



---

# **ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book**

Version 1.1

**June 2012**

Website [www.plxtech.com](http://www.plxtech.com)

Technical Support [www.plxtech.com/support](http://www.plxtech.com/support)

Phone 800 759-3735

408 774-9060

FAX 408 774-2169

---

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved – v1.1  
June, 2012

**NDA CONFIDENTIAL -- Inventec | King Michael**

## Revision History

Version	Date	Description of Changes
1.0	March, 2012	Production Release, Silicon Revision BA.
1.1	June, 2012	Production Release, Silicon Revision CA / Production Update, Silicon Revision BA. Changed many register bits from RO to ROS. Changed offsets 700h[0], 704h[0], and 760h[19:17] to <b>Factory Test Only</b> . For Silicon Revision CA, re-purposed offset 760h[16] to enable ECAM access to all PEX 8748 registers. Added register offset F4Ch[28:24]. Applied miscellaneous corrections and enhancements.

## Copyright Information

Copyright © 2012 PLX Technology, Inc. All Rights Reserved. The information in this document is proprietary and confidential to PLX Technology. No part of this document may be reproduced in any form or by any means or used to make any derivative work (such as translation, transformation, or adaptation) without written permission from PLX Technology.

PLX Technology provides this documentation without warranty, term or condition of any kind, either express or implied, including, but not limited to, express and implied warranties of merchantability, fitness for a particular purpose, and non-infringement. While the information contained herein is believed to be accurate, such information is preliminary, and no representations or warranties of accuracy or completeness are made. In no event will PLX Technology be liable for damages arising directly or indirectly from any use of or reliance upon the information contained in this document. PLX Technology may make improvements or changes in the product(s) and/or the program(s) described in this documentation at any time.

PLX Technology retains the right to make changes to this product at any time, without notice. Products may have minor variations to this publication, known as errata. PLX Technology assumes no liability whatsoever, including infringement of any patent or copyright, for sale and use of PLX Technology products.

PLX Technology and the PLX logo are registered trademarks and ExpressLane, *performancePAK*, Read Pacing, and *visionPAK* are trademarks of PLX Technology, Inc.

PCI Express is a trademark of the PCI Special Interest Group (PCI-SIG).

EUI-64 is a trademark of The Institute of Electrical and Electronics Engineers, Inc. (IEEE).

Tri-State is a registered trademark of Texas Instruments, Inc.

All product names are trademarks, registered trademarks, or service marks of their respective owners.

Document Number: 8748-BA/CA-SIL-DB-P1-1.1

**NDA CONFIDENTIAL -- Inventec | King Michael**

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

## Preface

The information in this data book is subject to change without notice. This PLX data book to be updated periodically as new information is made available.

## Audience

This data book provides functional details of PLX Technology's ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch, for hardware designers and software/firmware engineers. The information provided pertains to both Silicon Revisions (BA and CA), unless specified otherwise.

## Supplemental Documentation

This data book assumes that the reader is familiar with the following documents:

- PLX Technology, Inc., [www.plxtech.com](http://www.plxtech.com)  
The [PLX PEX 8748 Toolbox](#) includes this data book and other supporting documentation, *such as* errata, and design and application notes.
- The Institute of Electrical and Electronics Engineers, Inc. (IEEE), [www.ieee.org](http://www.ieee.org)
  - *IEEE Standard 1149.1-1990, IEEE Standard Test Access Port and Boundary-Scan Architecture*
  - *IEEE Standard 1149.1a-1993, IEEE Standard Test Access Port and Boundary-Scan Architecture*
  - *IEEE Standard 1149.1-1994, Specifications for Vendor-Specific Extensions*
  - *IEEE Standard 1149.6-2003, IEEE Standard Test Access Port and Boundary-Scan Architecture Extensions*
- Intel Corporation, [www.intel.com](http://www.intel.com)  
= [PHY Interface for the PCI Express Architecture, Version 2.00](#)
- NXP Semiconductors, [ics.nxp.com](http://ics.nxp.com)  
= [The I2C-Bus Specification, Version 2.1](#)
- PCI Special Interest Group (PCI-SIG), [www.pcisig.com](http://www.pcisig.com)
  - *PCI Local Bus Specification, Revision 3.0*
  - *PCI Bus Power Management Interface Specification, Revision 1.2*
  - *PCI Code and ID Assignment Specification, Revision 1.2*
  - *PCI to PCI Bridge Architecture Specification, Revision 1.2*
  - *PCI Express Base Specification, Revision 1.1*
  - *PCI Express Base Specification, Revision 2.0*
  - *PCI Express Base Specification, Revision 2.0 Errata*
  - *PCI Express Base Specification, Revision 2.1*
  - *PCI Express Base Specification, Revision 3.0*
  - *PCI Express Card Electromechanical Specification, Revision 2.0*
  - *PCI Express Mini Card Electromechanical Specification, Revision 1.1*
  - [PCI Express Architecture PCI Express Jitter and BER White Paper, Revision 1.0](#)

NDA CONFIDENTIAL -- Inventec | King Michael

- Personal Computer Memory Card International Association (PCMCIA), [www.pcmcia.org](http://www.pcmcia.org)
  - *ExpressCard Standard Release 1.0*
- PXI System Alliance (PXI), [www.pxisa.org](http://www.pxisa.org)
  - *PXI-5 PXI Express Hardware Specification, Revision 1.0*
- SBS Implementers Forum, [smbus.org](http://smbus.org)
  - *System Management Bus (SMBus) Specification, Version 2.0*

**Note:** In this data book, shortened titles are associated with the previously listed documents. The following table lists these abbreviations.

Abbreviation	Document
<i>PCI r3.0</i>	<i>PCI Local Bus Specification, Revision 3.0</i>
<i>PCI Power Mgmt. r1.2</i>	<i>PCI Bus Power Management Interface Specification, Revision 1.2</i>
<i>PCI-to-PCI Bridge r1.2</i>	<i>PCI to PCI Bridge Architecture Specification, Revision 1.2</i>
<i>PCI Express Base r1.1</i>	<i>PCI Express Base Specification, Revision 1.1</i>
<i>PCI Express Base r2.0</i>	<i>PCI Express Base Specification, Revision 2.0</i>
<i>PCI Express Base r2.1</i>	<i>PCI Express Base Specification, Revision 2.1</i>
<i>PCI Express Base r3.0</i>	<i>PCI Express Base Specification, Revision 3.0</i>
<i>PCI ExpressCard CEM r2.0</i>	<i>PCI Express Card Electromechanical Specification, Revision 2.0</i>
<i>PCI ExpressCard Mini CEM r1.1</i>	<i>PCI Express Mini Card Electromechanical Specification, Revision 1.1</i>
<i>IEEE Standard 1149.1-1990</i>	<i>IEEE Standard Test Access Port and Boundary-Scan Architecture</i>
<i>IEEE Standard 1149.6-2003</i>	<i>IEEE Standard Test Access Port and Boundary-Scan Architecture Extensions</i>
<i>I<sup>2</sup>C Bus v2.1</i> <i>I2C Bus v2.1<sup>a</sup></i>	<i>The I<sup>2</sup>C-Bus Specification, Version 2.1</i>
<i>SMBus v2.0</i>	<i>System Management Bus (SMBus) Specification, Version 2.0</i>

- a. Due to formatting limitations, the specification name may appear without the superscripted “2” in its title.

## Terms and Abbreviations

The following table lists common terms and abbreviations used in this data book. Most terms and abbreviations defined in the *PCI Express Base r3.0* are generally not included in this table.

Terms and Abbreviations	Definitions
8b/10b	Data-encoding scheme used on data transferred across a Link that is operating at either Gen 1 or Gen 2 Link speed (2.5 or 5.0 GT/s, respectively).
128b/130b	Data-encoding scheme used on data transferred across a Link that is operating at Gen 3 Link speed (8.0 GT/s).
ACK	Acknowledge Control Packet. A control packet used by a destination to acknowledge data packet receipt. A signal that acknowledges signal receipt.
AHB	Advanced High-Performance Bus.
ARI	Alternative Routing-ID Interpretation.
ARP	Address Resolution Protocol.
ATT	Attenuator.
BAR	Base Address register.
Base Mode	PEX 8748 acts as a standard PCI Express switch, supporting one Host hierarchy (equivalent to Conventional PCI mode).
BER	Bit error rate.
BIST	Built-In Self Test.
CDR	Clock and Data Recovery.
CMU	Clock Management Unit.
CRC	Cyclic Redundancy Check.
CSR	Configuration Space register.
Data Beat	Single data transfer in a single clock period.
DFE	Decision Feedback Equalization. Optionally added to a Receiver to improve the signal.
DLL	Data Link Layer.
Downstream Device	Device that is connected to a Downstream Port.
Downstream Port	Port that is used to communicate with a device below it in the system hierarchy. A switch can have one or more Downstream Ports.
DRI	<i>Data Rate Identifier</i> field in Training Sets.
ECC	Error-Correcting Code.
ECRC	End-to-end Cyclic Redundancy Check.
EDB	TLP framing symbol used in place of an END symbol to indicate that the TLP is bad.
EIES	Electrical Idle Exit Sequence.
EIOS	Electrical Idle Ordered-Set.
Electrical Idle	Transmitter is in a High-Impedance state (+ and - are both at common mode voltage).
EP	Endpoint.
FC	Flow Control.
Field	Multiple register bits that are combined for a single function.
FTS	Fast Training Sequence.

NDA CONFIDENTIAL -- Inventec | King Michael

Terms and Abbreviations	Definitions
Gen 1	<i>PCI Express Base r1.1</i> and below. Link transfer rate of 2.5 GT/s.
Gen 2	<i>PCI Express Base r2.1</i> . Link transfer rate of 5.0 GT/s.
Gen 3	<i>PCI Express Base r3.0</i> . Link transfer rate of 8.0 GT/s.
GPIO	General-Purpose Input/Output.
GPU	Graphics Processing Unit.
GT/s	Giga-Transfers per second.
INCH	Ingress Credit Handler.
ISR	Interrupt Service Routine.
Lane	Bidirectional pair of differential PCI Express I/O signals.
LCRC	Link Cyclic Redundancy Check.
LFSR	Linear Feedback Shift register.
Link	Active connection between two Ports.
Link Interface	Primary side of the NT Port, connects to external device pins. The secondary side of the NT Port is referred to as the <i>NT Port Virtual Interface</i> , and connects to the internal virtual PCI Express interface.
Local	Reference to PCI Express attributes ( <i>such as</i> credits) that belong to the PCI Express Station.
LTSSM	Link Training and Status State Machine.
LUT	Lookup Table.
LVDS	Low-Voltage Differential Signaling.
MPS	Maximum Payload Size.
MRL	Manually operated Retention Latch.
NACK	Negative Acknowledge. Used in the SMBus-related content.
NAK	Negative Acknowledge.
N_FTS	Number of Fast Training Sequences field in Training Sets.
NOP	No Operation.
NT	Non-Transparent. A bridging technique used in the PCI Express Switch to isolate Memory spaces by presenting the processor as an endpoint rather than another memory system. The PEX 8748 supports one NT Port.
OS	Ordered-Set.
P2P	Peer-to-Peer or PCI-to-PCI (as identified at point of use).
PCH	Intel Platform Controller Hub.
PCI Express Station	Functional unit that provides the PCI Express conforming system interface. Includes the Serializer/De-Serializer (SerDes) hardware interface modules and PCI Express interface, which provides the Physical Layer (PHY), Data Link Layer (DLL), and Transaction Layer (TL) logic.
PCS	Physical Coding Sublayer.
PEC	Packet Error Code.
PEX	PCI Express.
PHY	Physical Layer.
PIPE	PHY Interface for PCI Express architecture.

Terms and Abbreviations	Definitions
PLL	Phase-Locked Loop.
PM	Power Management.
PMA	Physical Media Attachment layer.
PME	Power Management Event.
PN	Port Number.
Port	Interface to a group of SerDes and supporting logic that is capable of creating a Link, for communication with another Port.
Port ID	Number, assigned in hardware, that associates a SerDes with a Port.
P-P	PCI-to-PCI.
PRBS	Pseudo-Random Bit Sequence.
QoS	Quality of Service.
RAS	Reliability, Availability, and Serviceability.
RoHS	Restrictions on the use of certain Hazardous Substances (RoHS) Directive.
RR	Round-Robin scheduling.
Rx	Receiver.
SATA	Serial ATA. Computer bus interface used for connecting Host Bus adapters to mass storage devices, <i>such as</i> hard disk and optical drives.
SerDes	Serializer/De-Serializer. A high-speed differential-signaling parallel-to-serial and serial-to-parallel conversion logic attached to Lane pads.
SMBus	System Management Bus.
SN	SerDes Number.
SPI	Serial Peripheral Interface.
SSC	Spread-Spectrum Clock.
Station	Logic block that implements the PCI Express function, bounded by the external pins of the differential Transceivers and the interface to the internal switch fabric.
Sticky Bits	Register bits in which the current values are unchanged by a Hot Reset, Link Down event or a Secondary Bus Reset, while the switch is powered. Sticky bits are reset to default values by a Fundamental Reset. HwInit, ROS, RWICS, and RWS CSR types. (Refer to <a href="#">Table 13-4, “Register Types, Grouped by User Accessibility,”</a> for CSR type definitions.)
Sticky State	Condition that causes a state machine to be stuck in a particular state, unable to make forward progress.

Terms and Abbreviations	Definitions
TC	Traffic Class.
TCB	Training Control Bits field in Training Sets.
TDM	Time Division Multiplex.
TL	Transaction Layer.
TLC	Transaction Layer Control. The module performing PCI Express Transaction Layer functions.
TLP	Transaction Layer Packet. PCI Express packet formation and organization.
Transparent	Refers to standard PCI Express Upstream-to-Downstream routing protocol.
TS	Tri-statable Output (High-Impedance when output is not driven). (Used in <a href="#">Chapter 3</a> , “Signal Ball Description,” in the <b>Type</b> column of the Signal tables.)
TS	Training Sequence.
TS1	Type 1 Training Sequence Ordered-Set.
TS2	Type 2 Training Sequence Ordered-Set.
Tx	Transceiver.
UDID	Unique Device Identifier.
UI	Unit Interval – 400 ps at 2.5 GT/s, 200 ps at 5.0 GT/s, 125 ps at 8.0 GT/s.
Upstream Device	Device that is connected to the Upstream Port(s).
Upstream Port	Port that is used to communicate with a device above it in the system hierarchy.
UTP	User Test Pattern.
VC	Virtual Channel. The PEX 8748 supports one Virtual Channel – VC0.
VC&T	Virtual Channel and Type.
Vector	Address and data.
Virtual Interface	Secondary side of the NT Port, connects to the internal virtual PCI Express interface.
Virtual Switch	Partition of a PCI Express switch that has its own virtual hierarchy. The PEX 8748 can support up to six virtual switches – VS0, VS1, VS2, VS3, VS4, and VS5. Each virtual switch is connected to its own Host.
Virtual Switch Mode	PEX 8748 supports up to six Hosts, creating up to six virtual switches within the PEX 8748 – each with its own virtual hierarchy.
VS	Virtual Switch.
WRR	Weighted Round-Robin scheduling.



## Data Book Notations and Conventions

Notation / Convention	Description
<a href="#">Blue text</a>	Indicates that the text is hyperlinked to its description elsewhere in the data book. Left-click the blue text to learn more about the hyperlinked information. This format is often used for register names, register bit and field names, register offsets, chapter and section titles, figures, and tables.
PEX_XXXn[x] PEX_XXXp[x]	When the signal name appears in all CAPS, with the primary Port description listed first, field [x] indicates the number associated with the signal balls/pads assigned to a specific SerDes module/Lane. The lowercase “n” (negative) or “p” (positive) suffix indicates the differential pair of signals, which are always used together.
# = Active-Low signals	Unless specified otherwise, Active-Low signals are identified by a “#” appended to the term ( <i>for example</i> , PEX_PERST#).
Program/code samples	Monospace font ( <i>program or code samples</i> ) is used to identify code samples or programming references. These code samples are case-sensitive, unless specified otherwise.
command_done	Interrupt format.
<b>Command/Status</b>	Register names.
<i>Parity Error Detected</i>	Register parameter [bit or field] or control function.
Upper Base Address[31:16]	Specific Function in 32-bit register bounded by bits [31:16].
Number multipliers	k = 1,000 ( $10^3$ ) is generally used with frequency response. K = 1,024 ( $2^{10}$ ) is used for Memory size references. KB = 1,024 bytes. M = meg. = 1,000,000 when referring to frequency (decimal notation) = 1,048,576 when referring to Memory sizes (binary notation)
255d	d = Suffix that identifies decimal values.
1Fh	h = Suffix that identifies hex values. Each prefix term is equivalent to a 4-bit binary value (Nibble). Legal prefix terms are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.
1010b	b = suffix which identifies binary notation ( <i>for example</i> , 01b, 010b, 1010b, and so forth). Not used with single-digit values of 0 nor 1.
0 through 9	Decimal numbers, or single binary numbers.
byte	Eight bits – abbreviated to “B” ( <i>for example</i> , 4B = 4 bytes).
LSB	Least-Significant Byte.
lsb	Least-significant bit.
MSB	Most-Significant Byte.
msb	Most-significant bit.
DWord or DW	Double-Word (32 bits) is the primary register size in these devices.
QWord	Quad-Word (64 bits).
<b>Reserved</b>	Do not modify <b>Reserved</b> bits and words. Unless specified otherwise, these bits read as 0 and must be written as 0.
word	16 bits.

THIS PAGE INTENTIONALLY LEFT BLANK.

**NDA CONFIDENTIAL -- Inventec | King Michael**

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

# Contents

<b>Chapter 1</b>	<b>Introduction</b>	<b>1</b>
	1.1 Overview	1
	1.2 Features	2
<b>Chapter 2</b>	<b>Features and Applications</b>	<b>5</b>
	2.1 Flexible and Feature-Rich 48-Lane, 12-Port Switch	5
	2.1.1 Highly Flexible Port Configurations	5
	2.1.2 Non-Blocking Crossbar Switch Architecture	6
	2.1.3 Multi-Host Architecture – NT and Virtual Switch Modes	7
	2.1.3.1 Dual-Host and Failover Support – NT Mode	7
	2.1.3.2 Virtual Switch Mode (Multi-Host) and Failover Support	8
	2.1.4 Low Packet Latency and High Performance	9
	2.1.4.1 Data Payloads	9
	2.1.4.2 Cut-Thru Mode	9
	2.1.5 Virtual Channel and Traffic Classes	10
	2.1.6 Data Integrity	10
	2.1.7 Configuration Flexibility	10
	2.1.8 Interoperability	10
	2.1.9 Low Power with Granular SerDes Control	10
	2.1.10 Dynamic Lane Reversal	11
	2.1.11 Hot Plug for High Availability	11
	2.1.12 Fully Compliant Power Management	11
	2.1.13 General-Purpose Input/Output Signals	11
	2.1.14 <i>performancePAK</i>	12
	2.1.14.1 Read Pacing	12
	2.1.14.2 Multicast	12
	2.1.14.3 Dynamic Buffer Pool	12
	2.1.15 <i>visionPAK</i>	13
	2.1.15.1 Performance Monitoring	13
	2.1.15.2 Error Injection	13
	2.1.15.3 SerDes Loopback	13
	2.1.15.4 SerDes Eye Capture	13
	2.2 Applications	14
	2.2.1 Host-Centric Fan-Out	14
	2.2.2 Multi-Host Systems	15
	2.2.3 Host Failover	16
	2.2.4 N+1 Failover in Storage Systems	17
	2.3 Software Usage Model	18
	2.3.1 System Configuration	18
	2.3.2 Interrupt Sources and Events	18

<b>Chapter 3</b>	<b>Signal Ball Description</b>	<b>19</b>
3.1	Introduction	19
3.2	Ball Assignment Abbreviations	20
3.3	Internal Pull-Up/Pull-Down Resistors	21
3.4	Signal Ball Descriptions	22
3.4.1	PCI Express Signals	23
3.4.2	Hot Plug Signals	25
3.4.2.1	Parallel Hot Plug Signals	25
3.4.2.2	Serial Hot Plug Signals	34
3.4.3	Serial EEPROM Signals	36
3.4.4	Strapping Signals	37
3.4.4.1	STRAP_TESTMODE[2:0] 3-State Input Configurations	48
3.4.5	JTAG Interface Signals	52
3.4.6	I <sup>2</sup> C/SMBus Slave Interface Signals	53
3.4.7	Device-Specific Signals	55
3.4.8	External Resistor Signals	66
3.4.9	Thermal Diode Signals	67
3.4.10	No Connect Signals	68
3.4.11	Power and Ground Signals	70
3.5	Physical Layout	71
<b>Chapter 4</b>	<b>Functional Overview</b>	<b>73</b>
4.1	Introduction	73
4.2	Modes of Operation	74
4.2.1	Base Mode	74
4.2.2	Virtual Switch Mode	75
4.3	Software Architecture	76
4.3.1	PCI-Compatible Software Model	76
4.3.2	PCI Express Memory-Mapped Configuration Space	77
4.3.3	NT Bridge – NT Mode	79
4.4	Hardware Architecture	80
4.4.1	Station and Port Functions	81
4.4.1.1	Port Configurations	81
4.4.1.2	Port Numbering	83
4.5	PCI Express Station Functional Description	84
4.6	Physical Layer	85
4.6.1	Physical Layer Features	86
4.6.2	Physical Layer Status and Command Registers	87
4.6.3	Hardware Link Interface Configuration	87
4.7	Transaction Layer	88
4.7.1	Locked Transactions	89
4.7.2	Relaxed Ordering	89
4.7.3	TL Transmit/Egress Protocol – End-to-End Cyclic Redundancy Check	89
4.7.4	TL Receive/Ingress Protocol	90
4.7.5	Flow Control Credit Initialization	90
4.7.6	Flow Control Protocol	90
<b>Chapter 5</b>	<b>Reset and Initialization</b>	<b>91</b>
5.1	Reset	91
5.1.1	Resets – Base Mode	91
5.1.1.1	Fundamental Reset – Base Mode	92
5.1.1.2	Hot Reset – Base Mode	92
5.1.1.3	Secondary Bus Reset – Base Mode	92
5.1.1.4	Register Bits that Affect Hot Reset – Base Mode	92

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

5.1.2	Resets – Virtual Switch Mode	93
5.1.2.1	Conventional Reset – Virtual Switch Mode	93
5.1.2.2	VSx_PERST# (Hot Reset)	93
5.1.2.3	Inband Reset (TS1 Ordered-Set) or Upstream Port DL_Down (Hot Reset) – Virtual Switch Mode	93
5.1.2.4	Secondary Bus Reset (Soft Reset) – Virtual Switch Mode	94
5.1.2.5	Reset Propagation Prevention – Virtual Switch Mode	94
5.1.3	Reset and Clock Initialization Timing	95
5.2	Initialization – Base Mode	96
5.2.1	Serial EEPROM Load Time	96
5.2.2	I <sup>2</sup> C Load Time	96
5.3	Initialization – Virtual Switch Mode	97
5.3.1	Management Port Policies	98
5.3.2	Active and Redundant Management Ports	99
5.3.3	Virtual Switch Table	100
5.3.3.1	Virtual Switch Table Registers	101
5.3.3.2	Virtual Switch Table Programming Sequence	102
5.3.4	Port Activity Vector	104
5.3.5	Link-Related Registers	104
5.3.6	Reconfiguration of Virtual Switches	105
5.3.6.1	Graceful De-Allocation of Downstream Port	105
5.3.6.2	Surprise Removal of Downstream Device	106
5.3.6.3	Graceful De-Allocation of Upstream Port	107
5.3.6.4	Surprise Removal of Upstream Port	107
5.3.6.5	Management-Capable Port Switch Over	107
<b>Chapter 6</b>	<b>Serial EEPROM Controller</b>	<b>109</b>
6.1	Overview	109
6.2	Features	110
6.3	Serial EEPROM Load	111
6.3.1	Serial EEPROM Load – Base Mode	111
6.3.2	Serial EEPROM Load – Virtual Switch Mode	112
6.4	Serial EEPROM Data Format	114
6.4.1	Serial EEPROM Cyclic Redundancy Check	115
6.5	Serial EEPROM Initialization	119
6.6	Serial EEPROM Registers	119
6.7	Serial EEPROM Random Write/Read Access	120
6.7.1	Writing to Serial EEPROM	120
6.7.2	Reading from Serial EEPROM	121
6.7.3	Programming a Blank Serial EEPROM	121
6.8	Serial EEPROM Loading of NT Port Link Interface Registers – NT Mode	122
6.9	NT Port Expansion ROM – NT Mode	122
<b>Chapter 7</b>	<b>I<sup>2</sup>C/SMBus Slave Interface Operation</b>	<b>123</b>
7.1	Introduction	123
7.2	I <sup>2</sup> C Slave Interface	123
7.2.1	I <sup>2</sup> C Support Overview	123
7.2.2	I <sup>2</sup> C Addressing – Slave Mode Access	125
7.2.3	I <sup>2</sup> C Command Format	125
7.2.4	I <sup>2</sup> C Register Write Access	126
7.2.4.1	I <sup>2</sup> C Register Write Example	130

NDA CONFIDENTIAL -- Inventec | King Michael

7.2.5	I <sup>2</sup> C Register Read Access	132
7.2.5.1	I <sup>2</sup> C Register Read Address Example – Phase and Command Packet	135
7.3	SMBus Slave Interface	136
7.3.1	SMBus Features	136
7.3.2	SMBus Operation	136
7.3.3	Supported SMBus Commands	137
7.3.3.1	SMBus Block Write	138
7.3.3.2	SMBus Block Read	141
7.3.3.3	CSR Read, Using SMBus Block Write - Block Read Process Call	144
7.3.4	SMBus Address Resolution Protocol	145
7.3.4.1	SMBus UDID	146
7.3.4.2	Supported SMBus ARP Commands	148
7.3.5	SMBus PEC Handling	150
7.3.6	Addressing PEX 8748 SMBus Slave	150
7.3.7	SMBus Timeout	151
7.4	Switching between SMBus and I <sup>2</sup> C Bus Protocols	151

<b>Chapter 8</b>	<b>Performance Features</b>	<b>153</b>
8.1	Introduction	153
8.2	DLLP Policies	154
8.2.1	ACK DLLP Policy	154
8.2.2	UpdateFC DLLP Policy	156
8.2.3	Unidirectional DLLP Policies	156
8.3	Latency	157
8.4	Queuing Options	158
8.4.1	Destination Queuing	159
8.4.2	Source Queuing	160
8.4.3	Port Arbitration	161
8.4.4	Port Bandwidth Allocation	162
8.5	Read Pacing	163
8.5.1	Read Pacing Example	164
8.5.2	Read Spacing (Spreading) Logic	166
8.5.3	Read Threshold	167
8.5.4	Read Pacing Benefits	167
8.5.5	Enabling Read Pacing and Read Spreading	168
8.6	Multicast	169
8.6.1	Multicast Address Range Segmentation	170
8.6.2	Multicast TLP Processing	171
8.6.3	Multicast Ordering	172
8.6.4	Multicast Extended Capability Structure Field Updates	172
8.6.5	MC Blocked TLP Processing	172
8.6.6	MC_Overlay and ECRC Re-Generation	173
8.6.6.1	Multicast to Endpoints without Multicast Extended Capability	174
8.6.6.2	Congestion Avoidance	174
8.6.7	Multicast Extended Capability	174
8.6.8	Multicast with NT – NT Mode	175
8.6.8.1	NT Multicast from NT Port Virtual to NT Port Link Interface Direction (NT Target) – NT Mode	175
8.6.8.2	NT Multicast from NT Port Link to NT Port Virtual Interface Direction (NT Source) – NT Mode	176

<b>Chapter 9</b>	<b>Interrupts</b>	<b>177</b>
9.1	Interrupt Support	177
9.1.1	Interrupt Sources or Events	179
9.1.2	Interrupt Handling	184
9.1.2.1	Interrupt Handling – Base Mode	184
9.1.2.2	Interrupt Handling – Virtual Switch Mode	184
9.2	INTx Emulation Support	185
9.2.1	INTx-Type Interrupt Message Re-Mapping and Collapsing	186
9.2.1.1	Interrupt Re-Mapping and Collapsing – NT Mode	187
9.3	MSI Support	188
9.3.1	MSI Operation	188
9.3.1.1	Message Signaled Interrupts – NT Mode	189
9.3.2	MSI Capability Registers	189
9.4	PEX_INTA# or VSx_PEX_INTA# Interrupts	190
9.5	General-Purpose Input/Output	191
9.6	Management Port Interrupts – Virtual Switch Mode	193
9.6.1	Switch Port Link Status Events – Virtual Switch Mode	193
9.6.1.1	Special Handling for Race Conditions – Virtual Switch Mode	194
9.6.2	Doorbell Interrupts – Virtual Switch Mode	194
<b>Chapter 10</b>	<b>Hot Plug Support</b>	<b>195</b>
10.1	Introduction	195
10.2	Hot Plug Features	196
10.3	Hot Plug Elements	197
10.4	Hot Plug Signals	198
10.4.1	Hot Plug Port External Signals	198
10.4.2	Hot Plug Output States for Disabled Hot Plug Slots	198
10.5	Hot Plug Registers	199
10.6	Hot Plug Interrupts	199
10.6.1	Software Testing of Hot Plug Interrupts	199
10.7	Hot Plug and Virtual Switch Mode Failover	200
10.7.1	Virtual Switch Mode Failover and Upstream Port Hot Plug Functionality	200
10.8	Hot Plug Controller Slot Power-Up/Down Sequence	201
10.8.1	Slot Power-Up Sequence	201
10.8.1.1	Configuring Slot Power-Up Sequence Features with Serial EEPROM	202
10.8.1.2	Slot Power-Up Sequencing when Power Controller Present Bit Is Set	203
10.8.1.3	HP_PERST_x# (Reset) and HP_PWRLED_x# Output Power-Up Sequencing when Power Controller Present Bit Is Cleared	206
10.8.1.4	Disabling Power-Up Hot Plug Output Sequencing	207
10.8.2	Slot Power-Down Sequence	207
10.9	Default Parallel Hot Plug Ports	208
10.10	HP_PWREN_x Output Polarity Control	212
10.11	Serial Hot Plug Controller	213
10.11.1	Hot Plug Operations by way of I <sup>2</sup> C I/O Expander	214
10.11.2	I <sup>2</sup> C I/O Expander Parts Selection and Pin Definition	215
10.11.3	Serial Hot Plug Port Enumeration, Assignment, and Initialization	221
10.11.4	I <sup>2</sup> C I/O Expander Interrupt Processing	222
10.11.5	Serial Hot Plug-Capable Port Command Completion	222
10.11.6	Physical Slot Number Loading from I <sup>2</sup> C I/O Expander	222
10.12	Hot Plug Board Insertion and Removal Process	223

<b>Chapter 11</b>	<b>Power Management</b>	<b>227</b>
11.1	Overview	227
11.2	Power Management Features	228
11.3	Power Management Capability	229
11.3.1	Device Power Management States	229
11.3.1.1	D0 State	229
11.3.1.2	D3hot State	229
11.3.2	Link Power Management States	230
11.3.3	Active State Power Management – L0s Link PM State Transition	231
11.3.3.1	Rx L0s Link PM State Entry	231
11.3.3.2	Rx L0s Link PM State Exit	231
11.3.3.3	Tx L0s Link PM State Entry Conditions	231
11.3.3.4	L0s Link PM State Exit Conditions	233
11.3.3.5	L1 Link PM State Entry	234
11.3.3.6	L1 Link PM State Exit Conditions	237
11.3.3.7	L1 Link PM State Entry Negotiation Interruption – ASPM and PCI Express PCI-PM	238
11.3.3.8	L2/L3 Link PM State Entry	239
11.4	Power Management Tracking	241
11.5	Power Management Event Handler	242
11.6	Power Management – Virtual Switch Mode	243
<b>Chapter 12</b>	<b>Virtual Switch Mode</b>	<b>245</b>
12.1	Multiple Virtual Switches	245
12.1.1	Default Virtual Switch Port Assignments	246
12.1.2	Combined NT and Multiple Virtual Switches	248
12.2	Management Port	249
12.2.1	Out-of-Band Interfaces	249
12.2.1.1	Unused PCI Express Port – Management-Capable Port	249
12.2.1.2	Strapping Balls	249
12.2.1.3	Serial EEPROM	249
12.2.1.4	I <sup>2</sup> C Bus/SMBus	249
12.2.2	In-Band Interface	250
12.2.2.1	Configuration and Management	250
12.2.2.2	In-Band Management Port	251
12.2.2.3	Management Ports and Restriction	252
12.3	Virtual Switch Reset and Initialization	254
12.3.1	Virtual Switch Reset	254
12.3.2	Virtual Switch Initialization	254
12.3.3	Virtual Switch Table Programming Sequence	254
12.4	Port Migration – Moving a Port from One Virtual Switch to Another (VSx to VSy)	254
12.5	Failover in Virtual Switch Mode	255
12.5.1	Virtual Switch Host Failover	255
12.5.2	Active Management Port Failover	255
12.5.3	Virtual Switch Host to NT Host Failover	256
12.6	Performance	257
12.7	Host-to-Host Communication	257



<b>Chapter 13</b>	<b>Transparent Port Registers</b>	<b>259</b>
13.1	Introduction	259
13.2	Type 1 Port Register Map	260
13.3	Port Register Configuration and Map	262
13.4	Register Access	265
13.4.1	PCI r3.0-Compatible Configuration Mechanism	265
13.4.2	PCI Express Enhanced Configuration Access Mechanism	267
13.4.3	Device-Specific Memory-Mapped Configuration Mechanism	267
13.5	Register Descriptions	269
13.6	Port Configurations and Station/Station Port/Port/ SerDes Module/Lane Relationship	270
13.6.1	Port Configurations	270
13.6.2	Virtual Switch Port Assignments – Virtual Switch Mode	272
13.7	PCI-Compatible Type 1 Configuration Header Registers (Offsets 00h – 3Ch)	274
13.8	PCI Power Management Capability Registers (Offsets 40h – 44h)	290
13.9	Message Signaled Interrupt Capability Registers (Offsets 48h – 64h)	293
13.10	PCI Express Capability Registers (Offsets 68h – A0h)	300
13.11	Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – FCh)	334
13.12	Device Serial Number Extended Capability Registers (Offsets 100h – 108h)	335
13.13	Secondary PCI Express Extended Capability Registers (Offsets 10Ch – 134h)	337
13.14	Power Budget Extended Capability Registers (Offsets 138h – 144h)	364
13.15	Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)	367
13.15.1	WRR Port Arbitration Table Registers (Offsets 178h – 1BCh)	373
13.16	Device-Specific Registers (Offsets 1C0h – A74h)	390
13.16.1	Device-Specific Registers – Read Pacing (Offsets 1D0h – 1D8h)	392
13.16.2	Device-Specific Registers – Captured Bus and Device Numbers (Offsets 1DCh – 1E4h)	396
13.16.3	Device-Specific Registers – Read Pacing (Offset 1E8h)	397
13.16.4	Device-Specific Registers – Physical Layer (Offsets 200h – 25Ch)	398
13.16.5	Device-Specific Registers – Serial EEPROM (Offsets 260h – 270h)	433
13.16.6	Device-Specific Registers – I <sup>2</sup> C and SMBus Slave Interfaces (Offsets 290h – 2FCh)	439
13.16.7	Device-Specific Registers – Port Configuration and Lane Status (Offsets 300h – 350h)	442
13.16.8	Device-Specific Registers – Port Configuration (Offsets 354h – 46Ch)	452
13.16.9	Device-Specific Registers – General-Purpose Input/Output (Offsets 600h – 68Ch)	477

NDA CONFIDENTIAL -- Inventec | King Michael

13.16.10	Device-Specific Registers – PORT_GOOD Selector (Offsets 6A0h – 6A8h) . . . . .	511
13.16.11	Device-Specific Registers – Error Checking and Debug (Offsets 700h – 75Ch) . . . . .	515
13.16.12	Device-Specific Registers – Control (Offsets 760h – 774h) . . . . .	543
13.16.13	Device-Specific Registers – Soft Error (Offsets 778h – 8FCh) . . . . .	549
13.16.14	Device-Specific Registers – Virtual Switch (Offsets 900h – 9E8h), Virtual Switch Mode . . . . .	560
13.16.15	Device-Specific Registers – Ingress Credit Handler (Offsets 9ECh – A2Ch) . . . . .	574
13.16.16	Device-Specific Registers – Virtual Switch Debug and GPIO Status and Control (Offsets A30h – A74h) . . . . .	583
13.17	Latency Tolerance Reporting Extended Capability Registers (Offsets B00h – B08h) . . . . .	609
13.18	Device-Specific Registers (Offsets B70h – DFCh) . . . . .	612
13.18.1	Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets B70h – B7Ch) . . . . .	613
13.18.2	Device-Specific Registers – Physical Layer (Offsets B80h – C88h) . . . . .	615
13.18.3	Device-Specific Registers – Requester ID Read Back (Offsets C8Ch – C90h) . . . . .	635
13.19	Multicast Extended Capability Registers (Offsets E00h – E2Ch) . . . . .	636
13.20	Device-Specific Registers – Virtual Switch (Offsets F00h – F20h), Virtual Switch Mode . . . . .	643
13.21	ACS Extended Capability Registers (Offsets F24h – F2Ch) . . . . .	646
13.22	Device-Specific Registers (Offsets F30h – FB0h) . . . . .	650
13.22.1	Device-Specific Registers – Egress Control (Offsets F30h – F44h) . . . . .	651
13.22.2	Device-Specific Registers – Ingress Control and Port Enable (Offsets F48h – F6Ch) . . . . .	655
13.22.3	Device-Specific Registers – Error Checking and Debug (Offsets F70h – FB0h) . . . . .	662
13.23	Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh) . . . . .	669

## **Chapter 14 Non-Transparent Bridging – NT Mode . . . . .681**

14.1	Introduction . . . . .	681
14.1.1	NT Device Type Identification . . . . .	682
14.1.2	Enabling Non-Transparent Bridging . . . . .	682
14.1.3	NT Port Features . . . . .	683
14.1.4	Intelligent Adapter Mode . . . . .	684
14.1.5	NT Port Reset . . . . .	685
14.1.5.1	Fundamental Reset (PERST#) . . . . .	685
14.1.5.2	Virtual Side Hot Reset . . . . .	685
14.1.5.3	Link Side Hot Reset . . . . .	686
14.1.5.4	Reset Propagation . . . . .	686
14.1.6	NT Port Memory-Mapped Base Address Registers . . . . .	687
14.1.7	Doorbell Registers . . . . .	689

**NDA CONFIDENTIAL -- Inventec | King Michael**

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

14.1.8	Scratchpad Registers	690
14.1.9	NT Base Address Registers	691
14.1.9.1	NT BARx Setup Registers	691
14.1.10	TLP Routing	693
14.1.11	Address Translation	694
14.1.11.1	Direct Address Translation	695
14.2	Requester ID Translation	697
14.2.1	Transaction Sequence	698
14.3	NT Port Power Management Handling	700
14.3.1	Active State Power Management	700
14.3.1.1	L0s Link PM State Transition – NT Port Link Interface	700
14.3.1.2	L1 Link PM State Entry – NT Port Link Interface	701
14.3.1.3	L2/L3 Link PM State Entry – NT Port Link Interface	703
14.3.2	PCI-PM and PME Turn Off Support	704
14.3.3	Message Generation	704
14.4	Expansion ROM	705
14.5	NT Port Interrupts	705
14.5.1	NT Port Virtual Interface Interrupts	705
14.5.2	NT Port Link Interface Interrupts	706
14.6	NT Port Error Handling	707
14.6.1	NT Port Link Interface Error Handling	707
14.6.2	NT Virtual Side Error Handling	708
14.7	Cursor Mechanism	709
14.8	Port Programmability	709
14.9	NT Failover	710
14.9.1	Active-Passive NT Failover	710
14.9.2	Active-Active NT Failover	712
14.9.3	NT Failover for Virtual Switch Operation	715
14.9.4	Failover Completion	716
<b>Chapter 15</b>	<b>NT Port Virtual Interface Registers – NT Mode</b>	<b>717</b>
15.1	Introduction	717
15.2	NT Port Virtual Interface Type 0 Register Map	718
15.3	Register Access	720
15.3.1	PCI Express Base r3.0 Configuration Mechanism	720
15.3.1.1	PCI r3.0-Compatible Configuration Mechanism	720
15.3.1.2	PCI Express Enhanced Configuration Access Mechanism	721
15.3.2	Device-Specific Memory-Mapped Configuration Mechanism	722
15.3.3	Device-Specific Cursor Mechanism	724
15.4	Register Descriptions	725
15.5	NT Port Virtual Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch)	726
15.6	NT Port Virtual Interface PCI Power Management Capability Registers (Offsets 40h – 44h)	736
15.7	NT Port Virtual Interface MSI Capability Registers (Offsets 48h – 64h)	739
15.8	NT Port Virtual Interface PCI Express Capability Registers (Offsets 68h – A0h)	745
15.9	NT Port Virtual Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh)	758
15.10	NT Port Virtual Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)	764
15.11	NT Port Virtual Interface Device-Specific Registers (Offsets 1C0h – A74h)	767

NDA CONFIDENTIAL -- Inventec | King Michael

15.11.1	NT Port Virtual Interface Device-Specific Registers – Captured Bus Number and Device Number (Offsets 1DCh – 1FCh) . . . . .	768
15.12	NT Port Virtual Interface Device-Specific Registers (Offsets B70h – C88h) . . . . .	770
15.12.1	NT Port Virtual Interface Device-Specific Registers – Vendor-Specific Extended Capability 4 (Offsets C34h – C88h) . . . . .	771
15.13	NT Port Virtual Interface NT Bridging-Specific Registers (Offsets C8Ch – DFCh) . . . . .	780
15.13.1	NT Bridging-Specific Registers – NT Port ID (Offsets C8Ch – C94h) . . . . .	781
15.13.2	NT Port Virtual Interface NT Bridging-Specific Registers – Requester ID Translation Lookup Table Entry (Addresses D94h – DD0h) . . . . .	782
15.13.3	NT Bridging-Specific Registers – Lookup Table Enable Link (Offsets DD4h – DDCh) . . . . .	785
15.14	NT Port Virtual Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh) . . . . .	786
15.15	NT Port Virtual Interface Device-Specific Registers – Link Error (Offsets FE0h – FFCh) . . . . .	795

## **Chapter 16 NT Port Link Interface Registers – NT Mode . . . . .797**

16.1	Introduction . . . . .	797
16.2	NT Port Link Interface Type 0 Register Map . . . . .	798
16.3	Register Access . . . . .	800
16.3.1	PCI Express Base r3.0 Configuration Mechanism . . . . .	800
16.3.1.1	PCI r3.0-Compatible Configuration Mechanism . . . . .	800
16.3.1.2	PCI Express Enhanced Configuration Access Mechanism . . . . .	801
16.3.2	Device-Specific Memory-Mapped Configuration Mechanism . . . . .	802
16.3.3	Device-Specific Cursor Mechanism . . . . .	804
16.4	Register Descriptions . . . . .	805
16.5	NT Port Link Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch) . . . . .	806
16.6	NT Port Link Interface PCI Power Management Capability Registers (Offsets 40h – 44h) . . . . .	816
16.7	NT Port Link Interface MSI Capability Registers (Offsets 48h – 64h) . . . . .	819
16.8	NT Port Link Interface PCI Express Capability Registers (Offsets 68h – A0h) . . . . .	823
16.9	NT Port Link Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh) . . . . .	837
16.10	NT Port Link Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh) . . . . .	843
16.11	NT Port Link Interface Device-Specific Registers (Offsets 1C0h – A74h) . . . . .	846
16.11.1	NT Port Link Interface Device-Specific Registers – Captured Bus Number and Device Number (Offsets 1DCh – 1FCh) . . . . .	847
16.11.2	NT Port Link Interface Device-Specific Registers – Error Checking and Debug (Offsets 700h – 75Ch) . . . . .	849
16.12	NT Port Link Interface Device-Specific Registers (Offsets B70h – C88h) . . . . .	850
16.12.1	NT Port Link Interface Device-Specific Registers – Vendor-Specific Extended Capability 4 (Offsets C34h – C88h) . . . . .	851

**NDA CONFIDENTIAL -- Inventec | King Michael**

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

16.13	NT Bridging-Specific Registers (Offsets C8Ch – EFCh) . . . . .	853
16.13.1	NT Bridging-Specific Registers – NT Port ID and Requester ID Read Back (Offsets C8Ch – C94h) . . . . .	854
16.13.2	NT Bridging-Specific Registers – Requester ID Translation Lookup Table Entry (Addresses DB4h – DF0h) . . . . .	855
16.13.3	NT Bridging-Specific Registers – Lookup Table Enable Link (Offsets DF4h – DFCh) . . . . .	857
16.14	NT Port Link Interface Device-Specific Registers (Offsets F00h – FB0h) . . . . .	858
16.14.1	NT Port Link Interface Device-Specific Registers – Virtual Switch (Offsets F00h – F20h), Virtual Switch Mode . . . . .	858
16.14.2	NT Port Link Interface Device-Specific Registers – Error Checking and Debug (Offsets F70h – FB0h) . . . . .	859
16.15	NT Port Link Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh) . . . . .	861
<b>Chapter 17</b>	<b>Test and Debug . . . . .</b>	<b>863</b>
17.1	Introduction . . . . .	863
17.2	Physical Layer Loopback Operation . . . . .	864
17.2.1	Overview . . . . .	864
17.2.2	Internal Loopback Mode (LB_NES) . . . . .	866
17.2.3	Analog Loopback Master Mode . . . . .	867
17.2.4	Digital Loopback Master Mode . . . . .	869
17.2.5	Analog Loopback Slave Mode (LB_FEP) . . . . .	870
17.2.6	Digital Loopback Slave Mode (LB_PIPE) . . . . .	871
17.3	Built-In Self-Test . . . . .	872
17.4	Using the SerDes Quad x Diagnostic Data Registers . . . . .	872
17.5	PHY Testability Features . . . . .	873
17.6	JTAG Interface . . . . .	875
17.6.1	IEEE 1149.1 and IEEE 1149.6 Test Access Port . . . . .	875
17.6.2	JTAG Instructions . . . . .	876
17.6.3	JTAG Boundary Scan . . . . .	877
17.6.4	JTAG Reset Input – JTAG_TRST# . . . . .	877
17.7	Port Good Status LEDs . . . . .	878
17.8	Thermal Sensor Diode . . . . .	880
<b>Chapter 18</b>	<b>Electrical Specifications . . . . .</b>	<b>881</b>
18.1	Introduction . . . . .	881
18.2	Power-Up/Power-Down Sequence . . . . .	882
18.3	Absolute Maximum Ratings . . . . .	882
18.4	Power Characteristics . . . . .	883
18.5	Power Consumption Estimates . . . . .	883
18.6	I/O Interface Signal Groupings . . . . .	884
18.7	Transmit Characteristics . . . . .	886
18.7.1	PCI Express Transmitter AC and DC Characteristics . . . . .	886
18.7.2	Tx Preshoot and De-Emphasis . . . . .	891
18.7.3	Tx Voltage and Jitter . . . . .	893
18.8	Receive Characteristics . . . . .	895
18.8.1	Receive Equalization . . . . .	895
18.8.2	PCI Express Receiver AC and DC Characteristics . . . . .	897

**Chapter 19 Thermal and Mechanical Specifications .....903**  
    19.1 Thermal Characteristics ..... 903  
    19.2 General Package Specifications ..... 904  
    19.3 Mechanical Dimensions ..... 905

**Appendix A General Information .....907**  
    A.1 Product Ordering Information ..... 907  
    A.2 United States and International Representatives and Distributors ..... 908  
    A.3 Technical Support ..... 908

# Registers

## Transparent Port Registers

<b>PCI-Compatible Type 1 Configuration</b>	
<b>Header Registers (Offsets 00h – 3Ch)</b>	<b>274</b>
13-1. 00h PCI Configuration ID	274
13-2. 04h PCI Command/Status	275
13-3. 08h PCI Class Code and Revision ID	278
13-4. 0Ch Miscellaneous Control	278
13-5. 10h Base Address 0	279
13-6. 14h Base Address 1	280
13-7. 18h Bus Number	280
13-8. 1Ch Secondary Status, I/O Limit, and I/O Base	281
13-9. 20h Memory Base and Limit	283
13-10. 24h Prefetchable Memory Base and Limit	284
13-11. 28h Prefetchable Memory Upper Base Address	285
13-12. 2Ch Prefetchable Memory Upper Limit Address	285
13-13. 30h I/O Upper Base and Limit Address	286
13-14. 34h Capability Pointer	286
13-15. 38h Expansion ROM Base Address	286
13-16. 3Ch Bridge Control and PCI Interrupt Signal	287
<b>PCI Power Management Capability Registers</b>	
<b>(Offsets 40h – 44h)</b>	<b>290</b>
13-17. 40h PCI Power Management Capability	290
13-18. 44h PCI Power Management Status and Control	291
<b>Message Signaled Interrupt Capability Registers</b>	
<b>(Offsets 48h – 64h)</b>	<b>293</b>
13-19. 48h MSI Capability	294
13-20. 4Ch MSI Address	295
13-21. 50h MSI Upper Address	295
13-22. 54h MSI Data	295
13-23. 58h MSI Mask	296
13-24. 5Ch MSI Status	298
<b>PCI Express Capability Registers</b>	
<b>(Offsets 68h – A0h)</b>	<b>300</b>
13-25. 68h PCI Express Capability List and Capability	301
13-26. 6Ch Device Capability	302
13-27. 70h Device Status and Control	304
13-28. 74h Link Capability	306
13-29. 78h Link Status and Control	309
13-30. 7Ch Slot Capability	313
13-31. 80h Slot Status and Control	318
13-32. 8Ch Device Capability 2	328
13-33. 90h Device Status and Control 2	329
13-34. 94h Link Capability 2	330
13-35. 98h Link Status and Control 2	331
<b>Subsystem ID and Subsystem Vendor ID</b>	
<b>Capability Registers (Offsets A4h – FCh)</b>	<b>334</b>
13-36. A4h Subsystem Capability	334
13-37. A8h Subsystem ID and Subsystem Vendor ID	334

NDA CONFIDENTIAL -- Inventec | King Michael

**Device Serial Number Extended Capability Registers**

<b>(Offsets 100h – 108h)</b>	<b>335</b>
13-38. 100h Device Serial Number Extended Capability Header	335
13-39. 104h Serial Number (Lower DW)	336
13-40. 108h Serial Number (Upper DW)	336

**Secondary PCI Express Extended Capability Registers**

<b>(Offsets 10Ch – 134h)</b>	<b>337</b>
13-41. 10Ch Secondary PCI Express Extended Capability Header	337
13-42. 110h Link Control 3	338
13-43. 114h Lane Error Status	339
13-44. 118h Lane 0, 16, or 32 and 1, 17, or 33 Equalization Control	340
13-45. 11Ch Lane 2, 18, or 34 and 3, 19, or 35 Equalization Control	343
13-46. 120h Lane 4, 20, or 36 and 5, 21, or 37 Equalization Control	346
13-47. 124h Lane 6, 22, or 38 and 7, 23, or 39 Equalization Control	349
13-48. 128h Lane 8, 24, or 40 and 9, 25, or 41 Equalization Control	352
13-49. 12Ch Lane 10, 26, or 42 and 11, 27, or 43 Equalization Control	355
13-50. 130h Lane 12, 28, or 44 and 13, 29, or 45 Equalization Control	358
13-51. 134h Lane 14, 30, or 46 and 15, 31, or 47 Equalization Control	361

**Power Budget Extended Capability Registers**

<b>(Offsets 138h – 144h)</b>	<b>364</b>
13-52. 138h Power Budget Extended Capability Header	364
13-53. 13Ch Data Select	365
13-54. 140h Power Budget Data	365
13-55. 144h Power Budget Capability	366

**Virtual Channel Extended Capability Registers**

<b>(Offsets 148h – 1BCh)</b>	<b>367</b>
13-56. 148h Virtual Channel Extended Capability Header	367
13-57. 14Ch Port VC Capability 1	368
13-58. 150h Port VC Capability 2	369
13-59. 154h Port VC Status and Control	369
13-60. 158h VC0 Resource Capability	370
13-61. 15Ch VC0 Resource Control	371
13-62. 160h VC0 Resource Status	372

**WRR Port Arbitration Table Registers**

<b>(Offsets 178h – 1BCh)</b>	<b>373</b>
13-63. 178h Port Arbitration Table Phases 0 to 3	374
13-64. 17Ch Port Arbitration Table Phases 4 to 7	375
13-65. 180h Port Arbitration Table Phases 8 to 11	376
13-66. 184h Port Arbitration Table Phases 12 to 15	377
13-67. 188h Port Arbitration Table Phases 16 to 19	378
13-68. 18Ch Port Arbitration Table Phases 20 to 23	379
13-69. 190h Port Arbitration Table Phases 24 to 27	380
13-70. 194h Port Arbitration Table Phases 28 to 31	381
13-71. 198h Port Arbitration Table Phases 32 to 35	382
13-72. 19Ch Port Arbitration Table Phases 36 to 39	383
13-73. 1A0h Port Arbitration Table Phases 40 to 43	384
13-74. 1A4h Port Arbitration Table Phases 44 to 47	385
13-75. 1A8h Port Arbitration Table Phases 48 to 51	386
13-76. 1ACh Port Arbitration Table Phases 52 to 55	387
13-77. 1B0h Port Arbitration Table Phases 56 to 59	388
13-78. 1B4h Port Arbitration Table Phases 60 to 63	389

**NDA CONFIDENTIAL -- Inventec | King Michael**

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved



<b>Device-Specific Registers</b> <b>(Offsets 1C0h – A74h)</b> .....	<b>390</b>
<b>Device-Specific Registers – Read Pacing</b> <b>(Offsets 1D0h – 1D8h)</b> .....	<b>392</b>
13-79. 1D0h Read Pacing Control .....	393
13-80. 1D4h Read Pacing Threshold 1 .....	395
13-81. 1D8h Read Pacing Threshold 2 .....	395
<b>Device-Specific Registers – Captured Bus and Device</b> <b>Numbers (Offsets 1DCh – 1E4h)</b> .....	<b>396</b>
13-82. 1DCh Captured Bus and Device Numbers .....	396
<b>Device-Specific Registers – Read Pacing</b> <b>(Offset 1E8h)</b> .....	<b>397</b>
13-83. 1E8h Read Pacing Watermark .....	397
<b>Device-Specific Registers – Physical Layer</b> <b>(Offsets 200h – 25Ch)</b> .....	<b>398</b>
13-84. 200h Physical Layer Receiver Detect Status and Electrical Idle for Compliance Mask .....	399
13-85. 204h Electrical Idle Detect/Receiver Detect Mask .....	401
13-86. 208h Port Control .....	403
13-87. 20Ch Physical Layer User Test Pattern, Bytes 0 through 3 .....	405
13-88. 210h Physical Layer User Test Pattern, Bytes 4 through 7 .....	405
13-89. 214h Physical Layer User Test Pattern, Bytes 8 through 11 .....	406
13-90. 218h Physical Layer User Test Pattern, Bytes 12 through 15 .....	406
13-91. 21Ch Physical Layer Command and Status .....	407
13-92. 220h Physical Layer Function Control .....	409
13-93. 224h Physical Layer Test 0 .....	412
13-94. 228h Physical Layer Test 1 .....	413
13-95. 22Ch Physical Layer Safety Bits .....	415
13-96. 230h Physical Layer Port Command .....	416
13-97. 234h SKIP Ordered-Set Interval .....	420
13-98. 238h SerDes Quad 0 Diagnostic Data .....	421
13-99. 23Ch SerDes Quad 1 Diagnostic Data .....	422
13-100. 240h SerDes Quad 2 Diagnostic Data .....	423
13-101. 244h SerDes Quad 3 Diagnostic Data .....	424
13-102. 248h Target Link Width .....	425
13-103. 254h Physical Layer Additional Status .....	428
13-104. 258h Physical Layer Additional Control .....	430
13-105. 25Ch Physical Layer Error Injection Control .....	431
<b>Device-Specific Registers – Serial EEPROM</b> <b>(Offsets 260h – 270h)</b> .....	<b>433</b>
13-106. 260h Serial EEPROM Status and Control .....	433
13-107. 264h Serial EEPROM Buffer .....	436
13-108. 268h Serial EEPROM Clock Frequency .....	437
13-109. 26Ch Serial EEPROM 3 <sup>rd</sup> Address Byte .....	438
13-110. 270h Serial EEPROM CRC Value .....	438
<b>Device-Specific Registers – I<sup>2</sup>C and SMBus Slave Interfaces</b> <b>(Offsets 290h – 2FCh)</b> .....	<b>439</b>
13-111. 294h I <sup>2</sup> C Configuration .....	439
13-112. 2C8h SMBus Configuration .....	440

**Device-Specific Registers – Port Configuration**

<b>and Lane Status (Offsets 300h – 350h)</b>	<b>442</b>
13-113. 300h Port Configuration	443
13-114. 304h x1 Port Configuration	445
13-115. 308h x2 Port Configuration	446
13-116. 314h Clock Enable	447
13-117. 330h Station 0/1 Lane Status	449
13-118. 334h Station 2 Lane Status	451

**Device-Specific Registers – Port Configuration**

<b>(Offsets 354h – 46Ch)</b>	<b>452</b>
13-119. 354h Management Port Control	454
13-120. 358h Virtual Switch Enable	456
13-121. 360h VS0 Upstream	457
13-122. 364h VS1 Upstream	459
13-123. 368h VS2 Upstream	459
13-124. 36Ch VS3 Upstream	460
13-125. 370h VS4 Upstream	461
13-126. 374h VS5 Upstream	461
13-127. 380h VS0 Port Vector	462
13-128. 384h VS1 Port Vector	464
13-129. 388h VS2 Port Vector	465
13-130. 38Ch VS3 Port Vector	466
13-131. 390h VS4 Port Vector	467
13-132. 394h VS5 Port Vector	468
13-133. 3A0h Port Reset	469
13-134. 3A4h Parallel Hot Plug Control	470
13-135. 3A8h VSx_PERST# and NT_PERST# Status	471
13-136. 3ACh Configuration Release	473
13-137. 46Ch Strap Configuration	474

**Device-Specific Registers – General-Purpose Input/Output**

<b>(Offsets 600h – 68Ch)</b>	<b>477</b>
13-138. 600h GPIO 0_9 Direction Control	478
13-139. 604h GPIO 10_11 Direction Control	483
13-140. 614h GPIO 0_11 Input De-Bounce	485
13-141. 61Ch GPIO 0_11 Input Data	488
13-142. 624h GPIO 0_11 Output Data	490
13-143. 62Ch GPIO 0_11 Interrupt Polarity	492
13-144. 634h GPIO 0_11 Interrupt Status	494
13-145. 63Ch GPIO 0_11 Interrupt Mask	496
13-146. 64Ch Virtual Switch GPIO Update	499
13-147. 650h VS0 GPIO_PG 0_11 Assignment	499
13-148. 654h VS1 GPIO_PG 0_11 Assignment	501
13-149. 658h VS2 GPIO_PG 0_11 Assignment	503
13-150. 65Ch VS3 GPIO_PG 0_11 Assignment	505
13-151. 660h VS4 GPIO_PG 0_11 Assignment	507
13-152. 664h VS5 GPIO_PG 0_11 Assignment	509

**Device-Specific Registers – PORT\_GOOD Selector**

<b>(Offsets 6A0h – 6A8h)</b>	<b>511</b>
13-153. 6A0h PORT_GOOD Selector 0	512
13-154. 6A4h PORT_GOOD Selector 1	513
13-155. 6A8h PORT_GOOD Selector 2	514

**NDA CONFIDENTIAL -- Inventec | King Michael**

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

**Device-Specific Registers – Error Checking and Debug**

<b>(Offsets 700h – 75Ch)</b>	<b>515</b>
13-156. 700h Device-Specific Error Status 1	516
13-157. 704h Device-Specific Error Mask 1	519
13-158. 708h Device-Specific Error Status 2	522
13-159. 70Ch Device-Specific Error Mask 2	525
13-160. 710h Device-Specific Error Status 3	528
13-161. 714h Device-Specific Error Mask 3	530
13-162. 718h Device-Specific Error Status 4	532
13-163. 71Ch Device-Specific Error Mask 4	534
13-164. 720h ECC Error Check Disable	536
13-165. 724h Gen 3 Framing Error Status	540
13-166. 728h Gen 3 Framing Error Mask	541
13-167. 72Ch Gen 3 Framing Error Disable	542

**Device-Specific Registers – Control**

<b>(Offsets 760h – 774h)</b>	<b>543</b>
13-168. 760h Station-Based Control	544
13-169. 764h Ingress Chip Control	547

**Device-Specific Registers – Soft Error**

<b>(Offsets 778h – 8FCh)</b>	<b>549</b>
13-170. 778h Ingress PLL RAM ECC 1-Bit Counter	550
13-171. 77Ch Accumulator Debug NAK	550
13-172. 780h Ingress Exerciser Completion Status	551
13-173. 784h Ingress Exerciser Completion Data	551
13-174. 800h Egress Station 0 Payload RAM Soft Error Counters	552
13-175. 804h Egress Station 1 Payload RAM Soft Error Counters	552
13-176. 808h Egress Station 2 Payload RAM Soft Error Counters	552
13-177. 810h Egress Header RAM Soft Error Counters 1	553
13-178. 814h Egress Header RAM Soft Error Counters 2	553
13-179. 818h Egress Header RAM Soft Error Counters 3	554
13-180. 81Ch Egress Header RAM Soft Error Counters 4	554
13-181. 820h Egress PLL RAM Soft Error Counters 1	555
13-182. 824h Egress PLL RAM Soft Error Counters 2	555
13-183. 828h Egress Miscellaneous RAM Soft Error Counters	556
13-184. 82Ch Soft Error Injection	557

**Device-Specific Registers – Virtual Switch**

<b>(Offsets 900h – 9E8h), Virtual Switch Mode</b>	<b>560</b>
13-185. 900h Switch Link Up	561
13-186. 904h Switch Link Down	563
13-187. 908h Switch Link Event Mask	565
13-188. 90Ch Switch Link Status	568
13-189. 910h VS Upstream to Management Upstream Doorbell Request	570
13-190. 914h VS Upstream to Management Upstream Doorbell Mask	570
13-191. 918h VS Upstream to Management Upstream Scratchpad 1	571
13-192. 91Ch VS Upstream to Management Upstream Scratchpad 2	571
13-193. 920h VS Upstream to Management Upstream Scratchpad 3	571
13-194. 924h VS Upstream to Management Upstream Scratchpad 4	571
13-195. 928h Management Upstream to VS Upstream Doorbell Request	572
13-196. 92Ch Management Upstream to VS Upstream Doorbell Mask	572
13-197. 930h Management Upstream to VS Upstream Scratchpad 1	573
13-198. 934h Management Upstream to VS Upstream Scratchpad 2	573
13-199. 938h Management Upstream to VS Upstream Scratchpad 3	573
13-200. 93Ch Management Upstream to VS Upstream Scratchpad 4	573

**NDA CONFIDENTIAL -- Inventec | King Michael**

**Device-Specific Registers – Ingress Credit Handler**

<b>(Offsets 9ECh – A2Ch)</b>	<b>574</b>
13-201. 9F0h INCH Station Pool Values	575
13-202. 9F8h INCH Reserve Pool	575
13-203. 9FCh INCH Port Pool	576
13-204. A00h INCH Threshold VC0 Posted	577
13-205. A04h INCH Threshold VC0 Non-Posted	579
13-206. A08h INCH Threshold VC0 Completion	581

**Device-Specific Registers – Virtual Switch Debug**

<b>and GPIO Status and Control (Offsets A30h – A74h)</b>	<b>583</b>
13-207. A30h Virtual Switch Debug	584
13-208. A34h Virtual Switch GPIO_PG 0_9 Direction Control	588
13-209. A38h Virtual Switch GPIO_PG 10_11 Direction Control	592
13-210. A3Ch Virtual Switch GPIO_PG 0_11 Availability	593
13-211. A40h Virtual Switch GPIO_PG 0_11 Input De-Bounce	594
13-212. A44h Virtual Switch GPIO_PG 0_11 Input Data	597
13-213. A48h Virtual Switch GPIO_PG 0_11 Output Data	600
13-214. A4Ch Virtual Switch GPIO_PG 0_11 Interrupt Polarity	602
13-215. A50h Virtual Switch GPIO_PG 0_11 Interrupt Status	604
13-216. A54h Virtual Switch GPIO_PG 0_11 Interrupt Mask	606

**Latency Tolerance Reporting Extended Capability**

<b>Registers (Offsets B00h – B08h)</b>	<b>609</b>
13-217. B00h LTR Extended Capability Header	609
13-218. B04h Max Snoop/No-Snoop Latency	610
13-219. B08h LTR Message Received Status	611

**Device-Specific Registers**

<b>(Offsets B70h – DFCh)</b>	<b>612</b>
------------------------------	------------

**Device-Specific Registers – Vendor-Specific Extended**

<b>Capability 2 (Offsets B70h – B7Ch)</b>	<b>613</b>
13-220. B70h Vendor-Specific Extended Capability 2	613
13-221. B74h Vendor-Specific Header 2	613
13-222. B78h PLX Hardwired Configuration ID	614
13-223. B7Ch PLX Hardwired Revision ID	614

**Device-Specific Registers – Physical Layer**

<b>(Offsets B80h – C88h)</b>	<b>615</b>
13-224. B80h SerDes Control	616
13-225. B84h Synchronous Advertised N_FTS	617
13-226. B8Ch Asynchronous Advertised N_FTS	617
13-227. B94h PIPE Transmit Parameter Control	618
13-228. B98h 2.5 GT/s -3.5 dB Transmit Parameters	619
13-229. B9Ch 5.0 GT/s -6 dB Transmit Parameters	620
13-230. BA0h 5.0 GT/s -3.5 dB Transmit Parameters	621
13-231. BA4h 8.0 GT/s 0 dB Transmit Parameters	622
13-232. BB8h DC Balance Control	623
13-233. BBCh Trained Coefficient Status	623
13-234. BC0h Port Accumulation Control	624
13-235. BC4h Recovery Diagnostic	625
13-236. BC8h User Lane Equalization Control	625
13-237. BD4h Gen 3 Coefficient Values	626
13-238. BD8h Equalization Control	627
13-239. BDCh Signal Detect Level	629

**NDA CONFIDENTIAL -- Inventec | King Michael**

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

13-240. BE0h Gen 3 LFSR Seed .....	630
13-241. BECh TLP Round Trip Timer Control .....	631
13-242. BF0h Port Receiver Error Counter .....	632
13-243. BFCh AHB Access Control .....	633
<b>Device-Specific Registers – Requester ID Read Back</b>	
<b>(Offsets C8Ch – C90h) .....</b>	<b>635</b>
13-244. C90h Requester ID Read Back .....	635
<b>Multicast Extended Capability Registers</b>	
<b>(Offsets E00h – E2Ch) .....</b>	<b>636</b>
13-245. E00h Multicast Extended Capability Header .....	637
13-246. E04h Multicast Extended Capability and Control .....	637
13-247. E08h Multicast BAR0 .....	638
13-248. E0Ch Multicast BAR1 .....	638
13-249. E10h Multicast Receive 0 .....	639
13-250. E14h Multicast Receive 1 .....	639
13-251. E18h Multicast Block All 0 .....	640
13-252. E1Ch Multicast Block All 1 .....	640
13-253. E20h Multicast Block Untranslated 0 .....	641
13-254. E24h Multicast Block Untranslated 1 .....	641
13-255. E28h Multicast Overlay BAR0 .....	642
13-256. E2Ch Multicast Overlay BAR1 .....	642
<b>Device-Specific Registers – Virtual Switch</b>	
<b>(Offsets F00h – F20h), Virtual Switch Mode .....</b>	<b>643</b>
13-257. F00h VS Failover .....	644
13-258. F20h Port Cut-Thru Enable Status .....	645
<b>ACS Extended Capability Registers</b>	
<b>(Offsets F24h – F2Ch) .....</b>	<b>646</b>
13-259. F24h ACS Extended Capability Header .....	646
13-260. F28h ACS Control and Capability .....	647
13-261. F2Ch Egress Control Vector .....	649
<b>Device-Specific Registers</b>	
<b>(Offsets F30h – FB0h) .....</b>	<b>650</b>
<b>Device-Specific Registers – Egress Control</b>	
<b>(Offsets F30h – F44h) .....</b>	<b>651</b>
13-262. F30h Egress Control and Status .....	652
13-263. F38h Port Egress TLP Threshold .....	654
<b>Device-Specific Registers – Ingress Control and Port Enable</b>	
<b>(Offsets F48h – F6Ch) .....</b>	<b>655</b>
13-264. F48h Ingress Port-Based Control .....	655
13-265. F4Ch Port Enable Status .....	656
13-266. F50h Negotiated Link Width for Ports 0, 1, 2, 3 .....	657
13-267. F54h Negotiated Link Width for Ports 8, 9, 10, 11 .....	658
13-268. F58h Negotiated Link Width for Ports 16, 17, 18, 19 .....	659
13-269. F5Ch Minimum/Maximum Status .....	659
13-270. F60h Ingress Control .....	660
<b>Device-Specific Registers – Error Checking and Debug</b>	
<b>(Offsets F70h – FB0h) .....</b>	<b>662</b>
13-271. F70h Power Management Hot Plug User Configuration .....	663
13-272. FA8h ACK Transmission Latency Limit .....	667
13-273. FACH Bad TLP Count .....	668
13-274. FB0h Bad DLLP Count .....	668

NDA CONFIDENTIAL -- Inventec | King Michael

**Advanced Error Reporting Extended Capability**

**Registers (Offsets FB4h – FDCh) . . . . . 669**

13-275. FB4h Advanced Error Reporting Extended Capability Header . . . . . 669

13-276. FB8h Uncorrectable Error Status . . . . . 670

13-277. FBCh Uncorrectable Error Mask . . . . . 672

13-278. FC0h Uncorrectable Error Severity . . . . . 674

13-279. FC4h Correctable Error Status . . . . . 676

13-280. FC8h Correctable Error Mask . . . . . 678

13-281. FCCh Advanced Error Capabilities and Control . . . . . 679

13-282. FD0h Header Log 0 . . . . . 680

13-283. FD4h Header Log 1 . . . . . 680

13-284. FD8h Header Log 2 . . . . . 680

13-285. FDCh Header Log 3 . . . . . 680

**NT Port Virtual Interface Registers – NT Mode****NT Port Virtual Interface PCI-Compatible Type 0**

<b>Configuration Header Registers (Offsets 00h – 3Ch)</b>	<b>726</b>
15-1. 00h PCI Configuration ID	726
15-2. 04h PCI Command/Status	727
15-3. 08h PCI Class Code and Revision ID	729
15-4. 0Ch Miscellaneous Control	730
15-5. 10h Base Address 0	731
15-6. 14h Base Address 1	731
15-7. 18h Base Address 2	732
15-8. 1Ch Base Address 3	732
15-9. 20h Base Address 4	733
15-10. 24h Base Address 5	733
15-11. 2Ch Subsystem ID and Subsystem Vendor ID	734
15-12. 30h Expansion ROM Base Address	734
15-13. 34h Capability Pointer	735
15-14. 3Ch PCI Interrupt	735

**NT Port Virtual Interface PCI Power Management**

<b>Capability Registers (Offsets 40h – 44h)</b>	<b>736</b>
15-15. 40h PCI Power Management Capability	736
15-16. 44h PCI Power Management Status and Control	737

**NT Port Virtual Interface MSI Capability Registers**

<b>(Offsets 48h – 64h)</b>	<b>739</b>
15-17. 48h MSI Capability	740
15-18. 58h MSI Mask	741
15-19. 5Ch MSI Status	743

**NT Port Virtual Interface PCI Express Capability**

<b>Registers (Offsets 68h – A0h)</b>	<b>745</b>
15-20. 68h PCI Express Capability List and Capability	746
15-21. 6Ch Device Capability	747
15-22. 70h Device Status and Control	749
15-23. 74h Link Capability	751
15-24. 78h Link Status and Control	754
15-25. 8Ch Device Capability 2	755
15-26. 90h Device Status and Control 2	756
15-27. 94h Link Capability 2	756
15-28. 98h Link Status and Control 2	757

**NT Port Virtual Interface Vendor-Specific Capability 3**

<b>Registers (Offsets C8h – FCh)</b>	<b>758</b>
15-29. C8h Vendor-Specific Capability 3	758
15-30. D0h NT Port Virtual Interface BAR0/1 Setup	759
15-31. D4h NT Port Virtual Interface Memory BAR2 Setup	759
15-32. D8h NT Port Virtual Interface Memory BAR2/3 Setup	760
15-33. DCh NT Port Virtual Interface Memory BAR4 Setup	761
15-34. E0h NT Port Virtual Interface Memory BAR4/5 Setup	762
15-35. F8h Configuration Address Window	763
15-36. FCh Configuration Data Window	763

**NT Port Virtual Interface Virtual Channel Extended**

<b>Capability Registers (Offsets 148h – 1BCh)</b>	<b>764</b>
15-37. 148h Virtual Channel Extended Capability Header	765
15-38. 14Ch Port VC Capability 1	765
15-39. 158h VC0 Resource Capability	766
15-40. 160h VC0 Resource Status	766

NDA CONFIDENTIAL -- Inventec | King Michael

<b>NT Port Virtual Interface Device-Specific Registers (Offsets 1C0h – A74h)</b>	<b>767</b>
<b>NT Port Virtual Interface Device-Specific Registers – Captured Bus Number and Device Number (Offsets 1DCh – 1FCh)</b>	<b>768</b>
15-41. 1DCh NT Captured Bus Number	769
15-42. 1E0h NT Captured Device Number	769
<b>NT Port Virtual Interface Device-Specific Registers (Offsets B70h – C88h)</b>	<b>770</b>
<b>NT Port Virtual Interface Device-Specific Registers – Vendor-Specific Extended Capability 4 (Offsets C34h – C88h)</b>	<b>771</b>
15-43. C34h Vendor-Specific Extended Capability 4	772
15-44. C38h Vendor-Specific Header 4	772
15-45. C3Ch Memory BAR2 Address Translation Lower	773
15-46. C40h Memory BAR3 Address Translation Upper	773
15-47. C44h Memory BAR4 Address Translation Lower	773
15-48. C48h Memory BAR5 Address Translation Upper	773
15-49. C4Ch Virtual Interface IRQ Set	774
15-50. C50h Virtual Interface IRQ Clear	774
15-51. C54h Virtual Interface IRQ Mask Set	775
15-52. C58h Virtual Interface IRQ Mask Clear	775
15-53. C5Ch Link Interface IRQ Set	776
15-54. C60h Link Interface IRQ Clear	776
15-55. C64h Link Interface IRQ Mask Set	777
15-56. C68h Link Interface IRQ Mask Clear	777
15-57. C6Ch NT Port SCRATCH0	778
15-58. C70h NT Port SCRATCH1	778
15-59. C74h NT Port SCRATCH2	778
15-60. C78h NT Port SCRATCH3	778
15-61. C7Ch NT Port SCRATCH4	778
15-62. C80h NT Port SCRATCH5	779
15-63. C84h NT Port SCRATCH6	779
15-64. C88h NT Port SCRATCH7	779
<b>NT Port Virtual Interface NT Bridging-Specific Registers (Offsets C8Ch – DFCh)</b>	<b>780</b>
<b>NT Bridging-Specific Registers – NT Port ID (Offsets C8Ch – C94h)</b>	<b>781</b>
15-65. C8Ch NT Port ID	781
<b>NT Port Virtual Interface NT Bridging-Specific Registers – Requester ID Translation Lookup Table Entry (Addresses D94h – DD0h)</b>	<b>782</b>
15-66. D94h – DD0h NT Port Virtual Interface Requester ID Translation LUT Entry <sub>n_m</sub>	784
<b>NT Bridging-Specific Registers – Lookup Table Enable Link (Offsets DD4h – DDCh)</b>	<b>785</b>
15-67. DD4h Lookup Table Enable Virtual	785
15-68. DD8h Lookup Table No Snoop Enable Virtual	785
15-69. DDCh Lookup Table Function Enable Virtual	785



**NT Port Virtual Interface Advanced Error Reporting****Extended Capability Registers (Offsets FB4h – FDCh) ..... 786**

15-70. FB4h Advanced Error Reporting Extended Capability Header.....	786
15-71. FB8h Uncorrectable Error Status .....	787
15-72. FBCh Uncorrectable Error Mask.....	789
15-73. FC0h Uncorrectable Error Severity.....	791
15-74. FC4h Correctable Error Status .....	793
15-75. FC8h Correctable Error Mask.....	794

**NT Port Virtual Interface Device-Specific Registers –****Link Error (Offsets FE0h – FFCh) ..... 795**

15-76. FE0h Link Error Status Virtual .....	796
15-77. FE4h Link Error Mask Virtual .....	796

**NT Port Link Interface Registers – NT Mode****NT Port Link Interface PCI-Compatible Type 0**

<b>Configuration Header Registers (Offsets 00h – 3Ch)</b>	<b>806</b>
16-1. 00h PCI Configuration ID	806
16-2. 04h PCI Command/Status	807
16-3. 08h PCI Class Code and Revision ID	809
16-4. 0Ch Miscellaneous Control	810
16-5. 10h Base Address 0	811
16-6. 14h Base Address 1	811
16-7. 18h Base Address 2	812
16-8. 1Ch Base Address 3	812
16-9. 20h Base Address 4	813
16-10. 24h Base Address 5	813
16-11. 2Ch Subsystem ID and Subsystem Vendor ID	814
16-12. 30h Expansion ROM Base Address	814
16-13. 34h Capability Pointer	815
16-14. 3Ch PCI Interrupt	815

**NT Port Link Interface PCI Power Management**

<b>Capability Registers (Offsets 40h – 44h)</b>	<b>816</b>
16-15. 40h PCI Power Management Capability	816
16-16. 44h PCI Power Management Status and Control	817

**NT Port Link Interface MSI Capability Registers**

<b>(Offsets 48h – 64h)</b>	<b>819</b>
16-17. 48h MSI Capability	820
16-18. 58h MSI Mask	821
16-19. 5Ch MSI Status	822

**NT Port Link Interface PCI Express Capability**

<b>Registers (Offsets 68h – A0h)</b>	<b>823</b>
16-20. 68h PCI Express Capability List and Capability	824
16-21. 6Ch Device Capability	825
16-22. 70h Device Status and Control	827
16-23. 74h Link Capability	829
16-24. 78h Link Status and Control	832
16-25. 8Ch Device Capability 2	834
16-26. 90h Device Status and Control 2	834
16-27. 98h Link Status and Control 2	835

**NT Port Link Interface Vendor-Specific Capability 3**

<b>Registers (Offsets C8h – FCh)</b>	<b>837</b>
16-28. C8h Vendor-Specific Capability 3	837
16-29. E4h NT Port Link Interface BAR0/1 Setup	838
16-30. E8h NT Port Link Interface Memory BAR2 Setup	838
16-31. ECh NT Port Link Interface Memory BAR2/3 Setup	839
16-32. F0h NT Port Link Interface Memory BAR4 Setup	840
16-33. F4h NT Port Link Interface Memory BAR4/5 Setup	841
16-34. F8h Configuration Address Window	842
16-35. FCh Configuration Data Window	842

**NT Port Link Interface Virtual Channel Extended**

<b>Capability Registers (Offsets 148h – 1BCh)</b>	<b>843</b>
16-36. 148h Virtual Channel Extended Capability Header	843
16-37. 14Ch Port VC Capability 1	844
16-38. 158h VC0 Resource Capability	844
16-39. 15Ch VC0 Resource Control	845
16-40. 160h VC0 Resource Status	845

**NDA CONFIDENTIAL -- Inventec | King Michael**

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

<b>NT Port Link Interface Device-Specific Registers</b> <b>(Offsets 1C0h – A74h)</b> .....	<b>846</b>
<b>NT Port Link Interface Device-Specific Registers –</b> <b>Captured Bus Number and Device Number (Offsets 1DCh – 1FCh)</b> .....	<b>847</b>
16-41. 1DCh NT Captured Bus Number .....	848
16-42. 1E0h NT Captured Device Number .....	848
<b>NT Port Link Interface Device-Specific Registers –</b> <b>Error Checking and Debug (Offsets 700h – 75Ch)</b> .....	<b>849</b>
<b>NT Port Link Interface Device-Specific Registers</b> <b>(Offsets B70h – C88h)</b> .....	<b>850</b>
<b>NT Port Link Interface Device-Specific Registers –</b> <b>Vendor-Specific Extended Capability 4 (Offsets C34h – C88h)</b> .....	<b>851</b>
16-43. C3Ch Memory BAR2 Address Translation Lower .....	852
16-44. C40h Memory BAR3 Address Translation Upper .....	852
16-45. C44h Memory BAR4 Address Translation Lower .....	852
16-46. C48h Memory BAR5 Address Translation Upper .....	852
<b>NT Bridging-Specific Registers</b> <b>(Offsets C8Ch – EFCh)</b> .....	<b>853</b>
<b>NT Bridging-Specific Registers – NT Port ID and Requester ID</b> <b>Read Back (Offsets C8Ch – C94h)</b> .....	<b>854</b>
16-47. C8Ch NT Port ID .....	854
16-48. C90h Requester ID Read Back .....	854
<b>NT Bridging-Specific Registers – Requester ID Translation</b> <b>Lookup Table Entry (Addresses DB4h – DF0h)</b> .....	<b>855</b>
16-49. DB4h – DF0h NT Port Link Interface Requester ID Translation LUT Entry <sub>n,m</sub> .....	856
<b>NT Bridging-Specific Registers – Lookup Table Enable Link</b> <b>(Offsets DF4h – DFCh)</b> .....	<b>857</b>
16-50. DF4h Lookup Table Enable Link .....	857
16-51. DF8h Lookup Table No Snoop Enable Link .....	857
16-52. DFCh Lookup Table Function Enable Link .....	857
<b>NT Port Link Interface Device-Specific Registers</b> <b>(Offsets F00h – FB0h)</b> .....	<b>858</b>
<b>NT Port Link Interface Device-Specific Registers –</b> <b>Virtual Switch (Offsets F00h – F20h), Virtual Switch Mode</b> .....	<b>858</b>
<b>NT Port Link Interface Device-Specific Registers –</b> <b>Error Checking and Debug (Offsets F70h – FB0h)</b> .....	<b>859</b>
16-53. F70h Power Management Hot Plug User Configuration .....	860
<b>NT Port Link Interface Advanced Error Reporting</b> <b>Extended Capability Registers (Offsets FB4h – FDCh)</b> .....	<b>861</b>
16-54. FB4h Advanced Error Reporting Extended Capability Header .....	861

THIS PAGE INTENTIONALLY LEFT BLANK.

## 1.1 Overview

This data book describes PLX Technology's ExpressLane™ PEX 8748, a fully non-blocking, low-latency, low-cost, and low-power 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch. Conforming to the *PCI Express Base r3.0*, the PEX 8748 enables users to add high-bandwidth input/output (I/O) to various products, including servers, storage systems, and communications platforms. The PEX 8748's flexible hardware configuration and software programmability allows the switch to be tailored for a wide variety of application requirements.

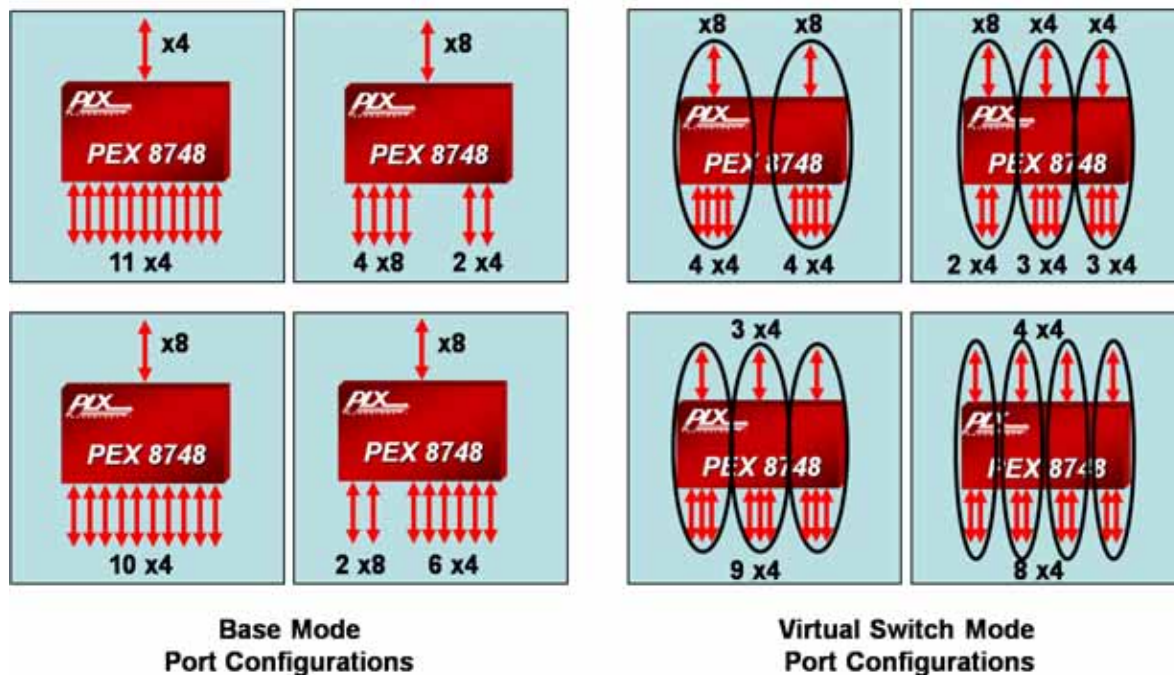
The PEX 8748 is well-suited for fan-in/out applications, as well as for applications requiring peer-to-peer communication. The PEX 8748 supports two functional modes – *Base* and *Virtual Switch*:

- **Base mode** – PEX 8748 acts as a standard PCI Express switch, supporting one Host hierarchy
- **Virtual Switch mode** – PEX 8748 supports up to six Hosts, creating up to six virtual switches within the PEX 8748, each with its own virtual hierarchy

The PEX 8748 supports multiple Port configuration options, to provide flexible solutions for optimal product design. [Figure 1-1](#) illustrates several of the possible PEX 8748 Port configurations, using differing Link widths, of x4, x8, or x16. When used with endpoints that require smaller Link widths, the PEX 8748 auto-negotiates down to support x1 and x2 Link width as well.

In Virtual Switch mode, the second Upstream Port, with no Downstream Ports assigned to it, can be used for failover. If the primary Host were to fail, the secondary Upstream Port would take over the Downstream Port assigned to the failing Upstream Port. The secondary Upstream Port need not initially have any Downstream Ports of its own.

**Figure 1-1. Common Port Configurations**



## 1.2 Features

The PEX 8748 supports the following features:

- 12-Port PCI Express switch
  - 48 Lanes with integrated on-chip SerDes
  - Low-power SerDes (under 90 mW per Lane)
  - Relaxed Ordering
  - Port configuration
    - 12 independent Ports
    - Choice of Link width (quantity of Lanes) per unique Link/Port – x4, x8, or x16, depending upon Port configuration; x1 and x2 Link widths are also supported
    - Configurable with serial EEPROM, I<sup>2</sup>C, SMBus, and/or Host software
    - Designate any Port as the *Upstream Port* (Port 0 is recommended in Base mode)
- Multi-Root support
  - Up to six Upstream Ports supported
  - 1+1 Host Failover (one active and one backup)
  - N+1 Host Failover (N active and one backup)
  - Supported in both Transparent and Non-Transparent modes
- High Performance
  - 768 GT/s aggregate bandwidth (8.0 GT/s/Lane x 48 SerDes x 2 (full duplex))
  - Integrated 8.0 GT/s SerDes speed negotiation, for each Port
  - Full line rate on all Ports
  - Cut-Thru packet latency of less than 126 ns between symmetric (x16 to x16) ingress and egress Ports
  - Non-blocking Crossbar Switch interface supports TLP bandwidth capacity of each Link
  - Maximum Payload Size – 2,048 bytes
- *performancePAK*<sup>™</sup>
  - Read Pacing<sup>™</sup> (intelligent bandwidth allocation)
  - Multicast
  - Dynamic Buffer Pool Architecture for faster credit updates
- *visionPAK*<sup>™</sup>
  - Performance Monitoring
    - Per-Port Payload and Header Counters
    - Per-traffic type (Write, Read, Completion) Counters
  - PCI Express Packet Generator
    - Capable of saturating a x16 Gen 3 Link
  - Error Injection and Pseudo-Random Bit Sequence (PRBS)
  - SerDes Loopback
  - SerDes Eye Capture

- Access Control Services (ACS) – Protection mechanisms for added data integrity in peer-to-peer transactions
- Alternative Routing-ID Interpretation (ARI) – Enables virtualized systems and/or highly integrated multi-function devices
- Quality of Service (QoS) support
  - All Ports support one, full-featured Virtual Channel, VCO
  - All Ports support eight Traffic Class (TC[7:0]) mapping, independently of the other Ports
  - Round-Robin (RR) and Weighted Round-Robin (WRR) Port arbitration
- Non-Transparent (NT) Bridging
  - Program any one Downstream Port as the *Upstream NT Port*
  - Enables Dual-Host, Dual-Fabric, Host-Failover applications
  - Moveable *Upstream Port*
  - Cross-link Port capability
- Reliability, Availability, Serviceability (RAS) features
  - PCI Express Standard Hot Plug Controller for three Transparent Downstream Ports, including optional usage models for Manually operated Retention Latch, by way of Manually operated Retention Latch (MRL) Sensor and Attention Button support
  - Serial Hot Plug, by way of I<sup>2</sup>C, for Hot Plug capability on up to 11 Transparent Downstream Ports
  - End-to-end Cyclic Redundancy Check (ECRC) and Poison bit support
  - Data path protection
  - Memory (RAM) error correction
  - Electromechanical Interlock supported with Power Enable output
  - Baseline and Advanced Error Reporting capability
  - Port (Link) Status bits available
  - Per-Port error diagnostics
  - Joint Test Action Group (JTAG) AC/DC boundary scan
- INTA# ([PEX\\_INTA#](#) (Base mode) or [VSx\\_PEX\\_INTA#](#) (Virtual Switch mode)) and FATAL ERROR ([FATAL\\_ERR#](#) (Base mode) or [VSx\\_FATAL\\_ERR#](#) (Virtual Switch mode)) (Conventional PCI SERR# equivalent) ball support
- 12 General-Purpose Input/Output (GPIO) balls
- Other PCI Express Capabilities
  - Lane reversal
  - Polarity reversal
  - Conventional PCI-compatible Link Power Management states – L0, L0s, L1, L2/L3 Ready, and L3 (with Vaux *not supported*)
  - Conventional PCI-compatible Device Power Management states – D0 and D3hot
  - Active State Power Management (ASPM)
  - Dynamic speed (2.5, 5.0, or 8.0 GT/s (Gen 1, Gen 2, and Gen 3, respectively)) negotiation, for each Port
  - Dynamic Link width negotiation
- Out-of-Band Initialization options
  - Serial EEPROM
  - I<sup>2</sup>C and SMBus (7-bit Slave address with 100 Kbps)

- Testability – AC/DC JTAG support
- 27 x 27 mm<sup>2</sup>, 676-ball Flip-Chip Ball Grid Array with Heat Spreader (HFC-BGA) package
- Typical power – 7.34W at 8.0 GT/s, with all Ports enabled
- Microsoft® Windows® Logo (WHQL)-compliant
- Compliant to the following specifications:
  - *PCI Local Bus Specification, Revision 3.0 (PCI r3.0)*
  - *PCI Bus Power Management Interface Specification, Revision 1.2 (PCI Power Mgmt. r1.2)*
  - *PCI Code and ID Assignment Specification, Revision 1.2*
  - *PCI to PCI Bridge Architecture Specification, Revision 1.2 (PCI-to-PCI Bridge r1.2)*
  - *PCI Express Base Specification, Revision 1.1 (PCI Express Base r1.1)*
  - *PCI Express Base Specification, Revision 2.0 (PCI Express Base r2.0)*
  - *PCI Express Base Specification, Revision 2.0 Errata*
  - *PCI Express Base Specification, Revision 2.1 (PCI Express Base r2.1)*
  - *PCI Express Base Specification, Revision 3.0 (PCI Express Base r3.0)*
  - *PCI Express Card Electromechanical Specification, Revision 2.0 (PCI ExpressCard CEM r2.0)*
  - *PCI Express Mini Card Electromechanical Specification, Revision 1.1 (PCI ExpressCard Mini CEM r1.1)*
  - *IEEE Standard 1149.1-1990, IEEE Standard Test Access Port and Boundary-Scan Architecture (IEEE Standard 1149.1-1990)*
  - *IEEE Standard 1149.1a-1993, IEEE Standard Test Access Port and Boundary-Scan Architecture*
  - *IEEE Standard 1149.1-1994, Specifications for Vendor-Specific Extensions*
  - *IEEE Standard 1149.6-2003, IEEE Standard Test Access Port and Boundary-Scan Architecture Extensions (IEEE Standard 1149.6-2003)*
  - *The I<sup>2</sup>C-Bus Specification, Version 2.1 (I<sup>2</sup>C Bus v2.1)*
  - *PHY Interface for the PCI Express Architecture, Version 2.00*
  - *System Management Bus Specification, Version 2.0 (SMBus v2.0)*





## Chapter 2 Features and Applications

### 2.1 Flexible and Feature-Rich 48-Lane, 12-Port Switch

#### 2.1.1 Highly Flexible Port Configurations

The PLX ExpressLane PEX 8748 PCI Express Gen 3 Multi-Root Switch offers a maximum of 12 configurable Ports (four per Station, three Stations total). To support specific bandwidth needs, Link widths can be individually configured as any power-of-two, from x1 to x16, through auto-negotiation, hardware strapping, an optional serial EEPROM, and/or the I<sup>2</sup>C Slave interface. Flexible buffer allocation, along with the PEX 8748's flexible packet flow control, maximizes throughput for applications where more traffic flows in the Downstream, rather than Upstream, direction.

The PEX 8748 architecture allows for the combining of multiple Lanes to flexible Port Link widths, as needed. *For example*, x16 Lanes can form a single x16 Port, two x8 Ports, or four x4 Ports.

Figure 1-1, “Common Port Configurations,” illustrates several of the possible PEX 8748 Port configurations in Conventional PCI mode (Base mode). The PEX 8748 can also be configured in Virtual Switch mode, where users can choose up to six Ports as Host/Upstream Ports, and assign a specific quantity of Downstream Ports to each Host. In this mode, a virtual switch is created for each Host Port and its associated Downstream Ports within the virtual switch. The traffic between a virtual switch's Ports is completely isolated from the traffic in other virtual switches. In addition to Base mode configurations, Figure 1-1 also illustrates common Port configurations in Virtual Switch mode, where each ellipse represents a virtual switch within the PEX 8748.

In Base mode, any one Port can be designated as, or dynamically changed to be, the Upstream Port (Port 0 is recommended). In Virtual Switch mode, any Port within the same virtual hierarchy can be designated as the Upstream Port for that particular hierarchy.

## 2.1.2 Non-Blocking Crossbar Switch Architecture

The Non-Blocking Crossbar Switch architecture is an on-chip interconnect switching fabric, which is built upon the existing PLX Switch Fabric Architecture technology. In addition to addressing simultaneous multiple flows, the Crossbar Switch architecture incorporates functions required to support an efficient PCI Express switch fabric, including:

- Deadlock avoidance
- Priority preemption
- PCI Express Ordering rules
- Packet fair queuing
- Oldest first scheduling

The Crossbar Switch interconnect physical topology is that of a packet-based Crossbar Switch fabric (internal fabric), designed to simultaneously connect multiple on-chip Stations. The Crossbar Switch protocol is sufficiently flexible and robust to support a variety of embedded system requirements. The protocol is specifically designed to ease chip integration, by strongly enforcing Station boundaries and standardizing communication between Stations. The Crossbar Switch architecture basic features include:

- Multiple concurrent Data transfers
- Global ordering within the PEX 8748
- Three types of transactions – Posted, Non-Posted, and Completion (P, NP, and Cpl, respectively) – meet PCI and PCI Express Ordering and Deadlock Avoidance rules
- Optional weighting of source Ports, to support Source Port arbitration

## 2.1.3 Multi-Host Architecture – NT and Virtual Switch Modes

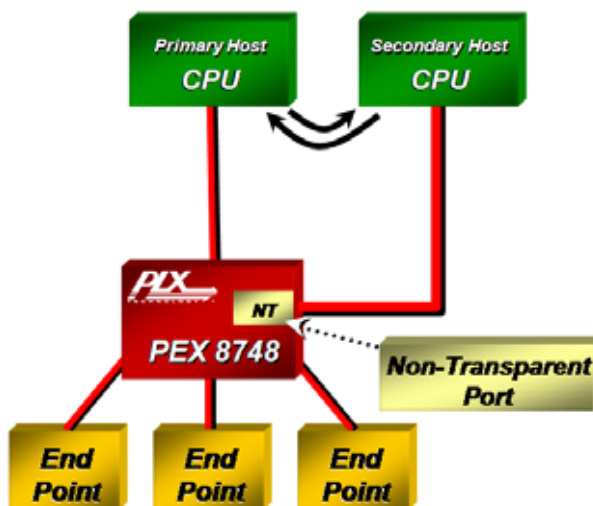
The PEX 8748 allows users to configure the switch as a single hierarchy of PCI-to-PCI bridges (Base mode), –or– as multiple hierarchies of PCI-to-PCI bridges, independent of one another (Virtual Switch mode). In Virtual Switch mode, the PEX 8748 can support up to six Hosts (one Host per virtual switch). This powerful architectural enhancement enables users to build PCI Express-based systems that support high-availability, failover, redundant, and clustered systems.

### 2.1.3.1 Dual-Host and Failover Support – NT Mode

The PEX 8748 supports a single Non-Transparent (NT) Port (Figure 2-1), which enables the implementation of dual-Host systems for redundancy and Host failover capability. The NT Port allows systems to isolate Host memory domains, by presenting the processor subsystem as an endpoint, rather than as another memory system:

- **Base Address** registers (BARs) are used to translate addresses
- **Doorbell** registers are used to signal interrupts between the address domains
- **Scratchpad** registers are accessible from both address domains, to allow inter-processor communication

Figure 2-1. NT Port – NT Mode



### 2.1.3.2 Virtual Switch Mode (Multi-Host) and Failover Support

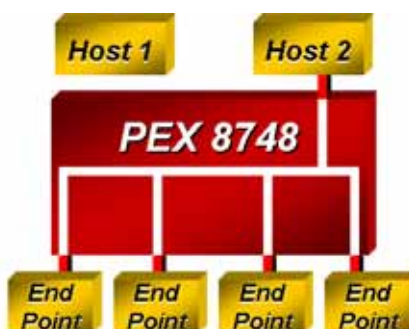
In Virtual Switch mode, the PEX 8748 can be configured with up to six Upstream Host Ports, each with its own dedicated Downstream Ports. The PEX 8748 can be configured for 1+1 or  $N+1$  Host redundancy. The PEX 8748 allows the Hosts to communicate their status to one another, using special **Doorbell** registers.

In Failover mode, if a Host fails, the Host designated for failover disables the Upstream Port attached to the failing Host, then programs the Downstream Ports of that Host to its own domain. [Figure 2-2a](#) illustrates a two-Host system in Virtual Switch mode, with two virtual switches within the PEX 8748. [Figure 2-2b](#) illustrates Host 1 as being disabled after failing, and Host 2 having taken over all of Host 1's endpoints.

**Figure 2-2. Virtual Switch Mode (Multi-Host) and Failover Support**



**Figure 2-2a. Multi-Host**



**Figure 2-2b. Multi-Host Failover**

## 2.1.4 Low Packet Latency and High Performance

The PEX 8748 architecture supports packet **Cut-Thru with a latency of less than 126 ns** between symmetric (**x16 to x16**) ingress and egress Ports. This, combined with large Packet memory, flexible common buffer/FC credit pool, and Non-Blocking Internal Switch architecture, provides full line rate on all Ports for performance-hungry applications, *such as* servers and switch fabrics. The low latency enables applications to achieve high throughput and performance. In addition to low latency, the PEX 8748 supports a Packet Payload size of up to 2,048 bytes, enabling users to achieve even higher throughput.

### 2.1.4.1 Data Payloads

The Data Payloads are variable length with a maximum of 2,048 bytes, as defined by the *Maximum Payload Size* field (available sizes are 128, 256, 512, 1,024, and 2,048, depending upon the quantity of enabled Ports, per Station). Read Requests **do not** include a Data Payload.

**Note:** Refer to the *Device Control* register *Maximum Payload Size* field (offset 70h[7:5]) for Maximum Payload Size Port limitations.

### 2.1.4.2 Cut-Thru Mode

Cut-Thru mode can reduce latency, especially for longer packets, because the entire packet does not need to be stored before being forwarded. Instead, after the Header is decoded, the packet can be immediately forwarded. The PEX 8748 is designed to cut through TLPs, to and from every Port. By default, all Ports are enabled for Cut-Thru.

Cut-Thru mode is supported for the PEX 8748's NT Port Link Interface, if the PEX 8748 is configured for NT mode.

**Caution:** *One of the drawbacks to using Cut-Thru mode is that the TLP is not known to be good until the last byte. If the TLP proves to be bad, the Cut-Thru packet must be discarded. If the TLP has already been forwarded to another device, that TLP will be framed with an EDB (End Data Bad), as opposed to the standard END.*

## 2.1.5 Virtual Channel and Traffic Classes

The PEX 8748 supports one Virtual Channel (VC0) and eight Traffic Classes (TC[7:0]). VC0 and TC0 are required by the *PCI Express Base r3.0*, and configured at device start-up.

## 2.1.6 Data Integrity

To enable designs that require **guaranteed error-free packets**, the PEX 8748 provides **End-to-end Cyclic Redundancy Check (ECRC)** protection and **Poison** bit support, as well as **Error-Correcting Code (ECC)** protection on the internal data paths and memory (RAM). ECC maintains packet integrity through the PEX 8748, by providing automatic correction of any 1-bit errors. These features are optional in the *PCI Express Base r3.0*; however, PLX provides them across its entire ExpressLane PCI Express Gen 2 and Gen 3 switch product lines.

## 2.1.7 Configuration Flexibility

The PEX 8748 provides several ways to configure its operations. *For example*, the PEX 8748 can be configured through Strapping balls, Host software, an optional serial EEPROM, and/or the I<sup>2</sup>C Slave interface. Additionally, the I<sup>2</sup>C Slave interface allows for easy debug during the Development phase, performance monitoring during the Operation phase, and driver or software upgrade.

## 2.1.8 Interoperability

The PEX 8748 is designed to be fully compliant with the *PCI Express Base r3.0*, and is backward-compatible to the *PCI Express Base r2.1*, *PCI Express Base r2.0*, *PCI Express Base r1.1*, and *PCI Express Base r1.0a*. Additionally, the switch supports **auto-negotiation**, **Lane reversal**, and **polarity reversal**, for maximum board design and board layout flexibility. Furthermore, the PEX 8748 is designed to be interoperable with many popular motherboards and server boards with PCI Express connections, and PCI Express endpoints (Ethernet, RAID Controllers), as well as PLX's family of PCI Express switches and bridges. All PLX ExpressLane devices undergo thorough interoperability testing at PLX's **Interoperability Lab** and compliance testing at the **PCI-SIG Compliance Workshop**, to ensure compatibility with PCI Express devices in the market.

## 2.1.9 Low Power with Granular SerDes Control

The PEX 8748 provides **low-power** capability that is fully compliant with the *PCI Express Base r3.0* and *PCI Power Mgmt. r1.2* Power Management (PM) specifications. Unused SerDes can be disabled, to further reduce power consumption.

The PEX 8748 supports **SerDes output software control**, to allow power and signal strength optimization within a system. The PLX SerDes implementation supports four power levels – *Off*, *Low*, *Typical*, and *High*. The SerDes block also supports **Loopback modes** and **Advanced Error Reporting**, which enables efficient system debug and management.

## 2.1.10 Dynamic Lane Reversal

The PEX 8748 supports dynamic Lane reversal during the Link training process. Lane reversal capability allows flexibility in determining board routing, so that PCI Express components can be connected without having to criss-cross wires. If the wiring of Lanes to a device is reversed (on both Transmitters and Receivers), only one of the two connected devices must support Lane reversal.

Either of the outside Lanes (Transmitter and Receiver pairs) of the PEX 8748 programmed Link width must be identified as being Lane 0. During Link training, both devices on the Link negotiate the Lane numbering. During the Link Training and Status State Machine (LTSSM)'s *Configuration* state, the Upstream device sends TS1 Ordered-Sets, in which each connected Lane is identified by a consecutive Lane Number, starting with Lane 0 corresponding to the physical Lane Number associated with the Port.

