRBDV[7:0] - Power Budgeting Data Value [7:0] (0x300 - 0x31C) on page 15-48		
0x31C	Dword	PWRBDV7	PWRBDV[7:0] - Power Budgeting Data Value [7:0] (0x300 - 0x31C) on page 15-48		
0x320	Dword	ACSECAPH	ACSECAPH - ACS Extended Capability Header (0x320) on page 15-49	N	
0x324	Word	ACSCAP	ACSCAP - ACS Capability Register (0x324) on page 15-49	N	
0x326	Word	ACSCTL	ACSCTL - ACS Control Register (0x326) on page 15-50	N	
0x328	Word	ACSECV	ACSECV - ACS Egress Control Vector (0x328) on page 15-51	N	
0x330	Dword	MCCAPH	MCCAPH - Multicast Enhanced Capability Header (0x330) on page 15-52		
0x334	Word	MCCAP	MCCAP - Multicast Capability (0x334) on page 15-52		
0x336	Word	MCCTL	MCCTL- Multicast Control (0x336) on page 15-52		
0x338	Dword	MCBARL	MCBARL- Multicast Base Address Low (0x338) on page 15-53		
0x33C	Dword	MCBARH	MCBARH- Multicast Base Address High (0x33C) on page 15-53		
0x340	Dword	MCRCVL	MCRCVL- Multicast Receive Low (0x340) on page 15-53		
0x344	Dword	MCRCVH	MCRCVH- Multicast Receive High (0x344) on page 15-54		
0x348	Dword	MCBLKALLL	MCBLKALLL- Multicast Block All Low (0x348) on page 15-54		
0x34C	Dword	MCBLKALLH	MCBLKALLH- Multicast Block All High (0x34C) on page 15-54		
0x350	Dword	MCBLKUTL	MCBLKUTL- Multicast Block Untranslated Low (0x350) on page 15-55		
0x354	Dword	MCBLKUTH	MCBLKUTH - Multicast Block Untranslated High (0x354) on page 15-55		
0x358	Dword	MCOVRBARL	MCOVRBARL- Multicast Overlay Base Address Low (0x358) on page 15-55		
0x35C	Dword	MCOVRBARH	MCOVRBARH- Multicast Overlay Base Address High (0x35C) on page 15-55		
0x360 — 0xFFF			Reserved		

Table 14.4 PCI-to-PCI Bridge Configuration Space Registers (Part 4 of 4)

Notes

IDT Proprietary Port Specific Registers

This section outlines the address range 0x400 through 0xFFF in the PCI-to-PCI bridge address space. This address range contains IDT proprietary registers that are port specific. Registers in this address range may be accessed using PCI configuration requests to the corresponding PCI-to-PCI bridge function 0 header, global address space access registers, SMBus, or serial EEPROM. Figure 14.2 shows the organization of the proprietary port specific registers. Offset addresses for proprietary port specific registers are listed in Table 14.5. Offset addresses are with respect to the corresponding PCI-to-PCI bridge function 0 base address.

Registers in this address range are referenced as Px_REGNAME where x represents the PES48T12G2 port number and REGNAME represents the register name in Table 14.5. Reading from or writing to an address not defined in Table 14.5 completes successfully but has an undefined effect.

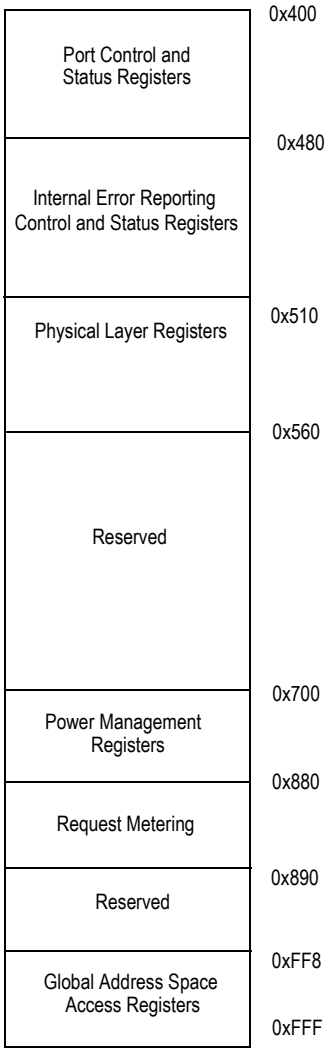


Figure 14.2 Proprietary Port Specific Register Organization

Notes

Cfg. Offset	Size	Register Mnemonic	Register Definition
0x420	Dword	PCIESCTLIV	PCIESCTLIV - PCI Express Slot Control Initial Value (0x420) on page 15-56
0x480	DWord	IERRORCTL	IERRORCTL - Internal Error Reporting Control (0x480) on page 15-58
0x484	DWord	IERRORSTS	IERRORSTS - Internal Error Reporting Status (0x484) on page 15-58
0x488	DWord	IERRORMSK	IERRORMSK - Internal Error Reporting Mask (0x488) on page 15-59
0x48C	DWord	IERRORSEV	IERRORSEV - Internal Error Reporting Severity (0x48C) on page 15-61
0x490	DWord	IERRORTST	IERRORTST - Internal Error Reporting Test (0x490) on page 15-64
0x510	Dword	SERDESCFG	SERDESCFG - SerDes Configuration (0x510) on page 15-65
0x51C	Dword	LANESTS0	LANESTS0 - Lane Status 0 (0x51C) on page 15-66
0x520	Dword	LANESTS1	LANESTS1 - Lane Status 1 (0x520) on page 15-66
0x530	Dword	PHYLCFG0	PHYLCFG0 - Phy Link Configuration 0 (0x530) on page 15-67
0x540	DWord	PHYLSTATE0	PHYLSTATE0 - Phy Link State 0 (0x540) on page 15-67
0x55C	DWord	PHYPRBS	PHYPRBS - Phy PRBS Seed (0x55C) on page 15-68
0x560 — 0x70C		Reserved	
0x710	DWord	L1ASPMRTC	L1ASPMRTC - L1 ASPM Rejection Timer Control (0x710) on page 15-68
0x880	DWord	RMCTL	RMCTL - Requester Metering Control (0x880) on page 15-69
0x88C	DWord	RMCOUNT	RMCOUNT - Requester Metering Count (0x88C) on page 15-69
0x890 — 0xF1C		Reserved	
0xFF8	Dword	GASAADDR	GASAADDR - Global Address Space Access Address (0xFF8) on page 15-70
0xFFC	Dword	GASADATA	GASADATA - Global Address Space Access Data (0xFFC) on page 15-70

Table 14.5 Proprietary Port Specific Registers

Notes

Switch Configuration and Status Registers

This section outlines switch configuration and status registers. These registers are accessible using global address space access registers (i.e., GASAADDR and GASADATA), SMBus, or serial EEPROM. Figure 14.3 shows the organization of the address space. Registers in this address range are referenced as REGNAME where REGNAME represents the register name in Table 14.6. Offset addresses for these registers can be found in Table 14.6 and register definitions are provided in Chapter 16. Reading from or writing to an address not defined in Table 14.6 completes successfully but has an undefined effect.

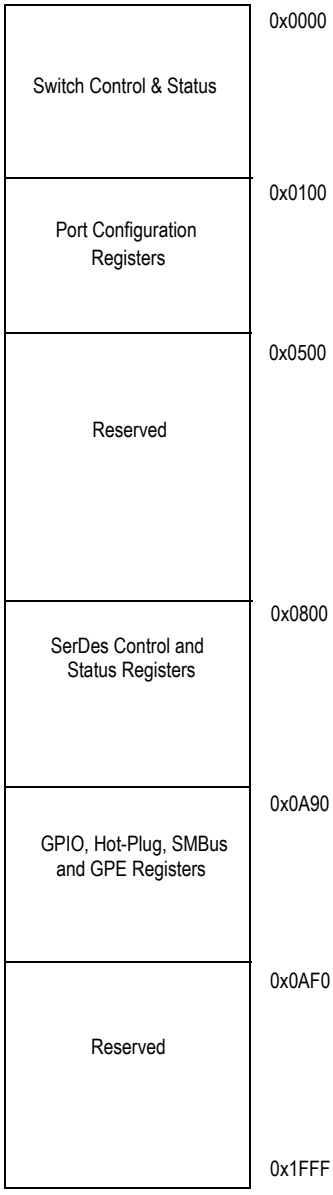


Figure 14.3 Switch Configuration and Status Space Organization

Notes

Cfg. Offset	Size	Register Mnemonic	Register Definition
0x0000	DWord	SWCTL	SWCTL - Switch Control (0x0000) on page 16-1
0x0004	DWord	BCVSTS	BCVSTS - Boot Configuration Vector Status (0x0004) on page 16-2
0x008C	DWord	USSBRDELAY	USSBRDELAY - Upstream Secondary Bus Reset Delay (0x008C) on page 17-4
0x0300	DWord	SWPORT0CTL	SWPORT[13:12, 9:0]CTL - Switch Port x Control on page 16-3
0x0304	DWord	SWPORT0STS	SWPORT[13:12, 9:0]STS - Switch Port x Status on page 16-4
0x0320	DWord	SWPORT1CTL	SWPORT[13:12, 9:0]CTL - Switch Port x Control on page 16-3
0x0324	DWord	SWPORT1STS	SWPORT[13:12, 9:0]STS - Switch Port x Status on page 16-4
0x0340	DWord	SWPORT2CTL	SWPORT[13:12, 9:0]CTL - Switch Port x Control on page 16-3
0x0344	DWord	SWPORT2STS	SWPORT[13:12, 9:0]STS - Switch Port x Status on page 16-4
0x0360	DWord	SWPORT3CTL	SWPORT[13:12, 9:0]CTL - Switch Port x Control on page 16-3
0x0364	DWord	SWPORT30STS	SWPORT[13:12, 9:0]STS - Switch Port x Status on page 16-4
0x0380	DWord	SWPORT4CTL	SWPORT[13:12, 9:0]CTL - Switch Port x Control on page 16-3
0x0384	DWord	SWPORT4STS	SWPORT[13:12, 9:0]STS - Switch Port x Status on page 16-4
0x03A0	DWord	SWPORT5CTL	SWPORT[13:12, 9:0]CTL - Switch Port x Control on page 16-3
0x03A4	DWord	SWPORT5STS	SWPORT[13:12, 9:0]STS - Switch Port x Status on page 16-4
0x03C0	DWord	SWPORT6CTL	SWPORT[13:12, 9:0]CTL - Switch Port x Control on page 16-3
0x03C4	DWord	SWPORT6STS	SWPORT[13:12, 9:0]STS - Switch Port x Status on page 16-4
0x03E0	DWord	SWPORT7CTL	SWPORT[13:12, 9:0]CTL - Switch Port x Control on page 16-3
0x03E4	DWord	SWPORT7STS	SWPORT[13:12, 9:0]STS - Switch Port x Status on page 16-4
0x0400	DWord	SWPORT8CTL	SWPORT[13:12, 9:0]CTL - Switch Port x Control on page 16-3
0x0404	DWord	SWPORT8STS	SWPORT[13:12, 9:0]STS - Switch Port x Status on page 16-4
0x0420	DWord	SWPORT9CTL	SWPORT[13:12, 9:0]CTL - Switch Port x Control on page 16-3
0x0424	DWord	SWPORT9STS	SWPORT[13:12, 9:0]STS - Switch Port x Status on page 16-4
0x0480	DWord	SWPORT12CTL	SWPORT[13:12, 9:0]CTL - Switch Port x Control on page 16-3
0x0484	DWord	SWPORT12STS	SWPORT[13:12, 9:0]STS - Switch Port x Status on page 16-4
0x04A0	DWord	SWPORT13CTL	SWPORT[13:12, 9:0]CTL - Switch Port x Control on page 16-3
0x04A4	DWord	SWPORT13STS	SWPORT[13:12, 9:0]STS - Switch Port x Status on page 16-4
0x0800	DWord	S0CTL	S[13:12, 9:0]CTL - SerDes x Control on page 16-5
0x0804	DWord	S0TXLCTL0	S[13:12, 9:0]TXLCTL0 - SerDes x Transmitter Lane Control 0 on page 16-6
0x0808	DWord	S0TXLCTL1	S[13:12, 9:0]TXLCTL1 - SerDes x Transmitter Lane Control 1 on page 16-9
0x0810	DWord	S0RXEQLCTL	S[13:12, 9:0]RXEQLCTL - SerDes x Receiver Equalization Lane Control on page 16-12
0x0820	DWord	S1CTL	S[13:12, 9:0]CTL - SerDes x Control on page 16-5

Table 14.6 Switch Configuration and Status (Part 1 of 4)

Notes

Cfg. Offset	Size	Register Mnemonic	Register Definition
0x0824	DWord	S1TXLCTL0	S[13:12, 9:0]TXLCTL0 - SerDes x Transmitter Lane Control 0 on page 16-6
0x0828	DWord	S1TXLCTL1	S[13:12, 9:0]TXLCTL1 - SerDes x Transmitter Lane Control 1 on page 16-9
0x0830	DWord	S1RXEQLCTL	S[13:12, 9:0]RXEQLCTL - SerDes x Receiver Equalization Lane Control on page 16-12
0x0840	DWord	S2CTL	S[13:12, 9:0]CTL - SerDes x Control on page 16-5
0x0844	DWord	S2TXLCTL0	S[13:12, 9:0]TXLCTL0 - SerDes x Transmitter Lane Control 0 on page 16-6
0x0848	DWord	S2TXLCTL1	S[13:12, 9:0]TXLCTL1 - SerDes x Transmitter Lane Control 1 on page 16-9
0x0850	DWord	S2RXEQLCTL	S[13:12, 9:0]RXEQLCTL - SerDes x Receiver Equalization Lane Control on page 16-12
0x0860	DWord	S3CTL	S[13:12, 9:0]CTL - SerDes x Control on page 16-5
0x0864	DWord	S3TXLCTL0	S[13:12, 9:0]TXLCTL0 - SerDes x Transmitter Lane Control 0 on page 16-6
0x0868	DWord	S3TXLCTL1	S[13:12, 9:0]TXLCTL1 - SerDes x Transmitter Lane Control 1 on page 16-9
0x0870	DWord	S3RXEQLCTL	S[13:12, 9:0]RXEQLCTL - SerDes x Receiver Equalization Lane Control on page 16-12
0x0880	DWord	S4CTL	S[13:12, 9:0]CTL - SerDes x Control on page 16-5
0x0884	DWord	S4TXLCTL0	S[13:12, 9:0]TXLCTL0 - SerDes x Transmitter Lane Control 0 on page 16-6
0x0888	DWord	S4TXLCTL1	S[13:12, 9:0]TXLCTL1 - SerDes x Transmitter Lane Control 1 on page 16-9
0x0890	DWord	S4RXEQLCTL	S[13:12, 9:0]RXEQLCTL - SerDes x Receiver Equalization Lane Control on page 16-12
0x08A0	DWord	S5CTL	S[13:12, 9:0]CTL - SerDes x Control on page 16-5
0x08A4	DWord	S5TXLCTL0	S[13:12, 9:0]TXLCTL0 - SerDes x Transmitter Lane Control 0 on page 16-6
0x08A8	DWord	S5TXLCTL1	S[13:12, 9:0]TXLCTL1 - SerDes x Transmitter Lane Control 1 on page 16-9
0x08B0	DWord	S5RXEQLCTL	S[13:12, 9:0]RXEQLCTL - SerDes x Receiver Equalization Lane Control on page 16-12
0x08C0	DWord	S6CTL	S[13:12, 9:0]CTL - SerDes x Control on page 16-5
0x08C4	DWord	S6TXLCTL0	S[13:12, 9:0]TXLCTL0 - SerDes x Transmitter Lane Control 0 on page 16-6
0x08C8	DWord	S6TXLCTL1	S[13:12, 9:0]TXLCTL1 - SerDes x Transmitter Lane Control 1 on page 16-9
0x08D0	DWord	S6RXEQLCTL	S[13:12, 9:0]RXEQLCTL - SerDes x Receiver Equalization Lane Control on page 16-12
0x08E0	DWord	S7CTL	S[13:12, 9:0]CTL - SerDes x Control on page 16-5

Table 14.6 Switch Configuration and Status (Part 2 of 4)

Notes

Cfg. Offset	Size	Register Mnemonic	Register Definition
0x08E4	DWord	S7TXLCTL0	S[13:12, 9:0]TXLCTL0 - SerDes x Transmitter Lane Control 0 on page 16-6
0x08E8	DWord	S7TXLCTL1	S[13:12, 9:0]TXLCTL1 - SerDes x Transmitter Lane Control 1 on page 16-9
0x08F0	DWord	S7RXEQLCTL	S[13:12, 9:0]RXEQLCTL - SerDes x Receiver Equalization Lane Control on page 16-12
0x0900	DWord	S8CTL	S[13:12, 9:0]CTL - SerDes x Control on page 16-5
0x0904	DWord	S8TXLCTL0	S[13:12, 9:0]TXLCTL0 - SerDes x Transmitter Lane Control 0 on page 16-6
0x0908	DWord	S8TXLCTL1	S[13:12, 9:0]TXLCTL1 - SerDes x Transmitter Lane Control 1 on page 16-9
0x0910	DWord	S8RXEQLCTL	S[13:12, 9:0]RXEQLCTL - SerDes x Receiver Equalization Lane Control on page 16-12
0x0920	DWord	S9CTL	S[13:12, 9:0]CTL - SerDes x Control on page 16-5
0x0924	DWord	S9TXLCTL0	S[13:12, 9:0]TXLCTL0 - SerDes x Transmitter Lane Control 0 on page 16-6
0x0928	DWord	S9TXLCTL1	S[13:12, 9:0]TXLCTL1 - SerDes x Transmitter Lane Control 1 on page 16-9
0x0930	DWord	S9RXEQLCTL	S[13:12, 9:0]RXEQLCTL - SerDes x Receiver Equalization Lane Control on page 16-12
0x0980	DWord	S12CTL	S[13:12, 9:0]CTL - SerDes x Control on page 16-5
0x0984	DWord	S12TXLCTL0	S[13:12, 9:0]TXLCTL0 - SerDes x Transmitter Lane Control 0 on page 16-6
0x0988	DWord	S12TXLCTL1	S[13:12, 9:0]TXLCTL1 - SerDes x Transmitter Lane Control 1 on page 16-9
0x0990	DWord	S12RXEQLCTL	S[13:12, 9:0]RXEQLCTL - SerDes x Receiver Equalization Lane Control on page 16-12
0x09A0	DWord	S13CTL	S[13:12, 9:0]CTL - SerDes x Control on page 16-5
0x09A4	DWord	S13TXLCTL0	S[13:12, 9:0]TXLCTL0 - SerDes x Transmitter Lane Control 0 on page 16-6
0x09A8	DWord	S13TXLCTL1	S[13:12, 9:0]TXLCTL1 - SerDes x Transmitter Lane Control 1 on page 16-9
0x09B0	DWord	S13RXEQLCTL	S[13:12, 9:0]RXEQLCTL - SerDes x Receiver Equalization Lane Control on page 16-12
0x0A90	DWord	GPIOFUNC0	GPIOFUNC0 - General Purpose I/O Function 0 (0x0A90) on page 16-12
0x0A94	DWord	GPIOFUNC1	GPIOFUNC1 - General Purpose I/O Function 1 (0x0A94) on page 16-13
0x0A98	DWord	GPIOAFSEL0	GPIOAFSEL0 - General Purpose I/O Alternate Function Select 0 (0x0A98) on page 16-13
0x0AA8	DWord	GPIOCFG0	GPIOCFG0 - General Purpose I/O Configuration 0 (0x0AA8) on page 16-14

Table 14.6 Switch Configuration and Status (Part 3 of 4)

Notes

Cfg. Offset	Size	Register Mnemonic	Register Definition
0x0AAC	DWord	GPIOCFG1	GPIOCFG1 - General Purpose I/O Configuration 1 (0x0AAC) on page 16-14
0x0AB0	DWord	GPIOD0	GPIOD0 - General Purpose I/O Data 0 (0x0AB0) on page 16-14
0x0AB4	DWord	GPIOD1	GPIOD1 - General Purpose I/O Data 1 (0x0AB4) on page 16-15
0x0AB8	DWord	HPSIGMAP	HPSIGMAP - Hot-Plug GPIO Signal Map (0x0AB8) on page 16-15
0x0ABC	DWord	HPCFGCTL	HPCFGCTL - Hot-Plug Configuration Control (0x0ABC) on page 16-16
0x0AC8	DWord	SMBUSSTS	SMBUSSTS - SMBus Status (0x0AC8) on page 16-17
0x0ACC	DWord	SMBUSCTL	SMBUSCTL - SMBus Control (0x0ACC) on page 16-19
0x0AD0	DWord	EEPROMINTF	EEPROMINTF - Serial EEPROM Interface (0x0AD0) on page 16-19
0x0AD8	DWord	IOEXPADDR0	IOEXPADDR0 - SMBus I/O Expander Address 0 (0x0AD8) on page 16-20
0x0ADC	DWord	IOEXPADDR1	IOEXPADDR1 - SMBus I/O Expander Address 1 (0x0ADC) on page 16-20
0x0AE0	DWord	IOEXPADDR2	IOEXPADDR2 - SMBus I/O Expander Address 2 (0x0AE0) on page 16-21
0x0AE4	DWord	IOEXPADDR3	IOEXPADDR3 - SMBus I/O Expander Address 3 (0x0AE4) on page 16-21
0x0AE8	DWord	GPECTL	GPECTL - General Purpose Event Control (0x0AE8) on page 16-21
0x0AEC	DWord	GPESTS	GPESTS - General Purpose Event Status (0x0AEC) on page 16-22
0x0AF0 — 0x0F10		Reserved	

Table 14.6 Switch Configuration and Status (Part 4 of 4)

PCI to PCI Bridge and Proprietary Port Specific Registers

Type 1 Configuration Header Registers

VID - Vendor Identification Register (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.

DID - Device Identification Register (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 bridge.

PCICMD - PCI Command Register (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 handles these as Unsupported Requests (UR). Additionally, the bridge handles non-posted transactions in the upstream direction 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.

Bit Field	Field Name	Type	Default Value	Description
5	VGAS	RO	0x0	**VGA Palette Snoop.** Not applicable.
6	PERRE	RW	0x0	**Parity Error Response Enable.** This bit controls the logging of poisoned TLPs in the Master Data Parity Error bit (MDPED) in the PCI Status (PCISTS) register.
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 section PCIEDCTL - PCI Express Device Control (0x048) on page 15-13). In addition, when this bit is set it enables the forwarding of ERR_NONFATAL and ERR_FATAL error messages from the secondary to the primary interface. ERR_CORR messages are unaffected by this bit and are always forwarded. 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-to-PCI bridge to generate an INTx interrupt message. When this bit is set, any interrupts generated by this bridge are negated. This may result in a change in the resolved interrupt state of the bridge. This bit has no effect on interrupts forwarded from the secondary to the primary interface.
15:11	Reserved	RO	0x0	Reserved field.