The Port reverses its Lane Numbers and attempts to re-train when any of the following conditions occur:

- No Receiver is detected on preferred Lane 0
- No valid Training Sets are received on preferred Lane 0 during the LTSSM's *Polling* state
- TS1 with a non-zero Lane Number Port is received on the Port's Lane 0

To confirm successful Lane Number negotiation, both devices exchange TS2 Ordered-Sets with identical Lane Numbers on each connected Lane.

## 2.1.11 Hot Plug for High Availability

Hot Plug capability allows users to replace hardware modules and perform maintenance, without having to power down the system. The PEX 8748 Hot Plug Capability and Advanced Error Reporting features make the switch suitable for High-Availability (HA) applications. The PEX 8748 supports both Parallel and Serial Hot Plug. Parallel Hot Plug is supported on three Transparent Downstream Ports, and/or Serial Hot Plug is supported on up to 11 Transparent Downstream Ports.

For further details, refer to [Chapter 10, "Hot Plug Support."](#)

## 2.1.12 Fully Compliant Power Management

The PEX 8748 supports Link (L0, L0s, L1, L2/L3 Ready, and L3) and Device (D0 and D3hot) PM states, in compliance with the *PCI Express Base r3.0* and *PCI Power Mgmt. r1.2* PM specifications.

For further details, refer to [Chapter 11, "Power Management."](#)

## 2.1.13 General-Purpose Input/Output Signals

The PEX 8748 contains 12 multiplexed General-Purpose Input/Output (GPIO) balls and associated registers, that can be programmed to function as GPIO, Link Status (PORT\_GOOD) indicators, and/or Interrupt inputs. Default functionality is GPIO input; however, serial EEPROM, I<sup>2</sup>C/SMBus, and/or software can program the **GPIO** registers to define functionality for each I/O. Default functionality can also be modified by the logical value of the STRAP\_TESTMODE[2:0] 3-state inputs, sampled at Fundamental Reset. Because typical designs implement PORT\_GOOD functionality for enabled Ports, GPIO[11:0] are renamed as [PORT\\_GOOD\[11:0\]#](#).

For details, refer to the PORT\_GOOD[11:0]# and STRAP\_TESTMODE[2:0] signal descriptions in [Section 3.4, "Signal Ball Descriptions."](#)

**Note:** The *PORT\_GOOD11#/GPIO11* signal exists only in Virtual Switch mode, when four, five, or six virtual switches are enabled.

Information related to how and when these multiplexed signals are used is provided in [Section 3.4.4.1, "STRAP\\_TESTMODE\[2:0\] 3-State Input Configurations."](#)

NDA CONFIDENTIAL -- Inventec | King Michael

## 2.1.14 ***performancePAK***

Exclusive to PLX, *performancePAK* is a suite of unique and innovative performance features that enable PLX switches to be the highest-performing switches available in the market today. The *performancePAK* features consist of Read Pacing, Multicast, and Dynamic Buffer Pool.

### 2.1.14.1 **Read Pacing**

The Read Pacing feature allows users to throttle the quantity of Read Requests being made by Downstream devices. When a Downstream device requests several long Reads back-to-back, the Root Complex services the Read Requests from this Downstream Port in a sequential order. If this Port has a narrow Link and is therefore slow in receiving these Read packets from the Root Complex, other Downstream Ports may become starved, thus negatively impacting performance. This feature enhances performance by allowing for the adequate servicing of all Downstream devices, by intelligent handling of Read Requests.

For further details, refer to [Section 8.5, “Read Pacing.”](#)

### 2.1.14.2 **Multicast**

Multicast (MC) allows programs to concurrently write the same data to a group of multiple destinations. When Posted Memory Write or Address Routed Message TLPs entering the PEX 8748 are addressed to the MC Address range (*MC BARs*), the PEX 8748 automatically generates and transmits, if enabled, a copy of the original TLP (referred to as the *MC Copy TLP*) to the destination Ports. The MC Address space is divided into *MC Groups (MCG)*, defined by using *MC Base Address* and *MC Index Position*. Each PEX 8748 Port can elect to receive an MC Copy TLP by belonging to an *MCG*, by Setting the corresponding *MC Receive* bit. An MC TLP can be blocked using the *MC Block All* bit, if required. *MC Overlay Bar* can be used to replace the original MC TLP's address to a Unicast Address space, if the endpoint does not support MC.

For further details, refer to [Section 8.6, “Multicast.”](#)

### 2.1.14.3 **Dynamic Buffer Pool**

The PEX 8748 uses a dynamic buffer pool for FC management, which uses a common pool of FC Credits that is shared among other Ports within the same Station. This shared buffer pool is user-programmable, so FC credits can be allocated among the enabled Ports, as needed. Not only does this prevent wasted buffers and inappropriate buffer assignments, any un-allocated buffers remain in the common buffer pool, which can then be used by other Ports within the same Station, for faster FC credit updates.



## 2.1.15 visionPAK

Another PLX exclusive, *visionPAK* is a debug diagnostics suite of integrated hardware and software instruments that users can use to help bring their systems to market faster. *visionPAK* features consist of Performance Monitoring, Error Injection, SerDes Loopback, SerDes Eye Capture, and more.

### 2.1.15.1 Performance Monitoring

The PEX 8748's real-time performance monitoring allows users to literally “see” ingress and egress performance on each Port as traffic passes through the switch, using PLX's Software Development Kit (SDK). The monitoring is completely passive, and therefore, has no effect on overall system performance. Internal Counters provide extensive granularity down to traffic and packet type, and even allow for the filtering of traffic (*that is*, count only Memory Writes).

### 2.1.15.2 Error Injection

Using the PEX 8748's Error Injection feature, users can inject malformed packets and/or Fatal errors into their system, then evaluate the system's ability to detect and recover from such errors.

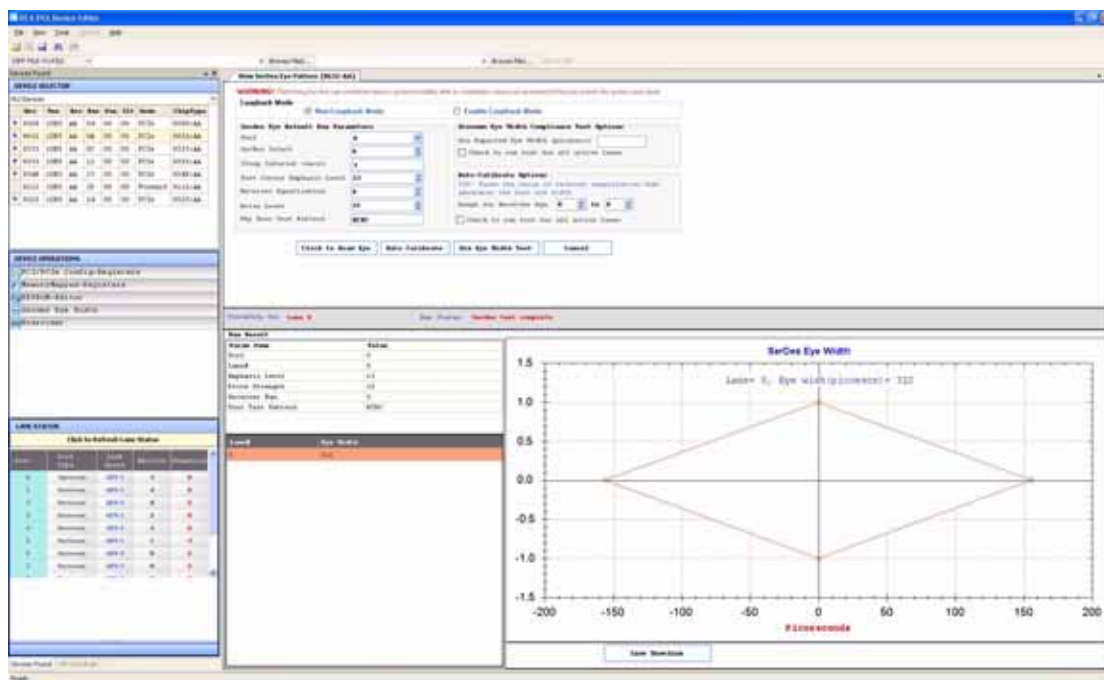
### 2.1.15.3 SerDes Loopback

The PEX 8748 supports External Tx, Recovered Clock, and Recovered Data Loopback modes.

### 2.1.15.4 SerDes Eye Capture

Users can evaluate their system's signal integrity at the Physical Layer (PHY), using the PEX 8748's SerDes Eye Capture feature. Using PLX's SDK, users can view the Receiver eye width of any Lane on the PEX 8748. Users can then modify SerDes Settings and see the impact on the Receiver eye. [Figure 2-3](#) presents a screenshot of the SDK's SerDes Eye Capture feature.

**Figure 2-3. PLX SDK SerDes Eye Capture Feature**



## 2.2 Applications

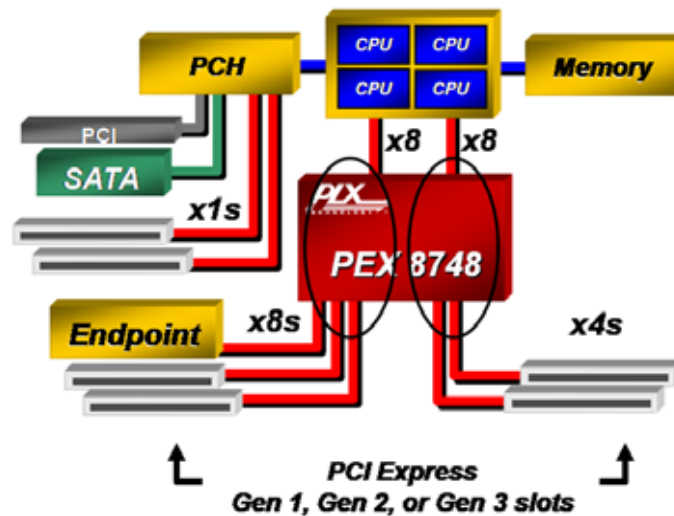
Suitable for **Host-centric** and **peer-to-peer traffic**, the PEX 8748 can be configured for a wide variety of form factors and applications.

### 2.2.1 Host-Centric Fan-Out

The PEX 8748, with its versatile symmetric or asymmetric Lane configuration capability, allows user-specific tuning to a variety of Host-centric applications.

Figure 2-4 illustrates a typical **server** design where, in a quad- or multi-processor system, users can assign endpoints/slots to CPU cores, to distribute the system load. The packets directed to different CPU cores go to different (user-assigned) PEX 8748 Upstream Port(s), providing better queuing and load-balancing capability, for higher performance. Conversely, the PEX 8748 can also be used in Base (Single Host) mode, to simply fan-out to endpoints.

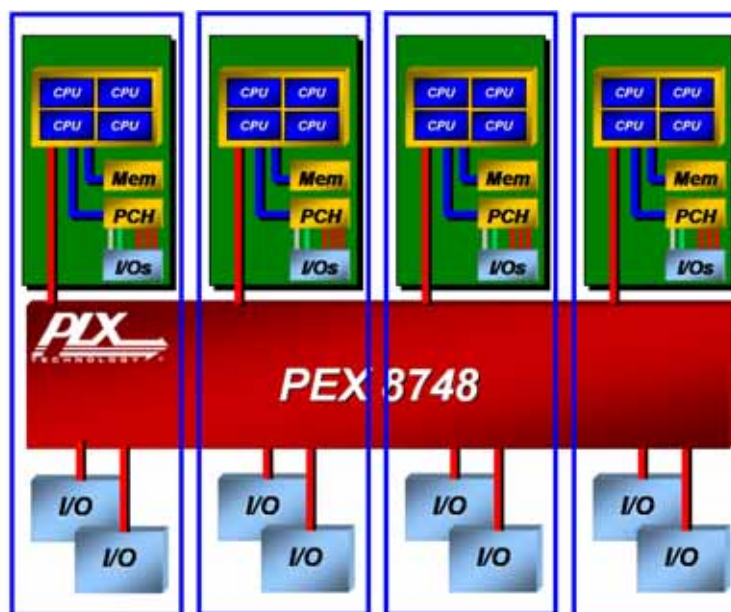
Figure 2-4. Host-Centric, Dual-Upstream



## 2.2.2 Multi-Host Systems

In Virtual Switch mode, with the ability to create and concurrently support up to six virtual switches/ Hosts, the PEX 8748 allows each Host to fan-out to its respective endpoints. This reduces the quantity of switches required for fan-out, saving precious board space and power consumption. In [Figure 2-5](#), the PEX 8748 is being shared by four different servers (Hosts), with each server running its own applications (I/Os). The PEX 8748 assigns the endpoints to the appropriate Host, and isolates them from the other Hosts.

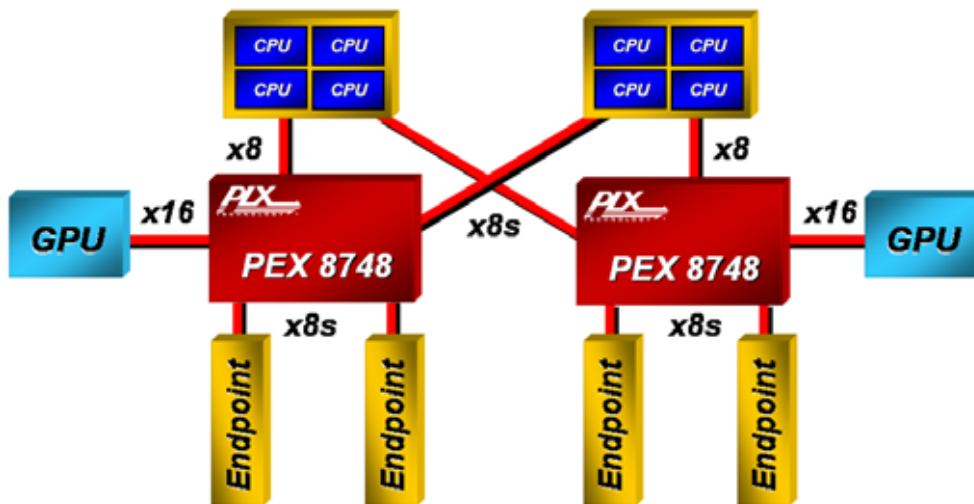
Figure 2-5. Multi-Host Systems



### 2.2.3 Host Failover

The PEX 8748 can also be used in applications where Host Failover is required. In the application illustrated in [Figure 2-6](#), two Hosts can be simultaneously active and controlling their own domains, while exchanging status information through **Doorbell** registers and/or the I<sup>2</sup>C Slave interface. The devices can be programmed to trigger failover if the heartbeat information is not provided. In the event of a failure, the surviving device resets the endpoints connected to the failing CPU, then enumerates those endpoints within its own domain, without impacting the operation of endpoints already within its domain.

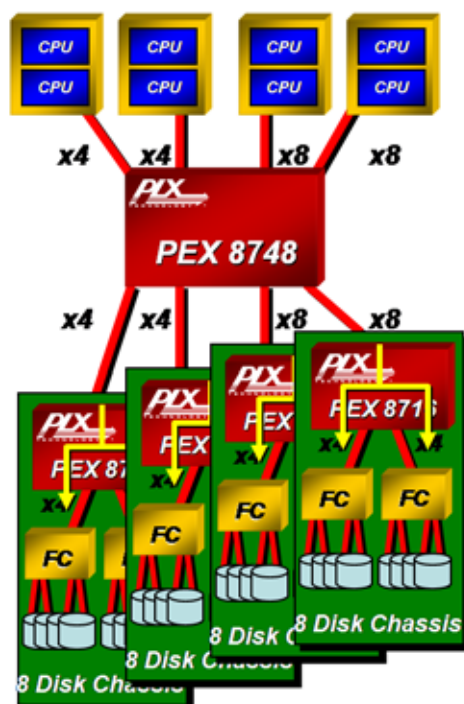
Figure 2-6. Host Failover



## 2.2.4 N+1 Failover in Storage Systems

The PEX 8748's Multi-Host feature can also be used to develop storage array clusters, where each Host manages a set of storage devices, independent of the other Hosts. Users can designate one of the Hosts as the Failover Host for all the other Hosts, while actively managing its own endpoints. The Failover Host communicates with the other Hosts for status/heartbeat information, and executes a Failover event if/when a Failover event is triggered. (Refer to [Figure 2-7](#).)

Figure 2-7. N+1 Failover



## 2.3 Software Usage Model

From the system model viewpoint, each PCI Express Port is a virtual PCI-to-PCI bridge, with its own set of PCI Express Configuration registers. The recommended Upstream Port in Base mode is Port 0; however, any Port can be configured as the Upstream Port through optional configuration, by way of a serial EEPROM, the I<sup>2</sup>C Slave interface, and/or Strapping balls. The BIOS or Host can configure the other Ports, by way of the Upstream Port, using Conventional PCI enumeration. In Virtual Switch mode, any Port within the same virtual hierarchy can be designated as the Upstream Port for that particular hierarchy.

### 2.3.1 System Configuration

The virtual PCI-to-PCI bridges within the PEX 8748 are compliant to the PCI and PCI Express system models. The Configuration Space registers (CSRs) in a virtual primary/secondary PCI-to-PCI bridge are accessible by Type 0 and Type 1 Configuration Requests, through the virtual primary bus interface (matching Bus Number, Device Number, and Function Number).

### 2.3.2 Interrupt Sources and Events

The PEX 8748 supports the INTx Interrupt Message type (compatible with *PCI r3.0* Interrupt signals) or Message Signaled Interrupts (MSIs), when enabled. The PEX 8748 generates interrupts/Messages for the following:

- Hot Plug or Link State events
- PCI Express Hot Plug events
- Device-Specific NT-Link Port events and/or errors
- GPIO-generated events
- NT Doorbell-generated events (NT mode only)
- Management Port Doorbell events (Transparent Ports and NT Port Virtual Interface only)
- Management Link Status events (Transparent Ports and NT Port Virtual Interface only)
- Baseline and Advanced Error Reporting

Internally generated interrupts and interrupts forwarded from Downstream Ports are re-mapped and collapsed at the Upstream Port(s).



## Chapter 3 Signal Ball Description

### 3.1 Introduction

This chapter describes the following signal ball-related information:

- [Ball Assignment Abbreviations](#)
- [Internal Pull-Up/Pull-Down Resistors](#)
- [Signal Ball Descriptions](#)
- [Physical Layout](#)

## 3.2 Ball Assignment Abbreviations

Table 3-1 lists the abbreviations used within this chapter.

**Table 3-1. Ball Assignment Abbreviations**

Abbreviation	Description
#	Active-Low signal
3-state input	Associated with the Strapping signals. Can be High (1), Low (0), or High-Impedance (Z; not connected/floating).
A	Analog Input signal
APWR	Power (VDD09A) balls for SerDes Analog circuits
HCSLn <sup>a</sup>	High-Speed Current Steering Logic (HCSL) negative Clock inputs
HCSLp <sup>a</sup>	HCSL positive Clock inputs
CMLRn	Differential low-voltage, high-speed, CML negative Receiver inputs
CMLRp	Differential low-voltage, high-speed, CML positive Receiver inputs
CMLTn	Differential low-voltage, high-speed, CML negative Transmitter outputs
CMLTp	Differential low-voltage, high-speed, CML positive Transmitter outputs
CPWR	0.9V Power (VDD09) balls for low-voltage Core circuits
GND	Common Ground (VSS) for all circuits
I	Input
I/O	Bidirectional (Input or Output)
I/OPWR	1.8V Power (VDD18) balls for Input and Output interfaces
O	Output
OD	Open Drain output
PD	Weak internal pull-down resistor
PLLWR	1.8V Power (VDD18A) balls for Phase-Locked Loop (PLL) circuits
PU	Weak internal pull-up resistor
SerDes	Serializer/De-Serializer differential low-voltage, high-speed, I/O signal pairs (negative and positive)
STRAP	Signals used for PEX 8748 configuration, operational mode Setting, and <b>Factory Test</b> ; these signals generally are not toggled at runtime
TS <sup>b</sup>	Tri-statable Output (High-Impedance when output is not driven)

a. For REFCLK inputs, HCSL source is recommended.

b. Used in the Signal tables, **Type** column. All other use of TS is for “Training Sequence.”



### 3.3 Internal Pull-Up/Pull-Down Resistors

The PEX 8748 contains I/O buffers that have weak internal pull-up or pull-down resistors, indicated in this chapter by PU or PD, respectively, in the signal ball tables (**Type** column). If a signal with this notation is used and no board trace is connected to the ball, the internal resistor is usually sufficient to keep the signal from toggling. However, if a signal with this notation is not used, but is connected to a board trace and is not used nor driven by an external source at all times, the internal resistors might not be sufficiently strong, to hold the signal in the inactive state. In cases such as these, it is recommended that the signal be pulled or tied High to **VDD18** or Low to **VSS** (Ground), as appropriate, through a 3K $\Omega$  to 10K $\Omega$  resistor.

Table 3-2 lists the internal pull-up and pull-down resistor values.

**Table 3-2. Internal Resistor Values**

Internal Resistor	Minimum	Typical	Maximum	Units
PU	38K	57K	92K	$\Omega$
PD	37K	53K	99K	$\Omega$

## 3.4 Signal Ball Descriptions

**Note:** *If there is more than one ball per signal name that includes a numbered range, the locations are listed in the same sequence in which the range is listed, starting at the top row, from left to right. For example, PEX\_PERn11 is located at AC17, PEX\_PERn10 is located at AC16, and so forth.*

*If there is more than one ball per signal name that does not include a numbered range (such as VDD09), the locations are listed in ascending alphanumeric order.*

The PEX 8748 signals are divided into the following groups:

- PCI Express Signals
- Hot Plug Signals
  - Parallel Hot Plug Signals
  - Serial Hot Plug Signals
- Serial EEPROM Signals
- Strapping Signals
  - STRAP\_TESTMODE[2:0] 3-State Input Configurations
- JTAG Interface Signals
- I2C/SMBus Slave Interface Signals
- Device-Specific Signals
- External Resistor Signals
- Thermal Diode Signals
- No Connect Signals
- Power and Ground Signals

### 3.4.1 PCI Express Signals

Table 3-3 defines the PCI Express SerDes and Control signals.

**Table 3-3. PCI Express Signals**

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
PEX_PERn[15:0]	AC22, AC21, AC20, AC19, AC17, AC16, AC15, AC14, AC13, AC12, AC11, AC10, AC8, AC7, AC6, AC5	CMLRn	No	–
		<b>Negative Half of PCI Express Receiver Differential Signal Pairs for Station 0 (16 Balls)</b>		
PEX_PERn[31:16]	D13, D12, D11, D10, D8, D7, D6, D5, J4, K4, L4, M4, P4, R4, T4, U4	CMLRn	No	–
		<b>Negative Half of PCI Express Receiver Differential Signal Pairs for Station 1 (16 Balls)</b>		
PEX_PERn[47:32]	D14, D15, D16, D17, D19, D20, D21, D22, J23, K23, L23, M23, P23, R23, T23, U23	CMLRn	No	–
		<b>Negative Half of PCI Express Receiver Differential Signal Pairs for Station 2 (16 Balls)</b>		
PEX_PERp[15:0]	AB22, AB21, AB20, AB19, AB17, AB16, AB15, AB14, AB13, AB12, AB11, AB10, AB8, AB7, AB6, AB5	CMLRp	No	–
		<b>Positive Half of PCI Express Receiver Differential Signal Pairs for Station 0 (16 Balls)</b>		
PEX_PERp[31:16]	E13, E12, E11, E10, E8, E7, E6, E5, J5, K5, L5, M5, P5, R5, T5, U5	CMLRp	No	–
		<b>Positive Half of PCI Express Receiver Differential Signal Pairs for Station 1 (16 Balls)</b>		
PEX_PERp[47:32]	E14, E15, E16, E17, E19, E20, E21, E22, J22, K22, L22, M22, P22, R22, T22, U22	CMLRp	No	–
		<b>Positive Half of PCI Express Receiver Differential Signal Pairs for Station 2 (16 Balls)</b>		
PEX_PERST#	AC1	I, PU	No	–
		<b>PCI Express Reset</b> Used to cause a Fundamental Reset.  <b>Base Mode</b> Refer to <a href="#">Section 5.1.1, “Resets – Base Mode,”</a> for further details.  <b>Virtual Switch Mode</b> In Virtual Switch mode, PEX_PERST# assertion resets all virtual switches. Refer to <a href="#">Section 5.1.2, “Resets – Virtual Switch Mode,”</a> for further details.  <i><b>Note:</b> The VSx_PERST# signals, defined in <a href="#">Table 3-14</a>, are the Reset inputs for individual virtual switches.</i>		

Table 3-3. PCI Express Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
PEX_PETn[15:0]	AF22, AF21, AF20, AF19, AF17, AF16, AF15, AF14, AF13, AF12, AF11, AF10, AF8, AF7, AF6, AF5	CMLTn	No	–
		<b>Negative Half of PCI Express Transmitter Differential Signal Pairs for Station 0 (16 Balls)</b> PEX_PETn[15:0] must be AC-coupled. Use a 0.22 µF capacitor.		
PEX_PETn[31:16]	A13, A12, A11, A10, A8, A7, A6, A5, J1, K1, L1, M1, P1, R1, T1, U1	CMLTn	No	–
		<b>Negative Half of PCI Express Transmitter Differential Signal Pairs for Station 1 (16 Balls)</b> PEX_PETn[31:16] must be AC-coupled. Use a 0.22 µF capacitor.		
PEX_PETn[47:32]	A14, A15, A16, A17, A19, A20, A21, A22, J26, K26, L26, M26, P26, R26, T26, U26	CMLTn	No	–
		<b>Negative Half of PCI Express Transmitter Differential Signal Pairs for Station 2 (16 Balls)</b> PEX_PETn[47:32] must be AC-coupled. Use a 0.22 µF capacitor.		
PEX_PETp[15:0]	AE22, AE21, AE20, AE19, AE17, AE16, AE15, AE14, AE13, AE12, AE11, AE10, AE8, AE7, AE6, AE5	CMLTp	No	–
		<b>Positive Half of PCI Express Transmitter Differential Signal Pairs for Station 0 (16 Balls)</b> PEX_PETp[15:0] must be AC-coupled. Use a 0.22 µF capacitor.		
PEX_PETp[31:16]	B13, B12, B11, B10, B8, B7, B6, B5, J2, K2, L2, M2, P2, R2, T2, U2	CMLTp	No	–
		<b>Positive Half of PCI Express Transmitter Differential Signal Pairs for Station 1 (16 Balls)</b> PEX_PETp[31:16] must be AC-coupled. Use a 0.22 µF capacitor.		
PEX_PETp[47:32]	B14, B15, B16, B17, B19, B20, B21, B22, J25, K25, L25, M25, P25, R25, T25, U25	CMLTp	No	–
		<b>Positive Half of PCI Express Transmitter Differential Signal Pairs for Station 2 (16 Balls)</b> PEX_PETp[47:32] must be AC-coupled. Use a 0.22 µF capacitor.		
PEX_REFCLKn	AF9	HCSLn	No	–
		<b>Negative Half of 100-MHz PCI Express Constant Frequency Reference Clock Input Signal Pair</b> An internal coupling capacitor is provided (an external capacitor is not needed).		
PEX_REFCLKp	AE9	HCSLp	No	–
		<b>Positive Half of 100-MHz PCI Express Constant Frequency Reference Clock Input Signal Pair</b> An internal coupling capacitor is provided (an external capacitor is not needed).		

## 3.4.2 Hot Plug Signals

The PEX 8748 includes signals for both Parallel and Serial Hot Plug support.

Parallel Hot Plug can be implemented on any of three Transparent Downstream Ports, as selected by the **Parallel Hot Plug Control** register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 3A4h[7:0, 15:8, and 23:16] for Parallel Hot Plug Controllers A, B, and C, respectively).

Serial Hot Plug can be implemented on any Transparent Downstream Port. If a Transparent Downstream Port is both Parallel- and Serial Hot Plug-capable, the Serial Hot Plug Controller is used, by default, unless the Port's **Power Management Hot Plug User Configuration** register *Serial Hot Plug Override Parallel Disable* bit (offset F70h[19]) is Set.

Hot Plug signals are enabled, configured, and accessed through the Transparent Downstream Port's **Slot Capability** and **Slot Status and Control** registers (Downstream Ports, offsets 7Ch and 80h, respectively). Also, each Port's **Power Management Hot Plug User Configuration** register provides additional Device-Specific configuration and control, for both Parallel and Serial Hot Plug implementations.

Both signal types are discussed in the sections that follow.

### 3.4.2.1 Parallel Hot Plug Signals

The PEX 8748 includes 10 signal balls, per Hot Plug-capable Port, that support the Parallel Hot Plug Controller (HP\_), as defined in Table 3-4. These signals are active only for Hot Plug-capable Transparent Downstream Ports configured at start-up.

For further details regarding Hot Plug, refer to Chapter 10, “Hot Plug Support.”

**Notes:** All Parallel Hot Plug signals are I/O; however, their logical operation is either input or output, as described for each signal.

All Parallel Hot Plug signals are duplicated for each Hot Plug-capable Port, as A, B, and C signals, which map to any Transparent Downstream Port, as selected by the **Parallel Hot Plug Control** register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 3A4h[7:0, 15:8, and 23:16] for Parallel Hot Plug Controllers A, B, and C, respectively).

Table 3-4. Parallel Hot Plug Signals for Ports C, B, A<sup>a</sup>

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
HP_ATNLED_[C, B, A]#	B25, AB1, AC26	I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a>
		<p><b>Hot Plug Attention LED Outputs (3 Balls)</b></p> <p>Active-Low Slot Control Logic outputs that are used to drive the Attention Indicator. Asserted Low to turn On (illuminate) the LED.</p> <p>Enabled when the Port's <b>Slot Capability</b> register <i>Attention Indicator Present</i> bit (Downstream Ports, offset 7Ch[3]) is Set and controlled by the Port's <b>Slot Control</b> register <i>Attention Indicator Control</i> field (Downstream Ports, offset 80h[7:6]). When software writes to the <i>Attention Indicator Control</i> field, a Command Completed interrupt can be generated to notify the Host that the command has been executed.</p> <p>An interrupt (INTx Message, MSI, or <a href="#">PEX_INTA#</a> (Base mode) or <a href="#">VSx_PEX_INTA#</a> (Virtual Switch mode) output, all mutually exclusive, on a per-Port basis) can be generated to the Host, when the following conditions are met:</p> <ul style="list-style-type: none"> <li>Port's <b>Slot Capability</b> register <i>Attention Indicator Present</i> bit (Downstream Ports, offset 7Ch[3]) is Set, and</li> <li>Port's <b>Slot Control</b> register <i>Command Completed Interrupt Enable</i> bit is not masked (Downstream Ports, offset 80h[4], is Set), and</li> <li>Port's <b>Slot Control</b> register <i>Hot Plug Interrupt Enable</i> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[5]) is Set.</li> </ul> <p><b>If HP_ATNLED_x# outputs are used, each requires an external current-limiting resistor. If HP_ATNLED_x# functionality is enabled but the outputs are not used, the signals can remain unconnected.</b></p> <p><i>Note:</i> Although these are I/O signals, their logical operation is output.</p>		

**Table 3-4. Parallel Hot Plug Signals for Ports C, B, A<sup>a</sup> (Cont.)**

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
HP_BUTTON_[C, B, A]#	C25, AB3, Y24	I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a>
		<p><b>Hot Plug Attention Button Inputs (3 Balls)</b></p> <p>Active-Low Slot Control Logic inputs that are connected directly to the Attention Button, with input assertion status latched in the Port's <b>Slot Status</b> register <i>Attention Button Pressed</i> bit (Downstream Ports, offset 80h[16]).</p> <p>Enabled when the Port's <b>Slot Capability</b> register <i>Attention Button Present</i> bit (Downstream Ports, offset 7Ch[0]) is Set.</p> <p>An interrupt (INT<sub>x</sub> Message, MSI, or <b>PEX_INTA#</b> (Base mode) or VS<sub>x</sub> <b>PEX_INTA#</b> (Virtual Switch mode) output, all mutually exclusive, on a per-Port basis) can be generated, to notify the Host of intended board insertion or removal, when the following conditions are met:</p> <ul style="list-style-type: none"> <li>Port's HP_BUTTON_<sub>x</sub># is not masked (Port's <b>Slot Control</b> register <i>Attention Button Pressed Enable</i> bit (Downstream Ports, offset 80h[0]) is Set), and</li> <li>Port's <b>Slot Capability</b> register <i>Hot Plug Capable</i> bit (Downstream Ports, offset 7Ch[6]) is Set, and</li> <li>Port's <b>Slot Control</b> register <i>Hot Plug Interrupt Enable</i> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[5]) is Set.</li> </ul> <p><b>If HP_BUTTON_<sub>x</sub># functionality is enabled but the inputs are not used, the signals can remain unconnected.</b></p> <p><i>Notes: HP_BUTTON_<sub>x</sub># is internally de-bounced, but must remain stable for at least 10 ms.</i></p> <p><i>Although these are I/O signals, their logical operation is input.</i></p>		
HP_CLKEN_[C, B, A]#	F24, AE1, AA24	I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a>
		<p><b>Hot Plug Reference Clock Enable Outputs (3 Balls)</b></p> <p>Active-Low Slot Control Logic outputs that control the connection of the external REFCLK to the slot.</p> <p>Enabled when the Port's <b>Slot Capability</b> register <i>Power Controller Present</i> bit (Downstream Ports, offset 7Ch[1]) is Set, and controlled by the Port's <b>Slot Control</b> register <i>Power Controller Control</i> bit (Downstream Ports, offset 80h[10]).</p> <p>The time delay from HP_PWREN_<sub>x</sub> output assertion to HP_CLKEN_<sub>x</sub># output assertion is programmable (through serial EEPROM load) from 128 to 512 ms, in the Port's <b>Power Management Hot Plug User Configuration</b> register <i>HPC Tpepv</i> field (Downstream Ports, offset F70h[4:3]). When this register field is programmed to 00b (default), HP_PWR_GOOD_<sub>x</sub> input assertion controls the time delay from HP_PWREN_<sub>x</sub> output assertion to HP_CLKEN_<sub>x</sub># output assertion.</p> <p><b>If HP_CLKEN_<sub>x</sub># functionality is enabled but the outputs are not used, the signals can remain unconnected.</b></p> <p><i>Note: Although these are I/O signals, their logical operation is output.</i></p>		

Table 3-4. Parallel Hot Plug Signals for Ports C, B, A<sup>a</sup> (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
HP_MRL_[C, B, A]#	G21, AB2, AB26	I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a>
		<p><b>Hot Plug Manually Operated Retention Latch Sensor Inputs (3 Balls)</b></p> <p>Active-Low Slot Control Logic inputs that are connected directly to an optional Manually operated Retention Latch (MRL) Sensor that is logic Low when the latch is closed.</p> <p>Enabled when the Port's <b>Slot Capability</b> register <i>MRL Sensor Present</i> bit (Downstream Ports, offset 7Ch[2]) is Set.</p> <p>When enabled, HP_MRL_x# input assertion enables Hot Plug output sequencing to turn On the slot's power (HP_PWREN_x and HP_PWRLED_x#) and clock (HP_CLKEN_x#), and de-assert Reset (HP_PERST_x#) after reset, as illustrated in <a href="#">Figure 10-2, "Hot Plug Outputs when Power Controller Present and Power Controller Control Bits Are Cleared,"</a> or under software control.</p> <p>A change in the HP_MRL_x# input states is latched in the Port's <b>Slot Status</b> register <i>MRL Sensor Changed</i> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[18]), and the state change can assert an interrupt to notify the Host of a change in the MRL Sensor state.</p> <p>An interrupt (INTx Message, MSI, or <a href="#">PEX_INTA#</a> (Base mode) or VSx_PEX_INTA# (Virtual Switch mode) output, all mutually exclusive, on a per-Port basis) can be generated, when the following conditions are met:</p> <ul style="list-style-type: none"> <li>Port's HP_MRL_x# is not masked (Port's <b>Slot Control</b> register <i>MRL Sensor Changed Enable</i> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[2]) is Set), and</li> <li>Port's <b>Slot Control</b> register <i>Hot Plug Interrupt Enable</i> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[5]) is Set.</li> </ul> <p>If the associated Hot Plug-capable Transparent Downstream Port connects to a PCI Express board slot that does not implement an MRL Sensor, HP_MRL_x# are typically connected to HP_PRSNT_x# and a pull-up resistor, with the common node connected to the PRSNT2# signal(s) at the slot.</p> <p>If the associated Hot Plug-capable Transparent Downstream Port instead connects directly to a device, and Hot Plug signals (<i>such as</i> HP_PERST_x#) are used, pull HP_MRL_x# Low to <b>VSS</b> (Ground), to enable the automatic Hot Plug output sequencing following switch Reset (<a href="#">PEX_PERST#</a> or Hot Reset) de-assertion. <b>Otherwise, if Hot Plug signals are not used, HP_MRL_x# can remain unconnected.</b></p> <p>When an HP_MRL_x# input is enabled and toggled High on a powered slot, REFCLK to the slot is automatically disconnected and the slot power is automatically turned Off, as illustrated in <a href="#">Figure 10-3, "Hot Plug Automatic Slot Power-Down Sequence."</a></p> <p><b>Notes:</b> HP_MRL_x# is internally de-bounced, but must remain stable for at least 10 ms. HP_MRL_x#, if enabled, is not de-bounced when sampled immediately after reset.</p> <p><i>Although these are I/O signals, their logical operation is input.</i></p>		



**Table 3-4. Parallel Hot Plug Signals for Ports C, B, A<sup>a</sup> (Cont.)**

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
HP_PERST_[C, B, A]#	G22, AD1, AB25	I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a>
		<b>Hot Plug Reset Outputs (3 Balls)</b> Active-Low Slot Control Logic outputs that are used to reset the slot. When the Port's <b>Slot Capability</b> register <i>Power Controller Present</i> bit (Downstream Ports, offset 7Ch[1]) is Set, the HP_PERST_x# output states can be controlled by software, using the Port's <b>Slot Control</b> register <i>Power Controller Control</i> bit (Downstream Ports, offset 80h[10]). <b>If HP_PERST_x# functionality is enabled but the outputs are not used, the signals can remain unconnected.</b> <i>Note: Although these are I/O signals, their logical operation is output.</i>		
HP_PRSNT_[C, B, A]#	D25, W5, AD26	I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a>
		<b>Hot Plug PRSNT2# Inputs (3 Balls)</b> Active-Low Slot Control Logic inputs that connect to the slot's PRSNT2# signal, which on the add-in board connects to the slot's PRSNT1# signal, which is typically grounded on the motherboard. A change in the HP_PRSNT_x# input states is latched in the Port's <b>Slot Status</b> register <i>Presence Detect Changed</i> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[19]), and the state change can assert an interrupt, to notify the Host of board presence or absence. An interrupt (INTx Message, MSI, or <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode) output, all mutually exclusive, on a per-Port basis) can be generated, when the following conditions are met: <ul style="list-style-type: none"> <li>Port's HP_PRSNT_x# is not masked (Port's <b>Slot Control</b> register <i>Presence Detect Changed Enable</i> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[3]) is Set), and</li> <li><b>Slot Control</b> register <i>Hot Plug Interrupt Enable</i> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[5]) is Set.</li> </ul> <b>If HP_PRSNT_x# functionality is enabled but the inputs are not used, the signals can remain unconnected.</b> <i>Notes: HP_PRSNT_x# is internally de-bounced, but must remain stable for at least 10 ms.</i> <i>Although these are I/O signals, their logical operation is input.</i>		

Table 3-4. Parallel Hot Plug Signals for Ports C, B, A<sup>a</sup> (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
HP_PWREN_[C, B, A]	D24, AC2, W22	I/O, PD	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a>
		<p><b>Hot Plug Power Enable Outputs (3 Balls)</b></p> <p>Active-High Slot Control Logic outputs that control the slot power state. When these outputs are High, power is enabled to the slot(s).</p> <p>Enabled when the Port's <b>Slot Capability</b> register <i>Power Controller Present</i> bit (Downstream Ports, offset 7Ch[1]) is Set.</p> <p>When software turns the slot's Power Controller On or Off (Port's <b>Slot Control</b> register <i>Power Controller Control</i> bit (Downstream Ports, offset 80h[10])), a Command Completed interrupt can be generated, to notify the Host that the command has been executed.</p> <p>An interrupt (INTx Message, MSI, or <a href="#">PEX_INTA#</a> (Base mode) or <a href="#">VSx_PEX_INTA#</a> (Virtual Switch mode) output, all mutually exclusive, on a per-Port basis) can be generated to the Host, when the following conditions are met:</p> <ul style="list-style-type: none"> <li>Port's <b>Slot Control</b> register <i>Command Completed Interrupt Enable</i> bit is not masked (Downstream Ports, offset 80h[4], is Set), and</li> <li>Port's <b>Slot Control</b> register <i>Hot Plug Interrupt Enable</i> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[5]) is Set.</li> </ul> <p>When HP_MRL_x# are enabled (Port's <b>Slot Capability</b> register <i>MRL Sensor Present</i> bit (Downstream Ports, offset 7Ch[2]) is Set), HP_MRL_x# input assertion enables Hot Plug output sequencing to turn On the slot's power, by asserting HP_PWREN_x after reset, as illustrated in <a href="#">Figure 10-2, "Hot Plug Outputs when Power Controller Present and Power Controller Control Bits Are Cleared,"</a> or under software control.</p> <p><b>If HP_PWREN_x functionality is enabled but the outputs are not used, the signals can remain unconnected.</b></p> <p><i>Notes:</i> HP_PWREN_x polarity is inverted with respect to HP_PWRENx# functionality in PLX ExpressLane Gen 1 switches.</p> <p>Although these are I/O signals, their logical operation is output.</p>		

**Table 3-4. Parallel Hot Plug Signals for Ports C, B, A<sup>a</sup> (Cont.)**

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
HP_PWRFLT_[C, B, A]#	B26, AC3, AE26	I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a>
		<p><b>Hot Plug Power Fault Inputs (3 Balls)</b></p> <p>Slot Control Logic inputs that, when asserted, indicate that the slot's external Power Controller detected a power fault on one or more supply rails.</p> <p>By default (with <a href="#">SPARE1</a>=H when sampled at <a href="#">PEX_PERST#</a> input de-assertion), HP_PWRFLT_<i>x</i># are Active-Low; otherwise, if <a href="#">SPARE1</a>=L, HP_PWRFLT_<i>x</i> are Active-High (however, the Serial Hot Plug <a href="#">PWRFLT#</a> inputs from the external I<sup>2</sup>C I/O Expanders are always Active-Low).</p> <p>Enabled when the Port's <a href="#">Slot Capability</a> register <a href="#">Power Controller Present</a> bit (Downstream Ports, offset <a href="#">7Ch[1]</a>) is Set, and input assertion status is latched in the Port's <a href="#">Slot Status</a> register <a href="#">Power Fault Detected</a> bit (Downstream Ports, offset <a href="#">80h[17]</a>).</p> <p>An interrupt (INT<sub>x</sub> Message, MSI, or <a href="#">PEX_INTA#</a> (Base mode) or VS<sub>x</sub>_<a href="#">PEX_INTA#</a> (Virtual Switch mode) output, all mutually exclusive, on a per-Port basis) can be generated, to notify the Host of a power fault, when the following conditions are met:</p> <ul style="list-style-type: none"> <li>Port's HP_PWRFLT_<i>x</i># is not masked (Port's <a href="#">Slot Control</a> register <a href="#">Power Fault Detector Enable</a> bit (Downstream Ports, offset <a href="#">80h[1]</a>) is Set), and</li> <li>Port's <a href="#">Slot Control</a> register <a href="#">Hot Plug Interrupt Enable</a> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset <a href="#">80h[5]</a>) is Set.</li> </ul> <p><b>If HP_PWRFLT_<i>x</i># functionality is enabled but the inputs are not used, the signals can remain unconnected.</b></p> <p><i>Notes:</i> If <a href="#">HP_PWREN_<i>x</i></a> and <a href="#">HP_CLKEN_<i>x</i></a> are not used, <a href="#">HP_PWRFLT_<i>x</i></a># can be used as general-purpose inputs, with status reflected in the Port's <a href="#">Power Fault Detected</a> bit, provided that the Port's <a href="#">Power Controller Present</a> bit is Set.</p> <p><i>Although these are I/O signals, their logical operation is input.</i></p>		

**Table 3-4. Parallel Hot Plug Signals for Ports C, B, A<sup>a</sup> (Cont.)**

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
HP_PWR_GOOD_[C, B, A]	F25, Y3, W21	I/O, PD	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a>
		<p><b>Hot Plug Power Good Inputs (3 Balls)</b></p> <p>Active-High (default) inputs that, when enabled (default), cause the Slot Control Logic to delay HP_CLKEN_# output assertion (to turn On REFCLK to the slot), until HP_PWR_GOOD_x inputs are asserted to indicate that the installed module's power supplies are active and stable.</p> <p>Signal polarity can be changed to Active-Low, by programming the serial EEPROM to Set the Port's <b>Power Management Hot Plug User Configuration</b> register <i>HP_PWR_GOOD_x Active-Low Enable</i> bit (Downstream Ports, offset <a href="#">F70h[6]</a>). <b>Polarity must not be changed by I<sup>2</sup>C, because that is too slow for initialization.</b></p> <p>HP_PWR_GOOD_x are disabled when the Port's register's <i>HPC Tpepv</i> field (Downstream Ports, offset <a href="#">F70h[4:3]</a>) is programmed to a value other than 00b, to cause HP_CLKEN_# output assertion to follow HP_PWREN_x assertion, by a fixed delay (128, 256, or 512 ms).</p> <p><b>If HP_PWR_GOOD_x functionality is enabled but the associated set of Hot Plug signals is not used, these inputs can remain unconnected.</b></p> <p><i>Note:</i> Although these are I/O signals, their logical operation is input.</p>		

**Table 3-4. Parallel Hot Plug Signals for Ports C, B, A<sup>a</sup> (Cont.)**

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
HP_PWRLED_[C, B, A]#	D26, AD2, Y23	I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a>
		<p><b>Hot Plug Power LED Outputs (3 Balls)</b></p> <p>Active-Low Slot Control Logic outputs that are used to drive the Power Indicator. Asserted Low to turn On (illuminate) the LED.</p> <p>Enabled when the Port's <b>Slot Capability</b> register <i>Power Indicator Present</i> bit (Downstream Ports, offset 7Ch[4]) is Set, and controlled by the Port's <b>Slot Control</b> register <i>Power Indicator Control</i> field (Downstream Ports, offset 80h[9:8]). When software writes to the <i>Power Indicator Control</i> field, a Command Completed interrupt can be generated, to notify the Host that the command has been executed.</p> <p>An interrupt (INT<sub>x</sub> Message, MSI, or <b>PEX_INTA#</b> (Base mode) or <b>VS<sub>x</sub> PEX_INTA#</b> (Virtual Switch mode) output, all mutually exclusive, on a per-Port basis) can be generated to the Host, when the following conditions are met:</p> <ul style="list-style-type: none"> <li>Port's <b>Slot Capability</b> register <i>Power Indicator Present</i> bit (Downstream Ports, offset 7Ch[4]) is Set, and</li> <li>Port's <b>Slot Control</b> register <i>Command Completed Interrupt Enable</i> bit is not masked (Downstream Ports, offset 80h[4], is Set), and</li> <li>Port's <b>Slot Control</b> register <i>Hot Plug Interrupt Enable</i> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[5]) is Set.</li> </ul> <p><b>If HP_PWRLED_x# outputs are used, each requires an external current-limiting resistor. If HP_PWRLED_x# functionality is enabled but the outputs are not used, the signals can remain unconnected.</b></p> <p><i>Note:</i> Although these are I/O signals, their logical operation is output.</p>		

- a. If Hot Plug outputs are used and HP\_MRL\_x# inputs are not used, pull the HP\_MRL\_x# inputs Low, so that Hot Plug outputs will properly sequence if the serial EEPROM is blank or missing. Default register values enable HP\_MRL\_x#, which must then be asserted, to cause Hot Plug outputs to toggle (for example, to de-assert HP\_PERST\_x# and assert HP\_PWRLED\_x#).

### 3.4.2.2 Serial Hot Plug Signals

Transparent Downstream Ports can implement Hot Plug, by using external I<sup>2</sup>C I/O Expanders (one 16-pin Maxim MAX7311, NXP PCA9555, or TI PCA9555 per slot, –or– one 40-pin NXP PCA9698 per two slots). All Ports implementing Serial Hot Plug can concurrently use either type of I/O Expander (16- or 40-pin). The Serial Hot Plug Controller queries each I/O Expander for its Device ID. 40-Pin I/O Expanders implement Device ID, and 16-pin I/O Expanders do not. If the device responds to the PEX 8748's Device ID query, the Serial Hot Plug Controller assumes that the I/O Expander is a 40-pin device. The query can be disabled, by Setting the Port's **Power Management Hot Plug User Configuration** register *40-Pin I/O Expander Scan Disable* bit (offset F70h[17]).

Table 3-4 defines the signal balls that support Serial Hot Plug. Additionally, the PEX 8748 supports external Serial Hot Plug signals on the I<sup>2</sup>C I/O Expander. (Refer to Section 10.11.2, “I<sup>2</sup>C I/O Expander Parts Selection and Pin Definition.”)

Table 3-5. Serial Hot Plug Signals

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
I2C_SCL1	AC25	OD	No	—
		<b>I<sup>2</sup>C Serial Clock Line for Serial Hot Plug Support</b> I <sup>2</sup> C Clock source. Used with the external I <sup>2</sup> C I/O Expander, and must be bused to each I/O Expander's Clock (SCL) pin. In combination with I2C_SDA1, forms the PEX 8748 I <sup>2</sup> C Master interface. Because the 1.8V signaling is less than the voltage defined by the <i>I<sup>2</sup>C Bus v2.1</i> (5V, 3.3V), external voltage translation circuitry might be required. <b>I2C_SCL1 requires an external pull-up resistor.</b>		
I2C_SDA1	AE25	OD	No	—
		<b>I<sup>2</sup>C Serial Data Line for Serial Hot Plug Support</b> Transmits and receives I <sup>2</sup> C data. Used with the external I <sup>2</sup> C I/O Expander, and must be bused to each I/O Expander's Data (SDA) pin. In combination with I2C_SCL1, forms the PEX 8748 I <sup>2</sup> C Master interface. Because the 1.8V signaling is less than the voltage defined by the <i>I<sup>2</sup>C Bus v2.1</i> (5V, 3.3V), external voltage translation circuitry might be required. <b>I2C_SDA1 requires an external pull-up resistor.</b>		
SHPC_INT#	B3	I/O, PU	No	—
		<b>Serial Hot Plug Controller Interrupt Input</b> Active-Low interrupt input from external I <sup>2</sup> C I/O Expanders. Used only by Serial Hot Plug-capable Transparent Downstream Ports. The I/O Expander asserts its INT# output whenever any of its inputs change state, and de-asserts its INT# output when the corresponding Input Port Data register (that changed state) is read. When the SHPC_INT# Interrupt input (connected to the INT# output of all I/O Expanders) is asserted, the I <sup>2</sup> C Master interface begins reading the Input Port registers of all I/O Expanders, and copies the values to the appropriate bits in the Port's <b>Slot Status</b> register (Downstream Ports, offset 80h). The I <sup>2</sup> C Master interface halts the reading of I/O Expander registers when the SHPC_INT# input de-asserts. <b>If used, SHPC_INT# input requires an external pull-up resistor; otherwise, this input can remain unconnected.</b> <i>Notes:</i> By default, SHPC_INT# is internally de-bounced, but must remain stable for at least 10 ms. Internal de-bouncing can be disabled, by Setting the Port's <b>Power Management Hot Plug User Configuration</b> register <i>Serial Hot Plug INTx De-Bounce Disable</i> bit (offset F70h[18]). Although this is an I/O signal, its logical operation is input.		

### 3.4.3 Serial EEPROM Signals

The PEX 8748 includes four signals for interfacing to a serial EEPROM, defined in [Table 3-6](#). The serial EEPROM is used to load register values immediately following Reset ([PEX\\_PERST#](#) or Hot Reset/DL\_Down). Most registers can be programmed by serial EEPROM, as indicated in the register descriptions.

For information regarding serial EEPROM use, refer to [Chapter 6, “Serial EEPROM Controller.”](#)

**Table 3-6. Serial EEPROM Signals**

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
EE_CS#	E26	I/O, PU	No	—
		<b>Active-Low Serial EEPROM Chip Select Output</b> <i>Note:</i> Although this is an I/O signal, its logical operation is output.		
EE_DI	G23	I/O, TS	No	—
		<b>PEX 8748 Output to Serial EEPROM Data Input</b> <i>Note:</i> Although this is an I/O signal with tri-statable output, its logical operation is output that is always driven.		
EE_DO	F26	I/O, PU	No	—
		<b>PEX 8748 Input from Serial EEPROM Data Output</b> Should be pulled High to <a href="#">VDD18</a> . <i>Note:</i> Although this is an I/O signal, its logical operation is input.		
EE_SK	E25	I/O, PU	No	—
		<b>Serial EEPROM Clock Frequency Output</b> Programmable, by way of the <a href="#">Serial EEPROM Clock Frequency</a> register <a href="#">EepFreq[2:0]</a> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset <a href="#">268h[2:0]</a> ), to the following: <ul style="list-style-type: none"> <li>• 1 MHz (default)</li> <li>• 1.98 MHz</li> <li>• 5 MHz</li> <li>• 9.62 MHz</li> <li>• 12.5 MHz</li> <li>• 15.6 MHz</li> <li>• 17.86 MHz</li> </ul> <i>Note:</i> Although this is an I/O signal, its logical operation is output.		



### 3.4.4 Strapping Signals

**Note:** *Table 4-1 indicates which Ports are enabled/used, and when, based upon the STRAP\_STNx\_PORTCFGx input states and/or serial EEPROM programming.*

The PEX 8748 Strapping signals, described in [Table 3-7](#), define the configuration of Upstream Port and NT Port assignment, Link width, and various setup and test modes.

Many of the Strapping signals are I/O 3-state inputs, which can be High (1), Low (0), or High-Impedance (Z; not connected/floating).

Strapping signals that have 2-state inputs (rather than 3-state inputs; indicated simply as “inputs”) use internal pull-up and pull-down resistors to define the default configuration; if the PEX 8748 configuration must be changed from the default, external pull-up and/or pull-down resistors can be connected. External resistors are not required unless the Strapping signals:

- Must be changed from the default logic state, –or–
- Are connected to circuit traces (the internal resistors are relatively weak, and may not be sufficiently strong, to hold circuit traces to the default input states)

After a Fundamental Reset, the **Link Capability**, **Port Configuration**, and **VS0 Upstream** registers (offsets 74h, 300h, and 360h, respectively) capture Strapping ball status. Strapping ball Configuration data can be changed by writing new data to these registers from the serial EEPROM (if present) and/or Management Port. I<sup>2</sup>C/SMBus can also change Strapping ball Configuration data; however, the STRAP\_I2C\_SMBUS\_CFG\_EN# input should be pulled or tied Low to VSS (Ground), to prevent linkup and Host enumeration. Then, when I<sup>2</sup>C programming is complete, I<sup>2</sup>C/SMBus should lastly Set the **Configuration Release** register *Initiate Configuration* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 3ACh[0]), to enable linkup and allow subsequent Host enumeration.

**Table 3-7. Strapping Signals**

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States	
		Description			
STRAP_GEN1_GEN2	H24	I/O, 3-state input	No	—	
		<b>PCI Express Data Rate Select</b> Selects the maximum PCI Express data rate for all Ports, by globally changing the values for <b>Maximum Link Speed</b> and <b>Target Link Speed</b> (offsets 74h[3:0] and 98h[3:0]), respectively), in all Ports. The value of this 3-state input is latched and reflected in the <b>Strap Configuration</b> register <b>Gen 2 Mode Status</b> and <b>Gen 1 Mode Status</b> bits (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 46Ch[12:11], respectively).			
		<b>STRAP_GEN1_GEN2 Input State and Link Speed</b>	<b>Offset 74h[3:0] Value</b>	<b>Offset 98h[3:0] Value</b>	<b>Offset 46Ch[12:11] Value</b>
		0 = Gen 1 (2.5 GT/s)	1h	1h	10b
		Z = Gen 2 (5.0 GT/s)	2h	2h	01b
		1 = Gen 3 (8.0 GT/s)	3h	3h	11b

Table 3-7. Strapping Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
STRAP_I2C_SMBUS_CFG_EN#	W6	I/O, PU	No	–
		<p><b>I<sup>2</sup>C Bus Configuration Enable</b></p> <p>Enables or disables the I<sup>2</sup>C/SMBus for initial device configuration prior to initial Link training. Must be pulled or tied High to <b>VDD18</b> or Low to <b>VSS</b> (Ground). The value of this input is latched and reflected in the <b>Strap Configuration</b> register <i>I2C Configuration Enable Status</i> bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset <b>46Ch[5]</b>).</p> <p>L = Enables I<sup>2</sup>C/SMBus for initial device configuration. The serial EEPROM can also be used in this mode. After I<sup>2</sup>C and/or the serial EEPROM Sets the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, offset <b>3ACh[0]</b>), all Ports start Link training.</p> <p>H = Disables I<sup>2</sup>C/SMBus for initial device configuration. The <i>Initiate Configuration</i> bit is Set by hardware, immediately after the PEX 8748 comes out of reset. After the serial EEPROM load finishes, all Ports start Link training (assuming that the serial EEPROM does not Clear the <i>Initiate Configuration</i> bit).</p> <p><i>Notes: The Delayed Linkup option might require software support to delay Host enumeration until I<sup>2</sup>C/SMBus completes switch initialization (by Setting the Initiate Configuration bit; otherwise system software could miss detection of the PCI Express hierarchy (precluding device enumeration). Although this is an I/O signal, its logical operation is input.</i></p>		
STRAP_I2C_SMBUS_EN	F2	I/O, 3-state input	No	–
		<p><b>I<sup>2</sup>C/SMBus Protocol Select Enable</b></p> <p>Selects the I<sup>2</sup>C or SMBus protocol, and defines the default <b>SMBus Configuration</b> register <i>SMBus Enable</i> and <i>ARP Disable</i> bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset <b>2C8h[0]</b> and <b>8]</b>, respectively) values, as listed in the subtable below. The value of this 3-state input is latched and reflected in the <b>Strap Configuration</b> register <i>SMBus Enable Status</i> bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset <b>46Ch[6]</b>).</p>		
		<b>STRAP_I2C_SMBUS_EN Input State</b>	<b>Offset 2C8h[0] Value</b>	<b>Offset 2C8h[8] Value</b>
		0	0	–
		Z, 1	1	1
<p>0 = Enables I<sup>2</sup>C Slave protocol on the <b>I2C_SCL0</b> and <b>I2C_SDA0</b> 2-wire bus</p> <p>Z, 1 = Enables SMBus with ARP, on the <b>I2C_SCL0</b> and <b>I2C_SDA0</b> 2-wire bus</p>				

Table 3-7. Strapping Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States												
		Description														
STRAP_MGMT_PORT_EN	G2	I/O, 3-state input	No	–												
		<b>Virtual Switch Mode Management Port Initialization</b>														
		Enables/Disables the Management Port in Virtual Switch mode. This 3-state input (which can be overridden by the serial EEPROM) is mapped to the following two register bits:														
		<ul style="list-style-type: none"><li>• <b>Management Port Control</b> register <i>Active Management Port Enable</i> bit (Port 0, accessible through the Management Port and Redundant Management Port, offset 354h[5])</li><li>• <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, accessible through the Management Port, offset 3ACh[0])</li></ul>														
		<table><tr><th>STRAP_MGMT_PORT_EN Input State</th><th>Offset 354h[5] Value</th><th>Offset 3ACh[0] Value</th></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>Z</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td></tr></table>			STRAP_MGMT_PORT_EN Input State	Offset 354h[5] Value	Offset 3ACh[0] Value	0	1	1	Z	1	0	1	0	1
		STRAP_MGMT_PORT_EN Input State	Offset 354h[5] Value	Offset 3ACh[0] Value												
		0	1	1												
		Z	1	0												
		1	0	1												
		0 = Management Port is enabled. After the serial EEPROM load finishes, all Ports start Link training (assuming that the serial EEPROM does not Clear the <i>Initiate Configuration</i> bit).														
Z = Management Port is enabled, and only the Management Port links up. After Management Port software, I <sup>2</sup> C/SMBus, and/or the serial EEPROM Sets the <i>Initiate Configuration</i> bit, all other Ports start Link training.																
1 = Management Port is disabled. After the serial EEPROM load finishes, all Ports start Link training.																
If the Management Port is enabled (STRAP_MGMT_PORT_EN=Z or 0):																
<ul style="list-style-type: none"><li>• <b>Option 1</b> – STRAP_MGMT_PORT_EN=Z. After the serial EEPROM (if present) is loaded, the Management Port comes up first, to configure the PEX 8748. When the Management Port has completed its configuration, Management Port software and/or I<sup>2</sup>C/SMBus must Set the <i>Initiate Configuration</i> bit, to release the hold that is preventing the remaining Links from coming up (assuming this hold is not de-asserted by the serial EEPROM Setting the <i>Initiate Configuration</i> bit).</li><li>• <b>Option 2</b> – STRAP_MGMT_PORT_EN=0. After the serial EEPROM (if present) is loaded, all Ports come up concurrently (provided that the serial EEPROM does not Clear the <i>Initiate Configuration</i> bit).</li></ul>																
<b>Notes:</b> Management Port software must <b>not</b> change the <b>Port Configuration</b> register (Port 0, accessible through the Management Port, offset 300h).																
When STRAP_MGMT_PORT_EN=Z, the Delayed Linkup option might require software support to delay Host enumeration until I <sup>2</sup> C/SMBus and/or the Management Port Host completes switch initialization (by Setting the <i>Initiate Configuration</i> bit; otherwise system software could miss detection of the PCI Express hierarchy (precluding device enumeration).																

Table 3-7. Strapping Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States	
		Description			
STRAP_NT_UPSTRM_POR TSEL[2:0]	AA18, G6, A2	I/O, 3-state input	No	–	
		<b>Select Upstream NT Port (3 Balls)</b>			
		Enable and select any Port to be the Upstream NT Port. The input values are sampled and latched upon <b>PEX_PERST#</b> input de-assertion, to define the default values of the <b>VS0 Upstream</b> register <i>NT Enable</i> and <i>NT Port</i> bits (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 360h[13 and 12:8], respectively). A value of 111 disables NT mode; all other values enable NT mode.			
		The STRAP_NT_UPSTRM_PORTSEL[2:0] 3-state input can be overridden by the serial EEPROM value for the <b>VS0 Upstream</b> register.			
		I <sup>2</sup> C/SMBus can enable and select a Port to be the Upstream NT Port, by writing to the <b>VS0 Upstream</b> register. To configure the PEX 8748 through I <sup>2</sup> C/SMBus, the <b>STRAP_I2C_SMBUS_CFG_EN#</b> input can be pulled or tied Low, to delay linkup and Host enumeration until I <sup>2</sup> C initialization is complete. After I <sup>2</sup> C initialization is complete, I <sup>2</sup> C/SMBus must then Set the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 3ACh[0]).			
		Refer to Section 14.8, “Port Programmability,” for further details.			
		<b>STRAP_NT_UPSTRM_PORTSEL[2:0] Input States</b>	<b>Offset 360h</b>	<b>Port Number</b>	
			<b>Field [12:8] Value</b>	<b>Bit 13 Value</b>	
		000	0_0000b	1	0
		00Z	0_0001b	1	1
001	0_0010b	1	2		
0Z0	0_0011b	1	3		
011	0_1000b	1	8		
Z00	0_1001b	1	9		
Z0Z	0_1010b	1	10		
Z01	0_1011b	1	11		
Z1Z	1_0000b	1	16		
Z11	1_0001b	1	17		
100	1_0010b	1	18		
10Z	1_0011b	1	19		
111	1_1111b	0	NT mode is disabled		
All other encodings are <i>Reserved</i> .					

Table 3-7. Strapping Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
STRAP_PLL_BYPASS#	Y26	I, PU	No	–
		<b>Factory Test Only</b> Must be pulled or tied High to <b>VDD18</b> .		
STRAP_PROBE_MODE#	Y21	I, PU	No	–
		<b>Factory Test Only</b> Must be pulled or tied High to <b>VDD18</b> .		
STRAP_RESERVED[16, 4:1]	AE24, F3, W3, N21, N6	I (all but 16) PWR (16)	No	–
		<b>Factory Test Only (5 Balls)</b> <b>Reserved for future use</b> Refer to <a href="#">Table 3-8</a> for specific pull-up/pull-down resistor requirements.		
STRAP_SERDES_MODE_EN#	W26	I, PU	No	–
		<b>Factory Test Only</b> Must be pulled or tied High to <b>VDD18</b> .		
STRAP_STN0_PORT_CFG[1:0]	AF24, AD24	I/O, 3-state input	No	–
		<p><b>Select Port Configuration for Station 0 (Quantity of Enabled Ports (1, 2, 3, or 4), and Maximum Quantity of Lanes for Each Specific Port) (2 Balls)</b></p> <p>Defines the enabled Port Numbers and their Link widths, for Station 0. Programs the <b>Port Configuration</b> register <i>Port Configuration for Station 0</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset <b>300h[2:0]</b>) default value.</p> <p><b>Notes:</b> The default Maximum Payload Size supported is based upon the STRAP_STNx_PORTCFGx 3-state inputs. If the serial EEPROM or I<sup>2</sup>C/SMBus changes the Port configuration (from the strapped value), it must also program the value of the Port's <b>Device Capability</b> register <i>Maximum Payload Size Supported</i> field (offset <b>6Ch[2:0]</b>), accordingly (with the value dependent upon the Station having the most enabled Ports).</p> <p>Refer to <a href="#">Table 4-1</a> for the Port Configurations associated with these values.</p> <p>00 = 0 = <b>Reserved</b>  0Z = 1 = x16  01 = 2 = x8x8  Z0 = 3 = x8x4x4  ZZ = 4 = x4x4x4x4  Z1 = 5 = <b>Reserved</b>  10 = 6 = <b>Reserved</b>  1Z = 7 = <b>Reserved</b>  11 = 8 = <b>Reserved</b></p>		

Table 3-7. Strapping Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
STRAP_STN1_PORT CFG[1:0]	G1, C4	I/O, 3-state input	No	–
		<p><b>Select Port Configuration for Station 1 (Quantity of Enabled Ports (1, 2, 3, or 4), and Maximum Quantity of Lanes for Each Specific Port) (2 Balls)</b></p> <p>Defines the enabled Port Numbers and their Link widths, for Station 1. Programs the <b>Port Configuration</b> register <i>Port Configuration for Station 1</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[5:3]) default value.</p> <p><i>Notes: The default Maximum Payload Size supported is based upon the STRAP_STNx_PORTCFGx 3-state inputs. If the serial EEPROM or I<sup>2</sup>C/SMBus changes the Port configuration (from the strapped value), it must also program the value of the Port's <b>Device Capability</b> register <i>Maximum Payload Size Supported</i> field (offset 6Ch[2:0]), accordingly (with the value dependent upon the Station having the most enabled Ports).</i></p> <p>Refer to <a href="#">Table 4-1</a> for the Port Configurations associated with these values.</p> <p>00 = 0 = <b>Reserved</b>  0Z = 1 = x16  01 = 2 = x8x8  Z0 = 3 = x8x4x4  ZZ = 4 = x4x4x4x4  Z1 = 5 = <b>Reserved</b>  10 = 6 = <b>Reserved</b>  1Z = 7 = <b>Reserved</b>  11 = 8 = <b>Reserved</b></p>		

Table 3-7. Strapping Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
STRAP_STN2_PORT CFG[1:0]	C26, C23	I/O, 3-state input	No	–
		<p><b>Select Port Configuration for Station 2 (Quantity of Enabled Ports (1, 2, 3, or 4), and Maximum Quantity of Lanes for Each Specific Port) (2 Balls)</b></p> <p>Defines the enabled Port Numbers and their Link widths, for Station 2. Programs the <b>Port Configuration</b> register <i>Port Configuration for Station 2</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6]) default value.</p> <p><b>Notes:</b> The default Maximum Payload Size supported is based upon the STRAP_STNx_PORTCFGx 3-state inputs. If the serial EEPROM or I<sup>2</sup>C/SMBus changes the Port configuration (from the strapped value), it must also program the value of the Port's <b>Device Capability</b> register <i>Maximum Payload Size Supported</i> field (offset 6Ch[2:0]), accordingly (with the value dependent upon the Station having the most enabled Ports).</p> <p>Refer to <a href="#">Table 4-1</a> for the Port Configurations associated with these values.</p> <p>00 = 0 = <b>Reserved</b>  0Z = 1 = x16  01 = 2 = x8x8  Z0 = 3 = x8x4x4  ZZ = 4 = x4x4x4x4  Z1 = 5 = <b>Reserved</b>  10 = 6 = <b>Reserved</b>  1Z = 7 = <b>Reserved</b>  11 = 8 = <b>Reserved</b></p>		

Table 3-7. Strapping Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
STRAP_TESTMODE[2:0]	AD4, AD3, Y6	I/O, 3-state input	No	Refer to <a href="#">Section 3.4.4.1</a> for details
		<p><b>Test Mode Selects (3 Balls)</b></p> <p>The STRAP_TESTMODE[2:0] input states (first value listed below) indicate the test mode (second value listed below), which defines the default functionality of the multiplexed I/O signals. These balls can be used to configure Hot Plug, Virtual Switch mode, GPIOx, and/or PORT_GOODx# signal combinations. The value of this 3-state input is latched and reflected in the <b>Strap Configuration</b> register <i>Strap Test Mode Status</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 46Ch[28:24]). (Refer to <a href="#">Section 3.4.4.1</a> for details regarding the test modes, and the multiplexed I/O signals enabled within each mode. Refer also to <a href="#">PORT_GOOD[11:0]#</a>, for details regarding PORT_GOODx# input and output functions, and VS_MODE, for input states that enable Virtual Switch mode.)</p> <p><b>Note:</b> The PORT_GOOD11#/GPIO11 signal exists only in Virtual Switch mode, when four, five, or six virtual switches are enabled.</p> <p>000 = 0 = Hot Plug A + Hot Plug C + PORT_GOOD[10:0]#  00Z = 1 = Hot Plug A + Hot Plug C + SHP_PERST[10:0]#  001 = 2 = Hot Plug A + Hot Plug C + GPIO[10:0] Input  0Z0 = 3 = Hot Plug A + Hot Plug C + Hot Plug B + PORT_GOOD0#  0ZZ = 4 = VS[2:0] + Hot Plug C + PORT_GOOD[10:0]#  0Z1 = 5 = VS[2:0] + Hot Plug C + SHP_PERST[10:0]#  010 = 6 = VS[2:0] + Hot Plug C + GPIO[10:0] Input  01Z = 7 = VS[2:0] + Hot Plug C + Hot Plug B + PORT_GOOD0#  011 = 8 = VS[2:0] + VS[5:3] + PORT_GOOD[11:0]#  Z00 = 9 = VS[2:0] + VS[5:3] + SHP_PERST[10:0]#  Z0Z = 10 = VS[2:0] + VS[5:3] + GPIO[11:0] Input  Z01 = 11 = VS[2:0] + VS[5:3] + Hot Plug B + PORT_GOOD[11, 0]#  ZZ0, ZZZ, ZZ1, Z10, Z1Z, Z11, 100, 10Z,  101, 1Z0, 1ZZ, 1Z1, 110, 11Z, 111 = <b>Factory Test Only</b></p>		



Table 3-7. Strapping Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
STRAP_UPSTRM_PORT SEL[2:0]	G26, G24, G25	I/O, 3-state input	No	—
		<b>Base Mode</b>		
		<b>Select Upstream Port (3 Balls)</b>		
		Select any Port as the VS0 Upstream Port. These inputs map to the <b>VS0 Upstream</b> register <i>VS0 Upstream Port</i> field (Port 0, offset 360h[4:0]).		
		If I <sup>2</sup> C/SMBus is used to configure the PEX 8748, the <b>STRAP_I2C_SMBUS_CFG_EN#</b> input can be pulled or tied Low to <b>VSS</b> (Ground), to delay linkup until I <sup>2</sup> C/SMBus initialization is complete. After I <sup>2</sup> C initialization is complete, I <sup>2</sup> C/SMBus must then Set the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, offset 3ACh[0]).		
		Refer to Section 14.8, “Port Programmability,” for further details.		
		<b>STRAP_UPSTRM_PORTSEL[2:0] Input States</b>	<b>Offset 360h[4:0] Value</b>	<b>Port Number</b>
		000	0_0000b	0
		00Z	0_0001b	1
		001	0_0010b	2
		0Z0	0_0011b	3
		011	0_1000b	8
		Z00	0_1001b	9
		Z0Z	0_1010b	10
		Z01	0_1011b	11
Z1Z	1_0000b	16		
Z11	1_0001b	17		
100	1_0010b	18		
10Z	1_0011b	19		
All other encodings are <i>Reserved</i> .				

Table 3-7. Strapping Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
STRAP_UPSTRM_PORT SEL[2:0]	G26, G24, G25	I/O, 3-state input	No	–
		<p><b>Virtual Switch Mode</b></p> <p><b>Select Management Port (3 Balls)</b></p> <p>If the <a href="#">STRAP_MGMT_PORT_EN</a>=0 or Z, the Management Port is enabled and these inputs select any virtual switch Upstream Port as the Management Port. These inputs map to the <b>Management Port Control</b> register <a href="#">Active Management Port</a> field (Port 0, accessible through the Management Port and Redundant Management Port, offset <a href="#">354h[4:0]</a>).</p> <p>The <b>Management Port Control</b> register also controls the enabling or disabling of the Active Management Port, and designation and enabling of the Redundant Management Port (which is programmed by Management Port software, serial EEPROM, and/or I<sup>2</sup>C). The Active Management Port, the Redundant Management Port, and/or I<sup>2</sup>C/SMBus can promote the Redundant Management Port to be the Active Management Port, by programming the register.</p> <p>If STRAP_MGMT_PORT_EN=0, after a Fundamental Reset and the serial EEPROM load, only the Management Port links up. When the Management Port has completed its configuration of the virtual switches, Management Port software and/or serial EEPROM (and not I<sup>2</sup>C/SMBus) must Set the <b>Configuration Release</b> register <a href="#">Initiate Configuration</a> bit (Port 0, accessible through the Management Port, offset <a href="#">3ACh[0]</a>), to release the hold that is preventing the remaining Links from coming up.</p> <p>If STRAP_MGMT_PORT_EN=Z, all Ports commence Link training after the serial EEPROM load completes, following a Fundamental Reset.</p> <p>If I<sup>2</sup>C is used to configure the PEX 8748, the <a href="#">STRAP_I2C_SMBUS_CFG_EN#</a> input can be pulled or tied Low to VSS (Ground), to delay linkup until I<sup>2</sup>C initialization is complete. After I<sup>2</sup>C initialization is complete, I<sup>2</sup>C and/or the serial EEPROM must then Set the <a href="#">Initiate Configuration</a> bit, to enable all Ports to begin Link training.</p> <p>Refer to the <a href="#">STRAP_UPSTRM_PORTSEL[2:0]</a> Base mode description for the signal and register values associated with each Port Number.</p> <p><b>Note:</b> Because the Virtual Switch Table registers (refer to <a href="#">Section 5.3.3.1, “Virtual Switch Table Registers”</a>) exist in Port 0, Port 0 must not be held in Reset during setup of Virtual Switch mode. Therefore, for whichever virtual switch includes Port 0, its VSx_PERST# input must be de-asserted (High). Assigning Port 0 to be the Management Port might provide the simplest solution toward meeting this requirement.</p>		

Table 3-7. Strapping Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
VS_MODE	W21	I/O, 3-state input	Yes Refer to Section 3.4.4.1	Refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a>
		<b>Virtual Switch Mode Virtual Switch Enable</b> VS_MODE exists only when the <a href="#">STRAP_TESTMODE[2:0]</a> 3-state inputs enable Virtual Switch mode. When used together, the VS_MODE and STRAP_TESTMODE[2:0] 3-state inputs, depending upon their input states, enable up to six virtual switches. The value of this 3-state input is latched and reflected in the <a href="#">Strap Configuration</a> register <i>VS Mode Status</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset <a href="#">46Ch[15:13]</a> ).  When the Test Mode is <a href="#">4</a> , <a href="#">5</a> , <a href="#">6</a> , or <a href="#">7</a> (STRAP_TESTMODE[2:0] input state is 0ZZ, 0Z1, 010, or 01Z, respectively (refer to <a href="#">Table 3-10</a> )), VS_MODE selects one, two, or three virtual switches.  When the Test Mode is <a href="#">8</a> , <a href="#">9</a> , <a href="#">10</a> , or <a href="#">11</a> (STRAP_TESTMODE[2:0] input state is 011, Z00, Z0Z, or Z01, respectively (refer to <a href="#">Table 3-11</a> )), VS_MODE selects 4, 5, or 6 virtual switches.  The quantity of enabled virtual switches is reflected in the <a href="#">Virtual Switch Enable</a> register <i>VSx Enable</i> bits (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset <a href="#">358h[5:0]</a> ).  <a href="#">Table 12-1, “Virtual Switch Default Port Assignments – Virtual Switch Mode,”</a> lists the default Port assignments for each virtual switch.  <i>Note:</i> In Base mode, this signal is the HP_PWR_GOOD_A input.		
		<b>STRAP_TEST MODE[2:0] Input States</b>	<b>VS_MODE Input State</b>	<b># of Enabled Virtual Switches</b>
		0ZZ, 0Z1, 010, or 01Z	0	1
			Z	2
			1	3
		011, Z00, Z0Z, or Z01	0	4
			Z	5
			1	6

Table 3-8. Factory Test Only STRAP\_RESERVED[16, 4:1] Input External Pull-Up/Pull-Down Resistor Requirements

Ball/Signal Name	Location	PU/PD	Pull-Up/Pull-Down Requirement
STRAP_RESERVED16	AE24	–	Must be tied directly to <a href="#">VSS</a> (Ground).
STRAP_RESERVED4	F3	PU	Pull or tie High to <a href="#">VDD18</a> ; can optionally remain unconnected, due to its internal pull-up resistor.
STRAP_RESERVED3	W3	–	Must be pulled or tied High to <a href="#">VDD18</a> .
STRAP_RESERVED2	N21	–	Must be pulled or tied High to <a href="#">VDD18</a> .
STRAP_RESERVED1	N6	–	Must be pulled or tied High to <a href="#">VDD18</a> .

NDA CONFIDENTIAL -- Inventec | King Michael

### 3.4.4.1 STRAP\_TESTMODE[2:0] 3-State Input Configurations

The PEX 8748 has 31 multiplexed I/O signals, listed in [Table 3-9](#), [Table 3-10](#), and [Table 3-11](#). Thirty multiplexed signals are grouped into three sets of 10, with the PORT\_GOOD0# signal being the 31<sup>st</sup> multiplexed signal.

The multiplexed signals include three sets (A, B, and C) of 10 Parallel Hot Plug signals, Virtual Switch signals (three per virtual switch, plus one strapping signal, VS\_MODE), and 10 GPIOx signals that can function as PORT\_GOODx#, PERST# (Reset) outputs (identified as SHP\_PERSTx# in [Table 3-9](#), [Table 3-10](#), and [Table 3-11](#)), or GPIOx input.

Default functionality of each multiplexed signal is determined by the STRAP\_TESTMODE[2:0] input states, which select one of 12 possible user configurations when sampled at Fundamental Reset. The three tables each include four columns (for a total of 12 columns), and each of these columns identifies one of the 12 possible configurations (only one of which can be selected at any time). Individual PORT\_GOODx# and GPIOx signal functionality can be changed by serial EEPROM, I<sup>2</sup>C/SMBus, and/or software, to function as PORT\_GOODx#, GPIOx inputs and/or outputs, and/or Interrupt inputs.

Of the 31 multiplexed signals, none require external connection, with the exception of VS\_MODE, which might require external termination, because the VS\_MODE and STRAP\_TESTMODE[2:0] 3-state inputs define the quantity of virtual switches).

If Hot Plug signals are to be used, the corresponding HP\_MRL\_x# input must be Low, to enable automatic signal sequencing to power up the slot. Because each HP\_MRL\_x# input has an internal pull-up resistor, external pull-down resistors (or connection to the corresponding PRSNT# input from the slot) can hold the HP\_MRL\_x# input Low.

**Notes:** The HP\_Parallel Hot Plug signals are defined in [Table 3-4](#). The VSx and PORT\_GOODx# signals are defined in [Table 3-14](#). STRAP\_TESTMODE[2:0] and VS\_MODE are defined in [Table 3-7](#). Serial Hot Plug PERST# (SHP\_PERSTx#) is defined in [Table 10-7](#).

*Software, serial EEPROM and/or I<sup>2</sup>C/SMBus can change the functionality of the PORT\_GOOD[11:0]#/SHP\_PERST[10:0]#/GPIO[11:0] signals from the default configuration selected by the STRAP\_TESTMODE[2:0] 3-state inputs. If Serial Hot Plug PERST# functionality is used, the STRAP\_TESTMODE[2:0] 3-state inputs should be configured to enable that functionality as default (because the Downstream slots must be held in Reset at power-up).*

*The PORT\_GOOD11#/GPIO11 signal exists only in Virtual Switch mode, when four, five, or six virtual switches are enabled.*

*PORT\_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT\_GOODx# signals can be re-assigned to any Port, however, by programming the **Port Good Selector x** register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) [6A0h](#) through [6A8h](#)).*

*[Table 4-1](#) indicates which Ports are enabled/used, and when, based upon the STRAP\_STNx\_PORTCFGx input states and/or serial EEPROM programming.*

**Table 3-9. STRAP\_TESTMODE[2:0] 3-State Input Configurations – Base Mode, Test Modes 0, 1, 2, 3**

Location	Type	Multiplexed I/O Signals Used, by Test Mode – Base Mode			
		0	1	2	3
		STRAP_TESTMODE[2:0] = 000	STRAP_TESTMODE[2:0] = 00Z	STRAP_TESTMODE[2:0] = 001	STRAP_TESTMODE[2:0] = 0Z0
		Hot Plug A + Hot Plug C + PORT_GOOD[10:0]#	Hot Plug A + Hot Plug C + SHP_PERST[10:0]#	Hot Plug A + Hot Plug C + GPIO[10:0] Input	Hot Plug A + Hot Plug C + Hot Plug B + PORT_GOOD0#
AC26	O, PU	HP_ATNLED_A#	HP_ATNLED_A#	HP_ATNLED_A#	HP_ATNLED_A#
Y24	I, PU	HP_BUTTON_A#	HP_BUTTON_A#	HP_BUTTON_A#	HP_BUTTON_A#
AA24	O, PU	HP_CLKEN_A#	HP_CLKEN_A#	HP_CLKEN_A#	HP_CLKEN_A#
AB26	I, PU	HP_MRL_A#	HP_MRL_A#	HP_MRL_A#	HP_MRL_A#
AB25	O, PU	HP_PERST_A#	HP_PERST_A#	HP_PERST_A#	HP_PERST_A#
AD26	I, PU	HP_PRSNT_A#	HP_PRSNT_A#	HP_PRSNT_A#	HP_PRSNT_A#
W21	I, PD	HP_PWR_GOOD_A	HP_PWR_GOOD_A	HP_PWR_GOOD_A	HP_PWR_GOOD_A
W22	O, PD	HP_PWREN_A	HP_PWREN_A	HP_PWREN_A	HP_PWREN_A
AE26	I, PU	HP_PWRFLT_A#	HP_PWRFLT_A#	HP_PWRFLT_A#	HP_PWRFLT_A#
Y23	O, PU	HP_PWRLED_A#	HP_PWRLED_A#	HP_PWRLED_A#	HP_PWRLED_A#
B25	O, PU	HP_ATNLED_C#	HP_ATNLED_C#	HP_ATNLED_C#	HP_ATNLED_C#
C25	I, PU	HP_BUTTON_C#	HP_BUTTON_C#	HP_BUTTON_C#	HP_BUTTON_C#
F24	O, PU	HP_CLKEN_C#	HP_CLKEN_C#	HP_CLKEN_C#	HP_CLKEN_C#
G21	I, PU	HP_MRL_C#	HP_MRL_C#	HP_MRL_C#	HP_MRL_C#
G22	O, PU	HP_PERST_C#	HP_PERST_C#	HP_PERST_C#	HP_PERST_C#
D25	I, PU	HP_PRSNT_C#	HP_PRSNT_C#	HP_PRSNT_C#	HP_PRSNT_C#
F25	I, PD	HP_PWR_GOOD_C	HP_PWR_GOOD_C	HP_PWR_GOOD_C	HP_PWR_GOOD_C
D24	O, PD	HP_PWREN_C	HP_PWREN_C	HP_PWREN_C	HP_PWREN_C
B26	I, PU	HP_PWRFLT_C#	HP_PWRFLT_C#	HP_PWRFLT_C#	HP_PWRFLT_C#
D26	O, PU	HP_PWRLED_C#	HP_PWRLED_C#	HP_PWRLED_C#	HP_PWRLED_C#
AB1	I/O, PU	PORT_GOOD1#	SHP_PERST1#	GPIO1 input	HP_ATNLED_B#
AB3	I/O, PU	PORT_GOOD2#	SHP_PERST2#	GPIO2 input	HP_BUTTON_B#
AE1	I/O, PU	PORT_GOOD3#	SHP_PERST3#	GPIO3 input	HP_CLKEN_B#
AB2	I/O, PU	PORT_GOOD4#	SHP_PERST4#	GPIO4 input	HP_MRL_B#
AD1	I/O, PU	PORT_GOOD5#	SHP_PERST5#	GPIO5 input	HP_PERST_B#
W5	I/O, PU	PORT_GOOD6#	SHP_PERST6#	GPIO6 input	HP_PRSNT_B#
Y3	I/O, PD	PORT_GOOD7#	SHP_PERST7#	GPIO7 input	HP_PWR_GOOD_B
AC2	I/O, PD	PORT_GOOD8#	SHP_PERST8#	GPIO8 input	HP_PWREN_B
AC3	I/O, PU	PORT_GOOD9#	SHP_PERST9#	GPIO9 input	HP_PWRFLT_B#
AD2	I/O, PU	PORT_GOOD10#	SHP_PERST10#	GPIO10 input	HP_PWRLED_B#
D3	I/O, PU	PORT_GOOD0#	SHP_PERST0#	GPIO0 input	PORT_GOOD0#

NDA CONFIDENTIAL -- Inventec | King Michael

**Table 3-10. STRAP\_TESTMODE[2:0] 3-State Input Configurations – Virtual Switch Mode, Test Modes 4, 5, 6, 7**

Location	Type	Multiplexed I/O Signals Used, by Test Mode – Virtual Switch Mode; One to Three Virtual Switches, as Selected by VS_MODE Input State			
		4	5	6	7
		STRAP_TESTMODE[2:0] = 0ZZ	STRAP_TESTMODE[2:0] = 0Z1	STRAP_TESTMODE[2:0] = 010	STRAP_TESTMODE[2:0] = 01Z
		VS[2:0] + Hot Plug C + PORT_GOOD[10:0]#	VS[2:0] + Hot Plug C + SHP_PERST[10:0]#	VS[2:0] + Hot Plug C + GPIO[10:0] Input	VS[2:0] + Hot Plug C + Hot Plug B + PORT_GOOD0#
AC26	O, PU	VS0_PEX_INTA#	VS0_PEX_INTA#	VS0_PEX_INTA#	VS0_PEX_INTA#
Y24	I, PU	VS0_PERST#	VS0_PERST#	VS0_PERST#	VS0_PERST#
AA24	O, PU	VS1_PEX_INTA#	VS1_PEX_INTA#	VS1_PEX_INTA#	VS1_PEX_INTA#
AB26	I, PU	VS1_PERST#	VS1_PERST#	VS1_PERST#	VS1_PERST#
AB25	O, PU	VS2_PEX_INTA#	VS2_PEX_INTA#	VS2_PEX_INTA#	VS2_PEX_INTA#
AD26	I, PU	VS2_PERST#	VS2_PERST#	VS2_PERST#	VS2_PERST#
W21	I, 3-state	VS_MODE	VS_MODE	VS_MODE	VS_MODE
W22	O, PD	VS2_FATAL_ERR#	VS2_FATAL_ERR#	VS2_FATAL_ERR#	VS2_FATAL_ERR#
AE26	I, PU	VS1_FATAL_ERR#	VS1_FATAL_ERR#	VS1_FATAL_ERR#	VS1_FATAL_ERR#
Y23	O, PU	VS0_FATAL_ERR#	VS0_FATAL_ERR#	VS0_FATAL_ERR#	VS0_FATAL_ERR#
B25	O, PU	HP_ATNLED_C#	HP_ATNLED_C#	HP_ATNLED_C#	HP_ATNLED_C#
C25	I, PU	HP_BUTTON_C#	HP_BUTTON_C#	HP_BUTTON_C#	HP_BUTTON_C#
F24	O, PU	HP_CLKEN_C#	HP_CLKEN_C#	HP_CLKEN_C#	HP_CLKEN_C#
G21	I, PU	HP_MRL_C#	HP_MRL_C#	HP_MRL_C#	HP_MRL_C#
G22	O, PU	HP_PERST_C#	HP_PERST_C#	HP_PERST_C#	HP_PERST_C#
D25	I, PU	HP_PRSNT_C#	HP_PRSNT_C#	HP_PRSNT_C#	HP_PRSNT_C#
F25	I/O, PD	HP_PWR_GOOD_C	HP_PWR_GOOD_C	HP_PWR_GOOD_C	HP_PWR_GOOD_C
D24	O, PD	HP_PWREN_C	HP_PWREN_C	HP_PWREN_C	HP_PWREN_C
B26	I, PU	HP_PWRFLT_C#	HP_PWRFLT_C#	HP_PWRFLT_C#	HP_PWRFLT_C#
D26	O, PU	HP_PWRLED_C#	HP_PWRLED_C#	HP_PWRLED_C#	HP_PWRLED_C#
AB1	I/O, PU	PORT_GOOD1#	SHP_PERST1#	GPIO1 input	HP_ATNLED_B#
AB3	I/O, PU	PORT_GOOD2#	SHP_PERST2#	GPIO2 input	HP_BUTTON_B#
AE1	I/O, PU	PORT_GOOD3#	SHP_PERST3#	GPIO3 input	HP_CLKEN_B#
AB2	I/O, PU	PORT_GOOD4#	SHP_PERST4#	GPIO4 input	HP_MRL_B#
AD1	I/O, PU	PORT_GOOD5#	SHP_PERST5#	GPIO5 input	HP_PERST_B#
W5	I/O, PU	PORT_GOOD6#	SHP_PERST6#	GPIO6 input	HP_PRSNT_B#
Y3	I/O, PD	PORT_GOOD7#	SHP_PERST7#	GPIO7 input	HP_PWR_GOOD_B
AC2	I/O, PD	PORT_GOOD8#	SHP_PERST8#	GPIO8 input	HP_PWREN_B
AC3	I/O, PU	PORT_GOOD9#	SHP_PERST9#	GPIO9 input	HP_PWRFLT_B#
AD2	I/O, PU	PORT_GOOD10#	SHP_PERST10#	GPIO10 input	HP_PWRLED_B#
D3	I/O, PU	PORT_GOOD0#	SHP_PERST0#	GPIO0 input	PORT_GOOD0#

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

**Table 3-11. STRAP\_TESTMODE[2:0] 3-State Input Configurations – Virtual Switch Mode, Test Modes 8, 9, 10, 11**

Location	Type	Multiplexed I/O Signals Used, by Test Mode – Virtual Switch Mode; Four to Six Virtual Switches, as Selected by VS_MODE Input State			
		8	9	10	11
		STRAP_TESTMODE[2:0] = 011	STRAP_TESTMODE[2:0] = Z00	STRAP_TESTMODE[2:0] = Z0Z	STRAP_TESTMODE[2:0] = Z01
		VS[2:0] + VS[5:3] + PORT_GOOD[11:0]#	VS[2:0] + VS[5:3] + SHP_PERST[10:0]#	VS[2:0] + VS[5:3] + GPIO[11:0] Input	VS[2:0] + VS[5:3] + Hot Plug B + PORT_GOOD[11, 0]#
AC26	O, PU	VS0_PEX_INTA#	VS0_PEX_INTA#	VS0_PEX_INTA#	VS0_PEX_INTA#
Y24	I, PU	VS0_PERST#	VS0_PERST#	VS0_PERST#	VS0_PERST#
AA24	O, PU	VS1_PEX_INTA#	VS1_PEX_INTA#	VS1_PEX_INTA#	VS1_PEX_INTA#
AB26	I, PU	VS1_PERST#	VS1_PERST#	VS1_PERST#	VS1_PERST#
AB25	O, PU	VS2_PEX_INTA#	VS2_PEX_INTA#	VS2_PEX_INTA#	VS2_PEX_INTA#
AD26	I, PU	VS2_PERST#	VS2_PERST#	VS2_PERST#	VS2_PERST#
W21	I, 3-state	VS_MODE	VS_MODE	VS_MODE	VS_MODE
W22	O, PD	VS2_FATAL_ERR#	VS2_FATAL_ERR#	VS2_FATAL_ERR#	VS2_FATAL_ERR#
AE26	I, PU	VS1_FATAL_ERR#	VS1_FATAL_ERR#	VS1_FATAL_ERR#	VS1_FATAL_ERR#
Y23	O, PU	VS0_FATAL_ERR#	VS0_FATAL_ERR#	VS0_FATAL_ERR#	VS0_FATAL_ERR#
B25	O, PU	VS3_PEX_INTA#	VS3_PEX_INTA#	VS3_PEX_INTA#	VS3_PEX_INTA#
C25	I, PU	VS3_PERST#	VS3_PERST#	VS3_PERST#	VS3_PERST#
F24	O, PU	VS4_PEX_INTA#	VS4_PEX_INTA#	VS4_PEX_INTA#	VS4_PEX_INTA#
G21	I, PU	VS4_PERST#	VS4_PERST#	VS4_PERST#	VS4_PERST#
G22	O, PU	VS5_PEX_INTA#	VS5_PEX_INTA#	VS5_PEX_INTA#	VS5_PEX_INTA#
D25	I, PU	VS5_PERST#	VS5_PERST#	VS5_PERST#	VS5_PERST#
F25	I/O, PD	PORT_GOOD11#	PERST11#	GPIO11 input	PORT_GOOD11#
D24	O, PD	VS5_FATAL_ERR#	VS5_FATAL_ERR#	VS5_FATAL_ERR#	VS5_FATAL_ERR#
B26	I, PU	VS4_FATAL_ERR#	VS4_FATAL_ERR#	VS4_FATAL_ERR#	VS4_FATAL_ERR#
D26	O, PU	VS3_FATAL_ERR#	VS3_FATAL_ERR#	VS3_FATAL_ERR#	VS3_FATAL_ERR#
AB1	I/O, PU	PORT_GOOD1#	SHP_PERST1#	GPIO1 input	HP_ATNLED_B#
AB3	I/O, PU	PORT_GOOD2#	SHP_PERST2#	GPIO2 input	HP_BUTTON_B#
AE1	I/O, PU	PORT_GOOD3#	SHP_PERST3#	GPIO3 input	HP_CLKEN_B#
AB2	I/O, PU	PORT_GOOD4#	SHP_PERST4#	GPIO4 input	HP_MRL_B#
AD1	I/O, PU	PORT_GOOD5#	SHP_PERST5#	GPIO5 input	HP_PERST_B#
W5	I/O, PU	PORT_GOOD6#	SHP_PERST6#	GPIO6 input	HP_PRSNT_B#
Y3	I/O, PD	PORT_GOOD7#	SHP_PERST7#	GPIO7 input	HP_PWR_GOOD_B
AC2	I/O, PD	PORT_GOOD8#	SHP_PERST8#	GPIO8 input	HP_PWREN_B
AC3	I/O, PU	PORT_GOOD9#	SHP_PERST9#	GPIO9 input	HP_PWRFLT_B#
AD2	I/O, PU	PORT_GOOD10#	SHP_PERST10#	GPIO10 input	HP_PWRLED_B#
D3	I/O, PU	PORT_GOOD0#	SHP_PERST0#	GPIO0 input	PORT_GOOD0#

NDA CONFIDENTIAL -- Inventec | King Michael

### 3.4.5 JTAG Interface Signals

The PEX 8748 includes five signals for performing Joint Test Action Group (JTAG) boundary scan, defined in [Table 3-12](#). The JTAG interface is described in [Section 17.6, “JTAG Interface.”](#)

**Table 3-12. JTAG Interface Signals**

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
JTAG_TCK	B2	I, PD	No	–
		<b>JTAG Test Clock Input</b> JTAG Test Access Port (TAP) Controller clock source. Frequency can be from 0 to 15 MHz.		
JTAG_TDI	C3	I, PD	No	–
		<b>JTAG Test Data Input</b> Serial input to the JTAG TAP Controller, for test instructions and data.		
JTAG_TDO	G5	I/O, TS	No	–
		<b>JTAG Test Data Output</b> Serial output from the JTAG TAP Controller test instructions and data.  <i><b>Note:</b> Although this is an I/O signal with tri-statable output, its logical operation is output that is driven when the TAP Controller is being used; otherwise, the output is High-Impedance.</i>		
JTAG_TMS	B1	I, PD	No	–
		<b>JTAG Test Mode Select</b> Input decoded by the JTAG TAP Controller, to control test operations.		
JTAG_TRST#	C2	I, PD	No	–
		<b>JTAG Test Reset</b> Active-Low input used to reset the Test Access Port. When JTAG functionality is not used, the JTAG_TRST# input should be driven Low, or pulled Low to <a href="#">VSS</a> (Ground) through a 1.5K $\Omega$ resistor, to place the JTAG TAP Controller into the <i>Test-Logic-Reset</i> state, which disables the test logic and enables standard logic operation. Alternatively, if JTAG_TRST# input is High, the JTAG TAP Controller can be placed into the <i>Test-Logic-Reset</i> state by initializing the JTAG TAP Controller's <b>Instruction</b> register to contain the <i>IDCODE</i> instruction, or by holding the JTAG_TMS input High for at least five rising edges of the JTAG_TCK input.		



### 3.4.6 I<sup>2</sup>C/SMBus Slave Interface Signals

Table 3-13 defines the three I<sup>2</sup>C/SMBus Slave interface signals. The I<sup>2</sup>C/SMBus Slave interface provides access to the PEX 8748 registers. Most registers that can be programmed by serial EEPROM (including registers that are Read Only to PCI Express), can be also programmed by I<sup>2</sup>C/SMBus. However, I<sup>2</sup>C/SMBus cannot replace the serial EEPROM programming of some registers, *such as* SerDes- and Hot Plug-related registers that (generally) must be configured prior to automatic link training, immediately following Reset. I<sup>2</sup>C/SMBus can also be used to program the serial EEPROM.

The I<sup>2</sup>C/SMBus Slave interface used with PLX software is a powerful debugging tool that enables register access, regardless of whether the PCI Express Link is Up. Figure 3-1 illustrates the suggested Header pin layout on 0.1-inch centers, for direct connection to the Total Phase Aardvark I<sup>2</sup>C-USB converter (as used with PLX Rapid Development Kits (RDKs)).

For further details, refer to Chapter 7, “I<sup>2</sup>C/SMBus Slave Interface Operation.”

Figure 3-1. Suggested I<sup>2</sup>C Header Pin Layout on 0.1-Inch Centers

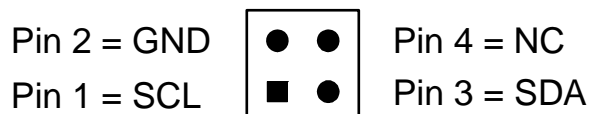


Table 3-13. I<sup>2</sup>C/SMBus Slave Interface Signals

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States								
		Description										
I2C_ADDR0	E2	I/O, 3-state input	No	–								
		<b>I<sup>2</sup>C Slave/SMBus Device Address Bit</b>  Used to define the two least significant bits of the PEX 8748 I <sup>2</sup> C/SMBus 7-bit Slave Address, which is reflected in both the <b>I<sup>2</sup>C Configuration</b> register <i>Slave Address</i> and <b>SMBus Configuration</b> register <i>SMBus Device Address</i> fields (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offsets <b>294h[6:0]</b> and <b>2C8h[7:1]</b> , respectively). If I <sup>2</sup> C or SMBus configuration is used, the I2C_ADDR0 3-state input should be strapped to a unique address, to avoid an address conflict with any other I <sup>2</sup> C/SMBus devices (on the same I <sup>2</sup> C Bus/ SMBus segment).  The three possible default I <sup>2</sup> C Slave/SMBus Device Address values are listed below.										
		<table> <tr> <th>I2C_ADDR0 Input State</th> <th>I<sup>2</sup>C Slave/SMBus Device Address</th> </tr> <tr> <td>0</td> <td>0111_000b (default)</td> </tr> <tr> <td>Z</td> <td>0111_001b</td> </tr> <tr> <td>1</td> <td>0111_010b</td> </tr> </table>			I2C_ADDR0 Input State	I <sup>2</sup> C Slave/SMBus Device Address	0	0111_000b (default)	Z	0111_001b	1	0111_010b
		I2C_ADDR0 Input State	I <sup>2</sup> C Slave/SMBus Device Address									
		0	0111_000b (default)									
Z	0111_001b											
1	0111_010b											

Table 3-13. I<sup>2</sup>C/SMBus Slave Interface Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
I2C_SCL0	G4	OD	No	—
		<b>I<sup>2</sup>C/SMBus Serial Clock Line</b> I <sup>2</sup> C/SMBus bidirectional Clock line. Data on the I <sup>2</sup> C Bus can be transferred at rates of up to 100 kbit/s (Standard mode). Because the 1.8V signaling is less than the voltage defined by the I <sup>2</sup> C Bus v2.1 (5V, 3.3V), external voltage translation circuitry might be required. <b>I2C_SCL0 requires an external pull-up resistor.</b> <i>Note: The PEX 8748 I<sup>2</sup>C/SMBus Slave Interface can stretch the Low period of the I<sup>2</sup>C/SMBus clock while a simultaneous in-band Request that also targets PEX 8748 registers is being processed.</i>		
I2C_SDA0	E3	OD	No	—
		<b>I<sup>2</sup>C/SMBus Serial Data Line</b> Transmits and receives I <sup>2</sup> C/SMBus data during I <sup>2</sup> C/SMBus accesses to PEX 8748 registers. Because the 1.8V signaling is less than the voltage defined by the I <sup>2</sup> C Bus v2.1 (5V, 3.3V), external voltage translation circuitry might be required. <b>I2C_SDA0 requires an external pull-up resistor.</b>		

### 3.4.7 Device-Specific Signals

Table 3-14 defines the Device-Specific signals – signals that are unique to the PEX 8748.