PCISTS - PCI Status Register (0x006)

Bit Field	Field Name	Type	Default Value	Description
2:0	Reserved	RO	0x0	Reserved field.
3	INTS	RO	0x0	**INTx Status.** This bit is set when an INTx interrupt is pending from the function. INTx emulation interrupts forwarded by switch ports from devices downstream of the bridge are not reflected in this bit. For downstream ports, this bit is set if an interrupt has been “asserted” by the corresponding port’s hot-plug controller.
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 field.
7	FB2B	RO	0x0	**Fast Back-to-Back (FB2B).** Not applicable.

Bit Field	Field Name	Type	Default Value	Description
8	MDPED	RW1C	0x0	**Master Data Parity Error Detected.** This bit is set by the bridge function if the PERRE bit in the PCI Command register (PCICMD) is set to 0x1 and either of the following two conditions occurs: the function receives a Poisoned Completion going Downstream, or the function transmits a Poisoned Request Upstream.
10:9	DEVT	RO	0x0	**DEVSEL# TIming.** Not applicable.
11	STAS	RW1C	0x0	**Signaled Target Abort.** This bit is set when the bridge completes a posted or non-posted request with a completer-abort error. In the switch, this bit is set when the bridge blocks a multicast TLP received on its primary side.
12	RTAS	RO	0x0	**Received Target Abort.** Not applicable (the bridge function never generates requests on its own behalf).
13	RMAS	RO	0x0	**Received Master Abort.** Not applicable (the bridge function never generates requests on its own behalf).
14	SSE	RW1C	0x0	**Signaled System Error.** This bit is set when the bridge function 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 signaled.
15	DPE	RW1C	0x0	**Detected Parity Error.** This bit is set by the bridge function whenever it receives a poisoned TLP on the primary side regardless of the state of the PERRE bit in the PCI Command (PCICMD) register.

RID - Revision Identification Register (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.

CCODE - Class Code Register (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.

CLS - Cache Line Size Register (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 Register (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.

PBUSN - Primary Bus Number Register (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. This field has no functional effect within the switch but is implemented as a read/write register for software compatibility.

SBUSN - Secondary Bus Number Register (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 Register (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 Register (0x01B)

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

IOBASE - I/O Base Register (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. This bit always reflects the value of the IOCAP field in the IOBASE register.
3:1	Reserved	RO	0x0	Reserved field.
7:4	IOBASE	RW	0xF	**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.

IOLIMIT - I/O Limit Register (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	Reserved field.
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 Register (0x01E)

Bit Field	Field Name	Type	Default Value	Description
7:0	Reserved	RO	0x0	Reserved field.
8	MDPED	RW1C	0x0	**Master Data Parity Error.** This bit is set by the bridge function if the PERRE bit in the Bridge Control (BCTL) register is set to 0x1 and either of the following two conditions occurs: the function receives a Poisoned Completion going Upstream, or the function transmits a Poisoned Request Downstream.
10:9	DVSEL	RO	0x0	Not applicable.
11	STAS	RW1C	0x0	**Signaled Target Abort Status.** This bit is set when the bridge completes a posted or non-posted request with a completer-abort error on its secondary side. In the switch, this bit is set when the bridge completes a posted or non-posted request received on its secondary side with completer-abort status as a result of an ACS violation, or when the bridge blocks a multicast TLP received on its secondary side.
12	RTAS	RO	0x0	**Received Target Abort Status.** Not applicable (the internal P2P bridges within the switch never generate requests on their own behalf).
13	RMAS	RO	0x0	**Received Master Abort Status.** Not applicable (the internal P2P bridges within the switch never generate requests on their own behalf).
14	RSE	RW1C	0x0	**Received System Error.** 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 internal P2P bridge within the switch whenever it receives a poisoned TLP on the secondary side regardless of the state of the PERRE bit in the PCI Command register.

MBASE - Memory Base Register (0x020)

Bit Field	Field Name	Type	Default Value	Description
3:0	Reserved	RO	0x0	Reserved field.
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 Register (0x022)

Bit Field	Field Name	Type	Default Value	Description
3:0	Reserved	RO	0x0	Reserved field.
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 Register (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	Reserved field.
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.

PMLIMIT - Prefetchable Memory Limit Register (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.

Bit Field	Field Name	Type	Default Value	Description
3:1	Reserved	RO	0x0	Reserved field.
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 Register (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 Register (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 Register (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.

IOLIMITU - I/O Limit Upper Register (0x032)

Bit Field	Field Name	Type	Default Value	Description
15:0	IOLIMITU	RW	0x0	**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 Register (0x034)

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

EROMBASE - Expansion ROM Base Address Register (0x038)

Bit Field	Field Name	Type	Default Value	Description
31:0	EROMBASE	RO	0x0	**Expansion ROM Base Address.** The internal P2P bridges within the switch do not implement expansion ROMs. Thus, this field is hardwired to zero.

INTRLINE - Interrupt Line Register (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 Register (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 internal P2P bridges 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. The switch's bridges may only be configured to generate INTA interrupts. Therefore, correct values for this field are only 0x0 and 0x1. 0x0 - (none) Bridge does not generate any interrupts. 0x1 - (INTA) Bridge generates INTA interrupts. 0x2 - (INTB) Bridge generates INTB interrupts. 0x3 - (INTC) Bridge generates INTC interrupts. 0x4 - (INTD) Bridge generates INTD interrupts.

BCTL - Bridge Control Register (0x03E)

Bit Field	Field Name	Type	Default Value	Description
0	PERRE	RW	0x0	**Parity Error Response Enable**. This bit controls the logging of poisoned TLPs in the Master Data Parity Error bit (MDPED) in the Secondary Status (SECSTS) register.
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	RW	0x0	**ISA Enable.** This bit controls the routing of ISA I/O transactions. 0 - (disable) Forward downstream all I/O addresses in the address range defined by the I/O base and I/O limit registers 1 - (enable) Forward upstream ISA I/O addresses in the address range defined by the I/O base and I/O limit registers that are in the first 64 KB of PCI I/O address space (top 768 bytes of each 1-KB block)
3	VGAEN	RW	0x0	**VGA Enable**. 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.
4	VGA16EN	RW	0x0	**VGA 16-bit Enable.** This bit only has an effect when the VGAEN bit is set in this register. This read/write bit enables system configuration software to select between 10-bit and 16-bit I/O space decoding for VGA transactions. 0 - (bit10) Perform 10-bit decoding. I/O space aliasing occurs in this mode. 1 - (bit16) Perform 16-bit decoding. No I/O space aliasing occurs in this mode.
5	Reserved	RO	0x0	Reserved field.
6	SRESET	RW	0x0	**Secondary Bus Reset.** Setting this bit triggers a secondary bus reset. In the upstream port, setting this bit initiates a Upstream Secondary Bus Reset. In a downstream port, setting this bit initiates a Downstream Secondary Bus Reset. Port Configuration Registers must not be changed except as required to update port status.
15:7	Reserved	RO	0x0	Reserved field.

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.
15:8	NXTPTR	RWL	Refer to section Capability Structures on page 14-3.	**Next Pointer.** This field contains a pointer to the next capability structure.
19:16	VER	RWL	0x2	**PCI Express Capability Version.** This field indicates the PCI-SIG defined PCI Express capability structure version number. The switch is compliant with the Express Capabilities Register Expansion ECN.
23:20	TYPE	RO	Upstream: 0x5 Downstream: 0x6	**Port Type.** This field identifies the type of switch port (upstream or downstream).
24	SLOT	RWL	0x0	**Slot Implemented.** This bit is set when the PCI Express link associated with this Port is connected to a slot. This field does not apply to an upstream port and should be set to zero.
29:25	IMN	RO	0x0	**Interrupt Message Number.** The function is allocated only one MSI. 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. 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 - (s4096) 4096 bytes max payload size 0x6 - reserved (treated as 128 bytes) 0x7 - reserved (treated as 128 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	RWL	0x1	**Extended Tag Field Support.** This field indicates the maximum supported size of the Tag field as a requester.

Bit Field	Field Name	Type	Default Value	Description
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 is only applicable to endpoint functions.
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 is only applicable to endpoint functions.
12	ABP	RO	0x0	**Attention Button Present.** In PCI Express base 1.0a when set, this bit indicates that an Attention Button is implemented on the card/module. The value of this field is undefined in PCI Express 2.0 specification.
13	AIP	RO	0x0	**Attention Indicator Present.** In PCI Express base 1.0a when set, this bit indicates that an Attention Indicator is implemented on the card/module. The value of this field is undefined in PCI Express 2.0 specification.
14	PIP	RO	0x0	**Power Indicator Present.** In PCI Express base 1.0a when set, this bit indicates that a Power Indicator is implemented on the card/module. The value of this field is undefined in PCI Express 2.0 specification
15	RBERR	RO	0x1	**Role Based Error Reporting.** This bit is set to indicate that the switch supports error reporting as defined in PCI Express 2.0 specification.
17:16	Reserved	RO	0x0	Reserved field.
25:18	CSPLV	RO	0x0	**Captured Slot Power Limit Value.** This field in combination with the Slot Power Limit Scale value, specifies the upper limit on power supplied by the slot. Power limit (in Watts) calculated by multiplying the value in this field by the value in the Slot Power Limit Scale field. The value of this field is set by a Set_Slot_Power_Limit Message and is only applicable for an upstream port.[1] A port in unattached mode does not modify this field as a result of receiving a Set_Slot_Power_Limit Message. This field is always zero in unattached or downstream ports.
27:26	CSPLS	RO	0x0	**Captured Slot Power Limit Scale.** This field specifies the scale used for the Slot Power Limit Value. The value of this field is set by a Set_Slot_Power_Limit Message and is only applicable for an upstream port. 0 - (v1) 1.0x 1 - (v1p1) 0.1x 2 - (v0p01) 0.01x 3 - (v0p001x) 0.001x A port in unattached mode does not modify this field as a result of receiving a Set_Slot_Power_Limit Message. This field is always zero in unattached or downstream ports.
31:28	Reserved	RO	0x0	Reserved field.

[1] NOTE: Set_Slot_Power_Limit messages received by a port implicitly target all functions in the port.

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. This field should be set to a value less than that advertised by the Maximum Payload Size Supported (MPAYLOAD) field in the PCI Express Device Capabilities (PCIEDCAP) register. Setting this field to a value larger than that advertised in the MPAYLOAD field produces undefined results. 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 - (s4096) 4096 bytes max payload size 0x6 - reserved (treated as 128 bytes) 0x7 - reserved (treated as 128 bytes)
8	ETFEN	RW	0x0	**Extended Tag Field Enable.** Since the bridge function never generates a transaction that requires a completion, this bit has no functional effect on the device during normal operation. To aid in debug, when the SEQTAG field is set in the TLCTL register, this field controls whether tags are generated in the range from 0 through 31 or from 0 through 255.
9	PFEN	RO	0x0	**Phantom Function Enable.** The internal P2P bridges within the switch do not support phantom function numbers. Therefore, this field is hardwired 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 bridge function does not generate transactions with the No Snoop bit set and passes transactions through the bridge with the No Snoop bit unmodified.
14:12	MRRS	RO	0x0	**Maximum Read Request Size.** The bridge function 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 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 P2P bridges within the switch do not issue Non-Posted Requests on their own behalf. Therefore, this field is hardwired to zero.
15:6	Reserved	RO	0x0	Reserved field.

PCIELCAP - PCI Express Link Capabilities (0x04C)

Bit Field	Field Name	Type	Default Value	Description
3:0	MAXLNKSPD	RO	0x2	**Maximum Link Speed.** This field indicates the supported link speeds of the port. 1 - (gen1) 2.5 Gbps 2 - (gen2) 5 Gbps others - reserved

Bit Field	Field Name	Type	Default Value	Description
9:4	MAXLNK-WDTH	RWL	HWINIT	**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 an invalid or reserved value is allowed and results in the port operating at its default (i.e., initial) value. The value written to this field is never modified by hardware 0 - reserved 1 - (x1) x1 link width 2 - (x2) x2 link width 4 - (x4) x4 link width 8 - (x8) x8 link width 12 - (x12) x12 link width 16 - (x16) x16 link width 32 - (x32) x32 link width others - reserved **Note:** This field must only be modified when reducing to reduce the port's maximum link width default value. Otherwise, operation is undefined. In addition, modifying this field when the port is in an operating mode other than disabled causes the port to immediately re-train the link via the PHY's Recovery and Configuration states. This immediate transition may result in receiver errors being logged in the AER Correctable Error Status (AERCES) register.
11:10	ASPMS	RWL	0x3	**Active State Power Management (ASPM) Support.** This default value of this field is 0x3 to indicate that L0s and L1 are supported. This field may be overridden to allow user control over the ASPM capabilities of this port (L0s and/or L1).
14:12	L0SEL	RWL	0x6	**L0s Exit Latency.** This field indicates the L0s exit latency for the given PCI Express link. Transitioning from L0s to L0 always requires approximately 2.04µs. Thus, default value indicates an L0s exit latency between 2µs and 4µs.
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 approximately 2.3µs. Therefore, a value 2µs to less than 4µs is reported with a default value of 0x2.
18	CPM	RWL	0x0	**Clock Power Management.** This bit indicates if the component tolerates removal of the reference clock via the "CLKREQ#" mach-anism. The switch does not support the removal of reference clocks.
19	SDERR	RWL	Upstream: 0x0 Downstream: 0x1	**Surprise Down Error Reporting.** The switch dDownstream ports support surprise down error reporting. This field does not apply to an upstream port and should be hard-wired to zero.
20	DLLLA	RWL	Upstream: 0x0 Downstream: 0x1	**Data Link Layer Link Active Reporting.** The switch downstream ports support the capability of reporting the DL_Active state of the data link control and management state machine. This field is not applicable for the upstream port and must be hard-wired to zero.

Bit Field	Field Name	Type	Default Value	Description
21	LBN	RWL	Upstream: 0x0 Downstream: 0x1	**Link Bandwidth Notification Capability.** When set, this bit indicates support for the link bandwidth notification status and interrupt mechanisms. The switch downstream ports support the capability. This field is not applicable for the upstream port and must be zero.
23:22	Reserved	RO	0x0	Reserved field.
31:24	PORTNUM	RO	Port 0: 0x0 Port 1: 0x1 Port 2: 0x2 Port 3: 0x3 Port 4: 0x4 Port 5: 0x5 Port 6: 0x6 Port 7: 0x7 Port 8: 0x8 Port 9: 0x9 Port 12: 0xC Port 13: 0xD	**Port Number.** This field indicates the PCI express port number for the corresponding link.

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 Note that "L0s enable entry" corresponds to the transmitter entering L0s (the receiver supports this function and is not affected by this setting).
2	Reserved	RO	0x0	Reserved field.
3	RCB	RO	0x0	**Read Completion Boundary.** This field is not applicable and is hardwired to zero.
4	LDIS	RW	0x0	**Link Disable.** When set in a downstream port, this bit disables the link. This bit is not applicable on an upstream port.
5	LRET	Upstream Port: RWL Downstream Port: 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. It is permitted to set this bit while simultaneously modifying other fields in this register. When this bit is set and the LTSSM is already in the Recovery or Configuration states, all modifications that affect link retraining are applied in the subsequent retraining. Else, if the LTSSM is not in the Recovery or Configuration states, modifications that affect link retraining are applied immediately.

Bit Field	Field Name	Type	Default Value	Description
6	CCLK	RW	0x0	**Common Clock Configuration.** When set, this bit indicates that this port and the port at the opposite end of the link are operating with a distributed common reference clock. After modifying this bit in both components of the link, software must trigger a link retrain by setting the link retrain bit in the upstream component's Link Control register. In the switch, the L0s and L1 exit latencies do not change among common and non-common clock configurations.
7	ESYNC	RW	0x0	**Extended Sync**b When set this bit forces transmission of additional ordered sets when exiting the L0s state and when in the recovery state.
8	CLKPWRMGT	RO	0x0	**Enable Clock Power Management.** The switch does not support this feature.
9	HAWD	RO	0x0	**Hardware Autonomous Width Disable.** When set, this bit disables hardware from changing the link width for reasons other than attempting to correct for unreliable link operation by reducing the link width. The switch ports do not have a hardware autonomous mechanism to change link width, except due to link reliability issues. Therefore, this bit is not applicable and is hardwired to zero. Note that this bit does not affect link width changes triggered by the link width re-configuration mechanism.
10	LBWINTEN	RW	0x0	**Link Bandwidth Management Interrupt Enable.** When set, this bit enables the generation of an interrupt to indicate that the LBWSTS bit has been set in the PCIELSTS register. If the LBN field in the PCIELCAP register is cleared, this field is hardwired to zero. This field is hardwired to zero in an upstream port.
11	LABWINTEN	RW	0x0	**Link Autonomous Bandwidth Interrupt Enable.** When set, this bit enables the generation of an interrupt to indicate that the LABWSTS bit has been set in the PCIELSTS register. If the LBN field in the PCIELCAP register is cleared, this field is hardwired to zero. This field is hardwired to zero in an upstream port.
15:12	Reserved	RO	0x0	Reserved field.

PCIELSTS - PCI Express Link Status (0x052)

Bit Field	Field Name	Type	Default Value	Description
3:0	CLS	RO	0x1	**Current Link Speed.** This field indicates the current link speed of the port. 1 - (gen1) 2.5 GT/s 2 - (gen2) 5 GT/s others - reserved

Bit Field	Field Name	Type	Default Value	Description
9:4	NLW	RO	HWINIT	**Negotiated Link Width.** This field indicates the negotiated width of the link. 00 0001b - x1 00 0010b - x2 00 0100b - x4 00 1000b - x8 00 1100b - x12 01 0000b - x16 10 0000b - x32 When the MAXLNKWDTH field in the PCIELCAP register selects a width not supported by the port, the value of this field corresponds to the setting of the MAXLNKWDTH field, regardless of the actual negotiated link width. When the MAXLNKWDTH field in the PCIELCAP register selects a width supported by the port, but the link is unable to train, the value in this field is set to 0x0.
10	Reserved	RO	0x0	Reserved field.
11	LTRAIN	RO	0x0	**Link Training.** When set, this bit indicates that link training is in progress. This bit is set when the Physical Layer LTSSM is in Configuration or Recovery State, or when 0x1 is written to LRET bit in the PCIELCTL register but Link training has not yet begun. Hardware clears this bit when LTSSM exits Configuration/ Recovery State. This bit is only valid when the port operates in 'Downstream Switch Port' mode. Else, this bit always has a value of 0x0.
12	SCLK	RWL	HWINIT	**Slot Clock Configuration.** When set, this bit indicates that the port uses the same physical reference clock used by its link partner (i.e., common-clock configuration). The initial value of this field depends on the port's clocking mode. Refer to Table 4.1 for further details.
13	DLLLA	RO	0x0	**Data Link Layer Link Active.** This bit indicates the status for the data link control and management state machine. 0x0 - (not_active) Data link layer not active state 0x1 - (active) Data link layer active state This bit must never be set by hardware if the DLLLA bit in the PCIELCAP register is cleared.
14	LBWSTS	RW1C	0x0	**Link Bandwidth Management Status.** This bit is set to indicate that either of the following have occurred without the link transitioning through the DL_Down state. A link retraining initiated by setting the LRET bit in the PCIELCTL register has completed. The PHY has autonomously changed link speed or width to attempt to correct unreliable link operation either through an LTSSM time-out or a higher level process. This bit must be set if the Physical Layer reports a speed or width change was initiated by the downstream component that was not indicated as an autonomous change. If the LBN field in the PCIELCAP register is cleared, this field is hardwired to zero. This field is hardwired to zero in an upstream port.

Bit Field	Field Name	Type	Default Value	Description
15	LABWSTS	RW1C	0x0	**Link Autonomous Bandwidth Status.** This bit is set to indicate that either that the PHY has autonomously changed link speed or width for reasons other than to attempt to correct unreliable link operation. This bit must be set if the Physical Layer reports a speed or width change was initiated by the downstream component that was indicated as an autonomous change. If the LBN field in the PCIELCAP register is cleared, this field is hardwired to zero. This field is hardwired to zero in an upstream port.

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.

Bit Field	Field Name	Type	Default Value	Description
14:7	SPLV	RW	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	RW	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.
17	EIP	RWL	0x0	**Electromechanical Interlock Present.** This bit is set if an electro-mechanical interlock is implemented on the chassis for this slot. This bit is read-only and has a value of zero when the SLOT bit in the PCIECAP 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.
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	HWINIT	**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. When the corresponding capability is enabled, the initial value of this field is equal to the value of the corresponding field in the PCIESCTLIV register.

Bit Field	Field Name	Type	Default Value	Description
1	PFDE	RW	HWINIT	**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. When the corresponding capability is enabled, the initial value of this field is equal to the value of the corresponding field in the PCIESCTLIV register.
2	MRLSCE	RW	HWINIT	**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. When the corresponding capability is enabled, the initial value of this field is equal to the value of the corresponding field in the PCIESCTLIV register.
3	PDCE	RW	HWINIT	**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. When the corresponding capability is enabled, the initial value of this field is equal to the value of the corresponding field in the PCIESCTLIV register.
4	CCIE	RW	HWINIT	**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. When the corresponding capability is enabled, the initial value of this field is equal to the value of the corresponding field in the PCIESCTLIV register.
5	HPIE	RW	HWINIT	**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. When the corresponding capability is enabled, the initial value of this field is equal to the value of the corresponding field in the PCIESCTLIV register.
7:6	AIC	RW	HWINIT	**Attention Indicator Control.** When read, this register returns the current state of the Attention Indicator. Writing to this register sets the indicator. 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. When the corresponding capability is enabled, the initial value of this field is equal to the value of the corresponding field in the PCIESCTLIV register. 0x0 - (reserved) Reserved 0x1 (on) On 0x2 - (blink) Blink 0x3 - (off) Off

Bit Field	Field Name	Type	Default Value	Description
9:8	PIC	RW	HWINIT	**Power Indicator Control.** When read, this register returns the current state of the Power Indicator. Writing to this register sets the indicator. 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. When the corresponding capability is enabled, the initial value of this field is equal to the value of the corresponding field in the PCIESCTLIV 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	HWINIT	**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. When the corresponding capability is enabled, the initial value of this field is equal to the value of the corresponding field in the PCIESCTLIV register. 0x0 -(on) Power on 0x1 -(off) Power off
11	EIC	RW	HWINIT	**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.
12	DLLLASCE	RW	HWINIT	**Data Link Layer Link Active State Change Enable.** This bit when set enables generation of a Hot-Plug interrupt or wake-up event on a data link layer active field state change. When the corresponding capability is enabled, the initial value of this field is equal to the value of the corresponding field in the PCIESCTLIV register.
15:13	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.

Bit Field	Field Name	Type	Default Value	Description
2	MRLSC	RW1C	0x0	**MRL Sensor Changed.** Set when an MRL Sensor state change is detected.
3	PDC	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. If the bit is already set, then it remains set. A single write to the PCI Express Slot Control (PCIESCTL) register is considered to be a single command even if it affects more than one field in that register. This command completed bit is not set until processing of all actions associated with all fields in the PCIESCTL register have completed (i.e., all associated SMBus I/O expander transactions have completed).
5	MRLSS	RO	0x0	**MRL Sensor State.** This field enclosed the current state of the MRL sensor. 0x0 - (closed) MRL closed 0x1 - (open) MRL open
6	PDS	RO	0x1	**Presence Detect State.** When the Slot Implemented (SLOT) bit is set in the PCI Express Capabilities (PCIECAP) register, this bit indicates the presence of a card in the slot corresponding to the port and reflects the state of the Presence Detect status. When the SLOT bit is cleared in the PCIECAP register, this bit is hardwired to one in downstream ports (i.e., it is read-only with a constant value of one). This bit is always cleared in upstream ports (i.e., it is read-only with a constant value of zero). 0x0 - (empty) Slot empty 0x1 - (present) Card present
7	EIS	RO	0x0	**Electromechanical Interlock Status.** When an electromechanical interlock is implemented, this bit indicates the current status of the interlock. The status of this bit is determined by the state of the corresponding PxILOCKST input signal on I/O expander 9. If the hot-plug signals associated with the port are mapped to GPIO pins or if I/O expander is not enabled, then the state of this bit defaults to zero (i.e., disengaged). 0x0 - (disengaged) Electromechanical interlock disengaged 0x1 - (engaged) Electromechanical interlock engaged
8	DLLLASC	RW1C	0x0	**Data Link Layer Link Active State Change.** This bit is set when the state of the data link layer active field in the link status register changes state. 0x0 - (nochange) No DLLLA state change 0x1 -(changed) DLLLA state change
15:9	Reserved	RO	0x0	Reserved field.

PCIEDCAP2 - PCI Express Device Capabilities 2 (0x064)

Bit Field	Field Name	Type	Default Value	Description
4:0	Reserved	RO	0x0	Reserved field.
5	ARIFS	RWL	0x1	**ARI Forwarding Supported.** This bit is set to indicate that the switch supports ARI Forwarding. When this bit is cleared, the ARI Forwarding Enable (ARIFEN) bit in the Device Control 2 register becomes read-only zero. This bit is read-only zero in an upstream port.
31:6	Reserved	RO	0x0	Reserved field.

PCIEDCTL2 - PCI Express Device Control 2 (0x068)

Bit Field	Field Name	Type	Default Value	Description
4:0	Reserved	RO	0x0	Reserved field.
5	ARIFEN	RW	0x0	**ARI Forwarding Enable.** When set, the downstream port disables its traditional Device Number field being zero enforcement when turning a Type 1 configuration request into a Type 0 configuration request, permitting access to the extended functions in an ARI device immediately below the port. When the ARIFS bit in the PCIEDCAP2 register is cleared, this bit is read-only zero. This bit is always read-only zero in an upstream port.
15:6	Reserved	RO	0x0	Reserved field.

PCIEDSTS2 - PCI Express Device Status 2 (0x06A)

Bit Field	Field Name	Type	Default Value	Description
15:0	Reserved	RO	0x0	Reserved field.

PCIELCAP2 - PCI Express Link Capabilities 2 (0x06C)

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

PCIELCTL2 - PCI Express Link Control 2 (0x070)

Bit Field	Field Name	Type	Default Value	Description
3:0	TLS	RW	0x2 Sticky	**Target Link Speed.** For downstream ports, this field sets an upper limit on the link operational speed by restricting the values advertised by the upstream component in its training sequences. For both upstream and downstream ports, this field is used to set the target compliance mode speed when software is using the ECOMP bit in this register to force a link into compliance mode. The switch supports 2.5 GT/s and 5.0 GT/s operation. Setting this field to an unsupported value produces undefined results. 0x1 - (gen1) 2.5 GT/s 0x2 - (gen2) 5.0 GT/s others - reserved
4	ECOMP	RW	0x0 Sticky	**Enter Compliance.** Software is permitted to force a link into compliance mode at the speed indicated by the TLS field by setting this bit in both components on a link and then initiating a hot reset on the link.
5	HASD	RO	0x0	**Hardware Autonomous Speed Disable**. When set, this bit prevents hardware from changing the link speed for device specific reasons other than to correct unreliable link operation by reducing the link speed. Initial transition to the highest supported common link speed is not blocked by this bit. Switch ports do not have an autonomous mechanism to regulate link speed, except due to link reliability issues. Therefore, this bit is not applicable to the switch ports. Note that this bit does not affect link speed changes triggered by software setting the target link speed and link-retrain bits.
6	SDE	RWL	0x0	**Selectable De-emphasis.** For switch downstream ports, this bit sets the de-emphasis level when the link operates at 5.0 GT/s. For the upstream port, this bit selects the de-emphasis preference advertised via training sets (the actual de-emphasis on the link is selected by the link partner). 0x0 - De-emphasis level = -6.0 dB 0x1 - De-emphasis level = -3.5 dB This bit has no effect when the link operates at 2.5 GT/s, or when the link operates in low-swing mode. When this field is modified, the newly selected de-emphasis is not applied until the PHY LTSSM transitions through the states in which it is allowed to modify the de-emphasis setting on the line (e.g., Recovery.Speed). Therefore, after modifying this field, it is recommended that the link be fully retrained by setting the FLRET bit in the PHYLSTATE0 register.

Bit Field	Field Name	Type	Default Value	Description
9:7	TM	RW	0x0 Sticky	**Transmit Margin.** This field controls the value of the non de-emphasized voltage level at the transmitter pins. This field is reset to 0x0 on entry to the LTSSM Polling.Configuration substate. 0x0 - Normal operating range 0x1 - 900 mV for full swing and 500 mV for low-swing 0x2 - 700 mV for full swing and 400 mV for low-swing 0x3 - 500 mV for full swing and 300 mV for low-swing 0x4 - 300 mV for full swing and 200 mv for low-swing 0x5 - 200 mV for full swing and 100 mv for low-swing 0x6 - 0x7 - Reserved This register is intended for debug, compliance testing purpose only. System firmware and software is allowed to modify this register only during debug or compliance testing. In all other cases, the system must ensure that this register is set to the default value. When this field is set to “Normal Operating Range”, the SerDes transmitter drive level is selected via the SerDes Transmitter Control registers (S[x]TXLCTL0 and S[x]TXLCTL1). Refer to section SerDes Transmitter Controls on page 7-1. When this field is modified, the newly selected value is not applied until the PHY LTSSM transitions through the states in which it is allowed to modify the transmit margin setting on the line (i.e., Recovery.RcvrLock). Therefore, after modifying this field, it is recommended that the link be retrained by setting the LRET bit in the PCIELCTL register.
10	EMC	RW	0x0 Sticky	**Enter Modified Compliance.** When this bit is set to 1b, the port transmits the modified compliance pattern if the LTSSM enters Polling.Compliance state. This register is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this register only during debug or compliance testing. In all other cases, the system must ensure that this register is set to the default value.
11	CSOS	RW	0x0 Sticky	**Compliance SOS.** When set to 1b, the LTSSM is required to send SOS periodically in between the compliance and modified compliance patterns.
12	CDE	RW	0x0 Sticky	**Compliance De-emphasis.** This bit selects the de-emphasis value in the Polling.Compliance state when this state was entered as a result of setting the Enter Compliance (ECOMP) bit in this register. 0x0 - De-emphasis level = -6.0 dB 0x1 - De-emphasis level = -3.5 dB This bit is intended for debug, compliance testing purposes. System firmware and software is allowed to modify this bit only during debug or compliance testing.
15:13	Reserved	RO	0x0	Reserved field.

PCIELSTS2 - PCI Express Link Status 2 (0x072)

Bit Field	Field Name	Type	Default Value	Description
0	CDE	RO	0x0	**Current De-emphasis.** The value of this bit indicates the current de-emphasis level when the link operates in 5.0 Gbps. 0x0 - De-emphasis level = -6.0 dB 0x1 - De-emphasis level = -3.5 dB The value of this bit in undefined when the link operates at 2.5 GT/s.
15:1	Reserved	RO	0x0	Reserved field.

PCIESCAP2 - PCI Express Slot Capabilities 2 (0x074)

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

PCIESCTL2 - PCI Express Slot Control 2 (0x078)

Bit Field	Field Name	Type	Default Value	Description
15:0	Reserved	RO	0x0	Reserved field.

PCIESSTS2 - PCI Express Slot Status 2 (0x07A)

Bit Field	Field Name	Type	Default Value	Description
15:0	Reserved	RO	0x0	Reserved field.

Power Management Capability Structure

PMCAP - PCI Power Management Capabilities (0x0C0)

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	RWL	Refer to section Capability Structures on page 14-3.	**Next Pointer.** This field contains a pointer to the next capability structure.

Bit Field	Field Name	Type	Default Value	Description
18:16	VER	RO	0x3	**Power Management Capability Version.** This field indicates compliance with version two of the specification. Complies with version the PCI Bus Power Management Interface Specification, Revision 1.2.
19	PMECLK	RO	0x0	**PME Clock.** Does not apply to PCI Express.
20	Reserved	RO	0x0	Reserved field.
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.** The PES48T12G2 does not use auxiliary current.
25	D1	RO	0x0	**D1 Support.** This field indicates that the PES48T12G2 does not support D1.
26	D2	RO	0x0	**D2 Support.** This field indicates that the PES48T12G2 does not support D2.
31:27	PME	RWL	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 P2P bridge within the switch 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. Modification of this field modifies the advertised capability value but does not modify the device behavior (i.e., PME is generated in the states noted in the default value).

PMCSR - PCI Power Management Control and Status (0x0C4)

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 the switch and reserved) 0x2 -(d2) D2 state (not supported by the switch and reserved) 0x3 -(d3) $D3_{hot}$ state
2	Reserved	RO	0x0	Reserved field.
3	NOSOFTRST	RWL	0x1	**No Soft Reset.** This bit indicates if the configuration context is preserved by the bridge when the device transitions from a D3hot to D0 power management state. 0x0 - (reset) State reset 0x1 - (preserved) State preserved
7:4	Reserved	RO	0x0	Reserved field.

Bit Field	Field Name	Type	Default Value	Description
8	PMEE	RW	0x0 Sticky	**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. 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	**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 P2P bridge within the switch is propagating a PME message but the port is not itself generating a PME. Since the upstream port never generates a PME, this bit will never be set in that port.
21:16	Reserved	RO	0x0	Reserved field.
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.

Message Signaled Interrupt Capability Structure

MSICAP - Message Signaled Interrupt Capability and Control (0x0D0)

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	RWL	Refer to section Capability Structures on page 14-3.	**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.
22:20	MME	RW	0x0	**Multiple Message Enable.** Hardwired to one message.
23	A64	RO	0x1	**64-bit Address Capable.** The P2P bridges within the switch are capable of generating messages using a 64-bit address.
31:24	Reserved	RO	0x0	Reserved field.

MSIADDR - Message Signaled Interrupt Address (0x0D4)

Bit Field	Field Name	Type	Default Value	Description
1:0	Reserved	RO	0x0	Reserved field.
31:2	ADDR	RW	0x0	**Message Address.** This field specifies the lower portion of the DWORD address of the MSI memory write transaction. The switch assumes that all downstream port generated MSIs are targeted to the root and routes these transactions to the upstream port. Configuring the address contained in a downstream port's MSIADDR and MSIADDRU registers to an address that does not route to the upstream port and generating an MSI produces undefined results.

MSIUADDR - Message Signaled Interrupt Upper Address (0x0D8)

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. The switch assumes that all downstream port generated MSIs are targeted to the root and routes these transactions to the upstream port. Configuring the address contained in a downstream port's MSIADDR and MSIADDRU registers to an address that does not route to the upstream port and generating an MSI produces undefined results.

MSIMDATA - Message Signaled Interrupt Message Data (0x0DC)

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 signaled.
31:16	Reserved	RO	0x0	Reserved field.

Subsystem ID and Subsystem Vendor ID

SSIDSSVIDCAP - Subsystem ID and Subsystem Vendor ID Capability (0x0F0)

Bit Field	Field Name	Type	Default Value	Description
7:0	CAPID	RO	0xD	**Capability ID.** The value of 0xD identifies this capability as a SSID/SSVID capability structure.
15:8	NXTPTR	RWL	Refer to section Capability Structures on page 14-3.	**Next Pointer.** This field contains a pointer to the next capability structure.
31:16	Reserved	RO	0x0	Reserved field.

SSIDSSVID - Subsystem ID and Subsystem Vendor ID (0x0F4)

Bit Field	Field Name	Type	Default Value	Description
15:0	SSVID	RWL	0x0	**SubSystem Vendor ID.** This field identifies the manufacturer of the add-in card or subsystem. SSVID values are assigned by the PCI-SIG to insure uniqueness.
31:16	SSID	RWL	0x0	**Subsystem ID.** This field identifies the add-in card or subsystem. SSID values are assigned by the vendor.

Extended Configuration Space Access Registers

ECFGADDR - Extended Configuration Space Access Address (0x0F8)

Bit Field	Field Name	Type	Default Value	Description
1:0	Reserved	RO	0x0	Reserved field.
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. 2.0. The value of this register must not be programmed to point to the address offset of this register (i.e., 0xF8) or the ECFGDATA register (i.e., 0xFC). Violation of this rule produces undefined results. Also, the value of this register must not be programmed to point to the address offset of the global address space registers (GASAADDR and GASADATA). Violation of this rule produces undefined results.

Bit Field	Field Name	Type	Default Value	Description
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. 2.0. The value of this register must not be programmed to point to the address offset of this register (i.e., 0xF8) or the ECFGDATA register (i.e., 0xFC). Violation of this rule produces undefined results. Also, the value of this register must not be programmed to point to the address offset of the global address space registers (GASAADDR and GASADATA). Violation of this rule produces undefined results.
31:12	Reserved	RO	0x0	Reserved field.

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. SMBus reads of this field return a value of zero and SMBus writes have no effect.

Advanced Error Reporting (AER) Enhanced Capability

AERCAP - AER Capabilities (0x100)

Bit Field	Field Name	Type	Default Value	Description
15:0	CAPID	RO	0x1	**Capability ID.** The value of 0x1 indicates an advanced error reporting capability structure.
19:16	CAPVER	RO	0x1	**Capability Version.** The value of 0x1. indicates compatibility with version 1 of the specification.
31:20	NXTPTR	RWL	Refer to section Capability Structures on page 14-3.	**Next Pointer.** Next capability pointer. The value of 0x0 terminates the list.

AERUES - AER Uncorrectable Error Status (0x104)

Bit Field	Field Name	Type	Default Value	Description
0	UDEF	RW1C	0x0 Sticky	**Undefined.** This bit is no longer used in this version of the specification.

Bit Field	Field Name	Type	Default Value	Description
3:1	Reserved	RO	0x0	Reserved field.
4	DLPERR	RW1C	0x0 Sticky	**Data Link Protocol Error Status.** This bit is set when a data link layer protocol error is detected.
5	SDOENERR	RW1C	0x0 Sticky	**Surprise Down Error Status.** This bit is set when a surprise down error is detected and the SDERR bit in the PCIELCAP register is set.
11:6	Reserved	RO	0x0	Reserved field.
12	POISONED	RW1C	0x0 Sticky	**Poisoned TLP Status.** This bit is set when a poisoned TLP is detected.
13	FCPERR	RW1C	0x0 Sticky	**Flow Control Protocol Error Status.** This bit is set when a flow control protocol error is detected.
14	COMPTO	RO	0x0	**Completion Timeout Status.** A switch port does not initiate non-posted requests on its own behalf. Therefore, this field is hardwired to zero.
15	CABORT	RO	0x0	**Completer Abort Status.** This bit is never set as the switch never responds to a non-posted request with a completer abort, except for ACS violations. For this exception case, the error is an ACS violation and is not logged as a completer abort error.
16	UECOMP	RW1C	0x0 Sticky	**Unexpected Completion Status.** This bit is set when an unexpected completion is detected.
17	RCVOVR	RW1C	0x0 Sticky	**Receiver Overflow Status.** This bit is set when a receiver overflow is detected.
18	MALFORMED	RW1C	0x0 Sticky	**Malformed TLP Status.** This bit is set when a malformed TLP is detected.
19	ECRC	RW1C	0x0 Sticky	**ECRC Status.** This bit is set when an ECRC error is detected.
20	UR	RW1C	0x0 Sticky	**UR Status.** This bit is set when an unsupported request is detected.
21	ACSV	RW1C	0x0 Sticky	**ACS Violation Status**. This bit is set when an ACS violation is detected on the port.
22	UIE	RW1C	0x0 Sticky	**Uncorrectable Internal Error Status.** This bit is set when an uncorrectable internal error associated with the port is detected.
23	MCBLKTLP	RW1C	0x0 Sticky	**MC Blocked TLP Status.** This bit is set when a multicast TLP is blocked by the ingress port in response to the setting of the MC_Block_All and MC_Block_Untranslated bits in the port's multicast extended capability structure.
31:24	Reserved	RO	0x0	Reserved field.

AERUEM - AER Uncorrectable Error Mask (0x108)

Bit Field	Field Name	Type	Default Value	Description
0	UDEF	RW	0x0 Sticky	**Undefined.** This bit is no longer used in this version of the specification.
3:1	Reserved	RO	0x0	Reserved field.
4	DLPERR	RW	0x0 Sticky	**Data Link Protocol Error Mask.** When this bit is set, the corresponding bit in the AERUES register is masked. When a bit is masked in the AERUES register, the corresponding event is not logged in the advanced capability structure, the First Error Pointer field (FEPTR) in the AERCTL register is not updated, and an error is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERUES register.
5	SDOENERR	RW	0x0 Sticky	**Surprise Down Error Mask.** When this bit is set, the corresponding bit in the AERUES register is masked. When a bit is masked in the AERUES register, the corresponding event is not logged in the advanced capability structure, the First Error Pointer field (FEPTR) in the AERCTL register is not updated, and an error is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERUES register.
11:6	Reserved	RO	0x0	Reserved field.
12	POISONED	RW	0x0 Sticky	**Poisoned TLP Mask.** When this bit is set, the corresponding bit in the AERUES register is masked. When a bit is masked in the AERUES register, the corresponding event is not logged in the advanced capability structure, the First Error Pointer field (FEPTR) in the AERCTL register is not updated, and an error is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERUES register.
13	FCPERR	RW	0x0 Sticky	**Flow Control Protocol Error Mask.** When this bit is set, the corresponding bit in the AERUES register is masked. When a bit is masked in the AERUES register, the corresponding event is not logged in the advanced capability structure, the First Error Pointer field (FEPTR) in the AERCTL register is not updated, and an error is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERUES register.
14	COMPTO	RO	0x0	**Completion Timeout Mask**. A switch port does not initiate non-posted requests on its own behalf. Therefore, this field is hardwired to zero.
15	CABORT	RO	0x0	**Completer Abort Mask.** The switch never responds to a non-posted request with a completer abort, except for ACS violations.

Bit Field	Field Name	Type	Default Value	Description
16	UECOMP	RW	0x0 Sticky	**Unexpected Completion Mask.** When this bit is set, the corresponding bit in the AERUES register is masked. When a bit is masked in the AERUES register, the corresponding event is not logged in the advanced capability structure, the First Error Pointer field (FEPTR) in the AERCTL register is not updated, and an error is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERUES register.
17	RCVOVR	RW	0x0 Sticky	**Receiver Overflow Mask.** When this bit is set, the corresponding bit in the AERUES register is masked. When a bit is masked in the AERUES register, the corresponding event is not logged in the advanced capability structure, the First Error Pointer field (FEPTR) in the AERCTL register is not updated, and an error is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERUES register.
18	MALFORMED	RW	0x0 Sticky	**Malformed TLP Mask.** When this bit is set, the corresponding bit in the AERUES register is masked. When a bit is masked in the AERUES register, the corresponding event is not logged in the advanced capability structure, the First Error Pointer field (FEPTR) in the AERCTL register is not updated, and an error is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERUES register.
19	ECRC	RW	0x0 Sticky	**ECRC Mask.** When this bit is set, the corresponding bit in the AERUES register is masked. When a bit is masked in the AERUES register, the corresponding event is not logged in the advanced capability structure, the First Error Pointer field (FEPTR) in the AERCTL register is not updated, and an error is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERUES register.
20	UR	RW	0x0 Sticky	**UR Mask.** When this bit is set, the corresponding bit in the AERUES register is masked. When a bit is masked in the AERUES register, the corresponding event is not logged in the advanced capability structure, the First Error Pointer field (FEPTR) in the AERCTL register is not updated, and an error is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERUES register.
21	ACSV	RW	0x0 Sticky	**ACS Violation Mask.** When this bit is set, the corresponding bit in the AERUES register is masked. When a bit is masked in the AERUES register, the corresponding event is not logged in the advanced capability structure, the First Error Pointer field (FEPTR) in the AERCTL register is not updated, and an error is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERUES register.

Bit Field	Field Name	Type	Default Value	Description
22	UIE	RW	0x1 Sticky	**Uncorrectable Internal Error Mask.** When this bit is set, the corresponding bit in the AERUES register is masked. When a bit is masked in the AERUES register, the corresponding event is not logged in the advanced capability structure, the First Error Pointer field (FEPTR) in the AERCTL register is not updated, and an error is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERUES register. When the Internal Error Reporting Enable (IERROREN) bit is cleared in the Internal Error Reporting Control (IERRORCTL) register, this field becomes read-only with a value of zero.
23	MCBLKTLP	RW	0x0 Sticky	**MC Blocked TLP Mask.** When this bit is set, the corresponding bit in the AERUES register is masked. When a bit is masked in the AERUES register, the corresponding event is not logged in the advanced capability structure, the First Error Pointer field (FEPTR) in the AERCTL register is not updated, and an error is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERUES register. When the Disable Multicast Error Reporting (DMCER) bit is set in the Switch Control (SWCTL) register, this field becomes read-only with a value of zero.
31:24	Reserved	RO	0x0	Reserved field.

AERUESV - AER Uncorrectable Error Severity (0x10C)

Bit Field	Field Name	Type	Default Value	Description
0	UDEF	RW	0x0 Sticky	**Undefined.** This bit is no longer used in this version of the specification.
3:1	Reserved	RO	0x0	Reserved field.
4	DLPERR	RW	0x1 Sticky	**Data Link Protocol Error Severity.** This bit controls the severity of the reported error. If this bit is set, the event is reported as a fatal error. When this bit is cleared, the event is reported as a non-fatal error.
5	SDOENERR	RW	0x1 Sticky	**Surprise Down Error Severity**. This bit controls the severity of the reported error. If this bit is set, the event is reported as a fatal error. When this bit is cleared, the event is reported as a non-fatal error.
11:6	Reserved	RO	0x0	Reserved field.
12	POISONED	RW	0x0 Sticky	**Poisoned TLP Status Severity**. This bit controls the severity of the reported error. If this bit is set, the event is reported as a fatal error. When this bit is cleared, the event is reported as a non-fatal error.
13	FCPERR	RW	0x1 Sticky	**Flow Control Protocol Error Severity.** This bit controls the severity of the reported error. If this bit is set, the event is reported as a fatal error. When this bit is cleared, the event is reported as a non-fatal error.