**Table 3-14. Device-Specific Signals**

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
FATAL_ERR#	C24	I/O, TS	No	–
		<p><b>Base Mode Fatal Error Output</b></p> <p>FATAL_ERR# is asserted Low when a Fatal error is detected in the PEX 8748 and the following conditions are met (all the same conditions that are required to send a Fatal Error Message to the Host).</p> <p>For <i>PCI Express Base r3.0</i> errors:</p> <ul style="list-style-type: none"> <li>Specific error is defined as <i>Fatal</i> in the <b>Uncorrectable Error Severity</b> register (offset <b>FC0h</b>), and</li> <li>Reporting of the specific error condition is enabled, not masked by the <b>Uncorrectable Error Mask</b> register's (offset <b>FBCh</b>) corresponding <i>Interrupt Mask</i> bit, and</li> <li><b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset <b>70h[2]</b>) or <b>PCI Command</b> register <i>SERR# Enable</i> bit (offset <b>04h[8]</b>) is Set</li> </ul> <p>Device-Specific (RAM ECC) errors can be reported as a PCI Express Uncorrectable Error, and/or by interrupt (INTx, MSI, or <b>PEX_INTA#</b>) signaling, by the FATAL_ERR# output:</p> <ul style="list-style-type: none"> <li>Reporting of the specific error condition in the <b>Device-Specific Error Status x</b> register bit(s), if not masked in their corresponding <b>Device-Specific Error Mask x</b> register bit(s) (Port 0, 8, or 16, offsets <b>700h</b>, <b>708h</b>, <b>710h</b>, and/or <b>718h</b> (Status) and offsets <b>704h</b>, <b>70Ch</b>, <b>714h</b>, and/or <b>71Ch</b> (Mask)).</li> <li>Uncorrectable Internal error in the <b>Uncorrectable Error Status</b> register <i>Uncorrectable Internal Error Status</i> bit (Port 0, 8, or 16, offset <b>FB8h[22]</b>) is defined as <i>Fatal</i> in the <b>Uncorrectable Error Severity</b> register <i>Uncorrectable Internal Error Severity</i> bit (Port 0, 8, or 16, offset <b>FC0h[22]</b>), and</li> <li>Reporting of Uncorrectable Internal errors is enabled (not masked) by the <b>Uncorrectable Error Mask</b> register <i>Uncorrectable Internal Error Mask</i> bit (Port 0, 8, or 16, offset <b>FBCh[22]</b>), and</li> <li><b>Device Control</b> register <i>Fatal Error Reporting Enable</i> bit (offset <b>70h[2]</b>) or <b>PCI Command</b> register <i>SERR# Enable</i> bit (offset <b>04h[8]</b>) is Set</li> </ul> <p>The <b>Device Status</b> register <i>Fatal Error Detected</i> bit (offset <b>70h[18]</b>) is Set, and the specific error is flagged in the <b>Uncorrectable Error Status</b> register (offset <b>FB8h</b>).</p> <p style="text-align: right;">Continued...</p>		

Table 3-14. Device-Specific Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
FATAL_ERR#	C24	I/O, TS	No	—
		<p>Continued...</p> <p>Software can be used to de-assert FATAL_ERR#, with the following procedure:</p> <ol style="list-style-type: none"> <li>1. Set the Port's <b>ECC Error Check Disable</b> register <i>Software Force Error Enable</i> bit (Transparent Downstream Ports, offset 720h[2]).</li> <li>2. Write 5h to bits [11:8] of the <b>Uncorrectable Error Status</b> register (offset FB8h), in addition to Clearing <i>Status</i> bits that are Set, using a Read-Modify-Write operation. When the Port's <i>Software Force Error Enable</i> bit is Set, writing 1 to an <i>Error Status</i> bit that is Cleared causes that bit to be Set, and generates the corresponding ERR_FATAL or ERR_NONFATAL Message, if enabled and not masked. Therefore, software should not, <i>for example</i>, write all 1s to this register while the Port's <i>Software Force Error Enable</i> bit is Set.</li> <li>3. Clear the Port's <i>Software Force Error Enable</i> bit.</li> </ol> <p><b>If FATAL_ERR# output is not used, this signal can remain unconnected.</b></p> <p><i>Notes:</i> Although this is an I/O signal with tri-statable output, its logical operation is output that is always driven.</p> <p>Device-Specific errors (RAM ECC errors) are reported in the lowest numbered enabled Port within the Station.</p>		

Table 3-14. Device-Specific Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
PEX_INTA#	D1	OD	No	—
		<p><b>Base Mode Interrupt Output</b></p> <p>PEX_INTA# Interrupt output is enabled if:</p> <ul style="list-style-type: none"> <li>INTx Messages are enabled (Port's <b>PCI Command</b> register <i>Interrupt Disable</i> bit, offset 04h[10], is Cleared), and MSIs are disabled (Port's <b>MSI Control</b> register <i>MSI Enable</i> bit, offset 48h[16], is Cleared)</li> <li>PEX_INTA# output is enabled (<b>ECC Error Check Disable</b> register <i>Enable PEX_INTA# Interrupt Output(s) for x Interrupt</i> bit(s) (offset 720h[7, 6, 5, and/or 4]; refer to the register description, for Port associations) are Set)</li> </ul> <p>The three interrupt mechanisms, listed below, are mutually exclusive modes of operation, on a per-Port basis, for all interrupt sources:</p> <ul style="list-style-type: none"> <li>Conventional PCI INTx Message generation</li> <li>Native MSI transaction generation</li> <li>Device-Specific PEX_INTA# assertion</li> </ul> <p>PEX_INTA# assertion (Low) indicates that one or more of the following events and/or errors (if not masked) were detected:</p> <ul style="list-style-type: none"> <li>Hot Plug or Link State events</li> <li>PCI Express Hot Plug events</li> <li>General-Purpose Input Interrupt events</li> <li>Device-Specific NT-Link Port errors and events</li> <li>NT-Virtual Doorbell events</li> <li>NT-Link Doorbell events</li> </ul> <p>Refer to Section 9.1.1, "Interrupt Sources or Events," for details.</p> <p><b>If used, PEX_INTA# output requires an external pull-up resistor; otherwise, this output can remain unconnected.</b></p>		
PEX_NT_PERST#	AD25	I, PU	No	—
		<p><b>Active-Low Input Used to Reset the NT Port Link Interface in NT Mode</b></p> <p><b>If PEX_NT_PERST# input is not used, the signal can either be pulled High, or left unconnected because the internal pull-up resistor is sufficiently strong, to hold PEX_NT_PERST# High.</b></p>		

Table 3-14. Device-Specific Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
PORT_GOOD[11:0]#	F25, AD2, AC3, AC2, Y3, W5, AD1, AB2, AE1, AB3, AB1, D3	PORT_GOOD[8:7]# – I/O, PD, PORT_GOOD[11:9, 6:0]# – I/O, PU	Yes Refer to Section 3.4.4.1	Refer to <a href="#">Section 3.4.4.1</a> for details
		<p><b>Active-Low PCI Express Port Linkup Status Indicator Outputs –or– Programmable General-Purpose I/O (12 Balls)</b></p> <p>PORT_GOODx# signals can function as PORT_GOODx# outputs, Serial Hot Plug Reset outputs (SHP_PERSTx#), or General-Purpose input/output (I/O). Default functionality for all PORT_GOODx# signals is selected according to the STRAP_TESTMODE[2:0] input states (sampled at Fundamental Reset (PEX_PERST#) de-assertion), and the functionality of each signal can be programmed by the serial EEPROM, I<sup>2</sup>C/SMBus, and/or software. If Serial Hot Plug functionality is required, the STRAP_TESTMODE[2:0] inputs should select that as the default functionality (STRAP_TESTMODE[2:0] = 00Z, 0Z1 or Z00).</p> <p>The General-Purpose Input/Output (GPIO) signals can function independently, as General Purpose inputs (default functionality), General-Purpose outputs, or Interrupt inputs (that can trigger interrupt outputs using MSI Writes, INTx Messages, PEX_INTA#, or VSx_PEX_INTA# signaling).</p> <p><i>Notes: Table 4-1 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</i></p> <p><i>The PORT_GOOD11#/GPIO11 signal exists only in Virtual Switch mode, when four, five, or six virtual switches are enabled.</i></p> <p><i>PORT_GOOD[11:0]# signals that are enabled to function as PORT_GOODx# outputs correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default.</i></p> <p><i>PORT_GOODx# signals that are enabled to function as PORT_GOODx# outputs can be re-assigned to any Port, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) 6A0h through 6A8h). However, if Serial Hot Plug Reset functionality (SHP_PERSTx#) functionality is enabled, the PORT_GOODx# signals <b>cannot</b> be re-assigned to a different Port. Instead, the PORT_GOODx# signals are assigned to Serial Hot Plug Ports, in increasing order, starting with PORT_GOOD0#:</i></p> <ul style="list-style-type: none"> <li>• PORT_GOOD0# is assigned to the first Serial Hot Plug Port (which is selected by the PORTID[4:0] inputs on the I/O Expander mapped to I<sup>2</sup>C/SMBus address 40h).</li> <li>• PORT_GOOD1# is assigned to the second Serial Hot Plug Port. If a 40-bit I/O Expander is used (for two Serial Hot Plug Ports), this Port assignment is determined by the value of the second set of PORTID inputs (PORTID1[4:0]). If a 16-bit I/O Expander is used, the Port is selected by the PORTID[4:0] inputs on the I/O Expander mapped to I<sup>2</sup>C/SMBus address 41h.</li> <li>• Port assignment continues, as described above, with each sequential I/O Expander I<sup>2</sup>C/SMBus address.</li> </ul> <p><i>PORT_GOODx# signals that are enabled to function as GPIO are always mapped to Port 0 registers in Base mode. In Virtual Switch mode, these GPIO signals can be re-assigned to VS Upstream Ports (in the Port 0 registers located at offsets 64Ch through 664h).</i></p> <p style="text-align: right;">Continued...</p>		

Table 3-14. Device-Specific Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
PORT_GOOD[11:0]#	F25, AD2, AC3, AC2, Y3, W5, AD1, AB2, AE1, AB3, AB1, D3	PORT_GOOD[8:7]# – I/O, PD, PORT_GOOD[11:9, 6:0]# – I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to Section 3.4.4.1 for details
		<p>Continued...</p> <p><b>PORT_GOOD Output Function</b></p> <p>For PORT_GOOD<math>x</math># signals that are assigned to enabled Ports (according to the <b>Port Good Selector <math>x</math></b> register(s), Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) <a href="#">6A0h</a> through <a href="#">6A8h</a>) and are configured for PORT_GOOD output functionality (by the STRAP_TESTMODE[2:0] input states, sampled at Fundamental Reset; refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a> for details), or by subsequent programming (by serial EEPROM, I<sup>2</sup>C, and/or software) of the appropriate <b>GPIO 0<math>_x</math> Direction Control</b> register <i>Direction Control</i> bit(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset(s) <a href="#">600h</a> and/or <a href="#">604h</a>) value(s). The output states are not directly available from a single register (due to encoded, possibly blinking output); however, software can determine LANE_GOOD status (Physical Layer Link status for each Lane) from the <b>Station <math>x</math> Lane Status</b> register(s) <i>Lane <math>x</math> Up Status</i> bits (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset(s) <a href="#">330h</a>[31:0] and <a href="#">334h</a>[15:0]). Software can also determine the Port's <b>Maximum Link Width</b> and <b>Maximum Link Speed</b>, from the Port's <b>Link Capability</b> register (offset <a href="#">74h</a>[9:4 and 3:0], respectively), as well as the Port's <b>Negotiated Link Width</b> and <b>Current Link Speed</b>, from the Port's <b>Link Status</b> register (offset <a href="#">78h</a>[25:20 and 19:16], respectively).</p> <p>If PORT_GOOD output functionality is enabled, but some Ports are not enabled due to the STRAP_STN<math>_x</math>_PORTCFG<math>_x</math> input states, the PORT_GOOD<math>x</math># signals associated with non-enabled Ports function as GPIO<math>_x</math> signals.</p> <p>LED behavior when connected to PORT_GOOD<math>x</math># signals:</p> <ul style="list-style-type: none"> <li>• <b>Off</b> – Link is Down</li> <li>• <b>Blinking, 512 ms On, 512 ms Off (1 Hz)</b> – Link is Up, 2.5 GT/s, any negotiated width</li> <li>• <b>Blinking, 256 ms On, 256 ms Off (2 Hz)</b> – Link is Up, 5.0 GT/s, any negotiated width</li> <li>• <b>On</b> – Link is Up, 8.0 GT/s, any negotiated width</li> </ul> <p>The PORT_GOOD<math>x</math>#/GPIO<math>_x</math> I/Os operate from the <a href="#">VDD18</a> supply, and maximum input voltage is 2.5V. Because LED forward voltage drop is typically greater than 1.8V, the LED voltage source will likely be higher than 2.5V. In this case, if the LED is connected directly to the PORT_GOOD<math>x</math># input, the input voltage level (while the LED is turned Off) would exceed the specification. Therefore, a PORT_GOOD<math>x</math># signal typically cannot drive an LED directly; however, it can be used to gate a transistor that switches the LED On and Off.</p> <p>For further details regarding LED behavior, refer to <a href="#">Section 17.7, “Port Good Status LEDs.”</a></p> <p>Continued...</p>		

**Table 3-14. Device-Specific Signals (Cont.)**

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
PORT_GOOD[11:0]#	F25, AD2, AC3, AC2, Y3, W5, AD1, AB2, AE1, AB3, AB1, D3	PORT_GOOD[8:7]# – I/O, PD, PORT_GOOD[11:9, 6:0]# – I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to Section 3.4.4.1 for details
		<p>Continued...</p> <p><b>Serial Hot Plug Reset</b></p> <p>The I<sup>2</sup>C I/O Expander provides a serialized Reset output; however, a parallel output is recommended because:</p> <ul style="list-style-type: none"> <li>I<sup>2</sup>C/SMBus is relatively slow. In the case of a power fault (<a href="#">PWRFLT#</a>=0), software should assert Reset to the Downstream slot, as quickly as possible.</li> <li>If <a href="#">PEX_PERST#</a> input to the PEX 8748 is asserted, Downstream devices should also be reset; however, the PEX 8748 I<sup>2</sup>C/SMBus Controller is in a Reset state, and cannot generate the I<sup>2</sup>C/SMBus command to assert the I/O Expander Reset output.</li> </ul> <p>Reset to the Downstream slot can be implemented with the output from a 2-input AND gate, that has PORT_GOOD<sub>x</sub># (SHP_PERST<sub>x</sub>#) and PEX_PERST# as inputs.</p> <p>Continued...</p>		



Table 3-14. Device-Specific Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
PORT_GOOD[11:0]#	F25, AD2, AC3, AC2, Y3, W5, AD1, AB2, AE1, AB3, AB1, D3	PORT_GOOD[8:7]# – I/O, PD, PORT_GOOD[11:9, 6:0]# – I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to Section 3.4.4.1 for details
		<p>Continued...</p> <p><b>General-Purpose Inputs</b></p> <p>For PORT_GOODx# signals that are configured as General-Purpose inputs (by the STRAP_TESTMODE[2:0] input states, sampled at Fundamental Reset; refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a> for details), or by subsequent programming (by serial EEPROM, I<sup>2</sup>C, and/or software) of the appropriate <b>GPIO 0_x Direction Control</b> register <i>Direction Control</i> bit(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset(s) <a href="#">600h</a> and/or <a href="#">604h</a>) value(s), input states are reflected in the <b>GPIO 0_11 Input Data</b> register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset <a href="#">61Ch</a>).</p> <p>Inputs can be internally de-bounced, by Setting the corresponding <b>GPIO 0_11 Input De-Bounce</b> register bits (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset <a href="#">614h</a>). De-bouncing is disabled, by default; if de-bouncing is enabled, an input must be stable for approximately 1.3 ms to be latched.</p> <p><b>General-Purpose Outputs</b></p> <p>For PORT_GOODx# signals that are configured as General-Purpose outputs in the <b>GPIO 0_x Direction Control</b> register <i>Direction Control</i> bit(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset(s) <a href="#">600h</a> and/or <a href="#">604h</a>), output states are controlled by the corresponding <b>GPIO 0_11 Output Data</b> register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset <a href="#">624h</a>) bit values.</p> <p><b>Interrupt Inputs</b></p> <p>For PORT_GOODx# signals that are configured as Interrupt inputs in the <b>GPIO 0_x Direction Control</b> register <i>Direction Control</i> bit(s), input states are reflected in the <b>GPIO 0_11 Interrupt Status</b> register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset <a href="#">634h</a>).</p> <p>Inputs can be internally de-bounced, by Setting the corresponding <b>GPIO 0_11 Input De-Bounce</b> register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset <a href="#">614h</a>) bits. De-bouncing is disabled, by default; if de-bouncing is enabled, an input must be stable for approximately 1.3 ms to be latched.</p> <p>Interrupt polarity (Active-High or Active-Low) is individually programmable in the <b>GPIO 0_11 Interrupt Polarity</b> register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset <a href="#">62Ch</a>).</p> <p>Interrupt generation can be selectively masked in the <b>GPIO 0_11 Interrupt Mask</b> register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset <a href="#">63Ch</a>). If a <b>GPIO 0_11 Interrupt Mask</b> register bit is Set, a read of the corresponding <b>GPIO 0_11 Interrupt Status</b> register <i>Interrupt Status</i> bit returns a value of 0.</p> <p style="text-align: right;">Continued...</p>		

Table 3-14. Device-Specific Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
PORT_GOOD[11:0]#	F25, AD2, AC3, AC2, Y3, W5, AD1, AB2, AE1, AB3, AB1, D3	PORT_GOOD[8:7]# – I/O, PD, PORT_GOOD[11:9, 6:0]# – I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to Section 3.4.4.1 for details
		<p>Continued...</p> <p><b>GPIO Function – Virtual Switch Mode</b></p> <p>In Virtual Switch mode, PORT_GOODx# signals that are enabled for GPIO functionality are associated to specific VS Upstream Ports, by one of two mechanisms:</p> <ol style="list-style-type: none"> <li>1. If <a href="#">STRAP_MGMT_PORT_EN</a>=Z, to enable the Management Port and initially Clear the <a href="#">Configuration Release</a> register <i>Initiate Configuration</i> bit (Port 0, accessible through the Management Port, offset <a href="#">3ACh[0]</a>) (to delay linkup on all other Ports until Management Port software, serial EEPROM, and/or I<sup>2</sup>C/SMBus releases the hold by Setting the <i>Initiate Configuration</i> bit), all PORT_GOODx# signals are initially assigned to the Management Port. In this case, the PORT_GOODx# signals are not associated to corresponding Ports in the virtual switches, until Management Port software, the serial EEPROM, and/or I<sup>2</sup>C/SMBus Sets the <a href="#">Virtual Switch GPIO Update</a> register <i>VS GPIOs Update</i> bit (Port 0, accessible through the Management Port, offset <a href="#">64Ch[0]</a>), to complete the assignment of individual PORT_GOODx# signals to specific virtual switches (after software, the serial EEPROM, and/or I<sup>2</sup>C/SMBus configures functionality in the <b>GPIO</b> registers (Port 0, accessible through the Management Port, offsets <a href="#">600h</a>, <a href="#">604h</a>, <a href="#">614h</a>, <a href="#">61Ch</a>, and <a href="#">624h</a>)), and assigns the signals to virtual switches by programming the <b>VSx GPIO_PG 0_11 Assignment</b> register(s) (Port 0, accessible through the Management Port, offset(s) <a href="#">650h</a> through <a href="#">664h</a>). Therefore, in this mode, the PORT_GOODx# signals do not reflect Link status for virtual switch Ports, until Management Port software, the serial EEPROM, and/or I<sup>2</sup>C/SMBus completes PORT_GOODx# initialization, with one exception – the PORT_GOODx# signal that corresponds to the Management Port reflects the Management Port Link status, provided that PORT_GOOD output functionality is enabled (either as default with <a href="#">STRAP_TESTMODE[2:0]</a> 3-state inputs (sampled at <a href="#">PEX_PERST#</a> de-assertion; refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a> for details), or by Management Port software, the serial EEPROM, and/or I<sup>2</sup>C/SMBus programming the corresponding <b>GPIO 0_x Direction Control</b> register(s) (Port 0, accessible through the Management Port, offset(s) <a href="#">600h</a> and/or <a href="#">604h</a>)). <p>Continued...</p> </li></ol>		

Table 3-14. Device-Specific Signals (*Cont.*)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
PORT_GOOD[11:0]#	F25, AD2, AC3, AC2, Y3, W5, AD1, AB2, AE1, AB3, AB1, D3	PORT_GOOD[8:7]# – I/O, PD, PORT_GOOD[11:9, 6:0]# – I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to Section 3.4.4.1 for details
		<p>Continued...</p> <p><b>GPIO Function – Virtual Switch Mode (<i>Cont.</i>)</b></p> <p>2. If <b>STRAP_MGMT_PORT_EN</b>=1 or 0 to either (respectively) disable the Management Port, or enable the Management Port and Set the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, accessible through the Management Port, offset <b>3ACh[0]</b>) to allow all Ports to begin Link training after the serial EEPROM download completes immediately following Fundamental Reset (assuming the serial EEPROM does not Clear the <i>Initiate Configuration</i> bit), the PORT_GOODx# signals are individually associated to corresponding Ports. Therefore, the PORT_GOODx# signals reflect Link status of all enabled Ports, provided that PORT_GOOD output functionality is enabled (either as default with <b>STRAP_TESTMODE[2:0]</b> 3-state inputs (sampled at <b>PEX_PERST#</b> de-assertion; refer to <b>STRAP_TESTMODE[2:0]</b> and <a href="#">Section 3.4.4.1</a> for details), or by Management Port software, the serial EEPROM, and/or I<sup>2</sup>C/SMBus programming the <b>GPIO 0_x Direction Control</b> register(s) (Port 0, accessible through the Management Port, offset(s) <b>600h</b> and/or <b>604h</b>)).</p> <p>To assign individual PORT_GOODx# signals to specific virtual switch Upstream Ports, Management Port software, the serial EEPROM, and/or I<sup>2</sup>C can program the associated <b>VSx GPIO_PG 0_11 Assignment</b> register(s) (Port 0, accessible through the Management Port, offset(s) <b>650h</b> through <b>664h</b>). Individual PORT_GOODx# signal assignments must be mutually exclusive among the virtual switches.</p> <p>The 12 bits in each <b>VSx GPIO_PG 0_11 Assignment</b> register correspond to the 12 PORT_GOODx# signals. (Refer to the registers for signal-to-bit mapping.) Setting a bit in one of the six registers assigns the signal to the virtual switch indicated by the register name. Because any PORT_GOODx# signal must not be assigned to more than one virtual switch, <b>each of the 12 bits can be Set exclusively in only one of the six registers</b>. A maximum of 12 PORT_GOODx# signals can be assigned to any one virtual switch. After the <b>VSx GPIO_PG 0_11 Assignment</b> register(s) are programmed, the actual assignments do not take effect, until Management Port software, the serial EEPROM, and/or I<sup>2</sup>C/SMBus Sets the <b>Virtual Switch GPIO Update</b> register <i>VS GPIOs Update</i> bit (Port 0, accessible through the Management Port, offset <b>64Ch[0]</b>).</p> <p>After the PORT_GOODx# signals are assigned to virtual switches, each virtual switch can then configure individual PORT_GOODx# signal functionality, by programming the <b>Virtual Switch GPIO_PG 0_x Direction Control</b> register(s) (VS Upstream Port(s), offset(s) <b>A34h</b> and/or <b>A38h</b>). In these registers, the PORT_GOODx# signals are virtualized, and the bit numbers do <b>not</b> correspond to specific signal names.</p> <p>Software can determine how many and which PORT_GOODx# signals are assigned to a unique virtual switch, by reading the <b>Virtual Switch GPIO_PG 0_11 Availability</b> register <i>Number of GPIO_PGs Available</i> field (VS Upstream Port(s), offset <b>A3Ch[3:0]</b>). Virtual switch software does not determine which of the actual PORT_GOODx# signals are assigned to the virtual switch. The bit numbering is relative within the context of each virtual switch.</p>		

Table 3-14. Device-Specific Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
VS5_FATAL_ERR# VS4_FATAL_ERR# VS3_FATAL_ERR# VS2_FATAL_ERR# VS1_FATAL_ERR# VS0_FATAL_ERR#	D24, B26, D26, W22, AE26, Y23	VS5_FATAL_ERR# – I/O, PD VS4_FATAL_ERR# – I/O, PU VS3_FATAL_ERR# – I/O, PU VS2_FATAL_ERR# – I/O, PD VS1_FATAL_ERR# – I/O, PU VS0_FATAL_ERR# – I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a>
		<b>Virtual Switch Mode Fatal Error Output (6 Balls)</b> VS <sub>x</sub> _FATAL_ERR# are used only in Virtual Switch mode (one per virtual switch – Virtual Switches 5 through 0, respectively). VS <sub>x</sub> _FATAL_ERR# are asserted Low when a Fatal error is detected in the PEX 8748 and the following conditions are met (all the same conditions that are required to send a Fatal Error Message to the Host): <ul style="list-style-type: none"> <li>Specific error is defined as <i>Fatal</i> in the <a href="#">Uncorrectable Error Severity</a> register (offset <a href="#">FC0h</a>), and</li> <li>Reporting of the specific error condition is enabled, not masked by the <a href="#">Uncorrectable Error Mask</a> register's (offset <a href="#">FBCh</a>) corresponding <i>Interrupt Mask</i> bit, and</li> <li><a href="#">Device Control</a> register <i>Fatal Error Reporting Enable</i> bit (offset <a href="#">70h[2]</a>) –or– <a href="#">PCI Command</a> register <i>SERR# Enable</i> bit (offset <a href="#">04h[8]</a>) is Set</li> </ul> The <a href="#">Device Status</a> register <i>Fatal Error Detected</i> bit (offset <a href="#">70h[18]</a> ) is Set, and the specific error is flagged in the <a href="#">Uncorrectable Error Status</a> register (offset <a href="#">FB8h</a> ). <i>Note:</i> Although these are I/O signals, their logical operation in Virtual Switch mode is output.		
VS5_PERST# VS4_PERST# VS3_PERST# VS2_PERST# VS1_PERST# VS0_PERST#	D25, G21, C25, AD26, AB26, Y24	I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a>
		<b>Virtual Switch Mode Fundamental Reset (6 Balls)</b> VS <sub>x</sub> _PERST# are used only in Virtual Switch mode (one per virtual switch – Virtual Switches 5 through 0, respectively). Causes a Fundamental Reset in Virtual Switch mode (PERST#). (Refer to <a href="#">Section 5.1.2, “Resets – Virtual Switch Mode,”</a> for further details.) <i>Notes:</i> Although these are I/O signals, their logical operation in Virtual Switch mode is input. The <a href="#">PEX_PERST#</a> signal, defined in <a href="#">Table 3-3</a> , is the Reset input used in Base mode. Because the Virtual Switch Table registers (refer to <a href="#">Section 5.3.3.1, “Virtual Switch Table Registers”</a> ) exist in Port 0, Port 0 must not be held in Reset during setup of Virtual Switch mode. Therefore, for whichever virtual switch includes Port 0, its VS <sub>x</sub> _PERST# input must be de-asserted (High). Assigning Port 0 to be the Management Port might provide the simplest solution toward meeting this requirement.		

Table 3-14. Device-Specific Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
VS5_PEX_INTA# VS4_PEX_INTA# VS3_PEX_INTA# VS2_PEX_INTA# VS1_PEX_INTA# VS0_PEX_INTA#	G22, F24, B25, AB25, AA24, AC26	I/O, PU	Yes Refer to <a href="#">Section 3.4.4.1</a>	Refer to <a href="#">STRAP_TESTMODE[2:0]</a> and <a href="#">Section 3.4.4.1</a>
		<p><b>Virtual Switch Mode Interrupt Outputs (6 Balls)</b></p> <p>VS<sub>x</sub>_PEX_INTA# are used only in Virtual Switch mode (one per virtual switch – Virtual Switches 5 through 0, respectively).</p> <p>VS<sub>x</sub>_PEX_INTA# Interrupt output is enabled if:</p> <ul style="list-style-type: none"> <li>INT<sub>x</sub> Messages are enabled (Port's <b>PCI Command</b> register <i>Interrupt Disable</i> bit, offset 04h[10], is Cleared), and MSIs are disabled (Port's <b>MSI Control</b> register <i>MSI Enable</i> bit, offset 48h[16], is Cleared)</li> <li>VS<sub>x</sub>_PEX_INTA# output (<b>ECC Error Check Disable</b> register <i>Enable PEX_INTA# or VS<sub>x</sub>_PEX_INTA# Interrupt Output(s) for x Interrupt</i> bit(s) (offset 720h[9, 8, 7, 6, 5, and/or 4]; refer to the register description, for Port associations)) is enabled</li> </ul> <p>The three interrupt mechanisms, listed below, are mutually exclusive modes of operation, on a per-Port basis, for all interrupt sources:</p> <ul style="list-style-type: none"> <li>Conventional PCI INT<sub>x</sub> Message generation</li> <li>Native MSI transaction generation</li> <li>Device-Specific VS<sub>x</sub>_PEX_INTA# assertion</li> </ul> <p>VS<sub>x</sub>_PEX_INTA# assertion (Low) indicates that one or more of the following events and/or errors (if not masked) were detected:</p> <ul style="list-style-type: none"> <li>Hot Plug or Link State events</li> <li>PCI Express Hot Plug events</li> <li>General-Purpose Input Interrupt events</li> <li>Device-Specific NT-Link Port errors and events</li> <li>NT-Virtual Doorbell events</li> <li>NT-Link Doorbell events</li> <li>Management Port Doorbell-Generated interrupts</li> <li>Management Link Status events</li> </ul> <p>Refer to <a href="#">Section 9.1.1</a>, “Interrupt Sources or Events,” for details.</p> <p><b>Note:</b> Although these are I/O signals, their logical operation in Virtual Switch mode is output.</p>		

### 3.4.8 External Resistor Signals

Table 3-15. External Resistor Signals

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
REXT_A[11:0]	D18, E24, N23, Y25, D9, A3, H3, N4, AB24, AC18, AC9, Y4	A	No	—
		<b>External Bias Resistor Balls (12 Balls)</b> One REXT_Ax ball is provided for each SerDes quad, per Station. <b>An external resistor (3.00KΩ, 1%) must be connected between each REXT_Ax ball and VSS (Ground).</b> Each resistor should be placed close to the PEX 8748, for minimal noise injection. <b>Bypass capacitors must not be connected to these balls.</b>  <i>Note: A 3.01K 0.1% resistor can be used, because that value is within the 3.00K ±1% specification.</i>		
REXT_B[11:0]	E18, F18, N22, W23, E9, C1, G3, N5, Y22, AB18, AB9, Y5	A	No	—
		<b>External Bias Resistor Balls (12 Balls)</b> One REXT_Bx ball is provided for each SerDes quad, per Station. <b>An external resistor (3.00KΩ, 1%) must be connected between each REXT_Bx ball and VSS (Ground).</b> Each resistor should be placed close to the PEX 8748, for minimal noise injection. <b>Bypass capacitors must not be connected to these balls.</b>  <i>Note: A 3.01K 0.1% resistor can be used, because that value is within the 3.00K ±1% specification.</i>		
REXT_CMU	Y1	A	No	—
		<b>Clock Management Unit for SerDes</b> <b>An external resistor (3.00KΩ, 1%) must be attached between REXT_CMU and VSS (Ground).</b>  <i>Note: A 3.01K 0.1% resistor can be used, because that value is within the 3.00K ±1% specification.</i>		

### 3.4.9 Thermal Diode Signals

The PEX 8748 includes a thermal-sensing diode, to facilitate the measurement of on-die device junction temperature. On the package level, the thermal diode inputs listed in [Table 3-16](#) connect to anode and cathode terminals of a diode-connected transistor (PNP) implemented on the die.

For further details, refer to [Section 17.8, “Thermal Sensor Diode.”](#)

**Table 3-16. Thermal Diode Signals**

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
THERMAL_DIODEn	AC24	I	No	–
		<b>Thermal Diode Negative</b> THERMAL_DIODEn can be connected to the negative terminal of a Temperature Sensor IC, to monitor the approximate temperature of the die. <i>Note:</i> Subtract 24°C from the temperature value reported by the Temperature Sensor IC, to determine the correct die temperature.		
THERMAL_DIODEp	AF25	I	No	–
		<b>Thermal Diode Positive</b> THERMAL_DIODEp can be connected to the positive terminal of a Temperature Sensor IC, to monitor the approximate temperature of the die. <i>Note:</i> Subtract 24°C from the temperature value reported by the Temperature Sensor IC, to determine the correct die temperature.		

### 3.4.10 No Connect Signals

**Caution:** Do not connect these balls to board electrical paths.  
These balls are internally connected to the device.

**Table 3-17. No Connect Signals**

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
N/C	A9, A18, A24, B9, B18, B24, E1, F1, F9, N1, N2, N25, N26, W1, W2, W4, W24, W25, Y2, AA9, AA25, AA26, AE2, AE3, AE18, AF2, AF3, AF18	<i>Reserved</i>	No	–
		<b>No Connect (28 Balls)</b> Do not connect these balls to board electrical paths.		
SPARE3	AA1	I/O, PD	No	–
		<b>Spare 3</b> <i>Reserved for future use</i> <b>Note:</b> Normally, this I/O should be pulled Low to <b>VSS</b> (Ground). Optionally, this I/O can remain unconnected, because the internal pull-down resistor holds the input Low.		
SPARE2	AA2	I/O, PU	No	–
		<b>Compatibility Enable for Non-Compliant Gen 1 Endpoints</b> When SPARE2 is Low and the Link training sequence fails during either the LTSSM <i>Polling</i> or <i>Configuration</i> state, the next time the LTSSM exits the <i>Detect</i> state, TS Ordered-Sets advertise only the Gen 1 data rate, and no Autonomous Change support. When SPARE2 is High, the Data Rate Identifier symbol in the TS Ordered-Sets always advertises support for both the Gen 3 data rate and Autonomous Change. <b>Notes:</b> This feature should be enabled only if a non-compliant Gen 1 device will not linkup if these Data Rate Identifier bits are Set. Normally, this I/O should be pulled High to <b>VDD18</b> . Although this is an I/O signal, its logical operation is input. Optionally, this I/O can remain unconnected, because the internal pull-up resistor holds the input High.		



Table 3-17. No Connect Signals (Cont.)

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
SPARE1	AA3	I/O, PU	No	–
		<p><b>HP_PWREN_x Polarity Inversion</b></p> <p>The SPARE1 input state is not mapped to a register bit (and therefore HP_PWREN_x polarity is not programmable by software, serial EEPROM, nor I<sup>2</sup>C/SMBus).</p> <p>An external resistor is not required unless:</p> <ul style="list-style-type: none"> <li>• HP_PWREN_x outputs must be Active-Low, –or–</li> <li>• SPARE1 is connected to a circuit trace (the internal resistor is relatively weak, and might not be sufficiently strong, to hold the SPARE1 input High)</li> </ul> <p>0 = HP_PWREN_x outputs are Active-Low  1 = HP_PWREN_x outputs are Active-High (default, by virtue of the internal pull-up resistor)</p> <p><i>Notes: Serial Hot Plug PWREN outputs (from connected I<sup>2</sup>C I/O Expanders) are always Active-High.</i></p> <p><i>Although this is an I/O signal, its logical operation is input.</i></p> <p><i>This input is continuously sampled.</i></p> <p><i>Normally, this I/O should be pulled High to VDD18. Optionally, this I/O can remain unconnected, because the internal pull-up resistor holds the input High.</i></p>		
SPARE0	D2	I/O, PU	No	–
		<p><b>Spare 0</b></p> <p><i>Reserved for future use</i></p> <p><i>Note: Normally, this I/O should be pulled High to VDD18. Optionally, this I/O can remain unconnected, because the internal pull-up resistor holds the input High.</i></p>		

### 3.4.11 Power and Ground Signals

In general, each power ball should connect to one de-coupling capacitor. Use a mix of values, *such as* 0.1 and 0.01  $\mu$ F.

Due to VDD09 in-rush current at the moment PEX\_PERST# input is de-asserted, bulk capacitance is needed, to prevent the voltage rail from falling below minimum voltage requirements. *For example*, the PEX 8748 RDK uses three 680  $\mu$ F capacitors for this. This particular capacitance value is not required; however, it is best to use multiple capacitors in parallel. Inductance must be low.

**Table 3-18. Power and Ground Signals**

Signal Name	Location	Type	Multiplexed Y/N	STRAP_TESTMODE[2:0] Input States
		Description		
VDD09	L11, L13, L15, M12, M14, M16, N11, N13, N15, P12, P14, P16, R11, R13, R15, T12, T14, T16	CPWR	No	–
		<b>0.9V -5%, +5.55% Power for Core and SerDes Digital Logic (18 Balls)</b>		
VDD09A	G9, G10, G11, G12, G13, G14, G15, G16, G17, G18, J7, J11, J13, J14, J16, J20, K7, K20, L7, L9, L18, L20, M7, M9, M18, M20, N7, N20, P7, P9, P18, P20, R7, R10, R17, R20, T7, T9, T18, T20, U7, U20, V11, V13, V14, V16, Y9, Y10, Y11, Y12, Y13, Y14, Y15, Y16, Y17, Y18	APWR	No	–
		<b>0.9V -5%, +5.55% Power for SerDes Analog Circuits (56 Balls)</b>		
VDD18	G7, G20, H8, H19, J9, J18, K10, K17, U10, U17, V8, V9, V18, V19, W7, W8, W19, W20, Y7, Y20	I/OPWR	No	–
		<b>1.8V -5%, +10% Power for I/O Logic Functions (20 Balls)</b>		
VDD18A	K12, K15, N10, N17, U12, U15	PLLWPR	No	–
		<b>1.8V -5%, +10% Power for PLL Circuits (6 Balls)</b>		
VSS	A1, A4, A23, A25, A26, B4, B23, C5, C6, C7, C8, C9, C10, C11, C12, C13, C14, C15, C16, C17, C18, C19, C20, C21, C22, D4, D23, E4, E23, F4, F5, F6, F7, F8, F10, F11, F12, F13, F14, F15, F16, F17, F19, F20, F21, F22, F23, G8, G19, H1, H2, H4, H5, H6, H7, H9, H10, H11, H12, H13, H14, H15, H16, H17, H18, H20, H21, H22, H23, H25, H26, J3, J6, J8, J10, J12, J15, J17, J19, J21, J24, K3, K6, K8, K9, K11, K13, K14, K16, K18, K19, K21, K24, L3, L6, L8, L10, L12, L14, L16, L17, L19, L21, L24, M3, M6, M8, M10, M11, M13, M15, M17, M19, M21, M24, N3, N8, N9, N12, N14, N16, N18, N19, N24, P3, P6, P8, P10, P11, P13, P15, P17, P19, P21, P24, R3, R6, R8, R9, R12, R14, R16, R18, R19, R21, R24, T3, T6, T8, T10, T11, T13, T15, T17, T19, T21, T24, U3, U6, U8, U9, U11, U13, U14, U16, U18, U19, U21, U24, V1, V2, V3, V4, V5, V6, V7, V10, V12, V15, V17, V20, V21, V22, V23, V24, V25, V26, W9, W10, W11, W12, W13, W14, W15, W16, W17, W18, Y8, Y19, AA4, AA5, AA6, AA7, AA8, AA10, AA11, AA12, AA13, AA14, AA15, AA16, AA17, AA19, AA20, AA21, AA22, AA23, AB4, AB23, AC4, AC23, AD5, AD6, AD7, AD8, AD9, AD10, AD11, AD12, AD13, AD14, AD15, AD16, AD17, AD18, AD19, AD20, AD21, AD22, AD23, AE4, AE23, AF1, AF4, AF23, AF26	GND	No	–
		<b>Ground Connections (246 Balls)</b>		

# 3.5 Physical Layout

Figure 3-2. Physical Ball Assignment (See-Through Top View)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26		
A	VSS	STRAP_NT UPSTRM PORTSEL0	REXT_A6	VSS	PEX_PETn 24	PEX_PETn 25	PEX_PETn 26	PEX_PETn 27	N/C	PEX_PETn 28	PEX_PETn 29	PEX_PETn 30	PEX_PETn 31	PEX_PETn 47	PEX_PETn 46	PEX_PETn 45	PEX_PETn 44	N/C	PEX_PETn 43	PEX_PETn 42	PEX_PETn 41	PEX_PETn 40	VSS	N/C	VSS	VSS	A	
B	JTAG_TMS	JTAG_TCK	SHPC_INT #	VSS	PEX_PETp 24	PEX_PETp 25	PEX_PETp 26	PEX_PETp 27	N/C	PEX_PETp 28	PEX_PETp 29	PEX_PETp 30	PEX_PETp 31	PEX_PETp 47	PEX_PETp 46	PEX_PETp 45	PEX_PETp 44	N/C	PEX_PETp 43	PEX_PETp 42	PEX_PETp 41	PEX_PETp 40	VSS	N/C	HP_ATNLE D_C#_VSS 4_FATAL ERR#	HP_PWRP LT_C#_VSS 4_FATAL ERR#	B	
C	REXT_B6	JTAG_TRS T#	JTAG_TDI	STRAP_ST N1_PORTC FG0	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	STRAP_ST N2_PORTC FG0	FATAL_ER R#	HP_BUTT ON_C#_VSS 3_PERST#	STRAP_ST N2_PORTC FG1	C	
D	PEX_INTA #	SPARE0	PORT_GO OD#	VSS	PEX_PERn 24	PEX_PERn 25	PEX_PERn 26	PEX_PERn 27	REXT_A7	PEX_PERn 28	PEX_PERn 29	PEX_PERn 30	PEX_PERn 31	PEX_PERn 47	PEX_PERn 46	PEX_PERn 45	PEX_PERn 44	REXT_A11	PEX_PERn 43	PEX_PERn 42	PEX_PERn 41	PEX_PERn 40	VSS	HP_PWRE N_C_VSS FATAL_ER R#	HP_PRSN T_C#_VSS 3_FATAL ERR#	HP_PWRL ED_C#_VSS 3_FATAL ERR#	D	
E	N/C	I2C_ADDR0	I2C_SDA0	VSS	PEX_PERp 24	PEX_PERp 25	PEX_PERp 26	PEX_PERp 27	REXT_B7	PEX_PERp 28	PEX_PERp 29	PEX_PERp 30	PEX_PERp 31	PEX_PERp 47	PEX_PERp 46	PEX_PERp 45	PEX_PERp 44	REXT_B11	PEX_PERp 43	PEX_PERp 42	PEX_PERp 41	PEX_PERp 40	VSS	REXT_A10	EE_SK	EE_CS#	E	
F	N/C	STRAP_I2 C_SMBUS _EN	STRAP_RE SERVED4	VSS	VSS	VSS	VSS	VSS	N/C	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	REXT_B10	VSS	VSS	VSS	VSS	VSS	HP_CLKEN C#_VSS4 PEX_INTA #	HP_PWRL GOOD_C- PORT_GO OD11#	HP_PWRL GOOD_C- PORT_GO OD11#	EE_DO	F
G	STRAP_ST N1_PORTC FG1	STRAP_M GMT_POR T_EN	REXT_B5	I2C_SCL0	JTAG_TDO	STRAP_NT UPSTRM PORTSEL1	VDD18	VSS	VDD09A	VDD09A	VDD09A	VDD09A	VDD09A	VDD09A	VDD09A	VDD09A	VDD09A	VDD09A	VSS	VDD18	HP_MRL_C #_VSS4_P E_RST#	HP_PERST C#_VSS PEX_INTA #	EE_DI	STRAP_UP STRM_PO RTSEL1	STRAP_UP STRM_PO RTSEL0	STRAP_UP STRM_PO RTSEL2	G	
H	VSS	VSS	REXT_A5	VSS	VSS	VSS	VSS	VDD18	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VDD18	VSS	VSS	VSS	VSS	STRAP_G EN1_GEN2	VSS	VSS	H	
J	PEX_PETn 23	PEX_PETp 23	VSS	PEX_PERn 23	PEX_PERp 23	VSS	VDD09A	VSS	VDD18	VSS	VDD09A	VSS	VDD09A	VDD09A	VSS	VDD09A	VSS	VDD18	VSS	VDD09A	VSS	PEX_PERp 39	PEX_PERn 39	VSS	PEX_PETp 39	PEX_PETn 39	J	
K	PEX_PETn 22	PEX_PETp 22	VSS	PEX_PERn 22	PEX_PERp 22	VSS	VDD09A	VSS	VSS	VDD18	VSS	VDD18A	VSS	VSS	VDD18A	VSS	VDD18	VSS	VSS	VDD09A	VSS	PEX_PERp 38	PEX_PERn 38	VSS	PEX_PETp 38	PEX_PETn 38	K	
L	PEX_PETn 21	PEX_PETp 21	VSS	PEX_PERn 21	PEX_PERp 21	VSS	VDD09A	VSS	VDD09A	VSS	VDD09	VSS	VDD09	VSS	VDD09	VSS	VDD09	VSS	VDD09A	VSS	VDD09A	VSS	PEX_PERp 37	PEX_PERn 37	VSS	PEX_PETp 37	PEX_PETn 37	L
M	PEX_PETn 20	PEX_PETp 20	VSS	PEX_PERn 20	PEX_PERp 20	VSS	VDD09A	VSS	VDD09A	VSS	VSS	VDD09	VSS	VDD09	VSS	VDD09	VSS	VDD09A	VSS	VDD09A	VSS	PEX_PERp 36	PEX_PERn 36	VSS	PEX_PETp 36	PEX_PETn 36	M	
N	N/C	N/C	VSS	REXT_A4	REXT_B4	STRAP_RE SERVED1	VDD09A	VSS	VSS	VDD18A	VDD09	VSS	VDD09	VSS	VDD09	VSS	VDD18A	VSS	VSS	VDD09A	VSS	STRAP_RE SERVED2	REXT_B9	REXT_A9	VSS	N/C	N/C	N
P	PEX_PETn 19	PEX_PETp 19	VSS	PEX_PERn 19	PEX_PERp 19	VSS	VDD09A	VSS	VDD09A	VSS	VSS	VDD09	VSS	VDD09	VSS	VDD09	VSS	VDD09A	VSS	VDD09A	VSS	PEX_PERp 35	PEX_PERn 35	VSS	PEX_PETp 35	PEX_PETn 35	P	
R	PEX_PETn 18	PEX_PETp 18	VSS	PEX_PERn 18	PEX_PERp 18	VSS	VDD09A	VSS	VSS	VDD09A	VDD09	VSS	VDD09	VSS	VDD09	VSS	VDD09A	VSS	VSS	VDD09A	VSS	PEX_PERp 34	PEX_PERn 34	VSS	PEX_PETp 34	PEX_PETn 34	R	
T	PEX_PETn 17	PEX_PETp 17	VSS	PEX_PERn 17	PEX_PERp 17	VSS	VDD09A	VSS	VDD09A	VSS	VSS	VDD09	VSS	VDD09	VSS	VDD09	VSS	VDD09A	VSS	VDD09A	VSS	PEX_PERp 33	PEX_PERn 33	VSS	PEX_PETp 33	PEX_PETn 33	T	
U	PEX_PETn 16	PEX_PETp 16	VSS	PEX_PERn 16	PEX_PERp 16	VSS	VDD09A	VSS	VSS	VDD18	VSS	VDD18A	VSS	VSS	VDD18A	VSS	VDD18	VSS	VSS	VDD09A	VSS	PEX_PERp 32	PEX_PERn 32	VSS	PEX_PETp 32	PEX_PETn 32	U	
V	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VDD18	VDD18	VSS	VDD09A	VSS	VDD09A	VDD09A	VSS	VDD09A	VSS	VDD18	VDD18	VSS	VSS	VSS	VSS	VSS	VSS	VSS	V	
W	N/C	N/C	STRAP_RE SERVED3	N/C	HP_PRSN T_B#_POR T_GOOD6#	STRAP_I2 C_SMBUS _CFG_EN#	VDD18	VDD18	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VDD18	VDD18	HP_PWRL GOOD_A- VS_MODE	HP_PWRE N_A_VSS2 FATAL_ER R#	REXT_B8	N/C	N/C	STRAP_SE RDES_MO DE_EN#	W	
Y	REXT_CM U	N/C	HP_PWRL GOOD_B- PORT_GO OD7#	REXT_A0	REXT_B0	STRAP_TE STMODE0	VDD18	VSS	VDD09A	VDD09A	VDD09A	VDD09A	VDD09A	VDD09A	VDD09A	VDD09A	VDD09A	VDD09A	VSS	VDD18	STRAP_PR OBE_MOD E#	REXT_B3	HP_PWRL ED_A#_VS 0_FATAL- ERR#	HP_BUTT ON_A#_VS 0_PERST#	REXT_A8	STRAP_PL L_BYPASS #	Y	
AA	SPARE3	SPARE2	SPARE1	VSS	VSS	VSS	VSS	VSS	N/C	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	STRAP_NT UPSTRM PORTSEL2	VSS	VSS	VSS	VSS	VSS	N/C	N/C	N/C	AA	
AB	HP_ATNLE D_B#_POR T_GOOD1#	HP_MRL_B D#_PORT_G OOD4#	HP_BUTT ON_B#_PO RT_GOOD2#	VSS	PEX_PERp 0	PEX_PERp 1	PEX_PERp 2	PEX_PERp 3	REXT_B1	PEX_PERp 4	PEX_PERp 5	PEX_PERp 6	PEX_PERp 7	PEX_PERp 8	PEX_PERp 9	PEX_PERp 10	PEX_PERp 11	REXT_B2	PEX_PERp 12	PEX_PERp 13	PEX_PERp 14	PEX_PERp 15	VSS	REXT_A3	HP_PERST _A#_VSS2 PEX_INTA #	HP_MRL_A #_VSS1_P E_RST#	AB	
AC	HP_PWRE N_B#_POR T_GOOD8#	HP_PWRL LT_B#_PO RT_GOOD9#	VSS	PEX_PERn 0	PEX_PERn 1	PEX_PERn 2	PEX_PERn 3	REXT_A1	PEX_PERn 4	PEX_PERn 5	PEX_PERn 6	PEX_PERn 7	PEX_PERn 8	PEX_PERn 9	PEX_PERn 10	PEX_PERn 11	REXT_A2	PEX_PERn 12	PEX_PERn 13	PEX_PERn 14	PEX_PERn 15	VSS	THERMAL DIODE#	I2C_SCL1	HP_PWRL LT_A#_VSS 1_FATAL ERR#	AC		
AD	HP_PERST ED_B#_PO RT_GOOD10#	HP_PWRL ED_B#_PO RT_GOOD10#	STRAP_TE STMODE1	STRAP_TE STMODE2	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	STRAP_ST NO_PORTC FG0	PEX_NT_P ERST#	HP_PRSN T_A#_VSS PERST#	AD	
AE	HP_CLKEN B#_PORT _GOOD3#	N/C	N/C	VSS	PEX_PETp 0	PEX_PETp 1	PEX_PETp 2	PEX_PETp 3	PEX_REFp Lk0	PEX_PETp 4	PEX_PETp 5	PEX_PETp 6	PEX_PETp 7	PEX_PETp 8	PEX_PETp 9	PEX_PETp 10	PEX_PETp 11	N/C	PEX_PETp 12	PEX_PETp 13	PEX_PETp 14	PEX_PETp 15	VSS	STRAP_RE SERVED16	I2C_SDA1	HP_PWRL LT_A#_VSS 1_FATAL ERR#	AE	
AF	VSS	N/C	N/C	VSS	PEX_PETn 0	PEX_PETn 1	PEX_PETn 2	PEX_PETn 3	PEX_REFp Lk0	PEX_PETn 4	PEX_PETn 5	PEX_PETn 6	PEX_PETn 7	PEX_PETn 8	PEX_PETn 9	PEX_PETn 10	PEX_PETn 11	N/C	PEX_PETn 12	PEX_PETn 13	PEX_PETn 14	PEX_PETn 15	VSS	STRAP_ST NO_PORTC FG1	THERMAL DIODE#	VSS	AF	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26		

THIS PAGE INTENTIONALLY LEFT BLANK.



## Chapter 4 Functional Overview

### 4.1 Introduction

The PEX 8748 is designed with a flexible, modular architecture. This chapter provides an overview of the PEX 8748's major functions:

- [Modes of Operation](#)
- [Software Architecture](#)
- [Hardware Architecture](#)
- [PCI Express Station Functional Description](#)
- [Physical Layer](#)
- [Transaction Layer](#)

Subsequent chapters go into further detail.

## 4.2 Modes of Operation

The PEX 8748 supports and implements two modes of operation – [Base Mode](#) and [Virtual Switch Mode](#).

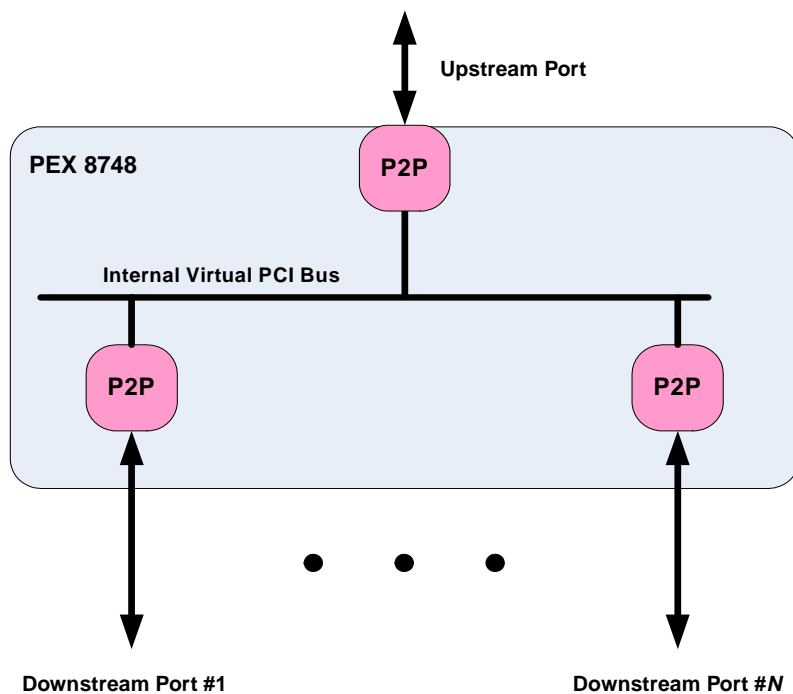
### 4.2.1 Base Mode

The PEX 8748 is a 48-Lane, 12-Port switch. [Figure 4-1](#) illustrates the PEX 8748 in Base mode, from a software point of view. In Base mode, the PEX 8748 supports one Upstream Port and up to 11 Downstream Ports.

The PEX 8748 also implements a single NT Port, which can be optionally enabled and configured to be any Port within the PEX 8748. NT mode is useful in supporting high-availability systems and failover.

**Note:** The PCI-to-PCI (P2P) blocks in [Figure 4-1](#) are a logical representation of how a Port presents itself to software. Each P2P block is a Type 1 PCI function, which is a PCI-to-PCI bridge.

**Figure 4-1. Software Point-of-View – Base Mode**



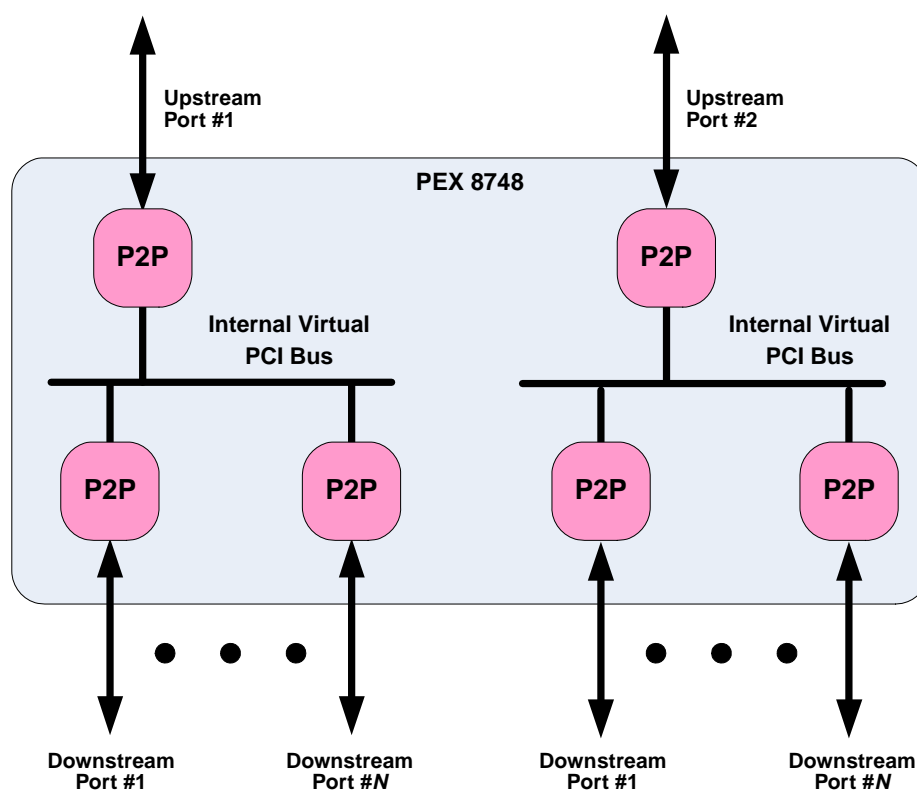
## 4.2.2 Virtual Switch Mode

In Virtual Switch mode, the PEX 8748 can be partitioned in up to six independent virtual switches, using one piece of silicon. The switches are still separate – there is no path between them. Each virtual switch is part of an independent PCI Express hierarchy, and the different virtual switches do not share Ports. Although the virtual switches share the same hardware infrastructure, they enforce the security between them so the traffic from one virtual switch does not leak into other virtual switches. Additionally, by re-programming the virtual switch to which a Port belongs, it is possible to support various failover and load balancing uses. A virtual switch can span across multiple Stations, and a Station can be shared by multiple virtual switches.

For example, the PEX 8748 can be partitioned into two virtual switches, as illustrated in Figure 4-2. This enables the PEX 8748 to be used in applications *such as* two-way machines, where one Root Complex is attached to one Upstream Port, and a second Root Complex is attached to a second Upstream Port.

Refer to Chapter 12, “Virtual Switch Mode,” for further details.

**Figure 4-2. Example of Using Two Virtual Switches – Virtual Switch Mode**



## 4.3 Software Architecture

### 4.3.1 PCI-Compatible Software Model

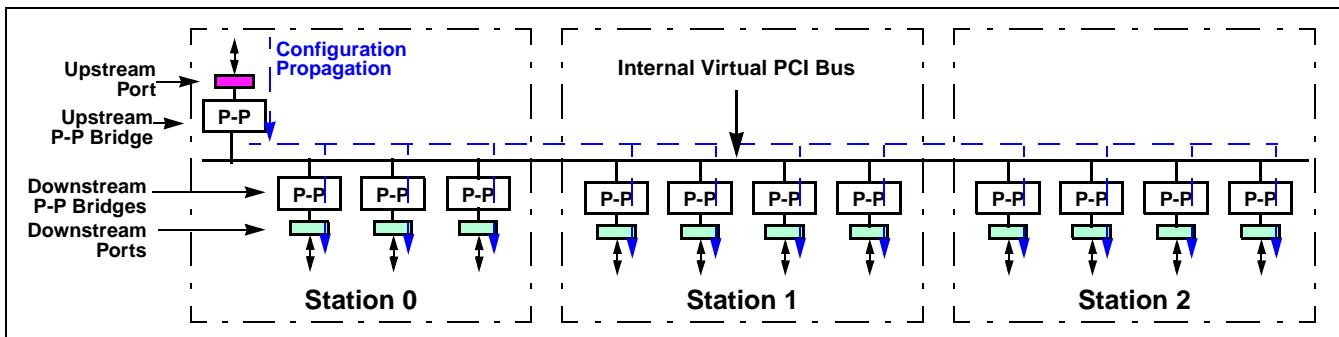
The PEX 8748 can be thought of as a hierarchy of PCI-to-PCI bridges, with one Upstream PCI-to-PCI bridge and one or more Downstream PCI-to-PCI bridges connected by an internal Virtual PCI Bus. (Refer to [Figure 4-3](#), which represents the PEX 8748 in Base mode.) PCI-to-PCI bridges are compliant with the PCI and PCI Express system models, allowing software written for Conventional PCI devices to work on the latest PCI Express switches.

The Upstream Port has the lowest Bus Number in a PCI Express switch. From the point of view of the PCI-to-PCI bridge, there is a:

- Primary Bus Number (the Upstream Bus Number),
- Secondary Bus Number (the Downstream Bus Number), and
- Subordinate Bus Number (highest Bus Number below the bridge).

The Secondary and Subordinate Bus Numbers form a range, within which all Downstream devices must fit. This can cause problems for systems that might allow hot-insertion of additional devices, unless planned for in advance, and a gap is created within in the Bus Number range, to allow for new device(s). Typically, system BIOS scans the PCI Bus and assigns Bus Numbers; therefore, support for hot insertion can depend upon the BIOS used.

**Figure 4-3. System Configuration Propagation – Base Mode**



**Note:** [Table 4-1](#) indicates which Ports are enabled/used, and when, based upon the STRAP\_STNx\_PORTCFGx input states and/or serial EEPROM programming.



The Configuration Space registers (CSRs) in the Upstream PCI-to-PCI bridge are accessible by Type 0 Configuration Requests that target the Upstream bus interface. The Upstream Port(s) capture(s) the Type 0 Configuration Write Target Bus Number and Device Number. The Upstream Port(s) then use(s) this Captured Bus Number and Captured Device Number, as part of the Requester ID and Completer ID for the Requests and Completions generated by the Upstream Port(s).

The CSRs in the Downstream Port PCI-to-PCI bridges are accessible by Type 1 Configuration Requests received at the Upstream Port(s) that target the internal virtual PCI Bus, by having a Bus Number value that matches the Upstream bridge's Secondary Bus Number value. Each Downstream bridge is associated with a unique Device Number, as explained in [Section 4.4.1.2](#).

The CSRs of Downstream devices are hit in two ways. If the Configuration Request matches the PEX 8748 Downstream Port Secondary Bus Number, the PEX 8748 converts the Type 1 Configuration Request into a Type 0 Configuration Request. However, if the Bus Number does *not* match the Secondary Bus Number, but falls within the Subordinate Bus Number range, the Type 1 Configuration Request is forwarded out of the PEX 8748, unchanged. A Type 1 Configuration Request that targets a Bus Number that is not within range is invalid, and is terminated by the PEX 8748 Upstream Port(s) as an Unsupported Request (UR).

After all PCI devices have been located and assigned Bus and Device Numbers, software can assign a Memory map and I/O map. In the PEX 8748, each Downstream bridge has its own Base and Limit. Software will Set the Upstream PCI-to-PCI bridge Base and Limit to include all Downstream Base and Limit ranges. As a result, all Downstream devices are contiguous in the Memory map. Requests (Memory or I/O) go Downstream if they fall within a bridge's Base and Limit range. Alternatively, Requests (Memory or I/O) go Upstream if they do not target anything within the Upstream bridge's Base and Limit range.

Completions are routed by the Bus Number established in the Configuration registers. If the Bus Number is in the Secondary or Subordinate range, the packet goes Downstream; otherwise, the packet goes Upstream.

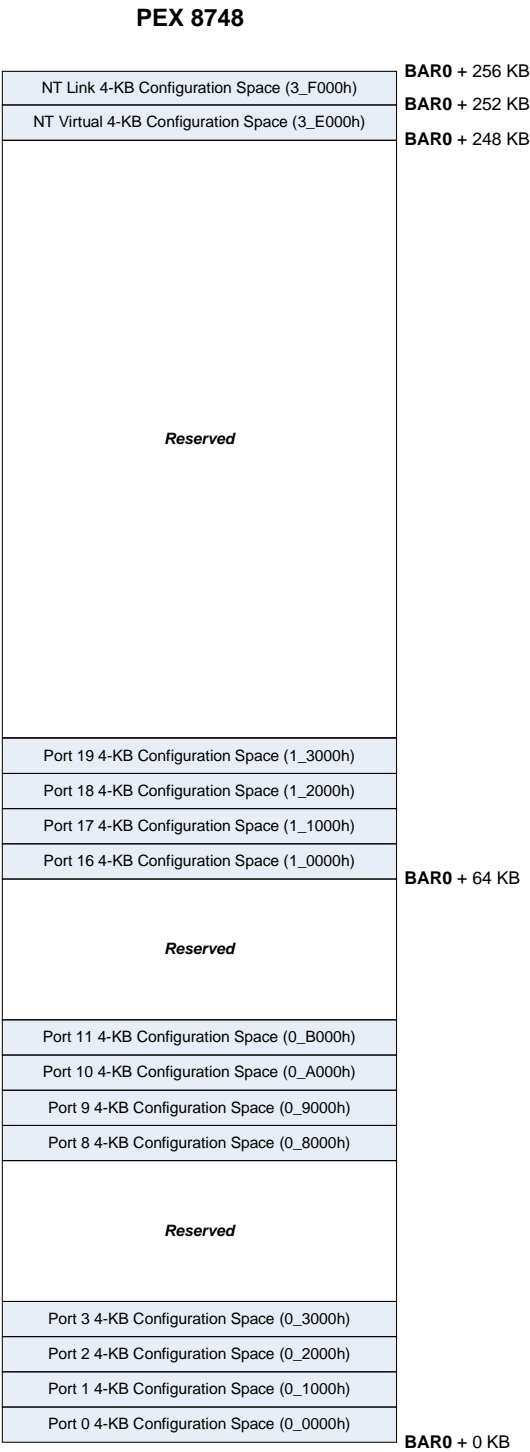
PLX has enhanced the basic PCI-to-PCI bridge with PCI Express-defined capabilities, Device-Specific capabilities, and additional functions.

### 4.3.2 PCI Express Memory-Mapped Configuration Space

The Configuration space is also mapped into Memory space that is seen by **Base Address Register 0 (BAR0)** of the PCI-to-PCI bridges, which is termed *Memory-Mapped Configuration space*. All PEX 8748 Ports, including the NT Port, fit into a single, flattened Memory space, as illustrated in [Figure 4-4](#).

Each Port Configuration space occupies 4 KB of the CSR Memory map. Port 0 is located at offset 0 KB, Port 1 is located at offset 4 KB, and so forth, for all enabled Ports, up to 256 KB. 4-KB register blocks that correspond to non-enabled Ports are *Reserved*.

Figure 4-4. PCI Express Memory-Mapped Configuration Space



***Note:** Table 4-1 indicates which Ports are enabled/used, and when, based upon the STRAP\_STNx\_PORTCFGx input states and/or serial EEPROM programming.*

### 4.3.3 NT Bridge – NT Mode

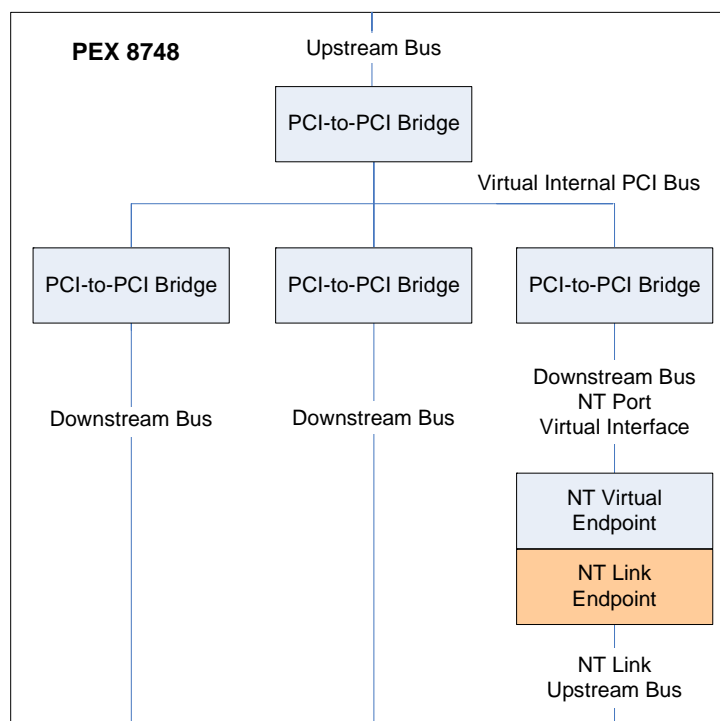
An NT bridge (NTB) is comprised of two back-to-back Type 0 endpoints. One of the endpoints is “found” by the Host that is enumerating the PCI Express Link – that is the *Link side endpoint*, or *NT Port Link Interface*. The other endpoint is “found” by the Host enumerating the internal virtual PCI Bus – that is the *Virtual side endpoint*, or *NT Port Virtual Interface*.

The NTB connects two different Host domains. Each NT endpoint has up to four Base Address registers, **BAR2** through **BAR5**, that can point to a window within the other Host’s Address space.

The PEX 8748’s NT endpoints show up to software as if they are separate devices Downstream of a PCI-to-PCI bridge, as illustrated in Figure 4-5.

For further details, refer to [Chapter 14, “Non-Transparent Bridging – NT Mode.”](#)

**Figure 4-5. NT Bridge – NT Mode in Base Mode**

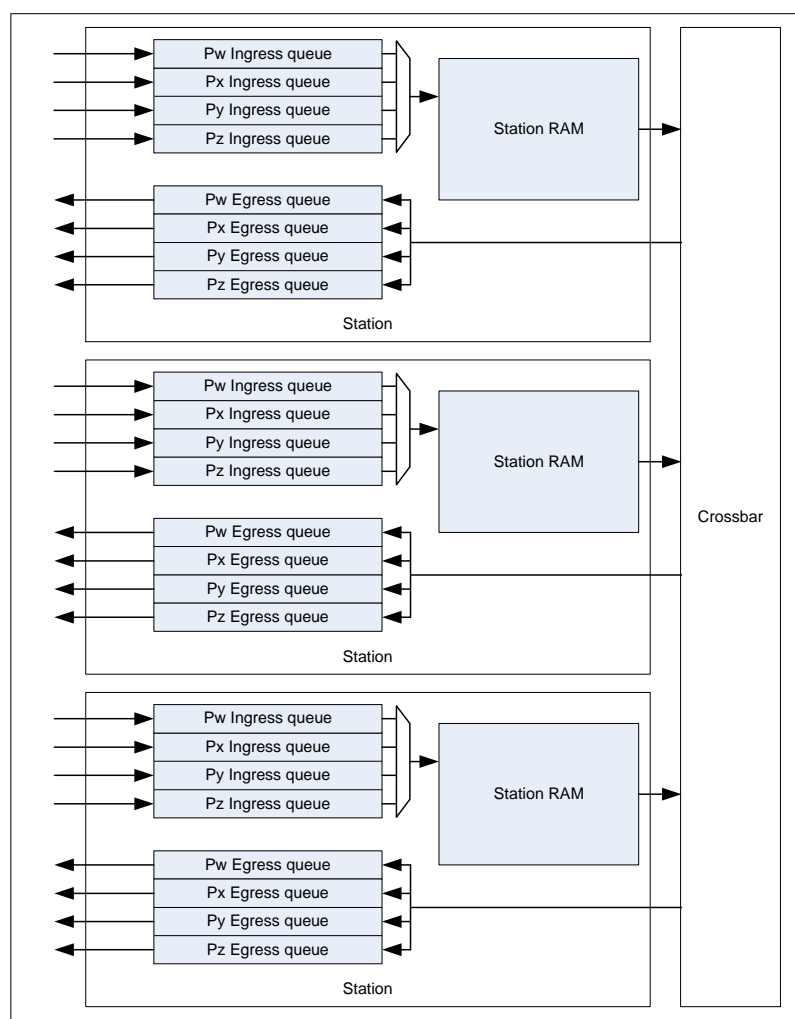


## 4.4 Hardware Architecture

The PEX 8748 has 48 PCI Express Lanes, implemented as 16 Lanes across three Stations (refer to [Section 4.4.1.1](#)). The Stations are connected to one another by the internal non-blocking fabric. [Figure 4-6](#) illustrates a block diagram of the PEX 8748.

Each Station can have from one to four Ports associated with it. The quantity of enabled Ports, and Link widths associated with those Ports, are configurable, on a Station-by-Station basis. Each Port has its own Ingress and Egress queue. Ports within a Station share the same Station RAM, but are otherwise completely independent of one another. Each Port implements the *PCI Express Base r3.0* Physical, Data Link, and Transaction Layers (PHY, DLL, and TL, respectively).

**Figure 4-6. PEX 8748 Block Diagram**



**Notes:** *Px* represents the first Port Number within the Station. *Py* represents the last enabled Port Number within the Station.

[Table 4-1](#) indicates which Ports are enabled/used, and when, based upon the *STRAP\_STNx\_PORTCFGx* input states and/or serial EEPROM programming.

## 4.4.1 Station and Port Functions

### 4.4.1.1 Port Configurations

[Table 4-1](#) defines the PEX 8748 Port configurations, as well as the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes. The Lanes are assigned to each enabled Port, in sequence.

The Upstream and Downstream Ports' maximum Link widths are initially defined by the STRAP\_STNx\_PORTCFGx 3-state inputs, which can be pulled or tied High (1) to **VDD18**, Low (0) to **VSS** (Ground), or left floating (Z, not connected). The serial EEPROM option can be used to re-configure the Ports, with the options defined in [Table 4-1](#). Serial EEPROM configuration occurs following a Fundamental Reset, and overrides the configuration defined by the STRAP\_STNx\_PORTCFGx 3-state inputs at that time. Port configuration can also be changed through the I<sup>2</sup>C Slave interface, if programmed prior to the I<sup>2</sup>C/SMBus Write sequence that Sets the **Configuration Release** register *Initiate Configuration* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset **3ACh[0]**) when **STRAP\_I2C\_SMBUS\_CFG\_EN#** is strapped Low. The final Link width can be automatically negotiated down from the programmed width, through Link-width negotiation for linkup to a device with fewer Lanes. The narrowest Port on one end of the Link determines the maximum Link width. Additionally, if a connection is broken on one of the Lanes, the training sequence removes the broken Lane and negotiates to a narrower Link width. *For example*, a x16 Port can negotiate down to x8, x4, x2, or x1.

If the Port cannot train to x1 (Lane 0 is broken), the Port reverses its Lanes and attempts to retrain. *For example*, a x16 Port that cannot train to x16 attempts to negotiate down to x8, x4, x2, or x1; if x1 linkup fails, the Port reverses its Lanes and re-attempts linkup negotiation. Either the lowest Lane (Lane 0) or highest Lane (if Lanes are reversed) of the programmed Link width must connect to the other device's Lane 0.

Each Port can run independently at Gen 1, Gen 2, or Gen 3 (2.5, 5.0, or 8.0 GT/s, respectively) Link speed.

Each Station's Port configurations are independent of the other Stations' Port configurations. Ports that are not configured or enabled are invisible to software.

**Note:** *Hot Plug Port assignment is discussed in [Section 10.9](#), “Default Parallel Hot Plug Ports.” Virtual Switch Port assignment is discussed in [Section 12.1.1](#), “Default Virtual Switch Port Assignments.”*

Table 4-1. Port Configurations

# <sup>a</sup>	Port Configuration Strapping Input States STRAP_STN0_PORTCFG[1:0] <sup>b</sup>	Port Configuration Register Value Port 0, Offset 300h[2:0]	Station 0 [Lanes/SerDes]/Port			
			Port 0 <sup>c</sup>	Port 1	Port 2	Port 3
1	0Z	001b	x16 [0-15]			
2	01	010b	x8 [0-7]	x8 [8-15]		
3	Z0	011b	x8 [0-7]	x4 [8-11]	x4 [12-15]	
4	ZZ	100b	x4 [0-3]	x4 [4-7]	x4 [8-11]	x4 [12-15]
# <sup>a</sup>	Port Configuration Strapping Input States STRAP_STN1_PORTCFG[1:0] <sup>b</sup>	Port Configuration Register Value Port 0, Offset 300h[5:3]	Station 1 [Lanes/SerDes]/Port			
			Port 8 <sup>c</sup>	Port 9	Port 10	Port 11
1	0Z	001b	x16 [16-31]			
2	01	010b	x8 [16-23]	x8 [24-31]		
3	Z0	011b	x8 [16-23]	x4 [24-27]	x4 [28-31]	
4	ZZ	100b	x4 [16-19]	x4 [20-23]	x4 [24-27]	x4 [28-31]
# <sup>a</sup>	Port Configuration Strapping Input States STRAP_STN2_PORTCFG[1:0] <sup>b</sup>	Port Configuration Register Value Port 0, Offset 300h[8:6]	Station 2 [Lanes/SerDes]/Port			
			Port 16 <sup>c</sup>	Port 17	Port 18	Port 19
1	0Z	001b	x16 [32-47]			
2	01	010b	x8 [32-39]	x8 [40-47]		
3	Z0	011b	x8 [32-39]	x4 [40-43]	x4 [44-47]	
4	ZZ	100b	x4 [32-35]	x4 [36-39]	x4 [40-43]	x4 [44-47]

- a. Port Configuration Number, which is the decimal equivalent of the **Port Configuration** register Port Configuration for Station x field(s)' (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6, 5:3, and/or 2:0]) binary value.
- b. “Z” means “floating.” Z is used for the 3-state inputs that have three functions, depending upon whether the input is pulled or tied High to **VDD18**, pulled or tied Low to **VSS** (Ground), or left floating (not connected). Each one of these three input states (1, 0, or Z, respectively) results in a different function for that input.
- c. Ports 0, 8, and 16 are also “Station Ports” within their respective Stations. Station Ports are used to control the Station-specific registers.

#### 4.4.1.2 Port Numbering

Available Port Numbers that can be used by the PEX 8748 are as follows (refer to [Table 4-1](#) and [Figure 4-7](#)):

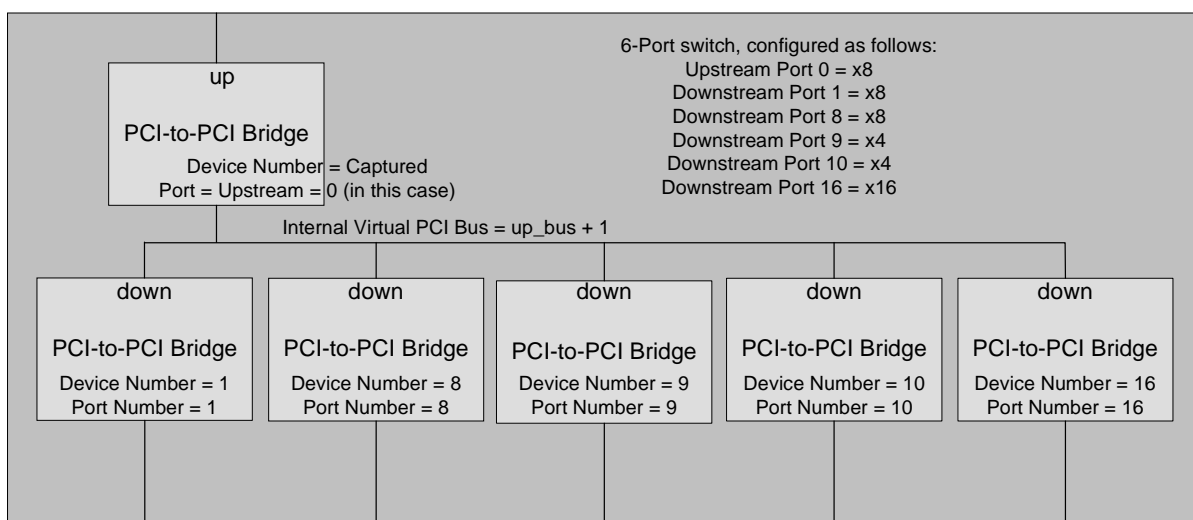
- **Station 0** – Ports 0, 1, 2, and 3
- **Station 1** – Ports 8, 9, 10, and 11
- **Station 2** – Ports 16, 17, 18, and 19

All Downstream Device Numbers match their corresponding Port Number. *For example*, if Port 9 is a Downstream Port, the PCI-to-PCI bridge associated with that Port is Device Number 9.

Any Port can be configured as, or dynamically changed to be, the Upstream Port (Port 0 is recommended in Base mode, described later in this chapter). The PCI-to-PCI bridge implemented on the Upstream Port does not assume a Device Number – it accepts the Device Number assigned by the Upstream device. Generally, the Upstream device assigns Device Number 0, according to the *PCI Express Base r3.0*.

[Figure 4-7](#) illustrates a sample Base mode configuration, with the PEX 8748 configured as a 6-Port switch with Upstream x8 on Port 0, and Downstream Ports as follows – Ports 1 and 8 (x8), Ports 9 and 10 (x4), and Port 16 (x16). The Device Numbers for the PCI-to-PCI bridges implemented on the Downstream Ports match the Port Numbers.

**Figure 4-7. Sample Port and Device Numbering for a 6-Port Configuration – Base Mode**



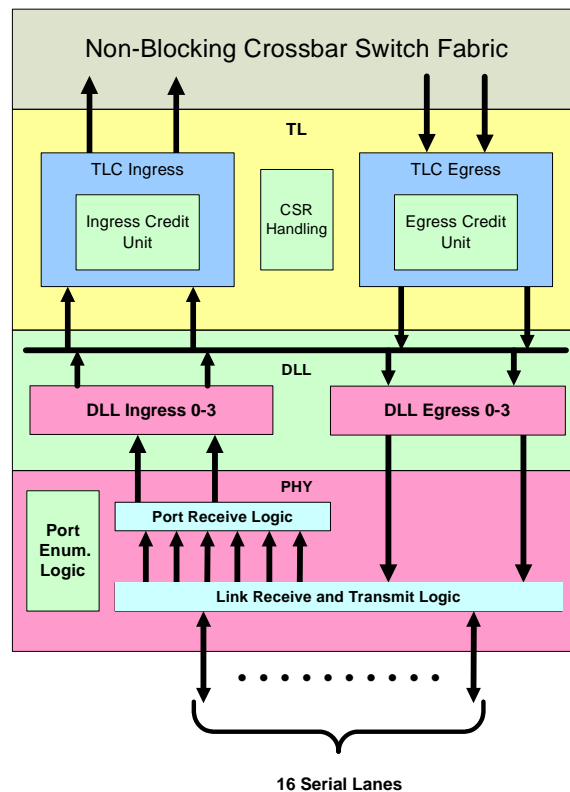
## 4.5 PCI Express Station Functional Description

The PEX 8748 groups 16 SerDes together into a Station, as listed in [Table 4-1](#). There are three Stations, comprised of one to four Ports each, depending upon the Station's Port configuration, with each Port having its own SerDes assignments.

Each Station implements the PCI Express PHY and DLL functions for each of its Ports, and aggregates traffic from these Ports onto a transaction-based, non-blocking internal crossbar fabric. The PCI Express Station also performs many TL functions, while the packet queuing and ordering aspects of this layer are handled by the Crossbar Switch Control blocks. During system initialization, software initiates Configuration Requests that set up the PCI Express interfaces, Device Numbers, and Address maps across the various Ports. These maps are used to direct traffic between Ports during standard system operation. Traffic flow between Ports within the same Station, or Ports on different Stations, is supported through the internal crossbar fabric.

At the top level, each Station has a layered organization consisting of the PHY, DLL, and TL blocks, as illustrated in [Figure 4-8](#). The PHY and DLL blocks have Port-specific data paths (one per PCI Express Port) that operate independently of one another. The Transaction Layer Control (TLC) ingress section of the TL block aggregates traffic for all ingress Ports within the Station, then sends the traffic to the internal crossbar fabric. The TLC egress section of the TL block accepts packets, by way of the internal crossbar fabric, from all ingress Ports, and schedules them to be sent out the appropriate egress Port.

**Figure 4-8. PCI Express Station Block Diagram**





## 4.6 Physical Layer

The Physical Layer (PHY) converts information received from the DLL into an appropriate serialized format and transmits it across the PCI Express Link. The PHY also receives the serialized input from the Serializer/De-Serializer (SerDes), converts it to parallel data (internal Data Bus), then writes it to the Transaction Layer Control (TLC) Ingress buffer.

The PHY includes all circuitry for PCI Express Link interface operation, including:

- Driver and input buffers
- Parallel-to-serial and serial-to-parallel conversion
- Phase-Locked Loops (PLLs) and clock circuitry
- Impedance matching circuitry
- Interface initialization and maintenance functions
- Programmable pre-emphasis
- Decision Feedback Equalization (DFE)
- Constant Time Linear Equalization (CTLE)

## 4.6.1 Physical Layer Features

- 48 SerDes, implemented across three Stations, as 16 Lanes, per Station (refer to [Section 4.4.1.1](#))
- Up to four Ports per Station
- Each Port can belong to the same or different virtual switch hierarchy, in any combination
- Multiple Upstream Port support in Virtual Switch mode – Up to six Upstream Ports, per instance
- Hardware Link training and initialization
- Hardware detection of polarity inversion
- Hardware detection of Lane reversal
- User-configurable Link widths, per Port (refer to [Table 4-1](#))
- Supported Link widths – x4, x8, or x16, depending upon Port configuration; x1 and x2 Link widths are also supported for auto-negotiation
- Supported Link speeds – 2.5, 5.0, and 8.0 GT/s
- Modified Compliance Pattern support with Receiver Error Counters, per Lane
- Hardware insertion of Sequence Number, STP, SDP, END, and EDB symbols
- Hardware Autonomous Speed Control supported
- Dynamic Link speed control supported
- Dynamic Link width supported
- Data scrambling and 8b/10b (Gen 1 and Gen 2 Link speeds) or 128b/130b (Gen 3 Link speed) encode/decode
- Receiver Error checking (Elastic buffer over/underflow, disparity and symbol encoding, Gen 3 framing)
- Modified Compliance Pattern support with Receiver Error Counters, per Lane
- Link state Power Management – supported power states are as follows:
  - L0
  - L0s
  - L1
  - L2/L3 Ready (condition before L2 or L3)
  - L3 (no Vaux)
- Upstream Port(s) can operate as Link Negotiation Master (Upstream cross-link)
- Downstream Port(s) can operate as Link Negotiation Slave (Downstream cross-link)
- Checks and removes Data Link Layer Packet (DLLP) framing symbols/tokens
- Checks and removes DLLP Link Cyclic Redundancy Check (LCRC)

## 4.6.2 Physical Layer Status and Command Registers

The PHY operating conditions are defined in:

- [Section 13.16.4, “Device-Specific Registers – Physical Layer \(Offsets 200h – 25Ch\)”](#)
- [Section 13.18.2, “Device-Specific Registers – Physical Layer \(Offsets B80h – C88h\)”](#)

The Link side Host can track the Link operating status and re-configure Link parameters, by way of these registers.

## 4.6.3 Hardware Link Interface Configuration

The PHY can include up to 16 integrated SerDes modules, on each Station. The SerDes modules are distributed among four SerDes quads (Quads 0, 1, 2, and 3) and provide the PCI Express hardware interface Lanes. The SerDes modules also provide all physical communication controls and functions required by the *PCI Express Base r3.0*, as well as the Links (clustered into Ports) that connect the PEX 8748 to other PCI Express devices.

The quantity of Ports, quantity of Lanes per Port, and SerDes connected to those Ports, are configurable, as defined in [Table 4-1](#). Initial Port configuration is determined by Strapped inputs, serial EEPROM, or auto Link-width negotiation. After the Ports are configured using the auto-negotiation process, the Link widths can narrow or widen (by combining multiple adjacent Lanes within the Station).

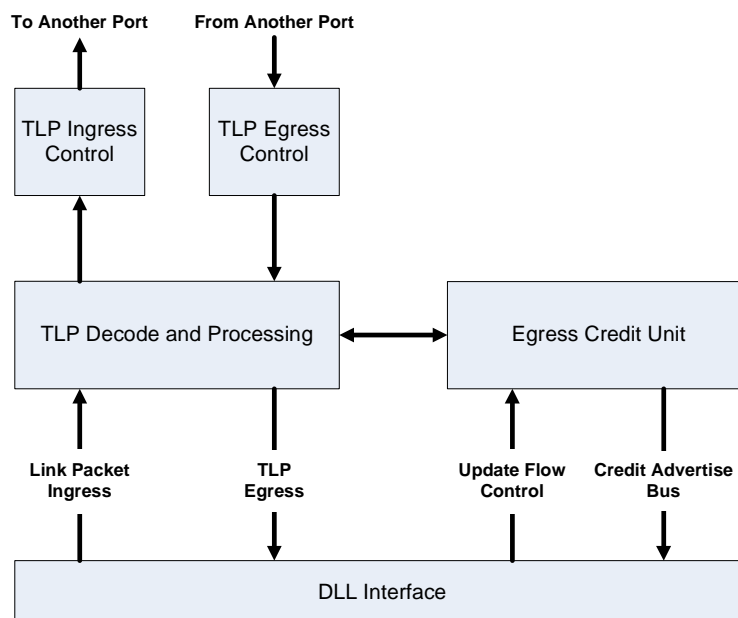
## 4.7 Transaction Layer

Transaction Layer (TL) functions include:

- Decoding and checking rules for the incoming TLP
- Memory-Mapped CSR access
- Checking incoming packets for malformed or unsupported packets
- Data Poisoning and end-to-end data integrity detection
- End-to-end Cyclic Redundancy Check (ECRC) of incoming packets
- Error logging and reporting for incoming packets
- TLP dispatching
- Write control to the packet RAM and packet Link List RAM
- Destination lookup and TC-VC mapping
- Credit-based scheduling
- Pipelined full Split Transaction protocol
- PCI/PCI-X-compatible ordering
- Interrupt handling (INTx or Message Signaled Interrupts (MSIs))
- Power Management (PM) support
- Hot Plug and PCI Express Hot Plug event support
- Link State event support
- QoS support
- Ordering
- Ingress and Egress credit management

The hardware functions provided by the PEX 8748 to implement the *PCI Express Base r3.0* TL requirements are illustrated in [Figure 4-9](#). The blocks provide a combination of ingress and egress control, as well as the data management at each stage within the flow sequence.

**Figure 4-9. TL Controller**



**NDA CONFIDENTIAL -- Inventec | King Michael**

### 4.7.1 Locked Transactions

The PEX 8748 forwards Locked transactions that are destined across the switch (that is, the PEX 8748 cannot be a target of a Locked Request; however, it can pass along the Request). Locked transactions that target PEX 8748 Memory space are terminated with an Unsupported Request (UR) error.

While the PEX 8748 understands Locked transactions, it does not lock the resources. This is consistent with limitations for Locked transaction use, as outlined in the *PCI r3.0* (Appendix F, “Exclusive Accesses”), and prevents potential deadlock, as well as serious performance degradation, that could occur with Locked transaction use.

### 4.7.2 Relaxed Ordering

The PEX 8748 supports Relaxed Ordering for Completions. By default, if the RO attribute is Set within a Completion, that Completion can bypass Posted transactions, if Posted TLPs are blocked at the egress Port (due to insufficient Posted credits from the connected device). This behavior can be disabled, by Setting the **Station-Based Control** register *No Special Treatment for Relaxed Ordering Traffic* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 760h[29]), in each Station.

### 4.7.3 TL Transmit/Egress Protocol – End-to-End Cyclic Redundancy Check

*End-to-End Cyclic Redundancy Check (ECRC)* is an optional 32-bit field appended to the end of the outgoing packet. ECRC is calculated over the entire packet, starting with the Header and including the Data Payload, except for the *EP* bit and bit 0 of the *Type* field, which are always considered to be a value of 1 for ECRC calculations. The *ECRC* field is transmitted, unchanged, as it moves through the fabric to the Completer device. The PEX 8748 checks the ECRC on all incoming TLPs, if enabled (**Advanced Error Capabilities and Control** register *ECRC Check Enable* bit (offset FCCh[8]) is Set, in each Port), and can optionally report detected errors. (When the ECRC is detected, the **Uncorrectable Error Status** register *ECRC Error Status* bit (offset FB8h[19]) can be used to log ECRC errors.)

Additionally, the PEX 8748 can optionally append ECRC to the end of internally generated TLPs, *such as* Interrupt and Error Messages, if enabled (**Advanced Error Capabilities and Control** register *ECRC Generation Enable* bit (offset FCCh[6]) is Set, in each Port).

#### 4.7.4 TL Receive/Ingress Protocol

The ingress side TL collects and stores inbound TLP traffic in the packet RAM. The incoming TLP is checked for LCRC error, valid type field, length matching the Header *Transfer Size* field, and other TLP-specific errors defined by the *PCI Express Base r3.0*.

Header and Data Payload information is forwarded to the Source Scheduler, to be routed across the internal fabric, to the egress Port. When ECRC errors are detected, the packet is discarded.

#### 4.7.5 Flow Control Credit Initialization

The initial quantity of VC0 Flow Control (FC) credits is advertised as programmed for each type of Header and Payload. After VC0 FC initialization is complete, the FC credits received are transferred to the TL egress. The TL ingress must schedule an UpdateFC Data Link Layer Packet (DLLP) for transmission, to increase the quantity of advertised credits.

#### 4.7.6 Flow Control Protocol

The PEX 8748 implements FC protocol that ensures that the switch:

- Does not transmit a TLP over a Link to a remote Receiver, unless the receiving device has sufficient VC Buffer space to accommodate the packet
- Generates FC credit updates to the remote Transmitter, to replace credits used to send TLPs to the PEX 8748

This FC is automatically managed by the hardware, and is transparent to software. Software is used only to enable additional Buffer space, to supplement the initial default buffer assignment.

The initial default FC credits, which are enabled after Link training, allow TLP traffic immediately after Link training completes. The Configuration transactions are the first transactions to use the default VC credits, to set up the initial device operating modes and capabilities.

The TL Ingress Credit Unit transmits DLLPs (referred to as *FC packets*) that update the FC to the remote Transmitter device, on a periodic basis. The DLLPs contain FC credit information that updates the Transmitter regarding the amount of available Buffer space in the PEX 8748.

The TL Egress Credit Unit receives DLLPs from the remote device, indicating the amount of Buffer space available in the remote Receiver. The unit uses this credit information to schedule the sending of TLPs to the remote device.



## Chapter 5 Reset and Initialization

### 5.1 Reset

This section describes the resets supported in Base mode and Virtual Switch mode.

*Reset* is a mechanism that returns a device to its initial state. Reset is propagated from Upstream to Downstream. Hardware or software mechanisms can trigger three different levels of reset – Fundamental, Hot, or Secondary Bus (each is described in the sections that follow). The re-initialized states following a reset vary, depending upon the reset type.

For details regarding reset in NT Mode, refer to [Section 14.1.5, “NT Port Reset.”](#)

#### 5.1.1 Resets – Base Mode

[Table 5-1](#) summarizes each type of reset in Base mode.

**Table 5-1. Reset Summary – Base Mode**

PCI Express Definition	Reset Source	Impact to Different Internal Components (upon De-Assertion)	Impact to Internal Registers
Fundamental Reset <ul style="list-style-type: none"> <li>Cold Reset</li> <li>Warm Reset</li> </ul>	<a href="#">PEX_PERST#</a> input assertion	<ul style="list-style-type: none"> <li>Initializes everything</li> <li>Serial EEPROM contents are loaded</li> <li>HwInit types are evaluated</li> </ul>	All registers are initialized
Hot Reset	<ul style="list-style-type: none"> <li>TS Ordered-Set <i>Hot Reset</i> bit is Set, at the Upstream Port</li> <li>Upstream Port enters the <i>DL_Down</i> state</li> </ul>	<ul style="list-style-type: none"> <li>Initializes all Station Ports</li> <li>Initializes internal credits and queues</li> <li>Selectively reloads serial EEPROM contents</li> </ul>	All registers, except: <ul style="list-style-type: none"> <li>Port Configuration registers</li> <li>All Sticky bits not affected by Hot Reset (HwInit, ROS, RW1CS, RWS)</li> </ul>
Secondary Bus Reset	Downstream Port's <a href="#">Bridge Control</a> register <a href="#">Secondary Bus Reset</a> bit (offset 3Ch[22]) is Set	<ul style="list-style-type: none"> <li>Downstream Port PHY generates a Hot Reset</li> <li>Downstream Port Data Link Layer (DLL) is down</li> <li>Downstream Port Transaction Layer (TL) is initialized, exhibits <i>DL_Down</i> behavior, and TLP Requests to that Port are dropped</li> <li>Upstream Port and Downstream Ports drain traffic, corresponding to the <i>DL_Down</i> condition on the Downstream Port, and initialize credits corresponding to that Downstream Port</li> </ul>	Does not affect registers (other than to initialize credits)
	Upstream Port's <a href="#">Bridge Control</a> register <a href="#">Secondary Bus Reset</a> bit (offset 3Ch[22]) is Set	<ul style="list-style-type: none"> <li>All Downstream Ports propagate a Hot Reset</li> <li>DLL of each Downstream Port is down</li> <li>TL of each Downstream Port is initialized, exhibits <i>DL_Down</i> behavior, and drops TLP Requests that target Downstream Ports</li> <li>Upstream Port TL exhibits <i>DL_Up</i> behavior</li> </ul>	Initializes Downstream Ports registers to default values

### 5.1.1.1 Fundamental Reset – Base Mode

Fundamental Reset is a hardware mechanism defined by the *PCI Express Base r3.0*, Section 6.6. Fundamental Reset input, through the `PEX_PERST#` signal, resets all Port states and Configuration registers to default conditions.

### 5.1.1.2 Hot Reset – Base Mode

Hot Reset is an in-band Reset that propagates from an Upstream PCI Express Link to all its Transparent Downstream Ports, through the Physical Layer (PHY) mechanism. The PHY mechanism communicates a reset to Downstream devices through a training sequence (TS1/TS2 Ordered-Set, in which the *Hot Reset* Training Control Bit is Set). Hot Reset is also referred to as a *Soft Reset*.

A Hot Reset initializes all Ports, resets registers that are not defined as Sticky, and resets the serial EEPROM logic to reload registers from serial EEPROM, if present. Hot Reset does not reset the Clock logic, and can be caused by any of the following:

- Upstream Port PHY receives two consecutive TS1 Ordered-Sets in which the *Hot Reset* Training Control bit is Set. Hot Reset is generated from an Upstream device, *such as* by Setting its **Bridge Control** register *Secondary Bus Reset* bit (offset 3Ch[22]).
- Upstream Port unexpectedly enters the *DL\_Down* state.  
**Exception** – If the Upstream Port Link is in the L2 Link PM state and the Link goes down, the Downstream Ports do *not* generate Hot Reset.
- Upstream Port PHY enters either the *Loopback* or *Disabled* state, upon receiving two consecutive TS1 or TS2 Ordered-Sets in which either the *Loopback* or *Disable Link* Training Control bit is Set, respectively. An Upstream device can generate the *Disable Link* sequence, by Setting its **Link Control** register *Link Disable* bit (offset 78h[4]).

### 5.1.1.3 Secondary Bus Reset – Base Mode

Any virtual Upstream or Downstream PCI-to-PCI bridge within the PEX 8748 can reset its Downstream hierarchy, by Setting the **Bridge Control** register *Secondary Bus Reset* bit (offset 3Ch[22]).

When the *Secondary Bus Reset* bit is Set on the Upstream Port, all the Downstream Ports are initialized to their default states, as defined by the *PCI Express Base r3.0*. Each of the Transparent Downstream Ports generates an in-band Hot Reset onto its Downstream Links (the NT Port Link Interface does *not* generate Hot Reset). In addition, writable registers defined by the *PCI Express Base r3.0*, in all Downstream Ports, are initialized to default values (Upstream Port registers are not reset, and the serial EEPROM does not reload registers).

When the *Secondary Bus Reset* bit is Set on a Downstream Port, that Port is reset to its default state as defined by the *PCI Express Base r3.0*, and generates an in-band Hot Reset onto its Downstream Link. The registers of that Downstream Port are not affected.

### 5.1.1.4 Register Bits that Affect Hot Reset – Base Mode

Setting the **Bridge Control** register *Secondary Bus Reset* bit (offset 3Ch[22]) generates a Hot Reset to Downstream Ports and Downstream devices.



## 5.1.2 Resets – Virtual Switch Mode

The *PCI Express Base r3.0* discusses a Conventional Reset (Cold, Warm, and Hot). For Virtual Switch mode, the above cases are slightly extended (from how they function in Base mode), as discussed in the sections that follow.

### 5.1.2.1 Conventional Reset – Virtual Switch Mode

#### **PEX\_PERST# (Cold and Warm Reset)**

The **PEX\_PERST#** input is used as the Fundamental Reset for the entire PEX 8748. This reset affects the virtual switches, and impacts all on-chip components – Stations and their Ports, Serial EEPROM Controller, Clock logic, so forth. All registers and states are initialized.

Use of the **PEX\_PERST#** input is the only way to Clear Fatal errors detected in the PEX 8748.

After **PEX\_PERST#** de-assertion, the PEX 8748 can be initialized by way of serial EEPROM, I<sup>2</sup>C, and/or the Management Port.

### 5.1.2.2 VSx\_PERST# (Hot Reset)

There is a virtual Fundamental Reset, per virtual switch. This input attempts to mimic the **PEX\_PERST#** input, but on a virtual-switch basis, and is controlled by the **Port Reset** register *Reset Port x Vector* bit(s) (Port 0, accessible through the Management Port, offset 3A0h[19:16, 11:8, 3:0]).

The **VSx\_PERST#** inputs reset all PCI-to-PCI bridges in the virtual switch hierarchy, from the Upstream PCI-to-PCI bridge, down to the Downstream PCI-to-PCI bridges and Downstream Ports owned by the virtual switch that caused **VSx\_PERST#** to assert.

Because some of the PEX 8748 data structures might be shared across virtual switches, **VSx\_PERST#** cannot reset the entire switch to a clean state. If there are any Fatal errors in any virtual switch, a **PEX\_PERST#** assertion is required.

The **VSx\_PERST#** inputs behave the same as an inband Hot Reset on the Upstream Port. Sticky registers preserve their values, and all other registers are returned to an initial state. The serial EEPROM reloads registers only for the corresponding virtual switch, unless that virtual switch's **Virtual Switch Debug** register *Disable Serial EEPROM Load on Hot Reset* bit (VS Upstream Port(s), offset A30h[3]) is Set.

### 5.1.2.3 Inband Reset (TS1 Ordered-Set) or Upstream Port DL\_Down (Hot Reset) – Virtual Switch Mode

The *PCI Express Base r3.0* defines an inband reset (with a TS1 Ordered-Set) or the Upstream Port going down as a Hot Reset. These work exactly as indicated by the *PCI Express Base r3.0*. The reset is, by default, propagated Downstream to all Ports in the virtual switch whose Upstream Port receives the Hot Reset.

#### 5.1.2.4 Secondary Bus Reset (Soft Reset) – Virtual Switch Mode

Every PCI-to-PCI bridge has a *Secondary Bus Reset* bit that resets the entire Downstream hierarchy. The Upstream PCI-to-PCI bridge's Secondary Bus Reset resets all Downstream PCI-to-PCI bridges owned by the virtual switch. A Downstream PCI-to-PCI bridge's Secondary Bus Reset sends a Hot Reset training set across the Link.

#### 5.1.2.5 Reset Propagation Prevention – Virtual Switch Mode

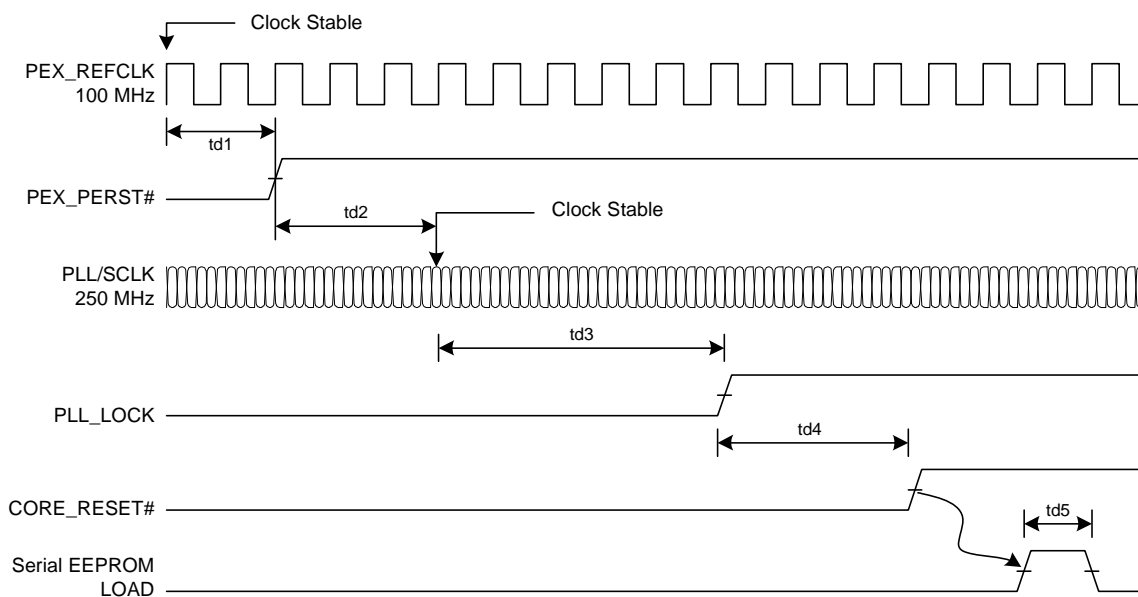
It is possible to prevent the propagation and effect of the various conventional resets (besides the Cold Reset). By Setting the virtual switch's **Virtual Switch Debug** register *Upstream Port and NT-Link Port DL\_Down Reset Propagation Disable* bit (VS Upstream Port(s), offset A30h[4]), Hot Resets are no longer propagated. The default value is that the bit is Cleared, and the resets are propagated.

### 5.1.3 Reset and Clock Initialization Timing

**Table 5-2. Reset and Clock Initialization Timing**

Symbol	Description	Typical Delay
td1	REFCLK stable to PEX_Reset release time	100 $\mu$ s
td2	PEX_Reset release to Reset de-bounce	1.32 ms
td3	Reset de-bounce to Phase-Locked Loop (PLL) Lock	105 $\mu$ s
td4	Reset de-bounce to Core Reset release	2.63 ms
td5	Serial EEPROM load time with no serial EEPROM present	17 $\mu$ s

**Figure 5-1. Reset and Clock Initialization Timing**



## 5.2 Initialization – Base Mode

The PEX 8748 initialization process starts upon exit from a Fundamental Reset. The serial EEPROM and/or I<sup>2</sup>C can be used to program initial register values, prior to BIOS/OS enumeration. If the [STRAP\\_I2C\\_SMBUS\\_CFG\\_EN#](#) input is Low, linkup of all Ports is delayed, until I<sup>2</sup>C software Sets the [Configuration Release](#) register *Initiate Configuration* bit (Port 0, offset 3ACh[0]).

Serial EEPROM download operates much faster than I<sup>2</sup>C access. I<sup>2</sup>C is relatively slow. Consequently, I<sup>2</sup>C initialization might not complete prior to the first BIOS/OS Configuration access, unless the system is designed to delay BIOS/OS Configuration access until the PCI Express subsystem is ready. If I<sup>2</sup>C is used to program the initial register values, and the PHY or DLL register values that affect SerDes parameters or Link initialization need to be changed, those registers must be programmed by serial EEPROM, so that the values are loaded prior to initial Link training.

### 5.2.1 Serial EEPROM Load Time

Serial EEPROM initialization loads only the Configuration register data that is specifically programmed into the serial EEPROM. Registers that are not included in the serial EEPROM data are initialized to default register values.

Each register entry in the serial EEPROM consists of two Address bytes and four Data bytes (refer to [Section 6.4, “Serial EEPROM Data Format”](#)); therefore, each register entry (6 bytes, or 48 bits) requires 48 serial EEPROM clocks to download. Thus, at the serial EEPROM clock default frequency of 1 MHz, after initial overhead to read the [Serial EEPROM Status](#) register (Port 0, offset 260h) (16 serial EEPROM clocks, or 16  $\mu$ s), plus another 40 serial EEPROM clocks (40  $\mu$ s) to begin reading the register data, each register entry in the serial EEPROM requires 48  $\mu$ s to download. A serial EEPROM containing 50 register entries (typical configuration, assuming the serial EEPROM is programmed only with non-default register values) and clocked at 1 MHz, takes approximately 5.2 ms to load ((16 + 40 + 48) x 50  $\mu$ s = 5,200  $\mu$ s).

To reduce the serial EEPROM initialization time, the first register entry in the serial EEPROM can increase the clock frequency, by programming the [Serial EEPROM Clock Frequency](#) register (Port 0, offset 268h), to a value of 2h (5 MHz), or 3h (9.62 MHz), if the serial EEPROM supports the higher frequency at the serial EEPROM supply voltage (typically 1.8 to 2.5V). At 5-MHz clocking, the serial EEPROM load time for 50 register entries can be reduced to approximately 575  $\mu$ s. Because the *PCI Express Base r3.0* allows a 20-ms budget for system hardware initialization, the default 1-MHz serial EEPROM clock is often sufficient when the quantity of Ports and registers programmed by serial EEPROM is relatively small.

If the NT Port Expansion ROM (stored within the serial EEPROM) is used, the serial EEPROM clock frequency ([EE\\_SK](#)) must be at least 5 MHz. For further details, refer to [Chapter 6, “Serial EEPROM Controller.”](#)

### 5.2.2 I<sup>2</sup>C Load Time

Initialization using I<sup>2</sup>C is slower than serial EEPROM initialization, because the I<sup>2</sup>C Slave interface operates at a lower clock frequency (100 KHz maximum) and the quantity of bits per Register access is increased (because the Device address is included in the bit stream). Writing one register using 100-KHz clocking takes approximately 830  $\mu$ s (83 clock periods).

For further details, refer to [Section 7.2, “I2C Slave Interface.”](#)

## 5.3 Initialization – Virtual Switch Mode

In addition to the Strapping balls, there are several ways to initialize the PEX 8748 when it is in Virtual Switch mode, by way of serial EEPROM, I<sup>2</sup>C/SMBus, and/or the Management Port. One of these three agents must initialize the Virtual Switch Table. The serial EEPROM and I<sup>2</sup>C/SMBus can effectively program all registers, whereas Management Port software generally cannot modify Read-Only (RO) registers. The serial EEPROM (programmable by I<sup>2</sup>C/SMBus and/or software) requires the least software support.

The PEX 8748 sequences these methods, in the following order:

1. Serial EEPROM
2. I<sup>2</sup>C/SMBus
3. Management Port

**Note:** *It is possible to use more than one method to initialize the PEX 8748.*

*If using a serial EEPROM and/or I<sup>2</sup>C, it programs the necessary Virtual Switch registers. If using a Virtual Switch Management Port, enabled by STRAP\_MGMT\_PORT\_EN=0 or Z, the Management Port Sets the required Virtual Switch registers. If STRAP\_MGMT\_PORT\_EN=Z to initially allow only the Management Port to link up, the Virtual Switch Management Port releases the remainder of the PEX 8748, by Setting the Configuration Release register Initiate Configuration bit (Port 0, accessible through the Management Port, offset 3ACh[0]).*

Strapping inputs are used to determine whether Virtual Switch mode uses I<sup>2</sup>C, or the Management Port, to initialize:

- If STRAP\_I2C\_SMBUS\_CFG\_EN#=L, I<sup>2</sup>C is used to configure the PEX 8748. When I<sup>2</sup>C has finished, it must write to the Initiate Configuration bit, to release the hold that is preventing linkup.
- If STRAP\_I2C\_SMBUS\_CFG\_EN#=H, I<sup>2</sup>C is not used to configure the PEX 8748.
- The STRAP\_MGMT\_PORT\_EN 3-state input is used to control two bringup options:
  - If STRAP\_MGMT\_PORT\_EN=0 after the serial EEPROM is loaded, all Ports come up at the same time.
  - If STRAP\_MGMT\_PORT\_EN=Z (after the serial EEPROM is loaded (or does not exist)), the Management Port comes up first to configure the PEX 8748. When the Management Port has completed its configuration, it must Set the Initiate Configuration bit, to release the hold that is preventing linkup.

**Note:** *Because the Virtual Switch Table registers (refer to Section 5.3.3.1) exist in Port 0, Port 0 must not be held in Reset during setup of Virtual Switch mode. Therefore, for whichever virtual switch includes Port 0, its VSx\_PERST# input must be de-asserted (High). Assigning Port 0 to be the Management Port might provide the simplest solution toward meeting this requirement.*

The Active Management Port is defined by the Management Port Control register (Port 0, accessible through the Management Port and Redundant Management Port, offset 354h). This register also defines an optional Redundant Management Port that can be used to back up the Active Management Port.

The RWS Setting in the **Management Port Control** register indicates that the bits are readable, writable, and sticky.

**Note:** *Sticky means that, as long as power is not removed, the bits are not changed with Hot Reset.*

*Note however, that if a Hot Reset causes a serial EEPROM and/or I<sup>2</sup>C load, these bits can be overwritten from their pre-reset value.*

The Active Management Port can read or write all the Virtual Switch Table registers. (Refer to [Section 5.3.3.1.](#)) No other Ports are allowed to do so.

The quantity of enabled virtual switches is reflected in the **Virtual Switch Enable** register *VSx Enable* bits (Port 0, accessible through the Management Port, offset [358h](#)). This register is initially configured by the [STRAP\\_TESTMODE\[2:0\]](#) and [VS\\_MODE](#) 3-state inputs.

The Upstream Port for each virtual switch can also be Set by the Management Port, using the **VSx Upstream** register (Port 0, accessible through the Management Port, offsets [360h](#) through [374h](#)), per virtual switch.

The Management Port configures which Downstream Port is owned by which Upstream Port, by writing a **VSx Port Vector** register (Port 0, accessible through the Management Port, offsets [380h](#) through [394h](#)), per virtual switch. Each Downstream Port can be owned by a single Upstream Port only. If software sets two or more owners for a Downstream Port, behavior is not deterministic. The NT endpoints are considered to be separate Ports for this virtual switch ownership.

After the PEX 8748 Ports ownership has been defined, the **Configuration Release** register *Initiate Configuration* bit (Port 0, accessible through the Management Port, offset [3ACh\[0\]](#)) is Set, allowing the Links to come up.

There can be a Redundant Management Port as well. The Redundant Management Port can promote itself to be the new Management Port, if the Management Port Host fails. The implementation of Redundant Management Port promotion to Management Port is application-specific.

After the *Initiate Configuration* bit is Set, each virtual switch Host enumerates its hierarchy, as if it had its own independent switch. Each hierarchy is independent and there is no order in which the hierarchies must be initialized. Each Host finds only the PCI-to-PCI bridges that it owns. Each Host has its own bus numbering scheme, which applies to the Upstream Port, through to the virtual internal PCI Bus and down to any Downstream buses. The same is true for memory addressing – each virtual switch has a completely independent Address map, both in 32- and 64-bit space.

### 5.3.1 Management Port Policies

All Device-Specific registers can be accessed from the Active Management Port, serial EEPROM, and/or I<sup>2</sup>C.

The Redundant Management Port can access only the **Management Port Control** register (Port 0, accessible through the Management Port and Redundant Management Port, offset [354h](#)). Each Upstream Port can access the PCI-SIG-defined registers of each Port, within its own virtual switch.

## 5.3.2 Active and Redundant Management Ports

The **Management Port Control** register *Active Management Port* field (Port 0, accessible through the Management Port and Redundant Management Port, offset 354h[4:0]) defines which Virtual Switch Upstream Port is assigned as the Active Management Port. In Virtual Switch mode, there is no Management Port when the following conditions are met:

- STRAP\_MGMT\_PORT\_EN 3-state input is pulled or tied High (1), and
- **Management Port Control** register *Active Management Port Enable* bit (Port 0, accessible through the Management Port and Redundant Management Port, offset 354h[5]) is Cleared, and
- **Configuration Release** register *Initiate Configuration* bit (Port 0, accessible through the Management Port, offset 3ACh[0]) is Set.

The **Management Port Control** register *Redundant Management Port* field (field [12:8]) defines which Virtual Switch Upstream Port is assigned as the Redundant Management Port. The Redundant Management Port provides a Failover capability, should the Management Port Host fail. Software can demote the Management Port and promote the Redundant Management Port to be the new Management Port, by programming the **Management Port Control** register. (Refer to [Section 5.3.6.5](#).)

This register can be accessed by the Management-capable (Active and Redundant) Ports, Strapping balls, and/or the I<sup>2</sup>C Bus. The **Reserved** register bits return zeros (0) during Reads. Writes to **Reserved** register bits do not affect the register.

The register's RWS fields/bits are represented in [Table 5-3](#). (For complete details, refer to the register offset 354h description provided in [Section 13.16.8](#), "Device-Specific Registers – Port Configuration (Offsets 354h – 46Ch).")

**Table 5-3. Management Port Control Register (Port 0, accessible through the Management Port and Redundant Management Port, Offset 354h)**

Bit(s) <sup>a</sup>	Description
4:0	<b>Active Management Port</b> Indicates the Port Number of the Active Management Port. The value of this field is latched in, upon reset de-assertion, from the STRAP_UPSTRM_PORTSEL[2:0] input states.
5	<b>Active Management Port Enable</b> Enables the Active Management Port. The value of this bit is latched in, upon reset de-assertion, from the STRAP_MGMT_PORT_EN input state.  0 = Management Port is disabled (STRAP_MGMT_PORT_EN=1) 1 = Management Port is enabled (STRAP_MGMT_PORT_EN=0 or Z)
6	<b>Active Management Port EEPROM Load on Hot Reset for Chip and Station Registers Enable</b> <i>Valid only for the Management Port.</i>  After the Management Port receives a Hot Reset or DL_Down condition, the serial EEPROM reloads registers, as described below.  0 = Serial EEPROM reloads Management Port Port-specific registers (default) 1 = Serial EEPROM reloads: <ul style="list-style-type: none"> <li>• Chip-specific registers (might affect all virtual switches),</li> <li>• Station-specific registers (might affect all virtual switches), and,</li> <li>• Management Port Port-specific registers</li> </ul>
12:8	<b>Redundant Management Port</b> Indicates the Port Number of the Redundant Management Port.
13	<b>Redundant Management Port Enable</b> Enables the Redundant Management Port.

a. Bits not identified in [Table 5-3](#) are **Reserved** or **Factory Test Only**.

NDA CONFIDENTIAL -- Inventec | King Michael

### 5.3.3 Virtual Switch Table

The PEX 8748 supports up to six virtual switches. The Virtual Switch Table defines the Port Numbers and associated Virtual Switch Numbers and Upstream Port of each virtual switch. Each table entry must be updated as a single atomic operation.

Additional information related to the table is discussed in the sections that follow.

**Note:** In Virtual Switch mode, the Virtual Switch Table registers include the **VSx Port Vector** and **VSx Upstream** registers (Port 0, accessible through the Management Port, offsets 380h through 394h, and 360h through 374h, respectively). These two sets of registers define which Ports are associated to each virtual switch, and which Port is the Upstream Port for each Virtual Switch, respectively. These registers must be initialized by one (or more) of the following agents:

- Serial EEPROM.
- I<sup>2</sup>C/SMBus, provided that the **STRAP\_I2C\_SMBUS\_CFG\_EN#** input is Low (to delay linkup until I<sup>2</sup>C/SMBus Sets the **Configuration Release** register **Initiate Configuration** bit (Port 0, accessible through the Management Port, offset 3ACh[0]). This option might require software support, to delay Host enumeration until I<sup>2</sup>C/SMBus Sets the **Initiate Configuration** bit after programming the PEX 8748 Configuration registers.
- Management Port software, provided that **STRAP\_MGMT\_PORT\_EN** is left floating (to enable the Management Port, and delay linkup of all other Ports until software and/or I<sup>2</sup>C/SMBus Sets the **Initiate Configuration** bit).

Use of serial EEPROM for the initialization might be the best choice for most applications, because it is the simplest solution. Therefore, for Virtual Switch mode applications, the serial EEPROM is required to initialize the Virtual Switch Table registers, unless I<sup>2</sup>C/SMBus and/or Management Port software can perform this task.



### 5.3.3.1 Virtual Switch Table Registers

The Virtual Switch Table consists of three registers, listed in Table 5-4. The first two can be accessed in any order, and the third (**Virtual Switch Enable**) must be accessed last.

**Table 5-4. Virtual Switch Table Registers<sup>a</sup>**

Offset(s)	Register	Description
358h[5:0]	<b>Virtual Switch Enable</b>	<p>The register's <i>VSx Enable</i> bits are used to enable or disable virtual switches within the system. There is one bit, per virtual switch (VS0 through VS5). Setting a bit enables the corresponding virtual switch. Clearing a bit disables the corresponding virtual switch.</p> <p>If a <i>VSx Enable</i> bit is Set, another virtual switch is being established with the new set of Upstream and Downstream Ports. One Write to this register can disable the previous virtual switch, and enable a new virtual switch. This ensures that Virtual Switch Enable and Disable can be implemented with a single Write to a register.</p> <p>The STRAP_TESTMODE[2:0] and VS_MODE 3-state inputs map to this register.</p>
360h – 374h	<b>VSx Upstream</b>	<p>These registers define the Upstream Port of each virtual switch. There is one register, per virtual switch (VS0 through VS5).</p> <p>The registers' <i>VSx Upstream Port</i> bits [4:0] define which Port is the singular Upstream Port, within the corresponding virtual switch. A virtual switch must include a single, unique Upstream Port.</p> <p>The STRAP_UPSTRM_PORTSEL[2:0] and STRAP_NT_UPSTRM_PORTSEL[2:0] 3-state inputs map to these registers.</p>
380h – 394h	<b>VSx Port Vector</b>	<p>These registers define the Upstream and Downstream Ports associated with each virtual switch. There is one register, per virtual switch (VS0 through VS5), and each register has one bit, per Port.</p> <p>Any Port can be assigned to a virtual switch. Setting a Port's bit in a <b>VSx Port Vector</b> register assigns that Port to the virtual switch associated with the register.</p> <p><b>A single Port can be assigned to only one virtual switch at any time.</b> Therefore, each bit (in the range listed above) must be Set in only one of the six <b>VSx Port Vector</b> registers, at any time. A Downstream Port can be re-assigned to a different virtual switch, by Clearing the Port's bit in one <b>VSx Port Vector</b> register, and Setting the same bit in another <b>VSx Port Vector</b> register.</p> <p>The STRAP_TESTMODE[2:0] and VS_MODE 3-state inputs map to these registers.</p>

a. All registers listed in this table are located in Port 0, accessible through the Management Port.

### 5.3.3.2 Virtual Switch Table Programming Sequence

The following describes the programming sequence for the Virtual Switch Table, by the Management Port Host.

1. Strap the **VS\_MODE** and **STRAP\_TESTMODE[2:0]** 3-state inputs to enable Virtual Switch mode and the quantity of virtual switches needed.

**Note:** When used together, the **VS\_MODE** and **STRAP\_TESTMODE[2:0]** 3-state inputs, depending upon their input states, enable up to six virtual switches. The quantity of enabled virtual switches is reflected in the **Virtual Switch Enable** register **VSx Enable** bits (Port 0, accessible through the Management Port, offset 358h[5:0]). (Refer to the **VS\_MODE** and **STRAP\_TESTMODE[2:0]** 3-state input descriptions, in Table 3-7, “Strapping Signals,” and Section 3.4.4.1, “STRAP\_TESTMODE[2:0] 3-State Input Configurations,” for details.)

2. Strap the **STRAP\_MGMT\_PORT\_EN** 3-state input to 0, to enable the Active Management Port and allow it to linkup, while delaying linkup of all other Ports. The Management Port Host must Set the **Configuration Release** register **Initiate Configuration** bit (Port 0, accessible through the Management Port, offset 3ACh[0]), to allow the remaining Ports to linkup.

**Note:** When the Active Management Port is enabled and the **VS\_MODE** and **STRAP\_TESTMODE[2:0]** 3-state inputs are strapped to enable one active virtual switch (VS0), all enabled Ports are assigned to VS0.

3. Strap the **STRAP\_UPSTRM\_PORTSEL[2:0]** 3-state inputs to the Active Management Port Number. The **VS0 Upstream** register **VS0 Upstream Port** field (Port 0, accessible through the Management Port, offset 360h[4:0]) defaults to the Port Number defined as the VS0 Upstream Port by these inputs.
4. Adjust all **VSx Upstream** and **VSx Port Vector** registers (Port 0, accessible through the Management Port, offsets 360h through 374h, and offsets 380h through 394h, respectively) to the new table values, ensuring that the following conditions are met:
  - a. The Management Port must remain in its original virtual switch.
  - b. A Port can appear in only one **VSx Port Vector** register. The exception is when the Management Port is being migrated from one virtual switch to another. If this is the case, the Management Port can appear in two **VSx Port Vector** registers – the original enabled virtual switch, and the newly disabled virtual switch.
  - c. The VS Upstream Port defined in the **VSx Upstream** register must be included in the corresponding **VSx Port Vector** register. If the Management Port is migrating to a different virtual switch, the new virtual switch’s **VSx Upstream** register must be programmed to the Management Port Number.
5. **If the Management Port is migrating to a different virtual switch** – Enable all other needed virtual switches, using the **Virtual Switch Enable** register **VSx Enable** bit(s) (Port 0, accessible through the Management Port, offset 358h[5:0], as appropriate), and disable the original virtual switches containing the Management Port, in the same Write operation. If the original virtual switch is to remain disabled, skip to step 7.

**If the Management Port is not migrating to a different virtual switch** – Enable all other needed virtual switches, using the **Virtual Switch Enable** register **VSx Enable** bit(s) (Port 0, accessible through the Management Port, offset 358h[5:0], as appropriate), and skip to step 8.

6. Remove the Management Port from the original Management Port virtual switch (**VSx Port Vector** register(s) (Port 0, accessible through the Management Port, offsets 380h through 394h)), and update the **VSx Upstream** register(s) (Port 0, accessible through the Management Port, offsets 360h through 374h) to the Management Port's new Port Number.
7. To enable all other virtual switches that need to be enabled, write to their corresponding **Virtual Switch Enable** register *VSx Enable* bit(s) (Port 0, accessible through the Management Port, offset 358h[5:0], as appropriate).
8. Set the **Configuration Release** register *Initiate Configuration* bit (Port 0, accessible through the Management Port, offset 3ACh[0]), to allow all Ports to begin the linkup process.

### 5.3.4 Port Activity Vector

The **Egress Control and Status** register *Port Activity* bit (offset F30h[31]) is implemented for each Port, and collectively, this same bit in all Ports forms the Port Activity Vector. The *Port Activity* bit is used to indicate that for this Port, there are no pending transactions inside the virtual switch. When the bit is Set, the Port's queues are empty. This bit is polled by software, to decommission and re-assign the Port to another virtual switch.

The PEX 8748 waits 20 ms after reset to the virtual switch, then indicates that the Port's queues are empty and Sets the bit. If the Port is known to be inactive, the user can write to the Virtual Switch Table, to switch over in less than 20 ms, without looking at this bit. Reset the *Port Activity* bit after writing the Virtual Switch Table entry for this Port.

The Virtual Switch Table is described in [Section 5.3.3](#).

### 5.3.5 Link-Related Registers

Link-related registers are used when a Management Port is used to manage the PEX 8748. These registers provide information, and optionally generate an interrupt to the Management Port Host when the PEX 8748's Ports change the Links' status. In standard operation, the Port's **Link Status** register (offset 78h) provides the Link status information to the respective Virtual Switch Host, and Transparent Downstream Ports can optionally generate a **Data Link Layer State Changed** interrupt when Downstream Port linkup is lost or regained. When a Management Port is connected to manage the PEX 8748, it also must be informed of the status of all Ports.

**Table 5-5. Link-Related Registers<sup>a</sup>**

Offset	Register	Description
3A0h	<b>Port Reset</b>	When driven with a value of 1, this register holds the Port in reset, including the Port PCI-to-PCI bridge and the hierarchy below it. A value of 0 indicates to un-reset the PCI-to-PCI bridge and the hierarchy below it. There is one bit, per Port.
900h	<b>Switch Link Up</b>	When the Port's Link state transitions from down to up, the Port's bit is Set. The Port's bit is Cleared, by using software to explicitly write 1 to the bit. The register has one bit, per Port.
904h	<b>Switch Link Down</b>	When the Port's Link state transitions from up to down, the Port's bit is Set. The Port's bit is Cleared, by using software to explicitly write 1 to the bit. The register has one bit, per Port.
908h	<b>Switch Link Event Mask</b>	If the <i>Mask</i> bit is Set, the corresponding <i>Up</i> or <i>Down</i> bit (located in register offsets 900h and 904h, respectively) transition does not generate an interrupt to the Management Port. If not masked, the bit transition generates an interrupt to the Active Management Port. The register has one <i>Mask</i> bit, per Port.
90Ch	<b>Switch Link Status</b>	This Read-Only register indicates Link status. The register has one <i>Status</i> bit, per Port.

*a. All registers listed in this table are located in Port 0, accessible through the Management Port.*

## 5.3.6 Reconfiguration of Virtual Switches

Virtual switches and their Ports can be re-configured for a variety of reasons, including failures or insertion or removal of cards on the Downstream or Upstream Ports.

### 5.3.6.1 Graceful De-Allocation of Downstream Port

To gracefully de-allocate a Downstream Port from a virtual switch, higher-level software ensures that new Requests are not initiated by the Port and its associated I/O endpoint. Higher-level software reads the Port activity register, and checks the Port's **Egress Control and Status** register *Port Activity* bit (offset F30h[31]), to determine whether the Port's queues are empty. All pending Requests are serviced. After the Port is in an Idle state, with no pending Requests (*Port Activity* bit is Cleared), the Port can be assigned to another virtual switch, by programming the Virtual Switch Table. (Refer to [Section 5.3.3.](#))

*The operation sequence is as follows:*

1. Management Port software writes in the **Bridge Control** register (offset 3Ch) or Hot Reset to the source virtual switch. (This can also be done by asserting VSx\_PERST# or Setting the corresponding bit.)
2. Management Port software re-programs the Virtual Switch Table, to add this Downstream Port to the new virtual switch.
3. Hot Plug Controller on the Upstream Port of the new virtual switch generates a **Presence Detect Changed** interrupt, if not masked. The Port's **Power Management Hot Plug User Configuration** register *Upstream Hot Plug Enable* bit (Upstream Port(s), offset F70h[14]) must be Set.
4. BIOS running on a Root Complex attached to the Upstream Port of the new virtual switch enumerates the devices. Whether the enumeration is partial or full, is dependent upon the software.
5. The driver is invoked on a Root Complex, and starts communicating with the endpoint connected to this Downstream Port.

*Definitions:*

- **BIOS** – Basic Input Output Software
- **Driver software** – Software running on the Host
- **Management Port software** – Software running on the Management Port
- **New virtual switch** – Location to which the Downstream Port is being re-allocated/assigned
- **Old virtual switch** – Location from which the Downstream Port is being de-allocated

### 5.3.6.2 Surprise Removal of Downstream Device

When an endpoint connected to a virtual switch Downstream Port is removed without any indication to software, the Link goes down and an interrupt is generated to the virtual switch's Host, as well as to the Management Port Host. If a Hot Plug Controller is implemented on the Downstream Port where the Surprise Down event occurs, an Uncorrectable Error Message is sent to the Upstream Host.

*The operation sequence is as follows:*

1. Management Port is connected to a Management Port.
2. Link Down interrupt is sent to the Management Port software running on the Management Port, and a **Data Link Layer State Changed** interrupt is sent to the Virtual Switch Host.
3. Host Software Clears the Port's **Slot Status** register **Data Link Layer State Changed** bit (Downstream Ports, offset 80h[24]) interrupt, and determines that the Link is Down (Port's **Link Status** register **Data Link Layer Link Active** bit (Downstream Ports, offset 78h[29]) is Cleared).
4. Host software marks the device as unreachable.
5. Interrupt Service Routine (ISR) of Management Port software reads the Device-Specific **Switch Link Up**, **Switch Link Down**, and **Switch Link Status** registers (Port 0, accessible through the Management Port, offsets 900h, 904h, and 90Ch, respectively) (Management Port's PCI-to-PCI bridge's *MSI Enable* bit is used to generate an INTx interrupt or Message Signaled Interrupt (MSI) to the Management Port).
6. ISR of Management Port software determines that the Link is Down, then Clears the interrupt.
7. ISR of Management Port software again reads the Device-Specific **Switch Link Up**, **Switch Link Down**, and **Switch Link Status** registers, and determines that the interrupt has been Cleared and the status is Link Down.
8. ISR leaves the routine, marking the Downstream Port Number as "down".
9. Management Port software reports the Downstream Port as "not in use" to the software's upper layer.
10. Based upon policy, this Downstream Port can be assigned to the Management Port, or remain with the virtual switch to which it belongs.

If there is no Management Port, the virtual switches set up by the serial EEPROM and Upstream Ports manage their own virtual hierarchies.

### 5.3.6.3 Graceful De-Allocation of Upstream Port

*When an Upstream Port must be de-commissioned for scheduled maintenance, the following actions are taken:*

1. Downstream Port's traffic is stopped, by disabling all endpoints it owns.
2. Software waits until the Upstream Port's **Egress Control and Status** register *Port Activity* bit (offset **F30h**[31]) indicates a value of 1.
3. Management Port software notifies the administrator that the virtual switch Upstream Port is idle.
4. Administrator can remove the Upstream Port device.
5. Hot Reset is sent (as a result of item 4) to all Downstream Ports that belong to this Root Port's hierarchy.
6. Based upon policy, Downstream Ports can be assigned to another Root Port's hierarchy, or remain as is for future use by the same Upstream Port.

### 5.3.6.4 Surprise Removal of Upstream Port

An Upstream Port can be removed from the Root Port's hierarchy, without notifying the Management Port software. Although this is not advisable, software and hardware ensure that the removal does not bring down shared resources, and that the system gracefully recovers from this event.

*The following actions are taken after the Upstream Port is detected as having been removed:*

1. Link goes down for this Upstream Port.
2. Management Port software is interrupted.
3. ISR reads the Device-Specific **Switch Link Up**, **Switch Link Down**, and **Switch Link Status** registers (Port 0, accessible through the Management Port, offsets **900h**, **904h**, and **90Ch**, respectively) (Management Port's PCI-to-PCI bridge's *MSI Enable* bit is used to generate an INTx interrupt or MSI to the Management Port).
4. ISR determines that the Port's **Switch Link Down** register *Port x Link Down* Interrupt bit (Port 0, accessible through the Management Port, offset **904h**[19:16, 11:8, 3:0]) is Set.
5. DL\_Down event on the Upstream Port generates a Hot Reset within its virtual hierarchy.
6. All Downstream Ports that belong to this virtual hierarchy are reset, and propagate a Hot Reset Downstream.
7. Based upon policy, Management Port software programs the Virtual Switch Table (refer to [Section 5.3.3](#)), then adds the Downstream Ports to its own virtual switch.

### 5.3.6.5 Management-Capable Port Switch Over

Typically, a system design implements a heartbeat mechanism between the Active and Redundant Management Ports. When the Redundant Management Port Host detects that the Active Management Port must be switched over, the Redundant Management Port Host writes to the **Management Port Control** register (Port 0, accessible through the Management Port and Redundant Management Port, offset **354h**), removes the existing Active Management Port, and promotes itself to be the Active Management Port Host.

THIS PAGE INTENTIONALLY LEFT BLANK.



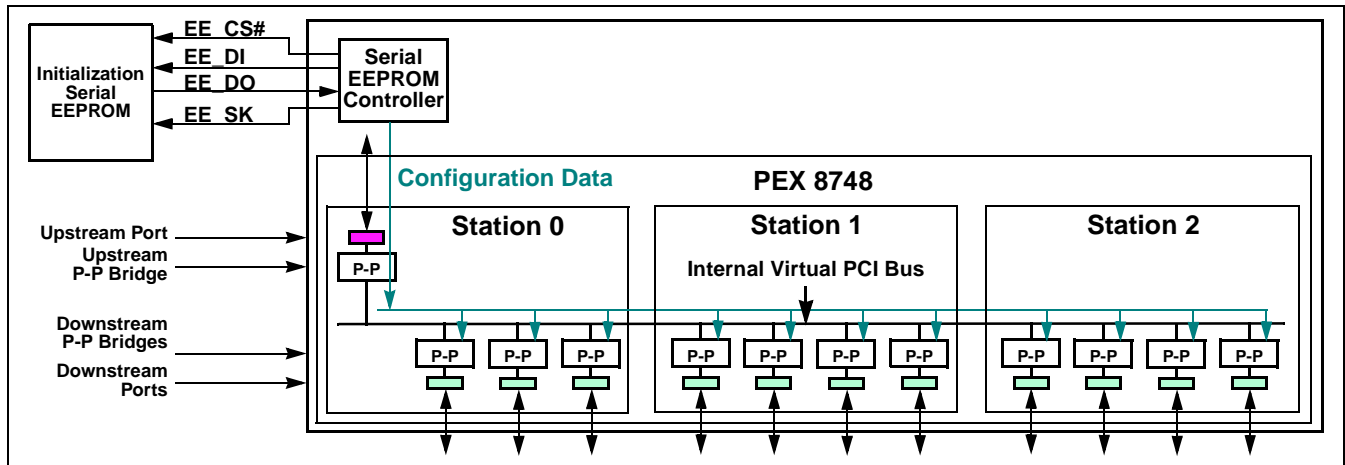
### 6.1 Overview

The PEX 8748 provides a Serial EEPROM Controller and interface to Serial Peripheral Interface (SPI)-compatible serial EEPROMs. (Refer to [Figure 6-1](#), which represents the PEX 8748 in Base mode.) This interface consists of a Chip Select, Clock, Serial Data In, and Serial Data Out signals, and operates at a programmable frequency of up to 17.86 MHz. The PEX 8748 supports serial EEPROMs that use 1-, 2-, or 3-byte addressing; the PEX 8748 automatically determines the appropriate addressing mode.

The controller provides access to non-volatile memory. This external memory can be used for three different purposes:

- The serial EEPROM can be used to store register data, for switch configuration and initialization. When a serial EEPROM device is connected to the PEX 8748, immediately after reset, the Serial EEPROM Controller reads data from the serial EEPROM that is used to update the PEX 8748 register default values.
- System or application data can be stored into, and read from, the serial EEPROM, by software, I<sup>2</sup>C and/or SMBus, initiating random-access Read or Write Requests to the serial EEPROM.
- In NT mode, the serial EEPROM can provide up to 32 KB of Expansion ROM, for the NT Port Link Interface (default) or NT Port Virtual Interface. When software reads the NT Port Expansion ROM (starting at the Expansion ROM Base Address), the PEX 8748 reads from the serial EEPROM, to return the requested ROM image.

**Figure 6-1. Serial EEPROM Connections – Base Mode**



**Note:** [Table 4-1](#) indicates which Ports are enabled/used, and when, based upon the STRAP\_STNx\_PORTCFGx input states and/or serial EEPROM programming.

## 6.2 Features

- Detection of whether a serial EEPROM is present/not present
- Supports high-speed serial EEPROMs with Serial Peripheral Interface (SPI) interface
- Non-volatile storage for register default values loaded during Power-On Reset
- Cyclic Redundancy Check (CRC) compared on each download, to ensure data integrity prior to loading registers
- 4-byte Write/Read access to the serial EEPROM, through the Upstream Port (Base mode), the Management Port (Virtual Switch mode), and/or I<sup>2</sup>C/SMBus
- Serial EEPROM data format allows for loading registers by Station/Port/Address location
- Required serial EEPROM size is dependent upon the quantity of registers being changed
- Automatic support for 1-, 2-, or 3-byte-addressable serial EEPROMs
- Manual override for quantity of serial EEPROM Address bytes
- Programmable serial EEPROM clock frequency
- Programmable serial EEPROM clock-to-chip select timings
- Supports one NT Port Expansion ROM (*not supported* for 1-byte address serial EEPROMs), for the NT Port Virtual or Link Interface

## 6.3 Serial EEPROM Load

### 6.3.1 Serial EEPROM Load – Base Mode

The Serial EEPROM Controller performs a serial EEPROM download when the following conditions are met:

- Serial EEPROM is present<sup>a</sup>, and
- Validation signature (first byte read from the serial EEPROM) value is **5Ah**, and
- One of the following events occurs:
  - **PEX\_PERST#** is returned High, following a Fundamental Reset (*such as* a Cold or Warm Reset to the entire PEX 8748)
  - Hot Reset is received at the Upstream Port (downloading upon this event can be optionally disabled, by Setting the **Virtual Switch Debug** register *Disable Serial EEPROM Load on Hot Reset* and/or *Upstream Hot Reset Control* bit (Upstream Port, offset **A30h**[3 and/or 2], respectively))
  - Upstream Port exits a *DL\_Down* state (downloading upon this event can be optionally disabled, by Setting the **Virtual Switch Debug** register *Upstream Port and NT-Link Port DL\_Down Reset Propagation Disable* and/or *Upstream Hot Reset Control* bit (Upstream Port, offset **A30h**[4 and/or 2], respectively))

---

*a. The serial EEPROM is considered to be present if it returns a non-zero value in response to an initial Read Status command of its Status register. This value is copied to the **Status Data from Serial EEPROM** register bits (Port 0, offset **260h**[31:24]). Serial EEPROM presence is reported in the register's *EepPrsnt*[1:0] field (field [17:16]); a value of 01b or 11b indicates that the serial EEPROM is present.*

### 6.3.2 Serial EEPROM Load – Virtual Switch Mode

Serial EEPROM functionality in Virtual Switch mode is identical to that in Base mode, with the following exceptions:

- Only the Management Port, I<sup>2</sup>C, and SMBus can access the serial EEPROM. A PCI Express Master can access the serial EEPROM, only if that virtual switch Upstream Port is designated as the Management Port (as reflected in the **Management Port Control** register *Active Management Port* field (Port 0, accessible through the Management Port and Redundant Management Port, offset 354h[4:0])).
- When the serial EEPROM reloads registers following a Reset, it reloads only the Ports that are associated with that virtual switch. Determination of which Ports are associated to a particular virtual switch is dependent upon the **VSx Port Vector** register (Port 0, accessible through the Management Port, offset(s) 380h through 394h) value(s), initially programmed to the configuration defined by the enabled virtual switches, as listed in Table 12-1, “Virtual Switch Default Port Assignments – Virtual Switch Mode,”
- By default, Chip- and Station-specific registers are not reloaded from serial EEPROM following a Virtual Switch Fundamental Reset (VSx\_PERST# is de-asserted), a Port's **Port Reset** register *Reset Port x Vector* bit (Port 0, accessible through the Management Port, offset 3A0h[19:16, 11:8, 3:0]) being toggled from 1 to 0, and/or a Soft Reset (Hot Reset or Upstream Port *DL\_Down* state).

A Chip- and Station-specific register reload from serial EEPROM can be enabled by the Management Port, I<sup>2</sup>C, and/or SMBus Setting the **Management Port Control** register *Active Management Port EEPROM Load on Hot Reset for Chip and Station Registers Enable* bit (Port 0, accessible through the Management Port and Redundant Management Port, offset 354h[6]).

The Serial EEPROM Controller performs a serial EEPROM download when the following conditions are met:

- Serial EEPROM is present<sup>a</sup>, and
- Validation signature (first byte read from the serial EEPROM) value is **5Ah**, and
- One of the following events occurs:
  - **PEX\_PERST#** is returned High, to exit a Fundamental Reset (*such as* a Cold or Warm Reset to the entire chip)
  - **VSx\_PERST#** is returned High, to exit a Virtual Switch Fundamental Reset (*such as* a Cold or Warm Reset to a single virtual switch)
  - **VSx\_PERST# and NT\_PERST# Status** register **VSx\_PERST# Control** bit(s) (Port 0, accessible through the Management Port, offset **3A8h**[13:8]) corresponding to one or more enabled virtual switches that are Set and then Cleared, to release the virtual switches from Fundamental Reset
  - Hot Reset is received at a virtual switch Upstream Port (downloading upon this event can be optionally disabled, by Setting the **Virtual Switch Debug** register **Disable Serial EEPROM Load on Hot Reset** bit (VS Upstream Port(s), offset **A30h**[3]))
  - Virtual switch Upstream Port exits a **DL\_Down** state (downloading upon this event can be optionally disabled, by Setting the **Virtual Switch Debug** register **Upstream Port and NT-Link Port DL\_Down Reset Propagation Disable** bit (VS Upstream Port(s), offset **A30h**[4]))

Serial EEPROM reload upon receiving a Soft Reset can also be disabled, by Setting the **Virtual Switch Debug** register **Upstream Hot Reset Control** bit (VS Upstream Port(s), offset **A30h**[2]). Setting this bit effectively converts Hot Reset severity to the lower severity of a Secondary Bus Reset.

*a. The serial EEPROM is considered to be present if it returns a non-zero value in response to an initial Read Status command of its Status register. This value is copied to the **Status Data from Serial EEPROM** register bits (Port 0, accessible through the Management Port, offset **260h**[31:24]). Serial EEPROM presence is reported in the register's **EepPrsnt[1:0]** field (field [17:16]); a value of 01b or 11b indicates that the serial EEPROM is present.*

## 6.4 Serial EEPROM Data Format

The data in the serial EEPROM is stored in the format defined in [Table 6-1](#). The [Validation Signature](#) byte is located in the first address. The Serial EEPROM Controller reads this byte to determine whether a valid serial EEPROM image exists versus a blank image. REG\_BYTE\_COUNT[15:0] contains the quantity of bytes of serial EEPROM data to be loaded. It is equal to the quantity of registers to be loaded times 6 (6 serial EEPROM bytes, per register). If the REG\_BYTE\_COUNT[15:0] value is not a multiple of 6, the last incomplete register entry is ignored.

For the remaining register-related locations, data is written into a 2-byte address that represents the Configuration register offset and Port Number, and the 4 bytes following are the data loaded for that Configuration register. Only Configuration register data specifically programmed into the serial EEPROM is loaded after the PEX 8748 exits reset.

[Table 6-2](#) defines the Configuration register Address format (REGADDR[15:0] from [Table 6-1](#)):

- Bits [9:0] represent bits [11:2] of the Register address
- Bits [15:10] represent the Port Number of the register selected to be programmed by serial EEPROM

Because the PEX 8748 Serial EEPROM Controller always accesses 4 bytes of serial EEPROM data (for DWord-aligned Register addresses), register offsets are stored in the serial EEPROM as DWord address values.

To determine the 2-byte serial EEPROM value that represents the PEX 8748 Port and register offset, shift the register offset 2 bits to the right (divide by 4), then OR the resulting value with the appropriate Port Identifier value from [Table 6-2](#).

*For example*, to load Port 16 register offset FA8h, shift the address to the right by 2 bits (this becomes 07Eh) and concatenate 1000\_00b. The resulting DWord address in the serial EEPROM will be 1000\_0000\_0111\_1110b (807Eh).

## 6.4.1 Serial EEPROM Cyclic Redundancy Check

A Cyclic Redundancy Check (CRC) value is always calculated for serial EEPROM data stored in serial EEPROM addresses 2h through BYTE COUNT +3. If the value of “REG BYTE COUNT plus 2” is not a multiple of 4, the final data is still put into a CRC calculation, with unread byte(s) of the final DWord equal to 00h. Expansion ROM code (if present in the serial EEPROM content) is not included in the CRC calculation. The constant value used for CRC calculation is DB71\_0641h. The calculated CRC value is always placed into the **Serial EEPROM CRC Value** register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 270h), after serial EEPROM auto load.

There is an optional CRC checking at the end of serial EEPROM auto load. If bit 7 of the second byte (address 01h, following the **Validation Signature**) is Set, CRC checking is enabled. When CRC checking is performed, the calculated CRC value is compared to a stored CRC value read from serial EEPROM addresses [BYTE\_COUNT+4] through [BYTE\_COUNT+7]. If the calculated CRC value does not match the stored CRC value, the **Serial EEPROM Status** register *EepCRCErr* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 260h[19]) is Set, and an Uncorrectable Internal Error is triggered in Port 0 of Station 0 (**Uncorrectable Error Status** register *Uncorrectable Internal Error Status* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset FB8h[22])). The serial EEPROM loading of registers is not aborted.

When programming the serial EEPROM, it is not necessary for software to calculate the CRC value that is to be stored within the serial EEPROM. Instead, CRC Checking can be initially disabled, then after the system is re-booted or reset (forcing EEPROM reload), the calculated CRC (from the Serial EEPROM CRC Value register) can be written to serial EEPROM addresses [BYTE\_COUNT+4] through [BYTE\_COUNT+7], and then CRC Checking can be enabled (by programming serial EEPROM address 01h to value 80h).

**Table 6-1. Serial EEPROM Data Format**

Location	Value	Description
0h	5Ah	Validation Signature
1h	–	Bit 7 = CRC checking enable <sup>a</sup>
2h	REG BYTE COUNT (LSB)	Configuration register Byte Count (LSB)
3h	REG BYTE COUNT (MSB)	Configuration register Byte Count (MSB)
4h	REGADDR (LSB)	1 <sup>st</sup> Configuration Register Address (LSB)
5h	REGADDR (MSB)	1 <sup>st</sup> Configuration Register Address (MSB)
6h	REGDATA (Byte 0)	1 <sup>st</sup> Configuration Register Data (Byte 0)
7h	REGDATA (Byte 1)	1 <sup>st</sup> Configuration Register Data (Byte 1)
8h	REGDATA (Byte 2)	1 <sup>st</sup> Configuration Register Data (Byte 2)
9h	REGDATA (Byte 3)	1 <sup>st</sup> Configuration Register Data (Byte 3)
Ah	REGADDR (LSB)	2 <sup>nd</sup> Configuration Register Address (LSB)
Bh	REGADDR (MSB)	2 <sup>nd</sup> Configuration Register Address (MSB)
Ch	REGDATA (Byte 0)	2 <sup>nd</sup> Configuration Register Data (Byte 0)
Dh	REGDATA (Byte 1)	2 <sup>nd</sup> Configuration Register Data (Byte 1)
Eh	REGDATA (Byte 2)	2 <sup>nd</sup> Configuration Register Data (Byte 2)
Fh	REGDATA (Byte 3)	2 <sup>nd</sup> Configuration Register Data (Byte 3)
...	...	...
End of Serial EEPROM data (BYTE COUNT +3)	REGDATA (Byte 3)	Last Configuration Register Data (Byte 3)
BYTE COUNT +4	Stored CRC Byte 0	Optional, used when CRC is enabled <sup>a</sup>
BYTE COUNT +5	Stored CRC Byte 1	
BYTE COUNT +6	Stored CRC Byte 2	
BYTE COUNT +7	Stored CRC Byte 3	



- a. There is an optional CRC checking at the end of serial EEPROM auto load. Bit 7 of the second byte (after the 5Ah Validation Signature) determines whether CRC checking is to be performed. The CRC value is calculated for serial EEPROM addresses REG BYTE COUNT (LSB) (that is, 2h) through BYTE COUNT + 3. If the “REG BYTE COUNT (LSB)” value is not a multiple of 4, two additional bytes (of value 0h) are included in the calculation, so that CRC polynomial is always applied to DWord data.

The calculated CRC value is always placed into the **Serial EEPROM CRC Value** register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 270h) after serial EEPROM auto load. If CRC checking is enabled, and the stored 32-bit CRC value read from the end of the serial EEPROM content does **not** match the calculated value:

1. The **Serial EEPROM Status** register **EepCRCErr** bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 260h[19]) is Set, and,
2. An Uncorrectable Internal error is triggered at Port 0 of Station 0 (**Uncorrectable Error Status** register **Uncorrectable Internal Error Status** bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset FB8h[22]) is Set). The severity for Internal errors is Fatal, by default (**Uncorrectable Error Severity** register **Uncorrectable Internal Error Severity** bit (Port 0, offset FC0h[22]) is Set).

Serial EEPROM CRC errors do **not** inhibit serial EEPROM auto load of registers.

**Table 6-2. Configuration Register Address Format**

Port Number <sup>a</sup>	REGADDR Bits [15:10] Value <sup>b, c</sup>	Port Identifier
Port 0	0000_00b	0000h
Port 1	0000_01b	0400h
Port 2	0000_10b	0800h
Port 3	0000_11b	0C00h
Port 8	0010_00b	2000h
Port 9	0010_01b	2400h
Port 10	0010_10b	2800h
Port 11	0010_11b	2C00h
Port 16	0100_00b	4000h
Port 17	0100_01b	4400h
Port 18	0100_10b	4800h
Port 19	0100_11b	4C00h
Virtual Switch Mode Station-Specific Registers		
Station 0, Port 0	1100_00b	C000h
Station 1, Port 8	1100_01b	C400h
Station 2, Port 16	1100_10b	C800h
NT Mode		
NT Port Link Interface	1110_00b	E000h
NT Port Virtual Interface	1110_01b	E400h
NT PCI-to-PCI <sup>d</sup>	0XXX_XXb	XX00h

- a. Refer to [Table 4-1](#) for the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes. The table also indicates which Ports are enabled/used, and when, based upon the STRAP\_STNx\_PORTCFGx input states and/or serial EEPROM programming.
- b. Encodings not listed are **Reserved**.
- c. REGADDR bits [15:13] indicate the Station Number. REGADDR bits [12:10] indicate the relative Port Number (0, 1, 2, or 3) within the Station.
- d. Use the values for the Station Number and Port Number for the Port that is configured as the NT Port in the **VS0 Upstream** register **NT Port** field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 360h[12:8]).

## 6.5 Serial EEPROM Initialization

After the device Reset is de-asserted, the PEX 8748 determines whether a serial EEPROM is present. The serial EEPROM is considered to be present if it returns a value other than FFh in response to an initial Read Status command of its **Status** register, in which case this value is copied to the **Status Data from Serial EEPROM** register bits (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 260h[31:24]). Serial EEPROM presence is reported in the **Serial EEPROM Status** register *EepPrsnt[1:0]* field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 260h[17:16]); a value of 01b or 11b indicates that the serial EEPROM is present.

If a serial EEPROM is detected, the first byte (**Validation Signature**) is read. If a value of 5Ah is read, it is assumed that the serial EEPROM is programmed for the PEX 8748. The serial EEPROM address width is determined while the first byte is read. If the first byte's value is not 5Ah, the serial EEPROM is blank or programmed with invalid data. In this case, no more data is read from the serial EEPROM, and the **Serial EEPROM Status** register *EepAddrWidth* field (offset 260h[23:22]) reports a value of 00b (undetermined width).

If the *EepAddrWidth* field reports a value of 00b, any subsequent accesses to the serial EEPROM (through the PEX 8748 Serial EEPROM registers) default to a serial EEPROM address width of 1 byte, unless the **Serial EEPROM Status** register *EepAddrWidth Override* bit (offset 260h[21]) is Set. The *EepAddrWidth* field is usually Read-Only (RO); however, it is writable if the *EepAddrWidth Override* bit is Set (both can be programmed by a single Write instruction).

If the serial EEPROM contains valid data, the REG\_BYTE\_COUNT values in Bytes 2 and 3 determine the quantity of serial EEPROM locations that contain Configuration register addresses and data. Each Configuration register entry consists of 2 bytes of register Address and 4 bytes of register Write data. The REG\_BYTE\_COUNT must be a multiple of 6.

The **EE\_SK** output clock frequency is determined by the **Serial EEPROM Clock Frequency** register *EepFreq[2:0]* field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 268h[2:0]). The default clock frequency is 1 MHz. At this clock rate, it takes approximately 48  $\mu$ s per DWORD during Configuration register initialization. For faster loading of large serial EEPROMs that support a faster clock, the first Configuration register load from the serial EEPROM could be to the **Serial EEPROM Clock Frequency** register.

There is an optional CRC checking at the end of serial EEPROM auto load. Bit 7 of the second byte (after the 5Ah **Validation Signature**) determines whether CRC checking is to be performed. The CRC value is calculated for serial EEPROM addresses 2h through BYTE COUNT +3. The calculated CRC value is always placed into the **Serial EEPROM CRC Value** register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 270h) after serial EEPROM auto load. If CRC checking is enabled, and the stored 32-bit CRC value read from the end the serial EEPROM content does not match the calculated value, an Uncorrectable Internal error is triggered at Port 0 of Station 0.

## 6.6 Serial EEPROM Registers

The Serial EEPROM register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offsets 260h through 270h) parameters defined in Section 13.16.5, “Device-Specific Registers – Serial EEPROM (Offsets 260h – 270h),” can be changed, using the serial EEPROM. It is recommended that the first serial EEPROM entry be used to change the **Serial EEPROM Clock Frequency** register (offset 268h) value, to increase the clock frequency, and thereby reduce the time needed for the remainder of the serial EEPROM load. When the NT Port Expansion ROM feature is used, the serial EEPROM clock frequency must be 5 MHz or higher. At the last serial EEPROM entry, the **Serial EEPROM Status and Control** register (offset 260h) can be programmed to issue a Write Status Register (WRSR) command, to enable the Write Protection feature(s) within the serial EEPROM data, if needed.

NDA CONFIDENTIAL -- Inventec | King Michael

## 6.7 Serial EEPROM Random Write/Read Access

To access the serial EEPROM, a PCI Express, I<sup>2</sup>C, or SMBus Master uses the following registers (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port):

- **Serial EEPROM Status and Control** (offset 260h)
- **Serial EEPROM Buffer** (offset 264h)
- **Serial EEPROM 3rd Address Byte** (offset 26Ch)

***Note:** To help streamline the text in the following subsections, the specific Port location/access of each register offset is not repeated – only the offset location is mentioned.*

The Master can only access the serial EEPROM on a DWord basis (4 bytes aligned to one DWord address).

### 6.7.1 Writing to Serial EEPROM

*To write a DWord to the serial EEPROM:*

1. If the 3<sup>rd</sup> Address byte (Address bits [23:16]) is needed (when the **Serial EEPROM Status** register *EepAddrWidth* field bits (offset 260h[23:22]) are both Set), write the value to the **Serial EEPROM 3rd Address Byte** register *Serial EEPROM 3rd Address Byte* field (offset 26Ch[7:0]).
2. Write the 32-bit data into the **Serial EEPROM Buffer** register (offset 264h).
3. Issue a Write Enable instruction to the serial EEPROM (Command = 110b, Set Write Enable Latch), by writing the value 0000\_C000h into the **Serial EEPROM Status and Control** register (offset 260h).
4. Calculate and write the combined Address and Command value to write into the **Serial EEPROM Control** register (offset 260h), by combining the serial EEPROM 3-bit Write Data instruction (value 010b) as the *EepCmd[2:0]* field [15:13], together with the serial EEPROM address. Serial EEPROM Address bits [14:2] must be programmed into the register's *EepBlkAddr* field [12:0], and serial EEPROM Address bit 15 must be programmed into the **Serial EEPROM Status** register *EepBlkAddr Upper Bit* bit (*that is*, Set offset 260h[20] if the serial EEPROM address is in the upper 32 KB of any 64-KB address block within the serial EEPROM). The data in the **Serial EEPROM Buffer** register is written to the serial EEPROM when the **Serial EEPROM Status and Control** register is written.
5. The serial EEPROM Write operation is complete when a subsequent read of the **Serial EEPROM Status** register *EepCmdStatus* bit (offset 260h[18]) returns a value of 0. At this time, another serial EEPROM access can be started.

Because each PEX 8748 Port and Register address value (REGADDR; refer to [Section 6.5](#)), and its corresponding Data value (REGDATA), require 6 bytes of serial EEPROM memory, and the PEX 8748 serial EEPROM interface accesses 4 bytes at a time, two serial EEPROM Writes may be needed to store each set of REGADDR (one word) and REGDATA (1 DWord) entries into the serial EEPROM. To avoid overwriting a word of another set of 6-byte REGADDR and REGDATA values, one of the two Serial EEPROM Writes might need to be a Read-Modify-Write type of operation (preserving one word read from the serial EEPROM, and writing the value back along with a new word value).

## 6.7.2 Reading from Serial EEPROM

*To read a DWord from the serial EEPROM:*

1. If the 3<sup>rd</sup> Address byte (Address bits [23:16]) is needed (when the **Serial EEPROM Status** register *EepAddrWidth* field bits (offset 260h[23:22]) are both Set), write the value to the **Serial EEPROM 3rd Address Byte** register *Serial EEPROM 3rd Address Byte* field (offset 26Ch[7:0]).
2. Calculate the combined Address and Command value to write into the **Serial EEPROM Control** register (offset 260h), by combining the serial EEPROM 3-bit Read Data instruction (value 011b) as bits [15:13], together with the serial EEPROM address. Serial EEPROM Address bits [14:2] must be programmed into **Serial EEPROM Control** register bits [12:0], and serial EEPROM Address bit 15 must be programmed into **Serial EEPROM Status** register bit 20 (*that is*, Set bit 20 if the serial EEPROM address is in the upper 32 KB of any 64-KB address block within the serial EEPROM).
3. Poll the **Serial EEPROM Status** register until the *EepCmdStatus* bit (offset 260h[18]) is Cleared, which signals that the transaction is complete.
4. Read the four bytes of serial EEPROM data from the **Serial EEPROM Buffer** register (offset 264h).

*For example*, to read the first DWord in the serial EEPROM, write the value 0000\_6000h to Port 0, register offset 260h, and then read Port 0, register offset 264h.

## 6.7.3 Programming a Blank Serial EEPROM

The PEX 8748 supports 1-, 2-, or 3-byte serial EEPROM addressing. 8-Kbit to 512-Kbit SPI EEPROMs use 2-byte addressing. The PEX 8748 requires that the first byte in the serial EEPROM must be the value 5Ah (ASCII Z), as a **Validation Signature**.

The 2<sup>nd</sup> and 3<sup>rd</sup> bytes contain the quantity of bytes within the serial EEPROM image, beginning with the first register entry at serial EEPROM address 04h. If this Byte Count value exceeds the actual quantity of register entries times 6 (*for example*, if the first DWord is programmed to the value 5A00\_FFFh), the system could hang. To simplify programming of a blank EEPROM (*such as* in a typical production build), the serial EEPROM could be pre-programmed with the first DWord, 0000\_005Ah.

A 2-byte address serial EEPROM that is blank (or corrupted) can be programmed according to the following procedure (when the PEX 8748 is in 1-Byte Address mode).

*To program a blank serial EEPROM:*

1. Write the value 0000\_005Ah into the **Serial EEPROM Buffer** register at address [Upstream Port BAR0 + 264h].
2. Issue a Write Enable instruction (Command = 110b, Set Write Enable Latch, and enable 2-byte addressing, by writing the value 00A0\_C000h into the **Serial EEPROM Status and Control** register (offset 260h).
3. Copy this data value to serial EEPROM location 0, by writing the value 00A0\_4000h into the **Serial EEPROM Status and Control** register. At this point, the first four bytes in the serial EEPROM now contain the value 0000\_005Ah.
4. Reboot the system, to reset the PEX 8748 so that it re-detects the serial EEPROM.

## 6.8 Serial EEPROM Loading of NT Port Link Interface Registers – NT Mode

The **Virtual Switch Debug** register *Load Only EEPROM NT-Link on Hot Reset* and *Inhibit EEPROM NT-Link Load on Hot Reset* bits (Upstream Port(s), offset A30h[27:26], respectively) control whether the serial EEPROM is to load registers following a Soft Reset (Hot Reset or DL\_Down) to the Upstream Port or NT Port Link Interface, as defined in Table 6-3.

**Table 6-3. Serial EEPROM Loading of NT Port Link Interface Registers (Upstream Port(s), Offset A30h[27:26] Values) – NT Mode**

Bit 27 Value	Bit 26 Value	Action
0	0	Load all registers from the serial EEPROM.
0	1	Load all registers, except the NT Port Link Interface registers, from the serial EEPROM.
1	0	Load only NT Port Link Interface registers from the serial EEPROM.
1	1	Disable serial EEPROM loading of all registers.

## 6.9 NT Port Expansion ROM – NT Mode

The PEX 8748 NT Port Virtual and Link Interfaces support Expansion ROM, as defined in the *PCI r3.0*. The NT Port Expansion ROM can be implemented for either interface, but not both concurrently. The Expansion ROM image is stored in the serial EEPROM, and its size can be either 16 KB (default, bit is Cleared) or 32 KB (maximum), based upon the **Serial EEPROM Clock Frequency** register *Expansion ROM Size* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 268h[16]) value. When the NT Port Expansion ROM feature is used, the serial EEPROM clock frequency must be 5 MHz or higher.

By default, the Expansion ROM is enabled on the NT Port Link Interface; however, it can be enabled instead for the NT Port Virtual Interface, by Setting the **Ingress Chip Control** register *Expansion ROM Virtual Side* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 764h[0]). The **Expansion ROM Base Address** register (BAR) must be enabled, by Setting the register's *Expansion ROM Enable* bit, in either the NT Port Virtual Interface (offset 30h[0]) or NT Port Link Interface (offset 30h[0]).

The NT Port Expansion ROM's location within the serial EEPROM is programmed in the **Serial EEPROM 3rd Address Byte** register *Expansion ROM Base Address* field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 26Ch[31:16]), of which the lower six bits, [21:16], map to serial EEPROM byte Address bits [15:10] (aligned to a 256-DWord (1-KB) boundary). The NT Port Expansion ROM must not straddle a 64-KB boundary within the serial EEPROM.

The default serial EEPROM Base Address value is as follows:

- **16-KB NT Port Expansion ROM (*Expansion ROM Size* bit is Cleared)** – The value is 0020h, which corresponds to serial EEPROM Byte address 2000h (8 KB). The serial EEPROM size must be at least 32 KB.
- **32-KB NT Port Expansion ROM (*Expansion ROM Size* bit is Set)** – The value is 0040h, which corresponds to serial EEPROM byte address 4000h (16 KB). The serial EEPROM size must be at least 64 KB.





## Chapter 7 I<sup>2</sup>C/SMBus Slave Interface Operation

### 7.1 Introduction

This chapter discusses the [I<sup>2</sup>C Slave Interface](#) and [SMBus Slave Interface](#).

### 7.2 I<sup>2</sup>C Slave Interface

#### 7.2.1 I<sup>2</sup>C Support Overview

**Note:** This section applies to the I<sup>2</sup>C Slave interface, which uses the [I2C\\_ADDR0](#), [I2C\\_SCL0](#), and [I2C\\_SDA0](#) signals for PEX 8748 register access by an I<sup>2</sup>C Master. The [I2C\\_SCL1](#) and [I2C\\_SDA1](#) signals form the PEX 8748 I<sup>2</sup>C Master interface, which is used only for Serial Hot Plug operation. (Refer to [Section 10.11](#), “Serial Hot Plug Controller.”)

Inter-Integrated Circuit (I<sup>2</sup>C) is a bus used to connect Integrated Circuits (ICs). Multiple ICs can be connected to an I<sup>2</sup>C Bus, and I<sup>2</sup>C devices that have I<sup>2</sup>C mastering capability can initiate a Data transfer. I<sup>2</sup>C is used for Data transfers between ICs at relatively low rates (100 Kbps), and is used in a variety of applications. For further details regarding I<sup>2</sup>C Buses, refer to the [I2C Bus, v2.1](#).

The PEX 8748 is an I<sup>2</sup>C Slave. Slave operations allow the PEX 8748 Configuration registers to be read from or written to by an I<sup>2</sup>C Master, external from the device. I<sup>2</sup>C is a sideband mechanism that allows the device Configuration registers to be programmed, read from, or written to, independent of the PCI Express Upstream Link.

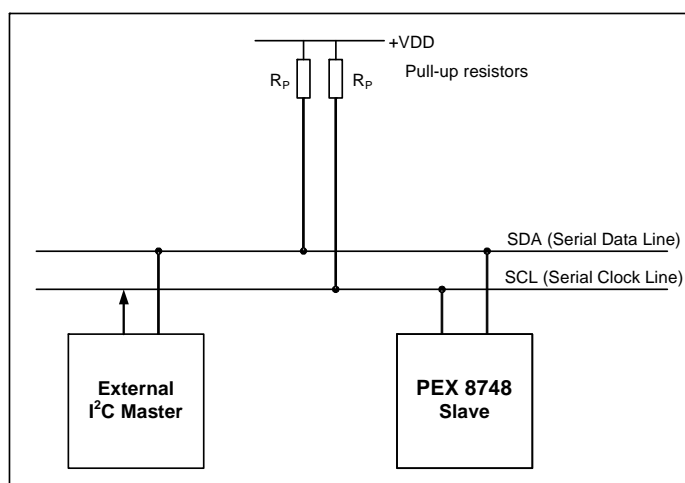
With I<sup>2</sup>C, users have the option of accessing all PEX 8748 registers through the I<sup>2</sup>C Slave interface. I<sup>2</sup>C provides an alternative to using a serial EEPROM, except the I<sup>2</sup>C/SMBus cannot access the registers (in time) prior to initial Link training (unless linkup is delayed by strapping the [STRAP\\_I2C\\_SMBUS\\_CFG\\_EN#](#) input Low, or by Clearing the [Configuration Release](#) register [Initiate Configuration](#) bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset [3ACh\[0\]](#)) by serial EEPROM); therefore, the serial EEPROM is recommended for complete programmability options. I<sup>2</sup>C can also be used for debugging, such as if the PEX 8748 Upstream Port(s) fail(s) to linkup.

Accordingly, it is recommended that both I<sup>2</sup>C/SMBus access, and the serial EEPROM (or at least its footprint), be included in designs.

The I2C\_SCL0 and I2C\_SDA0 signals can be brought out to a 2x2 pin header on the board, to allow PLX software (for example, running on a laptop computer) to access the PEX 8748 registers, using an Aardvark USB-I<sup>2</sup>C adapter connected to this header. (Refer to the [PEX 8748 RDK Hardware Reference Manual](#) for the header pin design.)

Figure 7-1 provides a block diagram that illustrates how standard devices connect to the I<sup>2</sup>C Bus.

**Figure 7-1. Standard Devices to I<sup>2</sup>C Bus Connection Block Diagram**





## 7.2.2 I<sup>2</sup>C Addressing – Slave Mode Access

To access the PEX 8748 Configuration registers through the I<sup>2</sup>C Slave interface, the PEX 8748 I<sup>2</sup>C Slave address must be configured.

The PEX 8748 supports a 7-bit I<sup>2</sup>C Slave address, defined by the `I2C_ADDR0` 3-state input, which can be strapped Low (0), left floating (Z), or High (1), to select a different Slave address – `0111_000b`, `0111_001b`, or `0111_010b`, respectively.

Up to three PEX 8748 devices can share the same I<sup>2</sup>C Bus segment without conflict, provided that each PEX 8748 has its `I2C_ADDR0` 3-state input strapped to a unique state. More than three PEX 8748 devices can share the I<sup>2</sup>C Bus, however, if the upper Address bits are programmed in the serial EEPROM.

The Slave address can also be programmed by the serial EEPROM (if present; recommended, if the default address must be changed), or by a Memory Write, in the **I2C Configuration** register *Slave Address* field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset `294h[6:0]`). The bits of field `[6:0]` correspond to Address Byte bits `[7:1]`, with bit 0 of the byte indicating a Write (0) or Read (1).

**Note:** *The I<sup>2</sup>C Slave address must not be changed by an I<sup>2</sup>C Write command.*

## 7.2.3 I<sup>2</sup>C Command Format

An I<sup>2</sup>C transfer starts as a packet with Address Phase bytes, followed by four Command Phase bytes, and one or more Data Phase bytes. The I<sup>2</sup>C packet Address Phase Byte format is illustrated in [Figure 7-2a](#). The I<sup>2</sup>C Command portion must include 4 bytes of data that contain the following:

- I<sup>2</sup>C Transfer type (Read/Write)
- PCI Express Configuration Register address
- PEX 8748 Port Number being accessed
- Byte Enable(s) of the register data being accessed

When the I<sup>2</sup>C Master is writing to the PEX 8748, the I<sup>2</sup>C Master must transmit the Data bytes to be written to that register within the same packet that contains the Command bytes. [Table 7-2](#) describes each I<sup>2</sup>C Command byte for Write access. [Figure 7-2b](#) illustrates the Command phase portion of an I<sup>2</sup>C Write packet.

When the I<sup>2</sup>C Master is reading from the PEX 8748, the I<sup>2</sup>C Master must separately transmit a Command Phase packet and Data Phase packet. [Table 7-6](#) describes each I<sup>2</sup>C Command byte for Read access. [Figure 7-4b](#) illustrates the Command phase portion of an I<sup>2</sup>C Read packet.

Each I<sup>2</sup>C packet must contain 4 bytes of data. Pad unused packet Data bytes with zeros (0) to meet this requirement.

## 7.2.4 I<sup>2</sup>C Register Write Access

The PEX 8748 Configuration registers can be read from and written to, based upon I<sup>2</sup>C register Read and Write operations, respectively. An I<sup>2</sup>C Write packet consists of Address Phase bytes and Command Phase bytes, followed by one to four additional I<sup>2</sup>C Data bytes. [Table 7-1](#) defines mapping of the I<sup>2</sup>C Data bytes to the Configuration register Data bytes. [Figure 7-2c](#) illustrates the I<sup>2</sup>C Data byte format.

The I<sup>2</sup>C packet starts with the *S* (START condition) bit. Data bytes are separated by the *A* (Acknowledge Control Packet (ACK)) or *N* (Negative Acknowledge (NAK)) bit. The packet ends with the *P* (STOP condition) bit.

If the Master generates an invalid command, the targeted PEX 8748 register is not modified.

The PEX 8748 considers the 1<sup>st</sup> Data byte of the 4-byte Data phase, following the four Command bytes in the Command phase, as register Byte 3 (bits [31:24]). The next three Data bytes access register Bytes 2 through 0, respectively. Four Data bytes are required, regardless of the Byte Enable Settings in the Command phase. The Master can then generate either a STOP condition (to finish the transfer) or a repeated START condition (to start a new transfer). If the I<sup>2</sup>C Master sends more than the four Data bytes (violating PEX 8748 protocol), the PEX 8748 returns a NAK for the extra Data byte(s). (For further details regarding I<sup>2</sup>C protocol, refer to the [I2C Bus, v2.1.](#))

[Table 7-2](#) describes each I<sup>2</sup>C Command byte for Write access. In the packet described in [Figure 7-2](#), Command Bytes 0 through 3 for Writes follow the format specified in [Table 7-2](#).

**Table 7-1. I<sup>2</sup>C Register Write Access**

I <sup>2</sup> C Data Byte Order	PCI Express Configuration Register Bytes
0	Written to register Byte 3
1	Written to register Byte 2
2	Written to register Byte 1
3	Written to register Byte 0

**Table 7-2. I<sup>2</sup>C Command Format for Write Access**

Field (Byte) On Bus	Bit(s)	Value/Description
Command Byte 1	7:3	<b>Reserved</b> Should be Cleared.
	2:0	<b>Command</b> 011b = Write register Do not use other encodings for Writes.
Command Byte 2	7:6	<b>Reserved</b> Should be Cleared.
	5:4	<b>Access Mode</b> 00b = Transparent Ports 01b = NT Port Link Interface 10b = NT Port Virtual Interface 11b = <b>Reserved</b>
	3:2	<b>Station Select (when <i>Access Mode</i> = 00b)</b> 00b = Station 0 01b = Station 1 10b = Station 2 11b = <b>Reserved</b>
	1:0	<b>Port Select, Bits [2:1]</b>

Table 7-2. I<sup>2</sup>C Command Format for Write Access (Cont.)

Field (Byte) On Bus	Bit(s)	Value/Description										
Command Byte 3	7	<b>Port Select, Bit 0</b> When <i>Access Mode</i> = 00b, <i>Port Select</i> [2:0] selects the Port to access, as indicated by <i>Station Select (when Access Mode = 00b)</i> .  000b = Port 0 of the Station 001b = Port 1 of the Station 010b = Port 2 of the Station 011b = Port 3 of the Station  <i><b>Note:</b> If the Port Number indicated is not enabled within the Station, the associated encoding is <b>Reserved</b>.</i>  When <i>Access Mode</i> = 01b or 10b, <i>Port Select</i> [2:0] selects the NT Port.  000b = NT Port 001b = <b>Reserved</b>  All other encodings are <b>Reserved</b> .										
	6	<b>Reserved</b> Should be Cleared.										
	5:2	<b>Byte Enables</b>  <table><tr><th>Bit</th><th>Description</th></tr><tr><td>2</td><td>Byte Enable for Byte 0 (PEX 8748 register bits [7:0])</td></tr><tr><td>3</td><td>Byte Enable for Byte 1 (PEX 8748 register bits [15:8])</td></tr><tr><td>4</td><td>Byte Enable for Byte 2 (PEX 8748 register bits [23:16])</td></tr><tr><td>5</td><td>Byte Enable for Byte 3 (PEX 8748 register bits [31:24])</td></tr></table> 0 = Corresponding PEX 8748 register byte will not be modified 1 = Corresponding PEX 8748 register byte will be modified  All 16 combinations are valid values.	Bit	Description	2	Byte Enable for Byte 0 (PEX 8748 register bits [7:0])	3	Byte Enable for Byte 1 (PEX 8748 register bits [15:8])	4	Byte Enable for Byte 2 (PEX 8748 register bits [23:16])	5	Byte Enable for Byte 3 (PEX 8748 register bits [31:24])
	Bit	Description										
2	Byte Enable for Byte 0 (PEX 8748 register bits [7:0])											
3	Byte Enable for Byte 1 (PEX 8748 register bits [15:8])											
4	Byte Enable for Byte 2 (PEX 8748 register bits [23:16])											
5	Byte Enable for Byte 3 (PEX 8748 register bits [31:24])											
1:0	<b>PEX 8748 Register Address, Bits [11:10]</b>											
Command Byte 4	7:0	<b>PEX 8748 Register Address [9:2]</b>  <i><b>Note:</b> All register addresses are DWord-aligned. Therefore, Address bits [1:0] are implicitly Cleared, and then internally incremented for successive I<sup>2</sup>C byte Writes.</i>										

**Figure 7-2. I<sup>2</sup>C Write Packet****Figure 7-2a I<sup>2</sup>C Write Packet Address Phase Bytes**

1 <sup>st</sup> Cycle			
START	7 6 5 4 3 2 1	0	ACK/NAK
S	Slave Address[7:1]	Read/Write Bit 0 = Write	A

**Figure 7-2b I<sup>2</sup>C Write Packet Command Phase Bytes**

Command Cycle							
7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK
Command Byte 0	A	Command Byte 1	A	Command Byte 2	A	Command Byte 3	A

**Figure 7-2c I<sup>2</sup>C Write Packet Data Phase Bytes**

Write Cycle								
7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	STOP
Data Byte 0 (to selected register Byte 3)	A	Data Byte 1 (to selected register Byte 2)	A	Data Byte 2 (to selected register Byte 1)	A	Data Byte 3 (to selected register Byte 0)	A	P

### 7.2.4.1 I<sup>2</sup>C Register Write Example

The following tables illustrate a sample I<sup>2</sup>C packet for writing the PEX 8748 **MSI Upper Address** register (offset **50h**) for Port 1 (Port 1 of Station 0) with data 1234\_5678h.

**Note:** The PEX 8748 has a default I<sup>2</sup>C Slave address value of **0111\_000b**, **0111\_001b**, or **0111\_010b**, according to whether the **I2C\_ADDR0** 3-state input is strapped Low (0), left floating (Z), or High (1), respectively. The actual address is reflected in the **I2C Configuration** register **Slave Address** field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset **294h**[6:0]).

*The following example assumes the I<sup>2</sup>C Slave address is 0111\_000b (I2C\_ADDR0=0). The byte sequence on the I<sup>2</sup>C Bus, as listed in the following tables and figures, occurs after the START and before the STOP bits, by which the I<sup>2</sup>C Master frames the transfer.*

**Table 7-3. I<sup>2</sup>C Register Write Access Example – 1<sup>st</sup> Cycle**

Phase	Value	Value/Description
Address	70h	<b>Bits [7:1] for PEX 8748 I<sup>2</sup>C Slave Address (0111_000b)</b> Last bit (bit 0) for Write = 0.

**Table 7-4. I<sup>2</sup>C Register Write Access Example – Command Cycle**

Byte	Value	Value/Description
0	03h	[7:3] <b>Reserved</b> Should be Cleared. [2:0] <b>Command</b> 011b = Write register
1	00h	[7:6] <b>Reserved</b> Should be Cleared. [5:4] <b>Access Mode</b> [3:2] <b>Station Select</b> [1:0] <b>Port Select, Bits [2:1]</b>
2	BCh	7 <b>Port Select, Bit 0</b> 6 <b>Reserved</b> Should be Cleared. [5:2] <b>Byte Enables</b> All active. [1:0] <b>PEX 8748 Register Address, Bits [11:10]</b>
3	14h	[7:0] <b>PEX 8748 Register Address [9:2]</b>

**Table 7-5. I<sup>2</sup>C Register Write Access Example – Write Cycle**

Byte	Value	Description
0	12h	<b>Data to Write for Byte 3</b>
1	34h	<b>Data to Write for Byte 2</b>
2	56h	<b>Data to Write for Byte 1</b>
3	78h	<b>Data to Write for Byte 0</b>

**Figure 7-3. I<sup>2</sup>C Write Command Packet Example**  
**Figure 7-3a I<sup>2</sup>C Write Packet Address Phase Bytes**

1 <sup>st</sup> Cycle			
START	7 6 5 4 3 2 1	0	ACK/NAK
S	Slave Address[7:1]	Read/Write Bit 0 0 = Write	A

**Figure 7-3b I<sup>2</sup>C Write Packet Command Phase Bytes**

Command Cycle							
7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK
Command Byte 0 0000_0011b	A	Command Byte 1 0000_0000b	A	Command Byte 2 1011_1100b	A	Command Byte 3 0001_0100b	A

**Figure 7-3c I<sup>2</sup>C Write Packet Data Phase Bytes**

Write Cycle								
7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	STOP
Data Byte 0 0001_0010b	A	Data Byte 1 0011_0100b	A	Data Byte 2 0101_0110b	A	Data Byte 3 0111_1000b	A	P

## 7.2.5 I<sup>2</sup>C Register Read Access

When the I<sup>2</sup>C Master attempts to read a PEX 8748 register, two packets are transmitted. The 1<sup>st</sup> packet consists of Address and Command Phase bytes to the Slave. The 2<sup>nd</sup> packet consists of Address and Data Phase bytes.

According to the [I2C Bus, v2.1](#), a Read cycle is triggered when the Read/Write bit (bit 0) of the 1<sup>st</sup> cycle is Set. The Command phase reads the requested register content into the internal buffer. When the I<sup>2</sup>C Read access occurs, the internal buffer value is transferred on to the I<sup>2</sup>C Bus, starting from Byte 3 (bits [31:24]), followed by the subsequent bytes, with Byte 0 (bits [7:0]) being transferred last. If the I<sup>2</sup>C Master requests more than four bytes, the PEX 8748 re-transmits the same byte sequence, starting from Byte 3 of the internal buffer.

The 1<sup>st</sup> and 2<sup>nd</sup> I<sup>2</sup>C Read packets (illustrated in [Figure 7-4](#) and [Figure 7-5](#), respectively) perform the following functions:

- **1<sup>st</sup> packet** – Selects the register to read
- **2<sup>nd</sup> packet** – Reads the register (sample 2<sup>nd</sup> packet provided is for a 7-bit PEX 8748 I<sup>2</sup>C Slave address)

Although two packets are shown for the I<sup>2</sup>C Read, the I<sup>2</sup>C Master can merge the two packets together into a single packet, by not generating the STOP at the end of the first packet (Master does not relinquish the bus) and generating REPEAT START.

[Table 7-6](#) describes each I<sup>2</sup>C Command byte for Read access. In the packet described in [Figure 7-4](#), Command Bytes 0 through 3 for Reads follow the format specified in [Table 7-6](#).

**Table 7-6. I<sup>2</sup>C Command Format for Read Access**

Field (Byte) On Bus	Bit(s)	Value/Description
Command Byte 1	7:3	<b>Reserved</b> Should be Cleared.
	2:0	<b>Command</b> 100b = Read register Do not use other encodings for Reads.
Command Byte 2	7:6	<b>Reserved</b> Should be Cleared.
	5:4	<b>Access Mode</b> 00b = Transparent Ports 01b = NT Port Link Interface 10b = NT Port Virtual Interface 11b = <b>Reserved</b>
	3:2	<b>Station Select (when <i>Access Mode</i> = 00b)</b> 00b = Station 0 01b = Station 1 10b = Station 2 11b = <b>Reserved</b>
	1:0	<b>Port Select, Bits [2:1]</b>



Table 7-6. I<sup>2</sup>C Command Format for Read Access (Cont.)

Field (Byte) On Bus	Bit(s)	Value/Description										
Command Byte 3	7	<b>Port Select, Bit 0</b> When <i>Access Mode</i> = 00b, <i>Port Select</i> [2:0] selects the Port to access, as indicated by <i>Station Select (when Access Mode = 00b)</i> .  000b = Port 0 of the Station 001b = Port 1 of the Station 010b = Port 2 of the Station 011b = Port 3 of the Station  <i>Note:</i> If the Port Number indicated is not enabled within the Station, the associated encoding is <b>Reserved</b> .  When <i>Access Mode</i> = 01b or 10b, <i>Port Select</i> [2:0] selects the NT Port.  000b = NT Port 001b = <b>Reserved</b>  All other encodings are <b>Reserved</b> .										
	6	<b>Reserved</b> Should be Cleared.										
	5:2	<b>Byte Enables</b>  <table><tr><th>Bit</th><th>Description</th></tr><tr><td>2</td><td>Byte Enable for Byte 0 (PEX 8748 register bits [7:0])</td></tr><tr><td>3</td><td>Byte Enable for Byte 1 (PEX 8748 register bits [15:8])</td></tr><tr><td>4</td><td>Byte Enable for Byte 2 (PEX 8748 register bits [23:16])</td></tr><tr><td>5</td><td>Byte Enable for Byte 3 (PEX 8748 register bits [31:24])</td></tr></table> 0 = Corresponding PEX 8748 register byte will not be modified 1 = Corresponding PEX 8748 register byte will be modified  All 16 combinations are valid values.	Bit	Description	2	Byte Enable for Byte 0 (PEX 8748 register bits [7:0])	3	Byte Enable for Byte 1 (PEX 8748 register bits [15:8])	4	Byte Enable for Byte 2 (PEX 8748 register bits [23:16])	5	Byte Enable for Byte 3 (PEX 8748 register bits [31:24])
	Bit	Description										
2	Byte Enable for Byte 0 (PEX 8748 register bits [7:0])											
3	Byte Enable for Byte 1 (PEX 8748 register bits [15:8])											
4	Byte Enable for Byte 2 (PEX 8748 register bits [23:16])											
5	Byte Enable for Byte 3 (PEX 8748 register bits [31:24])											
1:0	<b>PEX 8748 Register Address, Bits [11:10]</b>											
Command Byte 4	7:0	<b>PEX 8748 Register Address [9:2]</b>  <i>Note:</i> All register addresses are DWord-aligned. Therefore, Address bits [1:0] are implicitly Cleared, for the <i>I<sup>2</sup>C</i> Read.										

**Figure 7-4. I<sup>2</sup>C Read Command Packet (1<sup>st</sup> Packet)****Figure 7-4a I<sup>2</sup>C Read Command Packet Address Phase Bytes**

1 <sup>st</sup> Cycle			
START	7 6 5 4 3 2 1	0	ACK/NAK
S	Slave Address[7:1]	Read/Write Bit 0 = Write	A

**Figure 7-4b I<sup>2</sup>C Read Command Packet Command Phase Bytes**

Command Cycle								
7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	STOP
Command Byte 0	A	Command Byte 1	A	Command Byte 2	A	Command Byte 3	A	P

**Figure 7-5. I<sup>2</sup>C Read Data Packet (2<sup>nd</sup> Packet)****Figure 7-5a I<sup>2</sup>C Read Data Packet Address Phase Bytes**

1 <sup>st</sup> Cycle			
START	7 6 5 4 3 2 1	0	ACK/NAK
S	Slave Address[7:1]	Read/Write Bit, 1 = Read	A

**Figure 7-5b I<sup>2</sup>C Read Data Packet Data Phase Bytes**

Read Cycle								
7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	7 6 5 4 3 2 1 0	ACK/NAK	STOP
Register Byte 3	A	Register Byte 2	A	Register Byte 1	A	Register Byte 0	A	P

### 7.2.5.1 I<sup>2</sup>C Register Read Address Example – Phase and Command Packet

The following is a sample I<sup>2</sup>C packet for reading the PEX 8748 **MSI Upper Address** [63:32] register (offset **50h**) in Port 1 (Port 1 of Station 0), assuming the register value is ABCD\_EF01h.

**Note:** The PEX 8748 has a default I<sup>2</sup>C Slave address value of **0111\_000b**, **0111\_001b**, or **0111\_010b**, according to whether the **I2C\_ADDR0** 3-state input is strapped Low (0), left floating (Z), or High (1), respectively. The actual address is reflected in the **I2C Configuration** register **Slave Address** field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset **294h**[6:0]).

*The following example assumes the I<sup>2</sup>C Slave address is 0111\_000b (I2C\_ADDR0=0). The byte sequence on the I<sup>2</sup>C Bus, as listed in the following tables, occurs after the START and before the STOP bits, by which the I<sup>2</sup>C Master frames the transfer.*

**Table 7-7. I<sup>2</sup>C Register Read Access Example – 1<sup>st</sup> Packet**

Phase	Value	Value/Description
Address	70h	<b>Bits [7:1] for PEX 8748 I<sup>2</sup>C Slave Address (0111_000b)</b> Last bit (bit 0) for Write = 0.

**Table 7-8. I<sup>2</sup>C Register Read Access Example – Command Cycle**

Byte	Value	Value/Description
0	04h	[7:3] <b>Reserved</b> Should be Cleared. [2:0] <b>Command</b> 100b = Read register
1	00h	[7:6] <b>Reserved</b> Should be Cleared. [5:4] <b>Access Mode</b> [3:2] <b>Station Select</b> [1:0] <b>Port Select, Bits [2:1]</b>
2	BCh	7 <b>Port Select, Bit 0</b> 6 <b>Reserved</b> Should be Cleared. [5:2] <b>Byte Enables</b> All active. [1:0] <b>PEX 8748 Register Address, Bits [11:10]</b>
3	14h	[7:0] <b>PEX 8748 Register Address [9:2]</b>

## 7.3 SMBus Slave Interface

### 7.3.1 SMBus Features

- Compliant to the *SMBus v2.0*
- Supports the SMBus Slave function only
- PEX 8748 internal registers can be read and written, through the SMBus Slave interface
- Supports Address Resolution Protocol (ARP-capable)
- [I2C\\_ADDR0](#) input state, serial EEPROM, software, and/or ARP define the SMBus Device address
- Supports Block Read, Block Write, and Block Write - Block Read Process Call commands to access the registers
- Supports Packet Error Checking
- 10 to 100 KHz Bus operation frequency range

### 7.3.2 SMBus Operation

Based upon I<sup>2</sup>C's principles of operation, SMBus is a two-wire bus used for communication between IC components and the remainder of the system. Electrically, I<sup>2</sup>C and SMBus devices are compatible, and both protocol devices can co-exist on the same bus. Multiple devices, both Masters and Slaves, can be connected to an SMBus segment. PCI Express cards have two optional SMBus pins defined on the connector – SMCLK and SMDAT.

The PEX 8748 implements an *SMBus v2.0*-compliant Slave device, and is used to read and write PEX 8748 registers, through SMBus commands. The PEX 8748 SMBus uses the same SDA Data and SCL Clock balls that are used for I<sup>2</sup>C. Additionally, the SMBus Device Address is the same value as the I<sup>2</sup>C Slave address, used by the I<sup>2</sup>C protocol (default addresses [0111\\_000b](#), [0111\\_001b](#), or [0111\\_010b](#), according to whether the [I2C\\_ADDR0](#) 3-state input is strapped Low (0), left floating (Z), or High (1), respectively). At any time, either the I<sup>2</sup>C or SMBus feature is enabled, dependent upon the **SMBus Configuration** register *SMBus Enable* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset [2C8h\[0\]](#)) state, which is latched (at Fundamental Reset) according to the [STRAP\\_I2C\\_SMBUS\\_EN](#) 3-state input, to one of the following protocols:

- 0 = I<sup>2</sup>C (*SMBus Enable* bit is Cleared)
- Z, 1 = SMBus with ARP (*SMBus Enable* bit is Set)

Software can toggle the *SMBus Enable* bit to change between I<sup>2</sup>C and SMBus functionality.

The PEX 8748 SMBus logic also supports the commands that are required to support ARP. ARP is a feature specific to *SMBus v2.0*, through which an SMBus ARP Master can dynamically assign a unique address to each of the SMBus Targets residing on the same bus. Although ARP is an optional feature of the *SMBus v2.0*, PCI and PCI Express cards are required to support ARP. The ARP feature is enabled when the **SMBus Configuration** register *ARP Disable* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset [2C8h\[8\]](#)) is Cleared; this bit is initially latched (at Fundamental Reset) according to the [STRAP\\_I2C\\_SMBUS\\_EN](#) input state, as described above.

The PEX 8748 also supports Packet Error Checking and Packet Error Code (PEC) generation, as explained in the *SMBus v2.0*. The *SMBus v2.0* optional feature, *Notify ARP Master* (which requires Master capability on the SMBus) is **not** supported.

### 7.3.3 Supported SMBus Commands

For register access, the SMBus logic supports three commands:

- Block Write (command BEh) is used to write the registers
- Block Write (command BAh), followed by Block Read (command BDh), can be used to read the registers
- Block Write - Block Read Process Call (commands BAh, CDh) can also be used to read registers

Unsupported SMBus commands – Quick Command, Send Byte, Receive Byte, Write Byte, Write Word, Read Byte, Read Word, and Process Call – are NACKed.

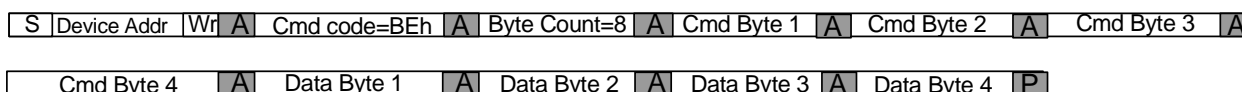
### 7.3.3.1 SMBus Block Write

The Block Write command is used to write to the PEX 8748 registers. General SMBus Block Writes are illustrated in [Figure 7-6](#) and [Figure 7-7](#). The sequence of Bytes include the following, in the sequence listed:

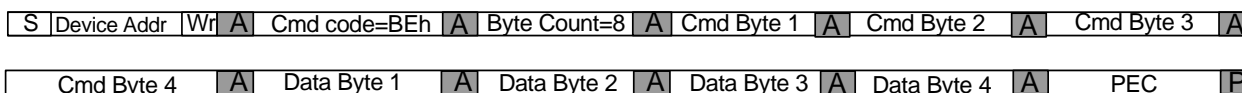
- 7-bit address,
- Command Code that indicates it is Block Write,
- *Byte Count* field with a value of 8h that indicates 4 bytes to set up the register to write (Port Number, register address, Command Byte Enable, and so forth), followed by
- 4 bytes of data to be written into the register

Figure 7-8 explains the elements used in Figure 7-6 and Figure 7-7, and Figure 7-9 indicates the Data Bytes written.

**Figure 7-6. SMBus Block Write Command Format, to Write to a PEX 8748 Register without PEC**



**Figure 7-7. SMBus Block Write Command Format, to Write to a PEX 8748 Register with PEC**



### Figure 7-8. SMBus Packet Protocol Diagram Element Key

S -> START condition

P -> STOP condition

A -> Acknowledge (this bit position may be 0 for an ACK or 1 for a NACK)

☐ -> Master to Slave

 -> Slave to Master

**Figure 7-9. SMBus Block Write Bytes, as Written to Register**

<b>31:24</b>	<b>23:16</b>	<b>15:8</b>	<b>7:0</b>
Data Byte 1	Data Byte 2	Data Byte 3	Data Byte 4

**Note:** In each byte, the Most Significant Byte (MSB) is transmitted first.

Table 7-9 provides a description of bytes for an SMBus Block Configuration Space register (CSR) Write.

Block Write transactions that are received with incorrect byte Settings are NACKed, starting from the wrong byte Setting, and including subsequent bytes in the packet. *For example*, if the Byte Count value is not 8, the PEX 8748 NACKs the byte corresponding to the Byte Count value, as well as any Data bytes following within the same packet.

The byte after Data Byte 4, if present, is taken as the PEC byte, and if present, the PEC is checked. If a packet fails Packet Error Checking, the PEX 8748 drops the packet (ignores the Write), and returns NACK for the PEC byte, to the SMBus Master. Packet Error Checking can be disabled, by Setting the **SMBus Configuration** register *PEC Check Disable* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 2C8h[9]). The Byte Count value, by definition, does not include the PEC byte.

**Table 7-9. Bytes for Block CSR Write on SMBus**

Field (Byte) On Bus	Bit(s)	Value/Description
Command Code	7:0	BEh for Block Write.
Byte Count	7:0	08h = 8 bytes to follow (4 Command and 4 Data bytes). The PEC byte is not counted.
Command Byte 1	7:3	<b>Reserved</b> Should be Cleared.
	2:0	<b>Command</b> 011b = Write register Do not use other encodings for Writes.
Command Byte 2	7:6	<b>Reserved</b> Should be Cleared.
	5:4	<b>Access Mode</b> 00b = Transparent Ports 01b = NT Port Link Interface 10b = NT Port Virtual Interface 11b = <b>Reserved</b>
	3:2	<b>Station Select (when Access Mode = 00b)</b> 00b = Station 0 01b = Station 1 10b = Station 2 11b = <b>Reserved</b>
	1:0	<b>Port Select, Bits [2:1]</b>

Table 7-9. Bytes for Block CSR Write on SMBus (Cont.)

Field (Byte) On Bus	Bit(s)	Value/Description										
Command Byte 3	7	<b>Port Select, Bit 0</b> When <i>Access Mode</i> = 00b, <i>Port Select</i> [2:0] selects the Port to access, as indicated by <i>Station Select (when Access Mode = 00b)</i> .  000b = Port 0 of the Station 001b = Port 1 of the Station 010b = Port 2 of the Station 011b = Port 3 of the Station  <i><b>Note:</b> If the Port Number indicated is not enabled within the Station, the associated encoding is <b>Reserved</b>.</i>  When <i>Access Mode</i> = 01b or 10b, <i>Port Select</i> [2:0] selects the NT Port.  000b = NT Port 001b = <i><b>Reserved</b></i>  All other encodings are <i><b>Reserved</b></i> .										
	6	<i><b>Reserved</b></i> Should be Cleared.										
	5:2	<b>Byte Enables</b>  <table><tr><th>Bit</th><th>Description</th></tr><tr><td>2</td><td>Byte Enable for Byte 0 (PEX 8748 register bits [7:0])</td></tr><tr><td>3</td><td>Byte Enable for Byte 1 (PEX 8748 register bits [15:8])</td></tr><tr><td>4</td><td>Byte Enable for Byte 2 (PEX 8748 register bits [23:16])</td></tr><tr><td>5</td><td>Byte Enable for Byte 3 (PEX 8748 register bits [31:24])</td></tr></table> 0 = Corresponding PEX 8748 register byte will not be modified 1 = Corresponding PEX 8748 register byte will be modified  All 16 combinations are valid values.	Bit	Description	2	Byte Enable for Byte 0 (PEX 8748 register bits [7:0])	3	Byte Enable for Byte 1 (PEX 8748 register bits [15:8])	4	Byte Enable for Byte 2 (PEX 8748 register bits [23:16])	5	Byte Enable for Byte 3 (PEX 8748 register bits [31:24])
	Bit	Description										
2	Byte Enable for Byte 0 (PEX 8748 register bits [7:0])											
3	Byte Enable for Byte 1 (PEX 8748 register bits [15:8])											
4	Byte Enable for Byte 2 (PEX 8748 register bits [23:16])											
5	Byte Enable for Byte 3 (PEX 8748 register bits [31:24])											
1:0	<b>PEX 8748 Register Address, Bits [11:10]</b>											
Command Byte 4	7:0	<b>PEX 8748 Register Address [9:2]</b>  <i><b>Note:</b> All register addresses are DWord-aligned. Therefore, Address bits [1:0] are implicitly Cleared, and then internally incremented for successive SMBus byte Writes.</i>										



### 7.3.3.2 SMBus Block Read

A Block Read command is used to read PEX 8748 registers. Similar to register Reads using I<sup>2</sup>C, an SMBus Write sequence must first be performed to select the register to read, followed by an SMBus Read of the corresponding register. There are two ways a PEX 8748 register can be read:

- Use a Block Write, followed by a Block Read. The Block Write sets up the parameters including Port Number, register address and Byte Enables, and the Block Read performs the actual Read operation.
- Use a Block Write - Block Read Process Call. This command is defined by the *SMBus v2.0*, and performs a Block Write and Block Read, using a two-part message. The Block Write portion of the message sets up the register to be read, and then a repeated START followed by the Block Read portion of the message returns the register data specified by the Block Write.

**Note:** There is no STOP condition before the repeated START condition.

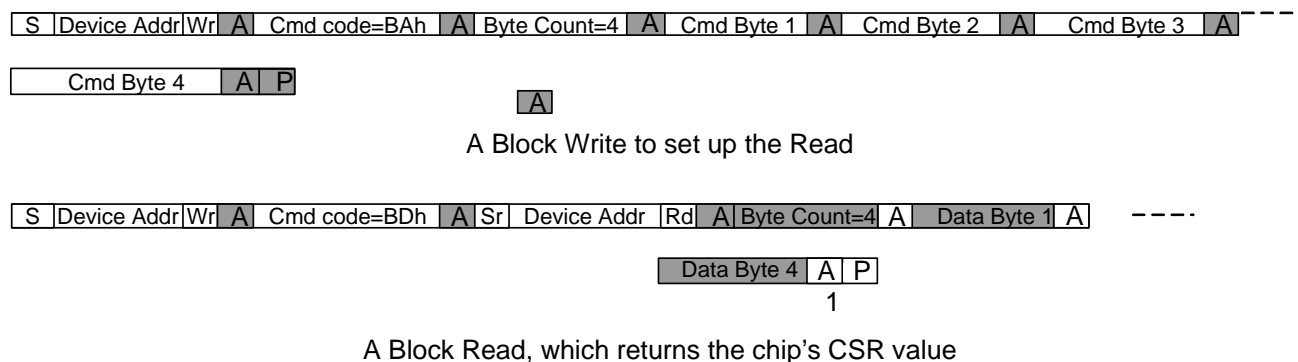
#### **CSR Read, Using SMBus Block Write, Followed by SMBus Block Read**

A general SMBus Block Write and Block Read sequence is illustrated in Figure 7-10.

Table 7-10 describes the Byte definitions for a Block Write bus protocol, to prepare for a subsequent Block Read of the PEX 8748 register.

The PEX 8748 always NACKs any incorrect command sequences, starting with the wrong Byte. Upon receiving the Block Read command (listed in Table 7-11), the PEX 8748 returns a PEC to the Master if, after the 4<sup>th</sup> byte of register data, the Master still requests one more Byte. As a Slave, the PEX 8748 recognizes the end of the Master's Read cycle, by observing the Master's NACK response for the last Data Byte transmitted by the PEX 8748.

**Figure 7-10. SMBus Block Write to Set up Read, and Resulting Read that Returns CSR Value**



**Table 7-10. SMBus Block Read Bytes**

Field (Byte) On Bus	Bit(s)	Value/Description
Command Code	7:0	BAh, to set up the Read, using Block Writes.
Byte Count	7:0	04h = 4 Command bytes.
Command Byte 1	7:3	<b>Reserved</b> Should be Cleared.
	2:0	<b>Command</b> 100b = Read register Do not use other encodings for Reads.
Command Byte 2	7:6	<b>Reserved</b> Should be Cleared.
	5:4	<b>Access Mode</b> 00b = Transparent Ports 01b = NT Port Link Interface 10b = NT Port Virtual Interface 11b = <b>Reserved</b>
	3:2	<b>Station Select (when <i>Access Mode</i> = 00b)</b> 00b = Station 0 01b = Station 1 10b = Station 2 11b = <b>Reserved</b>
	1:0	<b>Port Select, Bits [2:1]</b>

Table 7-10. SMBus Block Read Bytes (Cont.)

Field (Byte) On Bus	Bit(s)	Value/Description										
Command Byte 3	7	<b>Port Select, Bit 0</b> When <i>Access Mode</i> = 00b, <i>Port Select</i> [2:0] selects the Port to access, as indicated by <i>Station Select (when Access Mode = 00b)</i> .  000b = Port 0 of the Station 001b = Port 1 of the Station 010b = Port 2 of the Station 011b = Port 3 of the Station  <i><b>Note:</b> If the Port Number indicated is not enabled within the Station, the associated encoding is <b>Reserved</b>.</i>  When <i>Access Mode</i> = 01b or 10b, <i>Port Select</i> [2:0] selects the NT Port.  000b = NT Port 001b = <b>Reserved</b>  All other encodings are <b>Reserved</b> .										
	6	<b>Reserved</b> Should be Cleared.										
	5:2	<b>Byte Enables</b>  <table><tr><th>Bit</th><th>Description</th></tr><tr><td>2</td><td>Byte Enable for Byte 0 (PEX 8748 register bits [7:0])</td></tr><tr><td>3</td><td>Byte Enable for Byte 1 (PEX 8748 register bits [15:8])</td></tr><tr><td>4</td><td>Byte Enable for Byte 2 (PEX 8748 register bits [23:16])</td></tr><tr><td>5</td><td>Byte Enable for Byte 3 (PEX 8748 register bits [31:24])</td></tr></table> 0 = Corresponding PEX 8748 register byte will not be modified 1 = Corresponding PEX 8748 register byte will be modified  All 16 combinations are valid values.	Bit	Description	2	Byte Enable for Byte 0 (PEX 8748 register bits [7:0])	3	Byte Enable for Byte 1 (PEX 8748 register bits [15:8])	4	Byte Enable for Byte 2 (PEX 8748 register bits [23:16])	5	Byte Enable for Byte 3 (PEX 8748 register bits [31:24])
	Bit	Description										
2	Byte Enable for Byte 0 (PEX 8748 register bits [7:0])											
3	Byte Enable for Byte 1 (PEX 8748 register bits [15:8])											
4	Byte Enable for Byte 2 (PEX 8748 register bits [23:16])											
5	Byte Enable for Byte 3 (PEX 8748 register bits [31:24])											
1:0	<b>PEX 8748 Register Address, Bits [11:10]</b>											
Command Byte 4	7:0	<b>PEX 8748 Register Address [9:2]</b>  <i><b>Note:</b> All register addresses are DWord-aligned. Therefore, Address bits [1:0] are implicitly Cleared, for the SMBus Read.</i>										

Table 7-11. Command Format for SMBus Block Read

Field (Byte) On Bus	Bit(s)	Value/Description
Cmd Code	7:0	CDh, for Block Read (Process Call ReaD).

7.3.3.3 CSR Read, Using SMBus Block Write - Block Read Process Call

A general SMBus Block Write - Block Read Process Call sequence is illustrated in [Figure 7-11](#). Alternatively, a general SMBus Block Write - Block Read Process Call with PEC sequence is illustrated in [Figure 7-12](#).

Using this command, the register to be read can be set up and read back with one SMBus cycle (a transaction with a START and ending in STOP). There is no STOP condition before the repeated START condition.

[Table 7-11](#) lists the Command format for Block Read.

Figure 7-11. CSR Read Operation Using SMBus Block Write - Block Read Process Call

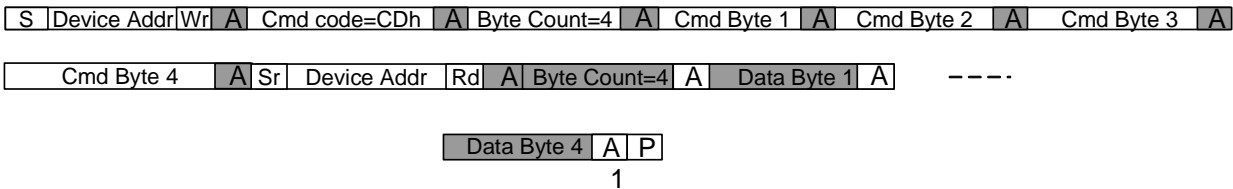
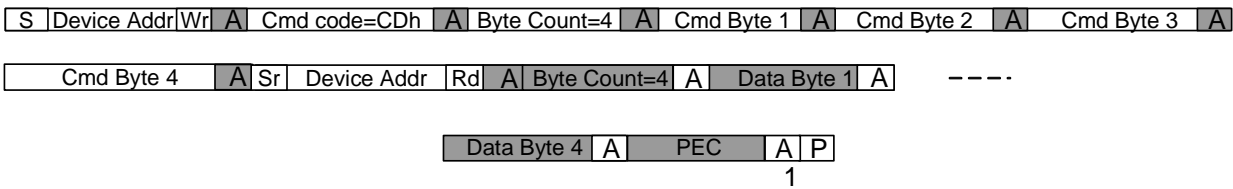


Figure 7-12. CSR Read Operation Using SMBus Block Write - Block Read Process Call with PEC



### 7.3.4 SMBus Address Resolution Protocol

Address Resolution Protocol (ARP) is a protocol by which SMBus devices that implement an assignable SMBus Device address feature are enumerated and dynamically assigned non-conflicting Device addresses, rather than using a fixed Device address. Although optional in the *SMBus v2.0*, it is mandatory per the *PCI r3.0* for add-in boards, to support ARP. This feature avoids conflicts with addresses used by other devices on a motherboard. ARP also allows multiple devices of the same type to co-exist on the same bus segment, without address conflicts.

To support this feature, a Slave device must implement a unique 128-bit ID, called *Unique Device Identifier (UDID)*. The fields of this ID are provided in [Figure 7-13](#). All ARP commands use the default Device Address, 1100\_001b. There are also two flags that the SMBus devices must implement to support the ARP process:

- **Address Resolved flag (AR)** – A flag bit or device internal state that indicates whether the ARP Master has resolved the device's SMBus Device address
- **Address Valid flag (AV)** – A flag bit or device internal state that indicates whether the device's SMBus Device address is valid

The process of assigning an SMBus Device address starts with the ARP Master issuing a Reset Device or Prepare to ARP command, using the default Device Address. This Clears the AR flag in the Slave device (both flags are Cleared by a Reset Device command). The Master then issues a general Get UDID command. This causes all devices that support ARP to start driving their UDID onto the serial bus. A Target that loses the SMBus arbitration, backs off. Arbitration loss means that a device keeps the SMDAT line floating and it detects 0 driven by another device on the bus. Slave devices that lose arbitration issue NACK in response to further Bytes transmitted on the bus. After the ARP Master finishes the Get UDID sequence, it issues a Set Address command to the Slave device, using the Slave's UDID. All Slave devices on the bus monitor the UDID that is transmitted by the ARP Master, but only the particular device that has the matching UDID adopts the new SMBus Device address, and Sets its own AV and AR flags. After the Slave device sets its AR flag, that device no longer responds to a general Get UDID command, which allows other devices to participate in the ARP process. All ARP commands require PEC checking and generation.