Bit Field	Field Name	Type	Default Value	Description
14	COMPTO	RO	0x0	**Completion Timeout Severity.** A switch port does not initiate non-posted requests on its own behalf. Therefore, this field is hardwired to zero.
15	CABORT	RO	0x0	**Completer Abort Severity.** Tthe switch never responds to a non-posted request with a completer abort, except for ACS violations.
16	UECOMP	RW	0x0 Sticky	**Unexpected Completion Severity**. This bit controls the severity of the reported error. If this bit is set, the event is reported as a fatal error. When this bit is cleared, the event is reported as a non-fatal error.
17	RCVOVR	RW	0x1 Sticky	**Receiver Overflow Severity.** This bit controls the severity of the reported error. If this bit is set, the event is reported as a fatal error. When this bit is cleared, the event is reported as a non-fatal error.
18	MALFORMED	RW	0x1 Sticky	**Malformed TLP Severity.** This bit controls the severity of the reported error. If this bit is set, the event is reported as a fatal error. When this bit is cleared, the event is reported as a non-fatal error.
19	ECRC	RW	0x0 Sticky	**ECRC Severity.** This bit controls the severity of the reported error. If this bit is set, the event is reported as a fatal error. When this bit is cleared, the event is reported as a non-fatal error.
20	UR	RW	0x0 Sticky	**UR Severity.** This bit controls the severity of the reported error. If this bit is set, the event is reported as a fatal error. When this bit is cleared, the event is reported as a non-fatal error.
21	ACSV	RW	0x0 Sticky	**ACS Violation Severity.** This bit controls the severity of the reported error. If this bit is set, the event is reported as a fatal error. When this bit is cleared, the event is reported as a non-fatal error.
22	UIE	RW	0x1 Sticky	**Uncorrectable Internal Error Severity.** This bit controls the severity of the reported error. If this bit is set, the event is reported as a fatal error. When this bit is cleared, the event is reported as a non-fatal error. When the Internal Error Reporting Enable (IERROREN) bit is cleared in the Internal Error Reporting Control (IERRORCTL) register, this field becomes read-only with a value of one.
23	MCBLKTLP	RW	0x0 Sticky	**MC Blocked TLP Severity.** This bit controls the severity of the reported error. If this bit is set, the event is reported as a fatal error. When this bit is cleared, the event is reported as a non-fatal error. When the Disable Multicast Error Reporting (DMCER) bit is cleared in the Switch Control (SWCTL) register, this field becomes read-only with a value of zero.
31:24	Reserved	RO	0x0	Reserved field.

AERCES - AER Correctable Error Status (0x110)

Bit Field	Field Name	Type	Default Value	Description
0	RCVERR	RW1C	0x0 Sticky	**Receiver Error Status.** This bit is set when the physical layer detects a receiver error.
5:1	Reserved	RO	0x0	Reserved field.

Bit Field	Field Name	Type	Default Value	Description
6	BADTLP	RW1C	0x0 Sticky	**Bad TLP Status.** This bit is set when a bad TLP is detected.
7	BADDLLP	RW1C	0x0 Sticky	**Bad DLLP Status.** This bit is set when a bad DLLP is detected.
8	RPLYROVR	RW1C	0x0 Sticky	**Replay Number Rollover Status.** This bit is set when a replay number rollover has occurred indicating that the data link layer has abandoned replays and has requested that the link be retrained.
11:9	Reserved	RO	0x0	Reserved field.
12	RPLYTO	RW1C	0x0 Sticky	**Replay Timer Timeout Status.** This bit is set when the replay timer in the data link layer times out.
13	ADVISORYNF	RW1C	0x0 Sticky	**Advisory Non-Fatal Error Status.** This bit is set when an advisory non-fatal error is detected as described in Section 6.2.3.2.4 of the PCI Express 2.0 specification.
14	CIE	RW1C	0x0 Sticky	**Correctable Internal Error Status.** This bit is set whenever an correctable internal error associated with the port is detected.
15	HLO	RW1C	0x0 Sticky	**Header Log Overflow Status.** This bit is set when an error that requires packet-header logging occurs but the packet header cannot be logged by the port's AER Header Log registers (AERHL[1:4]DW). A packet's header cannot be logged in the AER Header Log registers when an error occurs while the First Error Pointer (FEPTR field in the AERCTL register) is valid. The First Error Pointer is valid when it points to a set bit in the AERUES register (i.e., indicating the occurrence of a prior uncorrectable error which has not been cleared by software).
31:16	Reserved	RO	0x0	Reserved field.

AERCEM - AER Correctable Error Mask (0x114)

Bit Field	Field Name	Type	Default Value	Description
0	RCVERR	RW	0x0 Sticky	**Receiver Error Mask.** When this bit is set, the corresponding bit in the AERCES register is masked. When a bit is masked in the AERCES register, the corresponding event is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERCES register.
5:1	Reserved	RO	0x0	Reserved field.
6	BADTLP	RW	0x0 Sticky	**Bad TLP Mask.** When this bit is set, the corresponding bit in the AERCES register is masked. When a bit is masked in the AERCES register, the corresponding event is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERCES register.

Bit Field	Field Name	Type	Default Value	Description
7	BADDLLP	RW	0x0 Sticky	**Bad DLLP Mask.** When this bit is set, the corresponding bit in the AERCES register is masked. When a bit is masked in the AERCES register, the corresponding event is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERCES register.
8	RPLYROVR	RW	0x0 Sticky	**Replay Number Rollover Mask**. When this bit is set, the corresponding bit in the AERCES register is masked. When a bit is masked in the AERCES register, the corresponding event is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERCES register.
11:9	Reserved	RO	0x0	Reserved field.
12	RPLYTO	RW	0x0 Sticky	**Replay Timer Timeout Mask.** When this bit is set, the corresponding bit in the AERCES register is masked. When a bit is masked in the AERCES register, the corresponding event is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERCES register.
13	ADVISORYNF	RW	0x1 Sticky	**Advisory Non-Fatal Error Mask.** When this bit is set, the corresponding bit in the AERCES register is masked. When a bit is masked in the AERCES register, the corresponding event is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERCES register.
14	CIE	RW	0x1 Sticky	**Correctable Internal Error Mask.** When this bit is set, the corresponding bit in the AERCES register is masked. When a bit is masked in the AERCES register, the corresponding event is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERCES register. When the Internal Error Reporting Enable (IERROREN) bit is cleared in the Internal Error Reporting Control (IERRORCTL) register, this field becomes read-only with a value of zero.
15	HLO	RW	0x1 Sticky	**Header Log Overflow Mask.** When this bit is set, the corresponding bit in the AERCES register is masked. When a bit is masked in the AERCES register, the corresponding event is not reported to the root complex. This bit does not affect the state of the corresponding bit in the AERCES register. When the Internal Error Reporting Enable (IERROREN) bit is cleared in the Internal Error Reporting Control (IERRORCTL) register, this field becomes read-only with a value of zero.
31:16	Reserved	RO	0x0	Reserved field.

AERCTL - AER Control (0x118)

Bit Field	Field Name	Type	Default Value	Description
4:0	FEPTR	RO	0x0 Sticky	**First Error Pointer.** This field contains a pointer to the bit in the AERUES register that resulted in the first reported error. This field is valid only when the bit in the AERUES register pointed to by this field is set.
5	ECRCGC	RWL	0x1	**ECRC Generation Capable.** This bit indicates if the Function is capable of generating ECRC.
6	ECRCGE	RW	0x0 Sticky	**ECRC Generation Enable.** When this bit is set, ECRC generation is enabled for the Function.
7	ECRCCC	RWL	0x1	**ECRC Check Capable.** This bit indicates if the Function is capable of checking ECRC.
8	ECRCCE	RW	0x0 Sticky	**ECRC Check Enable.** When this bit is set, ECRC checking is enabled for the Function.
9	MHRC	RO	0x0	**Multiple Header Recording Capable.** The PES48T12G2 ports do not support recording of multiple packet headers.
10	MHRE	RO	0x0 Sticky	**Multiple Header Recording Enable.** The PES48T12G2 ports do not support recording of multiple packet headers. As a result, this bit is hardwired to 0x0.
31:11	Reserved	RO	0x0	Reserved field.

AERHL1DW - AER Header Log 1st Doubleword (0x11C)

Bit Field	Field Name	Type	Default Value	Description
31:0	HL	RWL	0x0 Sticky	**Header Log.** This field contains the 1st doubleword of the TLP header that resulted in the first reported uncorrectable error.

AERHL2DW - AER Header Log 2nd Doubleword (0x120)

Bit Field	Field Name	Type	Default Value	Description
31:0	HL	RWL	0x0 Sticky	**Header Log.** This field contains the 2nd doubleword of the TLP header that resulted in the first reported uncorrectable error.

AERHL3DW - AER Header Log 3rd Doubleword (0x124)

Bit Field	Field Name	Type	Default Value	Description
31:0	HL	RWL	0x0 Sticky	**Header Log.** This field contains the 3rd doubleword of the TLP header that resulted in the first reported uncorrectable error.

AERHL4DW - AER Header Log 4th Doubleword (0x128)

Bit Field	Field Name	Type	Default Value	Description
31:0	HL	RWL	0x0 Sticky	**Header Log.** This field contains the 4th doubleword of the TLP header that resulted in the first reported uncorrectable error.

Device Serial Number Enhanced Capability

SNUMCAP - Serial Number Capabilities (0x180)

Bit Field	Field Name	Type	Default Value	Description
15:0	CAPID	RO	0x3	**Capability ID.** The value of 0x3 indicates a device serial number capability structure.
19:16	CAPVER	RO	0x1	**Capability Version.** The value of 0x1 indicates compatibility with version 1 of the specification.
31:20	NXTPTR	RWL	Refer to section Capability Structures on page 14-3.	**Next Pointer.** Next capability pointer. The value of 0x0 terminates the list.

SNUMLDW - Serial Number Lower Doubleword (0x184)

Bit Field	Field Name	Type	Default Value	Description
31:0	SNUM	RWL	0x0	**Lower 32-bits of Device Serial Number.** This field contains the lower 32-bits of the IEEE defined 64-bit extended unique identifier (EUI-64) assigned to the device.

SNUMUDW - Serial Number Upper Doubleword (0x188)

Bit Field	Field Name	Type	Default Value	Description
31:0	SNUM	RWL	0x0	**Upper 32-bits of Device Serial Number.** This field contains the upper 32-bits of the IEEE defined 64-bit extended unique identifier (EUI-64) assigned to the device.

PCI Express Virtual Channel Capability

PCIEVCECAP - PCI Express VC Enhanced Capability Header (0x200)

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	RWL	Refer to section Capability Structures on page 14-3.	**Next Pointer.** Next capability pointer. The value of 0x0 terminates the list.

PVCCAP1- Port VC Capability 1 (0x204)

Bit Field	Field Name	Type	Default Value	Description
2:0	EVCCNT	RWL	0x0	**Extended VC Count.** A value 0x0 indicates that only the default VC (VC0) is implemented.
3	Reserved	RO	0x0	Reserved field.
6:4	LPEVCCNT	RWL	0x0	**Low Priority Extended VC Count.** Not applicable (only the default VC 0 is implemented).
7	Reserved	RO	0x0	Reserved field.
9:8	REFCLK	RO	0x0	**Reference Clock.** Not supported (i.e., Time-based WRR Port Arbitration is not implemented).
11:10	PATBLSIZ	RWL	0x2	**Port Arbitration Table Entry Size**. This field indicates the size of the port arbitration table in the device. It is set to 0x2 to indicate a table with 4-bit entries. 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	Reserved field.

PVCCAP2- Port VC Capability 2 (0x208)

Bit Field	Field Name	Type	Default Value	Description
7:0	VCARBCAP	RO	0x0	**VC Arbitration Capability.** Not applicable (only the default VC 0 is implemented).
23:8	Reserved	RO	0x0	Reserved field.

Bit Field	Field Name	Type	Default Value	Description
31:24	VCATBLOFF	RO	0x0	**VC Arbitration Table Offset.** Not applicable (only the default VC 0 is implemented).

PVCCTL - Port VC Control (0x20C)

Bit Field	Field Name	Type	Default Value	Description
0	LVCAT	RW	0x0	**Load VC Arbitration Table.** Not applicable.
3:1	VCARBSEL	RW	0x0	**VC Arbitration Select.** Not applicable (only the default VC 0 is implemented).
15:4	Reserved	RO	0x0	Reserved field.

PVCSTS - Port VC Status (0x20E)

Bit Field	Field Name	Type	Default Value	Description
0	VCATS	RO	0x0	**VC Arbitration Table Status.** Not applicable (only the default VC 0 is implemented).
15:1	Reserved	RO	0x0	Reserved field.

VCR0CAP- VC Resource 0 Capability (0x210)

Bit Field	Field Name	Type	Default Value	Description
7:0	PARBC	RWL	0x1	**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. bit 0 - Hardware Fixed bits 1 through 7 - reserved
14:8	Reserved	RO	0x0	Reserved field.
15	RJST	RO	0x0	**Reject Snoop Transactions.** No 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	Reserved field.
31:24	PATBLOFF	RWL	0x4	**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 (0x214)

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.
15:8	Reserved	RO	0x0	Reserved field.
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 (as indicated by a set bit in the Port Arbitration Select field). 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. If the port arbitration scheme selected in this field is not one of the supported advertised schemes, the operation of the device is undefined.
23:20	Reserved	RO	0x0	Reserved field.
26:24	VCID	RO	0x0	**VC ID.** This field assigns a VC ID to the VC resource. For VC0, this field is always hardwired to zero.
30:27	Reserved	RO	0x0	Reserved field.
31	VCEN	RO	0x1	**VC Enable.** This field, when set, enables a virtual channel. For VC0, this field is hardwired to 0x1 (enabled).

VCR0STS - VC Resource 0 Status (0x218)

Bit Field	Field Name	Type	Default Value	Description
15:0	Reserved	RO	0x0	Reserved field.
16	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.

Bit Field	Field Name	Type	Default Value	Description
17	VCNEG	RO	0x0	**VC Negotiation Pending.** This bit is not applicable for VC0 and is therefore hardwired to 0x0.
31:18	Reserved	RO	0x0	Reserved field.

VCR0TBL0 - VC Resource 0 Port Arbitration Table Entry 0 (0x240)

Bit Field	Field Name	Type	Default Value	Description
3:0	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_0) Port 0 0x1 - (port_1) Port 1 0x2 - (port_2) Port 2 0x3 - (port_3) Port 3 0x4 - (port_4) Port 4 0x5 - (port_5) Port 5 0x6 - (port_6) Port 6 0x7 - (port_7) Port 7 0x8 - (port_8) Port 8 0x9 - (port_9) Port 9 0xC - (port_12) Port 12 0xD - (port_13) Port 13
7:4	PHASE1	RW	0x0	**Phase 1.** This field contains the port ID for the corresponding port arbitration period.
11:8	PHASE2	RW	0x0	**Phase 2.** This field contains the port ID for the corresponding port arbitration period.
15:12	PHASE3	RW	0x0	**Phase 3.** This field contains the port ID for the corresponding port arbitration period.
19:16	PHASE4	RW	0x0	**Phase 4.** This field contains the port ID for the corresponding port arbitration period.
23:20	PHASE5	RW	0x0	**Phase 5.** This field contains the port ID for the corresponding port arbitration period.
27:24	PHASE6	RW	0x0	**Phase 6.** This field contains the port ID for the corresponding port arbitration period.
31:28	PHASE7	RW	0x0	**Phase 7.** This field contains the port ID for the corresponding port arbitration period.

VCR0TBL1 - VC Resource 0 Port Arbitration Table Entry 1 (0x244)

Bit Field	Field Name	Type	Default Value	Description
3:0	PHASE8	RW	0x0	**Phase 8.** This field contains the port ID for the corresponding port arbitration period.
7:4	PHASE9	RW	0x0	**Phase 9.** This field contains the port ID for the corresponding port arbitration period.
11:8	PHASE10	RW	0x0	**Phase 10.** This field contains the port ID for the corresponding port arbitration period.
15:12	PHASE11	RW	0x0	**Phase 11.** This field contains the port ID for the corresponding port arbitration period.
19:16	PHASE12	RW	0x0	**Phase 12.** This field contains the port ID for the corresponding port arbitration period.
23:20	PHASE13	RW	0x0	**Phase 13.** This field contains the port ID for the corresponding port arbitration period.
27:24	PHASE14	RW	0x0	**Phase 14.** This field contains the port ID for the corresponding port arbitration period.
31:28	PHASE15	RW	0x0	**Phase 15.** This field contains the port ID for the corresponding port arbitration period.

VCR0TBL2 - VC Resource 0 Port Arbitration Table Entry 2 (0x248)

Bit Field	Field Name	Type	Default Value	Description
3:0	PHASE16	RW	0x0	**Phase 16.** This field contains the port ID for the corresponding port arbitration period.
7:4	PHASE17	RW	0x0	**Phase 17.** This field contains the port ID for the corresponding port arbitration period.
11:8	PHASE18	RW	0x0	**Phase 18.** This field contains the port ID for the corresponding port arbitration period.
15:12	PHASE19	RW	0x0	**Phase 19.** This field contains the port ID for the corresponding port arbitration period.
19:16	PHASE20	RW	0x0	**Phase 20.** This field contains the port ID for the corresponding port arbitration period.
23:20	PHASE21	RW	0x0	**Phase 21.** This field contains the port ID for the corresponding port arbitration period.
27:24	PHASE22	RW	0x0	**Phase 22.** This field contains the port ID for the corresponding port arbitration period.
31:28	PHASE23	RW	0x0	**Phase 23.** This field contains the port ID for the corresponding port arbitration period.

VCR0TBL3 - VC Resource 0 Port Arbitration Table Entry 3 (0x24C)

Bit Field	Field Name	Type	Default Value	Description
3:0	PHASE24	RW	0x0	**Phase 24.** This field contains the port ID for the corresponding port arbitration period.
7:4	PHASE25	RW	0x0	**Phase 25.** This field contains the port ID for the corresponding port arbitration period.
11:8	PHASE26	RW	0x0	**Phase 26.** This field contains the port ID for the corresponding port arbitration period.
15:12	PHASE27	RW	0x0	**Phase 27.** This field contains the port ID for the corresponding port arbitration period.
19:16	PHASE28	RW	0x0	**Phase 28.** This field contains the port ID for the corresponding port arbitration period.
23:20	PHASE29	RW	0x0	**Phase 29.** This field contains the port ID for the corresponding port arbitration period.
27:24	PHASE30	RW	0x0	**Phase 30.** This field contains the port ID for the corresponding port arbitration period.
31:28	PHASE31	RW	0x0	**Phase 31.** This field contains the port ID for the corresponding port arbitration period.

Power Budgeting Enhanced Capability

PWRBCAP - Power Budgeting Capabilities (0x280)

Bit Field	Field Name	Type	Default Value	Description
15:0	CAPID	RWL	0x0	**Capability ID.** The value of 0x4 indicates a power budgeting capability structure. If the power budgeting capability is used, then this field should be initialized with data from a serial EEPROM.
19:16	CAPVER	RWL	0x0	**Capability Version.** The value of 0x1 indicates compatibility with the PCI Express 2.0 specification. If the power budgeting capability is used, then this field should be initialized with data from a serial EEPROM.
31:20	NXTPTR	RWL	Refer to section Capability Structures on page 14-3.	**Next Pointer.** Next capability pointer. The value of 0x0 terminates the list.

PWRBDSEL - Power Budgeting Data Select (0x284)

Bit Field	Field Name	Type	Default Value	Description
7:0	DVSEL	RW	0x0	**Data Value Select.** This field selects the Power Budgeting Data Value (PWRBDVx) register whose contents are reported in the Data (DATA) field of the Power Budgeting Data (PWRBD) register. Setting this field to a value greater than 7, causes zero to be returned in the DATA field of the PWRBD register.
31:8	Reserved	RO	0x0	Reserved field.

PWRBD - Power Budgeting Data (0x288)

Bit Field	Field Name	Type	Default Value	Description
31:0	DATA	RO	0x0	**Data.** If the Data Value Select (DVSEL) field in the Power Budgeting Data Select register contains a value of zero through 7, then this field returns the contents of the corresponding Power Budgeting Data Value (PWRBDVx) register. Otherwise, this field contains a value of zero.

PWRBPBC - Power Budgeting Power Budget Capability (0x28C)

Bit Field	Field Name	Type	Default Value	Description
0	SA	RWL	0x0	**System Allocated.** When this bit is set, it indicates that the power budget for the device is included within the system power budget and that reported power data for this device should be ignored. If the power budgeting capability is used, then this field should be initialized with data from a serial EEPROM.
31:1	Reserved	RO	0x0	Reserved field.

PWRBDV[7:0] - Power Budgeting Data Value [7:0] (0x300 - 0x31C)

Bit Field	Field Name	Type	Default Value	Description
31:0	DV	RW	Undefined Sticky	**Data Value.** This 32-bit field is used to hold power budget data in the format described in Section 7.15.3 in the PCI Express 2.0 Base Specification. This field may be read and written when the Power Budgeting Data Value Unlock (PWRBDVUL) bit is set in the Switch Control (SWCTL) register. When the PWRBDVUL bit is cleared, this register is read-only and writes are ignored. If the power budgeting capability is used, then this field should be initialized with data from a serial EEPROM.

ACS Extended Capability

ACSECAPH - ACS Extended Capability Header (0x320)

Bit Field	Field Name	Type	Default Value	Description
15:0	CAPID	RO	0xD	**Capability ID.** The value of 0xD indicates an ACS extended capability structure.
19:16	CAPVER	RO	0x1	**Capability Version.** The value of 0x1 indicates compatibility with the PCI Express 2.0 specification.
31:20	NXTPTR	RWL	Refer to section Capability Structures on page 14-3.	**Next Pointer.** Next capability pointer. The value of 0x0 terminates the list.