### 7.3.4.1 SMBus UDID

The 128-bit UDID is comprised of the following fields, as illustrated in [Figure 7-13](#) (not to scale). Each UDID field and its default value implemented in the PEX 8748 and meaning are explained in the tables that follow.

**Figure 7-13. 128-Bit SMBus UDID**

8 bits	8 bits	16 bits	16 bits	16 bits	16 bits	16 bits	32 bits
127:120	119:112	111:96	95:80	79:64	63:48	47:32	31:0
Device Capability	Version/Revision	Vendor ID	Device ID	Interface	Subsystem Vendor ID	Subsystem Device ID	Vendor-Specific ID

**Table 7-12. SMBus Vendor-Specific ID [31:0]**

Field	Name	Default Value	Description
31:0	Vendor-Specific ID	Depends upon the <a href="#">I2C_ADDR0</a> input state, which provides the following ID values: 0 = 7000_0000h Z = B000_0000h 1 = D000_0000h	The Vendor-Specific ID is used to provide a unique ID for functionally equivalent devices. This is for devices that would otherwise return identical UDIDs for the purpose of dynamic address assignment. The I2C_ADDR0 3-state input produces three unique Vendor-Specific ID values, for a maximum of three SMBus-enabled PEX 8748 switches to co-exist on the same SMBus segment.

**Table 7-13. SMBus Subsystem Device ID [47:32]**

Field	Name	Default Value	Description
15:0	Subsystem Device ID	8748h	PLX part number for the PEX 8748.

**Table 7-14. SMBus Subsystem Vendor ID [63:48]**

Field	Name	Default Value	Description
15:0	Subsystem Vendor ID	10B5h	PLX Vendor ID.

**Table 7-15. SMBus Interface [79:64]**

Field	Name	Default Value	Description
3:0	SMBus Version	0100b	<i>SMBus v2.0.</i>
15:4	<i>Reserved</i>	000h	Supported protocols.

**Table 7-16. SMBus Device ID [95:80]**

Field	Name	Default Value	Description
15:0	Device ID	8748h	PEX 8748 default Device ID value.

**Table 7-17. SMBus Vendor ID [111:96]**

Field	Name	Default Value	Description
15:0	Vendor ID	10B5h	PLX Vendor ID.

**Table 7-18. SMBus Version/Revision [119:112]**

Field	Name	Default Value	Description
2:0	Silicon Revision ID	001b	PEX 8748, Silicon Revisions BA.
5:3	UDID Version	001b	UDID version defined for the <i>SMBus v2.0.</i>
7:6	<i>Reserved</i>	00b	

**Table 7-19. SMBus Device Capability [127:120]**

Field	Name	Default Value	Description
0	PEC Supported	1	By default, PEC generation and checking are enabled.
5:1	<i>Reserved</i>	00_000b	
7:6	Address Type	10b	The PEX 8748 SMBus Address Type is implemented as dynamic and volatile. 00b = Fixed address 01b = Dynamic and persistent 10b = Dynamic and volatile (default) 11b = Random number device

### 7.3.4.2 Supported SMBus ARP Commands

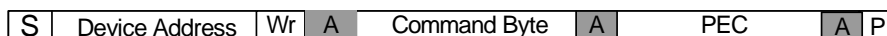
The PEX 8748 supports all ARP Slave commands. The Notify ARP Master command, which requires Master functionality, is *not* supported. Table 7-20 explains the PEX 8748 response to each received ARP command.

**Table 7-20. Supported SMBus ARP Commands, Format, and Actions**

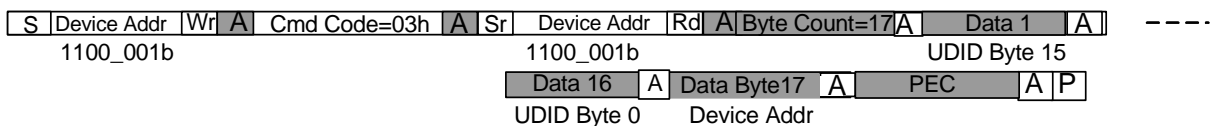
ARP Command	SMBus Command Format	SMBus Device Address	Command Code	Action
Prepare to ARP (Only General)	Send Byte (Refer to Figure 7-14)	SMBus default Device Address 1100_001b	01h	Clear the <i>AR Flag</i> and prepare for the ARP process. <i>AV Flag</i> will have no change.
Reset Device (General)	Send Byte (Refer to Figure 7-14)	SMBus default Device Address 1100_001b	02h	Clear the <i>AR Flag</i> and <i>AV Flag</i> .
Reset Device (Directed)	Send Byte (Refer to Figure 7-14)	SMBus default Device Address 1100_001b	Target Device Address[7:1] + 0	If the <i>AV Flag</i> is Set, Set ACK and Clear the <i>AR Flag</i> and <i>AV Flag</i> ; else, NACK/REJECT.
Get UDID (General)	Block Read (Refer to Figure 7-15)	SMBus default Device Address 1100_001b	03h	Respond only if the <i>AR Flag</i> is Cleared; else, NACK/REJECT. <i>AR Flag</i> and <i>AV Flag</i> are not changed. Address returned is all ones (1), if the <i>AV Flag</i> is Cleared.
Get UDID (Directed)	Block Read	SMBus default Device Address 1100_001b	Target Device Address[7:1] + 1	<i>AR Flag</i> and <i>AV Flag</i> are not changed. ACK if <i>AV Flag</i> =1; else, NACK/REJECT. Data Byte 17 returned will be the SMBus Device address.
Assign Address ARP	Block Write (Refer to Figure 7-16)	SMBus default Device Address 1100_001b	04h	Always ACK and Set the <i>AR Flag</i> and <i>AV Flag</i> , if the UDID matches.



**Figure 7-14. Prepare SMBus ARP Command and SMBus Reset Device Command Format**

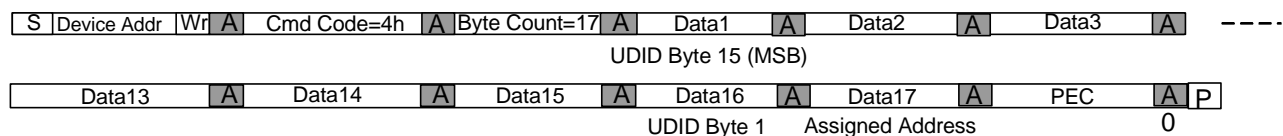


**Figure 7-15. Get SMBus UDID Command Format (General Get UDID Command with PEC)**



**Note:** If the **SMBus Configuration** register **AR Flag** bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 2C8h[11]) is Cleared, the device returns the register's **SMBus Device Address** field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 2C8h[7:1]) as 1111\_111b; otherwise, it returns the device SMBus Device address. Bit 0 (LSB) in the Data Byte 17 field should be 1.

### Figure 7-16. Assign SMBus Address ARP Command Format



**Note:** Bit 0 (LSB) of the Data 17 field is ignored in the Assign Address command field.

### 7.3.5 SMBus PEC Handling

The PEX 8748 supports the optional *SMBus v2.0* PEC generation and checking feature. This feature is required for the ARP process; however, it is optional for standard data transfer operation. The PEX 8748 supports PEC Cyclic Redundancy Check (CRC) generation and checking during ARP, as well as during Read/Write transfers to the PEX 8748 registers. The CRC polynomial used for PEC calculation is:

$$C(x) = x^8 + x^2 + x + 1$$

An 8-bit parallel CRC is implemented. The PEC calculation does not include ACK, NACK, START, STOP, nor repeated START bits. An SMBus Master can determine whether a Slave device supports PEC, from the UDID value returned by the Slave device, in response to a Get UDID command.

As a Slave device, the PEX 8748 checks the PEC, if the Master transmits the additional PEC byte and the PEX 8748 PEC checking feature is enabled (default). PEC checking can be disabled, by Setting the **SMBus Configuration** register *PEC Check Disable* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 2C8h[9]).

Additionally, when PEC is enabled, packets received with an incorrect PEC value are dropped. If PEC checking is disabled and a received PEC byte value is incorrect, the PEX 8748 accepts the packet. During a register Read, if the Master requests the additional PEC byte, the PEX 8748 generates and transmits the PEC byte after the register data.

### 7.3.6 Addressing PEX 8748 SMBus Slave

By default, the PEX 8748 supports ARP when **STRAP\_I2C\_SMBUS\_EN**=1, and expects the ARP Master to define the PEX 8748 SMBus Device Address. If ARP is disabled (**STRAP\_I2C\_SMBUS\_EN**=0, and the default **SMBus Configuration** register *ARP Disable* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 2C8h[8]) is Set), the default **SMBus Configuration** register *SMBus Device Address* bits [7:1] are 0111\_000b, 0111\_001b, or 0111\_010b, according to whether the **I2C\_ADDR0** 3-state input is strapped Low (0), left floating (Z), or High (1), respectively (Address bits [7:1] are 0111\_000b, with Address bit [1:0] values loaded from **I2C\_ADDR0**). The input state allows a maximum of three PEX 8748 SMBus Slaves to co-exist without address conflict on the SMBus, using SMBus Address byte values of 70h, 72h, and 74h. The **I2C\_ADDR0** 3-state input is loaded immediately after Fundamental Reset, and any subsequent change of input state does not affect functionality.

If the register's *UDID Address Type* field is programmed as fixed address (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 2C8h[13:12], are both Cleared) without disabling ARP, the PEX 8748 still participates in ARP, but does not Set the Device Address after ARP successfully completes.

The SMBus Slave Address can be changed at any time, by using software to write to the register's *SMBus Device Address* field (offset 2C8h[7:1]). ARP can also be enabled or disabled at runtime, by writing to the register's *ARP Disable* bit. If ARP is disabled by software after initially being enabled, the default address (70h) is not used for subsequent transactions. In this case, software must program a Device address into the *SMBus Device Address* field. When software writes the Device Address, it must also Set the register's *AR Flag* and *AV Flag* bits (offset 2C8h[11:10], respectively), to indicate that the address is valid and resolved.

Whenever software changes the register's *AV Flag*, *ARP Disable*, and/or *SMBus Device Address* values, software must also Set the register's *SMBus Parameter Reload* bit (offset 2C8h[15]). Writes to this register bit take effect only when the register's *SMBus Command In-Progress* bit (offset 2C8h[28]) is Cleared, which indicates that the PEX 8748 SMBus interface is in the Idle state.

### 7.3.7 SMBus Timeout

Unlike I<sup>2</sup>C, where the Slave or Master can indefinitely hold the I2C\_SCL0 line Low, SMBus has a timeout condition. No device is allowed to hold the I2C\_SCL0 line Low for more than 25 ms. When the PEX 8748, as a Slave-Transmitter, detects that it has pulled the I2C\_SCL0 line Low for more than 25 ms, the PEX 8748 releases I2C\_SCL0, and the logic returns to its default state and waits for another START condition. This can also occur when the Master pulls the I2C\_SCL0 line Low for more than 25 ms during any single Clock Low interval within a transfer in progress, or during the ACK phase if the Master pulls the I2C\_SCL0 line Low to process a task. Generally, the PEX 8748 pulls the I2C\_SCL0 line Low if SMBus access to registers is delayed by internal arbitration for register access.

## 7.4 Switching between SMBus and I<sup>2</sup>C Bus Protocols

The PEX 8748's I<sup>2</sup>C implementation allows switching between the SMBus and I<sup>2</sup>C protocols, by toggling the **SMBus Configuration** register *SMBus Enable* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 2C8h[0]).

When operating in SMBus mode, Clearing this bit, using the SMBus Block Write protocol, enables I<sup>2</sup>C protocol for subsequent register accesses. This SMBus Block Write can be transmitted from an SMBus or I<sup>2</sup>C Master, provided that the Block Write Byte sequence conforms to the sequence explained in [Section 7.3.3.1](#). In I<sup>2</sup>C mode, writing 1 to the *SMBus Enable* bit turns On the SMBus protocol, immediately after the Write operation is complete.

THIS PAGE INTENTIONALLY LEFT BLANK.



## Chapter 8 Performance Features

### 8.1 Introduction

This chapter discusses guidelines for programming on-chip registers, to boost performance beyond that provided by the general-purpose default values, specifically:

- [DLLP Policies](#)
- [Latency](#)
- [Queuing Options](#)
- [Read Pacing](#)
- [Multicast](#)

## 8.2 DLLP Policies

Data Link Layer Packet (DLLP) rates can vary from 0 to 2 or more DLLPs/TLP. The PEX 8748 allows programming to affect the DLLP rate. An increase in DLLPs reduces the total TLP throughput. Therefore, for designs that require high performance, it would be beneficial to minimize DLLP rates. Transmitting fewer DLLPs, however, can result in credit starvation or Replay buffer overfill, which can have a detrimental effect on TLP bandwidth. Care must be taken when changing the default PEX 8748 DLLP transmission rate.

Typically, TLPs have higher transmission priority on the wire than DLLPs. The PEX 8748, however, allows DLLPs to have higher priority under certain conditions, meaning that DLLPs can transmit before starting a new TLP. The decision to transmit a DLLP ahead of a TLP is referred to as *DLLP policy*.

The PEX 8748 can be programmed to alter its default DLLP policies, to emphasize improved TLP throughput, faster acknowledgement, more credit, or simplest behavior. The PEX 8748 default policies are designed to achieve optimal performance for most applications. Programmable choices for a DLLP policy, however, allow for further optimization.

### 8.2.1 ACK DLLP Policy

An *ACK DLLP* is a response indicating to the TLP Transmitter that the Receiver received a “good” copy of the TLP, meaning that it acknowledged receipt of the TLP. The simplest policy is to send 1 ACK for every received TLP, resulting in a 1 DLLP/TLP rate for ACK alone. What an ACK means to the TLP Transmitter is that the TLP Transmitter can remove any stored copy of that TLP, because it is unnecessary to resend the TLP. ACK DLLPs can be combined, so that one ACK DLLP can serve to acknowledge multiple TLPs. This collapsing of ACKs is the basis of the ACK DLLP policy choices. Less-frequent, more-collapsed ACKs have the least impact on TLP transmit bandwidth, meaning that less-frequent ACKs result in less than 1 DLLP/TLP.

The PEX 8748 ACK policy consists of two parts – a Timer and TLP Counter. The default ACK Timer policy/value varies according to the negotiated Link width, operating Link speed, and Maximum Packet Size, as recommended in the *PCI Express Base r1.1*, *PCI Express Base r2.1*, or *PCI Express Base r3.0*.

The ACK Transmission Latency Timer loads the appropriate value when a TLP is received and known to be good, meaning a few clocks after the END framing symbol is received. The Timer counts down each symbol time (every 4 ns (*PCI Express Base r1.1*) or 2 ns (*PCI Express Base r2.1*)). When the Timer reaches 0, an ACK DLLP takes higher priority over new TLPs (*that is*, an ACK DLLP is transmitted before a new TLP is started). The ACK DLLP transmitted acknowledges all TLPs, up to the most recently arrived good TLP.

The TLP Counter counts down on each TLP arrival until it reaches 0, then schedules a high-priority ACK DLLP. The default initialization value for the TLP Counter is 16, meaning a high-priority ACK is scheduled upon the arrival of 16 TLPs. The **Ingress Port-Based Control** register *ACK TLP Counter Timeout* field (Base mode – All Ports; Virtual Switch mode – VS Upstream Port(s), offset F48h[1:0]) value controls the ACK TLP Counter, as follows:

- 00b – Allows 16 TLPs before a high-priority ACK (default)
- 01b – Allows 8 TLPs before a high-priority ACK
- 10b – Allows 4 TLPs before a high-priority ACK
- 11b – Disables the Counter

Either the Latency Timer or TLP Counter mechanism can cause a high-priority ACK DLLP to be scheduled, and the first one to do so re-initializes both mechanisms to their starting parameters. *For example*, the time for 16 TLPs can be less than the ACK Timer above, in which case an ACK is sent earlier. The TLP Counter is useful for any system with a large programmed MPS (resulting in a large Timer value), that is capable of sending short TLPs (*such as* 12-byte Memory Reads). Rather than require the Transmitter to save possibly 100+ small TLPs, it need only save 16, plus whatever else arrives during the round-trip time.

If there is no TLP traffic being transmitted (*that is*, the Transmit Link is idle), an ACK DLLP can be transmitted immediately, before the Latency Timer expires. This is an opportunistic, low-priority ACK because it does not contend with a TLP in transmission. When an opportunistic, low-priority ACK is transmitted, both the Latency Timer and TLP Counter re-initialize, waiting for a new TLP to arrive to begin counting again.

The PEX 8748 allows a programmable override of the default Ack\_Latency\_Timer value, on a per-Port basis, by programming the **ACK Transmission Latency Limit** register *ACK Transmission Latency Limit* field (offset FA8h[11:0]). The value in this register is loaded when a new TLP arrives and a high-priority ACK DLLP is attempted when the Timer reaches 0. For fastest ACK response, this Timer can be programmed to 000h, resulting in one DLLP ACK transmitted immediately per each TLP received. For less impact on Transmit TLP bandwidth, a larger value can be programmed, resulting in less-frequent ACKs.

In general, a slower ACK response does not impact the Receive TLP stream, and aids the TLP Transmit stream. Every PCI Express device contains storage (Retry buffer) for storing TLPs while waiting for ACKs. The amount of Retry buffer storage a device contains is vendor-dependent. The quantity of TLPs the PEX 8748 can store depends upon the type and size of TLPs received. The PEX 8748 holds TLPs in the Retry buffer while waiting for an ACK. At some point, if the Retry buffer storage fills, no new TLPs can be sent until a new received ACK frees up space. In this case, the ACK can become a performance bottleneck.

## 8.2.2 UpdateFC DLLP Policy

An *UpdateFC DLLP* is transmitted in response to a received TLP. The UpdateFC DLLP replenishes the connected device with additional credit, to allow the Transmitter to transmit more TLPs of that type. Each TLP that arrives consumes credit, and eventually, a stream of TLPs consume all the available credit, unless an UpdateFC DLLP provides additional credit. However, if the connected device has sufficient credit to transmit more TLPs, it is not necessary to transmit UpdateFC DLLPs to it. The UpdateFC policy determines how and when to transmit an UpdateFC DLLP.

There are two parts to the UpdateFC policy – frequency of transmitting the updates and credit amount. This section discusses only the frequency.

If the PEX 8748 is not transmitting TLPs (*that is*, the Transmit Link is idle), and credit to replenish the credit used becomes available, the PEX 8748 immediately transmits an UpdateFC DLLP to the connected device. This is an opportunistic, low-priority UpdateFC DLLP.

However, if the PEX 8748 is busy transmitting TLPs to the connected device, the PEX 8748 does not transmit an UpdateFC DLLP until a programmed threshold is crossed. The PEX 8748 provides four threshold options – 100%, 75% (default), 50%, and 25%. Whenever the remaining credit drops below the programmed threshold, an UpdateFC DLLP is given high priority, meaning that the UpdateFC DLLP is transmitted before a new TLP is started. There is a separate threshold for Header and Payload credits, for each TLP type – Posted, Non-Posted, and Completion – located in the **Ingress Credit Handler (INCH) Threshold** registers (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offsets [A00h](#) through [A08h](#)).

## 8.2.3 Unidirectional DLLP Policies

For unidirectional traffic, the PEX 8748 DLLP policies allow the most-frequent DLLPs, because DLLPs do not interfere with TLPs. (DLLPs flow in the opposite direction of TLPs.)

The PEX 8748 can transmit a DLLP ACK almost immediately upon receiving and verifying a TLP. A faster ACK results in fast Transmitter de-allocation of the TLP, and can therefore allow a shallow TLP Replay buffer. The default values can be overwritten, to increase or decrease the ACK DLLP rate. For unidirectional traffic, a small number, *such as* 1, is recommended.

The number programmed into the **ACK Transmission Latency Limit** register *ACK Transmission Latency Limit* field (offset [FA8h\[11:0\]](#)) Sets the ACK Transmission Latency Timer, to count the quantity of symbol times after receiving a TLP, before transmitting an ACK.

Similar to the ACK programmability, the PEX 8748 can immediately transmit an UpdateFC after receiving only the TLP Header. By transmitting an UpdateFC earlier, the total credit advertised can be minimized. By programming fewer credits and having a fast UpdateFC policy, the system does not run out of credits and the PEX 8748 does not waste Buffer space on reservations that do not arrive. The following are the recommended Settings:

- Set the UpdateFC policy for unidirectional traffic to 100%
- Set the credits to be sufficient to allow 3 to 4 TLPs



## 8.3 Latency

*Latency* is the length of time it takes to proceed from one event to another. Latency can be measured in several different ways, but perhaps the most common measurement for a switch is *Start TLP-to-Start TLP (STP-to-STP) latency*. Figure 8-1 illustrates an STP-to-STP Latency Measurement. When the Egress Start TLP symbol is transmitted out of a switch before the Ingress Port End symbol arrives, the transfer is termed *Cut-Thru*. If an Egress Port queue is not already established, the PEX 8748 always cuts the packet through. The PEX 8748 has the same latency, regardless of whether the traffic is Upstream or peer-to-peer.

As expected with the PEX 8748 Cut-Thru architecture, STP-to-STP latency is basically constant for all Payload sizes. A faster Link can receive the Header for decode faster, with a slightly lower latency. There will generally be a constant latency for any ingress width to the same egress width, or any ingress width to a smaller egress width, operating at the same Link speed.

For cases in which the egress Port has a higher bandwidth than the ingress Port, then a fraction of the packet, given by the following formula:

$$F = (E - I) / E$$

must be buffered (to prevent under-run in the middle of the TLP), before the TLP can be forwarded to the egress Link.

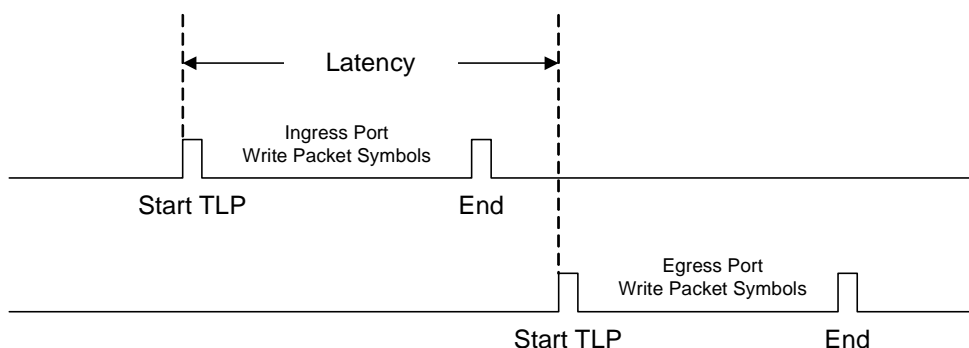
where

- $F$  is the fraction sum
- $E$  is egress bandwidth
- $I$  is ingress bandwidth

For example, with an ingress x4 Port and egress x16 Port,  $(16 - 4) / 16 = 3/4$  of the packet must be buffered before it can be forwarded. Moving in the opposite direction, with an ingress x16 Port and egress x4 Port, the ingress Port has more bandwidth than the egress Port; therefore, buffering is not needed, and the packet can immediately be cut through.

Latency is often important when a Downstream device is reading Main memory, Upstream, because the reader has a finite number of Reads that it can send. Often in this case, the Upstream Port is wider than the Downstream Port and, as a result, latency through the PEX 8748 is optimal in both directions. The Memory Read Request (MRd) that is transmitting Upstream is quite short (12 to 20 bytes); therefore, the buffering required for going from narrower to wider Link widths does not significantly increase latency. The resultant Completion with Data (CplD), with a Payload returning Downstream, is a significantly larger packet; however, the wider to narrower Link width rule applies, and therefore no buffering is needed before forwarding the packet.

**Figure 8-1. Start TLP-to-Start TLP Latency Measurement**



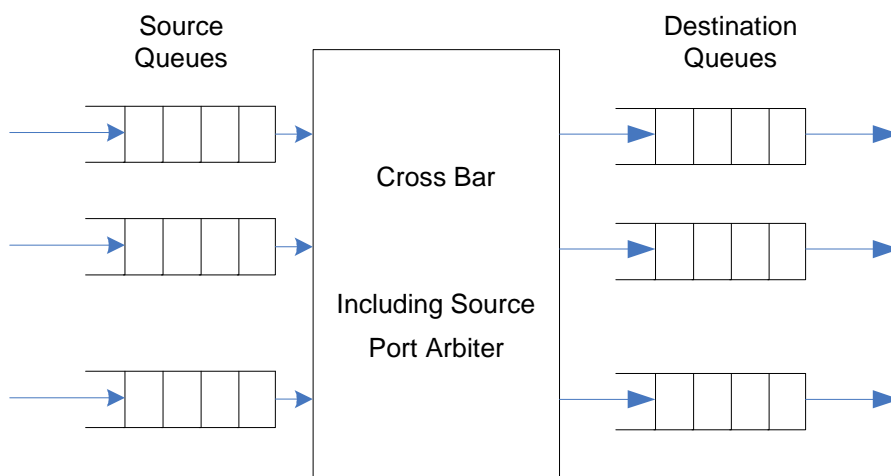
## 8.4 Queuing Options

On-chip queuing does not exist in balanced bandwidth scenarios, where the total ingress bandwidth is less than or equal to the egress bandwidth. In the common case, where the total ingress bandwidth is greater than the egress bandwidth, queues develop on the PEX 8748. The PEX 8748 provides two alternatives, as to where to locate such queuing (refer to [Figure 8-2](#)):

- **Destination queue** – Associated with a single destination Port. All the TLPs in a Destination queue will egress out the same Port.
- **Source queue** – Associated with a single ingress Port. All the TLPs in a Source queue come from the same Port.

Each queue is discussed in the sections that follow.

**Figure 8-2. On-Chip Queuing**



## 8.4.1 Destination Queuing

**Note:** For the queuing examples provided in this section, “Port 1” indicates “first Port,” not the Port physically identified as Port 1.

The default behavior is for all queues to develop at the destination Port. If TLPs are arriving from four sources to a common destination Port, the TLPs are scheduled according to First-In, First-Out (FIFO). The crossbar can forward a TLP every 4 ns, to each Destination queue; therefore, it is unlikely that a Source queue can develop or last very long.

A Destination queue develops whenever the *ingress rate* – the sum of all ingress Ports targeting a destination Port – exceeds the egress rate. A Destination queue might also develop in a credit-starved situation, where there is no credit available to forward TLPs.

For example, if TLPs arriving from four sources all go to a common destination Port, the TLPs are scheduled, based upon the order in which they arrive at the Destination queue FIFO<sup>a</sup>. If all four flows are equally active, the TLPs naturally interleave as 1,2,3,4,1,2,3,4. If three of the Ports, however, have a head start before the fourth Port turns On, the output can be 1,2,3,1,2,3,1,2,3,1,2,3,1,2,3,4,1,2,3,4. In this case, all the new Port (Port 4) TLPs must wait for the earlier Port 1,2,3 traffic to be transferred before the Port 4 TLPs can be transferred. Therefore, the latency for Port 4 traffic to travel through the PEX 8748 can widely vary, based upon the traffic passing through the switch.

---

a. Conventional PCI Strong Ordering rules can override the FIFO. Conventional PCI requires Posted TLPs to be able to pass Non-Posted and Completion TLPs, to avoid deadlock.

## 8.4.2 Source Queuing

**Caution:** *Source Queuing and [Read Pacing](#) should not be concurrently enabled. The two features are incompatible and doing so can result in Fatal errors.*

**Note:** *For the queuing examples provided in this section, “Port 1” indicates “first Port,” not the Port physically identified as Port 1.*

Source queuing can be enabled for applications that require deterministic bounded latency for traffic that flows to a single Port.

Source queuing limits the Destination queue depth by applying an upper and lower threshold. When the Destination queue reaches the programmed upper threshold, any subsequent TLPs targeting that destination Port are not forwarded, but instead are queued up in a per-Source Port-based queue. The Source Port queue does not begin to forward TLPs again (to any Port), until its first entry’s destination Port’s queue drops to a programmed lower threshold.

**Note:** *A Source Port queue that cannot forward to a Destination queue blocks all subsequent TLPs arriving on that same Source Port, even if those subsequent TLPs target different destination Ports.*

The **Port Egress TLP Threshold** register (offset [F38h](#)) controls the upper and lower queue thresholds. [Table 8-1](#) summarizes the register bit Settings. The Port Lower TLP Counter is the quantity of TLPs that the Destination queue must reach after becoming saturated, before re-enabling TLP forwarding. The Port Upper TLP Counter is the quantity of TLPs that can be queued in the Destination queue.

In the Destination queue example provided in [Section 8.4.1](#), the early arriving Port 1,2,3 TLPs stalled Port 4’s TLP for an indeterminate length of time. By programming, with source queuing enabled and a Destination Port Lower TLP Counter programmed to 1 and Port Upper TLP Counter programmed to 3 (TLPs), the worst case is that Port 4 must wait for three TLPs (1,2,3) before getting its first turn. With these Settings, the example TLP output would be 1,2,3,4,1,2,3,4,1,2,3,4. The *turn to be forwarded* refers to a Port Arbitration wait, described in [Section 8.4.3](#).

To avoid unnecessary idles on the destination Link, program a Port Lower TLP Counter of 1, and a Port Upper TLP Counter of 2.

**Table 8-1. Port Egress TLP Threshold Register Port Lower and Upper TLP Counters (Offset F38h)**

Bit(s) <sup>a</sup>	Name	Description
11:0	<b>Port Lower TLP Counter</b>	When Source Scheduling is disabled due to Threshold, it is re-enabled when the Port TLP Counter goes below this Threshold value.
27:16	<b>Port Upper TLP Counter</b>	When the Port TLP Counter is greater than or equal to this value, the Source Scheduler disables TLP scheduling to this egress Port.

a. Bits not identified in [Table 8-1](#) are **Reserved**.

### 8.4.3 Port Arbitration

*Port Arbitration* refers to the arbitration at an egress Port of a switch between other ingress Ports. In the PEX 8748, a different Port Arbiter connects all the Source queues to each Destination queue. The PEX 8748 supports, on a per-Port basis, either a hardware-fixed Round-Robin (RR) arbitration scheme or a 64-slot Weighted Round Robin (WRR) arbitration scheme, the assignment of which is a bit complicated and will be discussed at the end of this section.

The Port Arbiter has the most effect when source queuing is enabled (refer to [Section 8.4.2](#) for details), and TLPs from multiple ingress Ports are enqueued such that the Port Arbiter has choices about which TLP to forward. Every egress Port has its own independent Port Arbiter state machine.

Port arbitration in the PEX 8748 makes decisions on a per-TLP basis. As a consequence, the throughput may not look fair for two otherwise equal Ports if the average TLP size of each is different. *For example*, if two ingress Ports, PortA and PortB, target the same egress Port and both have the same weight, but PortA has short TLPs while PortB has long TLPs, PortB will appear to get a higher throughput. The seeming unfairness can be compensated for, by using the WRR Arbiter, and by adjusting the weights of the Ports to be in a ratio similar to the ratio of TLP size.

The RR Port Arbiter scheme decides which ingress Port to forward to a single egress Port, with all ingress Ports having the same weight. All Downstream Ports, by default, use an RR Port Arbiter.

The WRR Port Arbiter scheme also decides which ingress Port to forward to a single egress Port; however, the ingress Ports may have different weights. The initial weights are based upon the negotiated Link width. With 64 slots, the resolution of the weights goes down to a x1 Gen 3 Link (8 GT/s). Therefore, a x1 Gen 3 Link would have 1 slot, while a x8 Gen 3 Link would have 8 slots, and so forth. A x4 Gen 1 Link, x2 Gen 2 Link, and x1 Gen 3 Link all have the same weight, because each has the same throughput capability of approximately 8 GT/s.

Software can change the default WRR weights to any value. The WRR Port Arbiter uses the *PCI Express Base r3.0*-defined Port Arbitration Capability structure, as reflected in the Port's **VC0 Resource Capability** register *Port Arbitration Capability* field (offset 158h[2:0]), with the Port's **VC0 Resource Control** register *Port Arbitration Select* field (offset 15Ch[19:17]) programmed to 010b.

Software performs two steps to change the WRR weights.

In the first step, the weights are programmed in the **Port Arbitration Table Phase x** registers (offsets 178h through 1B4h). (Refer to the *PCI Express Base r3.0*, as well as [Section 13.15.1](#), “WRR Port Arbitration Table Registers (Offsets 178h – 1BCh),” for further details.)

The final step requires a write to the Port's **VC0 Resource Control** register *Load Port Arbitration Table* bit (offset 15Ch[16]). When written, the weights programmed in the **Port Arbitration Table Phase x** registers in the first step are transferred to the WRR arbitration logic and take effect. Before writing the *Load Port Arbitration Table* bit, the **Port Arbitration Table Phase x** registers have no effect.

The selection of which egress Port gets a WRR Port Arbiter is limited, due to hardware constraints. The main constraint is that only one Port of a Port group can use the WRR. The Port groups are as follows:

- Group 0: Ports 0, 2
- Group 1: Ports 1, 3
- Group 2: Ports 8, 10
- Group 3: Ports 9, 11
- Group 4: Ports 16, 18
- Group 5: Ports 17, 19

Each Port group has one WRR that is always enabled. As previously mentioned, the WRR defaults to weights based upon the negotiated Link width of each ingress Port.

Only one Port per group, however, can use the WRR capability. The decision algorithm looks at whether any Port is Upstream, in which case it “wins” use of the WRR capability. If there is no Upstream Port, then the first Port in the group gets WRR capability. Thus, a Downstream Port can use only the first Port within a group (Port 0, 1, 8, 9, 16, or 17).

In Virtual Switch mode, where there can be multiple Upstream Ports within the same group, the last Port within the group wins use of the WRR capability. *For example*, if the Upstream Port is Port 3, it uses the Group 1 WRR Port Arbiter, relegating (enabled) Port 1 to an RR Port Arbiter. The only way for each Upstream Port to get a WRR Port Arbiter is to choose Upstream Ports that belong to different Port groups.

## 8.4.4 Port Bandwidth Allocation

For applications that need to allocate a fixed bandwidth to each Port, the PEX 8748 can help enforce the relative bandwidth ratio between Ports in a congested scenario.

By combining source queuing, Port Arbitration, and initial credit, as well as some knowledge of average Payload size, many combinations of Port bandwidth allocation are possible.

## 8.5 Read Pacing

***Note: Source Queuing and Read Pacing should not be concurrently enabled. The two features are incompatible and doing so can result in Fatal errors.***

The Read Pacing feature is supported on all Ports. The Read Pacing Configuration registers, however, are implemented as follows – Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port.

PCI Express has a weakness concerning the number of outstanding bytes requested by Reads. It is possible that a single device can overwhelm the system with a relatively small number of large Read Requests, thereby impacting the performance of other connected devices, by filling the ingress transaction queue in the Root Complex.

The Root Complex must handle the transactions in the order in which they are posted. Transactions posted from less aggressive reading devices, which may be more sensitive to latency, suffer performance reductions due to the unfairly weighted path (head of line blocking) in the transaction queue that the large reads represent.

Read Pacing attempts to apply some rules to Memory Read Requests, so that no one Port can overwhelm a system. There are two aspects to the PEX 8748's Read Pacing capability:

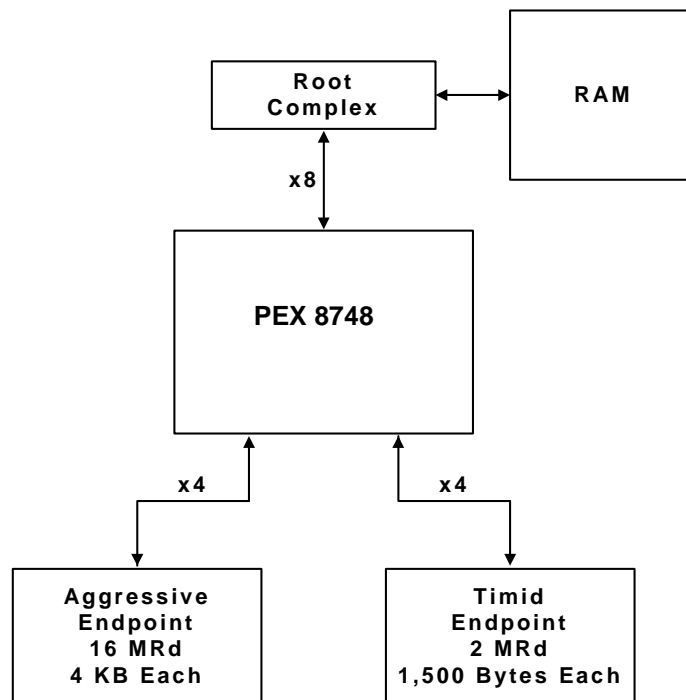
- Read spacing
- Read threshold

The following sections provide examples and further information regarding Read Pacing.

## 8.5.1 Read Pacing Example

Figure 8-3 illustrates an example of a system that benefits from Read Pacing.

Figure 8-3. Read Pacing Example





In a typical Host-centric application, endpoints have Direct Memory Access (DMA) engines that write to and read from Main memory. A performance bottleneck can occur during the Read to Main memory, through the Root Complex. For the example illustrated in Figure 8-3, the aggressive endpoint sends many large (16, 4-KB) Memory Read Requests, while another endpoint, or Timid Endpoint (TEP), sends only two 1,500-byte Memory Read Requests. The TEP then waits for a response before sending additional Read Requests<sup>a</sup>.

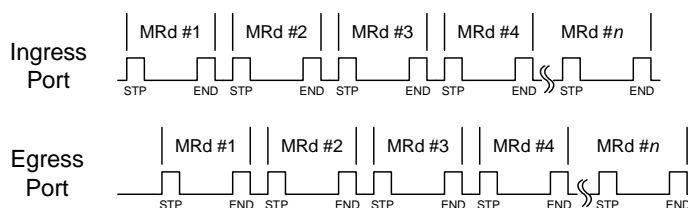
If either endpoint is running by itself, neither sees a problem. However, if both endpoints are concurrently active, the aggressive endpoint dominates the Root Complex Memory Controller. In addition, due to the bandwidth mismatch, Completions can queue up in the PEX 8748, creating too many Completions for the switch to store at one time. As a result, the PEX 8748 backpressures the Root Complex for Completions. The Root Complex can only forward Completions to the PEX 8748 at the aggressive endpoint's rate, which is significantly less than the Root Complex could otherwise handle.

The net impact is not to the aggressive endpoint, because there are a sufficient number of Completions queued up in the PEX 8748 to keep it busy. In fact, the aggressive endpoint experiences better performance with a switch, than connected directly to the Root Complex<sup>b</sup>. Rather, the TEP experiences lower performance results. Its Memory Read Requests wait in line behind multiple aggressive endpoint Requests, and the Root Complex can drain Requests only at the same rate of the PEX 8748, not at the Upstream Link's capacity.

Figure 8-4 illustrates how a PCI Express switch, without Read Pacing, forwards MRds.

Read Pacing solves the performance loss seen by the TEP, while improving the aggressive endpoint's performance. The following sections provide examples of the way in which the PEX 8748 functions when Read Pacing is enabled, and Read Spreading is enabled or disabled.

**Figure 8-4. Read Pacing Off (Disabled)**



a. This is based upon an actual setup in a third-party lab. Fibre Channel endpoints can easily send 16, 4-KB MRd at a time, while Gigabit Ethernet endpoints might send only one or two 1,500-byte endpoints at a time.

b. Without a switch, when the Root Complex has something else to do, the aggressive endpoint loses its data stream. With a switch, the buffering of multiple Completions hides the fact that the Root Complex is multitasking.

## 8.5.2 Read Spacing (Spreading) Logic

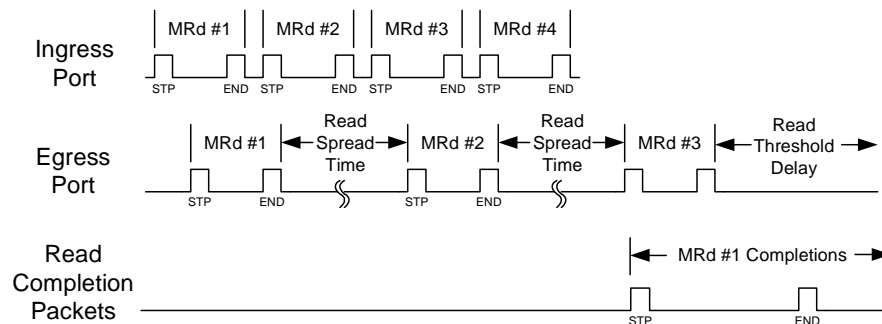
Read Spacing (also referred to as *Read Spreading*) spreads out Read Requests. The PEX 8748 Read Spacing logic looks at the Read Request size and the endpoint's bandwidth, to determine how often to forward subsequent Read Requests. *For example*, Read Requests arriving on a x4 Link can only sink data at a x4 rate. If a x4 endpoint submits multiple Read Requests to a x8 Link, the Read Spacing logic does not forward the subsequent Read Requests until the endpoint has sufficient time to sink a portion of the Completion data from the previous Read Requests.

Initially, a queue of Completions must build up to hide the time that it takes for the data to return. As a result, Reads are forwarded at 2x the endpoint's bandwidth. This 2x rate is maintained until a threshold of outstanding Read data is reached, at which time Reads are forwarded at 1x the endpoint's bandwidth.

**Read Pacing must be enabled for Read Spreading to be enabled.** That is, for a Port to have Read Spreading enabled, the Port's **Read Pacing Control** register *Port x Read Pacing Disable* and *Port x Memory Read Spreading Disable* bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 1D0h[1:0 and 17:16], respectively) must both be Cleared.

Figure 8-5 illustrates the way in which the PEX 8748 forwards Read Requests when the Read Pacing- and Read Spreading-related bits are enabled. (Refer to Section 8.5.5 for additional register/bit information.) The PEX 8748 continues to spread and forward the Read Requests, until the amount of Completion data for which it is waiting exceeds the value programmed in the **Read Pacing Threshold x** register(s) for that Link width (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offsets 1D4h and 1D8h).

**Figure 8-5. Read Pacing On (Enabled) and Read Spreading On (Enabled)**

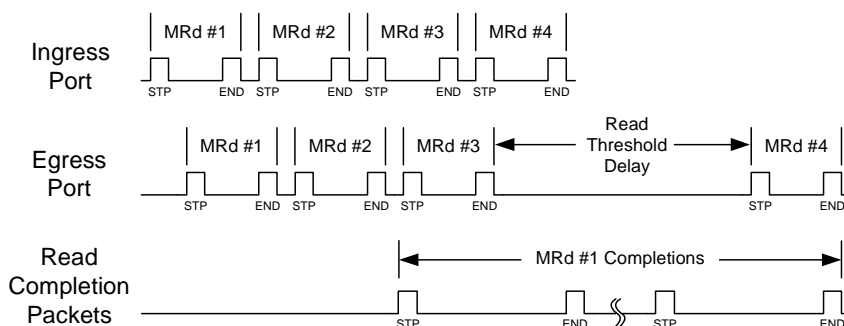


### 8.5.3 Read Threshold

The Read threshold is the maximum number of outstanding DWords (1 DWord = 4 bytes) that the endpoint Port requested to be read, but were not yet returned as Completion data. The threshold is related to the PEX 8748's buffering capacity – all outstanding Read data ought to be able to be buffered in the switch, to remain out of the way of other Completions for other endpoint's Read Requests.

After a Port reaches its Read threshold, subsequent Read Requests from that Port queue up in the PEX 8748, waiting for Completion data to reduce the outstanding count to below the threshold. If an overabundant number of Read Requests queue in the PEX 8748, no additional Read credit is allocated, which backpressures the Read Requester. Figure 8-6 illustrates the way in which the PEX 8748 forwards Read Requests when its Read Spacing logic is enabled and Read Spreading logic is disabled.

**Figure 8-6. Read Pacing On (Enabled) and Read Spreading Off (Disabled)**



### 8.5.4 Read Pacing Benefits

When Read Pacing logic is enabled, the PEX 8748 provides the follow benefits:

- Maximum Read latency that an endpoint may experience can be dramatically reduced.  
By reducing the amount of queued Read Requests, and therefore pending Read Completion data at the Root Complex, new Read Requests from Ports that do not have pending Read Requests can be serviced with a predictable and/or reasonable amount of latency.
- Timid endpoint bandwidth is dramatically increased in busy applications.  
Because queues of pending Read Requests in the Root Complex are limited, and congestion caused by a large amount of Completion data intended for a high-bandwidth, needy Port (or Ports) is avoided, the bandwidth needs of endpoints with smaller bandwidth requirements are met (*that is*, the endpoints are not starved).
- PEX 8748's Read Pacing Threshold logic allows all busy Ports to be equally serviced in congested scenarios, regardless of their individual Read requesting behavior.

*For example*, all Ports might simultaneously request data, some aggressively and some timidly. While unable to quickly drain their queued Completions, the Ports' Read Pacing Threshold logic forwards the additional Read Requests to the Root Complex, equally and fairly, while ensuring Completion data is available for each Port, when the Port is ready to accept it.

## 8.5.5 Enabling Read Pacing and Read Spreading

Read Pacing is disabled, by default. To enable Read Pacing, the Port's **Read Pacing Control** register *Port x Read Pacing Disable* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 1D0h[1:0]) must be Cleared. Each bit corresponds to a range of Ports. A value of 0 enables Read Pacing for the Ports associated with the bit, whereas a value of 1 (default) disables Read Pacing for the Ports associated with the bit.

The Port's **Read Pacing Control** register *Port x Memory Read Spreading Disable* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 1D0h[17:16]) is used to enable or disable Read Spreading. Each bit corresponds to a range of Ports. A value of 1 disables Read Spreading for the Ports associated with the bit. Read Spreading is enabled, by default (value of 0); however, it is overridden by the Port's *Port x Read Pacing Disable* bit, by default.

Both sets of Read Spreading and Pacing Control register bits are represented in Table 8-2. (For complete details, refer to the register offset 1D0h description provided in Section 13.16.1, “Device-Specific Registers – Read Pacing (Offsets 1D0h – 1D8h).”) Figure 8-4 through Figure 8-6 illustrate what occurs when the bits are enabled or disabled.

The Read Pacing thresholds are Set, based upon the Source Port's programmed Link width. The **Read Pacing Threshold 1** register controls the threshold values for x16 and x8 Link width threshold values, and the **Read Pacing Threshold 2** register controls the x4 and x2 Link width threshold values (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offsets 1D4h and 1D8h, respectively). The thresholds are in DWords. Narrower Link widths have lower thresholds, because they must buffer smaller quantities.

**Table 8-2. Read Pacing Control Register Read Pacing- and Memory Read Spreading Disable Bits (Offset 1D0h)**  
(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Default
1:0	<b>Port x Read Pacing Disable</b> 0 = Read Pacing is enabled for this Port 1 = Read Pacing is disabled for this Port	11b
17:16	<b>Port x Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for this Port 1 = Memory Read Spreading is disabled for this Port	00b

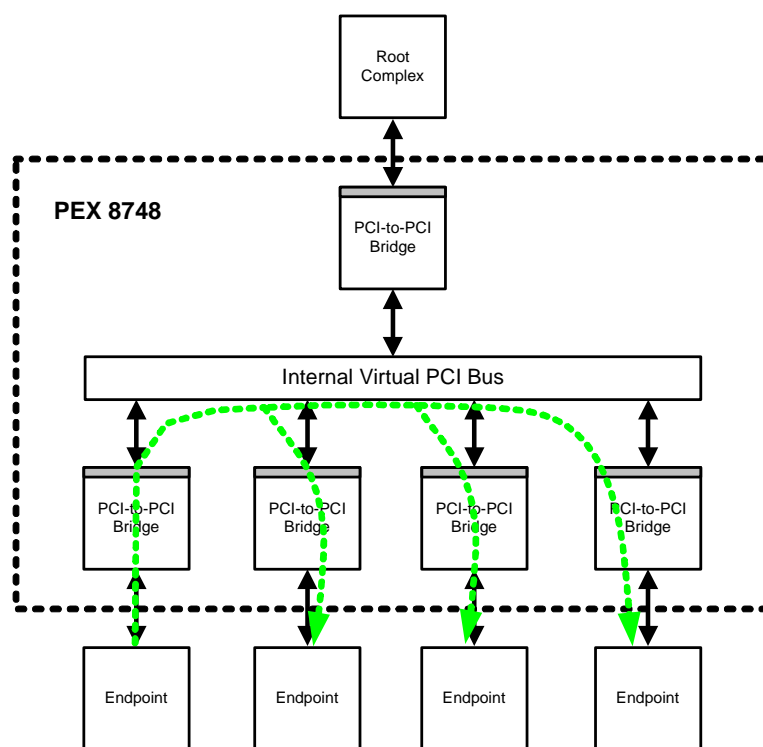
## 8.6 Multicast

This section describes the functions and registers of the Multicast (MC) feature.

Multicast allows software to concurrently write the same data to a group of multiple destinations. When a Posted Memory Write TLP entering the PEX 8748 is addressed to the MC Address Range (*MC BARs*), and Multicast is enabled, the PEX 8748 automatically generates and transmits a copy of the original TLP (referred to as the *MC Copy TLP*) to one or more destination Port(s). The MC Address Space is divided into *MC Groups (MCG)*, defined by using *MC Base Address* and *MC Index Position*. Each PEX 8748 Port can elect to receive an MC Copy TLP by belonging to an *MCG*, by Setting the corresponding *MC Receive* bit. An MC TLP can be blocked by using *MC Block All*, if required. *MC Overlay Bar* can be used to replace the original MC TLP's address to a Unicast address space, if the endpoint does not support MC.

Figure 8-7 illustrates an example of a Multicast operation on a peer-to-peer Write Request. An endpoint's Write Request is transmitted as copies to additional peer destination endpoints.

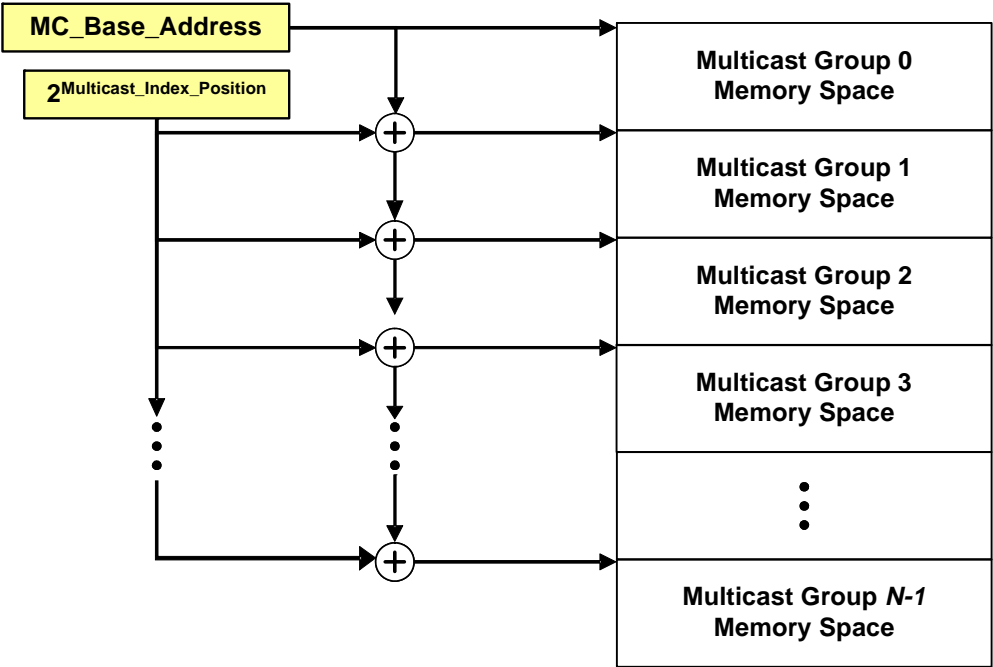
**Figure 8-7. Peer-to-Peer Multicast**



8.6.1 Multicast Address Range Segmentation

The **Multicast Extended Capability** structure defines an MC Address range, the segmentation of that range into a number, *N*, of equally sized MC windows, and the association of each MC window with an MCG, as illustrated in Figure 8-8. Each function that supports MC within a component implements a **Multicast Extended Capability** structure that provides routing directions and permission checking for each MCG for TLPs that pass through. The MCG is a field of up to 6 bits in width, which is embedded in the address, beginning at the MC\_Index\_Position.

Figure 8-8. Multicast Address Range Segmentation



## 8.6.2 Multicast TLP Processing

A TLP is processed as an MC TLP if an MC Hit occurs when all of the following conditions are true:

- MC\_Enable is Set
- TLP is a Memory Write (Posted Request)
- TLP Address  $\geq$  MC\_Base\_Address
- TLP Address  $< (MC\_Base\_Address + (MC\_Index\_Position^2) \times (MC\_Num\_Group + 1))$

While processing the TLP, each PEX 8748 ingress Port uses values of MC\_Enable, MC\_Base\_Address, MC\_Index\_Position, and MC\_Num\_Group from its registers. The software is required to identically configure all these fields in all Ports. If this is not the case, results are indeterminate.

If an MC Hit occurs, standard address routing rules do not apply. Instead, the TLP is processed by first extracting the MCG from the address in the TLP, using the ingress Port's values for MC\_Base\_Address and MC\_Index\_Position. Specifically:

$$MCG = ((Address_{TLP} - MC\_Base\_Address) \gg MC\_Index\_Position) \& 3Fh$$

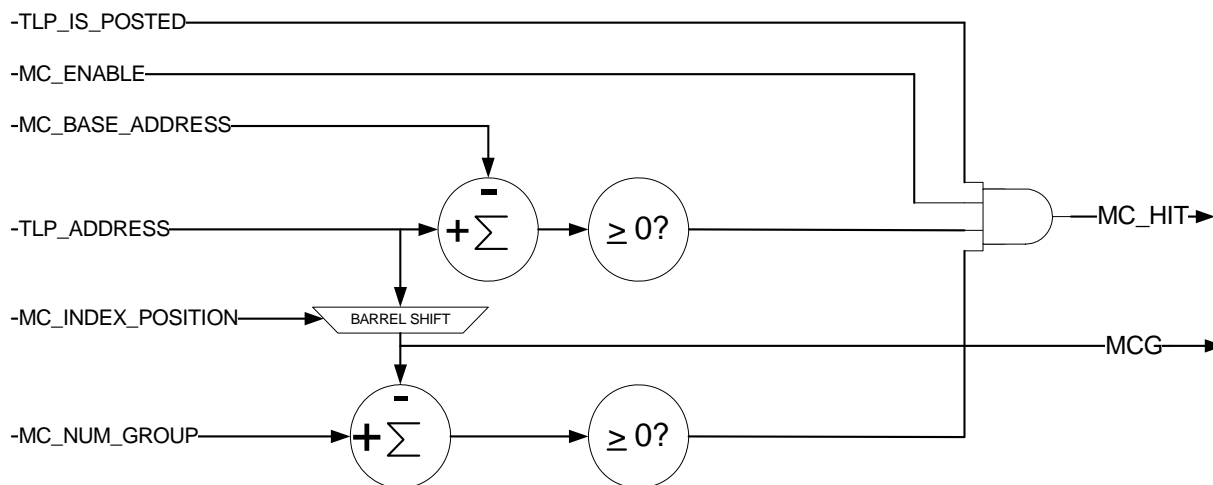
Next, the PEX 8748 checks the MC\_Block\_All and MC\_Block\_Untranslated bits corresponding to the extracted MCG, using the MC\_Block\_All and MC\_Block\_Untranslated registers associated with the ingress Port. If the MC\_Block\_All bit corresponding to the extracted MCG is Set, the TLP is handled as an MC Blocked TLP. If the MC\_Block\_Untranslated bit corresponding to the extracted MCG is Set, and the TLP contains an Untranslated Address, the TLP is also handled as an MC Blocked TLP.

If the TLP is not blocked in the PEX 8748, it is forwarded out of all the Ports, with the exception of its ingress Port, whose MC\_Receive bit corresponding to the extracted MCG is Set. If no Ports forward the TLP, the TLP is silently dropped.

Figure 8-9 illustrates Multicast processes TLPs.

**Note:** To prevent loops, it is prohibited for a PEX 8748 Port to forward a TLP back out through its ingress Port, regardless of whether it is specified to do so, by the Port's MC\_Receive register.

**Figure 8-9. Multicast TLP Processing**



An MC Hit suspends standard address routing, including default Upstream routing in the PEX 8748. When an MC Hit occurs, the TLP is forwarded out only through those egress Ports whose *MC\_Receive* bit associated with the MCG extracted from the address in the TLP is Set. The PEX 8748 attempts to simultaneously broadcast the packets. However, if the Egress queues are not empty, the packet remains queued for each egress Port, because ordering must be maintained. Therefore, the MC packets could be transmitted at different times, depending upon the traffic flow.

If the address in the TLP does not decode to a Downstream Port using standard address decode, the TLP is copied to the Upstream Port, if specified to do so, by the Upstream Port's **MC\_Receive** register.

If the address in a Non-Posted Memory Request hits in an MC window, no MC Hit occurs, and the TLP is processed as a Unicast.

If an MC Hit occurs, the only ACS access control that can apply is ACS Source Validation. In particular, neither ACS redirection nor the ACS Egress Control Vector affects operations during an MC Hit.

### 8.6.3 Multicast Ordering

No new ordering rules are defined for processing MC TLPs. All MC TLPs are Posted Requests and follow Posted Request ordering rules. MC TLPs are ordered per standard ordering rules, relative to other TLPs in a component's ingress stream, through the point of replication. Once copied into an egress stream, an MC TLP follows the same ordering as other Posted Requests in the stream.

### 8.6.4 Multicast Extended Capability Structure Field Updates

Certain fields of the **Multicast Extended Capability** structure can be changed at any time. Others cannot be changed with predictable results, unless the MC\_Enable bit is Cleared in every component function. The latter group includes MC\_Base\_Address and MC\_Index\_Position. Fields that software can change at any time include MC\_Enable, MC\_Num\_Group, MC\_Receive, MC\_Block\_All, and MC\_Block\_Untranslated. Updates to these fields must be ordered.

*For example*, TLPs *A* and *B* arrive in that order, at the same ingress Port, and in the same TC. If *A* uses *X* for one of these fields, then *B* must use the same value or a newer value.

### 8.6.5 MC Blocked TLP Processing

When a TLP is blocked by the MC\_Block\_All or MC\_Block\_Untranslated mechanisms, the TLP is dropped. The ingress Port blocking the TLP serves as the Completer. The ingress Port logs and signals this MC Blocked TLP. In addition, the ingress Port Sets the **PCI Status** register *Signaled Target Abort* bit (offset 04h[27]) or **Secondary Status** register *Signaled Target Abort* bit (offset 1Ch[27]), as appropriate. If the error occurs with a TLP received by an ingress Port, the error is reported by that ingress Port.



## 8.6.6 MC\_Overlay and ECRC Re-Generation

The MC Overlay mechanism is provided to allow a single BAR in an endpoint that does not contain a **Multicast Extended Capability** structure to be used for both MC and Unicast TLP reception. Software can configure the MC\_Overlay mechanism to affect this, by Setting the MC\_Overlay\_BAR in a Downstream Port of the PEX 8748, so that the MC Address range, or a portion thereof, is re-mapped (overlaid) onto the Memory Space range accepted by the endpoint's BAR. At the Upstream Port, the mechanism can be used to overlay a portion of the MC Address range onto a Memory Space range associated with Host memory.

When enabled, the overlay operation specifies that MC TLP Address bits, whose bit numbers are greater than or equal to the MC\_Overlay\_Size field, be replaced by the corresponding MC\_Overlay\_BAR bits. *That is:*

```
If (MC_Overlay_Size = 0)
Then:
    Egress_TLP_Addr = Ingress_TLP_Addr;
Otherwise:
    Egress_TLP_Addr =
        {MC_Overlay_BAR[63:MC_Overlay_Size], Ingress_TLP_Addr[MC_Overlay_Size-1:0]};
```

If the TLP with the modified address contains the optional ECRC, the unmodified ECRC will almost certainly indicate an error. The action to be taken if a TLP containing an ECRC is MC copied to an egress Port that has MC\_Overlay enabled, are outlined in [Table 8-3](#). If MC\_Overlay is not enabled, the TLP is forwarded, unmodified. If MC\_Overlay is enabled and the TLP has no ECRC, the modified TLP, with its address replaced as specified in the previous paragraph, is forwarded. If the TLP has an ECRC but ECRC re-generation is not enabled, then the modified TLP is forwarded with its ECRC dropped and the *TD* bit in the header Cleared, to indicate that no ECRC is attached. If the TLP has an ECRC and ECRC re-generation is enabled, an ECRC is performed before the TLP is forwarded. If the ECRC passes, the TLP is forwarded with a re-generated ECRC. If the ECRC check fails, the TLP is forwarded with an inverted re-generated ECRC.

**Table 8-3. ECRC Rules for MC\_Overlay<sup>a</sup>**

MC_Overlay Enabled	TLP with ECRC	ECRC Re-Generation Supported	Action if ECRC Check Passes	Action if ECRC Check Fails
No	X	X	Forward TLP unmodified.	
Yes	No	X	Forward modified TLP.	
Yes	Yes	Yes	Forward modified TLP with re-generated ECRC.	Forward modified TLP with inverted re-generated ECRC.

a. "X" is "Don't Care."

### 8.6.6.1 Multicast to Endpoints without Multicast Extended Capability

An endpoint function that does not contain a **Multicast Extended Capability** structure cannot distinguish MC TLPs from Unicast TLPs. It is possible to take advantage of this, to use such endpoints as MC targets. The PEX 8748 Port above the device can be configured to overlap at least part of the MC Address range, or the MC\_Overlay mechanism can be used.

### 8.6.6.2 Congestion Avoidance

The use of MC increases the output Link use of switches to a degree proportional to both the size of the MC groups used and the fraction of MC traffic to total traffic, which can increase the risk of congestion and spreading. To mitigate this risk, design components that are intended to serve as MC targets to consume MC TLPs at wire speed. Components intended to serve as MC sources should consider adding a rate-limiting mechanism.

### 8.6.7 Multicast Extended Capability

MC functionality is controlled by the **Multicast Extended Capability** structure. Multiple copies of this structure are required – one for each PEX 8748 Port that supports MC. To provide implementation efficiencies, certain fields within each of the MC Capability structures within a component, must be programmed the same. Results are indeterminate if this is not the case. The fields and registers that must be configured with the same values include MC\_Enable, MC\_Num\_Group, MC\_Base\_Address, and MC\_Index\_Position. These same fields in an endpoint's **Multicast Extended Capability** structure must match those configured within a **Multicast Extended Capability** structure of the PEX 8748 above the endpoint, or in which the Root Complex integrated endpoint is integrated.

## 8.6.8 Multicast with NT – NT Mode

The following subsections describe Multicast behavior in NT mode.

### 8.6.8.1 NT Multicast from NT Port Virtual to NT Port Link Interface Direction (NT Target) – NT Mode

When a PEX 8748 Transparent Port receives a TLP that is a Multicast Hit, the Multicast TLP that is routed to the egress Ports belongs to the extracted Multicast Group. If a virtual side NT endpoint is behind one of the Multicast Targeted Ports, the ingress Port routes the Multicast TLP through the virtual Downstream Port to the NT Port Virtual Interface, if all the following conditions are true:

- Virtual Downstream Port **Command** register *Memory Enable* bit is Set
- If Multicast overlay is not enabled, the Multicast TLP address hits the NT Port Virtual Interface BARs (**BAR2** to **BAR5**, offsets **18h** through **24h**, respectively)
- If Multicast overlay is enabled, the Multicast overlay translated address hits the NT Port Virtual Interface BARs
- NT Port Virtual Interface **PCI Command** register *Memory Access Enable* bit (offset **04h**[1]) is Set
- NT Port Link Interface **PCI Command** register *Bus Master Enable* bit (offset **04h**[2]) is Set
- Multicast TLP Requester ID hits the NT Port Virtual Interface Requester ID Translation LUT

If any one of these conditions is not met, the Multicast TLP to the NT Port is handled as an *Unsupported Request (UR)*. If a Multicast TLP's Traffic Class (TC) value does not map to the Port's **VC0 Resource Control** register *TC/VC Map* bits (offset **15Ch**[7:1 and 0]), the Multicast TLP is handled as a *Malformed TLP*.

The NT Port receives the Multicast TLP with either an un-translated address or Multicast overlay translated address (based upon the virtual Downstream Port MC\_Overlay\_Enable (**Multicast BAR0** and **Multicast BAR0** registers, offsets **E28h** and **E2Ch**, respectively). The NT Port does the Address and Requester ID translation, as defined in the NT Port Virtual Interface **Memory BARx Address Translation** registers (offsets **C3Ch** through **C48h**) and **Requester ID Translation LUT** registers (addresses **D94h** through **DD0h**). The translated Multicast TLP is sent out from the NT Port.

### 8.6.8.2 NT Multicast from NT Port Link to NT Port Virtual Interface Direction (NT Source) – NT Mode

When the PEX 8748 NT Port Link Interface receives a Memory Write TLP, it performs the following checks, before qualifying the TLP as a Multicast Hit:

- TLP hits the NT Port Link Interface BARs (**BAR2** to **BAR5**, offsets **18h** through **24h**, respectively).
- NT Port Link Interface **PCI Command** register *Memory Access Enable* bit (offset **04h**[1]) is Set.
- NT Port Virtual Interface **PCI Command** register *Bus Master Enable* bit (offset **04h**[2]) is Set.
- Virtual Downstream Port **PCI Command** register *Bus Master Enable* bit is Set.
- Received TLP Requester ID hits the NT Port Link Interface Requester ID Translation LUT registers.
- NT Port Link Interface performs Address translation and Requester ID translation, as defined in the NT Port Link Interface **Memory BARx Address Translation** registers (offsets **C3Ch** through **C48h**) and **Requester ID Translation LUT** registers (addresses **DB4h** through **DF0h**). The translated address hits the Multicast BAR in the virtual Downstream's Port **Multicast Extended Capability** structure and satisfies all other Multicast Hit conditions.

When the NT Port Link Interface receives a TLP that is a Multicast Hit, and the Multicast destination Port(s) **PCI Command** register *Memory Access Enable* or *Bus Master Enable* bit (based upon the traffic direction) is not Set, the Multicast TLP is handled as a UR for the corresponding destination Port(s).

When the NT Port Link Interface receives a TLP that is a Multicast Hit, and TLP TC value does not map to the enabled Virtual Channel, the Multicast TLP is handled as a Malformed TLP for the corresponding destination Port(s).

A Multicast TLP received at the NT Port Link Interface is forwarded with address translation and Requester ID translation to the Multicast destination Port(s). Multicast destination Port(s) transmit the TLP, either un-modified or with MC\_Overlay address translation (based upon MC\_Overlay\_Enable).



## Chapter 9 Interrupts

### 9.1 Interrupt Support

The PEX 8748 supports the PCI Express interrupt model, which uses two mechanisms:

- INTx Interrupt Message-type emulation (compatible with the *PCI r3.0*-defined Interrupt signals)
- Message Signaled Interrupt (MSI), when enabled

For Conventional PCI compatibility, the PCI INTx emulation mechanism is used to signal interrupts to the System Interrupt Controller. This mechanism is compatible with existing PCI software, provides the same level of service as the corresponding PCI interrupt signaling mechanism, and is independent of System Interrupt Controller specifics. The PCI INTx emulation mechanism virtualizes PCI physical Interrupt signals, by using an in-band signaling mechanism, for the assertion and de-assertion of INTx interrupt signals.

In addition to PCI INTx-compatible interrupt emulation, the PEX 8748 supports the MSI mechanism. The PCI Express MSI mechanism is compatible with the MSI Capability defined in the *PCI r3.0*.

INTx and MSIs are mutually exclusive, on a per-Port basis; either can be enabled in a system (depending upon which interrupt type the system software supports), but never concurrently within the same domain. (Refer to the **PCI Command** register *Interrupt Disable* bit, offset 04h[10], and **MSI Control** register *MSI Enable* bit, offset 48h[16], respectively.) The PEX 8748 does not convert received INTx Messages to MSI Messages.

The PEX 8748's external Interrupt output, **PEX\_INTA#** (Base mode) or **VSx\_PEX\_INTA#** (Virtual Switch mode), indicates the assertion and/or de-assertion of the internally generated INTx signal:

- **Non-Hot Plug-triggered interrupts** – PEX\_INTA# or VSx\_PEX\_INTA# assertion is controlled by the following **ECC Error Check Disable** register bits:
  - *Enable PEX\_INTA# or VSx\_PEX\_INTA# Interrupt Output for GPIO-Generated Interrupts* (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 720h[6])
  - **NT mode only**
    - *Enable PEX\_INTA# or VSx\_PEX\_INTA# Interrupt Output for NT-Virtual Doorbell-Generated Interrupts* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 720h[7])
    - *Enable PEX\_INTA# or VSx\_PEX\_INTA# Interrupt Output for Device-Specific NT-Link Port Event-Triggered Interrupts* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 720h[5])
  - **Virtual Switch mode only**
    - *Enable VSx\_PEX\_INTA# Interrupt Output for Management Link Status Event-Generated Interrupts* bit (Virtual Switch mode – Port 0, accessible through the Management Port, offset 720h[9])
    - *Enable VSx\_PEX\_INTA# Interrupt Output for Management Port Doorbell-Generated Interrupts* bit (Virtual Switch mode – Port 0, accessible through the Management Port, offset 720h[8])

When any of these bits are Set, Device-Specific errors trigger PEX\_INTA# or VSx\_PEX\_INTA# assertion; however, PEX\_INTA# or VSx\_PEX\_INTA# assertion and INTx Message generation are mutually exclusive, on a per-Port basis.

- **Hot Plug or Link State-triggered INTx events** – PEX\_INTA# or VSx\_PEX\_INTA# assertion is controlled by the **ECC Error Check Disable** register *Enable PEX\_INTA# or VSx\_PEX\_INTA# Interrupt Output for Hot Plug or Link State Event-Triggered Interrupts* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 720h[4]). When this bit is Set, Hot Plug or Link State events trigger PEX\_INTA# or VSx\_PEX\_INTA# assertion; however, an INTx Message is not generated in this case. PEX\_INTA# or VSx\_PEX\_INTA# assertion and INTx Message generation for Hot Plug or Link State cases are mutually exclusive, on a per-Port basis.

The NT Port Virtual and Link Interfaces can each independently support the interrupt mechanism (INTx or MSI) used in their respective domains. (Refer to [Section 14.5, “NT Port Interrupts,”](#) for details.)

## 9.1.1 Interrupt Sources or Events

The PEX 8748 internally generated interrupt/Message sources include:

- For Hot Plug-capable Ports
  - [Presence Detect Changed](#) (logical OR of PRSNT# (HP\_PRSNT\_# or Serial Hot Plug PRSNT# (SHP\_PRSNT\_#) input), and SerDes Receiver Detect<sup>a</sup> on Lane(s) associated with that Port)
  - [Attention Button Pressed](#)
  - [Power Fault Detected](#)
  - [MRL Sensor Changed](#)
  - [Command Completed](#)
  - [Data Link Layer State Changed](#)
  - [Link Bandwidth Management Status](#)
  - [Link Autonomous Bandwidth Status](#)
- For non-Hot Plug-capable Downstream Ports
  - [Presence Detect Changed](#) (SerDes Receiver Detect<sup>a</sup> on Lane(s) associated with that Port)
  - [Data Link Layer State Changed](#)
- Device-Specific NT-Link events (also refer to [Section 14.5, “NT Port Interrupts”](#))
  - NT-Link Port Correctable error (NT Port Link Interface, offset FC4h)
  - NT-Link Port Uncorrectable error (NT Port Link Interface, offset FB8h)
  - NT-Link Port Data Link Layer State change
  - NT-Link Port received (and consequently dropped) a Fatal or Non-Fatal Error Message
- General-Purpose Input/Output (GPIO) events
- Non-Transparent (NT) Doorbell events (also refer to [Section 14.5, “NT Port Interrupts”](#))
- Virtual Switch mode
  - Link Status management
  - Doorbells

The PEX 8748 externally generated interrupt/Message sources include INTx Messages from Downstream devices.

[Table 9-1](#) lists the interrupt sources.

*a. The SerDes Receiver Detect mechanism is comprised of the [Physical Layer Receiver Detect Status](#) register [Receiver Detected on Lane x](#) bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 200h[31:16]) or Serial Hot Plug PRSNT# (from external I<sup>2</sup>C I/O Expander) input for the Port.*

Table 9-1. Interrupt Sources

Event/Error	Description
<b>Base and Virtual Switch Modes</b>	
Hot Plug or Link State events	<p><b>Slot Status</b> register (offset 80h):</p> <ul style="list-style-type: none"> <li><i>Presence Detect Changed</i> (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, bit 19) is Set</li> <li><i>Data Link Layer State Changed</i> (Downstream Ports, bit 24 is Set)</li> </ul>
PCI Express Hot Plug events	<p>The master control of Hot Plug interrupts is the <b>Slot Control</b> register <i>Hot Plug Interrupt Enable</i> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[5]).</p> <p>There are six sources of Hot Plug interrupts. Each Hot Plug source has its own <i>Enable</i> bit in the <b>Slot Control</b> register:</p> <ul style="list-style-type: none"> <li>Downstream Ports <ul style="list-style-type: none"> <li><i>Attention Button Pressed</i> (bit 16)</li> <li><i>Power Fault Detected</i> (bit 17)</li> <li><i>Command Completed</i> (bit 20)</li> <li><i>Data Link Layer State Changed</i> (bit 24)</li> </ul> </li> <li>Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports <ul style="list-style-type: none"> <li><i>MRL Sensor Changed</i> (bit 18)</li> <li><i>Presence Detect Changed</i> (bit 19)</li> </ul> </li> </ul> <p>The interrupt status of each Hot Plug source is provided by the Port's <b>Slot Status</b> register (Downstream Ports, offset 80h).</p> <p><b>Note:</b> <i>Presence</i> (<b>Slot Status</b> register <i>Presence Detect State</i> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[22])) is determined by the logical OR of:</p> <ul style="list-style-type: none"> <li><i>SerDes Receiver Detect</i> (<b>Physical Layer Receiver Detect Status</b> register <i>Receiver Detected on Lane x</i> bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 200h[31:16])), and</li> <li><i>HP_PRSENT_x#</i>, –or–</li> <li><i>Serial Hot Plug PRSENT#</i> (<i>SHP_PRSENTx#</i>) (from external I<sup>2</sup>C I/O Expander) input for the Port</li> </ul>
General-Purpose Input Interrupt events	<p>External interrupt from any of the PORT_GOODx# signals that are configured as an Interrupt input in the <b>GPIO x_y Direction Control</b> register <i>Direction Control</i> bit(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset(s) 600h and 604h).</p>



Table 9-1. Interrupt Sources (Cont.)

Event/Error	Description
Device-Specific Error conditions	<ul style="list-style-type: none"> <li>Device-Specific (RAM ECC) errors indicated by the <b>Device-Specific Error Status <i>x</i></b> register bit(s), if not masked in their corresponding <b>Device-Specific Error Mask <i>x</i></b> register bit(s) (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offsets <b>700h</b>, <b>708h</b>, <b>710h</b>, and/or <b>718h</b> (Status) and offsets <b>704h</b>, <b>70Ch</b>, <b>714h</b>, and/or <b>71Ch</b> (Mask)).</li> </ul> <p><i>Note:</i> Device-Specific (RAM ECC) errors can be signaled by interrupt, and/or as an Uncorrectable Internal error that is fatal (<b>Uncorrectable Error Status</b> register <b>Uncorrectable Internal Error Status</b> and <b>Uncorrectable Error Severity</b> register <b>Uncorrectable Internal Error Severity</b> bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offsets <b>FB8h[22]</b> and <b>FC0h[22]</b>, respectively, are Set).</p> <ul style="list-style-type: none"> <li>NT Port Link Interface Correctable errors reported by the NT Port Virtual Interface (to the Host of the Upstream Port domain), collectively flagged in the <b>Link Error Status Virtual</b> register <b>Link Side Correctable Error Status</b> bit (NT Port Virtual Interface, offset <b>FE0h[0]</b>), with specific errors indicated in the <b>Correctable Error Status</b> register (NT Port Link Interface, offset <b>FC4h</b>), if not masked both globally in the <b>Link Error Mask Virtual</b> register <b>Link Side Correctable Error Mask</b> bit (NT Port Virtual Interface, offset <b>FE4h[0]</b>), nor individually in the <b>Correctable Error Mask</b> register (NT Port Link Interface, offset <b>FC8h</b>).</li> <li>NT Port Link Interface Uncorrectable errors reported by the NT Port Virtual Interface (to the Host of the Upstream Port domain), collectively flagged in the <b>Link Error Status Virtual</b> register <b>Link Side Uncorrectable Error Status</b> bit (NT Port Virtual Interface, offset <b>FE0h[1]</b>), with specific errors indicated in the <b>Uncorrectable Error Status</b> register (NT Port Link Interface, offset <b>FB8h</b>), if not masked both globally in the <b>Link Error Mask Virtual</b> register <b>Link Side Uncorrectable Error Mask</b> bit (NT Port Virtual Interface, offset <b>FE4h[1]</b>), nor individually in the <b>Uncorrectable Error Mask</b> register (NT Port Link Interface, offset <b>FBCh</b>).</li> </ul>

Table 9-1. Interrupt Sources (Cont.)

Event/Error	Description
<b>NT Mode Only</b>	
Device-Specific NT-Link Port errors and events	<ul style="list-style-type: none"> <li>NT Port Link Interface Correctable errors reported by the NT Port Virtual Interface (to the Host of the Upstream Port domain), collectively flagged in the <b>Link Error Status Virtual</b> register <i>Link Side Correctable Error Status</i> bit (NT Port Virtual Interface, offset FE0h[0]), with specific errors indicated in the <b>Correctable Error Status</b> register (NT Port Link Interface, offset FC4h), if not masked both globally in the <b>Link Error Mask Virtual</b> register <i>Link Side Correctable Error Mask</i> bit (NT Port Virtual Interface, offset FE4h[0]), nor individually in the <b>Correctable Error Mask</b> register (NT Port Link Interface, offset FC8h).</li> <li>NT Port Link Interface Uncorrectable errors reported by the NT Port Virtual Interface (to the Host of the Upstream Port domain), collectively flagged in the <b>Link Error Status Virtual</b> register <i>Link Side Uncorrectable Error Status</i> bit (NT Port Virtual Interface, offset FE0h[1]), with specific errors indicated in the <b>Uncorrectable Error Status</b> register (NT Port Link Interface, offset FB8h), if not masked both globally in the <b>Link Error Mask Virtual</b> register <i>Link Side Uncorrectable Error Mask</i> bit (NT Port Virtual Interface, offset FE4h[1]), nor individually in the <b>Uncorrectable Error Mask</b> register (NT Port Link Interface, offset FBCh).</li> <li>NT Port Link Interface State change – Interrupt to the NT Port Virtual Interface Host, if enabled (not masked) by the <b>Link Error Mask Virtual</b> register <i>Link Side DL Active Change Mask</i> bit (NT Port Virtual Interface, offset FE4h[2]).</li> <li>Link Side Uncorrectable Error Message Drop interrupt to the NT Port Virtual Interface Host, if enabled (not masked) by the <b>Link Error Mask Virtual</b> register <i>Link Side Uncorrectable Error Message Drop Mask</i> bit (NT Port Virtual Interface, offset FE4h[3]). This feature supports applications using back-to-back NT Ports, where an Uncorrectable Error Message received (and properly dropped) by the NT Port Link Interface can trigger an interrupt to the NT Port Virtual Interface Host.</li> </ul>
NT-Virtual Doorbell events	<b>NT Virtual Interface IRQ Set/Clear</b> register (offsets C4Ch[15:0] and/or C50h[15:0]) bit is Set while the corresponding <b>NT Virtual Interface IRQ Set/Clear</b> register (offsets C54h[15:0] and/or C58h[15:0]) bit is Cleared.
NT-Link Doorbell events	<b>NT Link Interface IRQ Set/Clear</b> register (offsets C5Ch[15:0] and/or C60h[15:0]) bit is Set while the corresponding <b>NT Link Interface IRQ Set/Clear</b> register (offsets C64h[15:0] and/or C68h[15:0]) bit is Cleared.

**Table 9-1. Interrupt Sources (Cont.)**

Event/Error	Description
<b>Virtual Switch Mode Only</b>	
Management Port Doorbell-Generated interrupts	<ul style="list-style-type: none"> <li>Writing 1 to the <b>VS Upstream to Management Upstream Doorbell Request</b> register (VS Upstream Port(s) and Management Port, offset 910h[3:0]) signals an interrupt (unless it is masked by the <b>VS Upstream to Management Upstream Doorbell Mask</b> register (VS Upstream Port(s) and Management Port, offset 914h[3:0])) to the Management Port</li> <li>Writing 1 to the <b>Management Upstream to VS Upstream Doorbell Request</b> register (VS Upstream Port(s) and Management Port, offset 928h[3:0]) signals an interrupt (unless it is masked by the <b>Management Upstream to VS Upstream Doorbell Mask</b> register (VS Upstream Port(s) and Management Port, offset 92Ch[3:0])) to the Management Port</li> </ul> <p><i><b>Note:</b> If a virtual switch Upstream Port is also the Management Port, that Upstream Port <b>cannot</b> send Doorbell interrupts to itself.</i></p>
Management Link Status events	<ul style="list-style-type: none"> <li>A Link transition from the <i>DL_Inactive</i> state to the <i>DL_Active</i> state Sets the bit that corresponds to the Port Number (<b>Switch Link Up</b> register (Port 0, accessible through the Management Port, offset 900h[19:16, 11:8, 3:0]), and generates an interrupt (unless it is masked by the <b>Switch Link Event Mask</b> register (Port 0, accessible through the Management Port, offset 908h[19:16, 11:8, 3:0])) to the Management Port</li> <li>A Link transition from the <i>DL_Active</i> state to the <i>DL_Inactive</i> state Sets the bit that corresponds to the Port Number (<b>Switch Link Down</b> register (Port 0, accessible through the Management Port, offset 904h[19:16, 11:8, 3:0]), and generates an interrupt (unless it is masked by the <b>Switch Link Event Mask</b> register (Port 0, accessible through the Management Port, offset 908h[19:16, 11:8, 3:0])) to the Management Port</li> </ul>

## 9.1.2 Interrupt Handling

The PEX 8748 provides an Interrupt Generation module with each Port. The module reads the Request for interrupts from different sources, then generates an MSI or PCI-compatible Assert\_INTx/Deassert\_INTx Interrupt Message. MSIs support a PCI Express edge-triggered interrupt, whereas Assert\_INTx and Deassert\_INTx Message transactions emulate PCI level-triggered interrupt signaling. The System Interrupt Controller functions include:

- Sensing Interrupt events
- Signaling the interrupt, by way of the INTx mechanism, and Setting the *Interrupt Status* bit
- Signaling the interrupt, by way of the MSI mechanism
- Handling INTx-type Interrupt Messages from Downstream devices

### 9.1.2.1 Interrupt Handling – Base Mode

Base mode supports INTx or MSIs generated by the PEX 8748, as per the *PCI Express Base r3.0*.

One INTA# Interrupt output, [PEX\\_INTA#](#), is implemented.

### 9.1.2.2 Interrupt Handling – Virtual Switch Mode

Virtual Switch mode supports INTx or MSIs generated by the PEX 8748, as per the *PCI Express Base r3.0*, within the respective hierarchy.

One INTA# Interrupt output, VSx\_PEX\_INTA#, is implemented, per virtual switch.

## 9.2 INTx Emulation Support

The PEX 8748 supports PCI INTx emulation, to signal interrupts to the System Interrupt Controller. This mechanism is compatible with existing PCI software. PCI INTx emulation virtualizes PCI physical Interrupt signals, by using the in-band signaling mechanism.

PCI **Interrupt** registers (defined in the *PCI r3.0*) are supported. The Port's *PCI r3.0* **PCI Command** register *Interrupt Disable* and **PCI Status** register *Interrupt Status* bits are also supported (offset 04h[10 and 19], respectively).

Although the *PCI Express Base r3.0* provides INTA#, INTB#, INTC#, and INTD# for INTx signaling, the PEX 8748 uses only INTA# for internal Interrupt Message generation, because it is a single-function device. However, incoming Messages from Downstream devices can be of INTA#, INTB#, INTC#, or INTD# type. Internally generated INTA# Messages from the Downstream Port are also re-mapped and collapsed at the Upstream Port(s), according to the Downstream Port's Device Number, with its own Device Number and Received Device Number from the Downstream device.

When an interrupt is requested, the Port's **PCI Status** register *Interrupt Status* bit is Set. If INTx interrupts are enabled (Port's **PCI Command** register *Interrupt Disable* and **MSI Control** register *MSI Enable* bits, offsets 04h[10] and 48h[16], are both Cleared, respectively), an Assert\_INTx Message is generated and transmitted Upstream to indicate the Port interrupt status. For each interrupt event, there is a corresponding *Interrupt Mask* bit; an Interrupt Message can be generated only when the corresponding *Interrupt Mask* bit is Cleared. Software reads and Clears the event and *Interrupt Status* bit after servicing the interrupt.

A Port de-asserts INTx, or PEX\_INTA# (Base mode) or VSx\_PEX\_INTA# (Virtual Switch mode), interrupts, in response to one or more of the following conditions:

- Port's **PCI Command** register *Interrupt Disable* bit (offset 04h[10]) is Set
- Corresponding *Interrupt Mask* bit is Set
- Upstream Port Link goes down (DL\_Down condition), or receives a Hot Reset (unless Hot Reset/DL\_Down Reset is disabled, by Setting the **Virtual Switch Debug** register *Upstream Port and NT-Link Port DL\_Down Reset Propagation Disable* bit (Upstream Port(s), offset A30h[4]))
- Software Clears the corresponding *Interrupt Status* bit

## 9.2.1 INTx-Type Interrupt Message Re-Mapping and Collapsing

The Upstream Port(s) re-map(s) and collapse(s) the INTx *virtual wires* received at a Downstream Port, based upon the Downstream Port's Device Number and Received INTx Message Requester ID Device Number, and generate(s) a new Interrupt Message, according to the mapping defined in [Table 9-2](#).

Each virtual PCI-to-PCI bridge of a Downstream Port specifies the Port Number associated with the INTx (Interrupt) Messages received or generated, and forwards the Interrupt Messages Upstream.

A Downstream Port transmits an Assert\_INTA/Deassert\_INTA Message to the Upstream Port(s), due to a Hot Plug and/or PCI Express Hot Plug, Link State, GPIO, NT Port Doorbell (NT mode only), Management Port Doorbell interrupt and/or Management Link Status (Virtual Switch mode only), and/or Device-Specific NT Port Link Interface (reported by the NT Port Virtual Interface) error/event.

Internally generated INTx Messages always originate as type INTA Messages, because the PEX 8748 is a single-function device. Internally generated Interrupt INTA Messages from Downstream Ports are re-mapped at the Upstream Port(s) to INTA, INTB, INTC, or INTD Messages, according to the mapping defined in [Table 9-2](#).

INTx Messages from Downstream devices and from internally generated Interrupt Messages are ORed together to generate INTA, INTB, INTC, or INTD level-sensitive signals, and edge-detection circuitry in the Upstream Port(s) generates the Assert\_INTx and Deassert\_INTx Messages. The Upstream Port(s) then forward(s) the new Messages Upstream, by way of its (their) Link.

**Table 9-2. Downstream/Upstream Port INTx Interrupt Message Mapping**

Device Number <sup>a</sup>	At Downstream Port	By Upstream Port(s)
0, 8, 16	INTA	INTA
	INTB	INTB
	INTC	INTC
	INTD	INTD
1, 9, 17	INTA	INTB
	INTB	INTC
	INTC	INTD
	INTD	INTA
2, 10, 18	INTA	INTC
	INTB	INTD
	INTC	INTA
	INTD	INTB
3, 11, 19	INTA	INTD
	INTB	INTA
	INTC	INTB
	INTD	INTC

a. [Table 4-1](#) indicates which Ports are enabled/used, and when, based upon the STRAP\_STNx\_PORTCFGx input states and/or serial EEPROM programming.

### 9.2.1.1 Interrupt Re-Mapping and Collapsing – NT Mode

In NT mode, an NT Port Virtual Interface-generated interrupt is treated like an external event to the PCI-to-PCI bridge immediately Upstream, for tracking purposes. In this mode, when the Upstream Port(s) receive(s) an INTx Message from the NT Port Virtual Interface, the Upstream Port(s)' re-mapping-collapsing logic performs double Swizzling, one based upon the NT Port Virtual Interface's Captured Device Number, and another based upon the virtual Downstream Port (PCI-to-PCI) Device Number.

If software asserts a [Secondary Bus Reset](#) to this PCI-to-PCI bridge, the PCI-to-PCI bridge de-asserts the NT Port Virtual Interface interrupt.

## 9.3 MSI Support

One of the interrupt schemes supported by the PEX 8748 is the MSI mechanism, which is required for PCI Express devices. The MSI method uses Memory Write transactions to deliver interrupts. MSIs are edge-triggered interrupts.

**Note:** *MSIs and INTx are mutually exclusive, on a per-Port basis. The mechanisms that generate these types of interrupts **cannot** be simultaneously enabled.*

### 9.3.1 MSI Operation

**Note:** *The descriptions in this section are applicable only to Transparent Ports. For the NT Port, refer to:*

- [Section 15.7, “NT Port Virtual Interface MSI Capability Registers \(Offsets 48h – 64h\)”](#)
- [Section 16.7, “NT Port Link Interface MSI Capability Registers \(Offsets 48h – 64h\)”](#)

At configuration time, system software traverses the function Capability list. If a **Capability ID** of 05h is found, the function implements MSIs. System software reads the **MSI Capability** structure registers, to determine function capabilities.

The Port’s **MSI Control** register *Multiple Message Capable* field (offset 48h[19:17]) default value is 011b, which indicates that the PEX 8748 can request up to eight MSI Vectors (Address and Data). When the Port’s register’s *Multiple Message Enable* field (offset 48h[22:20]) is Cleared (default), only one Vector is allocated, and therefore, the PEX 8748 can generate only one Vector for all errors or events. When system software writes a non-zero value to the *Multiple Message Enable* field, multiple-Vector support is enabled (the quantity of Vectors supported is dependent upon the value). [Table 9-3](#) lists the supported MSI Vector types.

System software initializes the Port’s MSI Address registers (offsets 4Ch and 50h) and **MSI Data** register (offset 54h), with a system-specified Vector. After system software enables the MSI function (by Setting the Port’s **MSI Control** register *MSI Enable* bit, offset 48h[16]), when an Interrupt event occurs, the Interrupt Generation module generates a DWord Memory Write to the address specified by the Port’s **MSI Address** (lower 32 bits of the *Message Address* field) and **MSI Upper Address** (upper 32 bits of the *Message Address* field) register contents (offsets 4Ch and 50h, respectively). The single DWord Payload includes zero (0) for the upper two bytes, and the lower two bytes are taken from the **MSI Data** register. The Port’s **MSI Control** register *Multiple Message Enable* field (offset 48h[22:20]) can be programmed to a value of 000b, 001b, 010b, or 011b. When programmed to 011b, the lower three bits of Message data are changed to indicate the general type of interrupt event that occurred. (Refer to [Table 9-3](#).)

**Table 9-3. Supported MSI Vector Types**

Vector Type	Modes	
	Base	Virtual Switch
Power Management, or Hot Plug or Link State events	✓	✓
Device-Specific NT-Port Link or RAM ECC error events	✓	✓
GPIO interrupts	✓	✓
Management Link Status events		✓
Management Port Doorbell interrupts		✓



The quantity of MSI Vectors generated is dependent upon the quantity enabled:

- If **one** MSI Vector is enabled (default), all six interrupt events are combined into a single Vector
- If **two** MSI Vectors are allocated, the individual Vectors indicate:
  - Vector[0] Hot Plug or Link State-/PM-triggered interrupt event, GPIO-generated event, Virtual Switch Doorbell event, and Management Port Link Status event
  - Vector[1] Device-specific NT-Link Port event or RAM ECC error event
- If **four** MSI Vectors are allocated, the individual Vectors indicate:
  - Vector[0] Hot Plug or Link State-/PM-triggered interrupt event
  - Vector[1] Device-specific NT-Link Port event or RAM ECC error event
  - Vector[2] GPIO-generated event
  - Vector[3] Virtual Switch Doorbell event, and Management Port Link Status event
- If **eight** MSI Vectors are allocated (up to six Vectors are used; the upper two Vectors (having the highest values of Message Data bits [2:0]) are **not** used), the individual Vectors indicate:
  - Vector[0] Hot Plug or Link State-/PM-triggered interrupt event
  - Vector[1] Device-specific NT-Link Port event or RAM ECC error event
  - Vector[2] GPIO-generated event
  - Vector[3] **Reserved**
  - Vector[4] Virtual Switch Doorbell event
  - Vector[5] Management Port Link Status event

If a non-masked Interrupt event occurs before system software Sets the **MSI Control** register *MSI Enable* bit, normally (but unlike Conventional PCI interrupts, which are level-triggered), an MSI packet is sent immediately after software Sets the *MSI Enable* bit, to notify the system of the prior event. Alternatively, MSIs for prior events can be disabled, on a per-Port basis, by Setting the **ECC Error Check Disable** register *Disable Sending MSI if MSI Is Enabled after Interrupt Status Set* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 720h[10]).

When the error or event that caused the interrupt is serviced, the PEX 8748 can generate a new MSI Memory Write as a result of new events. Because MSIs are edge-triggered events, the **MSI Mask** register *Interrupt Mask* bits (offset 58h[5:0]) are provided, to use for masking the events. A new MSI can be generated only after the *Interrupt Status* bits are serviced. System software should mask these bits when an MSI event is being processed.

The **MSI Control** register *MSI 64-Bit Address Capable* bit is enabled (offset 48h[23] is Set), by default. If the serial EEPROM and/or I<sup>2</sup>C/SMBus Clears the bit, the **MSI Capability** structure is reduced by 1 DWord (*that is*, register offsets 54h, 58h, and 5Ch change to 50h, 54h, and 58h, respectively).

### 9.3.1.1 Message Signaled Interrupts – NT Mode

If NT mode is enabled, NT Port Virtual Interface MSI TLPs are not generated if the **PCI Command** register *Bus Master Enable* bit (offset 04h[2]) is Cleared in the Upstream Port(s), NT Port Virtual Interface, and Virtual Downstream PCI-to-PCI bridge.

## 9.3.2 MSI Capability Registers

For details, refer to Section 13.9, “Message Signaled Interrupt Capability Registers (Offsets 48h – 64h).”

## 9.4 PEX\_INTA# or VSx\_PEX\_INTA# Interrupts

PEX\_INTA# (Base mode) or VSx\_PEX\_INTA# (Virtual Switch mode) Interrupt output is enabled when the following conditions are met:

- INTx Messages are enabled (Port's **PCI Command** register *Interrupt Disable* bit, offset 04h[10], is Cleared) and MSIs are disabled (Port's **MSI Control** register *MSI Enable* bit, offset 48h[16], is Cleared)
- PEX\_INTA# or VSx\_PEX\_INTA# output is enabled for the following errors and events, when the **ECC Error Check Disable** register bit associated with that error or event is Set:
  - *Enable PEX\_INTA# or VSx\_PEX\_INTA# Interrupt Output for GPIO-Generated Interrupts* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 720h[6])
  - *Enable PEX\_INTA# or VSx\_PEX\_INTA# Interrupt Output for Hot Plug or Link State Event-Triggered Interrupts* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 720h[4])
  - **NT mode only**
    - *Enable PEX\_INTA# or VSx\_PEX\_INTA# Interrupt Output for NT-Virtual Doorbell-Generated Interrupts* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 720h[7])
    - *Enable PEX\_INTA# or VSx\_PEX\_INTA# Interrupt Output for Device-Specific NT-Link Port Event-Triggered Interrupts* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 720h[5])
  - **Virtual Switch mode only**
    - *Enable VSx\_PEX\_INTA# Interrupt Output for Management Link Status Event-Generated Interrupts* bit (Virtual Switch mode – Port 0, accessible through the Management Port, offset 720h[9])
    - *Enable VSx\_PEX\_INTA# Interrupt Output for Management Port Doorbell-Generated Interrupts* bit (Virtual Switch mode – Port 0, accessible through the Management Port, offset 720h[8])

The three interrupt mechanisms, listed below, are mutually exclusive modes of operation, on a per-Port basis, for all interrupt sources:

- Conventional PCI INTx Message generation
- Native MSI transaction generation
- Device-Specific PEX\_INTA# or VSx\_PEX\_INTA# assertion

PEX\_INTA# or VSx\_PEX\_INTA# assertion (Low) indicates that the PEX 8748 detected one or more of the events and/or errors (if not masked) listed in [Table 9-1](#).

**Note:** *PEX\_INTA# or VSx\_PEX\_INTA# assertion and INTx messaging are mutually exclusive for a given interrupt event. When MSIs are enabled (Port's *MSI Enable* bit is Set), both INTx and PEX\_INTA# or VSx\_PEX\_INTA# are disabled for PEX 8748 internally generated interrupts. The forwarding of external INTx Messages received from a Downstream Port to an Upstream Port is always enabled.*

## 9.5 General-Purpose Input/Output

The PEX 8748 contains 12 multiplexed General-Purpose Input/Output (GPIO) balls. They can be used as GPIO, interrupt, or PORT\_GOOD indicators, or Serial Hot Plug **PERST#** pins. The balls/pins can default to the PORT\_GOOD output function or Serial Hot Plug **PERST#** function, either through register programming and/or the **STRAP\_TESTMODE[2:0]** input states.

In Base mode, the Virtual Switch **GPIO** registers (offsets **64Ch** through **664h** and **A34h** through **A54h**) are *not* used. In Virtual Switch mode, the Management Port and/or I<sup>2</sup>C has access to all **GPIO** registers (Chip-specific (offsets **600h** through **63Ch**), Management Port (offsets **64Ch** through **664h**), and Virtual Switch (offsets **A34h** through **A54h**) registers). Each virtual switch can access only its own **Virtual Switch** registers.

The 12 GPIO signals are divided among the virtual switches, according to the enabled Ports that are assigned to each virtual switch. The default Settings that determine which signals are assigned to each virtual switch are based upon the Virtual Switch Table registers (Virtual Switch mode – Port 0, accessible through the Management Port, offsets **354h** through **394h**). (Refer to [Section 5.3.3](#), “Virtual Switch Table.”)

The **Virtual Switch GPIO\_PG 0\_11 Availability** register *Number of GPIO\_PGs Available* field (VS Upstream Port(s), offset **A3Ch[3:0]**) can be read by the Virtual Switch Host, to determine how many GPIO\_PG signals are available to that virtual switch. The Management Port can then adjust the quantity of GPIO\_PG signals assigned to each virtual switch. A maximum of 12 GPIO\_PG signals can be assigned to any virtual switch. A single GPIO\_PG signal cannot be assigned to more than one virtual switch.

If the **STRAP\_TESTMODE[2:0]** 3-state inputs call for the Serial Hot Plug **PERST#** function by default, two GPIO\_SHP pins are assigned to each virtual switch, by default. It is up to the Management Port to make any further assignments of the GPIO\_SHP pins to the virtual switches.

The logic that controls the GPIO ball function is driven from the **GPIO** Chip-specific registers (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offsets **600h** through **63Ch**). When a GPIO signal is assigned to a virtual switch, the GPIO information from the **VS GPIO** registers is multiplexed into the corresponding Chip-specific **GPIO** registers. As a result, the **VS GPIO** registers now controls the data in the Chip-specific **GPIO** registers, for the GPIOs assigned to it, which leaves the Chip-specific **GPIO** registers to act as Read-Only registers for the GPIOs that are assigned to virtual switches.

Table 9-4 lists the registers used for GPIO functionality.

**Table 9-4. Registers Used for GPIO Functionality**

Offset	Register
<b>Chip-Specific Registers</b> (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)	
600h	GPIO 0_9 Direction Control
604h	GPIO 10_11 Direction Control
614h	GPIO 0_11 Input De-Bounce
61Ch	GPIO 0_11 Input Data
624h	GPIO 0_11 Output Data
62Ch	GPIO 0_11 Interrupt Polarity
634h	GPIO 0_11 Interrupt Status
63Ch	GPIO 0_11 Interrupt Mask
<b>Virtual Switch Management Port Registers</b> (Virtual Switch mode – Port 0, accessible through the Management Port)	
64Ch	Virtual Switch GPIO Update
650h	VS0 GPIO_PG 0_11 Assignment
654h	VS1 GPIO_PG 0_11 Assignment
658h	VS2 GPIO_PG 0_11 Assignment
65Ch	VS3 GPIO_PG 0_11 Assignment
660h	VS4 GPIO_PG 0_11 Assignment
664h	VS5 GPIO_PG 0_11 Assignment
<b>Virtual Switch Registers</b> (Virtual Switch mode – VS Upstream Port(s))	
A34h	Virtual Switch GPIO_PG 0_9 Direction Control
A38h	Virtual Switch GPIO_PG 10_11 Direction Control
A3Ch	Virtual Switch GPIO_PG 0_11 Availability
A40h	Virtual Switch GPIO_PG 0_11 Input De-Bounce
A44h	Virtual Switch GPIO_PG 0_11 Input Data
A48h	Virtual Switch GPIO_PG 0_11 Output Data
A4Ch	Virtual Switch GPIO_PG 0_11 Interrupt Polarity
A50h	Virtual Switch GPIO_PG 0_11 Interrupt Status
A54h	Virtual Switch GPIO_PG 0_11 Interrupt Mask

## 9.6 Management Port Interrupts – Virtual Switch Mode

The CPU connected to the Management Port receives interrupts for the following two events:

- Switch Ports Link Status
- Doorbell

Both are described in the sections that follow.

### 9.6.1 Switch Port Link Status Events – Virtual Switch Mode

The Management Port has four registers that provide Link status to the Management CPU (located in Port 0, accessible through the Management Port):

- **Switch Link Up** register (offset 900h)
- **Switch Link Down** register (offset 904h)
- **Switch Link Event Mask** register (offset 908h)
- **Switch Link Status** register (offset 90Ch)

In all four registers, each non-*Reserved* bit corresponds to one Port (for example, bit 0 corresponds to Port 0, bit 1 corresponds to Port 1, and so forth).

When a Port Link goes to an active state (DL\_ACTIVE=1) from a down state, it Sets the Port's **Switch Link Up** register *Port x Link Up* bit (offset 900h[19:16, 11:8, 3:0]), regardless of the Port's **Switch Link Event Mask** register *Port x Link Event Mask* bit (offset 908h[19:16, 11:8, 3:0]) value. If the interrupt is not masked, the Management Port interrupt handler signals an Assert\_INTA Message to the Management CPU if Conventional PCI interrupts (INTx) are enabled and MSIs are disabled. If instead MSIs are enabled, the Management Port generates an MSI Message instead of a Conventional PCI Assert\_INTA Message. When the Interrupt Service Routine. (ISR) services this interrupt, it writes 1 to the Port's **Switch Link Up** register *Port x Link Up* bit, to Clear this event. This Clear event or Interrupt Mask event generates a Deassert\_INTA Message to the Management Port, if all Interrupt events in the Port's **Switch Link Up** and **Switch Link Down** registers are Cleared, provided Conventional PCI interrupts are enabled and MSIs are disabled.

When a Port Link goes to a down state (DL\_ACTIVE=0) from an active state, it Sets the Port's **Switch Link Down** register *Port x Link Down* bit (offset 904h[19:16, 11:8, 3:0]), regardless of the Port's **Switch Link Event Mask** register *Port x Link Event Mask* bit (offset 908h[19:16, 11:8, 3:0]) value. If the interrupt is not masked, the Management Port interrupt handler signals an Assert\_INTA Message to the Management CPU, if Conventional PCI interrupts (INTx) are enabled and MSIs are disabled. If instead MSIs are enabled, the Management Port generates an MSI Message instead of a Conventional PCI Assert\_INTA Message. When the ISR services this interrupt, it writes 1 to the Port's **Switch Link Down** register *Port x Link Down* bit, to Clear this event. This Clear event or Interrupt Mask event generates a Deassert\_INTA Message to the Management Port, if all Interrupt events in the Port's **Switch Link Up** and **Switch Link Down** registers are Cleared, provided Conventional PCI interrupts are enabled and MSIs are disabled.