ACSCAP - ACS Capability Register (0x324)

Bit Field	Field Name	Type	Default Value	Description
0	V	Upstream Port: RO Downstream Port: RWL	Upstream Port: 0x0 Downstream Port: 0x1	**ACS Source Validation.** If set, indicates the port implements ACS Source Validation.
1	B	Upstream Port: RO Downstream Port: RWL	Upstream Port: 0x0 Downstream Port: 0x1	**ACS Translation Blocking.** If set, indicates the port implements ACS Translation Blocking.
2	R	Upstream Port: RO Downstream Port: RWL	Upstream Port: 0x0 Downstream Port: 0x1	**ACS P2P Request Redirect.** If set, indicates the port implements ACS Peer-to-Peer Request Redirect.
3	C	Upstream Port: RO Downstream Port: RWL	Upstream Port: 0x0 Downstream Port: 0x1	**ACS P2P Completion Redirect.** If set, indicates the port implements ACS Peer-to-Peer Completion Redirect.

Bit Field	Field Name	Type	Default Value	Description
4	U	Upstream Port: RO Down-stream Port: RWL	Upstream Port: 0x0 Downstream Port: 0x1	**ACS Upstream Forwarding.** If set, indicates the port implements ACS Upstream Forwarding.
5	E	Upstream Port: RO Down-stream Port: RWL	Upstream Port: 0x0 Downstream Port: 0x1	**ACS P2P Egress Control.** If set, indicates the port implements ACS Peer-to-Peer Egress Control.
6	T	Upstream Port: RO Down-stream Port: RWL	Upstream Port: 0x0 Downstream Port: 0x1	**ACS Direct Translated P2P.** If set, indicates the port implements ACS Direct Translated Peer-to-Peer.
7	Reserved	RO	0x0	Reserved field.
15:8	ECVS	Upstream Port: RO Down-stream Port: RWL	Upstream Port: 0x0 Downstream Port: 0x10	**Egress Control Vector Size.** Indicates the number of applicable bits in the ACS Egress Control Vector register. The value of 0x10 indicates that egress control may be done with up to 8 ports. The value of this field is undefined if the ACS P2P Egress Control bit in this register is set to 0x0.

ACSCTL - ACS Control Register (0x326)

Bit Field	Field Name	Type	Default Value	Description
0	V	Upstream Port: RO Downstream Port: RW	0x0	**ACS Source Validation Enable.** When set, the port performs ACS Source Validation. NOTE: This field remains read-write (RW) for downstream ports, even if the corresponding bit in the ACSCAP register is cleared.
1	B	Upstream Port: RO Downstream Port: RW	0x0	**ACS Translation Blocking Enable**. When set, the port performs ACS Translation Blocking. NOTE: This field remains read-write (RW) for downstream ports, even if the corresponding bit in the ACSCAP register is cleared.
2	R	Upstream Port: RO Downstream Port: RW	0x0	**ACS P2P Request Redirect Enable.** When set, the port performs ACS Peer-to-Peer Request Redirect. NOTE: This field remains read-write (RW) for downstream ports, even if the corresponding bit in the ACSCAP register is cleared.

Bit Field	Field Name	Type	Default Value	Description
3	C	Upstream Port: RO Downstream Port: RW	0x0	**ACS P2P Completion Redirect Enable.** When set, the port performs ACS Peer-to-Peer Completion Redirect. NOTE: This field remains read-write (RW) for downstream ports, even if the corresponding bit in the ACSCAP register is cleared.
4	U	Upstream Port: RO Downstream Port: RW	0x0	**ACS Upstream Forwarding Enable.** When set, the port performs ACS Upstream Forwarding. NOTE: This field remains read-write (RW) for downstream ports, even if the corresponding bit in the ACSCAP register is cleared.
5	E	Upstream Port: RO Downstream Port: RW	0x0	**ACS P2P Egress Control Enable.** When set, the port performs ACS Peer-to-Peer Egress Control. NOTE: This field remains read-write (RW) for downstream ports, even if the corresponding bit in the ACSCAP register is cleared.
6	T	Upstream Port: RO Downstream Port: RW	0x0	**ACS Direct Translated P2P Enable.** When set, the port performs ACS Direct Translated Peer-to-Peer control.
15:7	Reserved	RO	0x0	Reserved field.

ACSECV - ACS Egress Control Vector (0x328)

Bit Field	Field Name	Type	Default Value	Description
15:0	ECV	See Description	0x0	**Egress Control Vector.** This field is used to configure ACS peer-to-peer egress control. The value in this field is only valid when ACS peer-to-peer egress control is enabled in the ACSCTL register. Each bit in this register corresponds to a switch port. Bit[0] corresponds to port 0, bit[1] corresponds to port 1, and so on. When a bit is set, peer-to-peer requests targeting the associated port or function are blocked or redirected. Refer to Section 6.12.3 of PCI Express 2.0 specification for further details. The bit corresponding to this port number is read-only and hard-wired to 0x0. For example, bit[0] is read-only for port 0, bit[1] is read-only for port 1, and so on. When the port operates in merged mode, the bit corresponding to this even numbered port (i.e., bit[N]) and its next in sequence bit (i.e., bit[N+1]) are read-only with a value of 0x0.

Multicast Extended Capability

MCCAPH - Multicast Enhanced Capability Header (0x330)

Bit Field	Field Name	Type	Default Value	Description
15:0	CAPID	RO	0x12	**Capability ID.** The value of 0x12 indicates a multicast capability structure.
19:16	CAPVER	RO	0x1	**Capability Version.** The value of 0x1 indicates compatibility with the PCI-SIG Multicast ECN.
31:20	NXTPTR	RWL	Refer to section Capability Structures on page 14-3.	**Next Pointer.** Next capability pointer. The value of 0x0 terminates the list.

MCCAP - Multicast Capability (0x334)

Bit Field	Field Name	Type	Default Value	Description
5:0	MAXGROUP	RWL	0x1F	**Max Multicast Groups**. This field indicates the default number of multicast groups supported by the switch partition, which is 32. The maximum number of supported groups is 64, and this field may be re-programmed during initial switch configuration (e.g., via EEPROM) to 0x3F to enable support for 64 multicast groups. The number of supported groups is equal to the value in this field plus one.
14:6	Reserved	RO	0x0	Reserved field.
15	ECRCREG	RWL	0x1	**ECRC Regeneration Supported.** This bit is set to indicate that the switch supports multicast ECRC regeneration.

MCCTL- Multicast Control (0x336)

Bit Field	Field Name	Type	Default Value	Description
5:0	NUMGROUP	RW	0x0	**Number of Multicast Groups.** When the Multicast Enabler (MEN) bit is set, this field indicates the number of multicast groups that are enabled. The number of groups enabled is equal to the value in this field plus one. The behavior is undefined when the value in this field exceeds the value of the MAXGROUP field in the MCCAP register.
14:6	Reserved	RO	0x0	Reserved field.
15	MEN	RW	0x0	**Multicast Enable.** When this bit is set, multicast is enabled in the corresponding switch.

MCBARL- Multicast Base Address Low (0x338)

Bit Field	Field Name	Type	Default Value	Description
5:0	INDEXPOS	RW	0x0	**Index Position.** When multicast is enabled, this field specifies the least significant bit of the multicast group number within a TLP address. The behavior is undefined when multicast is enabled and this field is less than 12.
11:6	Reserved	RO	0x0	Reserved field.
31:12	MCBARL	RW	0x0	**Multicast BAR Low.** This field specifies the lower 20-bits (i.e., bits 12 through 31) of the multicast BAR. The behavior is undefined if bits in this field corresponding to address bits that contain the multicast group number or those less than the multicast index position (i.e., INDEXPOS) are non-zero.

MCBARH- Multicast Base Address High (0x33C)

Bit Field	Field Name	Type	Default Value	Description
31:0	MCBARH	RW	0x0	**Multicast BAR High.** This field specifies the upper 32-bits (i.e., bits 32 through 63) of the multicast BAR. The behavior is undefined if bits in this field corresponding to address bits that contain the multicast group number or those less than the multicast index position (i.e., INDEXPOS) are non-zero.

MCRCVL- Multicast Receive Low (0x340)

Bit Field	Field Name	Type	Default Value	Description
31:0	MCRCV	RW	0x0	**Multicast Receive.** Each bit in this field corresponds to one of the lower 32 multicast groups (e.g., bit 0 corresponds to multicast group 0, bit 1 corresponds to multicast group 1, and so on). When a bit is set in this field for an enabled multicast group, multicast TLPs associated with that multicast group that reach the virtual PCI bus of the partition and were not received on this port are forwarded out the port (i.e., the port associated with the PCI-to-PCI bridge in which this register resides). The value of bits greater than NUMGROUP in the MCCTL register is ignored.

MCRCVH- Multicast Receive High (0x344)

Bit Field	Field Name	Type	Default Value	Description
31:0	MCRCV	RW	0x0	**Multicast Receive.** Each bit in this field corresponds to one of the upper 32 multicast groups (e.g., bit 0 corresponds to multicast group 32, bit 1 corresponds to multicast group 33, and so on). When a bit is set in this field for an enabled multicast group, multicast TLPs associated with that multicast group that reach the virtual PCI bus of the partition and were not received on this port are forwarded out the port (i.e., the port associated with the PCI-to-PCI bridge in which this register resides). The value of bits greater than NUMGROUP in the MCCTL register is ignored.

MCBLKALLL- Multicast Block All Low (0x348)

Bit Field	Field Name	Type	Default Value	Description
31:0	MCBLKALL	RW	0x0	**Multicast Block All.** Each bit in this field corresponds to one of the lower 32 multicast groups (e.g., bit 0 corresponds to multicast group 0, bit 1 corresponds to multicast group 1, and so on). When a bit is set in this field for an enabled multicast group, the PCI-to-PCI bridge associated with this register is blocked from forwarding multicast TLPs associated with that multicast group received on that port. This is an ingress port function performed on received TLPs. The value of bits greater than NUMGROUP in the MCCTL register is ignored.

MCBLKALLH- Multicast Block All High (0x34C)

Bit Field	Field Name	Type	Default Value	Description
31:0	MCBLKALL	RW	0x0	**Multicast Block All.** Each bit in this field corresponds to one of the upper 32 multicast groups (e.g., bit 0 corresponds to multicast group 32, bit 1 corresponds to multicast group 33, and so on). When a bit is set in this field for an enabled multicast group, the PCI-to-PCI bridge associated with this register is blocked from forwarding multicast TLPs associated with that multicast group received on that port. This is an ingress port function performed on received TLPs. The value of bits greater than NUMGROUP in the MCCTL register is ignored.

MCBLKUTL- Multicast Block Untranslated Low (0x350)

Bit Field	Field Name	Type	Default Value	Description
31:0	MCBLKUT	RW	0x0	**Multicast Block Untranslated.** Each bit in this field corresponds to one of the lower 32 multicast groups (e.g., bit 0 corresponds to multicast group 0, bit 1 corresponds to multicast group 1, and so on). When a bit is set in this field for an enabled multicast group, the function associated with this register is blocked from forwarding untranslated multicast TLPs associated with that multicast group received on that port. This is an ingress port function performed on received TLPs. The value of bits greater than NUMGROUP in the MCCTL register is ignored.

MCBLKUTH - Multicast Block Untranslated High (0x354)

Bit Field	Field Name	Type	Default Value	Description
31:0	MCBLKUT	RW	0x0	**Multicast Block Untranslated**. Each bit in this field corresponds to one of the upper 32 multicast groups (e.g., bit 0 corresponds to multicast group 32, bit 1 corresponds to multicast group 33, and so on). When a bit is set in this field for an enabled multicast group, the function associated with this register is blocked from forwarding untranslated multicast TLPs associated with that multicast group received on that port. This is an ingress port function performed on received TLPs. The value of bits greater than NUMGROUP in the MCCTL register is ignored.

MCOVRBARL- Multicast Overlay Base Address Low (0x358)

Bit Field	Field Name	Type	Default Value	Description
5:0	OVRSIZE	RW	0x0	**Overlay Size.** This field specifies the size of the overlay aperture. When the value in this field is less than six, the overlay mechanism is disabled.
31:6	MCOVRBARL	RW	0x0	**Multicast Overlay BAR Low.** This field specifies the lower 24-bits (i.e., bits 6 through 31) of the multicast overlay base address.

MCOVRBARH- Multicast Overlay Base Address High (0x35C)

Bit Field	Field Name	Type	Default Value	Description
31:0	MCOVRBARH	RW	0x0	**Multicast Overlay BAR High.** This field specifies the upper 32-bits (i.e., bits 32 through 63) of the multicast overlay base address.

Proprietary Port Specific Registers

Port Control and Status Registers

PCIESCTLIV - PCI Express Slot Control Initial Value (0x420)

Bit Field	Field Name	Type	Default Value	Description
0	ABPE	RW	0x0 SWSticky	**Attention Button Pressed Enable.** This field contains the initial value of the corresponding field in the PCI Express Slot Control (PCIESCTL) register when the corresponding slot or hot-plug capability is enabled. A partition reset does not reset slot and hot-plug capability bits since they are RWL. The intent of this field is to allow the initial value of the corresponding field in the PCIESCTL register to be controlled following a partition fundamental reset. A write to this field causes an immediate effect in the corresponding field in the PCIESCTL register.
1	PFDE	RW	0x0 SWSticky	**Power Fault Detected Enable.** This field contains the initial value of the corresponding field in the PCI Express Slot Control (PCIESCTL) register when the corresponding slot or hot-plug capability is enabled. A partition reset does not reset slot and hot-plug capability bits since they are RWL. The intent of this field is to allow the initial value of the corresponding field in the PCIESCTL register to be controlled following a partition fundamental reset. A write to this field causes an immediate effect in the corresponding field in the PCIESCTL register.
2	MRLSCE	RW	0x0 SWSticky	**MRL Sensor Change Enable.** This field contains the initial value of the corresponding field in the PCI Express Slot Control (PCIESCTL) register when the corresponding slot or hot-plug capability is enabled. A partition reset does not reset slot and hot-plug capability bits since they are RWL. The intent of this field is to allow the initial value of the corresponding field in the PCIESCTL register to be controlled following a partition fundamental reset. A write to this field causes an immediate effect in the corresponding field in the PCIESCTL register.
3	PDCE	RW	0x0 SWSticky	**Presence Detected Changed Enable.** This field contains the initial value of the corresponding field in the PCI Express Slot Control (PCIESCTL) register when the corresponding slot or hot-plug capability is enabled. A partition reset does not reset slot and hot-plug capability bits since they are RWL. The intent of this field is to allow the initial value of the corresponding field in the PCIESCTL register to be controlled following a partition fundamental reset. A write to this field causes an immediate effect in the corresponding field in the PCIESCTL register.

Bit Field	Field Name	Type	Default Value	Description
4	CCIE	RW	0x0 SWSticky	**Command Complete Interrupt Enable.** This field contains the initial value of the corresponding field in the PCI Express Slot Control (PCIESCTL) register when the corresponding slot or hot-plug capability is enabled. A partition reset does not reset slot and hot-plug capability bits since they are RWL. The intent of this field is to allow the initial value of the corresponding field in the PCIESCTL register to be controlled following a partition fundamental reset. A write to this field causes an immediate effect in the corresponding field in the PCIESCTL register.
5	HPIE	RW	0x0 SWSticky	**Hot Plug Interrupt Enable.** This field contains the initial value of the corresponding field in the PCI Express Slot Control (PCIESCTL) register when the corresponding slot or hot-plug capability is enabled. A partition reset does not reset slot and hot-plug capability bits since they are RWL. The intent of this field is to allow the initial value of the corresponding field in the PCIESCTL register to be controlled following a partition fundamental reset. A write to this field causes an immediate effect in the corresponding field in the PCIESCTL register.
7:6	AIC	RW	0x3 SWSticky	**Attention Indicator Control.** This field contains the initial value of the corresponding field in the PCI Express Slot Control (PCIESCTL) register when the corresponding slot or hot-plug capability is enabled. A partition reset does not reset slot and hot-plug capability bits since they are RWL. The intent of this field is to allow the initial value of the corresponding field in the PCIESCTL register to be controlled following a partition fundamental reset. A write to this field causes an immediate effect in the corresponding field in the PCIESCTL register.
9:8	PIC	RW	0x1 SWSticky	**Power Indicator Control.** This field contains the initial value of the corresponding field in the PCI Express Slot Control (PCIESCTL) register when the corresponding slot or hot-plug capability is enabled. A partition reset does not reset slot and hot-plug capability bits since they are RWL. The intent of this field is to allow the initial value of the corresponding field in the PCIESCTL register to be controlled following a partition fundamental reset. A write to this field causes an immediate effect in the corresponding field in the PCIESCTL register.
10	PCC	RW	0x0 SWSticky	**Power Controller Control.** This field contains the initial value of the corresponding field in the PCI Express Slot Control (PCIESCTL) register when the corresponding slot or hot-plug capability is enabled. A partition reset does not reset slot and hot-plug capability bits since they are RWL. The intent of this field is to allow the initial value of the corresponding field in the PCIESCTL register to be controlled following a partition fundamental reset. A write to this field causes an immediate effect in the corresponding field in the PCIESCTL register.

Bit Field	Field Name	Type	Default Value	Description
11	Reserved	RO	0x0	Reserved field.
12	DLLLASCE	RW	0x0 SWSticky	**Data Link Layer Link Active State Change Enable.** This field contains the initial value of the corresponding field in the PCI Express Slot Control (PCIESCTL) register when the corresponding slot or hot-plug capability is enabled. A partition reset does not reset slot and hot-plug capability bits since they are RWL. The intent of this field is to allow the initial value of the corresponding field in the PCIESCTL register to be controlled following a partition fundamental reset. A write to this field causes an immediate effect in the corresponding field in the PCIESCTL register.
15:13	Reserved	RO	0x0	Reserved field.

Internal Error Control and Status Registers

IERRORCTL - Internal Error Reporting Control (0x480)

Bit Field	Field Name	Type	Default Value	Description
0	IERROREN	RW	0x1 SWSticky	**Internal Error Reporting Enable.** When this bit is set, internal error reporting is enabled and reported through AER. Refer to section Internal Errors on page 3-12 for details.
31:1	Reserved	RO	0x0	Reserved field.

IERRORSTS - Internal Error Reporting Status (0x484)

Bit Field	Field Name	Type	Default Value	Description
0	IFBPTLPTO	RW1C	0x0 SWSticky	I**FB Posted TLP Time-Out.** This bit is set when a posted TLP time-out is detected in the IFB.
1	IFBNPTLPTO	RW1C	0x0 SWSticky	**IFB Non-Posted TLP Time-Out.** This bit is set when a non-posted TLP time-out is detected in the IFB.
2	IFBCPTLPTO	RW1C	0x0 SWSticky	**IFB Completion TLP Time-Out.** This bit is set when a completion time-out is detected in the IFB.
3	Reserved	RO	0x0	Reserved field.
4	EFBPTLPTO	RW1C	0x0 SWSticky	**EFB Posted TLP Time-Out.** This bit is set when a posted TLP time-out is detected in the EFB.
5	EFBNPTLPTO	RW1C	0x0 SWSticky	**EFB Non-Posted TLP Time-Out**. This bit is set when a non-posted TLP time-out is detected in the EFB.
6	EFBCPTLPTO	RW1C	0x0 SWSticky	**EFB Completion TLP Time-Out**. This bit is set when a completion time-out is detected in the EFB.
7	IFBDATSBE	RW1C	0x0 SWSticky	**IFB Data Single Bit Error**. This bit is set when a single bit ECC error is detected and corrected in the IFB data RAM.

Bit Field	Field Name	Type	Default Value	Description
8	IFBDATDBE	RW1C	0x0 SWSticky	**IFB Data Double Bit Error**. This bit is set when a double bit ECC error is detected in the IFB data RAM.
9	IFBCTLSBE	RW1C	0x0 SWSticky	**IFB Control Single Bit Error.** This bit is set when a single bit ECC error is detected and corrected in the IFB control RAM.
10	IFBCTLDBE	RW1C	0x0 SWSticky	**IFB Control Double Bit Error**.This bit is set when a double bit ECC error is detected in the IFB control RAM.
11	EFBDATSBE	RW1C	0x0 SWSticky	**EFB Data Single Bit Error.** This bit is set when a single bit ECC error is detected and corrected in the EFB data RAM.
12	EFBDATDBE	RW1C	0x0 SWSticky	**EFB Data Double Bit Error.** This bit is set when a double bit ECC error is detected in the EFB data RAM.
13	EFBCTLSBE	RW1C	0x0 SWSticky	**EFB Control Single Bit Error**. This bit is set when a single bit ECC error is detected and corrected in the EFB control RAM.
14	EFBCTLDBE	RW1C	0x0 SWSticky	**EFB Control Double Bit Error.** This bit is set when a double bit ECC error is detected in the EFB control RAM.
15	E2EPE	RW1C	0x0 SWSticky	**End-to-End Data Path Parity Error.** This bit is set when an end-to-end data path parity error is detected.
16	ULD	RW1C	0x0 SWSticky	**Unreliable Link Detected**. This bit is set when the ULD bit is set in the port's Autonomous Link Reliability Status (ALRSTS) register.
17	RBCTLSBE	RW1C	0x0 SWSticky	**Replay Buffer Control Single Bit Error**. This bit is set when a single bit ECC error is detected and corrected in the Replay Buffer's control RAM.
18	RBCTLDBE	RW1C	0x0 SWSticky	**Replay Buffer Control Double Bit Error.** This bit is set when a double bit ECC error is detected in the Replay Buffer's control RAM.
31:19	Reserved	RO	0x0	Reserved field.

IERRORMSK - Internal Error Reporting Mask (0x488)

Bit Field	Field Name	Type	Default Value	Description
0	IFBPTLPTO	RW	0x0 SWSticky	**IFB Posted TLP Time-Out.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
1	IFBNPTLPTO	RW	0x0 SWSticky	**IFB Non-Posted TLP Time-Out.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
2	IFBCPTLPTO	RW	0x0 SWSticky	**IFB Completion TLP Time-Out.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
3	Reserved	RO	0x0	Reserved field.

Bit Field	Field Name	Type	Default Value	Description
4	EFBPTLPTO	RW	0x0 SWSticky	**EFB Posted TLP Time-Out.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
5	EFBNPTLPTO	RW	0x0 SWSticky	**EFB Non-Posted TLP Time-Out.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
6	EFBCPTLPTO	RW	0x0 SWSticky	**EFB Completion TLP Time-Out.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
7	IFBDATSBE	RW	0x0 SWSticky	**IFB Data Single Bit Error.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
8	IFBDATDBE	RW	0x0 SWSticky	**IFB Data Double Bit Error.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
9	IFBCTLSBE	RW	0x0 SWSticky	**IFB Control Single Bit Error.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
10	IFBCTLDBE	RW	0x0 SWSticky	**IFB Control Double Bit Error.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
11	EFBDATSBE	RW	0x0 SWSticky	**EFB Data Single Bit Error.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
12	EFBDATDBE	RW	0x0 SWSticky	**EFB Data Double Bit Error.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
13	EFBCTLSBE	RW	0x0 SWSticky	**EFB Control Single Bit Error.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.

Bit Field	Field Name	Type	Default Value	Description
14	EFBCTLDBE	RW	0x0 SWSticky	**EFB Control Double Bit Error.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
15	E2EPE	RW	0x0 SWSticky	**End-to-End Data Path Parity Error.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
16	ULD	RW	0x0 SWSticky	**Unreliable Link Detected.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
17	RBCTLSBE	RW	0x0 SWSticky	**Replay Buffer Control Single Bit Error.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
18	RBCTLDBE	RW	0x0 SWSticky	**Replay Buffer Control Double Bit Error.** When this bit is set, the corresponding error bit in the IERRORSTS register is masked from reporting an internal error to the AER Capability Structure. This bit does not affect the state of the corresponding bit in the IERRORSTS register.
31:19	Reserved	RO	0x0	Reserved field.

IERRORSEV - Internal Error Reporting Severity (0x48C)

Bit Field	Field Name	Type	Default Value	Description
0	IFBPTLPTO	RW	0x1 SWSticky	IFB Posted TLP Time-Out. This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
1	IFBNPTLPTO	RW	0x0 SWSticky	IFB Non-Posted TLP Time-Out. This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
2	IFBCPTLPTO	RW	0x0 SWSticky	IFB Completion TLP Time-Out. This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
3	Reserved	RO	0x0	Reserved field.

Bit Field	Field Name	Type	Default Value	Description
4	EFBPTLPTO	RW	0x1 SWSticky	EFB Posted TLP Time-Out. This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
5	EFBNPTLPTO	RW	0x0 SWSticky	EFB Non-Posted TLP Time-Out. This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
6	EFBCPTLPTO	RW	0x0 SWSticky	**EFB Completion TLP Time-Out.** This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
7	IFBDATSBE	RW	0x0 SWSticky	**IFB Data Single Bit Error.** This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
8	IFBDATDBE	RW	0x1 SWSticky	**IFB Data Double Bit Error.** This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
9	IFBCTLSBE	RW	0x0 SWSticky	**IFB Control Single Bit Error.** This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
10	IFBCTLDBE	RW	0x1 SWSticky	**IFB Control Double Bit Error.** This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
11	EFBDATSBE	RW	0x0 SWSticky	**EFB Data Single Bit Error.** This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.

Bit Field	Field Name	Type	Default Value	Description
12	EFBDATDBE	RW	0x1 SWSticky	**EFB Data Double Bit Error.** This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
13	EFBCTLSBE	RW	0x0 SWSticky	**EFB Control Single Bit Error.** This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
14	EFBCTLDBE	RW	0x1 SWSticky	**EFB Control Double Bit Error.** This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
15	E2EPE	RW	0x1 SWSticky	**End-to-End Data Path Parity Error.** This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
16	ULD	RW	0x0 SWSticky	**Unreliable Link Detected.** This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
17	RBCTLSBE	RW	0x0 SWSticky	**Replay Buffer Control Single Bit Error.** This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
18	RBCTLDBE	RW	0x1 SWSticky	**Replay Buffer Control Double Bit Error.** This bit controls how an unmasked error of the corresponding type is reported. When this bit is set and the corresponding status bit is set and unmasked, then the error is reported as an uncorrectable internal error. When this bit is cleared and the corresponding status bit is set and unmasked, then the error is reported as an correctable internal error.
31:19	Reserved	RO	0x0	Reserved field.

IERRORTST - Internal Error Reporting Test (0x490)

Bit Field	Field Name	Type	Default Value	Description
0	IFBPTLPTO	RW	0x0	IFB Posted TLP Time-Out. Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
1	IFBNPTLPTO	RW	0x0	IFB Non-Posted TLP Time-Out. Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
2	IFBCPTLPTO	RW	0x0	IFB Completion TLP Time-Out. Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
3	Reserved	RO	0x0	Reserved field.
4	EFBPTLPTO	RW	0x0	**EFB Posted TLP Time-Out.** Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
5	EFBNPTLPTO	RW	0x0	**EFB Non-Posted TLP Time-Out.** Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
6	EFBCPTLPTO	RW	0x0	**EFB Completion TLP Time-Out.** Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
7	IFBDATSBE	RW	0x0	**IFB Data Single Bit Error.** Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
8	IFBDATDBE	RW	0x0	**IFB Data Double Bit Error.** Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
9	IFBCTLSBE	RW	0x0	**IFB Control Single Bit Error.** Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
10	IFBCTLDBE	RW	0x0	**IFB Control Double Bit Error.** Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
11	EFBDATSBE	RW	0x0	**EFB Data Single Bit Error.** Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
12	EFBDATDBE	RW	0x0	**EFB Data Double Bit Error.** Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
13	EFBCTLSBE	RW	0x0	**EFB Control Single Bit Error.** Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
14	EFBCTLDBE	RW	0x0	**EFB Control Double Bit Error**. Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
15	E2EPE	RW	0x0	**End-to-End Data Path Parity Error.** Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.

Bit Field	Field Name	Type	Default Value	Description
16	ULD	RW	0x0	**Unreliable Link Detected.** Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read
17	RBCTLSBE	RW	0x0	**Replay Buffer Control Single Bit Error.** Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
18	RBCTLDBE	RW	0x0	**Replay Buffer Control Double Bit Error.** Writing a one to this bit sets the corresponding bit in the IERRORSTS register. This bit always returns a value of zero when read.
31:19	Reserved	RO	0x0	Reserved field.

Physical Layer Control and Status Registers

SERDESCFG - SerDes Configuration (0x510)

Bit Field	Field Name	Type	Default Value	Description
7:0	RCVD_OVRD	RW	0x0 SWSticky	**Receiver Detect Override.** Each bit in this register corresponds to a SerDes lane. Setting this bit causes the lane associated with this bit to indicate that a receiver has been detected on the line. For even numbered ports, the upper 4 bits are only valid when a port operates in merged mode. For odd numbered ports, the upper 4 bits are never valid (i.e., undefined). This field is not valid when the port operates in SerDes Test Mode.
15:8	Reserved	RO	0x0	Reserved field.
16	LSE	RW	0x0 SWSticky	**Low-Swing Mode Enable.** When set, this bit enables Low-Swing mode operation at the SerDes Transmit logic. Please refer to section Low-Swing Transmitter Voltage Mode on page 7-13 for further details. 0x0 - Full-Swing Mode 0x1 - Low-Swing Mode
31:17	Reserved	RO	0x0	Reserved field.

LANESTS0 - Lane Status 0 (0x51C)

Bit Field	Field Name	Type	Default Value	Description
7:0	PDE	RW1C	0x0	**Phy Disparity Error.** Each bit in this field corresponds to a SerDes lane associated with the port. A bit is set when an 8B10B coding violation has resulted in a running disparity error in the received data stream. For even numbered ports, the upper 4 bits are only valid when a port operates in merged mode. For odd numbered ports, the upper 4 bits are never valid (i.e., undefined). A bit can only be set when the LTSSM is in the L0 state.
15:8	Reserved	RO	0x0	Reserved field.
23:16	E8B10B	RW1C	0x0	**8B10B Error.** Each bit in this field corresponds to a SerDes lane associated with the port. A bit is set when an 8B10B decode error is detected in the received data stream. For even numbered ports, the upper 4 bits are only valid when a port operates in merged mode. For odd numbered ports, the upper 4 bits are never valid (i.e., undefined). A bit can only be set when the LTSSM is in the L0, Configuration, Disabled, or Hot-Reset states.
31:24	Reserved	RO	0x0	Reserved field.

LANESTS1 - Lane Status 1 (0x520)

Bit Field	Field Name	Type	Default Value	Description
7:0	UND	RW1C	0x0 Sticky	**Receiver Underflow Detected.** Each bit in this field corresponds to a SerDes lane associated with the port. A bit is set when the corresponding link receiver is unable to compensate for clock variance between link partners and has inserted one or more zero bytes into the stream. For even numbered ports, the upper 4 bits are only valid when a port operates in merged mode. For odd numbered ports, the upper 4 bits are never valid (i.e., undefined). A bit can only be set when the LTSSM is in the L0 state.
15:8	Reserved	RO	0x0	Reserved field.
23:16	OVR	RW1C	0x0 Sticky	**Receiver Overflow Detected.** Each bit in this field corresponds to a SerDes lane associated with the port. A bit is set when the corresponding link receiver is unable to compensate for clock variance between link partners and has dropped one or more bytes. For even numbered ports, the upper 4 bits are only valid when a port operates in merged mode. For odd numbered ports, the upper 4 bits are never valid (i.e., undefined). A bit can only be set when the LTSSM is in the L0 state.
31:24	Reserved	RO	0x0	Reserved field.

PHYLCFG0 - Phy Link Configuration 0 (0x530)

Bit Field	Field Name	Type	Default Value	Description
7:0	Reserved	RO	0x0	Reserved field.
8	G1CME	RW	0x0 SWSticky	**Gen1 Compatibility Mode Enable.** When this bit is set, the PHY operates in Gen1 Compatibility mode. In this mode, the PHY does not set training set bits not defined in the PCIe 1.1 specification.
10:9	Reserved	RO	0x0	Reserved field.
11	CLINKDIS	RW	0x0 SWSticky	**Disable Crosslink.** When this bit is set, crosslink link training is disabled and the device link trains as though crosslink were not implemented. Please refer to section Crosslink on page 6-15 for further details.
13:12	Reserved	RO	0x0	Reserved field.
14	ILSCC	RW	Downstream: 0x0 Other: 0x1 SWSticky	**Initial Link Speed Change Control**. This field determines whether a port automatically initiates a speed change to Gen2 speed, if Gen2 speed is permissible, after initial entry to L0 from Detect. 0x0 - (automatic) Automatically initiate speed change to Gen2 speed, if permissible, after the first entry to L0 from Detect. 0x1 - (nochange) Do not automatically initiate a speed change to Gen2 speed, stay in Gen1 speed. Note that the initial value of this field depends on the port operating mode.
31:15	Reserved	RO	0x0	Reserved field.

PHYLSTATE0 - Phy Link State 0 (0x540)

Bit Field	Field Name	Type	Default Value	Description
30:0	Reserved	RO	0x0	Reserved field.
31	FLRET	RW	0x0	**Full Link Retrain.** Writing a one to this field initiates full link retraining by directing the PHY LTSSM into the DETECT state. This bit always returns zero when read. Writing of a one to this bit always results in the switch returning a completion to the requester before the action specified by this bit takes effect. For an upstream port, writing of a one to this bit always results in the action specified by this bit to take effect after 1ms. The switch always returns a completion to the requester before the effect of this bit is applied.

PHYPRBS - Phy PRBS Seed (0x55C)

Bit Field	Field Name	Type	Default Value	Description
15:0	SEED	RW	0xFFFF SWSticky	**Phy PRBS Seed Value.** This field contains the PHY PRBS seed value used for crosslink operation. When the value in this register is modified, the PRBS counter associated with this seed is reset to the seed value and re-starts counting.
31:16	Reserved	RO	0x0	Reserved field.

Power Management Control and Status Registers

L1ASPMRTC - L1 ASPM Rejection Timer Control (0x710)

Bit Field	Field Name	Type	Default Value	Description
13:0	MTL1ER	RW	0x947 SWSticky	**Minimum Time between L1 Entry Requests.** This field indicates the minimum time (in 250Mhz cycles) that the port waits between detecting consecutive L1 ASPM entry requests. An L1 ASPM entry request consists of a number of PM_L1_Active_State_Request DLLPs followed by an acknowledge (positive or negative) from the downstream port receiving the request. The actual time may be calculated by multiplying the value in this field by 4ns. For example, a setting of 0x947 (i.e., 2375 d) corresponds to a time of (2375 cycles * 4ns = 9.5us). Refer to section 5.4.1.2.1 of the PCI Express Base Specification 2.0 for further details on the L1 ASPM Entry rejection protocol.
15:14	Reserved	RO	0x0	Reserved field.
16	TSCTL	RW	0x1 SWSticky	**Timer Start Control.** Upon rejecting an L1 ASPM entry request from the link partner, the switch port counts an amount of time equal to the value in the MTL1ER field before detecting a new request. This field selects the criteria for starting the timer. 0x0 - Timer starts counting when the port has issued a PM_L1_Active_State_Nak TLP. Reception of PM_L1_Active_State_Request DLLPs from the link partner prior to the expiration of the timer result in the timer being re-started. 0x1 - Timer starts counting when the port has issued a PM_L1_Active_State_Nak TLP. Reception of PM_L1_Active_State_Request DLLPs from the link partner prior to the expiration of the timer are ignored. Once the timer has expired, reception of an incoming PM_L1_Active_State_Request DLLP will be treated as a new request.
31:17	Reserved	RO	0x0	Reserved field.

Request Metering

RMCTL - Requester Metering Control (0x880)

Bit Field	Field Name	Type	Default Value	Description
0	EN	RW	0x0 SWSticky	**Enable.** When this bit is set, request metering is enabled on the corresponding input port. Refer to section Request Metering on page 3-7.
4:1	OVRFACTOR	RW	0x2 SWSticky	**Overhead Factor.** This field contains the overhead factor used in computing the completion size estimate for memory read requests.
5	Reserved	RO	0x0	Reserved field.
15:6	CNSTLIMIT	RW	0x10 SWSticky	**Constant Limit.** This field is used to control the algorithm used to compute the completion size estimate for non-posted read requests when request metering is enabled. When the number of DWords required in completions to service a non-posted read request is less than or equal to the value in this field, then a constant completion size estimate is used. When the number of DWords required in completions to service a non-posted read request is greater than the value in this field, then the completion size estimate considers the number of individual completion TLPs.
31:16	DVADJ	RW	0x0 SWSticky	**Decrement Value Adjustment.** This field contains the adjustment value used to determine the value by which the request metering counter is decremented each 250 MHz clock tick. The value in this field represents a positive signed-magnitude, fixed-point number with 4 integer bits and 11 fractional bits (i.e., a 0:4:11 format number). The most significant bit should always be set to 0b0.

RMCOUNT - Requester Metering Count (0x88C)

Bit Field	Field Name	Type	Default Value	Description
15:0	COUNT	RO	0x0	**Count.** This field contains the requester metering initial counter value for the last non-posted request. The request metering counter is a 24-bit counter that represents a fixed point 0:13:11 number (i.e., an unsigned number with 13 integer bits and 11 fractional bits). The value in this field represents an unsigned number with 13 integer bits and 3 fractional bits. The least significant eight fractional bits of the initial counter value are always zero.
31:16	Reserved	RO	0x0	Reserved field.

Global Address Space Access Registers

GASAADDR - Global Address Space Access Address (0xFF8)

Bit Field	Field Name	Type	Default Value	Description
1:0	Reserved	RO	0x0	Reserved field.
18:2	GADDR	RW	0x0	**Global Address.** This field selects the system address of the register to be accessed via the GASADATA register. The value of this register must not be programmed to point to the address of the GASAADDR or GASADATA register in this or any other port. Similarly, the value of this register must not be programmed to point to the address of the Extended Configuration Address (ECFGADDR) or Extended Configuration Data registers (ECFGDATA) in this or any other function. Violations of these rules produce undefined results.
31:19	Reserved	RO	0x0	Reserved field.

GASADATA - Global Address Space Access Data (0xFFC)

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

Switch Configuration and Status Registers

Switch Control and Status Registers

SWCTL - Switch Control (0x0000)

Bit Field	Field Name	Type	Default Value	Description
1:0	Reserved	RO	0x0	Reserved field.
2	RSTHALT	RW	HWINIT SWSticky	**Reset Halt.** When this bit is set, all of the switch logic except the SMBus interface remains in a quasi-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 switch fundamental reset sequence outlined in section Switch Fundamental Reset on page 5-2. The initial value of this bit is that of the RSTHALT signal in the boot configuration vector.
3	REGUNLOCK	RW	0x0 SWSticky	**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. 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 switch fundamental reset sequence to allow the serial EEPROM to modify the contents of RWL fields.
4	PWRBDVUL	RWL	0x0 SWSticky	**Power Budgeting Data Value Unlock.** When this bit is set, the Power Budgeting Data Value [7:0] (PWRBDV[7:0]) registers in all ports may be read and written. When this bit is cleared, then the PWRBDV registers in all ports are read-only.
8:5	Reserved	RO	0x0	Reserved field.

Bit Field	Field Name	Type	Default Value	Description
10:9	DDDNC	RW	0x0 SWSticky	**Disable Downstream Device Number Checking.** This field controls the extent to which device numbers are checked by downstream ports. This field is present for backwards compatibility with earlier IDT switches that implement a proprietary version of ARI forwarding. The setting of 0x1 corresponds to the operation dictated by the PCI Express base specification. 0x0 - (all) Inhibit the transmission of all TLPs that are routed by ID, specify a bus number associated with a downstream port link, and specify a device number other than zero. 0x1 - (cfg) Inhibit the transmission of configuration request TLPs that are routed by ID, specify a bus number associated with a downstream port link, and specify a device number other than zero. Non-configuration request TLPs are delivered to the device attached to a link associated with a downstream switch port regardless of the specified device number. 0x2 - (none) All TLPs are delivered to the device attached to a link associated with a downstream switch port regardless of the specified device number. 0x3 - reserved
18:11	Reserved	RO	0x0	Reserved field.
19	BDISCARD	RW	0x0 SWSticky	**Discard Vendor Defined Broadcast Messages.** When this bit is set, vendor defined Type 1 broadcast messages received on the upstream port are silently discarded and not forwarded downstream. Silently discarding a TLP means that flow control credits are returned, TLP contents are discarded, and no error bits are set.
31:20	Reserved	RO	0x0	Reserved field.

BCVSTS - Boot Configuration Vector Status (0x0004)

Bit Field	Field Name	Type	Default Value	Description
3:0	SWMODE	RO	HWINIT	**Switch Mode.** Boot configuration vector value sampled during a switch fundamental reset.
4	GCLKFSEL	RO	HWINIT	**Global Clock Frequency Select.** Boot configuration vector value sampled during a switch fundamental reset.
11:5	Reserved	RO	0x0	Reserved field.
15:12	SSMBADDR	RO	HWINIT	**Slave SMBus Address.** Boot configuration vector value sampled during a switch fundamental reset. Bits 5 and 1 through 3 of slave SMBus address.
16	RSTHALT	RO	HWINIT	**Reset Halt.** Boot configuration vector value sampled during a switch fundamental reset.
20:17	Reserved	RO	0x0	Reserved field.
23:21	CLKMODE	RO	HWINIT	**Clock Mode.** Boot configuration vector value sampled during a switch fundamental reset.

Bit Field	Field Name	Type	Default Value	Description
24	P01MERGEN	RO	HWINIT	**Port 0 and 1 Merge.** Boot configuration vector value sampled during a switch fundamental reset.
25	P23MERGEN	RO	HWINIT	**Port 2 and 3 Merge.** Boot configuration vector value sampled during a switch fundamental reset.
26	P45MERGEN	RO	HWINIT	**Port 4 and 5 Merge.** Boot configuration vector value sampled during a switch fundamental reset.
27	P67MERGEN	RO	HWINIT	**Port 6 and 7 Merge.** Boot configuration vector value sampled during a switch fundamental reset.
31:28	Reserved	RO	0x0	Reserved field.

Internal Switch Timer

USSBRDELAY - Upstream Secondary Bus Reset Delay (0x008C)

Bit Field	Field Name	Type	Default Value	Description
15:0	USSBR	RW	0x0 SWSticky	**Side Effect Delay**.This field specifies the delay in microseconds from when a configuration request that initiates a secondary bus reset is processed to the start of the secondary bus reset action. The default value corresponds to no delay. Decreasing this delay is discouraged, except for the cases explicitly listed in this specification.
31:16	Reserved	RO	0x0	Reserved field.

Switch Port Registers

SWPORT[13:12, 9:0]CTL - Switch Port x Control

Bit Field	Field Name	Type	Default Value	Description
9:0	Reserved	RO	0x0	Reserved field.
14:10	DEVNUM	RW	HWINIT SWSticky	**Device Number.** When the port is configured as a downstream switch port, this field specifies the device number associated with the port. In all other port modes, this field has no effect. The initial value of this field corresponds to the port number.
15	Reserved	RO	0x0	Reserved field.

Bit Field	Field Name	Type	Default Value	Description
17:16	OMA	RW	0x1 SWSticky	**Operating Mode Change Action.** This field specifies the action taken when a modification is made to the operating mode of a port. 0x0 - (noaction) No action - preserve state 0x1 - (reset) Port reset - behavior associated with fundamental reset 0x2 - (hotreset) Hot reset - behavior associated with a hot reset. **Note: This field must be set to 0x0 when changing the Device Number field in this register.**
18	Reserved	RO	0x0	Reserved field.
19	Reserved	RW	HWINIT SWSticky	This field is unused in the switch and serves as a place holder for the future.
21:20	Reserved	RW	HWINIT SWSticky	This field is unused in the switch and serves as a place holder for the future.
31:22	Reserved	RO	0x0	Reserved field.

SWPORT[13:12, 9:0]STS - Switch Port x Status

Bit Field	Field Name	Type	Default Value	Description
0	OMCI	RW1C	0x0 SWSticky	**Operating Mode Change Initiated.** This bit is set when a port operating mode change is initiated.
1	OMCC	RW1C	0x0 SWSticky	**Operating Mode Change Completed.** This bit is set when a port operating mode change is completed.
2	Reserved	RW1C	0x0 SWSticky	This field is unused in the switch and serves as a place holder for the future.
3	Reserved	RW1C	0x0 SWSticky	This field is unused in the switch and serves as a place holder for the future.
15:4	Reserved	RO	0x0	Reserved field.
20:16	DEVNUM	RO	HWINIT	**Device Number.** When the port is configured as a downstream switch port, this field contains the device number associated with the port. In all other port modes, this value of this field is undefined. Due to the time it takes for a device number change to complete, this value may be different than that in the DEVNUM field in the SWPORTxCTL register.
31:21	Reserved	RO	0x0	Reserved field.

SerDes Control and Status Registers

Refer to Chapter 7, SerDes, for a details on programming SerDes controls. Note that in order to program the SerDes controls for a given port, it is necessary to identify which SerDes block is associated with the port. Refer to section SerDes Numbering and Port Association on page 7-1 for details.

S[13:12, 9:0]CTL - SerDes x Control

Bit Field	Field Name	Type	Default Value	Description
4:0	LANESEL	RW	0x10 SWSticky	**Lane Select.** This field selects the lane on which the SerDes lane control registers (S[x]TXLCTL0, S[x]TXLCTL1, S[x]RXLCTL, and S[x]RXEQLCTL) operate when written. 0x0 - Operate on lane 0 only 0x1 - Operate on lane 1 only 0x2 - Operate on lane 2 only 0x3 - Operate on lane 3 only 0x10 - Operate on all lanes simultaneously Others - Reserved For example, when LANESEL=0x0, configuration writes to the above listed registers affect lane 0 of the SerDes only. When LANESEL=0x10, the settings in the SerDes lane control registers are applied to all lanes simultaneously. Read operations are not affected by this field (i.e., reading from a SerDes lane control register returns the last value written to that register, regardless of the setting of this field). Operating on a reserved lane results in undefined consequences.
5	POWERDN	RW	0x0 SWSticky	**SerDes Power-Down.** When this bit is set, the SerDes is placed in a deep low-power state (i.e., the SerDes lanes are placed in P2 and the CMU is powered-down). In addition, the PHY LTSSM in the corresponding port(s) is immediately transitioned to the Detect state. When this bit is cleared, the SerDes is powered-on, initialized, and the PHY LTSSM initiates link training. This bit has no effect when the SerDes is already powered-down (e.g., the SerDes quad associated with a disabled port). When a SerDes is powered-down, the serial Tx/Rx pins and reference resistor pins may be left unconnected. Refer to section SerDes Power Management on page 7-16 for further details on SerDes power management.
31:6	Reserved	RO	0x0	Reserved field.