### 9.6.1.1 Special Handling for Race Conditions – Virtual Switch Mode

If multiple DL\_ACTIVE and DL\_INACTIVE events occur before the ISR is able to service the Interrupt event, a race condition exists with event ordering. The Management Port implements the **Switch Link Status** register (Port 0, accessible through the Management Port, offset 90Ch) for this purpose. If the Port's bits are Set in both the **Switch Link Up** and **Switch Link Down** registers (Port 0, accessible through the Management Port, offsets 900h and 904h, respectively), the ISR looks at the **Switch Link Status** register, to determine the order of these two events:

- If a **Switch Link Status** register bit is Cleared, the first event is DL\_ACTIVE and the latest event is DL\_INACTIVE
- If a bit in the **Switch Link Status** register is Set, the first event is DL\_INACTIVE and the latest event is DL\_ACTIVE

### 9.6.2 Doorbell Interrupts – Virtual Switch Mode

In Virtual Switch mode, each virtual switch Upstream Port (other than the active Management Port) implements **Doorbell** and **Scratchpad** registers for communication (located in the VS Upstream Port(s) and Management Port):

- Virtual switch to Management CPU direction
  - **VS Upstream to Management Upstream Doorbell Request** register (offset 910h)
  - **VS Upstream to Management Upstream Doorbell Mask** register (offset 914h)
  - **VS Upstream to Management Upstream Scratchpad 1** register (offset 918h)
  - **VS Upstream to Management Upstream Scratchpad 2** register (offset 91Ch)
  - **VS Upstream to Management Upstream Scratchpad 3** register (offset 920h)
  - **VS Upstream to Management Upstream Scratchpad 4** register (offset 924h)
- Management CPU to virtual switch direction
  - **Management Upstream to VS Upstream Doorbell Request** register (offset 928h)
  - **Management Upstream to VS Upstream Doorbell Mask** register (offset 92Ch)
  - **Management Upstream to VS Upstream Scratchpad 1** register (offset 930h)
  - **Management Upstream to VS Upstream Scratchpad 2** register (offset 934h)
  - **Management Upstream to VS Upstream Scratchpad 3** register (offset 938h)
  - **Management Upstream to VS Upstream Scratchpad 4** register (offset 93Ch)

Software uses these registers to establish communication between the active Management CPU and the other CPUs. There is no in-band communication mechanism between CPUs outside the active Management Port domain.

A Fundamental Reset resets all the registers listed above.

A Hot Reset to the Management Port domain Clears the **VS Upstream to Management Upstream Doorbell Request**, **VS Upstream to Management Upstream Doorbell Mask**, and **VS Upstream to Management Upstream Scratchpad x** registers in the Virtual Switch to Management CPU direction.

A Hot Reset to virtual switches other than the Management Port domain Clears the **Management Upstream to VS Upstream Doorbell Request**, **Management Upstream to VS Upstream Doorbell Mask**, and **Management Upstream to VS Upstream Scratchpad x** registers in the Management CPU to Virtual Switch direction.

Software writing to either **Doorbell Request** register generates an interrupt if the corresponding **Doorbell Mask** register bit is not masked, and interrupt signaling is enabled.



## Chapter 10 Hot Plug Support

### 10.1 Introduction

*Note: In this chapter, unless stated otherwise, “Hot Plug Controller” references both the Parallel Hot Plug Controllers and Serial Hot Plug Controller.*

Hot Plug capability allows board insertion and removal from a running system, without adversely affecting the system. Boards are typically inserted or removed to repair faulty boards or re-configure the system without system down time. Hot Plug capability allows systems to isolate faulty boards in the event of a failure.

The PEX 8748 includes one Parallel Hot Plug Controller per Hot Plug-capable Transparent Downstream Port and one Serial Hot Plug Controller per PEX 8748, as well as signals for both Parallel and Serial Hot Plug support. The quantity of Parallel Hot Plug Ports depends upon the [STRAP\\_TESTMODE\[2:0\]](#) and [VS\\_MODE](#) input states, and can be up to three (A, B, and C). All Transparent Downstream Ports can support Serial Hot Plug.

Parallel Hot Plug can be implemented on any Transparent Downstream Port, as selected by the [Parallel Hot Plug Control](#) register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset [3A4h](#)[7:0, 15:8, and 23:16] for Parallel Hot Plug Controllers A, B, and C, respectively).

Serial Hot Plug can be implemented on any Transparent Downstream Port. If a Transparent Downstream Port is both Parallel- and Serial Hot Plug-capable, the Serial Hot Plug Controller is used, by default, unless the Port’s [Power Management Hot Plug User Configuration](#) register [Serial Hot Plug Override Parallel Disable](#) bit (offset [F70h](#)[19]) is Set.

Hot Plug signals are enabled, configured, and accessed through the Port’s [Slot Capability](#) and [Slot Status and Control](#) registers (Downstream Ports, offsets [7Ch](#) and [80h](#), respectively). Also, each Port’s [Power Management Hot Plug User Configuration](#) register provides additional Device-Specific configuration and control, for both Parallel and Serial Hot Plug implementations.

Normally, Hot Plug cannot be implemented on PEX 8748 Upstream Port(s). However, as an option, Presence Detect and its corresponding interrupt can be enabled for an Upstream Port, by Setting the Port’s [Power Management Hot Plug User Configuration](#) register [Upstream Hot Plug Enable](#) bit (Upstream Port(s), offset [F70h](#)[14]).



## 10.2 Hot Plug Features

The following are the PEX 8748 Hot Plug features:

- Hot Plug features are supported on all Transparent Downstream Ports.
- Zero to three Parallel Hot Plug Ports, depends upon the [STRAP\\_TESTMODE\[2:0\]](#) and [VS\\_MODE](#) input states.
- Any three Transparent Downstream Ports can be programmed as a Parallel Hot Plug Port.
- Additional Hot Plug signals for all other Transparent Downstream Ports (that are not being used as a Parallel Hot Plug Port, up to 11) are implemented with external I<sup>2</sup>C I/O Expanders, which alert the PEX 8748 through the [SHPC\\_INT#](#) input, that inputs have toggled, and the PEX 8748 internal Hot Plug Controllers and registers automatically communicate with and control the I/O Expanders, using the PEX 8748 I<sup>2</sup>C Master interface ([I2C\\_SCL1](#) and [I2C\\_SDA1](#) balls).
- Insertion and removal of PCI Express boards, without removing system power.
- Board Present and Manually operated Retention Latch (MRL) signals are implemented. Presence Detect is accomplished through either an in-band SerDes Receiver Detect mechanism ([Physical Layer Receiver Detect Status](#) register *Receiver Detected on Lane x* bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset [200h\[31:16\]](#))) –or– by using the [HP\\_PRSENT\\_x#](#) inputs.
- Power Indicator and Attention Indicator Output signals are controlled.
- Attention Button is monitored.
- Power Fault detection and Faulty board isolation.
- Power switch for controlling Downstream device power. In Parallel Hot Plug Ports, polarity of this output is defined by the [SPARE1](#) input sampled at [PEX\\_PERST#](#) input de-assertion. In Serial Hot Plug Ports, this output is Active-High.
- Generates Power Management Event (PME) for Hot Plug events in sleeping systems (D3hot Device Power Management (PM) state).
- Electromechanical Interlock Control feature available on Serial Hot Plug-capable Ports.
- Hot Plug interrupts can be sent in-band using [INTx](#) or MSI Messages, or signaled externally using [PEX\\_INTA#](#) (Base mode) or [VSx\\_PEX\\_INTA#](#) (Virtual Switch mode).



## 10.3 Hot Plug Elements

Table 10-1 summarizes the Hot Plug elements required for PCI Express Hot Plug implementation. For specific platform requirements, refer to the PCI Express Form Factor specifications.

**Table 10-1. Required Hot Plug Elements for PCI Express Implementation**

Element	Purpose
Attention Button	To request Hot Plug operations. Implemented on the PEX 8748's Hot Plug-capable Transparent Downstream Ports.
Attention Indicator	Implemented on the PEX 8748's Hot Plug-capable Transparent Downstream Ports. LED functions: <ul style="list-style-type: none"> <li>• <b>Off</b> – Standard operation.</li> <li>• <b>On</b> – Operational Problem at this slot.</li> <li>• <b>Blinking</b> – Slot is being identified at user's request. Blinking frequency is 1 Hz. 50% duty cycle.</li> </ul>
Power Indicator	Implemented on the PEX 8748's Hot Plug-capable Transparent Downstream Ports. LED functions: <ul style="list-style-type: none"> <li>• <b>Off</b> – Slot is powered Off. Board insertion or removal is permitted.</li> <li>• <b>On</b> – Board insertion or removal is not permitted.</li> <li>• <b>Blinking</b> – Slot is in the process of powering Up or Down. Blinking frequency is 2 Hz. 50% duty cycle.</li> </ul>
MRL	Manually-operated Retention Latch, that holds add-in boards in place.
MRL Sensor	Reports the position of a slot's MRL to the Port. A logic Low indicates that the latch is closed.
Electromechanical Interlock	Prevents removal of adapter from slot.

## 10.4 Hot Plug Signals

### 10.4.1 Hot Plug Port External Signals

The on-chip signals for Parallel Hot Plug Controller support are defined in [Section 3.4.2.1, “Parallel Hot Plug Signals.”](#)

The on-chip signals for Serial Hot Plug Controller support are defined in [Section 3.4.2.2, “Serial Hot Plug Signals.”](#) In addition to the set of on-chip Serial Hot Plug signals, the PEX 8748 supports Serial Hot Plug signals to and from the I<sup>2</sup>C I/O Expander, which are used with Serial Hot Plug-capable Transparent Downstream Ports. (Refer to [Section 10.11.2.](#))

### 10.4.2 Hot Plug Output States for Disabled Hot Plug Slots

When a Hot Plug slot is disabled, the Hot Plug outputs for that Port are in the logic states defined in [Table 10-2.](#)

**Table 10-2. Hot Plug Outputs for Disabled Hot Plug Slot**

Output Signal	Logic	Comments
HP_ATNLED_x#	High	Attention LED is turned Off
HP_CLKEN_x#	High	Reference Clock is not driven to the slot
HP_PERST_x#	Low	Slot remains in reset
HP_PWREN_x	Low	Power Controller is turned Off
HP_PWRLED_x#	High	Power LED is turned Off

## 10.5 Hot Plug Registers

All Transparent Downstream Ports and Stations include identical sets of Hot Plug registers, and all Hot Plug Ports use the identical register sets, regardless of whether Hot Plug is implemented using the PEX 8748 Hot Plug signals, or Serial Hot Plug signals on the external I<sup>2</sup>C I/O Expanders. Therefore, other than initial configuration (typically programmed by serial EEPROM), whether Hot Plug functionality for a Port is implemented using a Parallel Hot Plug Controller or Serial Hot Plug Controller (with external I<sup>2</sup>C I/O Expander) is effectively transparent to software.

The PCI Express Hot Plug Configuration, Capability, Command, Status, and Event registers are described in [Section 13.10, “PCI Express Capability Registers \(Offsets 68h – A0h\).”](#)

Device-Specific Hot Plug configuration features are programmable in the **Power Management Hot Plug User Configuration** register (offset F70h) of each Station and Transparent Downstream Port.

## 10.6 Hot Plug Interrupts

Refer to [Chapter 9, “Interrupts,”](#) for interrupt details.

### 10.6.1 Software Testing of Hot Plug Interrupts

Hot Plug interrupts can be generated by software (such as for testing Interrupt Handler software), without having to toggle Hot Plug signals to trigger interrupts. The Downstream Port **Slot Status** register bits at offset 80h[24:16] are usually Read Only (RO; as required by the *PCI Express Base r3.0*); however, if the Port’s **Power Management Hot Plug User Configuration** register *Software-Controlled Hot Plug Enable* bit (offset F70h[12]) is Set, the Hot Plug input signals (BUTTON#, PWRFLT#, MRL#, PRSNT#, and INTERLOCK#) are disconnected from the pins, and software is given control of the corresponding register bits.

When this feature is enabled, the following **Slot Status** register (Downstream Ports, offset 80h) *Status* bits for the Port can be Set by software writing 1 to these bit positions, but only if the *Status* bit value is already 0:

- For BUTTON#, *Attention Button Pressed* bit (bit 16)
- For PWRFLT#, *Power Fault Detected* bit (bit 17)
- For INTERLOCK#, *Electromechanical Interlock Status* bit (bit 23)
- *Command Completed* bit (bit 20)

If the bit is already Set, writing 1 Clears the *Status* bit (same as in standard operation).

Software *cannot* write 1 to directly change the Port’s **Slot Status** register *MRL Sensor Changed* and *Presence Detect Changed* bits. However, software can write to the following **Slot Status** register bits, if the software Write to the corresponding *Status* bits changed the original value:

- Writes to the *MRL Sensor State* bit (bit 21) change the register’s *MRL Sensor Changed* bit (bit 18)
- Writes to the *Presence Detect State* bit (bit 22) change the register’s *Presence Detect Changed* bit (bit 19)

During software testing, the Port’s **Power Management Hot Plug User Configuration** register *Software-Controlled Power Good* bit (offset F70h[13]) replaces the HP\_PWR\_GOOD\_x inputs.

If software Sets the Port’s **Slot Status** register *Power Fault Detected* bit (Downstream Ports, offset 80h[17]) when the bit is Cleared, PERST# output is asserted.

If software changes the MRL# state to 1 (open), that Port executes a Power Down sequence on its slot.

## 10.7 Hot Plug and Virtual Switch Mode Failover

In Virtual Switch mode, a Downstream Port belonging to a failed virtual switch can be moved to another virtual switch that is up and running. A Transparent Downstream Port that supports Parallel Hot Plug continues to support Parallel Hot Plug when that Port is re-assigned to another virtual switch. Likewise, a Downstream Port that does *not* support Parallel Hot Plug remains a non-Hot Plug Port when that Port is re-assigned to another virtual switch. The **Parallel Hot Plug Control** register (Port 0, accessible through the Management Port, offset 3A4h[7:0, 15:8, and 23:16] for Parallel Hot Plug Controllers A, B, and C, respectively) value is *not* changed by the PEX 8748 when the Port is re-assigned to a different virtual switch.

### 10.7.1 Virtual Switch Mode Failover and Upstream Port Hot Plug Functionality

To inform software of the addition of a bridge in failover while the PEX 8748 is in Virtual Switch mode, the Upstream Port Hot Plug (Device-Specific) functionality can be enabled, by Setting the Port's **Power Management Hot Plug User Configuration** register *Upstream Hot Plug Enable* bit (Upstream Port(s), offset F70h[14]). This feature enables the MRL Sensor and Presence Detect-related register bits, and generates the corresponding Hot Plug interrupts, using the Upstream Port(s)' **Slot Status and Control** register (offset 80h, which is usually *Reserved* in Upstream Ports).

The Hot Plug-related bits that can be enabled for the Upstream Port in the **Slot Status and Control** register are:

- *MRL Sensor Changed Enable* (bit 2)
- *Presence Detect Changed Enable* (bit 3)
- *Hot Plug Interrupt Enable* (bit 5)
- *MRL Sensor Changed* (bit 18)
- *Presence Detect Changed* (bit 19)
- *MRL Sensor State* (bit 21)
- *Presence Detect State* (bit 22)

If the *Upstream Hot Plug Enable* bit is Set, software can indirectly program the *MRL Sensor State* and *Presence Detect State* bit values, by writing to the Port's **Power Management Hot Plug User Configuration** register *Software MRL State Value* and *Software Present Detect State Value* bits (Upstream Port(s), offset F70h[28:27], respectively). The *MRL Sensor Changed* and *Presence Detect Changed* bits are updated accordingly, and Hot Plug interrupts can be generated (if enabled), to indicate the addition of a bridge behind the PEX 8748 Upstream Port.

## 10.8 Hot Plug Controller Slot Power-Up/Down Sequence

If a Hot Plug-capable Transparent Downstream Port is enabled, the Port's Hot Plug Controller can power-up or power-down the slot. This section describes how this process occurs.

### 10.8.1 Slot Power-Up Sequence

If a Hot Plug-capable Transparent Downstream Port is connected to a slot, its associated Hot Plug Controller can power up that slot, with or without an external serial EEPROM. Hot Plug Controller sequencing is determined by the states of the following bits:

- Port's **Slot Capability** register *Power Controller Present* bit (Downstream Ports, offset 7Ch[1])
- Port's **Slot Capability** register *MRL Sensor Present* bit (Downstream Ports, offset 7Ch[2])
- Port's **Slot Control** register *Power Controller Control* bit (Downstream Ports, offset 80h[10])

and the HP\_MRL\_x# input states, if the *MRL Sensor Present* bit is Set. Hot Plug-configurable features are programmable only by the serial EEPROM and/or I<sup>2</sup>C.

### 10.8.1.1 Configuring Slot Power-Up Sequence Features with Serial EEPROM

An external serial EEPROM can be used to configure the Hot Plug Controller and Hot Plug outputs. Features can be changed by using the registers defined in [Table 10-3](#). The Hot Plug Controller outputs remain in the default state described in [Table 10-2](#), before the serial EEPROM image is loaded into the device.

After the serial EEPROM image is loaded, the Hot Plug Controller starts a power-up sequence on each slot that has the Port's **Slot Capability** register *Power Controller Present* bit (Downstream Ports, offset 7Ch[1]) Set and **Slot Control** register *Power Controller Control* bit (Downstream Ports, offset 80h[10]) Cleared.

**Table 10-3. Configuring Slot Power-Up Sequence Features with Serial EEPROM**

Register Bit(s) <sup>a</sup>	Hot Plug Controller and Hot Plug Output Signal Configurable Features
<i>Power Controller Present</i>  <b>Slot Capability</b> register, offset 7Ch[1]	<b>Reserved</b> for the Upstream Port(s) and NT Port. The <i>Power Controller Present</i> bit enables or disables the Hot Plug Controller on the Hot Plug-capable Transparent Downstream Ports. If this bit is Cleared, the Hot Plug Controller is disabled for that slot and a power-up sequence is not executed. The slot remains in the disabled state, as defined in <a href="#">Table 10-2</a> . If this bit is Set, the Hot Plug Controller powers up the slot when the MRL is closed and the Port's <b>Slot Control</b> register <i>Power Controller Control</i> bit (Downstream Ports, offset 80h[10]) is Cleared. Otherwise, if the Port's <b>Slot Capability</b> register <i>MRL Sensor Present</i> bit (Downstream Ports, offset 7Ch[2]) is Cleared, the MRL's position has no effect on powering up the slot.
<i>MRL Sensor Present</i>  <b>Slot Capability</b> register, offset 7Ch[2]	<b>Reserved</b> for the Upstream Port(s) and NT Port. If this bit is Cleared, the MRL position is "Don't Care," and has no effect on powering up the slot. If this bit is Set, the PEX 8748 senses whether the MRL is open or closed for a slot. The MRL should be Low for power-on for that slot.
<i>Attention Indicator Present</i>  <b>Slot Capability</b> register, offset 7Ch[3]	<b>Reserved</b> for the Upstream Port(s) and NT Port. When Set, this bit controls whether the HP_ATNLED_x# output for the slot drives out Active-Low. Otherwise, this output is not functional on the slot.
<i>Power Indicator Present</i>  <b>Slot Capability</b> register, offset 7Ch[4]	<b>Reserved</b> for the Upstream Port(s) and NT Port. When Set, this bit controls whether the HP_PWRLED_x# output for the slot drives out Active-Low. Otherwise, this output is not functional on the slot.
<i>HPC Tpepv</i>  <b>Power Management Hot Plug User Configuration</b> register, offset F70h[4:3]	Functionality associated with this field is enabled only on Downstream Ports. This field indicates the delay from when an HP_PWREN_x output is asserted High, to when power is valid at a slot. (Refer to <a href="#">Section 10.8.1.2</a> .) 00b = Feature is disabled, and HP_PWR_GOOD_x inputs are used for Power Valid 01b = 128 ms 10b = 256 ms 11b = 512 ms
<i>HP_PWR_GOOD_x Active-Low Enable</i>  <b>Power Management Hot Plug User Configuration</b> register, offset F70h[6]	Functionality associated with this bit is enabled only on Downstream Ports. Controls the HP_PWR_GOOD_x input polarity. (Refer to <a href="#">Section 10.8.1.2</a> .) 0 = HP_PWR_GOOD_x inputs are Active-High 1 = HP_PWR_GOOD_x inputs are Active-Low

a. All register bits listed in this table are located in the Downstream Ports.

### 10.8.1.2 Slot Power-Up Sequencing when *Power Controller Present Bit Is Set*

By default, when the MRL is open, the Port's **Slot Capability** register *MRL Sensor Present* and *Power Controller Present* bits (Downstream Ports, offset 7Ch[2:1], respectively) and **Slot Control** register *Power Controller Control* bit (Downstream Ports, offset 80h[10]) are Set on a Hot Plug-capable Downstream Port. When the serial EEPROM is not present, present but blank, or programmed with default register values, the Hot Plug Controller is initially powered up, the Port's **PCI Express Capability** register *Slot Implemented* bit (Downstream Ports, offset 68h[24]) is Set, and the PEX 8748 is in the following state:

1. Hot Plug Controller is enabled for Hot Plug-capable Transparent Downstream Port.
2. Slots associated with Hot Plug-capable Transparent Downstream Ports are enabled to be powered up.
3. Attention LED (HP\_ATNLED\_x#) and Power LED (HP\_PWRLED\_x#) are High on the slot chassis.

Immediately after the PEX 8748 exits Reset (PEX\_PERST# and/or VSx\_PERST# input goes High), if the Hot Plug-capable Transparent Downstream Port's *MRL Sensor Present* bit is Set (default), the HP\_MRL\_x# input for that slot is sampled. If HP\_MRL\_x# input is enabled and asserted (value of 0), the device Clears the *Power Controller Control* bit, to enable slot power-up. If the *Power Controller Control* bit is not Cleared, either by initially enabling it (default) and asserting HP\_MRL\_x#, or by programming both the *MRL Sensor Present* and *Power Controller Control* bit values to 0 in the serial EEPROM, the Downstream slot is not powered up and remains in the disabled state, as defined in Table 10-2.

If a slot's *Power Controller Present* bit is Set, and the *Power Controller Control* bit is Cleared (either by initially enabling and asserting HP\_MRL\_x#, or by programming the *MRL Sensor Present* and *Power Controller Control* bits to 0 in the serial EEPROM), the slot starts power-up sequencing with HP\_PWREN\_x and HP\_PWRLED\_x# assertion, following PEX\_PERST# and/or VSx\_PERST# input de-assertion and serial EEPROM initialization. The serial EEPROM initialization delay is determined by the following:

- Serial EEPROM clock (EE\_SK) frequency, programmable through the **Serial EEPROM Clock Frequency** register *EepFreq[2:0]* field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 268h[2:0])
- Quantity of registers that are programmed to be initialized by the serial EEPROM

*The power-up sequence is as follows:*

1. The Hot Plug Controllers drive the HP\_PWRLED\_*x*# outputs Low, to turn On the Power Indicators, and drives the HP\_PWREN\_*x* outputs High, to turn On the external Power Controllers.
2. After the HP\_PWR\_GOOD\_*x* inputs are sampled asserted High or  $T_{pepv}$  delay following HP\_PWREN\_*x* output assertion, power to the slots are valid and the Hot Plug Controllers drive the HP\_CLKEN\_*x*# outputs Low, to turn On the Reference Clock (PEX\_REFCLKp/n) to the slots.

The  $T_{pepv}$  time delay is specified by programming the Port's **Power Management Hot Plug User Configuration** register *HPC Tpepv* field (Downstream Ports, offset F70h[4:3]) to a non-zero value. Values of 01b, 10b, or 11b program the delay to 128, 256, or 512 ms, respectively. The default value, 00b, disables the feature, and uses the HP\_PWR\_GOOD\_*x* inputs instead.

3. After the 100-ms  $T_{pvperl}$  time delay following HP\_CLKEN\_*x*# output assertion, the Hot Plug Controllers de-assert the HP\_PERST\_*x*# outputs, to release slot reset.

Consideration should be given to the combination of the serial EEPROM clock (EE\_SK) frequency (programmable through the **Serial EEPROM Clock Frequency** register *EepFreq[2:0]* field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 268h[2:0])), along with the quantity of registers to be initialized by serial EEPROM, as well as any delay for cascaded resets through multiple devices, and allow sufficient margin for devices to be ready for Host enumeration.



Figure 10-1 illustrates the timing sequence with the Port's **Slot Capability** register *Power Controller Present* bit (Downstream Ports, offset 7Ch[1]) Set. This timing sequence occurs at system power-up, or when a slot is being powered up by the user, using software control.

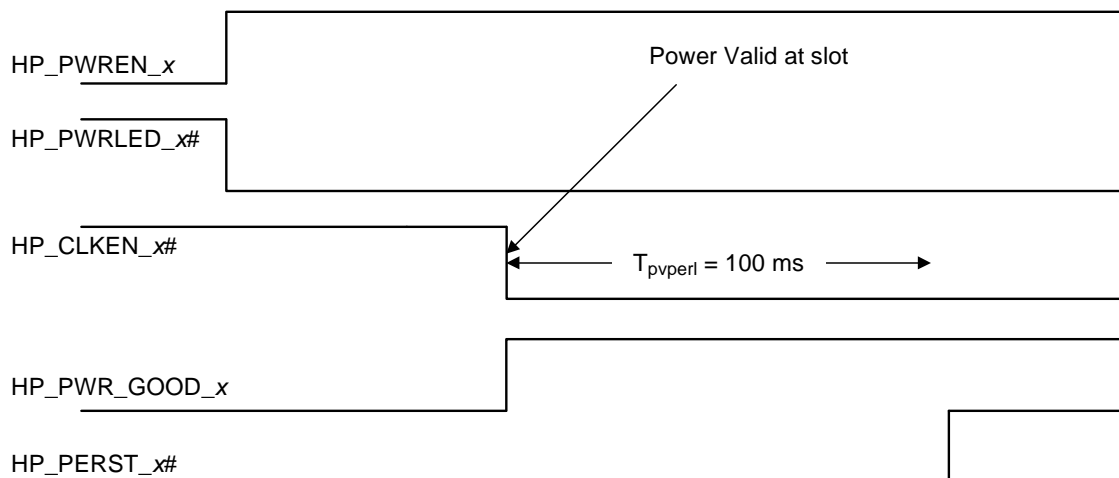
If the HP\_MRL\_x# inputs are enabled, but not asserted to power-up the slots immediately after reset, the HP\_MRL\_x# inputs can be asserted at runtime to start the slot power-up sequence, provided that the Port's **Slot Capability** register *MRL Sensor Present* and *Power Controller Present* bits (Downstream Ports, offset 7Ch[2:1], respectively) are both Set (either by default values when the serial EEPROM is not present or blank, or by programming the serial EEPROM to Set these bits), and the Port's **Slot Control** register *Power Controller Control* bit (Downstream Ports, offset 80h[10]) is Cleared (by the programmed serial EEPROM and/or software).

Power-up sequencing at runtime is controlled by software Clearing the *Power Controller Control* bit, in response to an interrupt caused by one or both of the following:

- HP\_MRL\_x# input assertion (if an MRL Sensor is present, and the Port's **Slot Control** register *Hot Plug Interrupt Enable* and *MRL Sensor Changed Enable* bits (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[5 and 2], respectively) are both Set)
- User pressing the Attention Button, if enabled (Port's register's *Attention Button Pressed Enable* (Downstream Ports, offset 80h[0]) and *Hot Plug Interrupt Enable* bits must both be Set)

HP\_MRL\_x# and HP\_BUTTON\_x# input assertion and de-assertion at runtime are not latched until the 10-ms de-bounce ensures that their state changes are stable.

**Figure 10-1. Slot Power-Up Timing when *Power Controller Present* Bit Is Set**



**Note:** HP\_PWRLED\_x# is **not** asserted if the serial EEPROM and/or I<sup>2</sup>C Slave interface Clears the Port's **Slot Capability** register *Power Indicator Present* bit (Downstream Ports, offset 7Ch[4]).

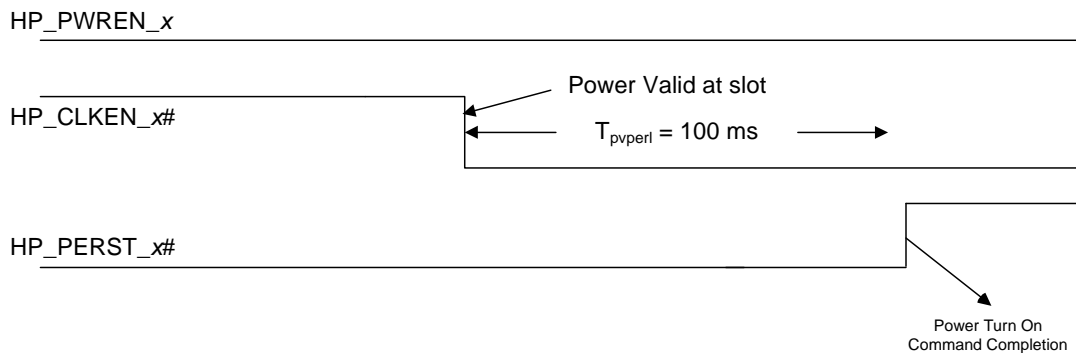
### 10.8.1.3 HP\_PERST\_x# (Reset) and HP\_PWRLED\_x# Output Power-Up Sequencing when *Power Controller Present* Bit Is Cleared

The HP\_PERST\_x# and HP\_PWRLED\_x# outputs can be used without enabling the Hot Plug Power Controllers (HP\_PWREN\_x and HP\_CLKEN\_x# outputs and HP\_PWRFLT\_x# inputs). *For example*, the HP\_PERST\_x# outputs can be used to reset an on-board Downstream device.

If the Port's **Slot Capability** register *Power Controller Present* bit (Downstream Ports, offset 7Ch[1]) and **Slot Control** register *Power Controller Control* bit (Downstream Ports, offset 80h[10]) are both Cleared by the serial EEPROM, the HP\_PERST\_x# outputs are de-asserted (High) and the HP\_PWRLED\_x# outputs are asserted (Low), after the Root Complex PERST# input is de-asserted, as illustrated in Figure 10-2. However, the HP\_PWRLED\_x# outputs are *not* asserted, if the serial EEPROM also Cleared the Port's **Slot Capability** register *Power Indicator Present* bit (Downstream Ports, offset 7Ch[4]).

If the serial EEPROM is initially blank, causing register default values to be loaded, the HP\_PERST\_x# outputs are asserted and the HP\_PWRLED\_x# outputs are *not* asserted, unless the HP\_MRL\_x# inputs are Low. Therefore, if the HP\_PERST\_x# and/or HP\_PWRLED\_x# outputs are used (and an MRL is *not* used), pull the HP\_MRL\_x# inputs Low, to allow the outputs to toggle, regardless of whether the serial EEPROM is blank.

**Figure 10-2. Hot Plug Outputs when *Power Controller Present* and *Power Controller Control* Bits Are Cleared**



**Note:** HP\_PWRLED\_x# are *not* asserted if the serial EEPROM and/or I<sup>2</sup>C Slave interface Clears the Port's Power Indicator Present bit.

### 10.8.1.4 Disabling Power-Up Hot Plug Output Sequencing

If the serial EEPROM is used to Set the Port's **Slot Control** register *Power Controller Control* bit (Downstream Ports, offset 80h[10]) after reset, the HP\_PWRLED\_x# and HP\_CLKEN\_x# outputs remain High, and the HP\_PERST\_x# and HP\_PWREN\_x outputs remain Low. The HP\_PWRLED\_x# and HP\_CLKEN\_x# outputs also remain High, if the HP\_MRL\_x# inputs are not asserted in the default Hot Plug power-up sequencing described in [Section 10.8.1.2](#).

## 10.8.2 Slot Power-Down Sequence

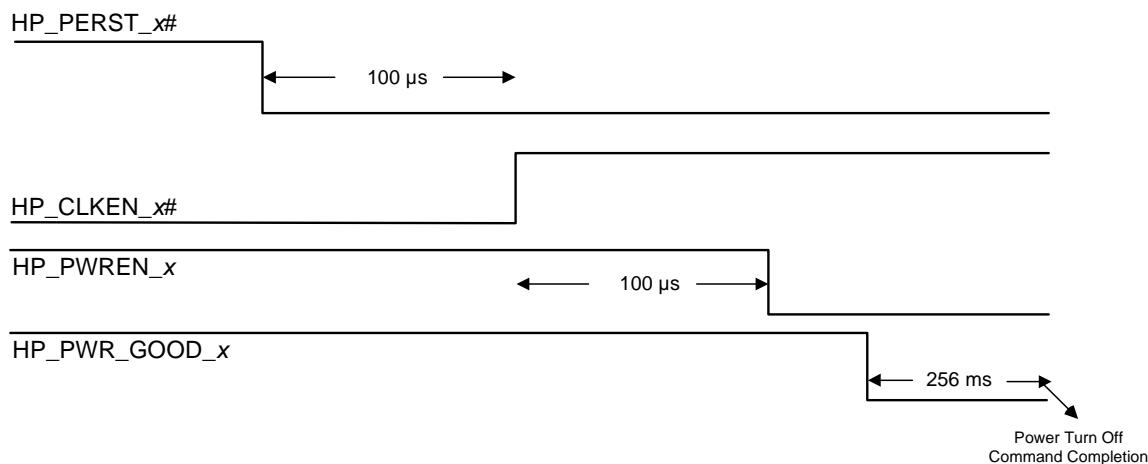
Slots can be powered down by either:

- De-asserting the HP\_MRL\_x# input, by manually opening the closed MRL, –or–
- Software Setting the Port's **Slot Control** register *Power Controller Control* bit (Downstream Ports, offset 80h[10])

[Figure 10-3](#) illustrates the following power-down timing sequence for either event:

1. HP\_PERST\_x# output to the Port is asserted.
2. HP\_CLKEN\_x# output is de-asserted to the slot, 100  $\mu$ s after HP\_PERST\_x# output is asserted.
3. HP\_PWREN\_x output is de-asserted to the slot, 100  $\mu$ s after HP\_CLKEN\_x# output is de-asserted.

**Figure 10-3. Hot Plug Automatic Slot Power-Down Sequence**



## 10.9 Default Parallel Hot Plug Ports

Default assignment of Hot Plug Controllers to Transparent Downstream Ports depends upon the test mode and Port configuration. By default, the test mode is defined by the [STRAP\\_TESTMODE\[2:0\]](#) 3-state inputs, and Port configuration is defined by the [STRAP\\_STNx\\_PORTCFGx](#) 3-state inputs, all of which are sampled and latched immediately following Fundamental Reset (at [PEX\\_PERST#](#) de-assertion); however, the following agents can subsequently override the Strapping inputs:

- Serial EEPROM
- I<sup>2</sup>C/SMBus, if [STRAP\\_I2C\\_SMBUS\\_CFG\\_EN#](#)=0, to delay linkup until the **Configuration Release** register *Initiate Configuration* bit (Port 0, offset [3ACh\[0\]](#)) is Set
- Management Port software (in Virtual Switch mode), if [STRAP\\_MGMT\\_PORT\\_EN](#)=Z, to delay linkup until the *Initiate Configuration* bit is Set

These three agents can also override the Strapping inputs in assigning Hot Plug Controllers to Transparent Downstream Ports, by programming the **Parallel Hot Plug Control** register (Port 0, offset [3A4h\[7:0, 15:8, and 23:16\]](#) for Parallel Hot Plug Controllers A, B, and C, respectively). However, if the Hot Plug Port assignment must be changed (from the default assignment determined by the Strapping inputs, as indicated in the **Parallel Hot Plug Control** register), use of the serial EEPROM is recommended, rather than I<sup>2</sup>C/SMBus or Management Port software, to configure the Hot Plug Slot signals as quickly as possible. By default, Port 1 of each Station (Ports 1, 9 and 17, if they are Transparent Downstream Ports), has highest priority for default assignment as a Parallel Hot Plug Port.

Hot Plug Controller availability is as follows:

- Hot Plug Controller A is available in Test modes [0 through 3].
- Hot Plug Controller B is available in Test modes [3, 7, and 11].
- Hot Plug Controller C is available in Test modes [0 through 7].
- Hot Plug Controllers are not available in Test Modes [8, 9, and 10].

For Test modes [0 through 6], Hot Plug Controllers A, B, and C are assigned in the following order:

1. Hot Plug Controller C,
2. Hot Plug Controller A, if the Test mode is [0, 1, 2, or 3],
3. Hot Plug Controller B, if the Test mode is [3].

For Test modes [7 and 11], the Hot Plug Controller assignment order is:

1. Hot Plug Controller B,
2. Hot Plug Controller C, if the Test mode is [7].

Transparent Downstream Ports have priority for default assignment of Hot Plug Controllers, in the following order – 1, 9, 17, 0, 8, 16, 2, 10, 18, 3, 11, and 19.

**Note:** [Table 4-1](#) indicates which Ports are enabled/used, and when, based upon the [STRAP\\_STNx\\_PORTCFGx](#) input states and/or serial EEPROM programming.

[Table 10-4](#), [Table 10-5](#), and [Table 10-6](#) lists the [STRAP\\_TESTMODE\[2:0\]](#) configurations, which includes which Hot Plug Controllers are enabled for each of the 12 Test mode configurations. For further details, refer to [Section 3.4.4.1, “STRAP\\_TESTMODE\[2:0\] 3-State Input Configurations.”](#)

**Table 10-4. STRAP\_TESTMODE[2:0] 3-State Input Configurations – Base Mode, Test Modes 0, 1, 2, 3**

Location	Type	Multiplexed I/O Signals Used, by Test Mode – Base Mode			
		0	1	2	3
		STRAP_TESTMODE[2:0] = 000	STRAP_TESTMODE[2:0] = 00Z	STRAP_TESTMODE[2:0] = 001	STRAP_TESTMODE[2:0] = 0Z0
		Hot Plug A + Hot Plug C + PORT_GOOD[10:0]#	Hot Plug A + Hot Plug C + SHP_PERST[10:0]#	Hot Plug A + Hot Plug C + GPIO[10:0] Input	Hot Plug A + Hot Plug C + Hot Plug B + PORT_GOOD0#
AC26	O, PU	HP_ATNLED_A#	HP_ATNLED_A#	HP_ATNLED_A#	HP_ATNLED_A#
Y24	I, PU	HP_BUTTON_A#	HP_BUTTON_A#	HP_BUTTON_A#	HP_BUTTON_A#
AA24	O, PU	HP_CLKEN_A#	HP_CLKEN_A#	HP_CLKEN_A#	HP_CLKEN_A#
AB26	I, PU	HP_MRL_A#	HP_MRL_A#	HP_MRL_A#	HP_MRL_A#
AB25	O, PU	HP_PERST_A#	HP_PERST_A#	HP_PERST_A#	HP_PERST_A#
AD26	I, PU	HP_PRSNT_A#	HP_PRSNT_A#	HP_PRSNT_A#	HP_PRSNT_A#
W21	I, PD	HP_PWR_GOOD_A	HP_PWR_GOOD_A	HP_PWR_GOOD_A	HP_PWR_GOOD_A
W22	O, PD	HP_PWREN_A	HP_PWREN_A	HP_PWREN_A	HP_PWREN_A
AE26	I, PU	HP_PWRFLT_A#	HP_PWRFLT_A#	HP_PWRFLT_A#	HP_PWRFLT_A#
Y23	O, PU	HP_PWRLED_A#	HP_PWRLED_A#	HP_PWRLED_A#	HP_PWRLED_A#
B25	O, PU	HP_ATNLED_C#	HP_ATNLED_C#	HP_ATNLED_C#	HP_ATNLED_C#
C25	I, PU	HP_BUTTON_C#	HP_BUTTON_C#	HP_BUTTON_C#	HP_BUTTON_C#
F24	O, PU	HP_CLKEN_C#	HP_CLKEN_C#	HP_CLKEN_C#	HP_CLKEN_C#
G21	I, PU	HP_MRL_C#	HP_MRL_C#	HP_MRL_C#	HP_MRL_C#
G22	O, PU	HP_PERST_C#	HP_PERST_C#	HP_PERST_C#	HP_PERST_C#
D25	I, PU	HP_PRSNT_C#	HP_PRSNT_C#	HP_PRSNT_C#	HP_PRSNT_C#
F25	I, PD	HP_PWR_GOOD_C	HP_PWR_GOOD_C	HP_PWR_GOOD_C	HP_PWR_GOOD_C
D24	O, PD	HP_PWREN_C	HP_PWREN_C	HP_PWREN_C	HP_PWREN_C
B26	I, PU	HP_PWRFLT_C#	HP_PWRFLT_C#	HP_PWRFLT_C#	HP_PWRFLT_C#
D26	O, PU	HP_PWRLED_C#	HP_PWRLED_C#	HP_PWRLED_C#	HP_PWRLED_C#
AB1	I/O, PU	PORT_GOOD1#	SHP_PERST1#	GPIO1 input	HP_ATNLED_B#
AB3	I/O, PU	PORT_GOOD2#	SHP_PERST2#	GPIO2 input	HP_BUTTON_B#
AE1	I/O, PU	PORT_GOOD3#	SHP_PERST3#	GPIO3 input	HP_CLKEN_B#
AB2	I/O, PU	PORT_GOOD4#	SHP_PERST4#	GPIO4 input	HP_MRL_B#
AD1	I/O, PU	PORT_GOOD5#	SHP_PERST5#	GPIO5 input	HP_PERST_B#
W5	I/O, PU	PORT_GOOD6#	SHP_PERST6#	GPIO6 input	HP_PRSNT_B#
Y3	I/O, PD	PORT_GOOD7#	SHP_PERST7#	GPIO7 input	HP_PWR_GOOD_B
AC2	I/O, PD	PORT_GOOD8#	SHP_PERST8#	GPIO8 input	HP_PWREN_B
AC3	I/O, PU	PORT_GOOD9#	SHP_PERST9#	GPIO9 input	HP_PWRFLT_B#
AD2	I/O, PU	PORT_GOOD10#	SHP_PERST10#	GPIO10 input	HP_PWRLED_B#
D3	I/O, PU	PORT_GOOD0#	SHP_PERST0#	GPIO0 input	PORT_GOOD0#

**Table 10-5. STRAP\_TESTMODE[2:0] 3-State Input Configurations – Virtual Switch Mode, Test Modes 4, 5, 6, 7**

Location	Type	Multiplexed I/O Signals Used, by Test Mode – Virtual Switch Mode; One to Three Virtual Switches, as Selected by VS_MODE Input State			
		4	5	6	7
		STRAP_TESTMODE[2:0] = 0ZZ	STRAP_TESTMODE[2:0] = 0Z1	STRAP_TESTMODE[2:0] = 010	STRAP_TESTMODE[2:0] = 01Z
		VS[2:0] + Hot Plug C + PORT_GOOD[10:0]#	VS[2:0] + Hot Plug C + SHP_PERST[10:0]#	VS[2:0] + Hot Plug C + GPIO[10:0] Input	VS[2:0] + Hot Plug C + Hot Plug B + PORT_GOOD0#
AC26	O, PU	VS0_PEX_INTA#	VS0_PEX_INTA#	VS0_PEX_INTA#	VS0_PEX_INTA#
Y24	I, PU	VS0_PERST#	VS0_PERST#	VS0_PERST#	VS0_PERST#
AA24	O, PU	VS1_PEX_INTA#	VS1_PEX_INTA#	VS1_PEX_INTA#	VS1_PEX_INTA#
AB26	I, PU	VS1_PERST#	VS1_PERST#	VS1_PERST#	VS1_PERST#
AB25	O, PU	VS2_PEX_INTA#	VS2_PEX_INTA#	VS2_PEX_INTA#	VS2_PEX_INTA#
AD26	I, PU	VS2_PERST#	VS2_PERST#	VS2_PERST#	VS2_PERST#
W21	I, 3-state	VS_MODE	VS_MODE	VS_MODE	VS_MODE
W22	O, PD	VS2_FATAL_ERR#	VS2_FATAL_ERR#	VS2_FATAL_ERR#	VS2_FATAL_ERR#
AE26	I, PU	VS1_FATAL_ERR#	VS1_FATAL_ERR#	VS1_FATAL_ERR#	VS1_FATAL_ERR#
Y23	O, PU	VS0_FATAL_ERR#	VS0_FATAL_ERR#	VS0_FATAL_ERR#	VS0_FATAL_ERR#
B25	O, PU	HP_ATNLED_C#	HP_ATNLED_C#	HP_ATNLED_C#	HP_ATNLED_C#
C25	I, PU	HP_BUTTON_C#	HP_BUTTON_C#	HP_BUTTON_C#	HP_BUTTON_C#
F24	O, PU	HP_CLKEN_C#	HP_CLKEN_C#	HP_CLKEN_C#	HP_CLKEN_C#
G21	I, PU	HP_MRL_C#	HP_MRL_C#	HP_MRL_C#	HP_MRL_C#
G22	O, PU	HP_PERST_C#	HP_PERST_C#	HP_PERST_C#	HP_PERST_C#
D25	I, PU	HP_PRSNT_C#	HP_PRSNT_C#	HP_PRSNT_C#	HP_PRSNT_C#
F25	I/O, PD	HP_PWR_GOOD_C	HP_PWR_GOOD_C	HP_PWR_GOOD_C	HP_PWR_GOOD_C
D24	O, PD	HP_PWREN_C	HP_PWREN_C	HP_PWREN_C	HP_PWREN_C
B26	I, PU	HP_PWRFLT_C#	HP_PWRFLT_C#	HP_PWRFLT_C#	HP_PWRFLT_C#
D26	O, PU	HP_PWRLED_C#	HP_PWRLED_C#	HP_PWRLED_C#	HP_PWRLED_C#
AB1	I/O, PU	PORT_GOOD1#	SHP_PERST1#	GPIO1 input	HP_ATNLED_B#
AB3	I/O, PU	PORT_GOOD2#	SHP_PERST2#	GPIO2 input	HP_BUTTON_B#
AE1	I/O, PU	PORT_GOOD3#	SHP_PERST3#	GPIO3 input	HP_CLKEN_B#
AB2	I/O, PU	PORT_GOOD4#	SHP_PERST4#	GPIO4 input	HP_MRL_B#
AD1	I/O, PU	PORT_GOOD5#	SHP_PERST5#	GPIO5 input	HP_PERST_B#
W5	I/O, PU	PORT_GOOD6#	SHP_PERST6#	GPIO6 input	HP_PRSNT_B#
Y3	I/O, PD	PORT_GOOD7#	SHP_PERST7#	GPIO7 input	HP_PWR_GOOD_B
AC2	I/O, PD	PORT_GOOD8#	SHP_PERST8#	GPIO8 input	HP_PWREN_B
AC3	I/O, PU	PORT_GOOD9#	SHP_PERST9#	GPIO9 input	HP_PWRFLT_B#
AD2	I/O, PU	PORT_GOOD10#	SHP_PERST10#	GPIO10 input	HP_PWRLED_B#
D3	I/O, PU	PORT_GOOD0#	SHP_PERST0#	GPIO0 input	PORT_GOOD0#

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

**Table 10-6. STRAP\_TESTMODE[2:0] 3-State Input Configurations – Virtual Switch Mode, Test Modes 8, 9, 10, 11**

Location	Type	Multiplexed I/O Signals Used, by Test Mode – Virtual Switch Mode; Four to Six Virtual Switches, as Selected by VS_MODE Input State			
		8	9	10	11
		STRAP_TESTMODE[2:0] = 011	STRAP_TESTMODE[2:0] = Z00	STRAP_TESTMODE[2:0] = Z0Z	STRAP_TESTMODE[2:0] = Z01
		VS[2:0] + VS[5:3] + PORT_GOOD[11:0]#	VS[2:0] + VS[5:3] + SHP_PERST[10:0]#	VS[2:0] + VS[5:3] + GPIO[11:0] Input	VS[2:0] + VS[5:3] + Hot Plug B + PORT_GOOD[11, 0]#
AC26	O, PU	VS0_PEX_INTA#	VS0_PEX_INTA#	VS0_PEX_INTA#	VS0_PEX_INTA#
Y24	I, PU	VS0_PERST#	VS0_PERST#	VS0_PERST#	VS0_PERST#
AA24	O, PU	VS1_PEX_INTA#	VS1_PEX_INTA#	VS1_PEX_INTA#	VS1_PEX_INTA#
AB26	I, PU	VS1_PERST#	VS1_PERST#	VS1_PERST#	VS1_PERST#
AB25	O, PU	VS2_PEX_INTA#	VS2_PEX_INTA#	VS2_PEX_INTA#	VS2_PEX_INTA#
AD26	I, PU	VS2_PERST#	VS2_PERST#	VS2_PERST#	VS2_PERST#
W21	I, 3-state	VS_MODE	VS_MODE	VS_MODE	VS_MODE
W22	O, PD	VS2_FATAL_ERR#	VS2_FATAL_ERR#	VS2_FATAL_ERR#	VS2_FATAL_ERR#
AE26	I, PU	VS1_FATAL_ERR#	VS1_FATAL_ERR#	VS1_FATAL_ERR#	VS1_FATAL_ERR#
Y23	O, PU	VS0_FATAL_ERR#	VS0_FATAL_ERR#	VS0_FATAL_ERR#	VS0_FATAL_ERR#
B25	O, PU	VS3_PEX_INTA#	VS3_PEX_INTA#	VS3_PEX_INTA#	VS3_PEX_INTA#
C25	I, PU	VS3_PERST#	VS3_PERST#	VS3_PERST#	VS3_PERST#
F24	O, PU	VS4_PEX_INTA#	VS4_PEX_INTA#	VS4_PEX_INTA#	VS4_PEX_INTA#
G21	I, PU	VS4_PERST#	VS4_PERST#	VS4_PERST#	VS4_PERST#
G22	O, PU	VS5_PEX_INTA#	VS5_PEX_INTA#	VS5_PEX_INTA#	VS5_PEX_INTA#
D25	I, PU	VS5_PERST#	VS5_PERST#	VS5_PERST#	VS5_PERST#
F25	I/O, PD	PORT_GOOD11#	PERST11#	GPIO11 input	PORT_GOOD11#
D24	O, PD	VS5_FATAL_ERR#	VS5_FATAL_ERR#	VS5_FATAL_ERR#	VS5_FATAL_ERR#
B26	I, PU	VS4_FATAL_ERR#	VS4_FATAL_ERR#	VS4_FATAL_ERR#	VS4_FATAL_ERR#
D26	O, PU	VS3_FATAL_ERR#	VS3_FATAL_ERR#	VS3_FATAL_ERR#	VS3_FATAL_ERR#
AB1	I/O, PU	PORT_GOOD1#	SHP_PERST1#	GPIO1 input	HP_ATNLED_B#
AB3	I/O, PU	PORT_GOOD2#	SHP_PERST2#	GPIO2 input	HP_BUTTON_B#
AE1	I/O, PU	PORT_GOOD3#	SHP_PERST3#	GPIO3 input	HP_CLKEN_B#
AB2	I/O, PU	PORT_GOOD4#	SHP_PERST4#	GPIO4 input	HP_MRL_B#
AD1	I/O, PU	PORT_GOOD5#	SHP_PERST5#	GPIO5 input	HP_PERST_B#
W5	I/O, PU	PORT_GOOD6#	SHP_PERST6#	GPIO6 input	HP_PRSENT_B#
Y3	I/O, PD	PORT_GOOD7#	SHP_PERST7#	GPIO7 input	HP_PWR_GOOD_B
AC2	I/O, PD	PORT_GOOD8#	SHP_PERST8#	GPIO8 input	HP_PWREN_B
AC3	I/O, PU	PORT_GOOD9#	SHP_PERST9#	GPIO9 input	HP_PWRFLT_B#
AD2	I/O, PU	PORT_GOOD10#	SHP_PERST10#	GPIO10 input	HP_PWRLED_B#
D3	I/O, PU	PORT_GOOD0#	SHP_PERST0#	GPIO0 input	PORT_GOOD0#

NDA CONFIDENTIAL -- Inventec | King Michael

## 10.10 HP\_PWREN\_x Output Polarity Control

HP\_PWREN\_x output polarity is configured by the SPARE1 input:

- **SPARE1 input is pulled or tied High to VDD18, or left unconnected (and therefore High due to the internal pull-up resistor)** – HP\_PWREN\_x outputs are Active-High. Therefore, when the slot power is turned On (Port's Slot Control register *Power Controller Control* bit (Downstream Ports, offset 80h[10]) is Cleared), the HP\_PWREN\_x outputs are driven High.
- **SPARE1 input is pulled or tied Low to VSS (Ground)** – HP\_PWREN\_x outputs are Active-Low. Therefore, when the slot power is turned On, the HP\_PWREN\_x outputs are driven Low.

HP\_PWREN\_x output polarity *cannot* be changed by serial EEPROM, I<sup>2</sup>C/SMBus, nor software. Polarity of I<sup>2</sup>C I/O Expander PWREN output(s) is *not* programmable.

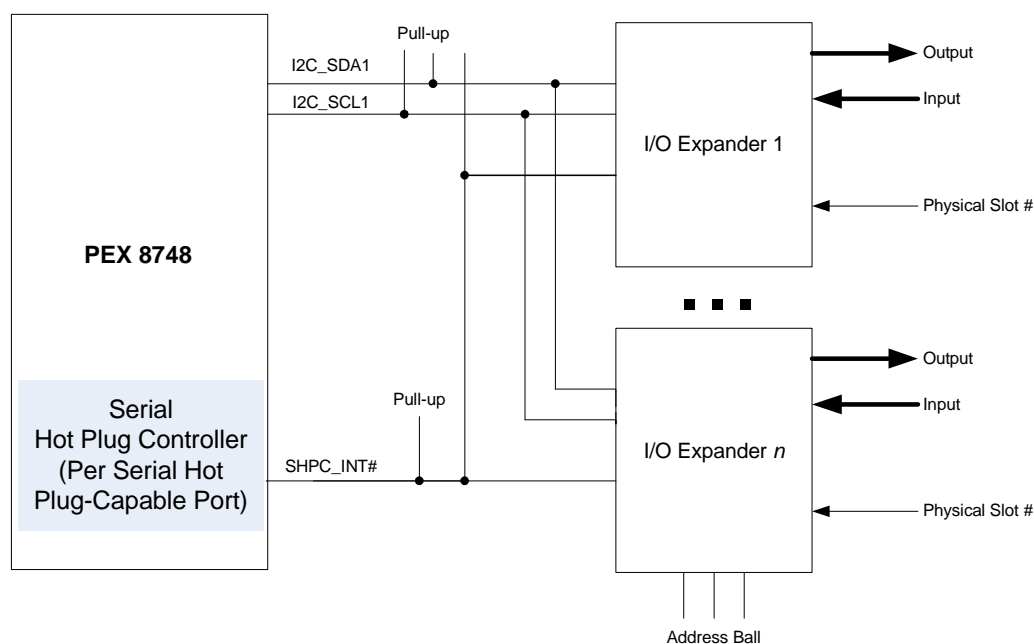


## 10.11 Serial Hot Plug Controller

**Note:** The I<sup>2</sup>C Master interface is described in this section. The Master capabilities are limited to the Serial Hot Plug Controller.

Using I/O Expander ICs sitting on an I<sup>2</sup>C Bus, the PEX 8748 has the option of Hot Plug capability on all its Transparent Downstream Ports. Figure 10-4 illustrates the internal Serial Hot Plug Controller interface. The Serial Hot Plug Controller controls the output Ports on the I/O Expanders and retrieves the Port status, *such as* device connect status, Power Fault, and MRL Sensor position, from all I/O Expanders. When there is an input change to an I/O Expander, an INTx interrupt from an I/O Expander goes Low and the PEX 8748 reads the I/O Expander. When an I/O Expander output Port requires updating with a new value, the PEX 8748 writes to the I/O Expander through the I<sup>2</sup>C Bus.

**Figure 10-4. Serial Hot Plug Interface Diagram**



### 10.11.1 Hot Plug Operations by way of I<sup>2</sup>C I/O Expander

When software issues a Slot Power On command, the Serial Hot Plug Controller issues an I<sup>2</sup>C Write to the I<sup>2</sup>C I/O Expander, to assert the **PWREN** output on the I/O Expander, and thereby turn On the power. After the Write is complete, either the **PWR\_GOOD** input is sampled asserted or the  $T_{pepv}$  time has elapsed, the Serial Hot Plug Controller issues another Write to the I/O Expander, to assert its **REFCLKEN#** output, and thereby turn On the Reference Clock (REFCLK) at the slot.

**Note:** The  $T_{pepv}$  value is used when the **PWR\_GOOD** input is not used, as indicated by the Port's **Power Management Hot Plug User Configuration** register **HPC Tpepv** field (Downstream Ports, offset **F70h[4:3]**) not being Cleared (Cleared is the default).

After the Serial Hot Plug **REFCLKEN#** output is asserted, the Serial Hot Plug Controller waits 100 ms, then issues another Write, to de-assert the Serial Hot Plug **PERST#** output to the slot. If there are commands, *such as* Attention or Power LED changes along with the Power Control command, the Serial Hot Plug Controller includes the LED output value change, along with these Writes, to change the LED status. The same procedure applies to other commands, *such as* Port Power Off. After the Serial Hot Plug Controller completes all Write operations, it Sets the Port's **Slot Status** register **Command Completed** bit (Downstream Ports, offset **80h[20]**). When another command is issued before the current command completes, the results are undefined. With a 100-kHz I<sup>2</sup>C clock, the time required to complete one Write operation to an I/O Expander is approximately 1 ms.

## 10.11.2 I<sup>2</sup>C I/O Expander Parts Selection and Pin Definition

Two types of I<sup>2</sup>C I/O Expanders can be used for Serial Hot Plug:

- **16-bit device** – For the 16-bit device, the 7-bit I<sup>2</sup>C Address must be 010X\_XXXb; a Maxim MAX7311, NXP PCA9555, or TI PCA9555 is recommended. I/O Expander addresses must begin with the lowest address (0100\_000b), and increment sequentially (corresponding to increasing PEX 8748 Port Numbers), for each device. For MAX7311 (which supports more than the eight addresses provided by the other 16-bit devices), the 7-bit I<sup>2</sup>C addresses can be in two ranges, 010X\_XXXb and 1010\_XXXb. A 16-bit device (and its 16 I/O pins) supports one Serial Hot Plug Port. (Refer to [Figure 10-5](#).)

For further details, refer to the manufacturer's data sheets for the Maxim MAX7311, NXP PCA9555, or TI PCA9555.

- **40-bit device** – For the 40-bit device, the 7-bit I<sup>2</sup>C Address must be within two ranges, 0100\_XXXb and 1010\_XXXb; an NXP PCA9698 is recommended. I/O Expander addresses must begin with the lowest address, and increment sequentially (corresponding to increasing PEX 8748 Port Numbers), for each device. The lower 32 I/O pins are used for two Ports. A 40-bit device can support two sets of Serial Hot Plug pins. The two sets are indicated with suffix PX and PY, in [Figure 10-6](#).

For further details, refer to the manufacturer's data sheet for the NXP PCA9698.

To provide Hot Plug services on all Transparent Downstream Ports, the PEX 8748 can concurrently support (refer to the [Power Management Hot Plug User Configuration](#) register *40-Pin I/O Expander Scan Disable* bit (offset F70h[17]):

- Up to 12, 16-bit I/O Expanders (for up to 11 Transparent Downstream Ports, plus one for optional Slot ID load), –or–
- 6, 40-bit I/O Expanders (for up to 11 Transparent Downstream Ports), –or–
- Mix of 16- and 40-bit I/O Expanders

The NXP PCA9555 and TI PCA9555 devices have only eight programmable I<sup>2</sup>C addresses; therefore, the maximum quantity of Serial Hot Plug-capable Ports with these I/O Expanders is eight. However, the Maxim MAX7311 can have up to 64 programmable Slave addresses; therefore, if using the MAX7311 I/O Expanders, all Transparent Downstream Ports can be made Hot Plug-capable.

[Table 10-7](#) defines the external I<sup>2</sup>C I/O Expander pins, in location order.

**Table 10-7. External I<sup>2</sup>C I/O Expander Pin Definitions used for Serial Hot Plug, by Location**

Signal Name	Direction	Description	Location	
			16-Bit Device <sup>a</sup>	40-Bit Device <sup>b</sup>
PWRLED#	O	<b>Serial Hot Plug Power LED Output</b> Same function as HP_PWRLED_x#.	IO0_0 or P00	IO0_0 IO2_0
ATNLED#	O	<b>Serial Hot Plug Attention LED Output</b> Same function as HP_ATNLED_x#.	IO0_1 or P01	IO0_1 IO2_1
PWREN	O	<b>Serial Hot Plug Power Enable Output</b> Same function as HP_PWREN_x.	IO0_2 or P02	IO0_2 IO2_2
REFCLKEN#	O	<b>Serial Hot Plug Reference Clock Enable Output</b> Same function as HP_CLKEN_x#.	IO0_3 or P03	IO0_3 IO2_3
PERST#	O	<b>Serial Hot Plug Reset Output</b> Same function as HP_PERST_x#.	IO0_4 or P04	IO0_4 IO2_4
INTERLOCK	O	<b>Serial Hot Plug Electromechanical Interlock Output Control</b> Used to physically lock the adapter in place, until software releases it. The signal default is 0. The current state of the Electromechanical Interlock is reflected in the <b>Slot Status</b> register <i>Electromechanical Interlock Status</i> bit (Downstream Serial Hot Plug-Enabled Ports, offset 80h[23]). This output can be toggled by writing 1 to the <b>Slot Control</b> register <i>Electromechanical Interlock Control</i> bit (Downstream Serial Hot Plug-Enabled Ports, offset 80h[11]). A Write of 0 has no effect. INTERLOCK is enabled when the <b>Slot Capability</b> register <i>Electromechanical Interlock Present</i> bit (Downstream Serial Hot Plug-Enabled Ports, offset 7Ch[17]) is Set (default for Serial Hot Plug-capable Transparent Downstream Ports).	IO0_5 or P05	IO0_5 IO2_5

**Table 10-7. External I<sup>2</sup>C I/O Expander Pin Definitions used for Serial Hot Plug, by Location (Cont.)**

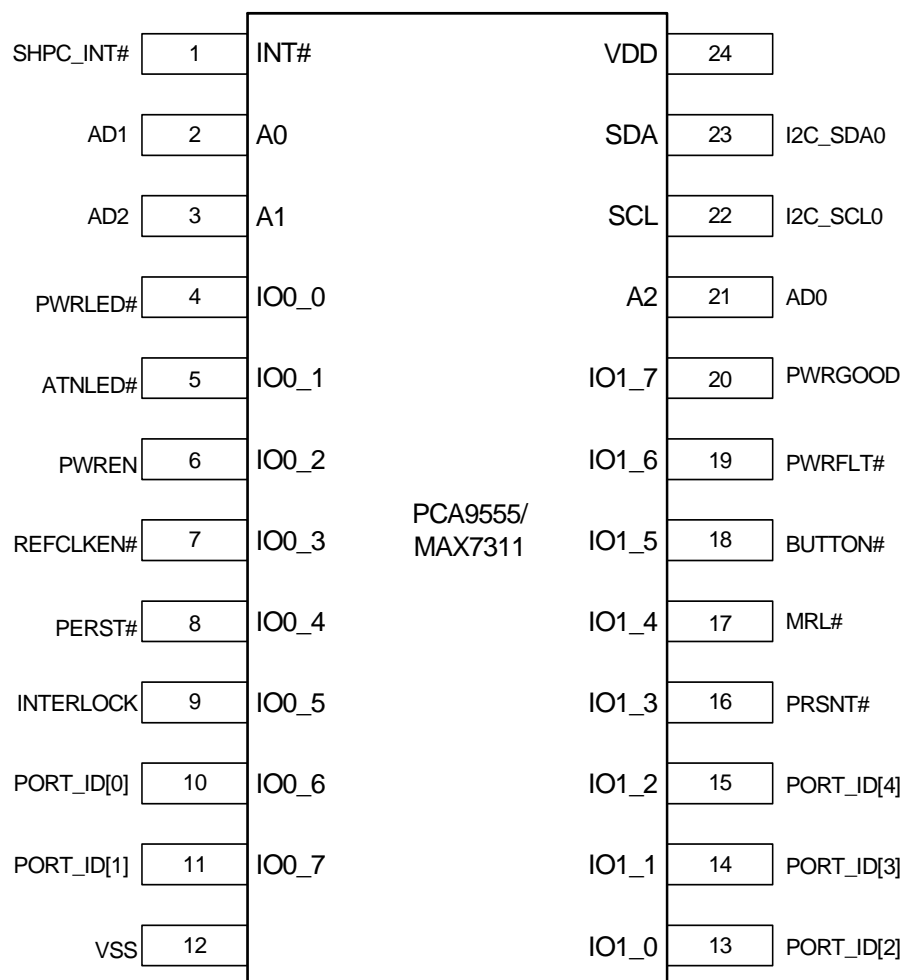
Signal Name	Direction	Description	Location	
			16-Bit Device <sup>a</sup>	40-Bit Device <sup>b</sup>
PORTID[4:0]	I	<p><b>Serial Hot Plug Port ID Straps</b></p> <p><i>Note:</i> Table 4-1 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p> <p>Configures to which Downstream Port this I<sup>2</sup>C I/O Expander maps.</p> <p>0_0000b = Port 0  0_0001b = Port 1  0_0010b = Port 2  0_0011b = Port 3  0_1000b = Port 8  0_1001b = Port 9  0_1010b = Port 10  0_1011b = Port 11  1_0000b = Port 16  1_0001b = Port 17  1_0010b = Port 18  1_0011b = Port 19</p> <p>1_1111b is valid only for 16-bit I/O Expanders, for loading the Port's <b>Slot Capability</b> register <i>Physical Slot Number</i> field (Downstream Ports, offset 7Ch[31:19]) from the I/O Expander SLOTID[12:5] inputs.</p> <p>All other encodings are <i>Reserved</i>.</p>	{IO1_2:0, IO0_7:6} or {P1[2:0], P0[7:6]}	{IO1_2:0, IO0_7:6} {IO3_2:0, IO2_7:6}
PRSNT#	I	<p><b>Serial Hot Plug PRSNT2# Input</b></p> <p>Same function as HP_PRSNT_x#.</p>	IO1_3 or P13	IO1_3 IO3_3
MRL#	I	<p><b>Serial Hot Plug Manually Operated Retention Latch Sensor Input</b></p> <p>Same function as HP_MRL_x#.</p>	IO1_4 or P14	IO1_4 IO3_4
BUTTON#	I	<p><b>Serial Hot Plug Attention Button Input</b></p> <p>Same function as HP_BUTTON_x#.</p>	IO1_5 or P15	IO1_5 IO3_5
PWRFLT#	I	<p><b>Serial Hot Plug Power Fault Input</b></p> <p>Same function as HP_PWRFLT_x#.</p>	IO1_6 or P16	IO1_6 IO3_6

**Table 10-7. External I<sup>2</sup>C I/O Expander Pin Definitions used for Serial Hot Plug, by Location (Cont.)**

Signal Name	Direction	Description	Location	
			16-Bit Device <sup>a</sup>	40-Bit Device <sup>b</sup>
PWR_GOOD	I	<b>Hot Plug Power Good Input</b> Same function as HP_PWR_GOOD_x.	IO1_7 or P17	IO1_7 IO3_7
SLOTID[12:5]	I	<b>Hot Plug Slot ID</b> Defines the value of the upper 8 bits of the Port's <b>Slot Capability</b> register <i>Physical Slot Number</i> field (bits [31:24] of Downstream Ports, offset 7Ch[31:19]; the lower five bits ([23:19]) are automatically programmed to be equal to the Serial Hot Plug-capable Port's Port Number). The 40-bit I <sup>2</sup> C I/O Expander has provision for two sets of SLOTID[12:5] inputs, for two Serial Hot Plug-capable Ports. A 16-bit I/O Expander can be used: <ul style="list-style-type: none"> <li>• For Serial Hot Plug functionality, –or–</li> <li>• Exclusively for SLOTID[12:5] input, with that 8-bit value applied globally to all Hot Plug-capable Ports (Parallel and Serial)</li> </ul>	{IO0_5:0, IO1_7:6} or {P0[5:0], P1[7:6]}	IO4_7:0

a. Refer to [Figure 10-5](#) for pinout.

b. Refer to [Figure 10-6](#) for pinout.

**Figure 10-5. 16-Bit I<sup>2</sup>C I/O Expander Pinout**

**Figure 10-6. 40-Bit I<sup>2</sup>C I/O Expander Pinout**

I2C_SDA1	1	SDA	RESET#	56	
I2C_SCL1	2	SCL	INT#	55	SHPC_INT#
PWRLED# (PX)	3	IO0_0	IO4_7	54	SLOTID[12]
ATNLED# (PX)	4	IO0_1	IO4_6	53	SLOTID[11]
PWREN (PX)	5	IO0_2	IO4_5	52	SLOTID[10]
	6	VSS	VSS	51	
REFCLKEN# (PX)	7	IO0_3	IO4_4	50	SLOTID[9]
PERST# (PX)	8	IO0_4	IO4_3	49	SLOTID[8]
INTERLOCK (PX)	9	IO0_5	IO4_2	48	SLOTID[7]
PORTID[0] (PX)	10	IO0_6	IO4_1	47	SLOTID[6]
	11	VSS	VDD	46	
PORTID[1] (PX)	12	IO0_7	IO4_0	45	SLOTID[5]
PORTID[2] (PX)	13	IO1_0	IO3_7	44	PWRGD (PY)
PORTID[3] (PX)	14	IO1_1	IO3_6	43	PWRFLT# (PY)
PORTID[4] (PX)	15	IO1_2	IO3_5	42	BUTTON# (PY)
PRSNT# (PX)	16	IO1_3	IO3_4	41	MRLN# (PY)
MRLN# (PX)	17	IO1_4	IO3_3	40	PRSNT# (PY)
	18	VDD	VSS	39	
BUTTON# (PX)	19	IO1_5	IO3_2	38	PORTID[4] (PY)
PWRFLT# (PX)	20	IO1_6	IO3_1	37	PORTID[3] (PY)
PWRGD (PX)	21	IO1_7	IO3_0	36	PORTID[2] (PY)
PWRLED# (PY)	22	IO2_0	IO2_7	35	PORTID[1] (PY)
	23	VSS	VSS	34	
ATNLED# (PY)	24	IO2_1	IO2_6	33	PORTID[0] (PY)
PWREN (PY)	25	IO2_2	IO2_5	32	INTERLOCK (PY)
REFCLKEN# (PY)	26	IO2_3	IO2_4	31	PERST# (PY)
	27	AD0	OE	30	
	28	AD1	AD2	29	

PCA9698



### 10.11.3 Serial Hot Plug Port Enumeration, Assignment, and Initialization

Serial Hot Plug can be implemented using either 16- or 40-bit I<sup>2</sup>C I/O Expanders, or a combination of both. The PEX 8748 Serial Hot Plug Controller has the intelligence to discover and differentiate between the 16- and 40-bit I/O Expanders, and assigns the I/O Expanders to a corresponding Transparent Downstream Port. The 16-bit I/O Expanders can support one Serial Hot Plug Port, and 40-bit I/O Expanders can support one or two Serial Hot Plug Ports.

After **PEX\_PERST#** and/or **VSx\_PERST#** input de-asserts and the serial EEPROM (if present) load completes, the Serial Hot Plug Controller scans the I<sup>2</sup>C Bus for I/O Expanders, starting with Device address 0100\_000b. If the Controller receives an Acknowledge (ACK) from the I/O Expander, it performs a Device ID code Read from that I/O Expander, to detect the presence of a 40-bit I/O Expander (40-bit device scan is enabled, by default). This scan can be disabled, by Setting the Transparent Downstream Port's **Power Management Hot Plug User Configuration** register *40-Pin I/O Expander Scan Disable* bit (offset **F70h**[17]). If the Device ID code Read fails, that I/O Expander is identified as a 16-bit I/O Expander, and the Serial Hot Plug **PORTID**[4:0] value on the 16-bit I/O Expander determines which PEX 8748 Port is associated with the I/O Expander. If the Device ID code Read fetches the correct Device ID, that I/O Expander is identified as being 40-bit capable. The two PEX 8748 Serial Hot Plug Ports that correspond to the two sets of Serial Hot Plug pins in a 40-bit I/O Expander are determined from two unique **PORTID**[4:0] values. (Refer to [Table 10-7](#) for valid **PORTID**[4:0] values.)

The Serial Hot Plug Controller logic uses the I<sup>2</sup>C Master interface to program the I/O Expander's I/O Configuration registers, and Sets the initial Hot Plug state for successfully scanned I/O Expanders. (Refer to [Table 10-2](#) for the initial states of the Hot Plug outputs.) On an I/O Expander scan, if the Serial Hot Plug Controller receives a Negative Acknowledge (NAK), the Hot Plug Controller stops scanning for I<sup>2</sup>C I/O Expanders. After the I<sup>2</sup>C scan is complete, the Serial Hot Plug Controller starts the slot power-on sequence for Ports in which the following is true:

- **MRL#** input is sampled Low (Port's **Slot Status** register *MRL Sensor State* bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset **80h**[21]) is Cleared), and
- Port's **Slot Control** register *Power Controller Control* bit (Downstream Ports, offset **80h**[10]) is not Set by the serial EEPROM load

The I<sup>2</sup>C address for the I/O Expanders must be contiguous, with the first I/O Expander's I<sup>2</sup>C address programmed to 0100\_000b, the next programmed to 0100\_001b, and so forth.

### 10.11.4 I<sup>2</sup>C I/O Expander Interrupt Processing

The I/O Expander Interrupt outputs (INT#) must all be connected together in an Open Drain (OD) manner, to the PEX 8748's SHPC\_INT# input. When an I/O Expander input state changes on any of the I/O Expanders, the SHPC\_INT# input is asserted Low. The Serial Hot Plug Controller, through the I<sup>2</sup>C Bus, scans the I/O Expanders, starting with address 40h, until the SHPC\_INT# input de-asserts High (as a result of the I<sup>2</sup>C Read), signaling which I/O Expander asserted the interrupt (INT#). SHPC\_INT# is internally de-bounced for 10 ms; therefore, if SHPC\_INT# asserts Low for less than 10 ms, its assertion is ignored. The 10 ms de-bounce on the SHPC\_INT# input can be disabled, by Setting the Port's **Power Management Hot Plug User Configuration** register *Serial Hot Plug INTx De-Bounce Disable* bit (offset F70h[18]). SHPC\_INT# assertion due to an I/O Expander input state change can cause a corresponding PEX 8748 **Slot Status** register bit to be Set, and cause the Port to send an INTx Message to the Root Complex, if the corresponding interrupt is enabled.

### 10.11.5 Serial Hot Plug-Capable Port Command Completion

For slot power On or Off commands from software to turn power On or Off to a specific Port, the Port's **Slot Status** register *Command Completed Interrupt Enable* bit (Downstream Ports, offset 80h[4]) is Set after the sequence of I<sup>2</sup>C Master Write operations to the I<sup>2</sup>C I/O Expander, that perform the power On or Off sequence, have completed.

### 10.11.6 Physical Slot Number Loading from I<sup>2</sup>C I/O Expander

The **Slot Capability** register *Physical Slot Number* field (Downstream Ports, offset 7Ch[31:19]) in Serial Hot Plug Ports must be assigned a unique, non-zero Port identifier within the system. The register's MSB [31:24] can be loaded from the SLOTID[12:5] input Settings on the I<sup>2</sup>C I/O Expander. The 40-bit I/O Expander has SLOTID[12:5] inputs implemented along with two sets of Serial Hot Plug pins. The SLOTID[12:5] inputs define the *Physical Slot Number* field in both Ports identified by the PORTID[4:0] (PX) and PORTID[4:0] (PY) inputs. LSB [4:0] (bits [23:19]) of the Physical Slot Number are loaded with the Port's Port Number. The combination of SLOTID[12:5] and Port Number forms a unique Physical Slot Number value, for each Port.

For 16-bit I/O Expanders, the PORTID[4:0] input value enables one of two possible implementations:

- If the PORTID[4:0] value matches a valid Downstream Port Number, the I/O Expander provides a set of Hot Plug pins for that designated Port, –or–
- If PORTID[4:0]=1\_1111b, this single 16-bit I/O Expander globally loads the set of {IO0\_5:0, IO1\_7:6} or {P0[5:0], P1[7:6]} as the upper 8-bits of a 13-bit Physical Slot Number, into all Hot Plug Ports. The lower five bits in each Port are initialized to the Port's Port Number, to provide a unique Physical Slot Number value for each Hot Plug-capable Port.

The 16-bit I/O Expander {IO1\_5:3} or {P1[5:3]} pins are not used for SLOTID[12:5] load. This I/O Expander itself is *not* associated to any Port, does *not* provide Hot Plug signaling, and does *not* enable Hot Plug.

**Note:** 16-bit I/O Expanders that have Serial Hot Plug PORTID[4:0] programmed as 1\_1111b *cannot* be used for Serial Hot Plug operation, because that value is used only for the Physical Slot Number field, as explained in Table 10-7. An alternative is to use the serial EEPROM to program the **Slot Capability** register.

## 10.12 Hot Plug Board Insertion and Removal Process

Table 10-8 defines the board insertion procedure supported by the PEX 8748. Table 10-9 defines the board removal procedure. Both processes apply to Parallel and Serial Hot Plug-capable Transparent Downstream Ports.

**Table 10-8. Hot Plug Board Insertion Process**

Operator / Action	Hot Plug Controller	Software
A. Places board in slot.	<ol style="list-style-type: none"> <li>1. Sets the Port's <b>Slot Status</b> register <i>Presence Detect State</i> bit (Downstream Ports, offset 80h[22]).</li> <li>2. Sets the Port's <b>Slot Status</b> register <i>Presence Detect Changed</i> bit (Downstream Ports, offset 80h[19]).</li> <li>3. Generates Interrupt Message due to <i>Presence Detect Changed</i> event, if enabled.</li> </ol>	Clears the <i>Presence Detect Changed</i> bit.
	<ol style="list-style-type: none"> <li>4. Transmits an Interrupt de-assertion Message, if enabled.</li> </ol>	
B. Locks MRL.	<ol style="list-style-type: none"> <li>5. Clears the Port's <b>Slot Status</b> register <i>MRL Sensor State</i> bit (Downstream Ports, offset 80h[21]).</li> <li>6. Sets the Port's <b>Slot Status</b> register <i>MRL Sensor Changed</i> bit (Downstream Ports, offset 80h[18]).</li> <li>7. Generates an Interrupt Message due to <i>MRL Sensor Changed</i> event, if enabled.</li> </ol>	Clears the <i>MRL Sensor Changed</i> bit.
	<ol style="list-style-type: none"> <li>8. Transmits an Interrupt de-assertion Message, if enabled.</li> </ol>	
C. Presses Attention Button.	<ol style="list-style-type: none"> <li>9. Sets the Port's <b>Slot Status</b> register <i>Attention Button Pressed</i> bit (Downstream Ports, offset 80h[16]).</li> <li>10. Generates an Interrupt Message due to <i>Attention Button Pressed</i> event, if enabled.</li> </ol>	Clears the <i>Attention Button Pressed</i> bit.
	<ol style="list-style-type: none"> <li>11. Transmits an Interrupt de-assertion Message, if enabled.</li> </ol>	Programs the Port's <b>Slot Control</b> register <i>Power Indicator Control</i> field (Downstream Ports, offset 80h[9:8]) value to 10b, to blink the Power Indicator LED, which indicates that the board is being powered up.
Continued...		

**Table 10-8. Hot Plug Board Insertion Process (Cont.)**

Operator / Action	Hot Plug Controller	Software
D. Power Indicator blinks.	<b>12.</b> Blinks the Power Indicator LED. <b>13.</b> Sets the Port's <b>Slot Status</b> register <i>Command Completed</i> bit (Downstream Ports, offset 80h[20]). <b>14.</b> Generates an Interrupt Message due to Power Indicator Blink command Completion, if enabled.	Clears the <i>Command Completed</i> bit.
	<b>15.</b> Transmits an Interrupt de-assertion Message, if enabled.	Clears the Port's <b>Slot Control</b> register <i>Power Controller Control</i> bit (Downstream Ports, offset 80h[20]), to turn On power to the Port.
	<b>16.</b> Slot is powered up. <b>17.</b> After the HP_PWR_GOOD <sub>x</sub> inputs are sampled asserted High or T <sub>pepv</sub> delay, Sets the <i>Command Completed</i> bit. <b>18.</b> Generates an Interrupt Message due to Power Turn On command Completion, if enabled.	Clears the <i>Command Completed</i> bit.
	<b>19.</b> Transmits an Interrupt de-assertion Message, if enabled.	Programs the Port's <b>Slot Control</b> register <i>Power Indicator Control</i> field (Downstream Ports, offset 80h[9:8]) value to 01b, to turn On the Power Indicator LED, which indicates that the slot is fully powered On.
E. Power Indicator On.	<b>20.</b> Turns On the Power Indicator LED. <b>21.</b> Transmits an Interrupt assertion Message due to Power Indicator Turn On command Completion, if enabled.	Clears the <i>Command Completed</i> bit.
	<b>22.</b> Transmits an Interrupt de-assertion Message, if enabled. <b>23.</b> After the Data Link Layer is up, Sets the Port's <b>Slot Status</b> register <i>Data Link Layer State Changed</i> bit (Downstream Ports, offset 80h[24]), and transmits the corresponding interrupt, if enabled.	Software can now read the Port's <b>Link Status</b> register <i>Data Link Layer Link Active</i> bit (Downstream Ports, offset 78h[29]). A value of 1 in this bit indicates that the board is ready to be used.  Clears the <i>Data Link Layer State Changed</i> bit and interrupt, if enabled.
	<b>24.</b> Transmits an Interrupt de-assertion Message, if enabled.	

**Table 10-9. Hot Plug Board Removal Process**

Operator / Action	Hot Plug Controller	Software
A. Presses Attention Button.	<ol style="list-style-type: none"> <li>1. Sets the Port's <b>Slot Status</b> register <i>Attention Button Pressed</i> bit (Downstream Ports, offset 80h[16])</li> <li>2. Generates an Interrupt Message due to Attention Button pressed, if enabled.</li> </ol>	Clears the <i>Attention Button Pressed</i> bit.
	<ol style="list-style-type: none"> <li>3. Transmits an Interrupt de-assertion Message, if enabled.</li> </ol>	Programs the Port's <b>Slot Control</b> register <i>Power Indicator Control</i> field (Downstream Ports, offset 80h[9:8]) value to 10b, to blink the Power Indicator LED, which indicates that the board is being powered down.
B. Power Indicator blinks.	<ol style="list-style-type: none"> <li>4. Blinks the Power Indicator LED.</li> <li>5. Sets the Port's <b>Slot Status</b> register <i>Command Completed</i> bit (Downstream Ports, offset 80h[20]).</li> <li>6. Generates an Interrupt Message due to Power Indicator Blink command Completion, if enabled.</li> </ol>	Clears the <i>Command Completed</i> bit.
	<ol style="list-style-type: none"> <li>7. Transmits an Interrupt de-assertion Message, if enabled.</li> </ol>	Sets the Port's <b>Slot Control</b> register <i>Power Controller Control</i> bit (Downstream Ports, offset 80h[20]), to turn Off power to the Port.
C. Power Indicator Off.	<ol style="list-style-type: none"> <li>8. Slot is powered Off.</li> <li>9. Sets the Port's <b>Slot Status</b> register <i>Data Link Layer State Changed</i> bit (Downstream Ports, offset 80h[24]), and transmits an interrupt, if enabled.</li> <li>10. After a 256-ms delay from HP_PWR_GOOD_x sampled de-asserted (if the HP_PWR_GOOD_x inputs are enabled through the Port's <b>Power Management Hot Plug User Configuration</b> register <i>HPC Tpepv</i> field (Downstream Ports, offset F70h[4:3], is Cleared)), Sets the <i>Command Completed</i> bit.</li> <li>11. Generates an Interrupt Message due to Power Turn Off command Completion, if enabled.</li> </ol>	<p>Clears the <i>Data Link Layer State Changed</i> bit and interrupt.</p> <p>Clears the <i>Command Completed</i> bit.</p> <p>Programs the <i>Power Indicator Control</i> field value to 11b, to turn Off the Power Indicator LED, which indicates that the slot is fully powered Off and the board can be removed.</p>
D. Power Indicator Off, board ready to be removed.	<ol style="list-style-type: none"> <li>12. Turns Off the Power Indicator LED.</li> <li>13. Sets the <i>Command Completed</i> bit, due to Power Indicator Off command Completion.</li> </ol>	Clears the <i>Command Completed</i> bit.
	<ol style="list-style-type: none"> <li>14. Transmits an Interrupt de-assertion Message, if enabled.</li> </ol>	
Continued...		

**Table 10-9. Hot Plug Board Removal Process (Cont.)**

Operator / Action	Hot Plug Controller	Software
E. Unlocks MRL.	<p><b>15.</b> Sets the Port's <b>Slot Status</b> register <i>MRL Sensor State</i> bit (Downstream Ports, offset 80h[21]).</p> <p><b>16.</b> Sets the Port's <b>Slot Status</b> register <i>MRL Sensor Changed</i> bit (Downstream Ports, offset 80h[18]).</p> <p><b>17.</b> Generates an Interrupt Message due to MRL Sensor state change, if enabled.</p>	Clears the <i>MRL Sensor Changed</i> bit.
	<b>18.</b> Transmits an Interrupt de-assertion Message, if enabled.	
F. Removes board from slot.	<p><b>19.</b> Clears the Port's <b>Slot Status</b> register <i>Presence Detect State</i> bit (Downstream Ports, offset 80h[22]).</p> <p><b>20.</b> Sets the Port's <b>Slot Status</b> register <i>Presence Detect Changed</i> bit (Downstream Ports, offset 80h[19]).</p> <p><b>21.</b> Generates an Interrupt Message due to Presence Detect change, if enabled.</p>	Clears the <i>Presence Detect Changed</i> bit.
	<b>22.</b> Transmits an Interrupt de-assertion Message, if enabled.	

## 11.1 Overview

The PEX 8748 Power Management (PM) features provide the following services:

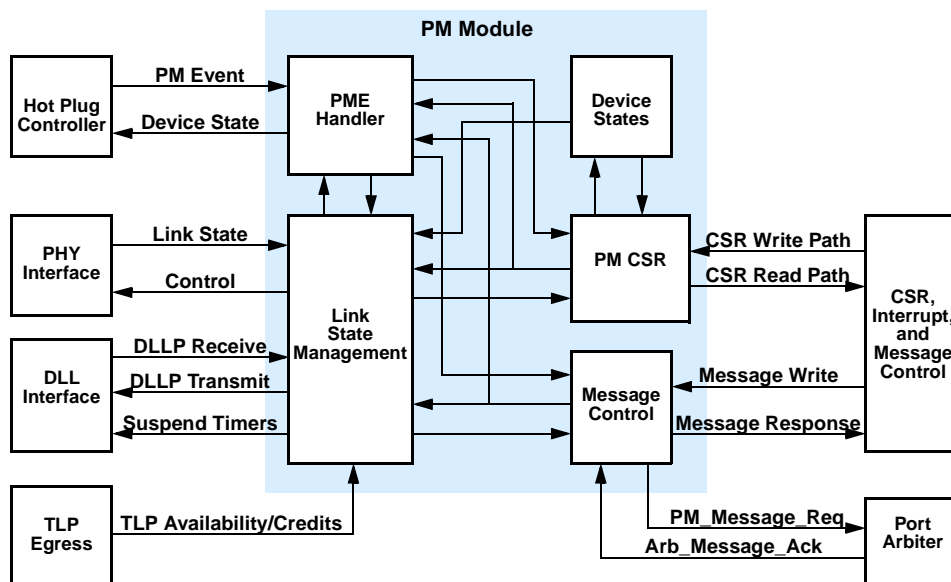
- Mechanisms to identify PM capabilities
- Ability to transition into certain PM states
- Notification of the current PM state of each Port
- Support for the option to wakeup the system upon a specific event

The PEX 8748 supports hardware-autonomous PM and software-driven D-State PM. The switch also supports the L0s and L1 Link PM states in hardware-autonomous Active State Power Management (ASPM), as well as the L1, L2/L3 Ready, and L3 Link PM states in Conventional PCI-compatible PM. D0, D3hot, and D3cold states are supported in Conventional PCI-compatible PM. Because the PEX 8748 does **not support** Vaux, Power Management Event (PME) generation from the D3cold state is **not supported**.

The PM module interfaces with a Physical Layer (PHY) electrical sub-block, to transition the Link state into a low-power state, when the module receives a Power State Change Request from a Downstream component, or an internal event forces the Link state entry into low-power states in hardware-autonomous ASPM mode. PCI Express Link states are not directly visible to Conventional PCI Bus driver software; however, they are derived from the PM state of the components residing on those Links.

Figure 11-1 provides a functional block diagram of the PEX 8748 PM module.

**Figure 11-1. PM Module Functional Block Diagram**



**Note:** The Hot Plug Controller is available only on Hot Plug-capable Transparent Downstream Ports.

## 11.2 Power Management Features

- *PCI Express Base r3.0*-compliant
- *PCI Power Mgmt. r1.2*-compliant
- Link Power Management States (*L-States*; also referred to as *Link PM states*)
  - PCI Bus Power Management – L1, L2/L3 Ready, and L3 (Vaux is ***not supported***)
  - Active State Power Management (ASPM) – L0s and L1
- Device Power Management State (*D-States*; also referred to as *PCI Express PCI-PM states*)
  - D0 (D0 Uninitialized and D0 Active) and D3 (D3hot and D3cold) support
- Power Management Event (PME) support from D3hot
- PME due to Hot Plug and/or PCI Express Hot Plug events
- Forwards PME\_Turn\_Off broadcast Messages
- Implements Gen 3-specific Control and Status register and associated interrupts
- Supports ASPM L0s, ASPM L1, PCI PM L1, and L2/L3 Ready Link PM states in NT mode, as well as Virtual Switch mode



## 11.3 Power Management Capability

### 11.3.1 Device Power Management States

The PEX 8748 supports the PCI Express PCI-PM D0 and D3hot states. The PCI-PM D1 and D2 states, which are optional in the *PCI Express Base r3.0*, are **not supported**.

The D3hot state can be entered from the D0 Uninitialized or D0 Active states, when system software programs the Port's **PCI Power Management Status and Control** register *Power State* field (offset 44h[1:0]) to 11b. The PCI-PM D0 Uninitialized state can be entered from the D3hot state when system software Clears the Port's *Power State* field.

#### 11.3.1.1 D0 State

A PCI Express device must be put into the PCI Express PCI-PM D0 state before it can be used. When the device enters the D0 state following a Fundamental Reset, it is in the D0 Uninitialized state. A subsequent Reset forces the device into the D0 Uninitialized state.

Other than the Fundamental Reset that brings the device to the D0 Uninitialized state, all other PM state transitions are explicitly controlled by software.

A device enters the PCI Express PCI-PM D0 Active state when system software Sets any combination of the **PCI Command** register *Bus Master Enable*, *Memory Access Enable*, and/or *I/O Access Enable* bits (offset 04h[2, 1, and/or 0], respectively).

#### 11.3.1.2 D3hot State

In the PCI Express PCI-PM D3hot state, Hot Plug or Link State operations can cause the PEX 8748 to generate a PME.

Only Type 0 Configuration accesses are allowed in the D3hot state. Memory and I/O transactions result in an Unsupported Request (UR). Completions flowing in either direction are not affected.

Type 1 transactions flowing toward a PEX 8748 Port in the D3hot state are terminated as URs. Type 0 Configuration transactions complete successfully. When an PEX 8748 Upstream Port is programmed to the D3hot state, the Port initiates Conventional PCI-PM L1 Link PM state entry.

## 11.3.2 Link Power Management States

PEX 8748 components hold their Upstream and Downstream Links in the L0 Link PM state during standard device operation (PCI Express PCI-PM D0 Active state). ASPM defines a mechanism for components in the PCI Express PCI-PM D0 state, to reduce Link power by placing their Links into a low-power state and instructing the other end of the Link to do likewise. This capability allows hardware-autonomous, dynamic Link power reduction beyond what is achievable by software-only-controlled PM. Table 11-1 defines the relationship between a component's Power state and Upstream Link. Table 11-2 defines the relationship between Link PM states and power-saving actions.

PCI Express PCI-PM L1 and L2/L3 Ready Link PM states are controlled by software programming the PEX 8748 into the PCI Express PCI-PM D3hot state, and subsequently causing the Root Complex to broadcast the PME\_Turn\_Off Message to the Downstream hierarchy.

**Table 11-1. Relationship between Component Power State and Upstream Link**

Downstream Component PCI Express PCI-PM State	Permissible Upstream Component PCI Express PCI-PM State	Permissible Interconnect Link PM State
D0	D0	L0, and optionally L0s and L1 if ASPM is enabled
D3hot	D0-to-D3hot	L1, L2/L3 Ready.
D3cold (no Vaux)	D0-to-D3cold	L3 (off). Zero power.

**Table 11-2. Relationship between Link PM States and Power-Saving Actions**

Link PM State	Power-Saving Actions
Tx L0s	PHY Tx Lanes are in a High-Impedance state.
Rx L0s	PHY Rx Lanes are in a low-power state.
L1	PHY Tx and Rx Lanes are in a low-power state. FC timers are suspended.
L2/L3 Ready	PHY Tx and Rx Lanes are in a low-power state. FC timers are suspended.
L3 (D3cold)	Component is fully powered Off.

### 11.3.3 Active State Power Management – L0s Link PM State Transition

L0s Link PM state entry is mandatory, and this state is allowable only in Active State Link Power Management (ASPM). Each device reports its support to L0s entry through the Port's **Link Capability** register *Active State Power Management (ASPM) Support* field (offset 74h[11:10]). Software enables L0s state entry, by writing into Active state Link PM Control register. The L0s Link PM state is optimized for short entry and exit latencies, while providing substantial power savings. If the L0s state is enabled in a device, transmit Links that are not being used must be brought into the L0s state.

Link Layer Timers are *not* affected by a transition to the L0 Link PM state. When in the L0s state, Links return to the L0 state, regardless of whether there are any TLP traffic situations to transmit Flow Control (FC) updates, which occurs every 30  $\mu$ s minimum.

#### 11.3.3.1 Rx L0s Link PM State Entry

The PEX 8748 allows its Rx Lanes to go into the L0s Link PM state, regardless of whether its Tx is enabled for L0s.

#### 11.3.3.2 Rx L0s Link PM State Exit

The endpoint Transmitter at the opposite end of a PCI Express Link transmits a quantity of FTS and goes back to L0 state. If the PEX 8748 Receiver achieves symbol and bit lock, it settles into the L0 Link PM state. Otherwise, the Receiver enters the *Recovery* state.

#### 11.3.3.3 Tx L0s Link PM State Entry Conditions

##### ***L0s Link PM State Entry Conditions – Upstream Port(s)***

The following four conditions must be met for Upstream Port(s) to enter the L0s state:

1. All PEX 8748 Downstream Port Receiver Lanes are in the L0s Link PM or deeper state. If the PEX 8748 Downstream Port Link is in a *Link Down* state, this condition is also satisfied.
2. No pending TLPs (including Retry TLPs), or there are no credits available to transmit pending TLPs.
3. No DLLPs pending in the DLL.
4. Enabled in the Port's **Link Capability** register *Active State Power Management (ASPM) Support* field (offset 74h[11:10]), programmed to 01b or 11b), for going into the L0s state.

The PM module directs the Upstream Link to go into the L0s state when Conditions 1, 2, and 3 are met for a specified period of time and Condition 4 is satisfied. The idle time can be selected, as either 1  $\mu$ s (default) or 4  $\mu$ s, in the Port's **Power Management Hot Plug User Configuration** register *L0s Entry Idle Count* bit (offset F70h[0]). If these conditions are met for the specified time period of time, PM directs the PHY to the L0s state.

***L0s Link PM State Entry Conditions – Downstream Ports***

The following four conditions must be met for the Upstream Port(s) to enter the L0s state:

1. All PEX 8748 Upstream Port Receiver Lanes are in the L0s state.
2. No pending TLPs (including Retry TLPs), or there are no credits available to transmit pending TLPs.
3. No DLLPs pending.
4. Enabled in the Port's **Link Capability** register *Active State Power Management (ASPM) Support* field (offset 74h[11:10], programmed to 01b or 11b), for going into the L0s state.

The PM module directs the Downstream Link to go into the L0s state when Conditions 1, 2, and 3 are met for a specified period of time and Condition 4 is satisfied. The idle time can be selected, as either 1  $\mu$ s (default) or 4  $\mu$ s, in the Port's **Power Management Hot Plug User Configuration** register *L0s Entry Idle Count* bit (offset F70h[0]).

### 11.3.3.4 L0s Link PM State Exit Conditions

Components from either end of a PCI Express Link can initiate exit from the L0s Link state.

#### ***L0s Link PM State Exit Conditions – Upstream Port(s)***

One or more of the following conditions must be met for Upstream Port(s) to exit the L0s state:

- If there are pending DLLPs or TLPs to transmit, –or–
- If any of the Downstream Port Rx Lanes exit the L0s state. When a Downstream Port Rx Lane exits the L0s or L1 Link PM state, a Link State Change vendor-defined Message is transmitted to the Upstream Port.
- Upstream Port ASPM L1 entry conditions are met.
- Upstream Port is placed into the PCI Express PCI-PM D3hot state.
- Link is Set to retrain, as a result of a DLLP timeout or Replay Number Rollover.

#### ***L0s Link PM State Exit Conditions – Downstream Ports***

One or more of the following conditions must be met for a Downstream Port to exit the L0s state and enter the L0 Link PM state:

- If there are pending DLLPs or TLPs to transmit, –or–
- If any of the Upstream Port Rx Lanes exit the L0s state. When an Upstream Port Rx Lane exits the L0s or L1 Link PM state, a Link State Change vendor-defined Message is transmitted to all Downstream Ports.
- Port's **Link Control** register *Retrain Link* or *Link Disable* bit (Downstream Ports, offset 78h[5:4], respectively) is Set.
- If a PCI-PM L1 Entry Request or Active State L1 Request is received from the Downstream direction.

The PM module directs the PHY to change from the L0s state to the operational L0 state, when any of the conditions listed above are met. The PHY transmits the Fast Training Sets (FTS), then enters the L0 state.

### 11.3.3.5 L1 Link PM State Entry

The L1 state is a deeper power-saving state than the L0s Link PM state, with both Tx and Rx Lanes settling into Electrical Idle after L1 negotiation is complete.

A PCI Express Upstream Link can enter the L1 state when:

- Upstream Port(s) are placed into the PCI Express PCI-PM D3hot state.
- Upstream Port(s) are in the D0 Active state, and Idle conditions (explained below) are satisfied and the Port is enabled in the Port's **Link Control** register *Active State Power Management (ASPM)* field is programmed for ASPM L1 (offset 78h[1:0] is programmed to 10b or 11b)

Downstream Ports never initiate a L1 state entry. Downstream Port Links can enter the L1 state when:

- Downstream device is in the D0 state and initiates L1 state entry (Port's **Link Control** register *Active State Power Management (ASPM)* field (offset 78h[1:0]) is programmed to 10b or 11b).
- Downstream device is placed into a PCI Express PCI-PM D3hot state, and initiates L1 state entry

Entry into the L1 state is always initiated by an Upstream Port or Downstream device. ASPM L1 state is optional, and software-controlled, by writing into the *Active State Power Management (ASPM)* field. However, L1 state entry is also possible in the PCI Express PCI-PM, when an Upstream Port is placed into a state other than the PCI Express PCI-PM D0 state. This transition is required. Conditions for entering the ASPM L1 state are implementation-specific. Downstream Ports never initiate L1 state entry.

A Port cannot directly go from the L0s to L1 state while in ASPM. In ASPM, the transition is L0s -> L0 -> L1. After L1 state entry, Flow Control Update Timers are suspended.

PCI-PM-compatible L1 entry protocol uses the PM\_enter\_L1\_DLLP; however, ASPM uses PM\_active\_state\_request\_L1\_Dllp. Downstream Ports never NAK the PM\_enter\_L1\_DLLP received from a Downstream device; however, it can NAK PM\_active\_state\_request\_L1\_Dllp, if there is something to transmit.

**ASPM L1 State Entry Conditions – Upstream Port(s)**

An Upstream Port requests ASPM L1 state entry and waits up to 1 to 2  $\mu$ s, when the following conditions are met:

- Upstream Port is in the PCI Express PCI-PM D0 state, and enabled for ASPM L1 state entry (Port's **Link Control** register *Active State Power Management (ASPM)* field (offset 78h[1:0]) is programmed to 10b or 11b)
- All Downstream Port Links are in the L1 or deeper state, or the Link is Down
- No pending TLPs to transmit, including TLPs waiting for DLL ACK/NAK (in the Retry buffer)
- No pending DLLPs, except for FC DLLPs

Also, the Upstream Port should have the maximum FC credits to transmit the maximum packet size for each enabled Virtual Channel (VC) packet type. The Upstream Port suspends the transmission of new TLPs.

The Upstream Port continuously transmits PM\_active\_state\_request\_L1\_Dllp, until it sees a PM\_request\_Ack DLLP or PM\_Active\_state\_Nak TLP Message. These DLLPs are transmitted with no more a 4-symbol time gap on the Upstream Link.

After the Upstream Port receives PM\_request\_Ack DLLP, it directs the PHY to enter the L1 state. The Link settles into the L1 state when an Electrical Idle Ordered-Set (EIOS) is transmitted, and Electrical Idle is received. After a PM\_Active\_state\_Nak TLP is received, the Link enters the L0s state, as soon as possible, if conditions are met.

PEX 8748 Upstream Port when enabled for L0s and L1 ASPM, and is in L0s for more than 2  $\mu$ s and above L1 entry conditions are met, exits L0s and negotiates for L1 entry.

***Upstream Port Requests PCI Express PCI-PM L1 Entry***

An Upstream Port requests PCI Express PCI-PM L1 state entry, when the following six conditions are met:

- Software places the Upstream Port into the PCI Express PCI-PM D3hot state.
- Upstream Port suspends the scheduling of new TLPs, after a Completion for that particular packet is transmitted Upstream.
- Upstream Port waits until all previously transmitted TLPs are acknowledged. During this time, the Upstream Port receives TLPs and DLLPs. Also transmits ACK/NAK and FC DLLPs.
- Upstream Port waits for a sufficient quantity of FC credits, for all types of TLP packets.
- Upstream Port waits for all the Retry TLPs to complete.
- After the Upstream Port transmits all pending TLPs, it transmits a PM\_EnterL1\_Dllp Upstream, with no more than a 4-symbol time gap.

When the Upstream Port receives the PM\_Request\_Ack DLLP, it directs the Link to enter the L1 state.

When in the L1 state, if an Electrical Idle exit is detected, or a TLP must be transmitted, the Upstream Port exits the L1 state, transmits the pending TLP(s), if required, then re-enters the PCI Express PCI-PM L1 state after a 16- $\mu$ s delay. The Upstream Port must detect a continuous 16  $\mu$ s of the PHY L0 state for the next state entry.

***Downstream Port Accepts Request from Downstream Device PM\_active\_state\_request\_L1\_Dllp***

A Downstream Port accepts a Request from the Downstream device PM\_active\_state\_request\_L1\_Dllp and suspends the transmission of new TLPs, when the following three conditions are met:

- Downstream Port supports, and is enabled for, ASPM L1 state entry (Port's [Link Control](#) register [Active State Power Management \(ASPM\)](#) field (Downstream Ports, offset 78h[1:0]) is programmed to 10b or 11b)
- No TLPs are scheduled for transmission
- No DLLPs, other than FC DLLPs, are pending

The Downstream Port transmits PM\_request\_Ack DLLP continuously until the receive Lanes goes to Electrical Idle, which indicates the other side of the Link has received Ack. After it receives Electrical Idle, power management directs the PHY to go to the L1 state.

If the conditions for going to L1 are not satisfied, PEX 8748 Downstream Port transmits PM\_Active\_state\_Nak Message and it remains in the L0 state.

***Downstream Port Accepts Request from Downstream Device PM\_Enter\_L1\_Dllp***

A Downstream Port accepts a Request from the Downstream device PM\_enter\_L1\_Dllp, if the following three conditions are met:

- No TLPs are scheduled for transmission
- TLP transmissions are suspended
- Downstream Port received a Link Layer ACK for all transmitted TLPs, and gathered sufficient credits for all enabled VC NT Ports

The Downstream Port continuously transmits PM\_Enter\_Ack\_DLLP, with no more than a 4-symbol time gap between subsequent DLLPs. When the Downstream Port “sees” the Rx Lanes go into Electrical Idle, the Port directs the PHY to enter the L1 state.



### 11.3.3.6 L1 Link PM State Exit Conditions

Devices on either end of a PCI Express Link can initiate exit from the L1 state. Unlike the L1 state entry protocol, in which both ends of the Link must negotiate for the resultant Link state, L1 state exit does *not* require negotiation.