S[13:12, 9:0]TXLCTL0 - SerDes x Transmitter Lane Control 0

Bit Field	Field Name	Type	Default Value	Description
1:0	CDC_FS3DBG1	RW	0x3 SWSticky	**Transmit Driver Coarse De-Emphasis Control for Full Swing mode in Gen1.** This field provides coarse level control of the transmit driver de-emphasis level in full-swing mode and Gen 1 data rate (i.e., 2.5 GT/s). This field has no effect when the port operates in low-swing mode (i.e., de-emphasis is turned-off in this mode). This field controls the voltage level for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Note that when operating in Gen 1 data rate, the de-emphasis should nominally be set to -3.5dB of the transmit driver voltage level. Refer to section Programmable Voltage Margining and De-Emphasis on page 7-4 for further details on programming this field.
3:2	CDC_FS3DBG2	RW	0x1 SWSticky	**Transmit Driver Coarse De-Emphasis Control for Full Swing mode with -3.5dB in Gen2.** This field provides coarse level control of the transmit driver de-emphasis level in Gen 2 mode, when the SDE field in the associated port's PCIELCTL2 register is set to -3.5dB de-emphasis. This field has no effect when the port operates in low-swing mode (i.e., de-emphasis is turned-off in this mode). This field controls the voltage level for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Refer to section Programmable Voltage Margining and De-Emphasis on page 7-4 for further details on programming this field.
5:4	CDC_FS6DBG2	RW	0x3 SWSticky	**Transmit Driver Coarse De-Emphasis Control for Full Swing mode with -6.0dB in Gen2.** This field provides coarse level control of the transmit driver de-emphasis level in Gen 2 mode, when the SDE field in the associated port's PCIELCTL2 register is set to -6.0dB de-emphasis. This field has no effect when the port operates in low-swing mode (i.e., de-emphasis is turned-off in this mode). This field controls the voltage level for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Refer to section Programmable Voltage Margining and De-Emphasis on page 7-4 for further details on programming this field.
7:6	Reserved	RO	0x0	Reserved field.
9:8	TX_EQ_3DBG1	RW	0x2 SWSticky	**Transmit Equalization for Full Swing Mode with -3.5dB in Gen1.** his field controls the transmit equalization in Gen 1 data rate, when the SDE field in the associated port's PCIELCTL2 register is set to -3.5 dB de-emphasis. This field controls the transmit equalization for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Valid values for this field are 0x0 to 0x3.

Bit Field	Field Name	Type	Default Value	Description
11:10	TX_EQ_3DBG2	RW	0x1 SWSticky	**Transmit Equalization for Full Swing Mode with -3.5dB in Gen2.** This field controls the transmit equalization in Gen 2 data rate, when the SDE field in the associated port's PCIELCTL2 register is set to -3.5 dB de-emphasis. This field controls the transmit equalization for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Valid values for this field are 0x0 to 0x3.
13:12	TX_EQ_6DBG2	RW	0x1 SWSticky	**Transmit Equalization for Full Swing Mode with -6.0dB in Gen2.** This field controls the transmit equalization in Gen 2 data rate, when the SDE field in the associated port's PCIELCTL2 register is set to -6.0 dB de-emphasis. This field controls the transmit equalization for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Valid values for this field are 0x0 to 0x3.
15:14	TX_EQ_LS	RW	0x0 SWSticky	**Transmit Equalization for Low Swing Mode.** This field controls the transmit equalization when the associated port operates in low-swing mode (i.e., the LSE bit in the port's SERDESCFG register is set to 0x1), for both Gen 1 and Gen 2 data rates. This field controls the transmit equalization for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Valid values for this field are 0x0 to 0x3.
17:16	TX_SLEW_G1	RW	0x2 SWSticky	**Transmit Slew Control in Gen1.** This field controls the output driver's slew rate at Gen1 data-rate, for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register. This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register) 0x0 - 48 ps 0x1 - 50 ps 0x2 - 60 ps 0x3 - 89 ps
20:18	TX_FSLEW_G1	RW	0x0 SWSticky	**Transmit Driver Fine Slew Adjustment in Gen1.** This field allows fine adjustment of the output driver's slew rate at Gen 1 data-rate, for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register. This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register).
22:21	TX_SLEW_G2	RW	0x0 SWSticky	**Transmit Slew Control in Gen2.** This field controls the output driver's slew rate at Gen2 data-rate, for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register. This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register) 0x0 - 48 ps 0x1 - 50 ps 0x2 - 60 ps 0x3 - 89 ps

Bit Field	Field Name	Type	Default Value	Description
25:23	TX_FSLEW_G2	RW	0x0 SWSticky	**Transmit Driver Fine Slew Adjustment in Gen2.** This field allows fine adjustment of the output driver's slew rate at Gen 2 data-rate, for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register. This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register).
26	TX_SLEW_C	RW	0x0 SWSticky	**Transmit Slew Control Disable.** When set, the slew control fields in this register (TX_SLEW_G1/2 and TX_FSLEW_G1/2) are disabled and the SerDes transmitter uses its maximum slew rate setting.
28:27	TX_AMPBOOST	RW	0x1 SWSticky	**Transmit Driver Amplitude Boost.** This field increases the transmitter driver's differential swing for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register. Valid settings are 0x0 to 0x3, where 0x0 selects no amplitude boost and 0x3 selects the highest amount of boost. Each increasing setting boosts the amplitude by ~5% over the previous setting. Increasing this setting will also increase the power consumed by the affected lanes. Each increasing setting increases power consumption by ~2% over the previous setting. This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register).
31:29	Reserved	RO	0x0	Reserved field.

S[13:12, 9:0]TXLCTL1 - SerDes x Transmitter Lane Control 1

Bit Field	Field Name	Type	Default Value	Description
4:0	TDVL_FS3DBG1	RW	0x11 SWSticky	**Transmit Driver Voltage Level for Full-Swing Mode with -3.5dB De-emphasis in Gen1**. This field controls the SerDes transmit driver voltage level in full-swing mode and Gen 1 data rate (i.e., 2.5 GT/s). The value of this field corresponds to the peak-to-peak differential voltage at the transmitter pins, prior to de-emphasis being applied. This field controls the voltage level for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Refer to section Programmable Voltage Margining and De-Emphasis on page 7-3 for further details on programming this field.
7:5	FDC_FS3DBG1	RW	0x4 SWSticky	**Transmit Driver Fine De-emphasis Control for Full Swing Mode with -3.5dB in Gen 1.** This field provides fine level control of the transmit driver de-emphasis level in full-swing mode and Gen 1 data rate (i.e., 2.5 GT/s). Note that when operating in Gen 1 data rate, the de-emphasis should nominally be set to -3.5dB of the transmit driver voltage level. This field has no effect when the port operates in low-swing mode (i.e., de-emphasis is turned-off in this mode). This field controls the voltage level for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Refer to section Programmable Voltage Margining and De-Emphasis on page 7-3 for further details on programming this field.
12:8	TDVL_FS3DBG2	RW	0x17 SWSticky	**Transmit Driver Voltage Level for Full-Swing Mode with -3.5dB De-emphasis in Gen 2.** This field controls the SerDes transmit driver voltage level in full-swing mode and Gen 2 data rate, when the SDE field in the associated port's PCIELCTL2 register is set to -3.5dB de-emphasis. The value of this field corresponds to the peak-to-peak differential voltage at the transmitter pins, prior to de-emphasis being applied. This field controls the voltage level for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Refer to section Programmable Voltage Margining and De-Emphasis on page 7-3 for further details on programming this field.

Bit Field	Field Name	Type	Default Value	Description
15:13	FDC_FS3DBG2	RW	0x4 SWSticky	**Transmit Driver Fine De-emphasis Control for Full Swing Mode with -3.5dB in Gen 2.** This field provides fine level control of the transmit driver de-emphasis level in Gen 2 mode, when the SDE field in the associated port's PCIELCTL2 register is set to -3.5dB de-emphasis. This field has no effect when the port operates in low-swing mode (i.e., de-emphasis is turned-off in this mode). This field controls the voltage level for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Refer to section Programmable Voltage Margining and De-Emphasis on page 7-3 for further details on programming this field.
20:16	TDVL_FS6DBG2	RW	0x13 SWSticky	**Transmit Driver Voltage Level for Full-Swing Mode with -6.0dB De-emphasis in Gen 2.** This field controls the SerDes transmit driver voltage level in full-swing mode and Gen 2 data rate, when the SDE field in the associated port's PCIELCTL2 register is set to -6.0dB de-emphasis. The value of this field corresponds to the peak-to-peak differential voltage at the transmitter pins, prior to de-emphasis being applied. This field controls the voltage level for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Refer to section Programmable Voltage Margining and De-Emphasis on page 7-3 for further details on programming this field.

Bit Field	Field Name	Type	Default Value	Description
23:21	FDC_FS6DBG2	RW	0x2 SWSticky	**Transmit Driver Fine De-emphasis Control for Full Swing Mode with -6.0dB in Gen 2.** This field provides fine level control of the transmit driver de-emphasis level in Gen 2 mode, when the SDE field in the associated port's PCIELCTL2 register is set to -6.0dB de-emphasis. This field has no effect when the port operates in low-swing mode (i.e., de-emphasis is turned-off in this mode). This field controls the voltage level for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Refer to section Programmable Voltage Margining and De-Emphasis on page 7-3 for further details on programming this field.
27:24	TDVL_LSG1	RW	0x9 SWSticky	**Transmit Driver Voltage Level for Low-Swing Mode in Gen 1.** This field controls the SerDes transmit driver voltage level when the associated port operates in low-swing mode (i.e., the LSE bit in the port's SERDESCFG register is set to 0x1) and Gen 1 data rate. The value of this field corresponds to the peak-to-peak differential voltage at the transmitter pins. This field controls the voltage level for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Refer to section Programmable Voltage Margining and De-Emphasis on page 7-3 for further details on programming this field.
31:28	TDVL_LSG2	RW	0x8 SWSticky	**Transmit Driver Voltage Level for Low-Swing Mode in Gen 2.** This field controls the SerDes transmit driver voltage level when the associated port operates in low-swing mode (i.e., the LSE bit in the port's SERDESCFG register is set to 0x1) and Gen 2 data rate. The value of this field corresponds to the peak-to-peak differential voltage at the transmitter pins. This field controls the voltage level for the lane(s) selected by the Lane Select (LANESEL[3:0]) field in the SerDes Control (S[x]CTL) register.This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register). Refer to section Programmable Voltage Margining and De-Emphasis on page 7-3 for further details on programming this field.

S[13:12, 9:0]RXEQLCTL - SerDes x Receiver Equalization Lane Control

Bit Field	Field Name	Type	Default Value	Description
2:0	RXEQZ	RW	0x1 SWSticky	**Receiver Equalization Zero.** Amplifies the high-frequency gain of the equalizer. A value of 0x0 results in the smallest amount of high frequency gain. A value of 0x7 results in the highest amount of high frequency gain. Together with the other fields in this register, the default value corresponds to a long, lossy channel. Setting both RXEQZ and RXEQB to zero results in turning off receiver equalization completely. Refer to section Receiver Equalization Controls on page 7-15 for further information of Receiver Equalization. This field controls the receiver equalization for the lane(s) selected by the Lane Select (LANESEL) field in the SerDes Control (S[x]CTL) register. This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL register).
5:3	RXEQB	RW	0x7 SWSticky	**Receive Equalization Boost.** Reduces the low-frequency gain of the equalizer. A value of 0x0 results in the largest low frequency gain and smallest amount of boost. A value of 0x7 results in the smallest low frequency gain and largest amount of boost. Together with the other fields in this register, the default value corresponds to a long, lossy channel. Setting both RXEQZ and RXEQB to zero results in turning off receiver equalization completely. Refer to section Receiver Equalization Controls on page 7-15 for further information of Receiver Equalization. This field controls the receiver equalization for the lane(s) selected by the Lane Select (LANESEL) field in the SerDes Control (S[x]CTL0) register. This value is SWSticky for all lanes (i.e., even those not selected by the LANESEL field in the S[x]CTL0 register).
31:6	Reserved	RO	0x0	Reserved field.

General Purpose I/O Registers

GPIOFUNC0 - General Purpose I/O Function 0 (0x0A90)

Bit Field	Field Name	Type	Default Value	Description
31:0	GPIOFUNC	RW	0x0 SWSticky	**GPIO Function.** Each bit in this field controls the corresponding GPIO pin. When set, the corresponding GPIO pin operates as the selected alternate function. When a bit is cleared, the corresponding GPIO pin operates as a general purpose I/O pin. Bit x in this field corresponds to GPIO pin x.

GPIOFUNC1 - General Purpose I/O Function 1 (0x0A94)

Bit Field	Field Name	Type	Default Value	Description
21:0	GPIOFUNC	RW	0x0 SWSticky	**GPIO Function.** Each bit in this field controls the corresponding GPIO pin. When set, the corresponding GPIO pin operates as the selected alternate function. When a bit is cleared, the corresponding GPIO pin operates as a general purpose I/O pin. Bit x in this field corresponds to GPIO pin (x+32).
31:22	Reserved	RO	0x0	Reserved field.

GPIOAFSEL0 - General Purpose I/O Alternate Function Select 0 (0x0A98)

Bit Field	Field Name	Type	Default Value	Description
1:0	AFSEL0	RW	0x0 SWSticky	**GPIO Pin 0 Alternate Function Select.** This field selects the alternate function associated with the corresponding GPIO pin when the GPIO pin is configured to operate as an alternate function. See Chapter 11 for details. 0x0 - (afunc0) Alternate function 0 0x1 - (afunc1) Alternate function 1 0x2 - (afunc2) Alternate function 2 0x3 - (afunc3) Alternate function 3 Bit x in this field corresponds to GPIO pin x.
3:2	AFSEL1	RW	0x0 SWSticky	**GPIO Pin 1 Alternate Function Select.** See AFSEL0 field description in the GPIOAFSEL0 register.
5:4	AFSEL2	RW	0x0 SWSticky	**GPIO Pin 2 Alternate Function Select.** See AFSEL0 field description in the GPIOAFSEL0 register.
7:6	AFSEL3	RW	0x0 SWSticky	**GPIO Pin 3 Alternate Function Select.** See AFSEL0 field description in the GPIOAFSEL0 register.
9:8	AFSEL4	RW	0x0 SWSticky	**GPIO Pin 4 Alternate Function Select.** See AFSEL0 field description in the GPIOAFSEL0 register.
11:10	AFSEL5	RW	0x0 SWSticky	**GPIO Pin 5 Alternate Function Select.** See AFSEL0 field description in the GPIOAFSEL0 register.
15:12	Reserved	RO	0x0	Reserved field.
17:16	AFSEL8	RW	0x0 SWSticky	**GPIO Pin 8 Alternate Function Select.** See AFSEL0 field description in the GPIOAFSEL0 register.
31:18	Reserved	RO	0x0	Reserved field.

GPIOCFG0 - General Purpose I/O Configuration 0 (0x0AA8)

Bit Field	Field Name	Type	Default Value	Description
31:0	GPIOCFG	RW	0x0 SWSticky	**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 cleared, 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. Bit x in this field corresponds to GPIO pin x.

GPIOCFG1 - General Purpose I/O Configuration 1 (0x0AAC)

Bit Field	Field Name	Type	Default Value	Description
21:0	GPIOCFG	RW	0x0 SWSticky	**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 cleared, 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. Bit x in this field corresponds to GPIO pin (x+32).
31:22	Reserved	RO	0x0	Reserved field.

GPIOD0 - General Purpose I/O Data 0 (0x0AB0)

Bit Field	Field Name	Type	Default Value	Description
31:0	GPIOD	RW	HWINIT SWSticky	**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. Bit x in this field corresponds to GPIO pin x.

GPIOD1 - General Purpose I/O Data 1 (0x0AB4)

Bit Field	Field Name	Type	Default Value	Description
21:0	GPIOD	RW	HWINIT SWSticky	**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. Bit x in this field corresponds to GPIO pin (x+32).
31:22	Reserved	RO	0x0	Reserved field.

Hot-Plug and SMBus Interface Registers

HPSIGMAP - Hot-Plug GPIO Signal Map (0x0AB8)

Bit Field	Field Name	Type	Default Value	Description
4:0	HP0GPIOPRT	RW	0x1F SWSticky	**Hot-Plug GPIO 0 Port Map.** This field selects the PES48T12G2 port whose hot-plug signals are mapped to GPIO alternate function HP0 signals. A value of all ones (i.e., 0x1F) indicates that no port is mapped to HPx GPIO alternate function signals. When no port is mapped, the state of the hot-plug input signals is ignored and the state of the hot-plug output signals is negated.
9:5	HP1GPIOPRT	RW	0x1F SWSticky	**Hot-Plug GPIO 1 Port Map.** This field selects the PES48T12G2 port whose hot-plug signals are mapped to GPIO alternate function HP1 signals. A value of all ones (i.e., 0x1F) indicates that no port is mapped to HPx GPIO alternate function signals. When no port is mapped, the state of the hot-plug input signals is ignored and the state of the hot-plug output signals is negated.
14:10	HP2GPIOPRT	RW	0x1F SWSticky	**Hot-Plug GPIO 2 Port Map.** This field selects the switch port whose hot-plug signals are mapped to GPIO alternate function HP2 signals. A value of all ones (i.e., 0x1F) indicates that no port is mapped to HPx GPIO alternate function signals. When no port is mapped, the state of the hot-plug input signals is ignored and the state of the hot-plug output signals is negated.
19:15	HP3GPIOPRT	RW	0x1F SWSticky	**Hot-Plug GPIO 3 Port Map.** This field selects the switch port whose hot-plug signals are mapped to GPIO alternate function HP3 signals. A value of all ones (i.e., 0x1F) indicates that no port is mapped to HPx GPIO alternate function signals. When no port is mapped, the state of the hot-plug input signals is ignored and the state of the hot-plug output signals is negated.

Bit Field	Field Name	Type	Default Value	Description
24:20	HP4GPIOPRT	RW	0x1F SWSticky	**Hot-Plug GPIO 4 Port Map.** This field selects the switch port whose hot-plug signals are mapped to GPIO alternate function HP4 signals. A value of all ones (i.e., 0x1F) indicates that no port is mapped to HPx GPIO alternate function signals. When no port is mapped, the state of the hot-plug input signals is ignored and the state of the hot-plug output signals is negated.
31:25	Reserved	RO	0x0	Reserved field.

HPCFGCTL - Hot-Plug Configuration Control (0x0ABC)

Bit Field	Field Name	Type	Default Value	Description
0	IPXAPN	RW	0x0 SWSticky	**Invert Polarity of PxAPN.** When this bit is set, the polarity of the PxAPN input is inverted in all ports.
1	IPXPDN	RW	0x0 SWSticky	**Invert Polarity of PxPDN.** When this bit is set, the polarity of the PxPDN input is inverted in all ports.
2	IPXPFN	RW	0x0 SWSticky	**Invert Polarity of PxPFN.** When this bit is set, the polarity of the PxPFN input is inverted in all ports.
3	IPXMRLN	RW	0x0 SWSticky	**Invert Polarity of PxMRLN.** When this bit is set, the polarity of the PxMRLN input is inverted in all ports.
4	IPXAIN	RW	0x0 SWSticky	**Invert Polarity of PxAIN.** When this bit is set, the polarity of the PxAIN output is inverted in all ports.
5	IPXPIN	RW	0x0 SWSticky	**Invert Polarity of PxPIN.** When this bit is set, the polarity of the PxPIN output is inverted in all ports.
6	IPXPEP	RW	0x0 SWSticky	**Invert Polarity of PxPEP.** When this bit is set, the polarity of the PxPEP output is inverted in all ports.
7	IPXILOCKP	RW	0x0 SWSticky	**Invert Polarity of PxILOCKP.** When this bit is set, the polarity of the PxILOCKP output is inverted in all ports.
8	IPXPWRGDN	RW	0x0 SWSticky	**Invert Polarity of PxPWRGDN.** When this bit is set, the polarity of the PxPWRGDN input is inverted in all ports.
10:9	PDETECT	RW	0x0 SWSticky	**Presence Detect Control.** This field controls the manner in which presence of an adapter in a slot is reported to the hot-plug controller associated with a downstream switch port. 0x0 - (both) Presence of an adapter in the slot is reported as the logical “OR” of the receiver detect mechanism and the hot-plug presence detect input (PxPDN). 0x1 - (signal) Presence of an adapter in the slot is reported as the state of the hot-plug presence detect input (PxPDN). 0x2 - (always) When selected, this mode always informs the hot-plug controller that an adapter is present. 0x3 - (never) When selected, this mode always informs the hot-plug controller that an adapter is not present.

Bit Field	Field Name	Type	Default Value	Description
11	MRLPWROFF	RW	0x1 SWSticky	**MRL Automatic Power Off.** When this bit is set and the Manual Retention Latch Present (MRLP) bit is set in the PCI Express Slot Capability (PCIESCAP) register, then power to the slot is automatically turned off when the MRL sensor indicates that the MRL is open. This occurs regardless of the state of the Power Controller Control (PCC) bit in the PCI Express Slot Control (PCIESCTL) register.
12	IPXILOCKST	RW	0x0 SWSticky	**Invert Polarity of PxILOCKST.** When this bit is set, the polarity of the PxILOCKST input is inverted in all ports.
13	TEMICTL	RW	0x0 SWSticky	**Toggle Electromechanical Interlock Control.** When this bit is cleared, the Electromechanical Interlock (PxILOCKP) output is pulsed for at least 100 ms and at most 150 ms when a one is written to the EIC bit in the PCIESCTL register. When this bit is set, writing a one to the EIC register inverts the state of the PxILOCKP output (i.e., the state of the PxILOCKP signal is imply inverted and not pulsed).
15:14	RSTMODE	RW	0x0 SWSticky	**Reset Mode.** This field controls the manner in which port reset outputs are generated. 0x0 - (pec) Power enable controlled reset output 0x1 - (pgc) Power good controlled reset output 0x2 - Reserved 0x3 - Reserved
23:16	PWR2RST	RW	0x14 SWSticky	**Slot Power to Reset Negation.** This field contains the delay from stable downstream port power to negation of the downstream port reset in units of 10 mS. A value of zero corresponds to no delay. This field may be used to meet the T_{PCPERL} specification. The default value corresponds to 200 mS.
31:24	RST2PWR	RW	0x14 SWSticky	**Reset Negation.** This field contains the delay from negation of a downstream port's reset to disabling of a downstream port's power in units of 10 mS. A value of zero corresponds to no delay. The default value corresponds to 200 mS.