#### ***ASPM L1 Link PM State Exit Conditions – Upstream Port(s)***

The following conditions must be met for Upstream Port(s) to exit the ASPM L1 state:

- Any Downstream Port exits the L1 state.
- TLPs are pending transmission.
- Any Upstream Port Rx Lane exits Electrical Idle.
- Upstream Port is placed in the PCI Express PCI-PM D3hot state.

#### ***PCI Express PCI-PM D3hot State Exit Conditions – Upstream Port(s)***

One or more of the following conditions must be met for Upstream Port(s) to exit the PCI Express PCI-PM D3hot state:

- Any Upstream Port Rx Lane exits Electrical Idle, –or–
- TLPs are pending transmission. (This occurs if a PME Message transmission is pending, due to Hot Plug.)
- Downstream Port exited the L1 state as the result of a Downstream Port Rx Lane exiting Electrical Idle while in the L1 state.

#### ***ASPM L1 Link PM State Exit Conditions – Downstream Ports***

The following conditions must be met for a Downstream Port to exit the ASPM L1 state:

- Any Downstream Rx Lanes exit Electrical Idle.
- TLPs are pending transmission on the Link.
- Upstream Port starts to exit the L1 state. Downstream Ports, which are in the L1 state as the result of a Downstream device being placed into the PCI Express PCI-PM D3hot state, are not affected by this transition.

#### ***PCI Express PCI-PM D3hot State Exit Conditions – Downstream Ports***

The following conditions must be met for a Downstream Port to exit the PCI Express PCI-PM D3hot state:

- Any Downstream Rx Lanes exit Electrical Idle.
- TLPs are pending transmission on the Link.

### 11.3.3.7 L1 Link PM State Entry Negotiation Interruption – ASPM and PCI Express PCI-PM

Per the *PCI Express Base r3.0*, when L1 state entry negotiation is interrupted, the state machine on both ends of the PCI Express Link should return to an Idle state. A Port can be in any of the states described in the sections that follow, when L1 state entry negotiation is interrupted, usually as the result of a trip through the *Recovery* state.

#### ***L1 Link PM State Entry Negotiation Interruption – Upstream Port(s)***

What occurs after L1 state entry negotiation is interrupted on the Upstream Port(s) is dependent upon the following:

- **Started requesting L1 entry, using PM\_request DLLPs** – When the Tx or Rx *Recovery* state occurs, the Port stops receiving Requests, returns to an Idle state, then waits 10  $\mu$ s before making another L1 state entry Request, –or– the Tx Lane should enter the L0s Link PM state.
- **Waiting for a PM request ACK** – When the Tx or Rx *Recovery* state occurs, the Port returns to an Idle state, then waits 10  $\mu$ s before making another L1 state entry Request, –or– the Tx Lane should enter the L0s Link PM state.
- **After receiving a PM Request ACK, directs the PHY to the L1 state, and the PHY has not reached the L1 state and the PHY enters the Tx or Rx Recovery state** – The Port returns to an Idle state, then waits 10  $\mu$ s before making another L1 state entry Request, –or– the Tx Lane should enter the L0s Link PM state.
- **After the PHY transmits an EIOS and settles in the Tx L1 state, the Rx has not yet reached the L1 state, and an Rx Recovery state occurs or the Port's Link Control register Retrain Link bit (Downstream Ports, offset 78h[5]) is Set** – The Port directs the PHY Tx to exit the L1 state.

#### ***L1 Link PM State Entry Negotiation Interruption – Downstream Ports***

What occurs after L1 state entry negotiation is interrupted on the Downstream Ports is dependent upon the following:

- **Transmits PM\_request ACK DLLPs for an L1 state entry Request** – When the Tx or Rx *Recovery* state occurs, the Port returns to an Idle state, then waits 10  $\mu$ s before validating another L1 state entry Request from the Downstream device, –or– the Rx should enter the L0s Link PM state, to validate the next L1 state entry Request.
- **Waits for Electrical Idle from the other side of the PCI Express Link** – When the Tx or Rx *Recovery* state occurs, the Port returns to an Idle state, then waits 10  $\mu$ s before validating another L1 state entry Request from the Downstream device, –or– the Rx should enter the L0s Link PM state, to validate the next L1 state entry Request.
- **Rx is in the L1 state and directs the PHY to enter the L1 state** – If the PHY enters the Tx *Recovery* state because the Port's Link Control register Retrain Link bit (Downstream Ports, offset 78h[5]) is Set, the Port returns to an Idle state, then waits 10  $\mu$ s before validating another L1 state entry Request from the Downstream device, –or– the Rx should enter the L0s Link PM state, to validate the next L1 state entry Request.

### 11.3.3.8 L2/L3 Link PM State Entry

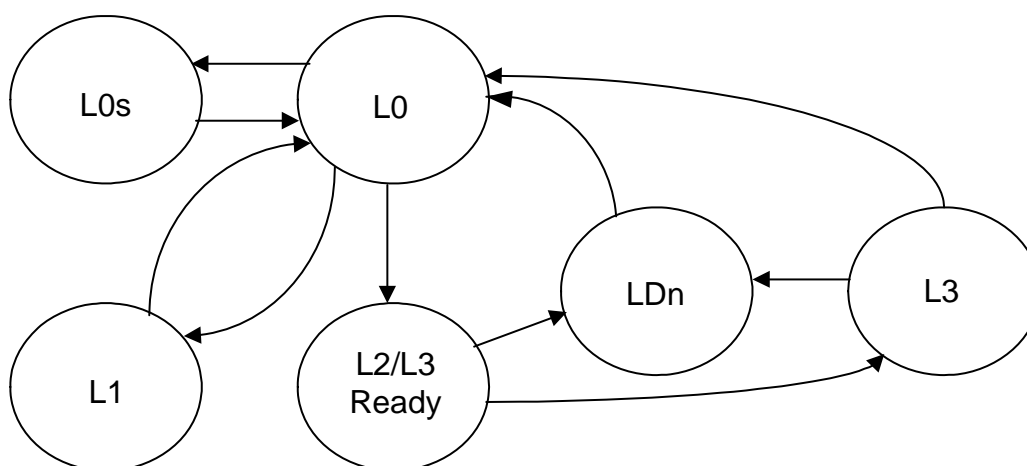
The L2 and L3 states are the staging points for removing main power. Support for the L2/L3 Ready transition protocol is required.

The L2 and L3 states are related to the PCI Express PCI-PM D-state transitions. L2/L3 Ready is the state in which a given Link enters when the platform is preparing to enter its system sleep state. Following completion of the L2/L3 Ready state transition protocol, the Link enters the L2/L3 Ready state. After main power is removed, the Link settles into the L3 state. The time for the L2/L3 Ready state entry transition is indicated by completion of the PME\_Turn\_Off/PME\_TO\_Ack handshake sequence. Any actions on the part of the Downstream component necessary to ready itself for power loss must be completed prior to initiating the transition to the L2/L3 Ready state. After all preparations for loss of power and clock are complete, the Downstream component initiates L2/L3 Ready state entry, by transmitting the PM\_EnterL23 DLLP Upstream.

Exit from the L2/L3 Ready state, back to the L0 Link PM state, can be initiated only by an Upstream-initiated transaction that targets the Downstream component (Upstream Electrical Idle exit) in the same manner that an Upstream-initiated transaction trigger would trigger the transition from the L1 to L0 Link PM states.

Figure 11-2 illustrates the Link PM state transitions.

**Figure 11-2. Link PM State Transitions**



**PME\_TO\_Ack TLP Flow from Downstream Device to Upstream Port(s)**

Because all previously transmitted packets must first be transmitted Upstream, before transmitting the PME\_TO\_Ack packet, the PEX 8748 performs the following tasks:

1. When a PME\_TO\_Ack is received from a Downstream device, it is routed to the Station's Upstream Egress module.
2. The Egress module decodes the PME\_TO\_Ack TLP, then informs the Upstream Port's PM module that it received the PME\_TO\_Ack.
3. When the Upstream Port receives the PME\_TO\_Ack Message from all enabled Downstream Ports, it transmits PME\_TO\_Ack Message toward the Root Complex. This Message flushes the previously transmitted PM\_PME Messages, due to PEX 8748 internal events.

**L2/L3 Ready Link PM State Entry Sequence – Upstream Port(s)**

System software places all Downstream devices into the PCI Express PCI-PM D3hot state, before broadcasting the PME\_Turn\_Off Message. The Root Complex then transmits the Message to the Upstream Port's PM module. The Upstream Port then unicasts the Message to all Downstream Ports, which reports an active Data Link Layer (DLL). Downstream Ports receive a PME\_Turn\_Off Message from Upstream, then transmit the Message to the Downstream device(s).

1. Upstream Port waits for a PME\_TO\_Ack Message, from all Downstream devices.
2. Waits for all pending TLP transmission to complete.
3. After all Downstream Ports receive the PME\_TO\_Ack, the PEX 8748 transmits the PME\_TO\_Ack Upstream.
4. PEX 8748 transmits the PM\_Enter\_L23\_DLLP Upstream, after all Downstream Ports are in the L2/L3 Ready state.
5. When in the L2/L3 Ready state, if a HP\_PWR\_GOOD\_x input is de-asserted (main power is removed), the Link settles into the L3 Link PM state.

**Note:** When in the L2/L3 Ready state, a Receiver exiting Electrical Idle also causes exit from the L2/L3 Ready state. The LTSSM enters the Link Down (LDn) state, then the PEX 8748 receives a Soft Reset, which also brings all Downstream Ports to the LDn state, from the L2/L3 Ready state.

After an Upstream Port is in the L2/L3 Ready state, Rx Electrical Idle exit causes the Link to transition to the LDn state, then subsequently enter the *Detect* state.

**L2/L3 Ready Link PM State Entry Sequence – Downstream Ports**

When a Downstream Port is going into the L2/L3 Ready state, the PEX 8748 performs the following tasks:

1. Downstream Port receives a PME\_Turn\_Off Message from an Upstream Port.
2. Downstream Port suspends the transmission of new PM\_PME Messages (upon global power-off).
3. Transmits PME\_Turn\_Off Message to the Downstream device after Clearing all pending TLPs.
4. Waits for a PM\_Enter\_L23\_DLLP from the Downstream device. When the PEX 8748 receives the PM\_Enter\_L23\_DLLP, the switch transmits a PM\_request\_ACK DLLP, then waits for the Rx to enter the L2/L3 Ready state. After the Rx goes into Electrical Idle, the Tx enters the L2/L3 Ready state. The Upstream scoreboard is updated though vendor-defined Messages.
5. When in the L2/L3 Ready state, if a HP\_PWR\_GOOD\_x input is de-asserted (main power is removed), the Link settles into the L3 Link PM state.

## 11.4 Power Management Tracking

**Note:** *NT Port Link Interface entry and exit to ASPM and Conventional PCI PM-compatible power states are not dependent upon the Transparent Upstream nor Downstream Port power states or traffic. They are solely dependent upon the NT Port Link Interface's traffic conditions.*

Upstream Port logic tracks the Link status of each Downstream and Upstream Port Link, to derive the following conditions:

- Upstream Port(s) enter(s) the L0s Link PM state when all enabled Downstream Receivers are in the L0s Link PM state or deeper, or in a Link Down state.
- Upstream Port(s) enter(s) the active L1 Link PM state, only when all Downstream Ports are in the active L1 Link PM state or deeper, or the Link is Down.
- When a Downstream Port is in the active L1 Link PM state and an ASPM L1 Link PM state exit is occurring in the Downstream Port, the Upstream Port(s) exit(s) the L1 Link PM state.
- When the Upstream Port(s) is (are) in the active L1 Link PM state and an active L1 Link PM state exit is occurring, due to Receiver Electrical Idle exit, the Downstream Port exits the L1 Link PM state.
- When a PME\_TO\_Ack Message is received only on all active (not in Link Down) Downstream Ports, a PME\_TO\_Ack Message is issued toward the Upstream Port(s). The NT Port Virtual Interface is marked as being in the *DL\_Down* state.
- When all Downstream Ports are in the L2/L3 Ready Link PM or Link Down state, the Upstream Port(s) transmit(s) PM\_ENTER\_L23 Data Link Layer Packets (DLLPs) toward the Root Complex.

## 11.5 Power Management Event Handler

PM\_PME Messages are Posted Transaction Layer Packets (TLPs) that inform the PM software which agent within the PCI Express hierarchy has requested a PM-state change. PM\_PME Messages are always routed toward the Root Complex.

PCI Express components are permitted to wake the system from any supported PM state, through a PME Request.

When a PEX 8748 Transparent Downstream Port is in the PCI Express PCI-PM D3hot state, the following Hot Plug and/or PCI Express Hot Plug events cause the Port's **PCI Power Management Status and Control** register *PME Status* bit (offset 44h[15]) to be Set:

- For Hot Plug-capable Ports:
  - **Presence Detect Changed** (logical OR of PRSNT# (HP\_PRSNT\_x# or Serial Hot Plug PRSNT# (SHP\_PRSNTx#) input), and SerDes Receiver Detect<sup>a</sup> on Lane(s) associated with that Port)
  - **Attention Button Pressed**
  - **Power Fault Detected**
  - **MRL Sensor Changed**
  - **Command Completed**
  - **Data Link Layer State Changed**
  - **Link Bandwidth Management Status**
  - **Link Autonomous Bandwidth Status**
- For non-Hot Plug-capable Downstream Ports:
  - **Presence Detect Changed** (SerDes Receiver Detect<sup>a</sup> on Lane(s) associated with that Port)
  - **Data Link Layer State Changed**

This causes the Downstream Port to generate a PM\_PME Message, if the Port's **PCI Power Management Status and Control** register *PME Enable* bit (offset 44h[8]) is Set.

*a. The SerDes Receiver Detect mechanism is comprised of the **Physical Layer Receiver Detect Status** register **Receiver Detected on Lane x** bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 200h[31:16]) or Serial Hot Plug PRSNT# (from external I<sup>2</sup>C I/O Expander) input for the Port.*

## 11.6 Power Management – Virtual Switch Mode

In Virtual Switch mode, the PEX 8748 can have up to six Upstream Ports. Downstream Ports are assigned to different Upstream Ports, depending upon the Virtual Switch Table. (Refer to [Section 5.3.3, “Virtual Switch Table.”](#)) Each virtual switch works as an independent switch for deciding entry into, or exit from, different PM states.

ASPM L0s and L1 Link PM state entry/exit works on the same rules defined in the previous sections. However, a particular virtual switch Upstream Port monitors the Link status only on the Downstream Ports that belong to that virtual switch, for ASPM L0s and L1 Link PM state entry. Similarly, the Downstream Ports monitor the corresponding virtual switch's Upstream Port Link state, for deciding entry/exit from the ASPM L0s and L1 Link PM states.

A virtual switch Upstream Port that is programmed into the D3hot state (Port's **PCI Power Management Status and Control** register *Power State* field bits, offset 44h[1:0], are both Set) requests the PCI L1 Link PM state, and finally settles into the PCI Link PM L1 state after the Upstream Port returns a Completion for this Configuration Write and L1 Link PM state negotiation successfully completes.

A PM\_Turn\_Off message received at a virtual switch Upstream Port is broadcast only to the Downstream Ports assigned to that virtual switch. When a PME\_TO\_Ack Message is subsequently received from all Downstream Ports assigned to the virtual switch, a single PME\_TO\_Ack message is sent Upstream of that virtual switch.

If a virtual switch has only an Upstream Port and no Downstream Ports assigned to it (*such as* the case of a Management Port), the virtual switch behaves like an endpoint for entry/exit to the ASPM L0s, ASPM L1, and PCI L1 and L2/L3 Ready Link PM states. Entry/exit to different Power Management states depends only upon the idle/traffic conditions on that Upstream Port.

When a new Downstream Port is added to a virtual switch due to re-configuration, future entry/exit into different power states on the virtual switch's Upstream Port also depends upon the newly added Downstream Port's power states. After a Downstream Port is removed from a virtual switch, that Port's Link states are not taken into consideration for future entry/exit into power states.

Interrupts and PM\_PME messages generated on a virtual switch's Downstream Ports are routed to its Upstream Port.

THIS PAGE INTENTIONALLY LEFT BLANK.



### 12.1 Multiple Virtual Switches

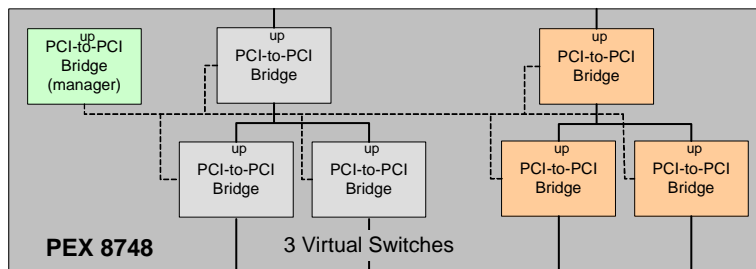
The PEX 8748 implements multiple virtual PCI Express switches, up to six total. Each virtual switch has its own Upstream Port and zero or more Downstream Ports. From a software point of view, each virtual switch consists of one Upstream PCI-to-PCI (P2P or P-P) bridge and zero to  $N$  Downstream PCI-to-PCI bridges. Each virtual switch has its own Reset, Interrupt, and Error signals. While the virtual switches share the same physical switch, traffic from one virtual switch cannot migrate to another virtual switch.

Figure 12-1 illustrates a partitioning of three virtual switches in a single physical switch – a Management Port hierarchy and two regular PCI Express 3-Port switches. In this example, the right three PCI-to-PCI bridges are owned by the right-most Upstream Port, and the middle three PCI-to-PCI bridges are owned by the middle Upstream Port. In this scenario, a third Upstream Port is used for a Management Port, and this Upstream Port has access to all other hierarchies, as indicated by the dashed line. The Management Port configures the virtual switches so that each Port – Upstream or Downstream – is placed in the proper configuration. Each Port can be assigned to only one virtual switch at a time. The Management Port can own Downstream Ports, although this mode is not shown in Figure 12-1.

Host-to-Host messages across virtual switch boundaries can be sent by the Management Port's **Scratchpad** and **Mailbox** registers.

Base mode can be emulated in Virtual Switch mode, by enabling two virtual switches, with all remaining Ports assigned to VS0.

**Figure 12-1. Multiple Virtual Switches in a Single Physical Switch**



## 12.1.1 Default Virtual Switch Port Assignments

**Note:** Default Port assignments must be changed, by using the serial EEPROM, I<sup>2</sup>C/SMBus, and/or the Management Port, prior to Setting the **Configuration Release** register *Initiate Configuration* bit (Port 0, accessible through the Management Port, offset 3ACh[0]).

The Management Port, if enabled (STRAP\_MGMT\_PORT\_EN=0 or Z) is assigned by the STRAP\_UPSTRM\_PORTSEL[2:0] 3-state inputs, or as programmed in the **Management Port Control** register (Port 0, accessible through the Management Port and Redundant Management Port, offset 354h). The value must correspond to a virtual switch Upstream Port.

Table 12-1 lists the default Port configuration according to the quantity of enabled virtual switches. The Port Numbers referenced correspond to those listed in Table 4-1, “Port Configurations.”

**Table 12-1. Virtual Switch Default Port Assignments – Virtual Switch Mode**

# of Enabled Virtual Switches	Test Modes	STRAP_TESTMODE[2:0] Input States	VS_MODE Input State	Virtual Switch Enable Register (Offset 358h) Value	VSx Upstream Register Offset and Value (Upstream Port Number) <sup>a</sup>	VSx Downstream Ports <sup>a</sup>	VSx Vector Register Offset
1	4	0ZZ	0	1h	VS0 (360h) = 0h (0)	1, 2, 3, 8, 9, 10, 11, 16, 17, 18, 19	VS0 (380h)
	5	0Z1					
	6	010					
	7	01Z					
2	4	0ZZ	Z	3h	VS0 (360h) = 0h (0) VS1 (364h) = 4h	VS0 = 1, 2, 3, 10, 11 VS1 = 8, 9, 16, 17, 18, 19	VS0 (380h) VS1 (384h)
	5	0Z1					
	6	010					
	7	01Z					
3	4	0ZZ	1	7h	VS0 (360h) = 0h (0) VS1 (364h) = 4h VS2 (368h) = Ah (10)	VS0 = 1, 2, 3, 16, 17, 18, 19 VS1 = 8, 9 VS2 = 11	VS0 (380h) VS1 (384h) VS2 (388h)
	5	0Z1					
	6	010					
	7	01Z					
4	8	011	0	Fh	VS0 (360h) = 0h (0) VS1 (364h) = 4h VS2 (368h) = Ah (10) VS3 (36Ch) = 10h (16)	VS0 = 1, 2, 3 VS1 = 8, 9 VS2 = 11 VS3 = 17, 18, 19	VS0 (380h) VS1 (384h) VS2 (388h) VS3 (38Ch)
	9	Z00					
	10	Z0Z					
	11	Z01					

**Table 12-1. Virtual Switch Default Port Assignments – Virtual Switch Mode (Cont.)**

# of Enabled Virtual Switches	Test Modes	STRAP_TESTMODE[2:0] Input States	VS_MODE Input State	Virtual Switch Enable Register (Offset 358h) Value	VSx Upstream Register Offset and Value (Upstream Port Number) <sup>a</sup>	VSx Downstream Ports <sup>a</sup>	VSx Vector Register Offset
5	8	011	Z	1Fh	VS0 (360h) = 0h (0) VS1 (364h) = 4h VS2 (368h) = Ah (10) VS3 (36Ch) = 10h (16) VS4 (370h) = 13h (19)	VS0 = 1, 2, 3 VS1 = 8, 9 VS2 = 11 VS3 = 17, 18 VS4 = None	VS0 (380h) VS1 (384h) VS2 (388h) VS3 (38Ch) VS4 (390h)
	9	Z00					
	10	Z0Z					
	11	Z01					
6	8	011	1	3Fh	VS0 (360h) = 0h (0) VS1 (364h) = 3h (3) VS2 (368h) = 8h (8) VS3 (36Ch) = Bh (11) VS4 (370h) = 10h (16) VS5 (374h) = 13h (19)	VS0 = 1, 2 VS1 = None VS2 = 9, 10 VS3 = None VS4 = 17, 18 VS5 = None	VS0 (380h) VS1 (384h) VS2 (388h) VS3 (38Ch) VS4 (390h) VS5 (394h)
	9	Z00					
	10	Z0Z					
	11	Z01					

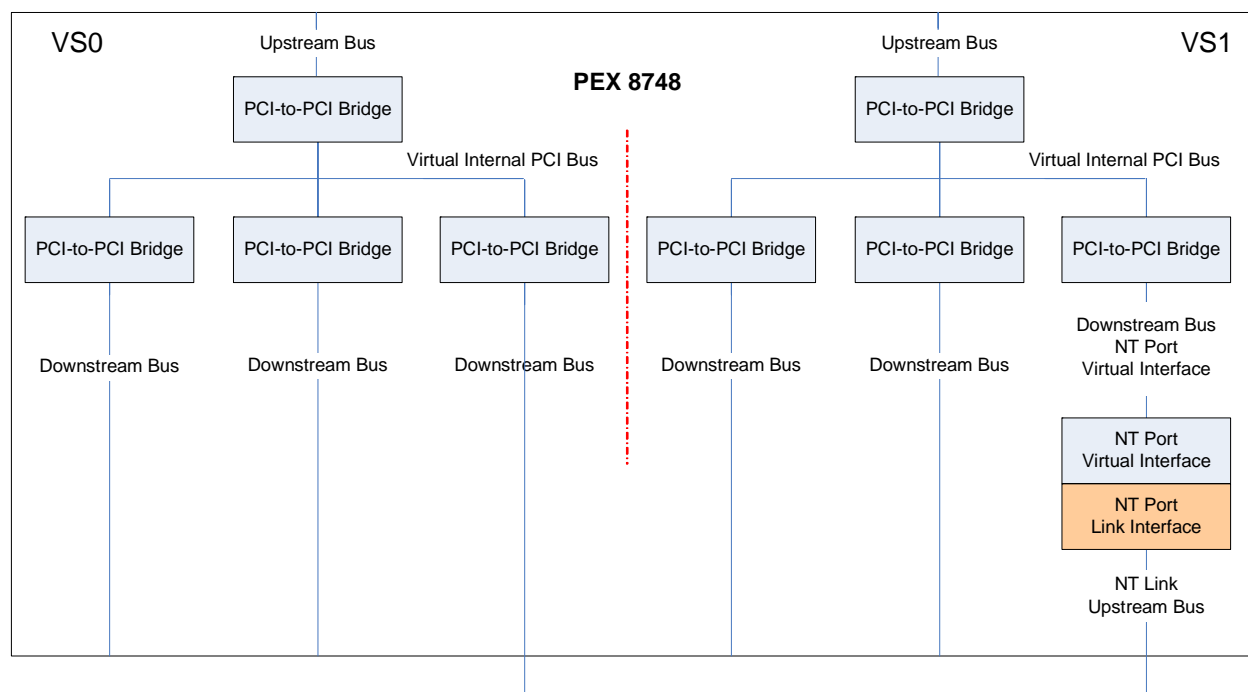
a. Table 4-1, “Port Configurations,” indicates which Ports are enabled/used, and when, based upon the STRAP\_STNx\_PORTCFGx input states and/or serial EEPROM programming.

## 12.1.2 Combined NT and Multiple Virtual Switches

The PEX 8748 supports the combination of a Non-Transparent bridge (NTB) and multiple virtual switches. The NT Port Virtual Interface is enumerated by only one of the virtual switches, as illustrated in Figure 12-2, for a sample configuration when using the VS0 and VS1 virtual switches.

In this example, a Downstream Port in VS0 can be connected to the VS1 NT Port Link Interface, so that the VS0 and VS1 Hosts can communicate with one another, while having their own independent hierarchies.

**Figure 12-2. Sample Configuration when NT Is Used in Combination with Multiple Virtual Switches**



## 12.2 Management Port

The PEX 8748 supports configuration and management through various options, described in the sections that follow. In Virtual Switch mode, the PEX 8748's registers are accessible by the designated Active Management Port, serial EEPROM, and/or I<sup>2</sup>C Slave interface. The PEX 8748 can be configured by serial EEPROM and/or Strapping balls, without restrictions. With the PEX 8748, any two PCI Express Ports can be configured as Management Ports, with one designated as the primary Management Port, and the other as the Redundant Management Port.

The Management Port can be used to:

- Configure virtual switches
- Move Ports from one virtual switch to another
- Monitor all virtual switch Links
- Configure PEX 8748-wide registers (*such as* the Physical Layer (PHY) registers)
- Access the serial EEPROM – In Virtual Switch mode, in-band access to the serial EEPROM is restricted to the Management Port, as designated by the **Management Port Control** register *Active Management Port* field (Port 0, accessible through the Management Port and Redundant Management Port, offset 354h[4:0])

The Redundant Management Port can be promoted to become the new Active Management Port, by software (through the Redundant Management Port, Management Port, and/or I<sup>2</sup>C/SMBus) copying the value of the register's *Redundant Management Port* field [12:8] to the *Active Management Port* field. (Refer to [Section 12.5.2](#) for further details.)

### 12.2.1 Out-of-Band Interfaces

This section briefly describes the out-of-band interfaces supported by the PEX 8748.

#### 12.2.1.1 Unused PCI Express Port – Management-Capable Port

Any Port can be designated as the Management Port and connected to a small service processor. This configuration is used in Virtual Switch mode.

#### 12.2.1.2 Strapping Balls

The Strapping balls are used to load default configurations. Refer to [Section 3.4.4, “Strapping Signals,”](#) for details.

#### 12.2.1.3 Serial EEPROM

An on-board serial EEPROM can be used to override the Strapping balls and configure the PEX 8748. Refer to [Chapter 6, “Serial EEPROM Controller,”](#) for details.

#### 12.2.1.4 I<sup>2</sup>C Bus/SMBus

The PEX 8748 supports both the I<sup>2</sup>C Bus and System Management Bus (SMBus) interfaces for configuring the registers. However, if the [STRAP\\_I2C\\_SMBUS\\_CFG\\_EN#](#) input is enabled to delay linkup until the **Configuration Release** register *Initiate Configuration* bit (Port 0, accessible through the Management Port, offset 3ACh[0]) is Set, this register Write can only be performed by I<sup>2</sup>C (not SMBus).

Refer to [Chapter 7, “I2C/SMBus Slave Interface Operation,”](#) for details.

## 12.2.2 In-Band Interface

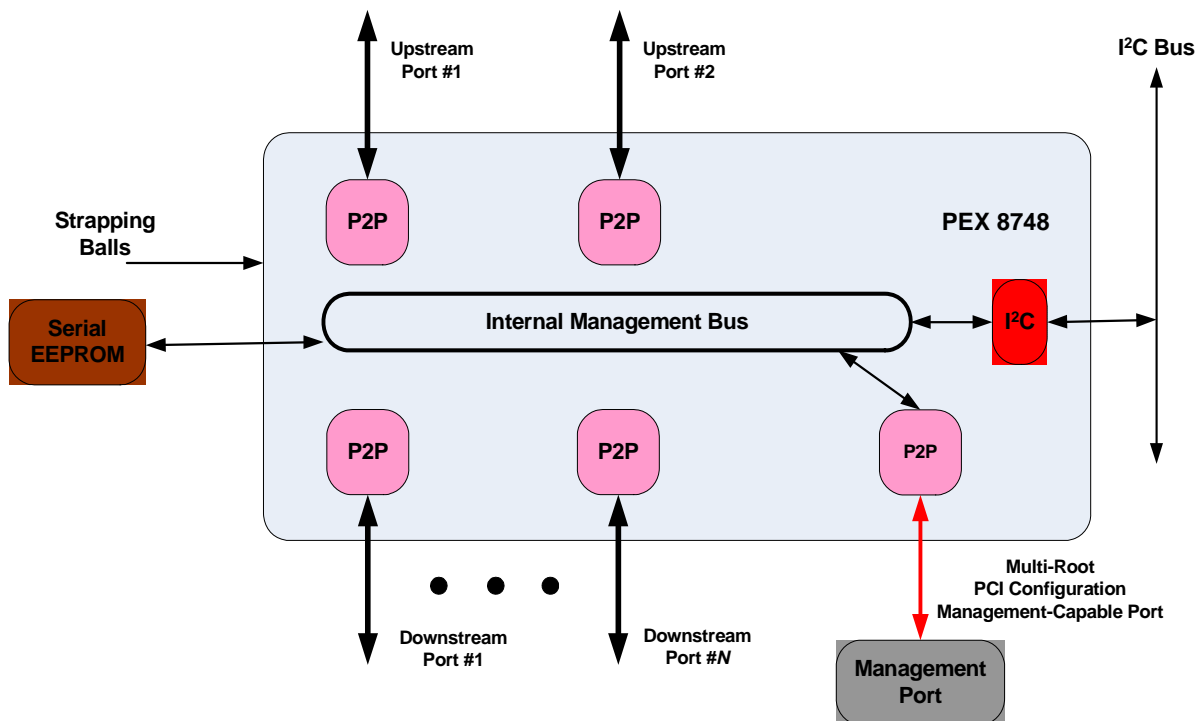
An in-band interface is typically used to manage PCI Express fabric within a single domain hierarchy, where the virtual switch's Upstream Port manages the PCI Express switch. In the case of Multi-Root switches, the shared resources are being modified by a trusted resource, *such as* a designated PCI Express Port. PLX Multi-Root switches support this mode of operation, in which one of the switch's Ports can be used for in-band operations, where the CPU can send data (regular traffic), as well as control data to control the PCI Express switch fabric's behavior. Additionally, a second Port can be designated as a Redundant Management Port. If the primary Port fails, the Redundant Management Port takes over the responsibility of managing the PCI Express switch fabric.

### 12.2.2.1 Configuration and Management

Figure 12-3 illustrates various options for accessing the Management Bus and internal registers in Virtual Switch mode, using two virtual switches. Each Upstream Port manages its own partition; however, another Port or interface is designated as the Management Port, with responsibility for defining and modifying partitions, as required. The PEX 8748 can also be configured by Strapping inputs, serial EEPROM, and/or I<sup>2</sup>C.

**Note:** The PCI-to-PCI (P2P) blocks in Figure 12-3 are a logical representation of how a Port presents itself to software.

**Figure 12-3. Configuration and Management – Virtual Switch Mode**



**Note:** Table 12-1, “Virtual Switch Default Port Assignments – Virtual Switch Mode,” lists the default virtual switch Port assignments. Table 4-1 indicates which Ports are enabled/used, and when, based upon the STRAP\_STNx\_PORTCFGx input states and/or serial EEPROM programming.

### 12.2.2.2 In-Band Management Port

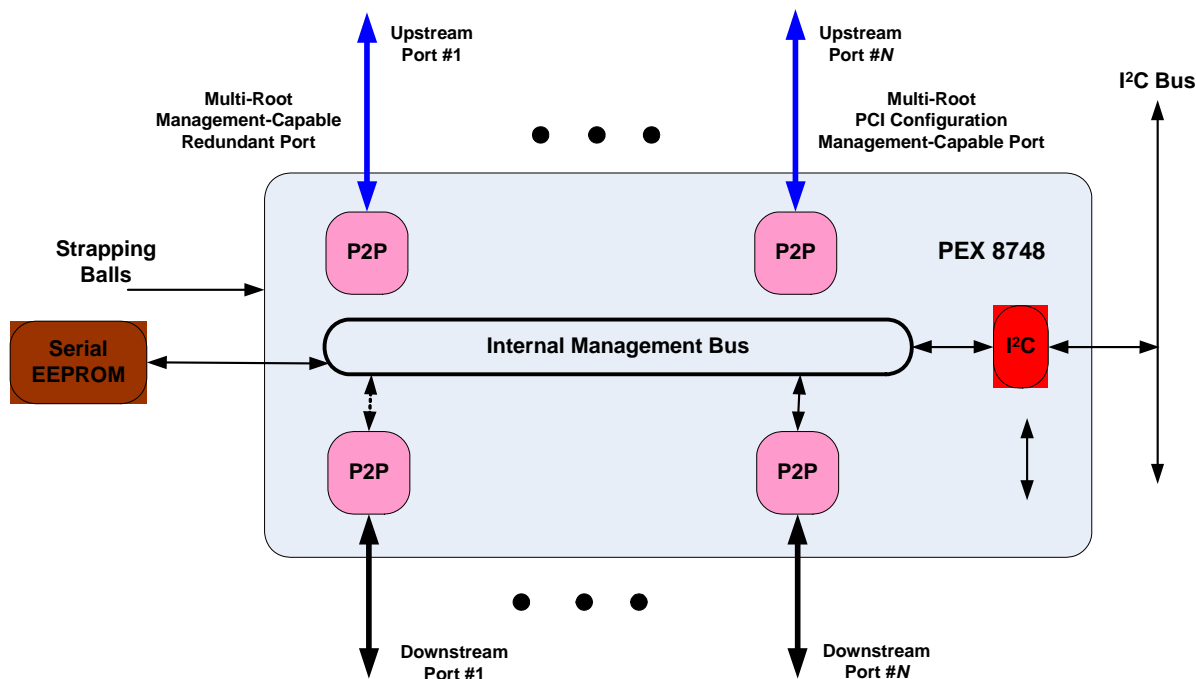
In Virtual Switch mode, an in-band PCI Express Port can be used as a Management Port and have access to all registers. (Refer to [Figure 12-4](#).)

The Redundant Management Port can access only the **Management Port Control** register (Port 0, accessible through the Management Port and Redundant Management Port, offset 354h), to promote itself to be the Management Port if the Active Management Port Host fails, using Memory-Mapped access to offset 354h within its own **BAR0/1** register map.

**Notes:** All Configuration Space registers (CSRs) requiring access by the Management Port are memory-mapped, using the Management Port's **Base Address 0** and **Base Address 1** registers (**BAR0** and **BAR1**, respectively), located in the Type 1 CSR headers. The In-Band Management Port uses Memory-Mapped transactions, to access all registers in all virtual switches. Because Ports within a Station can be assigned to different virtual switches, Management Port Writes to Station-specific registers can affect multiple virtual switches.

The PCI-to-PCI (P2P) blocks in [Figure 12-4](#) are a logical representation of how a Port presents itself to software.

**Figure 12-4. Configuration and Management of In-Band Management – Virtual Switch Mode**



**Note:** [Table 12-1](#), “Virtual Switch Default Port Assignments – Virtual Switch Mode,” lists the default virtual switch Port assignments. [Table 4-1](#) indicates which Ports are enabled/used, and when, based upon the `STRAP_STNx_PORTCFGx` input states and/or serial EEPROM programming.

### 12.2.2.3 Management Ports and Restriction

The shared registers listed in [Table 12-2](#), and all PCI-to-PCI registers, are accessed by the Active Management Port. The Redundant Management Port can access the **Management Port Control** register (Port 0, accessible through the Management Port and Redundant Management Port, offset [354h](#)), to promote itself to be the Management Port if the Active Management Port Host fails.

In Virtual Switch mode, the [STRAP\\_UPSTRM\\_PORTSEL\[2:0\]](#) 3-state inputs define which Port is the Active Management Port, when the Management Port is enabled ([STRAP\\_MGMT\\_PORT\\_EN](#)=0 or Z).

The Management Port can also be enabled and designated by serial EEPROM and/or I<sup>2</sup>C writing an appropriate value into the register's *Active Management Port Enable* and *Active Management Port* bits (bits [5 and 4:0], respectively). As mentioned previously, the Redundant Management Port can be promoted to become the new Active Management Port, by software (through the Redundant Management Port, Management Port, and/or I<sup>2</sup>C/SMBus) copying the register's *Redundant Management Port* field [12:8] to the *Active Management Port* field value.

**Note:** *In Virtual Switch mode, the Virtual Switch Table registers include the VSx Port Vector and VSx Upstream registers (Port 0, accessible through the Management Port, offsets [380h](#) through [394h](#), and [360h](#) through [374h](#), respectively). These two sets of registers define which Ports are associated to each virtual switch, and which Port is the Upstream Port for each Virtual Switch, respectively. These registers must be initialized by one (or more) of the following agents:*

- *Serial EEPROM.*
- *I<sup>2</sup>C/SMBus, provided that the STRAP\_I2C\_SMBUS\_CFG\_EN# input is pulled or tied Low (to delay linkup until I<sup>2</sup>C/SMBus Sets the **Configuration Release** register *Initiate Configuration* bit (Port 0, accessible through the Management Port, offset [3ACh\[0\]](#)). This option might require software support, to delay Host enumeration until I<sup>2</sup>C/SMBus Sets the *Initiate Configuration* bit after programming the PEX 8748 Configuration registers.*
- *Management Port software, provided that STRAP\_MGMT\_PORT\_EN is left floating (to enable the Management Port, and delay linkup of all other Ports until software and/or I<sup>2</sup>C/SMBus Sets the *Initiate Configuration* bit).*

Use of serial EEPROM for the initialization might be the best choice for most applications, because it is the simplest solution. Therefore, for Virtual Switch mode applications, the serial EEPROM is required to initialize the Virtual Switch Table registers, unless I<sup>2</sup>C/SMBus and/or Management Port software can perform this task.



**Table 12-2. Virtual Switch Management Registers (accessible from Primary Management Port)<sup>a</sup>**

Offset(s)	Register	Description <sup>b</sup>
354h	<b>Management Port Control</b>	This register contains bits that enable and indicate the Port Number of the Active and Redundant Management Ports.
358h[5:0]	<b>Virtual Switch Enable</b>	The register's <i>VSx Enable</i> bits are used to enable or disable virtual switches within the system. There is one bit, per virtual switch (VS0 through VS5). <sup>c</sup>
360h – 374h	<b>VSx Upstream</b>	These registers define the Upstream Port of each virtual switch. There is one register, per virtual switch (VS0 through VS5). <sup>c</sup>
380h – 394h	<b>VSx Port Vector</b>	These registers define the Upstream and Downstream Ports associated with each virtual switch. There is one register, per virtual switch (VS0 through VS5). Each register has one bit, per Port. <sup>c</sup>
3A0h	<b>Port Reset</b>	When driven with a value of 1, this register holds the Port in reset, including the Port PCI-to-PCI bridge and the hierarchy below it. A value of 0 indicates to un-reset the PCI-to-PCI bridge and the hierarchy below it. There is one bit, per Port.
900h	<b>Switch Link Up</b>	When the Port's Link state transitions from down to up, the Port's bit is Set. The Port's bit is Cleared, by using software to explicitly write 1 to the bit. The register has one bit, per Port.
904h	<b>Switch Link Down</b>	When the Port's Link state transitions from up to down, the Port's bit is Set. The Port's bit is Cleared, by using software to explicitly write 1 to the bit. The register has one bit, per Port.
908h	<b>Switch Link Event Mask</b>	If the <i>Mask</i> bit is Set, the corresponding <i>Up</i> or <i>Down</i> bit (located in register offsets 900h and 904h, respectively) transition does not generate an interrupt to the Management Port. If not masked, the bit transition generates an interrupt to the Active Management Port. The register has one <i>Mask</i> bit, per Port.
90Ch	<b>Switch Link Status</b>	This Read-Only register indicates Link status. The register has one <i>Status</i> bit, per Port.

- a. All registers listed in this table are located in Port 0, accessible through the Management Port.
- b. For more complete descriptions, refer to the individual registers listed.
- c. Additional information is also provided in [Table 5-4, “Virtual Switch Table Registers.”](#)

## 12.3 Virtual Switch Reset and Initialization

### 12.3.1 Virtual Switch Reset

For details regarding the Virtual Switch reset, refer to [Section 5.1.2, “Resets – Virtual Switch Mode.”](#)

### 12.3.2 Virtual Switch Initialization

For information on Virtual Switch Initialization, refer to [Section 5.3, “Initialization – Virtual Switch Mode.”](#)

### 12.3.3 Virtual Switch Table Programming Sequence

For details regarding the Virtual Switch Table Programming Sequence, refer to [Section 5.3.3.2, “Virtual Switch Table Programming Sequence.”](#)

## 12.4 Port Migration – Moving a Port from One Virtual Switch to Another (VS<sub>x</sub> to VS<sub>y</sub>)

A Downstream Port can be moved from one virtual switch (VS<sub>x</sub>) to another (VS<sub>y</sub>), by programming both virtual switch's **VS<sub>x</sub> Port Vector** registers (Port 0, accessible through the Management Port, offsets [380h](#) through [394h](#)). (The reason for the Port to be moved, and the coordination with the Management Port, are beyond the scope of this data book.) Prior to moving the Port, traffic to the target virtual switch must be stopped or blocked, by programming the Port's routing registers, such that no space exists throughout the Port.

First, the **VS<sub>x</sub> Port Vector** register for VS<sub>x</sub> must **remove** the Port. This can be done by Clearing the bit(s) in the active Port Vector for the Port(s) that is (are) moving **out** of VS<sub>x</sub>.

Second, the **VS<sub>x</sub> Port Vector** register for VS<sub>y</sub> must **add** the Port. This can be done by Setting the bit(s) in the active Port Vector for the Port(s) that just left VS<sub>x</sub>, that is (are) moving **to** VS<sub>y</sub>.

## 12.5 Failover in Virtual Switch Mode

This section discusses the various types of failover in Virtual Switch mode:

- [Virtual Switch Host Failover](#)
- [Active Management Port Failover](#)
- [Virtual Switch Host to NT Host Failover](#)

Another type of failover in NT mode, that is related to Virtual Switch mode, is described in [Section 14.9.3, “NT Failover for Virtual Switch Operation.”](#)

### 12.5.1 Virtual Switch Host Failover

A planned or unplanned event can cause a Host in one virtual switch to no longer be active. In either case, the first step is to make the Management Host aware that another Host is no longer available, and to quiesce the traffic in Ports owned by the no-longer-responding Host. After that, both planned or unplanned failover follow the same steps to complete the failover.

### 12.5.2 Active Management Port Failover

The Active Management Port has a backup (Redundant) Management Port. In the event that the Active Management Port fails, the Redundant Management Port can promote itself to be the Active Management Port while simultaneously demoting the previous Active Management Port to a backup status. The Redundant Management Port writes the [Management Port Control](#) register (Port 0, accessible through the Management Port and Redundant Management Port, offset [354h](#)) with the following sequence:

1. Initially, the **Management Port Control** register status is:
  - [Active Management Port](#) field [4:0] holds the Port Number of the Active Management Port
  - [Redundant Management Port](#) field [12:8] holds the Port Number of the Redundant Management Port
  - [Active Management Port Enable](#) and [Redundant Management Port Enable](#) bits (bits [5 and 13], respectively)
2. The Redundant Management Port determines that the Active Management Port is no longer active.
3. The Redundant Management Port writes to the **Management Port Control** register:
  - Its own Management Port Number, to the *Active Management Port* field
  - The Port Number of the previous Active Management Port, to the *Redundant Management Port* field
  - Holds the *Active Management Port Enable* and *Redundant Management Port Enable* bits Set
4. Failover is then complete. The new Active Management Port should scan the [VS Upstream to Management Upstream Doorbell Request](#) and [Management Upstream to VS Upstream Doorbell Request](#) registers (VS Upstream Port(s) and Management Port, offsets [910h](#) and [928h](#), respectively), to determine whether there are any active interrupts to service, and might inform other Hosts that a new Active Management Port has been promoted.

### 12.5.3 Virtual Switch Host to NT Host Failover

The NT Port can be held as a backup for one or more Hosts in different virtual switches. Suppose a Host within a virtual switch fails – the NT Host can replace it.

To do this, the Management Port must first move the NT Port to the failed virtual switch. (The reason is beyond the scope of this data book, but one reason, *for example*, might be because it stopped receiving heartbeats from an Upstream virtual switch.) This is done by writing the NT Port's Virtual Switch Port Vector Table, to remove it from the original virtual switch hierarchy, and then writing the VSx Port Vector Table, to add the NT Port to the new virtual switch.

After the NT Port is in the same hierarchy as the failed Upstream Root Port, the NT Port can perform a standard NT failover sequence. In brief, the NT Port upgrades itself to be the Upstream Port, it might reset the Downstream Ports, and takes over handling of Ports within that hierarchy.

## 12.6 Performance

The virtual switch is built upon a non-blocking, peer-to-peer switch. The virtual switch assignment merely restricts which Ports can communicate with one another. Performance for each virtual switch should be equivalent to an ideal, single-hierarchy switch.

There is one exception, with respect to the Data buffers. The Data buffers are set up on a Station-basis (16 neighboring Lanes). If two or more Ports within a Station belong to different virtual switches, the common pool shared by all Ports associated with the Station's Lanes can be unfairly used by one virtual switch, to the detriment of another. To prevent this from occurring, the initial credits and Port pool should be adjusted (Base mode – All Ports; Virtual Switch mode – Port 0, accessible through the Management Port, offset 9FCh), so as to reduce the common pool to zero (0).

## 12.7 Host-to-Host Communication

In Base mode, the PEX 8748 is usually managed by I<sup>2</sup>C or the Root Port. In contrast, for bladed systems, Root Complexes are not typically trusted entities, and are therefore managed by a dedicated Management Port connected through an out-of-band mechanism (*such as* I<sup>2</sup>C or a dedicated PCI Express Port).

A virtual switch sends Doorbell interrupts and **Scratchpad** register data to the Management Port, using either Configuration Requests, or Memory Requests to the offset within its own **BAR0/1** register map.

The Management Port sends Doorbell interrupts and **Scratchpad** register data to individual virtual switches, by Memory-Mapped access to the offset within the virtual switch Upstream Port registers (rather than the offset within the Management Port, which would generate an interrupt to itself). *For example*, if Port 1 is a non-Management virtual switch Upstream Port, the Management Port can generate a Doorbell interrupt to it, by writing to offset 1928h in the Management Port's **BAR0/1** register map.

Every virtual switch Upstream Port has the following **Scratchpad** registers and four corresponding **Doorbell** registers (located in the VS Upstream Port(s) and Management Port):

- Virtual switch to Management CPU direction
  - **VS Upstream to Management Upstream Doorbell Request** register (offset 910h)
  - **VS Upstream to Management Upstream Doorbell Mask** register (offset 914h)
  - **VS Upstream to Management Upstream Scratchpad 1** register (offset 918h)
  - **VS Upstream to Management Upstream Scratchpad 2** register (offset 91Ch)
  - **VS Upstream to Management Upstream Scratchpad 3** register (offset 920h)
  - **VS Upstream to Management Upstream Scratchpad 4** register (offset 924h)
- Management CPU to virtual switch direction
  - **Management Upstream to VS Upstream Doorbell Request** register (offset 928h)
  - **Management Upstream to VS Upstream Doorbell Mask** register (offset 92Ch)
  - **Management Upstream to VS Upstream Scratchpad 1** register (offset 930h)
  - **Management Upstream to VS Upstream Scratchpad 2** register (offset 934h)
  - **Management Upstream to VS Upstream Scratchpad 3** register (offset 938h)
  - **Management Upstream to VS Upstream Scratchpad 4** register (offset 93Ch)

Virtual switches cannot send messages directly to another virtual switch; instead, a virtual switch Upstream Port must first send the message to the Management Port, and the Management Port can then send the message to the other virtual switch.

Refer to [Section 9.6.2, “Doorbell Interrupts – Virtual Switch Mode,”](#) for further details.

NDA CONFIDENTIAL -- Inventec | King Michael

THIS PAGE INTENTIONALLY LEFT BLANK.



## Chapter 13 Transparent Port Registers

### 13.1 Introduction

This chapter defines the PEX 8748 Transparent Port registers. Each PEX 8748 Port has its own Configuration, Capability, Control, and Status register space. The register mapping is the same for each Port. (Refer to [Table 13-1](#).) This chapter also presents the PEX 8748 programmable registers and the order in which they appear in the register map. Register descriptions, when applicable, include details regarding their use and meaning in the Upstream Port(s) and Downstream Ports. (Refer to [Table 13-3](#).) Other registers are defined in:

- [Chapter 15, “NT Port Virtual Interface Registers – NT Mode”](#)
- [Chapter 16, “NT Port Link Interface Registers – NT Mode”](#)

All PEX 8748 registers can be accessed by Memory Requests. Registers outside the following Device-Specific ranges can also be accessed by Configuration Requests:

- 200h through AFCh
- B0Ch through B6Ch
- B80h through C30h

For reliable operation, software should not write to register offsets that are identified as **Reserved**.

For further details regarding register names and descriptions, refer to the following specifications:

- *PCI r3.0*
- *PCI Power Mgmt. r1.2*
- *PCI-to-PCI Bridge r1.2*
- *PCI Express Base r3.0*

## 13.2 Type 1 Port Register Map

Table 13-1 defines the Transparent mode Type 1 Port register mapping.

**Table 13-1. Type 1 Port Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		00h
PCI-Compatible Type 1 Configuration Header Registers (Offsets 00h – 3Ch)			...
			34h
			...
			3Ch
		Next Capability Pointer (48h)	Capability ID (01h)
PCI Power Management Capability Registers (Offsets 40h – 44h)			40h
			44h
		Next Capability Pointer (68h)	Capability ID (05h)
Message Signaled Interrupt Capability Registers (Offsets 48h – 64h)			48h
			...
			64h
		Next Capability Pointer (A4h)	Capability ID (10h)
PCI Express Capability Registers (Offsets 68h – A0h)			68h
			...
			A0h
		Next Capability Pointer (00h)	SSID/SSVID Capability ID (0Dh)
Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – FCh)			A4h
			...
			FCh
Next Capability Offset (FB4h)	1h	PCI Express Extended Capability ID (0003h)	100h
Device Serial Number Extended Capability Registers (Offsets 100h – 108h)			...
			108h
Next Capability Offset (148h)	1h	PCI Express Extended Capability ID (0019h)	10Ch
Secondary PCI Express Extended Capability Registers (Offsets 10Ch – 134h)			...
			134h
Next Capability Offset (10Ch)	1h	PCI Express Extended Capability ID (0004h)	138h
Power Budget Extended Capability Registers (Offsets 138h – 144h)			...
			144h
Next Capability Offset (E00h)	1h	PCI Express Extended Capability ID (0002h)	148h
Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)			...
			1BCh
Device-Specific Registers (Offsets 1C0h – A74h)			1C0h
			...
			A74h
<b>Reserved</b>			A78h – AFCh

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved



**Table 13-1. Type 1 Port Register Map (Cont.)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																		
Next Capability Offset (B70h)																1h																PCI Express Extended Capability ID (0018h)																B00h		
Latency Tolerance Reporting Extended Capability Registers (Offsets B00h – B08h)																																																...	B08h	
Reserved																																																B0Ch –	B6Ch	
Next Capability Offset 2 (000h)																1h																PCI Express Extended Capability ID 2 (000Bh)																B70h		
Device-Specific Registers (Offsets B70h – DFCh)																																																...	DFCh	
Next Capability Offset (B00h or F24h)																1h																PCI Express Extended Capability ID (0012h)																E00h		
Multicast Extended Capability Registers (Offsets E00h – E2Ch)																																																...	E2Ch	
Reserved																																																E30h –	EFCh	
Device-Specific Registers – Virtual Switch (Offsets F00h – F20h), Virtual Switch Mode																																																F00h	...	F20h
Next Capability Offset (B70h)																1h																PCI Express Extended Capability ID (000Dh)																F24h		
ACS Extended Capability Registers (Offsets F24h – F2Ch)																																																...	F2Ch	
Device-Specific Registers (Offsets F30h – FB0h)																																																F30h	...	FB0h
Next Capability Offset (138h)																1h																PCI Express Extended Capability ID (0001h)																FB4h		
Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)																																																...	FDCh	
Reserved																																																FE0h –	FFCh	

## 13.3 Port Register Configuration and Map

The PEX 8748 Port registers are configured similarly – not all the same. Port 0 of Station 0, Port 8 of Station 1, and Port 16 of Station 2 include more Device-Specific registers than the other Ports. Port 0 also contains registers that are used to set up and control the PEX 8748, as well as a serial EEPROM interface, I<sup>2</sup>C Slave interface, and SMBus Slave interface logic and control. The Port registers contain setup and control information specific to the Station and its Port(s). [Table 13-2](#) defines the Port register configuration and map.

**Note:** In Virtual Switch mode, Chip- and Station-specific registers are accessible only to the Management Port, serial EEPROM, I<sup>2</sup>C, and SMBus.

**Table 13-2. Port Register Configuration and Map**

Register Types	Station 0, Port 0	Station 1, Port 8 Station 2, Port 16	Station 0, Ports 1, 2, 3 Station 1, Ports 9, 10, 11 Station 2, Ports 17, 18, 19 <sup>a</sup>
PCI-Compatible Type 1 Configuration Header Registers (Offsets 00h – 3Ch)	00h – 0Ch, 18h – 3Ch	00h – 0Ch, 18h – 3Ch	00h – 0Ch, 18h – 3Ch
	Upstream Port(s) 10h, 14h		
PCI Power Management Capability Registers (Offsets 40h – 44h)	40h – 44h	40h – 44h	40h – 44h
Message Signaled Interrupt Capability Registers (Offsets 48h – 64h)	48h – 64h	48h – 64h	48h – 64h
PCI Express Capability Registers (Offsets 68h – A0h)	68h – 78h 84h – A0h	68h – 78h 84h – A0h	68h – 78h 84h – A0h
	Downstream Ports 7Ch, 80h		
Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – FCh)	A4h – FCh	A4h – FCh	A4h – FCh
Device Serial Number Extended Capability Registers (Offsets 100h – 108h)	100h – 108h	100h – 108h	100h – 108h
Secondary PCI Express Extended Capability Registers (Offsets 10Ch – 134h)	10Ch – 134h	10Ch – 134h	10Ch – 134h
Power Budget Extended Capability Registers (Offsets 138h – 144h)	Upstream Port(s) 138h – 144h		
Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)	148h – 1BCh	148h – 1BCh	148h – 174h
WRR Port Arbitration Table Registers (Offsets 178h – 1BCh)	178h – 1BCh	178h – 1BCh	
Device-Specific Registers (Offsets 1C0h – A74h)			
Device-Specific Registers – Read Pacing (Offsets 1D0h – 1D8h)	1D0h – 1D8h	1D0h – 1D8h	
Device-Specific Registers – Captured Bus and Device Numbers (Offsets 1DCh – 1E4h)	1DCh – 1E4h	1DCh – 1E0h	1DCh – 1E0h
Device-Specific Registers – Read Pacing (Offset 1E8h)	1E8h		

**Table 13-2. Port Register Configuration and Map (Cont.)**

Register Types	Station 0, Port 0	Station 1, Port 8 Station 2, Port 16	Station 0, Ports 1, 2, 3 Station 1, Ports 9, 10, 11 Station 2, Ports 17, 18, 19 <sup>a</sup>
Device-Specific Registers – Physical Layer (Offsets 200h – 25Ch)	200h – 25Ch	200h – 25Ch	
Device-Specific Registers – Serial EEPROM (Offsets 260h – 270h)	260h – 270h		
Device-Specific Registers – I2C and SMBus Slave Interfaces (Offsets 290h – 2FCh)	290h – 2FCh		
Device-Specific Registers – Port Configuration and Lane Status (Offsets 300h – 350h)	300h – 350h		
Device-Specific Registers – Port Configuration (Offsets 354h – 46Ch)	354h – 46Ch		
Device-Specific Registers – General-Purpose Input/Output (Offsets 600h – 68Ch)	600h – 68Ch		
Device-Specific Registers – PORT_GOOD Selector (Offsets 6A0h – 6A8h)	6A0h – 6A8h		
Device-Specific Registers – Error Checking and Debug (Offsets 700h – 75Ch)	700h – 75Ch	700h – 75Ch	724h, 728h
Device-Specific Registers – Control (Offsets 760h – 774h)	760h – 774h	760h – 774h	
Device-Specific Registers – Soft Error (Offsets 778h – 8FCh)	778h – 8FCh	778h – 8FCh	
Device-Specific Registers – Virtual Switch (Offsets 900h – 9E8h), Virtual Switch Mode	900h – 90Ch 940h – 9E8h		
	VS Upstream Port(s) and Management Port 910h – 93Ch		
Device-Specific Registers – Ingress Credit Handler (Offsets 9ECh – A2Ch)	9ECh – A2Ch	9ECh – A2Ch	9FCh – A2Ch (Base mode)
	VS Upstream Port(s) 9FCh – A2Ch (Virtual Switch mode)		
Device-Specific Registers – Virtual Switch Debug and GPIO Status and Control (Offsets A30h – A74h)	Upstream Port(s) A30h – A74h VS Upstream Port(s) A34h – A74h		
Latency Tolerance Reporting Extended Capability Registers (Offsets B00h – B08h)	Upstream Port(s) B00h – B08h		
Device-Specific Registers (Offsets B70h – DFCh)			
Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets B70h – B7Ch)	B70h – B7Ch	B70h – B7Ch	B70h – B7Ch
Device-Specific Registers – Physical Layer (Offsets B80h – C88h)	B80h – C88h	B80h – C88h	
Device-Specific Registers – Requester ID Read Back (Offsets C8Ch – C90h)	C8Ch – C90h	C8Ch – C90h	C8Ch – C90h
Multicast Extended Capability Registers (Offsets E00h – E2Ch)	E00h – E2Ch	E00h – E2Ch	E00h – E2Ch
Device-Specific Registers – Virtual Switch (Offsets F00h – F20h), Virtual Switch Mode	VS Upstream Port(s) F00h – F20h		
ACS Extended Capability Registers (Offsets F24h – F2Ch)	Downstream Ports F24h – F2Ch		

NDA CONFIDENTIAL -- Inventec | King Michael

**Table 13-2. Port Register Configuration and Map (Cont.)**

Register Types	Station 0, Port 0	Station 1, Port 8 Station 2, Port 16	Station 0, Ports 1, 2, 3 Station 1, Ports 9, 10, 11 Station 2, Ports 17, 18, 19 <sup>a</sup>
Device-Specific Registers (Offsets F30h – FB0h)			
Device-Specific Registers – Egress Control (Offsets F30h – F44h)	F30h – F44h	F30h – F44h	F30h – F44h
Device-Specific Registers – Ingress Control and Port Enable (Offsets F48h – F6Ch)	F48h, F5Ch (Base mode)	F48h, F5Ch (Base mode)	F48h, F5Ch (Base mode)
	Upstream Port(s) F4Ch – F58h, F60h VS Upstream Port(s) F48h, F5Ch (Virtual Switch mode)		
Device-Specific Registers – Error Checking and Debug (Offsets F70h – FB0h)	F70h – FB0h	F70h – FB0h	F70h – FB0h
Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)	FB4h – FDCh	FB4h – FDCh	FB4h – FDCh

a. Some Ports are used only within specific Port configurations. Refer to [Table 13-5](#) for details.

## 13.4 Register Access

Each PEX 8748 Port implements a 4-KB Configuration Space. The lower 256 bytes (offsets 00h through FFh) comprise the PCI-compatible Configuration Space, and the upper 960 DWords (offsets 100h through FFFh) comprise the PCI Express Extended Configuration Space. The PEX 8748 supports six mechanisms for accessing the Transparent Mode registers:

- [PCI r3.0-Compatible Configuration Mechanism](#)
- [PCI Express Enhanced Configuration Access Mechanism](#)
- [Device-Specific Memory-Mapped Configuration Mechanism](#)
- I<sup>2</sup>C Slave Interface (refer to [Section 7.2, “I<sup>2</sup>C Slave Interface”](#))
- SMBus Slave Interface (refer to [Section 7.3, “SMBus Slave Interface”](#))
- Serial Peripheral Interface (SPI) Bus (refer to [Chapter 6, “Serial EEPROM Controller”](#))

The sideband register access mechanisms (serial EEPROM, I<sup>2</sup>C, and/or SMBus) can modify Read-Only (RO) register values.

Each Port captures the Bus Number and Device Number on every Type 0 Configuration Write, as required by the *PCI r3.0*. Therefore, following a Fundamental Reset, software must initially perform a Configuration Write to each Port (using either the [PCI r3.0-Compatible Configuration Mechanism](#) or [PCI Express Enhanced Configuration Access Mechanism](#)), to allow each Port to capture its designated Bus Number and Device Number. The initial access to each Port, *for example*, could be a Configuration Write Request to a RO register, such as the **Device ID** / **Vendor ID** register (offset 00h).

**Note:** *An option is provided to allow the serial EEPROM and/or I<sup>2</sup>C/SMBus to initialize the Port’s [Captured Bus and Device Numbers](#) registers (offset 1DCh), instead of the required initial Configuration Write to each Port; however, this option is not recommended.*

### 13.4.1 PCI r3.0-Compatible Configuration Mechanism

The *PCI r3.0*-Compatible Configuration mechanism provides standard access to the PCI Express Configuration Space within each Port. PCI Express Configuration Request transactions use the following Addressing fields:

- *Bus Number*
- *Device Number*
- *Function Number*
- *Extended Register Number and Register Number*

**Notes:** *When an Alternative Routing-ID Interpretation (ARI) device is targeted, and the Downstream Port immediately above it is enabled for ARI Forwarding, the Device Number (in associated Routing IDs, Requester IDs, and Completer IDs) is implied to be 0, and the traditional 5-bit Device Number and 3-bit Function Number fields are interpreted as a single 8-bit Function Number field.*

*For ARI devices, discovery and enumeration of Extended Functions require ARI-aware software.*

Together, the *Extended Register Number* and *Register Number* fields specify the Configuration Space address of the register being accessed (concatenated such that the *Extended Register Number* forms the more-significant bits). For PCI-compatible Configuration Requests (that target the first 256 bytes of a Function's Configuration Space, the *Extended Register Address* field must be all zeros (0). The PCI Express Extended Configuration Space (offsets 100h through FFFh) can be accessed by the [PCI Express Enhanced Configuration Access Mechanism](#) (ECAM), except for the register ranges noted in [Section 13.4.2](#), or [Device-Specific Memory-Mapped Configuration Mechanism](#), which can access all registers.

The *PCI r3.0*-Compatible Configuration mechanism uses PCI Type 0 and Type 1 Configuration transactions to access the PEX 8748 Configuration registers. Each Port can convert a Type 1 Configuration Request (destined to a Downstream Port or device) to a Type 0 Configuration Request (targeting the next Downstream Port or device), as described below.

The PEX 8748 decodes all Type 1 Configuration accesses received on its Upstream Port(s), when any of the following conditions are met:

- If the Bus Number in the Configuration access is not within the Upstream Port(s') Secondary Bus Number and Subordinate Bus Number range, the Upstream Port(s) respond with an Unsupported Request (UR).
- Specified Bus Number in the Configuration access is the PEX 8748 internal virtual PCI Bus Number, the PEX 8748 automatically converts the Type 1 Configuration access into the appropriate Type 0 Configuration access for the specified device.
  - If the specified device corresponds to the PCI-to-PCI bridge in one of the PEX 8748 Downstream Ports, the PEX 8748 processes the Read or Write Request to the specified Downstream Port register specified in the original Type 1 Configuration access.
  - If the specified Device Number does not correspond to any of the PEX 8748 Downstream Port Device Numbers, the PEX 8748 responds with a UR.
- If the specified Bus Number in the Type 1 Configuration access is not the PEX 8748 internal virtual PCI Bus Number, but is the Bus Number of one of the PEX 8748 Downstream Port secondary/subordinate buses, the PEX 8748 passes the Configuration access on to the PCI Express Link attached to that PEX 8748 Downstream Port.
  - If the specified Bus Number is the Downstream Port Secondary Bus Number, and the specified Device Number is 0, the PEX 8748 converts the Type 1 Configuration access to a Type 0 Configuration access before passing it on.
  - If the specified Device Number is not 0, the Downstream Port drops the Transaction Layer Packet (TLP) and generates a UR.
- If the specified Bus Number is not the Downstream Port Secondary Bus Number, the PEX 8748 passes along the Type 1 Configuration access, without change.

## 13.4.2 PCI Express Enhanced Configuration Access Mechanism

The PCI Express Enhanced Configuration Access mechanism (ECAM) is implemented on all PCI Express PCs and on systems that do not implement a processor-specific firmware interface to the Configuration Space. The mechanism provides a Memory-Mapped Address space in the Root Complex, through which the Root Complex translates a Memory access into one or more Configuration Requests. Device drivers normally use an application programming interface (API) provided by the Operating System (OS), to use this mechanism.

The mechanism can be used to access PEX 8748 registers that are defined by Specifications. This mechanism **cannot**, however, access the Device-Specific registers (which are instead accessible by the Device-Specific Memory-Mapped Configuration Mechanism) in the following offset ranges, per Port:

- 200h through AFCh
- B0Ch through B6Ch
- B80h through C30h

Silicon Revision CA provides an option to enable ECAM access to all PEX 8748 registers, however, by Setting the **Station-Based Control** register *Device-Specific Register Access by Extended Configuration Request Enable* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 760h[16]), in the Station(s) that contain an Upstream or NT Port.

## 13.4.3 Device-Specific Memory-Mapped Configuration Mechanism

The Device-Specific Memory-Mapped Configuration mechanism provides a method to access the Configuration registers of all Ports, within a single 256-KB Memory map, as listed in Table 13-3. The registers of each Port are contained within a 4-KB range. The PEX 8748 supports a maximum of up to 12 simultaneously active Ports.

This mechanism follows the *PCI Express Base r3.0* Configuration Request Routing rules, which do not allow the propagation of Configuration Requests from Downstream-to-Upstream, nor peer-to-peer. By default, if any PEX 8748 Downstream Port receives a Memory Request from a Downstream device targeting the PEX 8748 Configuration registers, the Port:

- Responds to a Memory Read Request with a Completion indicating an Unsupported Request (UR) Uncorrectable error
- For Memory Writes:
  - Silently discards a Memory Write Request (default, in compliance with the *PCI Express Base r2.1*), –or–
  - If the **ECC Error Check Disable** register *Software Force Error Enable* bit (Transparent Downstream Ports, offset 720h[2]) is Set, the Port responds with a UR

In Memory Requests that target PEX 8748 registers, the Payload Length indicated within the Memory Request Header must be 1 DWord. Lengths greater than 1 DWord result in a Completer Abort error.

To use this mechanism, program the Upstream Port(s) Type 1 Configuration Space **Base Address 0** and **Base Address 1** registers (**BAR0** and **BAR1**, offsets 10h and 14h, respectively), which are typically enumerated at boot time by BIOS or the OS software. After the PEX 8748 Upstream Port(s) BARs are enumerated, Port 0 registers can be accessed with Memory Reads from and Writes to the first 4 KB (0000h to 0FFFh), Port 1 registers can be accessed with Memory Reads from and Writes to the second 4 KB (1000h to 1FFFh), and so forth. Within each of these 4-KB windows, individual registers are located at the DWord offsets indicated in Table 13-1.

In Virtual Switch mode, each virtual switch Upstream Port has its own BAR0/1 register for Memory Request access to all Port registers (excluding Chip- and Station-specific registers) within its Virtual Switch hierarchy. The Port mapping listed in [Table 13-3](#) applies to all virtual switches; however, only the Management Port mapping can access the Chip- and Station-specific registers, as well as the registers of all virtual switches. The 4-KB Address blocks corresponding to Ports that exist in a different virtual switch *cannot* be accessed with the **BAR0/1** mapping of an Upstream Port that is not the designated Management Port. Such attempted accesses are handled as No Operation (NOP) (Writes are ignored, Reads return a value of 0h).

**Table 13-3. Register Offsets from Upstream Port BAR0/1 Base Address**

Port Number	Internal Register 4-KB Memory Space Range	Location Range
Port 0	0_0000h to 0_0FFFh	0 to 4 KB
Port 1	0_1000h to 0_1FFFh	4 to 8 KB
Port 2	0_2000h to 0_2FFFh	8 to 12 KB
Port 3	0_3000h to 0_3FFFh	12 to 16 KB
Port 8	0_8000h to 0_8FFFh	32 to 36 KB
Port 9	0_9000h to 0_9FFFh	36 to 40 KB
Port 10	0_A000h to 0_AFFFh	40 to 44 KB
Port 11	0_B000h to 0_BFFFh	44 to 48 KB
Port 16	1_0000h to 1_0FFFh	64 to 68 KB
Port 17	1_1000h to 1_1FFFh	68 to 72 KB
Port 18	1_2000h to 1_2FFFh	72 to 76 KB
Port 19	1_3000h to 1_3FFFh	76 to 80 KB



## 13.5 Register Descriptions

The remainder of this chapter details the PEX 8748 registers, including:

- Bit/field names
- Description of register functions for the PEX 8748 Upstream Port(s), Downstream Ports, and virtual switches
- Type (*such as* RW or HwInit; refer to [Table 13-4](#) for Type descriptions)
- Whether the power-on/reset value can be modified, by way of the PEX 8748 serial EEPROM and/or I<sup>2</sup>C/SMBus Initialization feature
- Default power-on/reset value

**Table 13-4. Register Types, Grouped by User Accessibility**

Type	Description
HwInit	<b>Hardware-Initialized</b> Refers to the PEX 8748 Hardware-Initialization mechanism or PEX 8748 Serial EEPROM and/or I <sup>2</sup> C register Initialization features. RO after initialization and can only be reset with a Fundamental Reset. HwInit register bits are not modified by a Soft Reset.
RO	<b>Read-Only</b> Read-Only and cannot be altered by software. Permitted to be initialized by the PEX 8748 Hardware-Initialization mechanism or PEX 8748 serial EEPROM and/or I <sup>2</sup> C/SMBus register Initialization features.
ROS	<b>Read-Only, Sticky</b> Same as RO, except that bits are neither initialized nor modified by a Soft Reset.
RsvdP	<b>Reserved and Preserved</b> <i>Reserved</i> for future RW implementations. Registers are RO and must return a value of 0 when read. Software must preserve value read for Writes to bits.
RsvdZ	<b>Reserved and Zero</b> <i>Reserved</i> for future RW1C implementations. Registers are RO and must return a value of 0 when read. Software must use 0 for Writes to bits.
RW	<b>Read-Write</b> Read/Write and permitted to be Set or Cleared by software to the needed state.
RW1C	<b>Write 1 to Clear Status (Transparent mode)</b> Indicates status when read. A status bit Set by the system (to indicate status) is Cleared, by writing 1 to that bit. Writing 0 has no effect. <b>Read-Write, Clear Interrupt (NT mode)</b> In the NT registers, RW1C is used for the Doorbell interrupts, to indicate that a value of 1 Clears the interrupt.
RW1CS	<b>Write 1 to Clear, Sticky</b> Same as RW1C, except that bits are neither initialized nor modified by a Soft Reset.
RW1S	<b>Read-Write, Set Interrupt (NT mode)</b> In the NT registers, RW1S is used for the Doorbell interrupts, to indicate that a value of 1 Sets the interrupt.
RWS	<b>Read-Write, Sticky</b> Same as RW, except that bits are Set or Cleared by software to the needed state. Bits are neither initialized nor modified by a Soft Reset.
RZ	<b>Software Read Zero</b> Software Read always returns a value of 0; however, software is allowed to write this register.

## 13.6 Port Configurations and Station/Station Port/Port/SerDes Module/Lane Relationship

This section discusses the PEX 8748 Port configurations, in Base mode and Virtual Switch mode, as well as information related to Stations, Station Ports, Ports, SerDes modules, and Lanes.

In this chapter, the term “SerDes quad” or “quad” refers to assembling SerDes modules into groups of four contiguous Lanes for testing purposes.

### 13.6.1 Port Configurations

Table 13-5 defines the PEX 8748 Port configurations, as well as the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes. For further details regarding Port configuration, refer to Section 4.4.1.1, “Port Configurations.”

**Table 13-5. Port Configurations**

# <sup>a</sup>	Port Configuration Strapping Input States STRAP_STN0_PORTCFG[1:0] <sup>b</sup>	Port Configuration Register Value Port 0, Offset 300h[2:0]	Station 0 [Lanes/SerDes]/Port			
			Port 0 <sup>c</sup>	Port 1	Port 2	Port 3
1	0Z	001b	x16 [0-15]			
2	01	010b	x8 [0-7]	x8 [8-15]		
3	Z0	011b	x8 [0-7]	x4 [8-11]	x4 [12-15]	
4	ZZ	100b	x4 [0-3]	x4 [4-7]	x4 [8-11]	x4 [12-15]
# <sup>a</sup>	Port Configuration Strapping Input States STRAP_STN1_PORTCFG[1:0] <sup>b</sup>	Port Configuration Register Value Port 0, Offset 300h[5:3]	Station 1 [Lanes/SerDes]/Port			
			Port 8 <sup>c</sup>	Port 9	Port 10	Port 11
1	0Z	001b	x16 [16-31]			
2	01	010b	x8 [16-23]	x8 [24-31]		
3	Z0	011b	x8 [16-23]	x4 [24-27]	x4 [28-31]	
4	ZZ	100b	x4 [16-19]	x4 [20-23]	x4 [24-27]	x4 [28-31]

**Table 13-5. Port Configurations (Cont.)**

# <sup>a</sup>	Port Configuration Strapping Input States STRAP_STN2_PORTCFG[1:0] <sup>b</sup>	Port Configuration Register Value Port 0, Offset 300h[8:6]	Station 2 [Lanes/SerDes]/Port			
			Port 16 <sup>c</sup>	Port 17	Port 18	Port 19
1	0Z	001b	x16 [32-47]			
2	01	010b	x8 [32-39]	x8 [40-47]		
3	Z0	011b	x8 [32-39]	x4 [40-43]	x4 [44-47]	
4	ZZ	100b	x4 [32-35]	x4 [36-39]	x4 [40-43]	x4 [44-47]

- Port Configuration Number, which is the decimal equivalent of the **Port Configuration** register Port Configuration for Station *x* field(s)' (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6, 5:3, and/or 2:0]) binary value.
- “Z” means “floating.” Z is used for the 3-state inputs that have three functions, depending upon whether the input is pulled or tied High to **VDD18**, pulled or tied Low to **VSS** (Ground), or left floating (not connected). Each one of these three input states (1, 0, or Z, respectively) results in a different function for that input.
- Ports 0, 8, and 16 are also “Station Ports” within their respective Stations. Station Ports are used to control the Station-specific registers.

## 13.6.2 Virtual Switch Port Assignments – Virtual Switch Mode

**Note:** Default Port assignments must be changed, by using the serial EEPROM, I<sup>2</sup>C/SMBus, and/or the Management Port, prior to Setting the **Configuration Release** register **Initiate Configuration** bit (Port 0, accessible through the Management Port, offset **3ACh[0]**).

The Strapping balls define the quantity of enabled virtual switches (up to six, based upon the **STRAP\_TESTMODE[2:0]** and **VS\_MODE** input states), the Upstream and Downstream Ports, Port size, and Management Port. The Management Port and/or I<sup>2</sup>C/SMBus Slave interface can re-configure the virtual switches and dynamically re-assign Downstream Ports belonging to one virtual switch, to any other virtual switch, by programming the **VSx Port Vector** register(s) (Port 0, accessible through the Management Port, offset(s) **380h** through **394h**).

Each Port can be assigned to only one virtual switch; therefore, a Port must be removed from the Virtual Switch *x* Vector to which it is assigned, prior to adding that Port to a different Virtual Switch *x* Vector. Following dynamic re-assignment of a Downstream Port from one virtual switch to another virtual switch, that Port should be reset by Management Port software and/or the I<sup>2</sup>C/SMBus Slave interface by Setting, and then Clearing, the Port's **Port Reset** register **Reset Port *x* Vector** bit(s) (Port 0, accessible through the Management Port, offset **3A0h[19:16, 11:8, 3:0]**).

The Management Port, serial EEPROM, and/or I<sup>2</sup>C/SMBus Slave interface can configure any single Port within a virtual switch as that virtual switch's Upstream Port, by programming the **VSx Upstream** register(s) (Port 0, accessible through the Management Port, offset(s) **360h** through **374h**).

**Table 13-6** lists the default virtual switch Port assignment, according to the quantity of enabled virtual switches. The Port Numbers referenced correspond to those listed in **Table 13-5**.

For further details, refer to **Chapter 12, "Virtual Switch Mode."**

**Table 13-6. Virtual Switch Default Port Assignments – Virtual Switch Mode**

# of Enabled Virtual Switches	Test Modes	STRAP_TESTMODE[2:0] Input States	VS_MODE Input State	Virtual Switch Enable Register (Offset 358h) Value	VSx Upstream Register Offset and Value (Upstream Port Number) <sup>a</sup>	VSx Downstream Ports <sup>a</sup>	VSx Vector Register Offset
1	4	0ZZ	0	1h	VS0 (360h) = 0h (0)	1, 2, 3, 8, 9, 10, 11, 16, 17, 18, 19	VS0 (380h)
	5	0Z1					
	6	010					
	7	01Z					
2	4	0ZZ	Z	3h	VS0 (360h) = 0h (0) VS1 (364h) = 4h	VS0 = 1, 2, 3, 10, 11 VS1 = 8, 9, 16, 17, 18, 19	VS0 (380h) VS1 (384h)
	5	0Z1					
	6	010					
	7	01Z					
3	4	0ZZ	1	7h	VS0 (360h) = 0h (0) VS1 (364h) = 4h VS2 (368h) = Ah (10)	VS0 = 1, 2, 3, 16, 17, 18, 19 VS1 = 8, 9 VS2 = 11	VS0 (380h) VS1 (384h) VS2 (388h)
	5	0Z1					
	6	010					
	7	01Z					
4	8	011	0	Fh	VS0 (360h) = 0h (0) VS1 (364h) = 4h VS2 (368h) = Ah (10) VS3 (36Ch) = 10h (16)	VS0 = 1, 2, 3 VS1 = 8, 9 VS2 = 11 VS3 = 17, 18, 19	VS0 (380h) VS1 (384h) VS2 (388h) VS3 (38Ch)
	9	Z00					
	10	Z0Z					
	11	Z01					
5	8	011	Z	1Fh	VS0 (360h) = 0h VS1 (364h) = 4h VS2 (368h) = Ah (10) VS3 (36Ch) = 10h (16) VS4 (370h) = 13h (19)	VS0 = 1, 2, 3 VS1 = 8, 9 VS2 = 11 VS3 = 17, 18 VS4 = None	VS0 (380h) VS1 (384h) VS2 (388h) VS3 (38Ch) VS4 (390h)
	9	Z00					
	10	Z0Z					
	11	Z01					
6	8	011	1	3Fh	VS0 (360h) = 0h VS1 (364h) = 3h VS2 (368h) = 8h VS3 (36Ch) = Bh (11) VS4 (370h) = 10h (16) VS5 (374h) = 13h (19)	VS0 = 1, 2 VS1 = None VS2 = 9, 10 VS3 = None VS4 = 17, 18 VS5 = None	VS0 (380h) VS1 (384h) VS2 (388h) VS3 (38Ch) VS4 (390h) VS5 (394h)
	9	Z00					
	10	Z0Z					
	11	Z01					

a. *Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP\_STNx\_PORTCFGx input states and/or serial EEPROM programming.*

This section details the PCI-Compatible Type 1 Configuration Header registers. [Table 13-7](#) defines the register map.

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																								
Device ID																Vendor ID																00h								
PCI Status																PCI Command																04h								
PCI Class Code																PCI Revision ID																08h								
PCI BIST <i>(Not Supported)</i>				PCI Header Type												Master Latency Timer <i>(Not Supported)</i>								Cache Line Size								0Ch								
Base Address 0																																10h								
Base Address 1																																14h								
Secondary Latency Timer <i>(Not Supported)</i>								Subordinate Bus Number												Secondary Bus Number												Primary Bus Number								18h
Secondary Status				<i>Not Supported/Reserved</i>												I/O Limit								I/O Base								1Ch								
Memory Limit																Memory Base																20h								
Prefetchable Memory Limit																Prefetchable Memory Base																24h								
Prefetchable Memory Upper Base Address																																28h								
Prefetchable Memory Upper Limit Address																																2Ch								
I/O Limit Upper 16 Bits																I/O Base Upper 16 Bits																30h								
<i>Reserved</i>																								Capability Pointer (40h)								34h								
Expansion ROM Base Address <i>(Reserved)</i>																																38h								
<i>Not Supported/Reserved</i>				Bridge Control												PCI Interrupt Pin								PCI Interrupt Line								3Ch								

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Vendor ID</b> Identifies the device manufacturer. Defaults to the PCI-SIG-issued Vendor ID of PLX (10B5h), if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	10B5h
31:16	<b>Device ID</b> Identifies the particular device. Defaults to the PLX part number for the PEX 8748, if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	8748h

**Register 13-2. 04h PCI Command/Status  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Command</b>				
0	<b>I/O Access Enable</b> 0 = PEX 8748 ignores I/O Space accesses on the Port's primary interface 1 = PEX 8748 responds to I/O Space accesses on the Port's primary interface	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8748 ignores Memory Space accesses on the Port's primary interface 1 = PEX 8748 responds to Memory Space accesses on the Port's primary interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8748 Memory and I/O Request forwarding Upstream. Does not affect Message (including INTx Interrupt Messages) forwarding nor Completions traveling Upstream or Downstream. 0 = PEX 8748 handles Memory and I/O Requests received on the Port's Downstream/secondary interface as Unsupported Requests (URs); for Non-Posted Requests, the PEX 8748 returns a Completion with UR Completion status. Because MSI Messages are in-band Memory Writes, disables MSI Messages as well. 1 = PEX 8748 forwards Memory and I/O Requests Upstream.	RW	Yes	0
3	<b>Special Cycle Enable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
4	<b>Memory Write and Invalidate Enable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
5	<b>VGA Palette Snoop</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
6	<b>Parity Error Response Enable</b> Controls bit 24 ( <i>Master Data Parity Error Detected</i> ).	RW	Yes	0
7	<b>IDSEL Stepping/Wait Cycle Control</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0

**Register 13-2. 04h PCI Command/Status  
(All Ports) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>SERR# Enable</b> Controls bit 30 ( <i>Signaled System Error</i> ).  1 = Enables reporting of Fatal and Non-Fatal errors detected by the device to the Root Complex, and, enables primary interface forwarding of ERR_FATAL and ERR_NONFATAL Messages from Downstream Ports and devices when the Port's <b>Bridge Control</b> register <i>SERR# Enable</i> bit (offset 3Ch[17]) is Set	RW	Yes	0
9	<b>Fast Back-to-Back Transactions Enable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
10	<b>Interrupt Disable</b> 0 = Port is enabled to generate INT <sub>x</sub> Interrupt Messages and assert <i>PEX_INTA#</i> (Base mode) or VS <sub>x</sub> _PEX_INTA# (Virtual Switch mode) output 1 = Port is prevented from generating INT <sub>x</sub> Interrupt Messages and asserting PEX_INTA# (Base mode) or VS <sub>x</sub> _PEX_INTA# (Virtual Switch mode) output	RW	Yes	0
15:11	<i>Reserved</i>	RsvdP	No	0-0h
<b>PCI Status</b>				
18:16	<i>Reserved</i>	RsvdP	No	000b
19	<b>Interrupt Status</b> 0 = No INT <sub>x</sub> Interrupt Message is pending 1 = INT <sub>x</sub> Interrupt Message is pending internally to the Port, –or– <i>PEX_INTA#</i> (Base mode) or VS <sub>x</sub> _PEX_INTA# (Virtual Switch mode) (if enabled) is asserted	RO	No	0
20	<b>Capability List</b> Capability function is supported. Set, as required by the <i>PCI Express Base r3.0</i> .	RO	Yes	1
21	<b>66 MHz Capable</b> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
22	<i>Reserved</i>	RsvdP	No	0
23	<b>Fast Back-to-Back Transactions Capable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0



**Register 13-2. 04h PCI Command/Status**  
**(All Ports) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Master Data Parity Error Detected</b> If bit 6 ( <i>Parity Error Response Enable</i> ) is Set, the Port Sets this bit when the Port: <ul style="list-style-type: none"> <li>Forwards the poisoned TLP Write Request from the secondary to the primary interface, –or–</li> <li>Receives a Completion marked as poisoned on the primary interface</li> </ul> If the <i>Parity Error Response Enable</i> bit is Cleared, the PEX 8748 never Sets this bit. This error is natively reported by the <b>Uncorrectable Error Status</b> register <i>Poisoned TLP Status</i> bit (offset FB8h[12]), which is mapped to this bit for Conventional PCI backward compatibility.	RW1C	Yes	0
26:25	<b>DEVSEL# Timing</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	00b
27	<b>Signaled Target Abort</b> The Upstream Port(s) Set this bit when any one of the following conditions are met: <ul style="list-style-type: none"> <li>Upstream Port receives a Memory Request targeting a PEX 8748 register, and the Payload Length (indicated within the Memory Request Header) is greater than 1 DWord</li> <li>Upstream Port receives a Memory Request targeting a PEX 8748 register address within a non-existent Port</li> <li>Transparent Downstream Port Sets this bit if it detects an Access Control Services (ACS) violation</li> </ul> This error is reported by the <b>Uncorrectable Error Status</b> register <i>Completer Abort Status</i> bit (offset FB8h[15]), which is mapped to this bit for Conventional PCI backward compatibility.	RW1C	Yes	0
28	<b>Received Target Abort</b> <i>Reserved</i>	RsvdP	No	0
29	<b>Received Master Abort</b> <i>Reserved</i>	RsvdP	No	0
30	<b>Signaled System Error</b> If bit 8 ( <i>SERR# Enable</i> ) is Set, the Port Sets this bit when it transmits or forwards an ERR_FATAL or ERR_NONFATAL Message Upstream. This error is natively reported by the <b>Device Status</b> register <i>Fatal Error Detected</i> and <i>Non-Fatal Error Detected</i> bits (offset 70h[18:17], respectively), which are mapped to this bit for Conventional PCI backward compatibility.	RW1C	Yes	0
31	<b>Detected Parity Error</b> This error is natively reported by the <b>Uncorrectable Error Status</b> register <i>Poisoned TLP Status</i> bit (offset FB8h[12]), which is mapped to this bit for Conventional PCI backward compatibility. 1 = Port received a Poisoned TLP on its primary side, regardless of the bit 6 ( <i>Parity Error Response Enable</i> ) state	RW1C	Yes	0

**Register 13-3. 08h PCI Class Code and Revision ID  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Revision ID</b>				
7:0	<b>Revision ID</b> Unless overwritten by the serial EEPROM, returns the Silicon Revision (BAh), the PLX-assigned Revision ID for this version of the PEX 8748. The PEX 8748 Serial EEPROM register Initialization capability is used to replace the PLX Revision ID with another Revision ID.	RO	Yes	BAh
<b>PCI Class Code</b>				<b>060400h</b>
15:8	<b>Register-Level Programming Interface</b> The PEX 8748 Ports support the <i>PCI-to-PCI Bridge r1.2</i> requirements, but not subtractive decoding, on their Upstream interface.	RO	Yes	00h
23:16	<b>Sub-Class Code</b> PCI-to-PCI bridge.	RO	Yes	04h
31:24	<b>Base Class Code</b> Bridge device.	RO	Yes	06h

**Register 13-4. 0Ch Miscellaneous Control  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Cache Line Size</b>				
7:0	<b>Cache Line Size</b> System Cache Line Size. Implemented as a RW field for Conventional PCI compatibility purposes and does not impact PEX 8748 functionality.	RW	Yes	00h
<b>Master Latency Timer</b>				
15:8	<b>Master Latency Timer</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	00h
<b>PCI Header Type</b>				
22:16	<b>Configuration Layout Type</b> The Port's Configuration Space Header adheres to the Type 1 PCI-to-PCI Bridge Configuration Space layout defined by the <i>PCI-to-PCI Bridge r1.2</i> .	RO	No	01h
23	<b>Multi-Function Device</b> 0 = Single-function device 1 = Indicates multiple (up to eight) functions (logical devices), each containing its own, individually addressable Configuration Space, 256 DWords in size	RO	No	0
<b>PCI BIST</b>				
31:24	<b>PCI BIST</b> <i>Not supported</i> Built-In Self-Test (BIST) Pass or Fail.	RsvdP	No	00h

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

**Register 13-5. 10h Base Address 0  
(Upstream Port(s))**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Base Address register maps the PEX 8748 Configuration registers into Memory space <i>Note: The Upstream Port(s) are hardwired to 0.</i>	Upstream	RO	No	0
	<i>Reserved</i>	Downstream	RsvdP	No	0
2:1	<b>Memory Map Type</b> 00b = Base Address register is 32 bits wide and can be mapped anywhere in the 32-bit Memory space 10b = Base Address register is 64 bits wide and can be mapped anywhere in the 64-bit Address space All other encodings are <i>Reserved</i> .	Upstream	RO	Yes	00b
	<i>Reserved</i>	Downstream	RsvdP	No	00b
3	<b>Prefetchable</b> 0 = Base Address register maps the PEX 8748 Configuration registers into Non-Prefetchable Memory space <i>Note: The Upstream Port(s) are hardwired to 0.</i>	Upstream	RO	Yes	0
	<i>Reserved</i>	Downstream	RsvdP	No	0
17:4	<i>Reserved</i>		RsvdP	No	0-0h
31:18	<b>Base Address 0</b> Base Address ( <b>BAR0</b> ) for the Device-Specific Memory-Mapped Configuration mechanism.	Upstream	RW	Yes	0-0h
	<i>Reserved</i>	Downstream	RsvdP	No	0-0h

**Register 13-6. 14h Base Address 1  
(Upstream Port(s))**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Base Address 1</b> For 64-bit addressing, Base Address 1 ( <b>BAR1</b> ) extends Base Address 0 ( <b>BAR0</b> ) to provide the upper 32 Address bits when the <b>Base Address 0</b> register <i>Memory Map Type</i> field (Upstream Port(s), offset 10h[2:1]) is programmed to 10b.	Upstream	RW	Yes	0000_0000h
	RO when the <b>Base Address 0</b> register is not enabled as a 64-bit BAR ( <i>Memory Map Type</i> field (Upstream Port(s), offset 10h[2:1]) is not programmed to 10b).		RO	Yes	0000_0000h
	<i>Reserved</i>	Downstream	RsvdP	No	0000_0000h

**Register 13-7. 18h Bus Number  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Primary Bus Number</b> Primary Bus Number of this PCI-to-PCI bridge. Records the Bus Number of the PCI Bus segment to which the primary interface of this Port is connected. Programmed by Configuration software.	RW	Yes	00h
15:8	<b>Secondary Bus Number</b> Secondary Bus Number of this PCI-to-PCI bridge. Records the Bus Number of the PCI Bus segment that is the secondary interface of this Port. Programmed by Configuration software.	RW	Yes	00h
23:16	<b>Subordinate Bus Number</b> Subordinate Bus Number of this PCI-to-PCI bridge. Records the Bus Number of the highest numbered PCI Bus segment that is subordinate to this Port. Programmed by Configuration software.	RW	Yes	00h
31:24	<b>Secondary Latency Timer</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	00h

**Register 13-8. 1Ch Secondary Status, I/O Limit, and I/O Base (All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Note:</b> If I/O Address decoding is enabled (<b>PCI Command</b> register <b>I/O Access Enable</b> bit, offset 04h[0], is Set), the Port forwards I/O transactions from the Port's primary interface to the Port's secondary interface (Downstream) if an I/O address is within the range defined by the <b>I/O Base</b> and <b>I/O Limit</b> registers when the Base is less than or equal to the Limit.</p> <p>Conversely, the Port forwards I/O transactions from the Port's secondary interface to the Port's primary interface (Upstream) if an I/O address is outside this Address range. If the Port does not implement an I/O Address range, the Port forwards all I/O transactions on the Port's secondary interface Upstream, to the Port's primary interface.</p>				
<b>I/O Base</b>				
3:0	<b>I/O Base Addressing Capability</b> 1h = 32-bit I/O Address decoding is supported All other encodings are <b>Reserved</b> .	RO	Yes	1h
7:4	<b>I/O_BAR[15:12]</b> I/O Base Address[15:12]. The Ports use their <b>I/O Base</b> and <b>I/O Limit</b> registers to determine the address range of I/O transactions to forward from one interface to the other. I/O Base Address[15:12] bits specify the Port's I/O Base Address[15:12]. The PEX 8748 assumes I/O Base Address[11:0]=000h. For 16-bit I/O addressing, the PEX 8748 assumes Address[31:16]=0000h. For 32-bit addressing, the PEX 8748 decodes Address[31:0], and uses the <b>I/O Upper Base and Limit Address</b> register <b>I/O Base Upper 16 Bits</b> and <b>I/O Limit Upper 16 Bits</b> fields (offset 30h[15:0 and 31:16], respectively).	RW	Yes	Fh
<b>I/O Limit</b>				
11:8	<b>I/O Limit Addressing Capability</b> 1h = 32-bit I/O Address decoding is supported All other encodings are <b>Reserved</b> .	RO	Yes	1h
15:12	<b>I/O_Limit[15:12]</b> I/O Limit Address[15:12]. The Ports use their <b>I/O Base</b> and <b>I/O Limit</b> registers to determine the Address range of I/O transactions to forward from one interface to the other. I/O Limit Address[15:12] bits specify the Port's I/O Limit Address[15:12]. The PEX 8748 assumes I/O Limit Address[11:0]=FFFh. For 16-bit I/O addressing, the PEX 8748 decodes Address bits [15:0] and assumes I/O Limit Address[31:16]=0000h. For 32-bit addressing, the PEX 8748 decodes Address bits [31:0], and uses the <b>I/O Upper Base and Limit Address</b> register <b>I/O Base Upper 16 Bits</b> and <b>I/O Limit Upper 16 Bits</b> fields (offset 30h[15:0 and 31:16], respectively).	RW	Yes	0h

**Register 13-8. 1Ch Secondary Status, I/O Limit, and I/O Base  
(All Ports) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Secondary Status</b>				
20:16	<i>Reserved</i>	RsvdP	No	0-0h
21	<b>66 MHz Capable</b> <i>Not supported</i> 0 = Not enabled, because PCI Express does <i>not support</i> 66 MHz	RsvdP	No	0
22	<i>Reserved</i>	RsvdP	No	0
23	<b>Fast Back-to-Back Transactions Capable</b> <i>Reserved</i> Not enabled, because PCI Express does <i>not support</i> this function.	RsvdP	No	0
24	<b>Master Data Parity Error</b> If the <b>Bridge Control</b> register <i>Parity Error Response Enable</i> bit (offset 3Ch[16]) is Set, the Port Sets this bit when transmitting or receiving a TLP on its Downstream side, and when either of the following two conditions occur: <ul style="list-style-type: none"> <li>Port receives Completion marked poisoned</li> <li>Port forwards poisoned TLP Write Request</li> </ul> If the <i>Parity Error Response Enable</i> bit is Cleared, the PEX 8748 never Sets this bit. These errors are reported by the Port's <b>Uncorrectable Error Status</b> register <i>Poisoned TLP Status</i> bit (offset FB8h[12]), and mirrored to this bit for Conventional PCI backward compatibility.	RW1C	Yes	0
26:25	<b>DEVSEL# Timing</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	00b
27	<b>Signaled Target Abort</b> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
28	<b>Received Target Abort</b> Cleared, as required by the <i>PCI Express Base r3.0</i> , because the PEX 8748 never initiates a Request itself.	RsvdP	No	0
29	<b>Received Master Abort</b> Cleared, as required by the <i>PCI Express Base r3.0</i> , because the PEX 8748 never initiates a Request itself.	RsvdP	No	0
30	<b>Received System Error</b> 1 = Downstream Port received an ERR_FATAL or ERR_NONFATAL Message on its secondary interface from a Downstream device	RW1C	Yes	0
31	<b>Detected Parity Error</b> 1 = Downstream Port received a poisoned TLP from a Downstream device (Set regardless of the <b>Bridge Control</b> register <i>Parity Error Response Enable</i> bit (offset 3Ch[16]) state)	RW1C	Yes	0

**Register 13-9. 20h Memory Base and Limit  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Note:</b> The Port forwards Memory transactions from its primary interface to its secondary interface (Downstream) if a Memory address is within the range defined by the <b>Memory Base</b> and <b>Memory Limit</b> registers (when the Base is less than or equal to the Limit).</p> <p>Conversely, the Port forwards Memory transactions from its secondary interface to its primary interface (Upstream) if a Memory address is outside this Address range (provided that the address is not within the range defined by the <b>Prefetchable Memory Base</b> (offsets 28h + 24h[15:0]) and <b>Prefetchable Memory Limit</b> (offsets 2Ch + 24h[31:16]) registers).</p>				
<b>Memory Base</b>				
3:0	<b>Reserved</b>	RsvdP	No	0h
15:4	<b>MEM_BAR[31:20]</b> Memory Base Address[31:20]. Specifies the Port's Non-Prefetchable Memory Base Address[31:20]. The PEX 8748 assumes Memory Base Address[19:0]=0_0000h.	RW	Yes	FFFh
<b>Memory Limit</b>				
19:16	<b>Reserved</b>	RsvdP	No	0h
31:20	<b>MEM_Limit[31:20]</b> Memory Limit Address[31:20]. Specifies the Port's Non-Prefetchable Memory Limit Address[31:20]. The PEX 8748 assumes Memory Limit Address[19:0]=F_FFFFh.	RW	Yes	000h

**Register 13-10. 24h Prefetchable Memory Base and Limit  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Note:</b> The Port forwards Memory transactions from its primary interface to its secondary interface (Downstream) if a Memory address is within the range defined by the <b>Prefetchable Memory Base</b> (offsets <b>28h</b> + 24h[15:0]) and <b>Prefetchable Memory Limit</b> (offsets <b>2Ch</b> + 24h[31:16]) registers (when the Base is less than or equal to the Limit).</p> <p>Conversely, the Port forwards Memory transactions from its secondary interface to its primary interface (Upstream) if a Memory address is outside this Address range (provided that the address is not within the range defined by the <b>Memory Base</b> and <b>Memory Limit</b> registers (offset <b>20h</b>)).</p>				
<b>Prefetchable Memory Base</b>				
0	<b>Prefetchable Memory Base Capability</b> 0 = Port supports 32-bit Prefetchable Memory Addressing 1 = Port defaults to 64-bit Prefetchable Memory Addressing support, as required by the <i>PCI Express Base r3.0</i>  <b>Note:</b> If the application needs 32-bit only Prefetchable space, the serial EEPROM and/or I <sup>2</sup> C must Clear both this bit and bit 16 (Prefetchable Memory Limit Capability).	RO	Yes	1
3:1	<b>Reserved</b>	RsvdP	No	000b
15:4	<b>PMEM_BAR[31:20]</b> Prefetchable Memory Base Address[31:20]. Specifies the Port's Prefetchable Memory Base Address[31:20]. The PEX 8748 assumes Prefetchable Memory Base Address[19:0]=0_0000h.	RW	Yes	FFFh
<b>Prefetchable Memory Limit</b>				
16	<b>Prefetchable Memory Limit Capability</b> 0 = Port supports 32-bit Prefetchable Memory Addressing 1 = Port defaults to 64-bit Prefetchable Memory Addressing support, as required by the <i>PCI Express Base r3.0</i>	RO	Yes	1
19:17	<b>Reserved</b>	RsvdP	No	000b
31:20	<b>PMEM_Limit[31:20]</b> Prefetchable Memory Limit Address[31:20]. Specifies the Port's Prefetchable Memory Limit Address[31:20]. The PEX 8748 assumes Prefetchable Memory Limit Address[19:0]=F_FFFFh.	RW	Yes	000h



**Register 13-11. 28h Prefetchable Memory Upper Base Address  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
31:0	<b>PBUP[63:32]</b> Prefetchable Memory Base Address[63:32]. The PEX 8748 uses this register for Prefetchable Memory Upper Base Address[63:32].	Offset 24h[0]=1	RW	Yes	0000_0000h
	When the <b>Prefetchable Memory Base</b> register <i>Prefetchable Memory Base Capability</i> field indicates 32-bit addressing, this register is RO and returns a value of 0000_0000h.	Offset 24h[0]=0	RO	No	0000_0000h

**Register 13-12. 2Ch Prefetchable Memory Upper Limit Address  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
31:0	<b>PLIMUP[63:32]</b> Prefetchable Memory Limit Address[63:32]. The PEX 8748 uses this register for Prefetchable Memory Upper Limit Address[63:32]. When the <b>Prefetchable Memory Limit</b> register <i>Prefetchable Memory Limit Capability</i> field indicates 32-bit addressing, this register is RO and returns a value of 0000_0000h. <i><b>Note:</b> The serial EEPROM must not write a non-zero value into this register when the RO attribute is Set for this register.</i>	Offset 24h[16]=1	RW	Yes	0000_0000h
	Offset 24h[16]=0	RO	No	0000_0000h	

**Register 13-13. 30h I/O Upper Base and Limit Address  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
I/O Base Upper 16 Bits					
15:0	<b>I/O Base Upper 16 Bits</b> The PEX 8748 uses this register for I/O Base Address[31:16]. When the <b>I/O Base</b> register <i>I/O Base Addressing Capability</i> field indicates 16-bit addressing, this register is RO and returns a value of 0000h.	Offset 1Ch[3:0]=1h	RW	Yes	0000h
		Offset 1Ch[3:0]=0h	RO	No	0000h
I/O Limit Upper 16 Bits					
31:16	<b>I/O Limit Upper 16 Bits</b> The PEX 8748 uses this register for I/O Limit Address[31:16]. When the <b>I/O Limit</b> register <i>I/O Limit Addressing Capability</i> field indicates 16-bit addressing, this register is RO and returns a value of 0000h. <i>Note: The serial EEPROM must not write a non-zero value into this register when the RO attribute is Set for this register.</i>	Offset 1Ch[11:8]=1h	RW	Yes	0000h
		Offset 1Ch[11:8]=0h	RO	No	0000h

**Register 13-14. 34h Capability Pointer  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Capability Pointer</b> Default 40h points to the <b>PCI Power Management Capability</b> structure.	RO	Yes	40h
31:8	<i>Reserved</i>	RsvdP	No	0000_00h

**Register 13-15. 38h Expansion ROM Base Address  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Expansion ROM Base Address</b> <i>Reserved</i>	RsvdP	No	0000_0000h

**Register 13-16. 3Ch Bridge Control and PCI Interrupt Signal (All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Interrupt Signal</b>				
7:0	<b>PCI Interrupt Line</b> Interrupt line routing value. The PEX 8748 does <i>not</i> use this register; however, the register is included for operating system and device driver use.	RW	Yes	00h
15:8	<b>PCI Interrupt Pin</b> Identifies the Conventional PCI Interrupt Message(s) that the device (or device function) uses. Only value 00h or 01h is allowed in the PEX 8748.  00h = Indicates that the device does not use Conventional PCI Interrupt Message(s) 01h, 02h, 03h, and 04h = Maps to Conventional PCI Interrupt Messages for INTA#, INTB#, INTC#, and INTD#, respectively	RO	Yes	01h
<b>Bridge Control</b>				
16	<b>Parity Error Response Enable</b> Controls the response to Poisoned TLPs.  0 = Disables the Port's <b>Secondary Status</b> register <i>Master Data Parity Error</i> bit (offset 1Ch[24]) 1 = Enables the Port's <b>Secondary Status</b> register <i>Master Data Parity Error</i> bit (offset 1Ch[24])	RW	Yes	0
17	<b>SERR# Enable</b> Controls forwarding of ERR_COR, ERR_FATAL, and ERR_NONFATAL from the secondary interface to the primary interface. When Set, and the Port's <b>PCI Command</b> register <i>SERR# Enable</i> bit (offset 04h[8]) is also Set, enables the Port's <b>PCI Status</b> register <i>Signaled System Error</i> bit (offset 04h[30]).	RW	Yes	0
18	<b>ISA Enable</b> Modifies the PEX 8748's response to Conventional PCI ISA I/O addresses enabled by the Port's <b>I/O Base</b> and <b>I/O Limit</b> registers (offset 1Ch[7:0 and 15:8], respectively) and located in the first 64 KB of the PCI I/O Address space (0000_0000h to 0000_FFFFh).  0 = Port's <b>I/O Base</b> and <b>I/O Limit</b> registers, and <b>I/O Base Upper 16 Bits</b> and <b>I/O Limit Upper 16 Bits</b> registers (offset 30h[15:0 and 31:16], respectively), define the Port's I/O Address range 1 = If the Port's I/O Address range (defined by the Port's <b>I/O Base</b> , <b>I/O Limit</b> , <b>I/O Base Upper 16 Bits</b> , <b>I/O Limit Upper 16 Bits</b> registers) includes I/O addresses below 64 KB, the upper 768 I/O addresses within each 1-KB block below 64 KB are excluded from the Port's I/O Address range  <i>Note: Refer also to the <b>Ingress Control</b> register <i>Disable VGA BIOS Memory Access Decoding</i> bit (Upstream Port(s), offset F60h[28]).</i>	RW	Yes	0

**Register 13-16. 3Ch Bridge Control and PCI Interrupt Signal  
(All Ports) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19	<p><b>VGA Enable</b></p> <p>Modifies the bridge response to VGA-compatible addresses.</p> <p>When Set, the bridge positively decodes and forwards the following addresses on the primary interface to the secondary interface (and, conversely, blocks forwarding of these addresses from the secondary interface to the primary interface):</p> <ul style="list-style-type: none"> <li>Memory addresses within the range 000A_0000h to 000B_FFFFh</li> <li>I/O addresses in the first 64 KB of the I/O Address space (AD[31:16] is 0000h), where AD[9:0] is within the ranges 3B0h to 3BBh and 3C0h to 3DFh (inclusive of ISA address aliases – AD[15:10] is not decoded)</li> </ul> <p>Additionally, when Set, forwarding of these addresses is independent of the:</p> <ul style="list-style-type: none"> <li>Memory and I/O Address ranges defined by the bridge <b>I/O Base</b>, <b>I/O Limit</b>, <b>Memory Base</b>, <b>Memory Limit</b>, <b>Prefetchable Memory Base</b>, and <b>Prefetchable Memory Limit</b> registers</li> <li>Bit 18 (<i>ISA Enable</i>) Setting</li> </ul> <p>VGA address forwarding is qualified by the Port's <b>PCI Command</b> register <i>Memory Access Enable</i> and <i>I/O Access Enable</i> bits (offset 04h[1:0], respectively). The default state of this bit after reset must be 0.</p> <p>0 = Do not forward VGA-compatible Memory and I/O addresses from the primary to the secondary interface (addresses defined above) unless they are enabled for forwarding by the defined Memory and I/O Address ranges</p> <p>1 = Forward VGA-compatible Memory and I/O addresses (addresses defined above) from the primary interface to the secondary interface (when the <i>Memory Access Enable</i> and <i>I/O Access Enable</i> bits are Set), independent of the Memory and I/O Address ranges and independent of the <i>ISA Enable</i> bit</p> <p><b>Notes:</b> When Set in an egress Port, the Port is configured as a non-Cut-Thru path. (Refer to <a href="#">Section 2.1.4.2, "Cut-Thru Mode,"</a> for further details.)</p> <p>Refer also to the <b>Ingress Control</b> register <i>Disable VGA BIOS Memory Access Decoding</i> bit (Upstream Port(s), offset F60h[28]).</p> <p><b>Conventional PCI VGA support</b> – To avoid potential I/O address conflicts, if the VGA Enable bit is Set in an Upstream Port and a Downstream Port, Set the Port's <b>PCI Command</b> register <i>I/O Access Enable</i> bit (offset 04h[0]) in the remaining Downstream Ports, unless those Downstream Ports are configured to use default 32-bit address decoding and their I/O Address range is programmed above 1_0000h.</p>	RW	Yes	0

**Register 13-16. 3Ch Bridge Control and PCI Interrupt Signal  
(All Ports) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>VGA 16-Bit Decode Enable</b> Used only when bit 19 ( <i>VGA Enable</i> ) is also Set, enabling VGA I/O decoding and forwarding by the bridge. Status after reset is 0. Enables system configuration software to select between 10- and 16-bit I/O address decoding, for VGA I/O register accesses forwarded from the primary interface to the secondary interface. 0 = Execute 10-bit address decodes on VGA I/O accesses 1 = Execute 16-bit address decodes on VGA I/O accesses <i>Note:</i> Refer also to the <i>Ingress Control</i> register <i>Disable VGA BIOS Memory Access Decoding</i> bit (Upstream Port(s), offset F60h[28]).	RW	Yes	0
21	<b>Master Abort Mode</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
22	<b>Secondary Bus Reset</b> 1 = Causes a Hot Reset on the Port's Downstream Link	RW	Yes	0
23	<b>Fast Back-to-Back Transactions Enable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
24	<b>Primary Discard Timer</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
25	<b>Secondary Discard Timer</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
26	<b>Discard Timer Status</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
27	<b>Discard Timer SERR# Enable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
31:28	<b>Reserved</b>	RsvdP	No	0h

This section details the PCI Power Management Capability registers. [Table 13-8](#) defines the register map.