SMBUSSTS - SMBus Status (0x0AC8)

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.
21:8	Reserved	RO	0x0	Reserved field.

Bit Field	Field Name	Type	Default Value	Description
22	BLANK	RW1C	0x0	**Blank Serial EEPROM.** When the switch is configured to operate in a mode in which serial EEPROM initialization occurs during a Switch Fundamental Reset, this bit is set when a blank serial EEPROM is detected. Due to a small design problem, this bit may also be set when any of the following conditions is met. 1) The NUMDW[15:0] field in programmed incorrectly in a sequential double word initialization sequence stored in the EEPROM (see Figure 12.3 in Chapter 12). 2) Programming the first byte of the configuration done sequence stored in the EEPROM to a value other than 0xC0 (see Figure 12.4 in Chapter 12).
23	ROLLOVER	RW1C	0x0	**Serial EEPROM Rollover.** When the switch is configured to operate in a mode in which serial EEPROM initialization occurs during a Switch Fundamental Reset, this bit is set when a Serial EEPROM address rollover error is detected.
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 Switch 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	Reserved	RO	0x0	Reserved field.
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 (0x0ACC)

Bit Field	Field Name	Type	Default Value	Description
15:0	MSMBCP	RW	HWINIT SWSticky	**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 SWSticky	**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 SWSticky	**Ignore Checksum Errors.** When this bit is set, serial EEPROM initialization checksum errors are ignored (i.e., the checksum always passes).
21:18	Reserved	RO	0x0	Reserved field.
22	SMBDTO	RW	0x0	**SMBus Disable Time-out.** When this bit is set, SMBus timeouts are disabled on the master and slave SMBuses.
31:23	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 (0x0AD0)

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.
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

Bit Field	Field Name	Type	Default Value	Description
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.

IOEXPADDR0 - SMBus I/O Expander Address 0 (0x0AD8)

Bit Field	Field Name	Type	Default Value	Description
0	Reserved	RO	0x0	Reserved field.
7:1	IOE0ADDR	RWL	0x0 SWSticky	**I/O Expander 0 Address.** This field contains the SMBus address assigned to I/O expander 0 on the master SMBus interface.
8	Reserved	RO	0x0	Reserved field.
15:9	IOE1ADDR	RWL	0x0 SWSticky	**I/O Expander 1 Address.** This field contains the SMBus address assigned to I/O expander 1 on the master SMBus interface.
16	Reserved	RO	0x0	Reserved field.
23:17	IOE2ADDR	RWL	0x0 SWSticky	**I/O Expander 2 Address.** This field contains the SMBus address assigned to I/O expander 2 on the master SMBus interface.
24	Reserved	RO	0x0	Reserved field.
31:25	IOE3ADDR	RWL	0x0 SWSticky	**I/O Expander 3 Address.** This field contains the SMBus address assigned to I/O expander 3 on the master SMBus interface.

IOEXPADDR1 - SMBus I/O Expander Address 1 (0x0ADC)

Bit Field	Field Name	Type	Default Value	Description
0	Reserved	RO	0x0	Reserved field.
7:1	IOE4ADDR	RWL	0x0 SWSticky	**I/O Expander 4 Address.** This field contains the SMBus address assigned to I/O expander 4 on the master SMBus interface.
8	Reserved	RO	0x0	Reserved field.
15:9	IOE5ADDR	RWL	0x0 SWSticky	**I/O Expander 5 Address.** This field contains the SMBus address assigned to I/O expander 5 on the master SMBus interface.
16	Reserved	RO	0x0	Reserved field.
23:17	IOE6ADDR	RWL	0x0 SWSticky	**I/O Expander 6 Address.** This field contains the SMBus address assigned to I/O expander 6 on the master SMBus interface.
24	Reserved	RO	0x0	Reserved field.
31:25	IOE7ADDR	RWL	0x0 SWSticky	**I/O Expander 7 Address.** This field contains the SMBus address assigned to I/O expander 7 on the master SMBus interface.

IOEXPADDR2 - SMBus I/O Expander Address 2 (0x0AE0)

Bit Field	Field Name	Type	Default Value	Description
0	Reserved	RO	0x0	Reserved field.
7:1	IOE8ADDR	RWL	0x0 SWSticky	**I/O Expander 8 Address.** This field contains the SMBus address assigned to I/O expander 8 on the master SMBus interface.
8	Reserved	RO	0x0	Reserved field.
15:9	IOE9ADDR	RWL	0x0 SWSticky	**I/O Expander 9 Address.** This field contains the SMBus address assigned to I/O expander 9 on the master SMBus interface.
16	Reserved	RO	0x0	Reserved field.
23:17	IOE10ADDR	RWL	0x0 SWSticky	**I/O Expander 10 Address.** This field contains the SMBus address assigned to I/O expander 2 on the master SMBus interface.
24	Reserved	RO	0x0	Reserved field.
31:25	IOE11ADDR	RWL	0x0 SWSticky	**I/O Expander 11 Address.** This field contains the SMBus address assigned to I/O expander 3 on the master SMBus interface.

IOEXPADDR3 - SMBus I/O Expander Address 3 (0x0AE4)

Bit Field	Field Name	Type	Default Value	Description
0	Reserved	RO	0x0	Reserved field.
7:1	IOE12ADDR	RWL	0x0 SWSticky	**I/O Expander 12 Address.** This field contains the SMBus address assigned to I/O expander 12 on the master SMBus interface.
8	Reserved	RO	0x0	Reserved field.
15:9	IOE13ADDR	RWL	0x0 SWSticky	**I/O Expander 13 Address.** This field contains the SMBus address assigned to I/O expander 13 on the master SMBus interface.
31:16	Reserved	RO	0x0	Reserved field.

GPECTL - General Purpose Event Control (0x0AE8)

Bit Field	Field Name	Type	Default Value	Description
15:0	GPEE	RW	0x0 SWSticky	**General Purpose Event Enable.** Each bit in this field corresponds to a switch port. When a bit is set, the hot-plug INTx, MSI and PME event notification mechanisms defined by the PCI Express 2.0 specification are disabled for that port and are instead signaled through General Purpose Event (GPEN) signal assertions. GPEN is a GPIO alternate function.
30:16	Reserved	RO	0x0	Reserved field.

Bit Field	Field Name	Type	Default Value	Description
31	IGPE	RW	0x0 SWSticky	**Invert General Purpose Event Enable Signal Polarity.** When this bit is set, the polarity of all General Purpose Event (GPEN) signals is inverted. 0x0 - (normal) GPEN signals are active low 0x1 - (invert) GPEN signals are active high

GPESTS - General Purpose Event Status (0x0AEC)

Bit Field	Field Name	Type	Default Value	Description
15:0	GPES	RO	0x0	**General Purpose Event Status.** Each bit in this field corresponds to a switch port. When a bit is set, the corresponding port is signalling a general purpose event by asserting the GPEN signal. This bit is never set if the corresponding general purpose event is not enabled in the port's GPECTL register.
31:16	Reserved	RO	0x0	Reserved field.

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 switch: AC-coupled and DC-coupled (also called AC and DC pins). IEEE 1149.1 compliant boundary scan allows testing of the DC pins. The DC pins are the “normal” pins that do not require AC-coupling. The presence of AC-coupling capacitors on some of the device pins prevents DC values from being driven between a driver and receiver. AC Boundary Scan methodology described in IEEE 1149.6, is available to provide a time-varying signal to pass through the AC-coupling when in AC test mode. The IDT device supports both of these standards.

Test Access Point

The system logic utilizes a 16-state, TAP controller, a six-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 switch's many external signal pins. The JTAG TAP Controller can also be used for identifying the device part number. The JTAG logic of the switch is depicted in Figure 17.1.

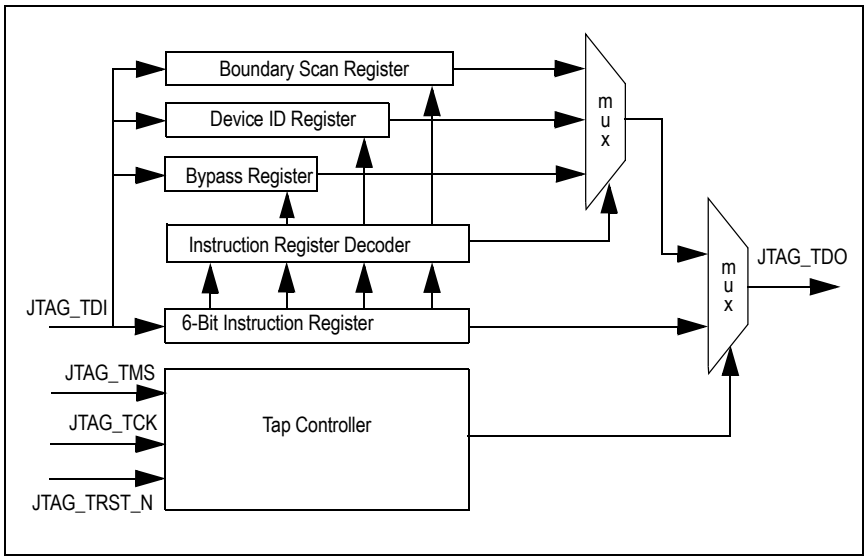


Figure 17.1 Diagram of the JTAG Logic

Refer to the IEEE 1149.1 document for an 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 17.1. 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 (active low) 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 17.1 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 17.2. 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.

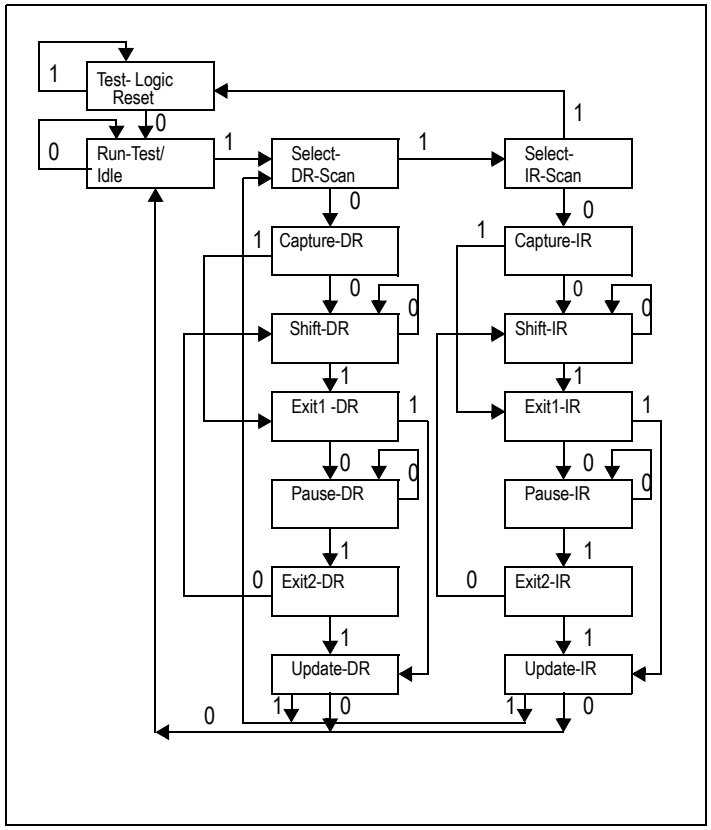


Figure 17.2 State Diagram of the TAP Controller

Boundary Scan Chain

Notes

Function	Pin Name	Type[1]	Boundary Cell[2]
PCI Express Interface	PE00RN[3:0]	I	O
	PE00RP[3:0]	I	
	PE00TN[3:0]	O	C
	PE00TP[3:0]	O	
	PE01RN[3:0]	I	O
	PE01RP[3:0]	I	
	PE01TN[3:0]	O	C
	PE01TP[3:0]	O	
	PE02RN[3:0]	I	O
	PE02RP[3:0]	I	
	PE02TN[3:0]	O	C
	PE02TP[3:0]	O	
	PE03RN[3:0]	I	O
	PE03RP[3:0]	I	
	PE03TN[3:0]	O	C
	PE03TP[3:0]	O	
	PE04RN[3:0]	I	O
	PE04RP[3:0]	I	
	PE04TN[3:0]	O	C
	PE04TP[3:0]	O	
	PE05RN[3:0]	I	O
	PE05RP[3:0]	I	
	PE05TN[3:0]	O	C
	PE05TP[3:0]	O	
	PE06RN[3:0]	I	O
	PE06RP[3:0]	I	
	PE06TN[3:0]	O	C
	PE06TP[3:0]	O	
	PE07RN[3:0]	I	O
	PE07RP[3:0]	I	
	PE07TN[3:0]	O	C
	PE07TP[3:0]	O	
	PE08RN[3:0]	I	O
	PE08RP[3:0]	I	
	PE08TN[3:0]	O	C
	PE08TP[3:0]	O	
	PE09RN[3:0]	I	O
	PE09RP[3:0]	I	
	PE09TN[3:0]	O	C
	PE09TP[3:0]	O	

Table 17.2 Boundary Scan Chain (Part 1 of 3)

Notes

Function	Pin Name	Type[1]	Boundary Cell[2]
	PE12RN[3:0]	I	O
	PE12RP[3:0]	I	
	PE12TN[3:0]	O	C
	PE12TP[3:0]	O	
	PE13RN[3:0]	I	O
	PE13RP[3:0]	I	
	PE13TN[3:0]	O	C
	PE13TP[3:0]	O	
	GCLKN[1:0]	I	—
	GCLKP[1:0]	I	
SMBus	MSMBCLK	I/O	O/C
	MSMBDAT	I/O	O/C
	SSMBADDR[2,1]	I	O
	SSMBCLK	I/O	O/C
	SSMBDAT	I/O	O/C
General Purpose I/O	GPIO[8:0]	I/O	O/C
System Pins	CLKMODE[1:0]	I	O
	GCLKFSEL	I	O
	P01MERGEN	I	
	P23MERGEN	I	
	P45MERGEN	I	
	P67MERGEN	I	
	P89MERGEN	I	
	P1213MERGEN	I	
	PERSTN	I	O
	RSTHALT	I	O
	SWMODE[3:0]	I	—
EJTAG / JTAG	JTAG_TCK	I	—
	JTAG_TDI	I	—
	JTAG_TDO	O	—
	JTAG_TMS	I	—
	JTAG_TRST_N	I	—

Table 17.2 Boundary Scan Chain (Part 2 of 3)

Notes

Function	Pin Name	Type[1]	Boundary Cell[2]
SerDes Reference Resistors	REFRES00	I/O	—
	REFRES01	I/O	—
	REFRES02	I/O	—
	REFRES03	I/O	—
	REFRES04	I/O	—
	REFRES05	I/O	—
	REFRES06	I/O	—
	REFRES07	I/O	—
	REFRES08	I/O	—
	REFRES09	I/O	—
	REFRES12	I/O	—
	REFRES13	I/O	—
	REFRESPLL	I/O	—

Table 17.2 Boundary Scan Chain (Part 3 of 3)

1. I = Input, O = Output
2. O = Observe, C = Control

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 17.3.

Notes

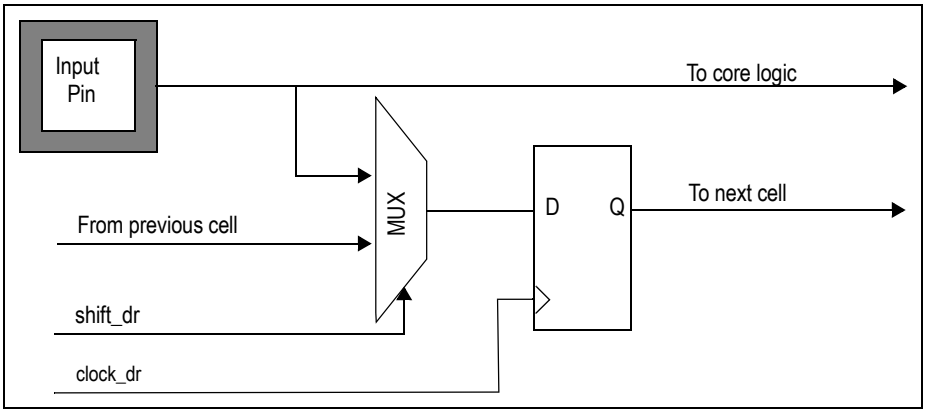


Figure 17.3 Diagram of Observe-only Input Cell

The simplified logic configuration of the output cells is shown in Figure 17.4.

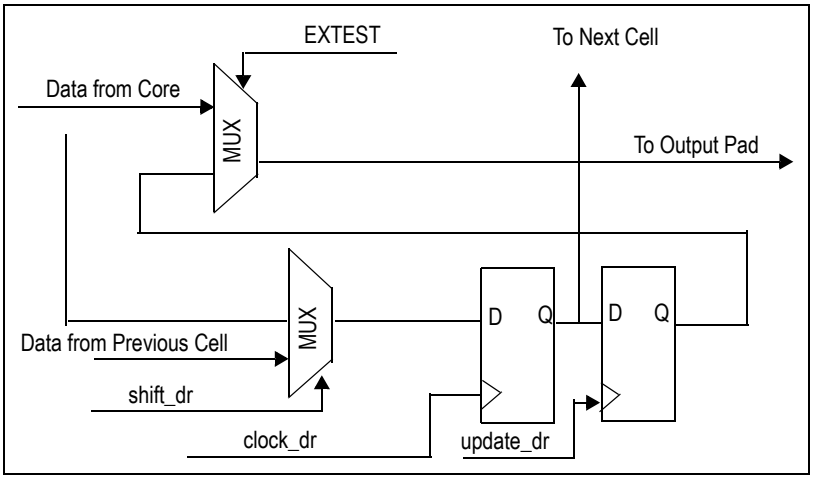


Figure 17.4 Diagram of Output Cell

The output enable cells are also output cells. The simplified logic is shown in Figure 17.5.

Notes

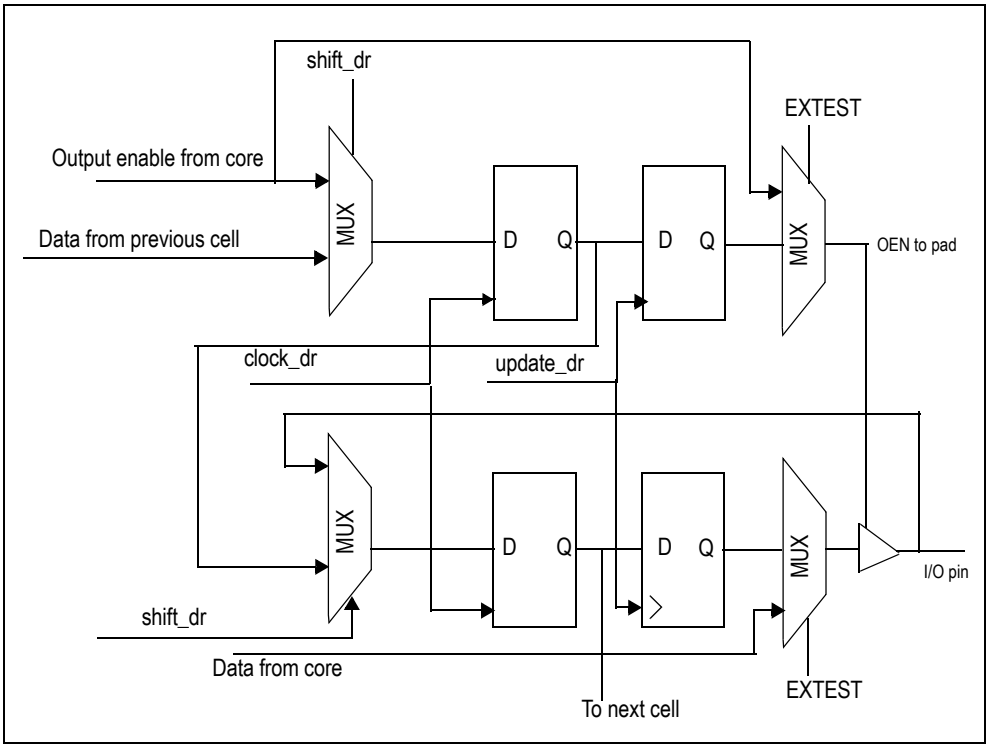


Figure 17.5 Diagram of Bidirectional Cell

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 17.5.

Instruction Register (IR)

The Instruction register allows an instruction to be shifted serially into the device 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 17.3.

Notes

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
IDCODE	Provided to select Device Identification to read out manufacturer's identity, part, and version number.	000010
HIGHZ	Tri-states all output and bidirectional boundary scan cells.	000011
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
EXTEST_TRAIN	Used for AC pin test (IEEE 1149.6 specification)	111100
EXTEST_PULSE	Used for AC pin test (IEEE 1149.6 specification)	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
All other Opcodes are RESERVED		

Table 17.3 Instructions Supported by the JTAG Boundary Scan

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 device. 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.

Notes

Therefore, instead of having to shift many times to get a value through the device, 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.

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 bypass register is selected between TDI and TDO and the scan chain passes through this register to devices further downstream.

IDCODE

The IDCODE 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 IDCODE 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 shown in Figure 17.6.

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.	R	0x807B
31:28	Version	**Version** (4 bits) This field identifies the silicon revision of the PES48T12G2.	R	silicon-dependent

Table 17.4 System Controller Device Identification Register

Version	Part Number	Mnfg. ID	LSB
xxxx	1000|0000|0111|1011	0000|0011|0011	1

Figure 17.6 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.

EXTEST_TRAIN

EXTEST_TRAIN instruction listed and explained in the IEEE 1149.6 JTAG specification. It is used to test AC pins during boundary scan by shifting data from TDI to TDO within the Shift-DR-TAP controller State. This instruction becomes effective on the falling edge of TCK in the Update-IR state.

After this instruction is asserted, the amount of time for which the pulses are generated is the amount of time for which the JTAG state machine is held in the Run-Test/Idle state.

Notes

If the Run-Test/Idle state is not entered, the output of the AC pins is not distinguishable from the output of the DC EXTEST instruction.

EXTEST_PULSE

EXTEST_PULSE is an instruction listed in IEEE 1149.6 JTAG specification and is used to test AC pins during boundary scan by shifting data from TDI to TDO within the Shift-DR-TAP controller State. This instruction becomes effective on the falling edge of TCK in the Update-IR state.

After this instruction is asserted, the width of the pulse is the amount of time for which the JTAG state machine is held in the Run-Test/Idle state.

If the Run-Test/Idle state is not entered, the output of the AC pins is not distinguishable from the output of the DC EXTEST instruction.

RESERVED

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

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.