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

PCI Power Management Capability		Next Capability Pointer (48h)	Capability ID (01h)	40h
PCI Power Management Data	PCI Power Management Control/Status Bridge Extensions <i>(Reserved)</i>	PCI Power Management Status and Control		44h

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Capability ID</b> Program to 01h, to indicate that the Capability structure is the <b>PCI Power Management Capability</b> structure.	RO	Yes	01h
15:8	<b>Next Capability Pointer</b> Default 48h points to the <b>MSI Capability</b> structure.	RO	Yes	<a href="#">48h</a>
18:16	<b>Version</b> Default 011b indicates compliance with the <i>PCI Power Mgmt. r1.2</i> .	RO	Yes	011b
19	<b>PME Clock</b> Power Management Event (PME) clock. Does not apply to PCI Express. Returns a value of 0.	RsvdP	No	0
20	<b>Reserved</b>	RsvdP	No	0
21	<b>Device-Specific Initialization</b> 0 = Device-Specific Initialization is <i>not</i> required	RO	Yes	0
24:22	<b>AUX Current</b> The PEX 8748 does <i>not support</i> PME generation from the D3cold Device Power Management (PM) state; therefore, the serial EEPROM value for this field should be 000b.	RO	Yes	000b
25	<b>D1 Support</b> <i>Not supported</i> 0 = PEX 8748 does <i>not support</i> the D1 Device PM state	RsvdP	No	0
26	<b>D2 Support</b> <i>Not supported</i> 0 = PEX 8748 does <i>not support</i> the D2 Device PM state	RsvdP	No	0
31:27	<b>PME Support</b> Bits [31, 30, and 27] must be Set, to indicate that the PEX 8748 will forward PME Messages, as required by the <i>PCI Express Base r3.0</i> .	RO	Yes	19h

**Register 13-18. 44h PCI Power Management Status and Control  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Power Management Status and Control</b>				
1:0	<b>Power State</b> Used to determine the Port's current Device PM state, and to program the Port into a new Device PM state. 00b = D0 01b = D1 – <i>Not supported</i> 10b = D2 – <i>Not supported</i> 11b = D3hot If software attempts to write an unsupported state to this field, the Write operation completes normally; however, the data is discarded and no state change occurs.	RW	Yes	00b
2	<b>Reserved</b>	RsvdP	No	0
3	<b>No Soft Reset</b> 1 = Devices transitioning from the D3hot to D0 state, because of Power State commands, do not perform an internal reset	ROS	Yes	1
7:4	<b>Reserved</b>	RsvdP	No	0h
8	<b>PME Enable</b> 0 = Disables PME generation by the Port <sup>a</sup> 1 = Enables PME generation by the Port	RWS	No	0
12:9	<b>Data Select</b> Initially writable by serial EEPROM and/or I <sup>2</sup> C only <sup>b</sup> . This Configuration Space register (CSR) access privilege changes to RW after a Serial EEPROM and/or I <sup>2</sup> C Write occurs to this register. Selects the Port's field [14:13] ( <i>Data Scale</i> ) and <b>PCI Power Management Data</b> register <i>Data</i> field (offset 44h[31:24]). 0h = D0 power consumed 3h = D3hot power consumed 4h = D0 power dissipated 7h = D3hot power dissipated All other encodings are <b>Reserved</b> .	RO	Yes	0h
14:13	<b>Data Scale</b> Writable by serial EEPROM and/or I <sup>2</sup> C only <sup>b</sup> . Indicates the scaling factor to be used when interpreting the <b>PCI Power Management Data</b> register <i>Data</i> field (offset 44h [31:24]) value. The value and meaning of the <i>Data Scale</i> field varies, depending upon which data value is selected by field [12:9] ( <i>Data Select</i> ). There are four internal <i>Data Scale</i> fields (one each, per <i>Data Select</i> values 0h, 3h, 4h, and 7h), per Port. For other <i>Data Select</i> values, the <i>Data Scale</i> value returned is 0h.	RO	Yes	00b
15	<b>PME Status</b> 0 = PME is not generated by the Port <sup>a</sup> 1 = PME is being generated by the Port	RW1CS	No	0

**Register 13-18. 44h PCI Power Management Status and Control  
(All Ports) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Power Management Control/Status Bridge Extensions</b>				
21:16	<i>Reserved</i>	RsvdP	No	0-0h
22	<b>B2/B3 Support</b> <i>Reserved</i> Cleared, as required by the <i>PCI Power Mgmt. r1.2</i> .	RsvdP	No	0
23	<b>Bus Power/Clock Control Enable</b> <i>Reserved</i> Cleared, as required by the <i>PCI Power Mgmt. r1.2</i> .	RsvdP	No	0
<b>PCI Power Management Data</b>				
31:24	<b>Data</b> Writable by serial EEPROM and/or I <sup>2</sup> C only <sup>b</sup> . There are four supported <i>Data Select</i> values (0h, 3h, 4h, and 7h), per Port. For other <i>Data Select</i> values, the <i>Data</i> value returned is 00h. Selected by the Port's <b>PCI Power Management Status and Control</b> register <i>Data Select</i> field (offset 44h[12:9]).	RO	Yes	00h

- a. Because the PEX 8748 does not consume auxiliary power, this bit is not sticky, and is always Cleared at power-on reset.
- b. With no serial EEPROM nor previous I<sup>2</sup>C programming, Reads return a value of 00h for the **PCI Power Management Status and Control** register *Data Scale* and **PCI Power Management Data** register *Data* fields (for all *Data Selects*).



This section details the Message Signaled Interrupt (MSI) Capability registers. [Table 13-9](#) defines the register map.

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

MSI Control	Next Capability Pointer (68h)	Capability ID (05h)	48h
MSI Address			4Ch
MSI Upper Address			50h
<i>Reserved</i>	MSI Data		54h
<i>Reserved</i>		MSI Mask	58h
MSI Status			5Ch
<i>Reserved</i>			60h – 64h

**Register 13-19. 48h MSI Capability  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>MSI Capability Header</b>				
7:0	<b>Capability ID</b> Program to 05h, as required by the <i>PCI r3.0</i> .	RO	Yes	05h
15:8	<b>Next Capability Pointer</b> Program to 68h, to point to the <b>PCI Express Capability</b> structure.	RO	Yes	68h
<b>MSI Control</b>				
16	<b>MSI Enable</b> 0 = MSIs for the Port are disabled 1 = MSIs for the Port are enabled, and INT <sub>x</sub> Interrupt Messages and <a href="#">PEX_INTA#</a> (Base mode) or VS <sub>x</sub> _PEX_INTA# (Virtual Switch mode) output assertion are disabled	RW	Yes	0
19:17	<b>Multiple Message Capable</b> 000b = Port can request only one MSI Vector 001b = Port can request two MSI Vectors 010b = Port can request four MSI Vectors 011b = Port can request eight MSI Vectors All other encodings are <i>Reserved</i> .	RO	Yes	011b
22:20	<b>Multiple Message Enable</b> 000b = Port is allocated one MSI Vector, by default. 001b = Port is allocated two MSI Vectors. 010b = Port is allocated four MSI Vectors. 011b = Port is allocated eight MSI Vectors. Up to six MSI Vectors are used; the upper two MSI Vectors (having the highest values of Message Data bits [2:0]) are <i>not</i> used. All other encodings are <i>Reserved</i> . <i>Note:</i> This field should not be programmed with a value larger than that of field [19:17] (Multiple Message Capable). If the value of this field is larger than that of field [19:17], the Multiple Message Capable value takes effect.	RW	Yes	000b
23	<b>MSI 64-Bit Address Capable</b> 0 = PEX 8748 is capable of generating MSI 32-bit addresses (Port's <b>MSI Address</b> register, offset 4Ch, is the Message address) 1 = PEX 8748 is capable of generating MSI 64-bit addresses (Port's <b>MSI Address</b> register, offset 4Ch, is the lower 32 bits of the Message address, and <b>MSI Upper Address</b> register, offset 50h, is the upper 32 bits of the Message address)	RO	Yes	1
24	<b>Per Vector Masking Capable</b> 0 = PEX 8748 does not have Per Vector Masking capability 1 = PEX 8748 has Per Vector Masking capability	RO	Yes	1
31:25	<i>Reserved</i>	RsvdP	No	0-0h

**Register 13-20. 4Ch MSI Address  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<i>Reserved</i>	RsvdP	No	00b
31:2	<b>Message Address</b> <i>Note: If the Port's <b>MSI Control</b> register <b>MSI 64-Bit Address Capable</b> bit (offset 48h[23]) is Set (default), refer to the Port's <b>MSI Upper Address</b> register for the MSI Upper Address (offset 50h).</i>	RW	Yes	0-0h

**Register 13-21. 50h MSI Upper Address  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Message Upper Address</b> This register is valid/used only when the Port's <b>MSI Control</b> register <b>MSI 64-Bit Address Capable</b> bit (offset 48h[23]) is Set. MSI Write transaction upper address[63:32]. <i>Note: Refer to register offset 4Ch for the MSI Address.</i>	RW	Yes	0000_0000h

**Register 13-22. 54h MSI Data  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note: The offset for this register changes from 54h, to 50h, when the Port's <b>MSI Control</b> register <b>MSI 64-Bit Address Capable</b> bit (offset 48h[23]) is Cleared.</i>				
15:0	<b>Message Data</b> MSI Write transaction TLP payload.	RW	Yes	0000h
31:16	<i>Reserved</i>	RsvdP	No	0000h

**Register 13-23. 58h MSI Mask  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>Port interrupt sources are grouped into six categories – Hot Plug or Link State-/PM events, Device-Specific NT-Link Port or RAM ECC error events, GPIO-generated interrupts, Virtual Switch mode Doorbell events, and Virtual Switch mode Link Status events. The quantity of MSI Vectors that are generated is determined by the Port's <b>MSI Control</b> register <i>Multiple Message Capable</i> and <i>Multiple Message Enable</i> fields (offset 48h[19:17 and 22:20], respectively):</p> <ul style="list-style-type: none"> <li>• If <b>one</b> MSI Vector is enabled (default), all six interrupt events are combined into a single Vector</li> <li>• If <b>two</b> MSI Vectors are allocated, the individual Vectors indicate: <ul style="list-style-type: none"> <li>– Vector[0] Hot Plug or Link State-/PM-triggered interrupt event, GPIO-generated event, Virtual Switch Doorbell event, and Management Port Link Status event</li> <li>– Vector[1] Device-specific NT-Link Port event or RAM ECC error event</li> </ul> </li> <li>• If <b>four</b> MSI Vectors are allocated, the individual Vectors indicate: <ul style="list-style-type: none"> <li>– Vector[0] Hot Plug or Link State-/PM-triggered interrupt event</li> <li>– Vector[1] Device-specific NT-Link Port event or RAM ECC error event</li> <li>– Vector[2] GPIO-generated event</li> <li>– Vector[3] Virtual Switch Doorbell event, and Management Port Link Status event</li> </ul> </li> <li>• If <b>eight</b> MSI Vectors are allocated (up to six Vectors are used; the upper two Vectors (having the highest values of Message Data bits [2:0]) are <b>not</b> used), the individual Vectors indicate: <ul style="list-style-type: none"> <li>– Vector[0] Hot Plug or Link State-/PM-triggered interrupt event</li> <li>– Vector[1] Device-specific NT-Link Port event or RAM ECC error event</li> <li>– Vector[2] GPIO-generated event</li> <li>– Vector[3] <b>Reserved</b></li> <li>– Vector[4] Virtual Switch Doorbell event</li> <li>– Vector[5] Management Port Link Status event</li> </ul> </li> </ul> <p><b>Notes:</b> The offset for this register changes from 58h, to 54h, when the Port's <b>MSI Control</b> register <i>MSI 64-Bit Address Capable</i> bit (offset 48h[23]) is Cleared.</p> <p>The bits in this register can be used to mask the Port's respective <b>MSI Status</b> register bits (offset 5Ch).</p>					
0	<b>MSI Mask for Hot Plug or Link State Events</b> MSI mask for Power Management event- or Hot Plug or Link State event-generated interrupts.	All Ports, when offset 48h[22:20]=010b or 011b	RW	Yes	0
	<b>MSI Mask for Shared Interrupt Sources</b> MSI mask for all interrupt sources when the <b>MSI Control</b> register <i>Multiple Message Enable</i> field indicates that the Host has allocated one or two Vectors.	All Ports, when offset 48h[22:20]≤001b	RW	Yes	0

**Register 13-23. 58h MSI Mask  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
1	<b>MSI Mask for Device-Specific NT-Link Port Events</b> <i>This bit is valid only in NT mode.</i> MSI mask for Device-Specific NT-Link Port event-generated interrupts. This bit (implemented only for the NT Port Virtual Interface) enables MSIs for the following NT-Link Port events, defined in the <b>Link Error Status Virtual</b> and <b>Link Error Mask Virtual</b> registers (NT Port Virtual Interface, offsets FE0h and FE4h, respectively): <ul style="list-style-type: none"> <li>NT-Link Port Correctable error (NT Port Link Interface, offset FC4h)</li> <li>NT-Link Port Uncorrectable error (NT Port Link Interface, offset FB8h)</li> <li>NT-Link Port Data Link Layer State change</li> <li>NT-Link Port received (and consequently dropped) a Fatal or Non-Fatal Error Message</li> </ul>	NT Port Virtual Interface, when offset 48h[22:20]=001b, 010b or 011b	RW	Yes	0
	<i>Reserved</i>	Otherwise, when offset 48h[22:20]=000b	RsvdP	No	0
2	<b>Base Mode MSI Mask for GPIO-Generated Interrupts</b>	Port 0, when offset 48h[22:20]=010b or 011b	RW	Yes	0
	<b>Virtual Switch Mode MSI Mask for GPIO-Generated Interrupts</b>	VS Upstream Port(s), when offset 48h[22:20]=010b or 011b	RW	Yes	0
	<i>Reserved</i>	Otherwise, when offset 48h[22:20]≤001b	RsvdP	No	0
3	<i>Reserved</i>		RW	Yes	0
4	<b>Base Mode</b> <i>Reserved</i>		RsvdP	No	0
	<b>Virtual Switch Mode MSI Mask for Management Port Doorbell-Generated Interrupts</b>	VS Upstream Port(s) and Management Port, when offset 48h[22:20]=011b	RW	Yes	0
	<i>Reserved</i>	Otherwise, when offset 48h[22:20]≤010b	RsvdP	No	0
5	<b>Base Mode</b> <i>Reserved</i>		RsvdP	No	0
	<b>Virtual Switch Mode MSI Mask for Management Link Status Event</b>	Management Port, when offset 48h[22:20]=011b	RW	Yes	0
	<i>Reserved</i>	Otherwise, when offset 48h[22:20]≤010b	RsvdP	No	0
7:6	<i>Reserved</i>		RW	Yes	00b
31:8	<i>Reserved</i>		RsvdP	No	0000_00h

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 13-24. 5Ch MSI Status  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>Port interrupt sources are grouped into six categories – Hot Plug or Link State-/PM events, Device-Specific NT-Link Port or RAM ECC error events, GPIO-generated interrupts, Virtual Switch mode Doorbell events, and Virtual Switch mode Link Status events. The quantity of MSI Vectors that are generated is determined by the Port's <b>MSI Control</b> register <i>Multiple Message Capable</i> and <i>Multiple Message Enable</i> fields (offset 48h[19:17 and 22:20], respectively):</p> <ul style="list-style-type: none"> <li>• If <b>one</b> MSI Vector is enabled (default), all six interrupt events are combined into a single Vector</li> <li>• If <b>two</b> MSI Vectors are allocated, the individual Vectors indicate: <ul style="list-style-type: none"> <li>– Vector[0] Hot Plug or Link State-/PM-triggered interrupt event, GPIO-generated event, Virtual Switch Doorbell event, and Management Port Link Status event</li> <li>– Vector[1] Device-specific NT-Link Port event or RAM ECC error event</li> </ul> </li> <li>• If <b>four</b> MSI Vectors are allocated, the individual Vectors indicate: <ul style="list-style-type: none"> <li>– Vector[0] Hot Plug or Link State-/PM-triggered interrupt event</li> <li>– Vector[1] Device-specific NT-Link Port event or RAM ECC error event</li> <li>– Vector[2] GPIO-generated event</li> <li>– Vector[3] Virtual Switch Doorbell event, and Management Port Link Status event</li> </ul> </li> <li>• If <b>eight</b> MSI Vectors are allocated (up to six Vectors are used; the upper two Vectors (having the highest values of Message Data bits [2:0]) are <b>not</b> used), the individual Vectors indicate: <ul style="list-style-type: none"> <li>– Vector[0] Hot Plug or Link State-/PM-triggered interrupt event</li> <li>– Vector[1] Device-specific NT-Link Port event or RAM ECC error event</li> <li>– Vector[2] GPIO-generated event</li> <li>– Vector[3] <b>Reserved</b></li> <li>– Vector[4] Virtual Switch Doorbell event</li> <li>– Vector[5] Management Port Link Status event</li> </ul> </li> </ul> <p><b>Notes:</b> The offset for this register changes from 5Ch, to 58h, when the Port's <b>MSI Control</b> register <i>MSI 64-Bit Address Capable</i> bit (offset 48h[23]) is Cleared.</p> <p>The bits in this register can be masked by the Port's respective <b>MSI Mask</b> register bits (offset 58h).</p>					
0	<b>MSI Pending Status for Hot Plug or Link State Events</b> MSI pending status for Power Management event- or Hot Plug or Link State event-generated interrupts.	All Ports, when offset 48h[22:20]=010b or 011b	RO	No	0
	<b>MSI Pending Status for Shared Interrupt Sources</b> MSI pending status for all interrupt sources when the <b>MSI Control</b> register <i>Multiple Message Enable</i> field indicates that the Host has allocated one or two Vectors.	All Ports, when offset 48h[22:20]≤001b	RsvdP	No	0

**Register 13-24. 5Ch MSI Status**  
**(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
1	<b>MSI Pending Status for Device-Specific NT-Link Port Events</b> <i>This bit is valid only in NT mode.</i>  This bit (implemented only for the NT Port Virtual Interface) enables MSIs for the following NT-Link Port events, defined in the <b>Link Error Status Virtual</b> and <b>Link Error Mask Virtual</b> registers (NT Port Virtual Interface, offsets FE0h and FE4h, respectively): <ul style="list-style-type: none"> <li>NT-Link Port Correctable error (NT Port Link Interface, offset FC4h)</li> <li>NT-Link Port Uncorrectable error (NT Port Link Interface, offset FB8h)</li> <li>NT-Link Port Data Link Layer State change</li> <li>NT-Link Port received (and consequently dropped) a Fatal or Non-Fatal Error Message</li> </ul>	NT Port Virtual Interface, when offset 48h[22:20]=001b, 010b or 011b	RO	No	0
	<i>Reserved</i>	Otherwise, when offset 48h[22:20]=000b	RsvdP	No	0
2	<b>Base Mode MSI Pending Status for GPIO-Generated Interrupts</b>	Port 0, when offset 48h[22:20]=010b or 011b	RO	No	0
	<b>Virtual Switch Mode MSI Pending Status for GPIO-Generated Interrupts</b>	VS Upstream Port(s), when offset 48h[22:20]=010b or 011b	RO	No	0
	<i>Reserved</i>	Otherwise, when offset 48h[22:20]≤001b	RsvdP	No	0
3	<i>Reserved</i>		RW	Yes	0
4	<b>Base Mode MSI Pending Status for Management Port Doorbell-Generated Interrupts</b>	VS Upstream Port(s) and Management Port, when offset 48h[22:20]=011b	RO	No	0
	<i>Reserved</i>	Otherwise, when offset 48h[22:20]≤010b	RsvdP	No	0
	<i>Reserved</i>		RsvdP	No	0
5	<b>Base Mode MSI Pending Status for Management Link Status Event</b>	Management Port, when offset 48h[22:20]=011b	RO	No	0
	<i>Reserved</i>	Otherwise, when offset 48h[22:20]≥010b	RsvdP	No	0
	<i>Reserved</i>		RsvdP	No	0
31:6	<i>Reserved</i>		RsvdP	No	0-0h

## 13.10 PCI Express Capability Registers (Offsets 68h – A0h)

This section details the PCI Express Capability registers. Hot Plug Capability, Command, Status, and Events are included in these registers. [Table 13-10](#) defines the register map.

**Table 13-10. PCI Express Capability Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																																
PCI Express Capability																Next Capability Pointer (A4h)																Capability ID (10h)																68h	
Device Capability																																																6Ch	
Device Status																<i>Reserved/Not supported</i>																Device Control																70h	
Link Capability																																																74h	
Link Status																Link Control																																78h	
<i>Reserved</i> (Upstream)																																																7Ch	
Slot Capability (Downstream)																																																	
<i>Reserved</i> (Upstream)																																																80h	
Slot Status (Downstream)																Slot Control (Downstream)																																	
<i>Reserved</i>																																																84h –	88h
Device Capability 2																																																8Ch	
Device Status 2 ( <i>Reserved</i> )																Device Control 2																																90h	
Link Capability 2																																																94h	
Link Status 2																Link Control 2																																98h	
<i>Reserved</i>																																																9Ch –	A0h



**Register 13-25. 68h PCI Express Capability List and Capability (All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Express Capability List</b>					
7:0	<b>Capability ID</b> Program to 10h, as required by the <i>PCI Express Base r3.0</i> .		RO	Yes	10h
15:8	<b>Next Capability Pointer</b> Program to A4h, to point to the <b>Subsystem Capability</b> structure.		RO	Yes	A4h
<b>PCI Express Capability</b>					
19:16	<b>Capability Version</b> The PEX 8748 Ports program this field to 2h, as required by the <i>PCI Express Base r3.0</i> .		RO	Yes	2h
23:20	<b>Device/Port Type</b> Set at reset, as required by the <i>PCI Express Base r3.0</i> .	Upstream	RO	Yes	5h
		Downstream	RO	Yes	6h
24	<b>Slot Implemented</b> 0 = Disables or connects to an Upstream Port	Upstream	RsvdP	No	0
	0 = Disables or connects to an integrated component 1 = Indicates that the Downstream Port connects to a slot, as opposed to being connected to an integrated component or being disabled <i>Note: The PEX 8748 serial EEPROM register Initialization capability, and/or I<sup>2</sup>C/SMBus when STRAP_I2C_SMBUS_CFG_EN#=L, can be used to Clear this bit, indicating that the Downstream Port connects to an integrated component or is disabled.</i>	Downstream	ROS	Yes	1
29:25	<b>Interrupt Message Number</b> The serial EEPROM writes 00_000b, because the Base Message and MSI Messages are the same.		RO	Yes	00_000b
31:30	<b>Reserved</b>		RsvdP	No	00b

**Register 13-26. 6Ch Device Capability  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
2:0	<b>Maximum Payload Size Supported</b> <i>Note:</i> The default Maximum Payload Size supported is based upon the STRAP_STNx_PORTCFGx 3-state inputs. If the serial EEPROM or I <sup>2</sup> C/SMBus changes the Port configuration (from the strapped value), it must also program the value of this field, accordingly (with the value dependent upon the Station having the most enabled Ports). The value for all Ports is determined by the Station having the most enabled Ports: <ul style="list-style-type: none"> <li>One or two enabled Ports = 100b</li> <li>Three enabled Ports = 011b</li> <li>Four enabled Ports = 010b</li> </ul> 000b = Port supports a 128-byte maximum payload 001b = Port supports a 256-byte maximum payload 010b = Port supports a 512-byte maximum payload 011b = Port supports a 1,024-byte maximum payload 100b = Port supports a 2,048-byte maximum payload No other encodings are supported.		HwInit	Yes	Defined by STRAP_STNx_PORTCFGx input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station x</i> field(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6, 5:3, and/or 2:0])
4:3	<b>Phantom Functions Supported</b> <i>Not supported</i>		RO	Yes	00b
5	<b>Extended Tag Field Supported</b> 0 = Maximum <i>Tag</i> field is 5 bits 1 = Maximum <i>Tag</i> field is 8 bits		RO	Yes	0
8:6	<b>Endpoint L0s Acceptable Latency</b> <i>Not supported</i> Because the PEX 8748 is a switch and not an endpoint, the PEX 8748 does <b>not support</b> this feature. 000b = Disables the capability		RO	Yes	000b
11:9	<b>Endpoint L1 Acceptable Latency</b> <i>Not supported</i> Because the PEX 8748 is a switch and not an endpoint, the PEX 8748 does <b>not support</b> this feature. 000b = Disables the capability		RO	Yes	000b
14:12	<b>Reserved</b> , as required by the <i>PCI Express Base r3.0</i> .		RsvdP	No	000b
15	<b>Role-Based Error Reporting</b>		RO	Yes	1

**Register 13-26. 6Ch Device Capability  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
17:16	<i>Reserved</i>		RsvdP	No	00b
25:18	<b>Captured Slot Power Limit Value</b> For Upstream Port(s), the upper limit on power supplied by the slot is determined by multiplying the value in this field by the value in field [27:26] ( <i>Captured Slot Power Limit Scale</i> ).	Upstream	RO	Yes	00h
	<i>Not valid</i>	Downstream	RsvdP	No	00h
27:26	<b>Captured Slot Power Limit Scale</b> For Upstream Port(s), the upper limit on power supplied by the slot is determined by multiplying the value in this field by the value in field [25:18] ( <i>Captured Slot Power Limit Value</i> ). 00b = 1.0 01b = 0.1 10b = 0.01 11b = 0.001	Upstream	RO	Yes	00b
	<i>Not valid</i>	Downstream	RsvdP	No	00b
31:28	<i>Reserved</i>		RsvdP	No	0h

**Register 13-27. 70h Device Status and Control  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Control</b>				
0	<b>Correctable Error Reporting Enable</b> 0 = Disables 1 = Enables the Port to report Correctable errors	RW	Yes	0
1	<b>Non-Fatal Error Reporting Enable</b> 0 = Disables 1 = Enables the Port to report Non-Fatal errors	RW	Yes	0
2	<b>Fatal Error Reporting Enable</b> 0 = Disables 1 = Enables the Port to report Fatal errors	RW	Yes	0
3	<b>Unsupported Request Reporting Enable</b> 0 = Disables 1 = Enables the Port to report UR errors	RW	Yes	0
4	<b>Enable Relaxed Ordering</b> <i>Not supported</i> The value of this bit has no effect upon internal logic.	RW	Yes	1
7:5	<b>Maximum Payload Size</b> Software can change this field to configure the Ports to support other Payload Sizes; however, software cannot change this field to a value larger than that indicated by the <b>Device Capability</b> register <i>Maximum Payload Size Supported</i> field (offset 6Ch[2:0]).  000b = Port supports a 128-byte maximum payload 001b = Port supports a 256-byte maximum payload 010b = Port supports a 512-byte maximum payload 011b = Port supports a 1,024-byte maximum payload 100b = Port supports a 2,048-byte maximum payload  No other encodings are supported.	RW	Yes	000b
8	<b>Extended Tag Field Enable</b> <i>Not supported</i>	RsvdP	No	0
9	<b>Phantom Functions Enable</b> <i>Not supported</i>	RsvdP	No	0
10	<b>AUX Power PM Enable</b> <i>Not supported</i>	RsvdP	No	0
11	<b>Enable No Snoop</b> <i>Not supported</i>	RW	Yes	1
14:12	<b>Max Read Request Size</b> <i>Not supported</i>	RsvdP	No	000b
15	<b>Reserved</b> Hardwired to 0, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

**Register 13-27. 70h Device Status and Control  
(All Ports) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Status</b>				
16	<b>Correctable Error Detected</b> 0 = Port did not detect a Correctable error 1 = Port detected a Correctable error, regardless of the bit 0 ( <i>Correctable Error Reporting Enable</i> ) state	RW1C	Yes	0
17	<b>Non-Fatal Error Detected</b> 0 = Port did not detect a Non-Fatal error 1 = Port detected a Non-Fatal error, regardless of the bit 1 ( <i>Non-Fatal Error Reporting Enable</i> ) state	RW1C	Yes	0
18	<b>Fatal Error Detected</b> 0 = Port did not detect a Fatal error 1 = Port detected a Fatal error, regardless of the bit 2 ( <i>Fatal Error Reporting Enable</i> ) state	RW1C	Yes	0
19	<b>Unsupported Request Detected</b> 0 = Port did not detect a UR 1 = Port detected a UR, regardless of the bit 3 ( <i>Unsupported Request Reporting Enable</i> ) state	RW1C	Yes	0
20	<b>AUX Power Detected</b> <i>Not supported</i>	RsvdP	No	0
21	<b>Transactions Pending</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 13-28. 74h Link Capability  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Ports and Lanes (Link widths).					
3:0	<b>Maximum Link Speed</b> Indicates the Port's maximum Link speed. The encoding is the binary value of the bit location in the Port's <b>Link Capability 2</b> register <i>Supported Link Speeds Vector</i> field (offset 94h[7:1]) that corresponds to the maximum Link speed. <i>For example</i> , the default value 3h indicates that the maximum Link speed is that which corresponds to bit 3 in the <i>Supported Link Speeds Vector</i> field, which is 8.0 GT/s. The default value for all Ports can be globally changed to 2h (Gen 2 maximum speed) or 1h (Gen 1 maximum speed), according to the STRAP_GEN1_GEN2 input state (Z = 2h (Gen 2), or 0 = 1h (Gen 1)).		RO	Yes	3h
9:4	<b>Maximum Link Width</b> The PEX 8748 maximum Link width is x16 = 01_0000b. Actual maximum Link width is defined by the STRAP_STNx_PORTCFGx 3-state inputs. 00_0000b = <b>Reserved</b> 00_0001b = x1 00_0010b = x2 00_0100b = x4 00_1000b = x8 01_0000b = x16 All other encodings are <b>not supported</b> .		ROS	No	Defined by STRAP_STNx_PORTCFGx input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> -related register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offsets 300h through 308h)
11:10	<b>Active State Power Management (ASPM) Support</b> Active State Link PM support. Indicates the level of ASPM supported by the Port. 00b = No ASPM support 01b = L0s Link PM state is supported 10b = L1 Link PM state is supported 11b = L0s and L1 Link PM states are supported		RO	Yes	11b

**Register 13-28. 74h Link Capability  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
14:12	<p><b>L0s Exit Latency</b></p> <p>Indicates the L0s Link PM state exit latency for the Port's PCI Express Link. The value reported indicates the length of time this Port requires to complete the L0s to L0 Link PM state transition.</p> <p>If L0s is not supported, the value of this field is undefined; however, for the recommended value, refer to the Implementation Note in the <i>PCI Express Base r3.0</i>, Section 5.4.1.1, "Potential Issues with Legacy Software When L0s is Not Supported."</p> <p>000b = Less than 64 ns  001b = 64 ns to less than 128 ns  010b = 128 ns to less than 256 ns  011b = 256 ns to less than 512 ns  100b = 512 ns to less than 1 <math>\mu</math>s  101b = 1 <math>\mu</math>s to less than 2 <math>\mu</math>s  110b = 2 to 4 <math>\mu</math>s (default)  111b = More than 4 <math>\mu</math>s</p>		RO	No	110b

**Register 13-28. 74h Link Capability  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
17:15	<b>L1 Exit Latency</b> Indicates the L1 Link PM state exit latency for the Port's PCI Express Link. The value reported indicates the length of time that this Port requires to complete the ASPM L1 to L0 Link PM state transition. If L0 is not supported, the value of this field is undefined.  000b = Less than 1 $\mu$ s 001b = 1 $\mu$ s to less than 2 $\mu$ s (5.0 GT/s) 010b = 2 $\mu$ s to less than 4 $\mu$ s (default, 2.5 GT/s) 011b = 4 $\mu$ s to less than 8 $\mu$ s 100b = 8 $\mu$ s to less than 16 $\mu$ s 101b = 16 $\mu$ s to less than 32 $\mu$ s 110b = 32 to 64 $\mu$ s 111b = More than 64 $\mu$ s		RO	Yes	010b
18	<b>Clock Power Management Capable</b> <i>Not supported</i>	Upstream	RO	Yes	0
	<i>Not supported</i> Must be hardwired to 0.	Downstream	RO	No	0
19	<b>Reserved</b> Must be hardwired to 0, for Upstream Port(s) and components that do not support this optional capability.	Upstream	RsvdP	No	0
	<b>Surprise Down Error Reporting Capable</b> Must be Set if the component supports the optional capability of detecting and reporting a Surprise Down error condition. If this bit is Cleared, the Port's <b>Uncorrectable Error Status</b> register <i>Surprise Down Error Status</i> bit (Downstream Ports, offset FB8h[5]) is disabled.	Downstream	RO	Yes	1



**Register 13-28. 74h Link Capability  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>Data Link Layer Link Active Reporting Capable</b>	Downstream	RO	Yes	1
21	<i>Reserved</i> Hardwired to 0, as required by the <i>PCI Express Base r3.0</i> .	Upstream	RsvdP	No	0
	<b>Link Bandwidth Notification Capability</b> 1 = Indicates support for the Link Bandwidth Notification status and interrupt mechanisms	Downstream	RO	Yes	1
22	<b>ASPM Optionality Compliance</b>		HwInit	Yes	1
23	<i>Reserved</i>		RsvdP	No	0
31:24	<b>Port Number</b> The Port Number is defined by 3-state input Strapping options.  <i>Note: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</i>  Station 0, STRAP_STN0_PORTCFG[1:0] – Ports 0, 1, 2, 3 Station 1, STRAP_STN1_PORTCFG[1:0] – Ports 8, 9, 10, 11 Station 2, STRAP_STN2_PORTCFG[1:0] – Ports 16, 17, 18, 19		ROS	Yes	Defined by STRAP_STNx_PORTCFGx input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station x</i> field(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6, 5:3, and/or 2:0])

**Register 13-29. 78h Link Status and Control  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Link Control</b>					
1:0	<b>Active State Power Management (ASPM)</b> 00b = Disable <sup>a</sup> 01b = Enables only L0s Link PM state Entry 10b = Enables only L1 Link PM state Entry 11b = Enables both L0s and L1 Link PM state Entries		RW	Yes	00b
2	<i>Reserved</i>		RsvdP	No	0
3	<b>Read Request Return Parameter Control</b> Read Request Return Parameter “R” control. Read Completion Boundary (RCB). Cleared, as required by the <i>PCI Express Base r3.0</i> .		RO	Yes	0

**Register 13-29. 78h Link Status and Control  
(All Ports) (Cont.)**

Bit(s)	Description		Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<i>Not valid</i>	Offset F70h[29]=0	Upstream	RsvdP	No	0
	<b>Link Disable</b> Functionality is enabled when the Port's <b>Power Management Hot Plug User Configuration</b> register <i>PHY Upstream Port Control Write Enable</i> bit (offset F70h[29]) is Set.  1 = Places the Upstream Port(s)' Link(s) into the <i>Disabled</i> Link Training state	Offset F70h[29]=1	Upstream	RW	Yes	0
	1 = Places the Downstream Port's Link into the <i>Disabled</i> Link Training state		Downstream	RW	Yes	0
5	<i>Not valid</i> Always read as 0.	Offset F70h[29]=0	Upstream	RsvdP	No	0
	<b>Retrain Link</b> Functionality is enabled when the Port's <b>Power Management Hot Plug User Configuration</b> register <i>PHY Upstream Port Control Write Enable</i> bit (offset F70h[29]) is Set.  Always returns a value of 0 when read; however, software is allowed to write this register.  Writing 1 to this bit causes Upstream Port(s) to initiate retraining of its (their) PCI Express Link.	Offset F70h[29]=1	Upstream	RZ	Yes	0
	Always returns a value of 0 when read; however, software is allowed to write this register.  Writing 1 to this bit causes the Downstream Port to initiate retraining of its PCI Express Link.		Downstream	RZ	Yes	0
6	<b>Common Clock Configuration</b> 0 = Port and the device at the other end of the Port's PCI Express Link use an asynchronous Reference Clock source 1 = Port and the device at the other end of the Port's PCI Express Link use a common (synchronous) Reference Clock source (constant phase relationship)			RW	Yes	0
7	<b>Extended Sync</b> Setting this bit causes the Port to transmit: <ul style="list-style-type: none"><li>• 4,096 FTS Ordered-Sets in the L0s Link PM state,</li><li>• Followed by a single SKIP Ordered-Set prior to entering the L0 Link PM state,</li><li>• Finally, transmission of 1,024 TS1 Ordered-Sets in the <i>Recovery</i> state.</li></ul>			RW	Yes	0

**Register 13-29. 78h Link Status and Control**  
**(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Clock Power Management Enable</b> <i>Not supported</i>		RsvdP	No	0
9	<b>Hardware-Autonomous Width Disable</b> <i>Reserved</i>		RsvdP	No	0
10	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>Link Bandwidth Management Interrupt Enable</b> 0 = Disables interrupt generation 1 = Enables generation of an interrupt, to indicate that the Port's <b>Link Status</b> register <i>Link Bandwidth Management Status</i> bit (Downstream Ports, offset 78h[30]) has been Set	Downstream	RW	Yes	0
11	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>Link Autonomous Bandwidth Interrupt Enable</b> 0 = Disables interrupt generation 1 = Enables generation of an interrupt, to indicate that the Port's <b>Link Status</b> register <i>Link Autonomous Bandwidth Status</i> bit (Downstream Ports, offset 78h[31]) has been Set	Downstream	RW	Yes	0
15:12	<i>Reserved</i>		RsvdP	No	0h
<b>Link Status</b>					
19:16	<b>Current Link Speed</b> Indicates the negotiated Link speed of the Port's PCI Express Link. The encoding is the binary value of the bit location in the Port's <b>Link Capability 2</b> register <i>Supported Link Speeds Vector</i> field (offset 94h[7:1]) that corresponds to the current Link speed. <i>For example</i> , a value of 1h indicates that the current Link speed is that which corresponds to bit 1 in the <i>Supported Link Speeds Vector</i> field, which is 2.5 GT/s. If the Link is not up, the value of this field is undefined.		RO	No	1h
25:20	<b>Negotiated Link Width</b> Dependent upon the physical Port configuration. Link width is determined by the negotiated value with the attached Lane/Port. If the Link is not up, the value of this field is undefined.  00_0000b = Link is Down (default) 00_0001b = x1 00_0010b = x2 00_0100b = x4 00_1000b = x8 01_0000b = x16  All other encodings are <i>not supported</i> .		RO	No	00_0000b
26	<i>Reserved</i>		RsvdP	No	0
27	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>Link Training</b> 1 = Indicates that the Downstream Port requested Link training, and the Link training is in-progress or about to start	Downstream	RO	No	0

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 13-29. 78h Link Status and Control  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
28	<b>Slot Clock Configuration</b> 0 = Indicates that the PEX 8748 uses an independent clock 1 = Indicates that the PEX 8748 uses the same physical Reference Clock that the platform provides on the connector		HwInit	Yes	0
29	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>Data Link Layer Link Active</b> When Set, and the Port's <b>Link Capability</b> register <i>Data Link Layer Link Active Reporting Capable</i> bit (Downstream Ports, offset 74h[20]) is also Set, indicates the following: <ul style="list-style-type: none"> <li>Data Link Layer (DLL) is in the <i>DL_Active</i> state</li> <li>Link is operational</li> <li>Flow Control (FC) Initialization has successfully completed</li> </ul>	Downstream	RO	Yes	0
30	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>Link Bandwidth Management Status</b> Set by hardware to indicate that either of the following has occurred, without the Port transitioning through <i>DL_Down</i> status: <ul style="list-style-type: none"> <li>Link retraining has completed following a Write of 1 to the Port's <b>Link Control</b> register <i>Retrain Link</i> bit (Downstream Ports, offset 78h[5])</li> <li>Hardware has changed Link speed or width, to attempt to correct unreliable Link operation, either through an Link Training and Status State Machine (LTSSM) timeout or higher-level process</li> </ul>	Downstream	RW1C	Yes	0
31	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>Link Autonomous Bandwidth Status</b> Set by hardware to indicate that hardware has autonomously changed Link speed or width, without the Port transitioning through <i>DL_Down</i> status, for reasons other than to attempt to correct unreliable Link operation.	Downstream	RW1C	Yes	0

a. The Port Receiver must be capable of entering the L0s Link PM state, regardless of whether the state is disabled.

**Register 13-30. 7Ch Slot Capability  
(Downstream Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> For bits [6, 4:0], the default values are shown to be 1 for Downstream Ports, which is true only if the Port is Parallel or Serial Hot Plug-capable; otherwise, the default value is 0. This also applies to bit 17 for Serial Hot Plug Ports. Serial Hot Plug-capable means that the PEX 8748 has detected that an I/O Expander is present.</p> <p>Each Transparent Downstream Port can support one <b>Parallel</b> Hot Plug Controller, which uses the set of on-chip Hot Plug I/O signals designated with suffixes A, B, and C. The assignment of Parallel Hot Plug Controllers, to individual Ports, is programmed in the <b>Parallel Hot Plug Control</b> register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 3A4h[7:0, 15:8, and 23:16] for Parallel Hot Plug Controllers A, B, and C, respectively). All other Transparent Downstream Ports support a <b>Serial</b> Hot Plug Controller, which uses signals on an external I<sup>2</sup>C I/O Expander, for Hot Plug signaling.</p> <p>By default, all Transparent Downstream Ports use a <b>Serial</b> Hot Plug Controller, unless the Port's <b>Power Management Hot Plug User Configuration</b> register <b>Serial Hot Plug Override Parallel Disable</b> bit (offset F70h[19]) is Set. Ports that use a Serial Hot Plug Controller have the register's <b>Port Is Serial Hot Plug Port</b> bit (offset F70h[15]) Set.</p>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Attention Button Present</b> Set if the Port is Parallel or Serial Hot Plug-capable.  0 = Attention Button is not implemented 1 = Attention Button is implemented on this Hot Plug-capable Transparent Downstream Port's slot chassis	Downstream	RO	Yes	1
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Power Controller Present</b> Enables or disables the Hot Plug Controller on the Hot Plug-capable Transparent Downstream Ports. Set if the Port is Parallel or Serial Hot Plug-capable.  0 = Power Controller is not implemented. The Hot Plug Controller is disabled for that slot and a power-up sequence is not executed. The slot remains in the disabled state. 1 = Power Controller is implemented for the Hot Plug-capable Transparent Downstream Port's slot. The Hot Plug Controller powers up the slot when the Manually operated Retention Latch (MRL) is closed and the Port's <b>Slot Control</b> register <b>Power Controller Control</b> bit (Downstream Ports, offset 80h[10]) is Cleared. Otherwise, if bit 2 ( <b>MRL Sensor Present</b> ) is disabled (Cleared), the MRL's position has no effect on powering up the slot.	Downstream	RO	Yes	1

**Register 13-30. 7Ch Slot Capability  
(Downstream Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
2	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>MRL Sensor Present</b> Set if the Port is Parallel or Serial Hot Plug-capable. 0 = MRL Sensor is not implemented. MRL position is “Don’t Care” for that slot. 1 = MRL Sensor is implemented on this Hot Plug-capable Transparent Downstream Port’s slot chassis. The PEX 8748 senses whether the MRL is open or closed for a slot. The MRL should be Low for power-on for that slot.	Downstream	RO	Yes	1
3	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Attention Indicator Present</b> Set if the Port is Parallel or Serial Hot Plug-capable. 0 = Attention Indicator is not implemented. HP_ATNLED_x# output is not functional on the slot. 1 = Attention Indicator is implemented on this Hot Plug-capable Transparent Downstream Port’s slot chassis. Controls whether the HP_ATNLED_x# output for the slot drives out Active-Low.	Downstream	RO	Yes	1

**Register 13-30. 7Ch Slot Capability  
(Downstream Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Power Indicator Present</b> Set if the Port is Parallel or Serial Hot Plug-capable. 0 = Power Indicator is not implemented. HP_PWRLED_# output is not functional on the slot. 1 = Power Indicator is implemented on the slot chassis of this Hot Plug-capable Transparent Downstream Port. Controls whether the HP_PWRLED_# output for the slot drives out Active-Low.	Downstream	RO	Yes	1
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Hot Plug Surprise</b> 0 = No device in the Downstream Port slot is removed from the system without prior notification 1 = Device in the Downstream Port slot can be removed from the system without prior notification	Downstream	RO	Yes	0
6	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Hot Plug Capable</b> Set if the Port is Parallel or Serial Hot Plug-capable. 0 = Downstream Port slot is not capable of supporting Hot Plug operations 1 = Downstream Port slot is capable of supporting Hot Plug operations	Downstream	RO	Yes	1
14:7	<b>Reserved</b>	Upstream	RsvdP	No	00h
	<b>Slot Power Limit Value</b> The maximum power supplied by the corresponding PEX 8748 Downstream slot is determined by multiplying the value in this field (expressed in decimal; 25d = 19h) by the field [16:15] ( <i>Slot Power Limit Scale</i> ) value. This field must be implemented if the <b>PCI Express Capability</b> register <i>Slot Implemented</i> bit (offset 68h[24]) is Set (default). Serial EEPROM and/or I <sup>2</sup> C Writes to this register or a DLL Up event causes the Downstream Port to send the Set_Slot_Power_Limit Message to the device connected to it, so as to convey the Limit value to the Downstream device's Upstream Port <b>Device Capability</b> register <i>Captured Slot Power Limit Value</i> and <i>Captured Slot Power Limit Scale</i> fields.	Downstream	RO	Yes	19h

**Register 13-30. 7Ch Slot Capability  
(Downstream Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
16:15	<b>Reserved</b>	Upstream	RsvdP	No	00b
	<b>Slot Power Limit Scale</b> The maximum power supplied by the corresponding PEX 8748 Downstream slot is determined by multiplying the value in this field by the field [14:7] ( <i>Slot Power Limit Value</i> ) value. This field must be implemented if the <b>PCI Express Capability</b> register <i>Slot Implemented</i> bit (offset 68h[24]) is Set (default). Serial EEPROM and/or I <sup>2</sup> C Writes to this register or a DLL Up event causes the Downstream Port to send the Set_Slot_Power_Limit Message to the device connected to it, so as to convey the Limit value to the Downstream device's Upstream Port <b>Device Capability</b> register <i>Captured Slot Power Limit Value</i> and <i>Captured Slot Power Limit Scale</i> fields. 00b = 1.0x 01b = 0.1x 10b = 0.01x 11b = 0.001x	Downstream	RO	Yes	00b
17	<b>Reserved</b>	Upstream; Downstream non-Serial Hot Plug-enabled	RsvdP	No	0
	<b>Electromechanical Interlock Present</b> This bit is valid for Serial Hot Plug Ports that have an external I <sup>2</sup> C I/O Expander; this bit is <i>not</i> valid for Parallel Hot Plug Ports. 0 = Electromechanical Interlock is not implemented on the chassis for this slot 1 = Electromechanical Interlock is implemented on the chassis for this slot	Downstream Serial Hot Plug-enabled	RO	Yes	1
18	<b>No Command Completed Support</b> <b>Reserved</b>		RsvdP	No	0



**Register 13-30. 7Ch Slot Capability  
(Downstream Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default	
31:19	<b>Reserved</b>	Upstream	RsvdP	No	0-0h	
	<b>Physical Slot Number</b>  <i><b>Note:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</i>  Indicates the physical Slot Number attached to this Port. If the <b>PCI Express Capability</b> register <i>Slot Implemented</i> bit (offset 68h[24]) is Set (default), this field must be hardware-initialized to a value that assigns a Slot Number that is unique within the chassis, regardless of the form factor associated with the slot. Must be initialized to 0h for Ports connected to devices that are integrated on the system board. Bit usage depends upon whether the Port is Serial Hot Plug-capable.	Downstream	RO	Yes	0-0h	
	<b>Serial Hot Plug-Capable Downstream Ports</b>					
	<b>Bit(s)</b>					<b>Description/Function</b>
	23:19					Port Numbers 0, 1, 2, 3, 8, 9, 10, 11, 16, 17, 18, 19
	31:24					Loaded from I <sup>2</sup> C I/O Expander
	<b>Parallel Hot Plug-Capable Downstream Ports</b>					
	<b>Bit(s)</b>					<b>Description/Function</b>
	23:19					Port Numbers 0, 1, 2, 3, 8, 9, 10, 11, 16, 17, 18, 19
	25:24					I2C_ADDR0 input binary value (input state 0, Z, or 1, is converted to binary value 00b, 01b, or 10b, respectively)
	31:26	0000_00b				

**Register 13-31. 80h Slot Status and Control  
(Downstream Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Note:</b> <i>Virtual Switch mode only</i> – To inform software of the addition of a bridge in failover while the PEX 8748 is in Virtual Switch mode, the Upstream Port Hot Plug (Device-Specific) functionality can be enabled, by Setting the Port's <b>Power Management Hot Plug User Configuration</b> register <b>Upstream Hot Plug Enable</b> bit (Upstream Port(s), offset <b>F70h[14]</b>). This feature enables the MRL Sensor and Presence Detect-related register bits, and generates the corresponding Hot Plug interrupts, using the Upstream Port(s) copy of this register, which is usually <b>Reserved</b>.</p> <p>The Hot Plug-related bits that can be enabled for the Upstream Port in the <b>Slot Status and Control</b> register are:</p> <ul style="list-style-type: none"> <li>• <b>MRL Sensor Changed Enable</b> (bit 2)</li> <li>• <b>Presence Detect Changed Enable</b> (bit 3)</li> <li>• <b>Hot Plug Interrupt Enable</b> (bit 5)</li> <li>• <b>MRL Sensor Changed</b> (bit 18)</li> <li>• <b>Presence Detect Changed</b> (bit 19)</li> <li>• <b>MRL Sensor State</b> (bit 21)</li> <li>• <b>Presence Detect State</b> (bit 22)</li> </ul>					
<b>Slot Control</b>					
<p><b>Note:</b> To change the values of the <b>MRL Sensor Changed Enable</b>, <b>Power Fault Detector Enable</b>, and/or <b>Attention Button Pressed Enable</b> bits (bits [2:0], respectively), the corresponding bit(s) in the <b>Slot Capability</b> register (Downstream Ports, offset <b>7Ch[2:0]</b>, respectively) must be Set first.</p>					
0	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Attention Button Pressed Enable</b> 0 = Function is disabled 1 = Enables software notification with an interrupt if the Port is in the D0 state (Port's <b>PCI Power Management Status and Control</b> register <b>Power State</b> field, offset <b>44h[1:0]</b> , is Cleared), or with a PME Message if the Port is in the D3hot state (the <b>Power State</b> field bits are both Set), for an <b>Attention Button Pressed</b> event on this Hot Plug-capable Transparent Downstream Port	Offset <b>7Ch[0]=0</b> Downstream	RO	No	0
		Offset <b>7Ch[0]=1</b> Downstream	RW	Yes	0
1	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Power Fault Detector Enable</b> 0 = Function is disabled 1 = Enables software notification with an interrupt if the Port is in the D0 state (Port's <b>PCI Power Management Status and Control</b> register <b>Power State</b> field, offset <b>44h[1:0]</b> , is Cleared), or with a PME Message if the Port is in the D3hot state (the <b>Power State</b> field bits are both Set), for a <b>Power Fault Detected</b> event on this Hot Plug-capable Transparent Downstream Port	Offset <b>7Ch[1]=0</b> Downstream	RO	No	0
		Offset <b>7Ch[1]=1</b> Downstream	RW	Yes	0

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

**Register 13-31. 80h Slot Status and Control  
(Downstream Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
2	<b>Base Mode</b> <i>Reserved</i>	Upstream	RsvdP	No	0
	<b>Virtual Switch Mode</b> <b>MRL Sensor Changed Enable</b> Enabled for Upstream Port(s) when the Port's <b>Power Management Hot Plug User Configuration</b> register <i>Upstream Hot Plug Enable</i> bit (Upstream Port(s), offset F70h[14]) is Set.  0 = Function is disabled 1 = Enables software notification with an interrupt if the Port is in the D0 state (Port's <b>PCI Power Management Status and Control</b> register <i>Power State</i> field, offset 44h[1:0], is Cleared), or with a PME Message if the Port is in the D3hot state (the <i>Power State</i> field bits are both Set), for an <b>MRL Sensor Changed</b> event on this Upstream Port	Upstream	RO	No	0
		Upstream	RW	Yes	0
	<b>Base and Virtual Switch Modes</b> 0 = Function is disabled 1 = Enables software notification with an interrupt if the Port is in the D0 state (Port's <b>PCI Power Management Status and Control</b> register <i>Power State</i> field, offset 44h[1:0], is Cleared), or with a PME Message if the Port is in the D3hot state (the <i>Power State</i> field bits are both Set), for an <b>MRL Sensor Changed</b> event on this Hot Plug-capable Transparent Downstream Port	Downstream	RO	No	0
		Downstream	RW	Yes	0

**Register 13-31. 80h Slot Status and Control  
(Downstream Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
3	<b>Base Mode</b> <i>Not valid</i> Offset F70h[14]=0	Upstream	RsvdP	No	0
	<b>Virtual Switch Mode</b> <b>Presence Detect Changed Enable</b> Enabled for Upstream Port(s) when the Port's <b>Power Management Hot Plug User Configuration</b> register <i>Upstream Hot Plug Enable</i> bit (Upstream Port(s), offset F70h[14]) is Set. A <b>Presence Detect Changed</b> event is triggered by either the Upstream Port(s)' SerDes Receiver Detect ( <b>Physical Layer Receiver Detect Status</b> register <i>Receiver Detected on Lane x</i> bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 200h[31:16])), or by HP_PRSNT_x# input or Serial Hot Plug PRSNT# (SHP_PRSNTx#) input (from external I <sup>2</sup> C I/O Expander) on this Upstream Port. 0 = Function is disabled 1 = Enables software notification with an interrupt if the Port is in the D0 state (Port's <b>PCI Power Management Status and Control</b> register <i>Power State</i> field, offset 44h[1:0], is Cleared), or with a PME Message if the Port is in the D3hot state (the <i>Power State</i> field bits are both Set), for a Presence Detect Changed event on the Upstream Port	Upstream	RW	Yes	0
	<b>Base and Virtual Switch Modes</b> A <b>Presence Detect Changed</b> event is triggered by either the Downstream Port's SerDes Receiver Detect ( <b>Physical Layer Receiver Detect Status</b> register <i>Receiver Detected on Lane x</i> bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 200h[31:16])), or by HP_PRSNT_x# input or Serial Hot Plug PRSNT# input (from external I <sup>2</sup> C I/O Expander) on this Hot Plug-capable Transparent Downstream Port. 0 = Function is disabled 1 = Enables software notification with an interrupt if the Port is in the D0 state (Port's <b>PCI Power Management Status and Control</b> register <i>Power State</i> field, offset 44h[1:0], is Cleared), or with a PME Message if the Port is in the D3hot state (the <i>Power State</i> field bits are both Set), for a Presence Detect Changed event on the Downstream Port	Downstream	RW	Yes	0

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

**Register 13-31. 80h Slot Status and Control  
(Downstream Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Command Completed Interrupt Enable</b> 0 = Function is disabled 1 = Enables software notification with an interrupt when a command is completed by the Hot Plug Controller on this Hot Plug-capable Transparent Downstream Port	Downstream	RW	Yes	0
5	<b>Base Mode</b> <b>Reserved</b> Offset F70h[14]=0	Upstream	RsvdP	No	0
	<b>Virtual Switch Mode</b> <b>Hot Plug Interrupt Enable</b> Enabled for Upstream Port(s) when the Port's <b>Power Management Hot Plug User Configuration</b> register <i>Upstream Hot Plug Enable</i> bit (Upstream Port(s), offset F70h[14]) is Set. 0 = Function is disabled 1 = Enables an interrupt on enabled Hot Plug/Link State events for Upstream Port(s)	Upstream	RW	Yes	0
	<b>Base and Virtual Switch Modes</b> 0 = Function is disabled 1 = Enables an interrupt on enabled Hot Plug/Link State events for the Downstream Port	Downstream	RW	Yes	0
7:6	<b>Reserved</b>	Upstream	RsvdP	No	00b
	<b>Attention Indicator Control</b> Controls the Downstream Port's slot's Attention Indicator. Reads return the Hot Plug-capable Transparent Downstream Port's Attention Indicator's current state. Writing a non-zero value triggers a <b>Command Completed</b> event (even if the value written is the same as the existing value). Writing 00b preserves the current value and does not trigger a Command Completed event. 00b = <b>Reserved</b> – Writes are ignored 01b = Turns On indicator to constant On state 10b = Causes indicator to blink 11b = Turns Off indicator	Downstream	RW	Yes	11b

**Register 13-31. 80h Slot Status and Control  
(Downstream Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
9:8	<b>Reserved</b>	Upstream	RsvdP	No	00b
	<b>Power Indicator Control</b> Controls the Hot Plug-capable Transparent Downstream Port's slot's Power Indicator. Reads return this Hot Plug-capable Transparent Downstream Port's Power Indicator's current state. Writing a non-zero value triggers a <b>Command Completed</b> event (even if the value written is the same as the existing value). Writing 00b preserves the current value and does not trigger a Command Completed event.  00b = <b>Reserved</b> – Writes are ignored 01b = Turns On indicator to constant On state 10b = Causes indicator to blink 11b = Turns Off indicator	Downstream	RW	Yes	11b (MRL open) 01b (MRL closed)
10	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Power Controller Control</b> Controls the Hot Plug-capable Transparent Downstream Port's slot's Power Controller.  0 = Turns On the Power Controller; requires some delay to be effective 1 = Turns Off the Power Controller	Downstream	RW	Yes	1 (MRL open) 0 (MRL closed)
11	<b>Reserved</b>	Upstream; Downstream non-Serial Hot Plug-enabled	RsvdP	No	0
	<b>Electromechanical Interlock Control</b> This bit is valid for Serial Hot Plug Ports that have an external I <sup>2</sup> C I/O Expander; this bit is <b>not</b> valid for Parallel Hot Plug Ports.  If an Electromechanical Interlock is implemented, writing 1 to this bit causes the state of the interlock to toggle. A Write of 0 to this bit has no effect. A Read of this bit always returns a value of 0.	Downstream Serial Hot Plug-enabled	RW	Yes	0
12	<b>Not valid</b>	Upstream	RsvdP	No	0
	<b>Data Link Layer State Changed Enable</b> Enables software notification with an interrupt if the Port is in the D0 state (Port's <b>PCI Power Management Status and Control</b> register <b>Power State</b> field, offset 44h[1:0], is Cleared), or with a PME Message if the Port is in the D3hot state (the <b>Power State</b> field bits are both Set), when the Port's <b>Link Status</b> register <b>Data Link Layer Link Active</b> bit (Downstream Ports, offset 78h[29]) is changed.	Downstream	RW	Yes	0
15:13	<b>Reserved</b>		RsvdP	No	000b

**Register 13-31. 80h Slot Status and Control  
(Downstream Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Slot Status</b>					
16	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Attention Button Pressed</b> 1 = Hot Plug-capable Transparent Downstream Port's slot's Attention Button was pressed	Downstream	RW1C	Yes	0
17	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Power Fault Detected</b> 1 = Hot Plug-capable Transparent Downstream Port's slot's Power Controller detected a Power Fault at the slot	Downstream	RW1C	Yes	0
18	<b>Base Mode</b> <b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Virtual Switch Mode</b> <b>MRL Sensor Changed</b> Enabled for Upstream Port(s) when the Port's <b>Power Management Hot Plug User Configuration</b> register <i>Upstream Hot Plug Enable</i> bit (Upstream Port(s), offset F70h[14]) is Set.  1 = Hot Plug-capable Transparent Upstream Port's slot detected an MRL Sensor state change	Upstream	RW1C	Yes	0
	<b>Base and Virtual Switch Modes</b> 1 = Hot Plug-capable Transparent Downstream Port's slot detected an MRL Sensor state change	Downstream	RW1C	Yes	0

**Register 13-31. 80h Slot Status and Control  
(Downstream Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
19	<b>Base Mode</b> <i>Reserved</i> Offset F70h[14]=0	Upstream	RsvdP	No	0
	<b>Virtual Switch Mode</b> <b>Presence Detect Changed</b> Enabled for Upstream Port(s) when the Port's <b>Power Management Hot Plug User Configuration</b> register <i>Upstream Hot Plug Enable</i> bit (Upstream Port(s), offset F70h[14]) is Set. A Presence Detect Changed event is triggered by either the Upstream Port's SerDes Receiver Detect ( <b>Physical Layer Receiver Detect Status</b> register <i>Receiver Detected on Lane x</i> bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 200h[31:16])), or by HP_PRSENT_x# or Hot Plug-capable Transparent Downstream Port's Serial Hot Plug <b>PRSNT#</b> (SHP_PRSENTx#) input (from external I <sup>2</sup> C I/O Expander). Write 1 to Clear. 1 = Value reported in bit 22 ( <i>Presence Detect State</i> ) changed Offset F70h[14]=1	Upstream	RW1C	Yes	0
	<b>Base and Virtual Switch Modes</b> A Presence Detect Changed event is triggered by either the Downstream Port's SerDes Receiver Detect ( <b>Physical Layer Receiver Detect Status</b> register <i>Receiver Detected on Lane x</i> bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 200h[31:16])), or by HP_PRSENT_x# or Hot Plug-capable Transparent Downstream Port's Serial Hot Plug <b>PRSNT#</b> (SHP_PRSENTx#) input (from external I <sup>2</sup> C I/O Expander). Write 1 to Clear. 1 = Value reported in bit 22 ( <i>Presence Detect State</i> ) changed	Downstream	RW1C	Yes	0



**Register 13-31. 80h Slot Status and Control  
(Downstream Ports) (Cont.)**

Bit(s)	Description		Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Reserved</b>		Upstream	RsvdP	No	0
	<b>Command Completed</b> 1 = Hot Plug-capable Transparent Downstream Port's slot's Hot Plug Controller completed an issued command to: <ul style="list-style-type: none"><li>• <i>Attention Indicator Control</i> (field [7:6])</li><li>• <i>Power Indicator Control</i> (field [9:8])</li><li>• <i>Power Controller Control</i> (bit 10)</li><li>• <i>Electromechanical Interlock Control</i> (bit 11) (Serial Hot Plug-enabled Ports only)</li></ul>		Downstream	RW1C	Yes	0
21	<b>Base Mode</b>	Offset F70h[14]=0	Upstream	RsvdP	No	0
	<b>Reserved</b>					
	<b>Virtual Switch Mode</b> <b>MRL Sensor State</b> Enabled for Upstream Port(s) when the Port's <b>Power Management Hot Plug User Configuration</b> register <i>Upstream Hot Plug Enable</i> bit (Upstream Port(s), offset F70h[14]) is Set. Reveals the Hot Plug-capable Transparent Upstream Port's MRL Sensor's current state.  0 = MRL Sensor is closed 1 = MRL Sensor is open	Offset F70h[14]=1	Upstream	RO	No	0
	<b>Base and Virtual Switch Modes</b> Reveals the Hot Plug-capable Transparent Downstream Port's MRL Sensor's current state.  0 = MRL Sensor is closed 1 = MRL Sensor is open		Downstream	RO	No	0

**Register 13-31. 80h Slot Status and Control  
(Downstream Ports) (Cont.)**

Bit(s)	Description		Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
22	<b>Base Mode</b> <i>Not valid</i>	Offset F70h[14]=0	Upstream	RsvdP	No	0
	<b>Virtual Switch Mode Presence Detect State</b> For Upstream Port(s) that implement slots, indicates the presence of an adapter in the slot, reflected by the logical OR of the Upstream Port's SerDes Receiver Detect ( <b>Physical Layer Receiver Detect Status</b> register <i>Receiver Detected on Lane x</i> bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 200h[31:16])), and, if present, the Port's HP_PRSNT_x# input (de-bounced) or Serial Hot Plug PRSNT# (SHP_PRSNTx#) input (from external I <sup>2</sup> C I/O Expander) for the Serial Hot Plug-enabled Port. Hardwired to 1 when the Port's <b>PCI Express Capability</b> register <i>Slot Implemented</i> bit (offset 68h[24]) value is 0. 0 = Slot is empty, or device is not present 1 = Slot is occupied, or device is present	Offset 68h[24]=1 and Offset F70h[14]=1	Upstream	RO	No	0
		Offset 68h[24]=0 and Offset F70h[14]=1	Upstream	RO	No	1
	<b>Base and Virtual Switch Modes</b> For Downstream Ports that implement slots, indicates the presence of an adapter in the slot, reflected by the logical OR of the Downstream Port's SerDes Receiver Detect ( <b>Physical Layer Receiver Detect Status</b> register <i>Receiver Detected on Lane x</i> bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 200h[31:16])), and, if present, the Port's HP_PRSNT_x# input (de-bounced) or Serial Hot Plug PRSNT# (SHP_PRSNTx#) input (from external I <sup>2</sup> C I/O Expander) for the Serial Hot Plug-enabled Port. Hardwired to 1 when the Port's <b>PCI Express Capability</b> register <i>Slot Implemented</i> bit (offset 68h[24]) value is 0. 0 = Slot is empty, or device is not present 1 = Slot is occupied, or device is present	Offset 68h[24]=1	Downstream	RO	No	0
		Offset 68h[24]=0	Downstream	RO	No	1

**Register 13-31. 80h Slot Status and Control  
(Downstream Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
23	<b>Reserved</b>	Upstream; Downstream non-Serial Hot Plug-enabled	RsvdZ	No	0
	<b>Electromechanical Interlock Status</b> This bit is valid for Serial Hot Plug Ports that have an external I <sup>2</sup> C I/O Expander; this bit is <i>not</i> valid for Parallel Hot Plug Ports. When an Electromechanical Interlock is implemented, indicates the Electromechanical Interlock's current status. 0 = Electromechanical Interlock is disengaged 1 = Electromechanical Interlock is engaged	Downstream Serial Hot Plug-enabled	RW1C	Yes	0
24	<b>Not valid</b>	Upstream	RsvdP	No	0
	<b>Data Link Layer State Changed</b> In response to a <a href="#">Data Link Layer State Changed</a> event, software must read the Port's <a href="#">Link Status</a> register <a href="#">Data Link Layer Link Active</a> bit (Downstream Ports, offset 78h[29]), to determine whether the Link is active before initiating Configuration Requests to the device. 1 = Value reported in the <a href="#">Link Status</a> register <a href="#">Data Link Layer Link Active</a> bit changed	Downstream	RW1C	Yes	0
31:25	<b>Reserved</b>		RsvdZ	No	0-0h

**Register 13-32. 8Ch Device Capability 2  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
4:0	<i>Reserved</i>		RsvdP	No	0-0h
5	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ARI Forwarding Supported</b> 0 = Alternative Routing-ID Interpretation (ARI) forwarding is not supported 1 = ARI forwarding is supported	Downstream	RO	Yes	1
6	<b>AtomicOp Routing Supported</b>		RO	Yes	1
10:7	<i>Reserved</i>		RsvdP	No	0h
11	<b>LTR Mechanism Supported</b> 0 = Latency Tolerance Reporting (LTR) mechanism is not supported 1 = LTR mechanism is supported		RO	Yes	1
17:12	<i>Reserved</i>		RsvdP	No	0-0h
18	<b>OBFF Mechanism Supported Bit 0</b> Indicates whether the Optimized Buffer Flush/Fill (OBFF) mechanism is supported. 0 = OBFF is not supported 1 = OBFF is supported, using Message signaling only		RO	Yes	1
31:19	<i>Reserved</i>		RsvdP	No	0-0h

**NDA CONFIDENTIAL -- Inventec | King Michael**

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

**Register 13-33. 90h Device Status and Control 2  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Control 2</b>					
4:0	<i>Reserved</i>		RsvdP	No	0-0h
5	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ARI Forwarding Enable</b> 0 = Disabled 1 = Enabled; Downstream Port disables its traditional Device Number field from being forced to 0 when turning a Type 1 Configuration Request into a Type 0 Configuration Request, permitting access to extended functions in an ARI device immediately below the Port	Downstream	RW	Yes	0
6	<i>Reserved</i>		RsvdP	No	0
7	<b>AtomicOp Egress Blocking</b>		RW	Yes	0
9:8	<i>Reserved</i>		RsvdP	No	00b
10	<b>LTR Mechanism Enable</b> 0 = Disables LTR mechanism 1 = Enables LTR mechanism		RW	Yes	0
12:11	<i>Reserved</i>		RsvdP	No	00b
14:13	<b>OBFF Enable</b> 00b = Disabled 01b = Enabled using Message Signaling [Variation A] 10b = Enabled using Message Signaling [Variation B] 11b = <i>Reserved</i>		RW	Yes	00b
15	<i>Reserved</i>		RsvdP	No	0
<b>Device Status 2</b>					
31:16	<i>Reserved</i>		RsvdP	No	0000h

**Register 13-34. 94h Link Capability 2  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
0	<i>Reserved</i>	RsvdP	No	0	
7:1	<b>Supported Link Speeds Vector</b> Supports 2.5, 5.0, and 8.0 GT/s Link speeds. Indicates the supported Link speed(s) of the associated Port. For each bit, a value of 1 indicates that the corresponding Link speed is supported; otherwise, the Link speed is <i><b>not supported</b></i> .	RO	Yes	07h	
	<b>Bit(s)</b>				<b>Description/Function</b>
	1				2.5 GT/s (Gen 1)
	2				5.0 GT/s (Gen 2)
	3				8.0 GT/s (Gen 3)
	7:4				<i>Reserved</i>
8	<b>Cross-Link Supported</b> 0 (Ports that support Link speeds of 8.0 GT/s or higher) = Indicates that the associated Port does not support cross-links 0 (Ports that support only Link speeds of 2.5 or 5.0 GT/s) = Does not provide information regarding the Port’s level of cross-link support 1 = Indicates that the associated Port supports cross-links  <i>Notes: It is recommended that this bit be Set in any Port that supports cross-links, even though doing so is only required for Ports that also support operating at 8.0 GT/s or higher Link speeds.</i>  <i>Software should use this bit when referencing fields whose definition depends upon whether the Port supports cross-links.</i>	RO	Yes	1	
31:9	<i>Reserved</i>	RsvdP	No	0-0h	

**Register 13-35. 98h Link Status and Control 2  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Link Control 2</b>					
3:0	<b>Target Link Speed</b> Sets the Target Compliance mode speed when software is using bit 4 ( <i>Enter Compliance</i> ) to force a Link into Compliance mode.	Upstream	RWS	Yes	3h
	Sets an upper limit on Link operational speed, by restricting the values advertised by the Upstream component in its training sequences. Also Sets the Target Compliance mode speed when software is using bit 4 ( <i>Enter Compliance</i> ) to force a Link into Compliance mode. The encoding is the binary value of the bit location in the Port's <b>Link Capability 2</b> register <i>Supported Link Speeds Vector</i> field (offset 94h[7:1]) that corresponds to the target Link speed. <i>For example</i> , a value of 3h indicates that the current Link speed is that which corresponds to bit 3 in the <i>Supported Link Speeds Vector</i> field, which is 8.0 GT/s. If a value written to this field does not correspond to a supported speed (as indicated by the Port's <b>Link Capability</b> register <i>Maximum Link Speed</i> field (offset 74h[3:0])), the value of this field is undefined. The default value of this field is the highest supported Link speed (as reported in the Port's <i>Maximum Link Speed</i> field); however, the default value for all Ports can be globally changed to 2h (Gen 2 maximum speed) or 1h (Gen 1 maximum speed), according to the STRAP_GEN1_GEN2 input state (Z = 2h (Gen 2), or 0 = 1h (Gen 1)).	Downstream	RWS	Yes	3h

**Register 13-35. 98h Link Status and Control 2**  
**(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>Enter Compliance</b> Software is permitted to force a Link to enter Compliance mode at the speed indicated in field [3:0] ( <i>Target Link Speed</i> ), by Setting this bit in both components on a Link, and then initiating a Hot Reset on the Link. The default value of this bit following a Fundamental Reset is 0.		RWS	Yes	0
5	<b>Hardware-Autonomous Speed Disable</b> <i>Reserved</i> Initial transition to the highest supported common Link speed is not blocked by this bit.		RsvdP	No	0
6	<i>Not valid</i>	Upstream	RsvdP	Yes	0
	<b>Selectable De-Emphasis</b> Selects the standard de-emphasis level when the Link is operating at 5.0 GT/s. When the Link is operating at 2.5 GT/s, the Setting of this bit has no effect (de-emphasis at 2.5 GT/s is -3.5 dB). 0 = -6 dB (Link is operating at 5.0 GT/s) 1 = -3.5 dB (Link is operating at 2.5 GT/s)	Downstream	HwInit	Yes	0 (5.0 GT/s) 1 (2.5 GT/s)
9:7	<b>Transmit Margin</b> Intended for debug and compliance testing only.		RWS	Yes	000b
10	<b>Enter Modified Compliance</b> Intended for debug and compliance testing only.		RWS	Yes	0
11	<b>Compliance SOS</b> <i>This bit is applicable only when the Link is operating at 2.5 or 5.0 GT/s.</i> 1 = LTSSM must periodically send SKIP Ordered-Sets between sequences when sending the Compliance Pattern or Modified Compliance Pattern		RWS	Yes	0



**Register 13-35. 98h Link Status and Control 2**  
**(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:12	<b>Compliance Preset/De-Emphasis</b> Intended for debug and compliance testing only.  <b>8.0 GT/s Data Rate</b> Sets the Transmitter Preset in the <i>Polling.Compliance</i> substate, if the entry occurred due to bit 4 ( <i>Enter Compliance</i> ) being Set. The encodings are defined in the <i>PCI Express Base r3.0</i> , Section 4.2.3.2, “Encoding of Presets.” Results are undefined if a <b>Reserved</b> preset encoding is used when entering the <i>Polling.Compliance</i> substate in this way.  <b>5.0 GT/s Data Rate</b> Sets the de-emphasis level in the <i>Polling.Compliance</i> substate, if the entry occurred due to bit 4 ( <i>Enter Compliance</i> ) being Set.  0h = -6 dB (Link is operating at 5.0 GT/s) 1h = -3.5 dB (Link is operating at 2.5 GT/s)  <b>2.5 GT/s Data Rate</b> The programming of this field has no effect.		RWS	Yes	0h
<b>Link Status 2</b>					
16	<b>Current De-Emphasis Level</b> Reflects the de-emphasis level when the Link is operating at 5.0 GT/s. 0 = -6 dB (Link is operating at 5.0 GT/s) 1 = -3.5 dB (Link is operating at either 2.5 or 5.0 GT/s)		RO	Yes	0
17	<b>Equalization Complete</b> 1 = Indicates that the Transmitter Equalization procedure has completed		ROS	Yes	0
18	<b>Equalization Phase 1 Successful</b> 1 = Indicates that Phase 1 of the Transmitter Equalization procedure has successfully completed		ROS	Yes	0
19	<b>Equalization Phase 2 Successful</b> 1 = Indicates that Phase 2 of the Transmitter Equalization procedure has successfully completed		ROS	Yes	0
20	<b>Equalization Phase 3 Successful</b> 1 = Indicates that Phase 3 of the Transmitter Equalization procedure has successfully completed		ROS	Yes	0
21	<b>Link Equalization Request</b> Set by hardware, to request the Link equalization process to be performed on the Link.		RW1C	Yes	0
31:22	<b>Reserved</b>		RsvdP	No	0-0h

## 13.11 Subsystem ID and Subsystem Vendor ID Capability Registers (Offsets A4h – FCh)

This section details the Subsystem ID and Subsystem Vendor ID Capability registers. [Table 13-11](#) defines the register map.

**Table 13-11. Subsystem ID and Subsystem Vendor ID Capability Register Map (All Ports)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
<i>Reserved</i>	Next Capability Pointer (00h)	SSID/SSVID Capability ID (0Dh) A4h
Subsystem ID	Subsystem Vendor ID A8h	
<i>Reserved</i>		ACh – FCh

**Register 13-36. A4h Subsystem Capability (All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>SSID/SSVID Capability ID</b> SSID/SSVID registers for the PCI-to-PCI bridge. Program to 0Dh, as required by the <i>PCI-to-PCI Bridge r1.2</i> .	RO	Yes	0Dh
15:8	<b>Next Capability Pointer</b> 00h = This capability is the last capability in the PEX 8748 Port's Capabilities list The PEX 8748 Extended Capabilities list starts at offset 100h.	RO	Yes	00h
31:16	<i>Reserved</i>	RsvdP	No	0000h

**Register 13-37. A8h Subsystem ID and Subsystem Vendor ID (All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Subsystem Vendor ID</b> The <a href="#">Vendor ID</a> (offset 00h[15:0]) identifies the manufacturer of the PEX 8748, and the Subsystem Vendor ID optionally identifies the board or system vendor. As with the Vendor ID value, the Subsystem Vendor ID value must be a valid PCI-SIG-assigned Vendor ID. The value of this field is usually identical for all PEX 8748 Transparent Ports.	RO	Yes	10B5h
31:16	<b>Subsystem ID</b> The <a href="#">Device ID</a> (offset 00h[31:16]) identifies the PEX 8748, and optionally the Subsystem ID in combination with the Subsystem Vendor ID, uniquely identifies the board or system. The value of this field is usually identical for all PEX 8748 Ports, and is chosen or assigned only by the “owner” of the valid Vendor ID value used for the Subsystem Vendor ID. If the board or system vendor is not a PCI-SIG member, PLX can assign, free of charge, a unique Subsystem ID value, in which case the Subsystem Vendor ID remains the PLX default value, 10B5h.	RO	Yes	8748h

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

This section details the Device Serial Number Extended Capability registers. [Table 13-12](#) defines the register map.

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16                      15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

Next Capability Offset (FB4h)	Capability Version (1h)	PCI Express Extended Capability ID (0003h)	100h
Serial Number (Lower DW)			104h
Serial Number (Upper DW)			108h

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>PCI Express Extended Capability ID</b> Program to 0003h, as required by the <i>PCI Express Base r3.0</i> .	RO	Yes	0003h
19:16	<b>Capability Version</b> Program to 1h, as required by the <i>PCI Express Base r3.0</i> .	RO	Yes	1h
31:20	<b>Next Capability Offset</b> Program to FB4h, which addresses the <b>Advanced Error Reporting Extended Capability</b> structure.	RO	Yes	FB4h

**Register 13-39. 104h Serial Number (Lower DW)**  
**(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>PCI Express Device Serial Number (1st DW)</b> Lower half of a 64-bit register. Value programmed by Serial EEPROM register initialization. Per the <i>PCI Express Base r3.0</i> , all switch Ports must contain the same value; therefore, one physical register is shared by all PEX 8748 Ports. The Serial Number registers contain the IEEE-defined 64-bit Extended Unique Identifier (EUI-64 <sup>TM</sup> ), of which the upper 24 bits are the Company ID value (00_0EDFh) assigned by the IEEE Registration Authority, and the lower 40 bits are the Extension ID assigned by the identified Company (PLX). The most through least significant bytes are contained within the lowest through highest Byte addresses, respectively.	RO	Yes	B5DF_0E00h

**Register 13-40. 108h Serial Number (Upper DW)**  
**(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>PCI Express Device Serial Number (2nd DW)</b> Upper half of a 64-bit register. Value programmed by Serial EEPROM register initialization. Per the <i>PCI Express Base r3.0</i> , all switch Ports must contain the same value; therefore, one physical register is shared by all PEX 8748 Ports. The Serial Number registers contain the IEEE-defined 64-bit Extended Unique Identifier (EUI-64), of which the upper 24 bits are the Company ID value (00_0EDFh) assigned by the IEEE Registration Authority, and the lower 40 bits are the Extension ID assigned by the identified Company (PLX). The most through least significant bytes are contained within the lowest through highest Byte addresses, respectively.	RO	Yes	Silicon Revision BA: BA_8700_10h  Silicon Revision CA: CA_8700_10h

This section details the Secondary PCI Express Extended Capability registers. [Table 13-13](#) defines the register map.

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16                      15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

Next Capability Offset (148h)	Capability Version (1h)	PCI Express Extended Capability ID (0019h)	10Ch
Link Control 3			110h
<i>Reserved</i>		Lane Error Status	114h
Lane 1, 17, or 33 Equalization Control		Lane 0, 16, or 32 Equalization Control	118h
Lane 3, 19, or 35 Equalization Control		Lane 2, 18, or 34 Equalization Control	11Ch
Lane 5, 21, or 37 Equalization Control		Lane 4, 20, or 36 Equalization Control	120h
Lane 7, 23, or 39 Equalization Control		Lane 6, 22, or 38 Equalization Control	124h
Lane 9, 25, or 41 Equalization Control		Lane 8, 24, or 40 Equalization Control	128h
Lane 11, 27, or 43 Equalization Control		Lane 10, 26, or 42 Equalization Control	12Ch
Lane 13, 29, or 45 Equalization Control		Lane 12, 28, or 44 Equalization Control	130h
Lane 15, 31, or 47 Equalization Control		Lane 14, 30, or 46 Equalization Control	134h

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>PCI Express Extended Capability ID</b> Program to 0019h, as required by the <i>PCI Express Base r3.0</i> .	RO	Yes	0019h
19:16	<b>Capability Version</b> Program to 1h, as required by the <i>PCI Express Base r3.0</i> .	RO	Yes	1h
31:20	<b>Next Capability Offset</b> Program to 148h, which addresses the <b>Virtual Channel Extended Capability</b> structure.	RO	Yes	148h

**Register 13-42. 110h Link Control 3  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Reserved</b> if the Port's <b>Link Capability 2</b> register <b>Cross-Link Supported</b> bit (offset 94h[8]) is Cleared.	Upstream	RsvdP	No	0
	<b>Perform Equalization</b> Writable if the Port's <b>Link Capability 2</b> register <b>Cross-Link Supported</b> bit (offset 94h[8]) is programmed to a non-zero value.  1 = When 1 is written to the Port's <b>Link Control</b> register <b>Retrain Link</b> bit (Downstream Ports, offset 78h[5]), with the Port's <b>Link Control 2</b> register <b>Target Link Speed</b> field (offset 98h[3:0]) programmed to 8.0 GT/s, the Downstream Port must perform Transmitter Equalization	Upstream/ Downstream	RW	Yes	0
1	<b>Reserved</b> if the Port's <b>Link Capability 2</b> register <b>Cross-Link Supported</b> bit (offset 94h[8]) is Cleared.	Upstream	RsvdP	No	0
	<b>Link Equalization Request Interrupt Enable</b> Writable if the Port's <b>Link Capability 2</b> register <b>Cross-Link Supported</b> bit (offset 94h[8]) is programmed to a non-zero value.  1 = Enables the generation of an interrupt, to indicate that the Port's <b>Link Status 2</b> register <b>Link Equalization Request</b> bit (offset 98h[21]) has been Set	Upstream/ Downstream	RW	Yes	0
31:2	<b>Reserved</b>		RsvdP	No	0-0h

**Register 13-43. 114h Lane Error Status  
(All Ports)**

Bit(s)	Description	Lane x	Type	Serial EEPROM and I <sup>2</sup> C	Default
Bits [15:0] are intended for debug purposes only. <i>Notes:</i> Except for Ports that are configured as x16, the bits in this register reflect status for all Ports within the Station, rather than the individual Port associated with the Lane. The <b>Lane x</b> column is provided to indicate the Lane Number associated with each bit. Lane 0 is associated with Station 0, Lane 16 is associated with Station 1, and so forth. For further details regarding the Lane/Station relationship, refer to <a href="#">Table 13-5</a> .					
0	<b>Lane x Error Status</b> 0 = No Lane-based error is detected on the Lane 1 = Lane-based error is detected on the Lane	0, 16, or 32	RW1C	No	0
1		1, 17, or 33	RW1C	No	0
2		2, 18, or 34	RW1C	No	0
3		3, 19, or 35	RW1C	No	0
4		4, 20, or 36	RW1C	No	0
5		5, 21, or 37	RW1C	No	0
6		6, 22, or 38	RW1C	No	0
7		7, 23, or 39	RW1C	No	0
8		8, 24, or 40	RW1C	No	0
9		9, 25, or 41	RW1C	No	0
10		10, 26, or 42	RW1C	No	0
11		11, 27, or 43	RW1C	No	0
12		12, 28, or 44	RW1C	No	0
13		13, 29, or 45	RW1C	No	0
14		14, 30, or 46	RW1C	No	0
15		15, 31, or 47	RW1C	No	0
31:16	<b>Reserved</b>		RsvdP	No	0000h

**Register 13-44. 118h Lane 0, 16, or 32 and 1, 17, or 33 Equalization Control (All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 0, 16, or 32 Equalization Control</b>					
3:0	<i>Reserved</i>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <b>BC8h</b> ).	Downstream	ROS	Yes	0h
6:4	<i>Reserved</i>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
7	<i>Reserved</i>		RsvdP	No	0



**Register 13-44. 118h Lane 0, 16, or 32 and 1, 17, or 33 Equalization Control (All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
11:8	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
14:12	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
15	<b>Reserved</b>		RsvdP	No	0

**Register 13-44. 118h Lane 0, 16, or 32 and 1, 17, or 33 Equalization Control (All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 1, 17, or 33 Equalization Control</b>					
19:16	<b>Reserved</b>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <a href="#">User Lane Equalization Control</a> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <a href="#">BC8h</a> ).	Downstream	ROS	Yes	0h
22:20	<b>Reserved</b>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
23	<b>Reserved</b>		RsvdP	No	0
27:24	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <a href="#">Link Control 2</a> register <i>Transmit Margin</i> field (offset <a href="#">98h[9:7]</a> ) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <a href="#">User Lane Equalization Control</a> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <a href="#">BC8h</a> ).	Downstream	ROS	Yes	0h
30:28	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <a href="#">Link Control 2</a> register <i>Transmit Margin</i> field (offset <a href="#">98h[9:7]</a> ) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
31	<b>Reserved</b>		RsvdP	No	0

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

**Register 13-45. 11Ch Lane 2, 18, or 34 and 3, 19, or 35 Equalization Control (All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 2, 18, or 34 Equalization Control</b>					
3:0	<i>Reserved</i>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <a href="#">User Lane Equalization Control</a> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <a href="#">BC8h</a> ).	Downstream	ROS	Yes	0h
6:4	<i>Reserved</i>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
7	<i>Reserved</i>		RsvdP	No	0

**Register 13-45. 11Ch Lane 2, 18, or 34 and 3, 19, or 35 Equalization Control  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
11:8	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
14:12	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
15	<b>Reserved</b>		RsvdP	No	0

**Register 13-45. 11Ch Lane 2, 18, or 34 and 3, 19, or 35 Equalization Control  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 3, 19, or 35 Equalization Control</b>					
19:16	<b>Reserved</b>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
22:20	<b>Reserved</b>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
23	<b>Reserved</b>		RsvdP	No	0
27:24	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
30:28	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
31	<b>Reserved</b>		RsvdP	No	0

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 13-46. 120h Lane 4, 20, or 36 and 5, 21, or 37 Equalization Control (All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 4, 20, or 36 Equalization Control</b>					
3:0	<b>Reserved</b>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <b>BC8h</b> ).	Downstream	ROS	Yes	0h
6:4	<b>Reserved</b>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
7	<b>Reserved</b>		RsvdP	No	0

**Register 13-46. 120h Lane 4, 20, or 36 and 5, 21, or 37 Equalization Control (All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
11:8	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
14:12	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
15	<b>Reserved</b>		RsvdP	No	0

**Register 13-46. 120h Lane 4, 20, or 36 and 5, 21, or 37 Equalization Control (All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 5, 21, or 37 Equalization Control</b>					
19:16	<b>Reserved</b>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <a href="#">User Lane Equalization Control</a> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <a href="#">BC8h</a> ).	Downstream	ROS	Yes	0h
22:20	<b>Reserved</b>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
23	<b>Reserved</b>		RsvdP	No	0
27:24	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <a href="#">Link Control 2</a> register <i>Transmit Margin</i> field (offset <a href="#">98h[9:7]</a> ) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <a href="#">User Lane Equalization Control</a> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <a href="#">BC8h</a> ).	Downstream	ROS	Yes	0h
30:28	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <a href="#">Link Control 2</a> register <i>Transmit Margin</i> field (offset <a href="#">98h[9:7]</a> ) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
31	<b>Reserved</b>		RsvdP	No	0

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1



**Register 13-47. 124h Lane 6, 22, or 38 and 7, 23, or 39 Equalization Control (All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 6, 22, or 38 Equalization Control</b>					
3:0	<i>Reserved</i>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <a href="#">User Lane Equalization Control</a> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <a href="#">BC8h</a> ).	Downstream	ROS	Yes	0h
6:4	<i>Reserved</i>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
7	<i>Reserved</i>		RsvdP	No	0

**Register 13-47. 124h Lane 6, 22, or 38 and 7, 23, or 39 Equalization Control (All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
11:8	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
14:12	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
15	<b>Reserved</b>		RsvdP	No	0

**Register 13-47. 124h Lane 6, 22, or 38 and 7, 23, or 39 Equalization Control  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 7, 23, or 39 Equalization Control</b>					
19:16	<b>Reserved</b>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
22:20	<b>Reserved</b>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
23	<b>Reserved</b>		RsvdP	No	0
27:24	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
30:28	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
31	<b>Reserved</b>		RsvdP	No	0

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 13-48. 128h Lane 8, 24, or 40 and 9, 25, or 41 Equalization Control (All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 8, 24, or 40 Equalization Control</b>					
3:0	<b>Reserved</b>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <b>BC8h</b> ).	Downstream	ROS	Yes	0h
6:4	<b>Reserved</b>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
7	<b>Reserved</b>		RsvdP	No	0

**Register 13-48. 128h Lane 8, 24, or 40 and 9, 25, or 41 Equalization Control (All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
11:8	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
14:12	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
15	<b>Reserved</b>		RsvdP	No	0

**Register 13-48. 128h Lane 8, 24, or 40 and 9, 25, or 41 Equalization Control (All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 9, 25, or 41 Equalization Control</b>					
19:16	<b>Reserved</b>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <a href="#">User Lane Equalization Control</a> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <a href="#">BC8h</a> ).	Downstream	ROS	Yes	0h
22:20	<b>Reserved</b>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
23	<b>Reserved</b>		RsvdP	No	0
27:24	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <a href="#">Link Control 2</a> register <i>Transmit Margin</i> field (offset <a href="#">98h[9:7]</a> ) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <a href="#">User Lane Equalization Control</a> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <a href="#">BC8h</a> ).	Downstream	ROS	Yes	0h
30:28	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <a href="#">Link Control 2</a> register <i>Transmit Margin</i> field (offset <a href="#">98h[9:7]</a> ) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
31	<b>Reserved</b>		RsvdP	No	0

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

**Register 13-49. 12Ch Lane 10, 26, or 42 and 11, 27, or 43 Equalization Control (All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 10, 26, or 42 Equalization Control</b>					
3:0	<i>Reserved</i>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <a href="#">User Lane Equalization Control</a> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <a href="#">BC8h</a> ).	Downstream	ROS	Yes	0h
6:4	<i>Reserved</i>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
7	<i>Reserved</i>		RsvdP	No	0

**Register 13-49. 12Ch Lane 10, 26, or 42 and 11, 27, or 43 Equalization Control  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
11:8	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
14:12	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
15	<b>Reserved</b>		RsvdP	No	0



**Register 13-49. 12Ch Lane 10, 26, or 42 and 11, 27, or 43 Equalization Control (All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 11, 27, or 43 Equalization Control</b>					
19:16	<b>Reserved</b>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
22:20	<b>Reserved</b>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
23	<b>Reserved</b>		RsvdP	No	0
27:24	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
30:28	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
31	<b>Reserved</b>		RsvdP	No	0

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 13-50. 130h Lane 12, 28, or 44 and 13, 29, or 45 Equalization Control (All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 12, 28, or 44 Equalization Control</b>					
3:0	<b>Reserved</b>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <b>BC8h</b> ).	Downstream	ROS	Yes	0h
6:4	<b>Reserved</b>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
7	<b>Reserved</b>		RsvdP	No	0

**Register 13-50. 130h Lane 12, 28, or 44 and 13, 29, or 45 Equalization Control (All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
11:8	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
14:12	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
15	<b>Reserved</b>		RsvdP	No	0

**Register 13-50. 130h Lane 12, 28, or 44 and 13, 29, or 45 Equalization Control (All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 13, 29, or 45 Equalization Control</b>					
19:16	<b>Reserved</b>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <b>BC8h</b> ).	Downstream	ROS	Yes	0h
22:20	<b>Reserved</b>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
23	<b>Reserved</b>		RsvdP	No	0
27:24	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset <b>98h[9:7]</b> ) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <b>BC8h</b> ).	Downstream	ROS	Yes	0h
30:28	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset <b>98h[9:7]</b> ) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
31	<b>Reserved</b>		RsvdP	No	0

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

**Register 13-51. 134h Lane 14, 30, or 46 and 15, 31, or 47 Equalization Control (All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 14, 30, or 46 Equalization Control</b>					
3:0	<i>Reserved</i>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
6:4	<i>Reserved</i>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
7	<i>Reserved</i>		RsvdP	No	0

**Register 13-51. 134h Lane 14, 30, or 46 and 15, 31, or 47 Equalization Control  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
11:8	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
14:12	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
15	<b>Reserved</b>		RsvdP	No	0

**Register 13-51. 134h Lane 14, 30, or 46 and 15, 31, or 47 Equalization Control  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Lane 15, 31, or 47 Equalization Control</b>					
19:16	<b>Reserved</b>	Upstream	RsvdP	No	0h
	<b>Downstream Port Transmitter Preset</b> Preset used by the Transparent Downstream Port's Transmitters during Equalization Phase 1. Can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
22:20	<b>Reserved</b>	Upstream	RsvdP	No	000b
	<b>Downstream Port Receiver Preset Hint</b> Contains the Receiver Preset Hint value received from the Downstream device during Link Equalization, which can be used to adjust the Transparent Downstream Port's Receivers upon the change to Gen 3 speed. Use of the Receiver Preset Hint is optional (per the <i>PCI Express Base r3.0</i> ), and the Port does not use the Hint.	Downstream	ROS	Yes	000b
23	<b>Reserved</b>		RsvdP	No	0
27:24	<b>Upstream Port Transmitter Preset</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Transmit Preset value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	Fh
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device must use this value for its Transmitter Preset upon the speed change to Gen 3 speed. For Downstream Ports, the value can be loaded using the <b>User Lane Equalization Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BC8h).	Downstream	ROS	Yes	0h
30:28	<b>Upstream Port Receiver Preset Hint</b> For Upstream Port(s), this field is intended for debug and diagnostics. If cross-link (enabled by default) is disabled for this Port (bit 8 of the Port's <b>Link Control 2</b> register <i>Transmit Margin</i> field (offset 98h[9:7]) is Cleared), this field contains the Receiver Preset Hint value captured from the Equalization TS2s that are received on the associated Lane during Link Equalization; otherwise, this field is not used.	Upstream	ROS	No	111b
	For Transparent Downstream Ports, the Port transmits this value to the Downstream device, within the equalization TS2s sent during the <i>Recovery.RcvrCfg</i> substate preceding the speed change to Gen 3 speed. The Downstream device can use this value to adjust its Receivers upon the speed change to Gen 3 speed. For Downstream Ports, contains the value sent on the associated Lane during Link Equalization.	Downstream	ROS	Yes	000b
31	<b>Reserved</b>		RsvdP	No	0

NDA CONFIDENTIAL -- Inventec | King Michael

### 13.14 Power Budget Extended Capability Registers (Offsets 138h – 144h)

This section details the Power Budget Extended Capability registers. These registers work differently than the others, especially with respect to serial EEPROM Reads and Writes. *For example*, when writing to Index 5 of the **Power Budget Data** register (Upstream Port(s), offset 140h), write 5 into the **Data Select** register *Data Select* field (Upstream Port(s), offset 13Ch[7:0]), then write the value into the **Power Budget Data** register. Table 13-14 defines the register map.

**Table 13-14. Power Budget Extended Capability Register Map (Upstream Port(s))**

31 30 29 28 27 26 25 24 23 22 21 20											19 18 17 16											15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0										
Next Capability Offset (10Ch) (Upstream)											Capability Version (1h) (Upstream)					PCI Express Extended Capability ID (0004h) (Upstream)																138h
Reserved (Downstream)																																
Reserved (Upstream)																		Data Select (Upstream)												13Ch		
Reserved (Downstream)																																
Power Budget Data (Upstream) Reserved (Downstream)																														140h		
Power Budget Capability (Upstream) Reserved (Downstream)																														144h		

### Register 13-52. 138h Power Budget Extended Capability Header (Upstream Port(s))

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>PCI Express Extended Capability ID</b> Program to 0004h, as required by the <i>PCI Express Base r3.0</i> .	Upstream	RO	Yes	0004h
	<i>Reserved</i>	Downstream	RsvdP	No	0000h
19:16	<b>Capability Version</b> Program to 1h, as required by the <i>PCI Express Base r3.0</i> .	Upstream	RO	Yes	1h
	<i>Reserved</i>	Downstream	RsvdP	No	0h
31:20	<b>Next Capability Offset</b> Program to 10Ch, which addresses the <b>Secondary PCI Express Extended Capability</b> structure.	Upstream	RO	Yes	10Ch
	<i>Reserved</i>	Downstream	RsvdP	No	000h



**Register 13-53. 13Ch Data Select  
(Upstream Port(s))**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Data Select</b> Indexes the Power Budget data reported, by way of eight <b>Power Budget Data</b> registers, per Upstream Port, and selects the DWord of Power Budget data that appears in each <b>Power Budget Data</b> register. Index values start at 0, to select the first DWord of Power Budget data; subsequent DWords of Power Budget data are selected by increasing index values 1 to 7.	Upstream	RW	Yes	00h
	<b>Reserved</b>	Downstream	RsvdP	No	00h
31:8	<b>Reserved</b>		RsvdP	No	0000_00h

**Register 13-54. 140h Power Budget Data  
(Upstream Port(s))**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> Eight registers, per Upstream Port, can be programmed, through the serial EEPROM, I <sup>2</sup> C, and/or SMBus. Each non-zero register value describes the power usage for a different operating condition. Each configuration is selected by writing to the <b>Data Select</b> register <b>Data Select</b> field (offset 13Ch[7:0]).					
7:0	<b>Base Power</b> Eight registers, per Upstream Port. Specifies (in Watts) the base power value in the operating condition. This value must be multiplied by the field [9:8] ( <b>Data Scale</b> ) contents, to produce the actual power consumption value.	Upstream	RO	Yes	00h
	<b>Reserved</b>	Downstream	RsvdP	No	00h
9:8	<b>Data Scale</b> Specifies the scale to apply to the Base Power value. The device power consumption is determined by multiplying the field [7:0] ( <b>Base Power</b> ) contents with the value corresponding to the encoding returned by this field. 00b = 1.0x 01b = 0.1x 10b = 0.01x 11b = 0.001x	Upstream	RO	Yes	00b
	<b>Reserved</b>	Downstream	RsvdP	No	00b
12:10	<b>PM Sub-State</b> 000b = Power Management substate of the operating condition being described	Upstream	RO	Yes	000b
	<b>Reserved</b>	Downstream	RsvdP	No	000b

**Register 13-54. 140h Power Budget Data  
(Upstream Port(s)) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
14:13	<b>PM State</b> Power Management state of the operating condition being described. 00b = D0 state 11b = D3 state All other encodings are <i>Reserved</i> .	Upstream	RO	Yes	00b
	<i>Reserved</i>	Downstream	RsvdP	No	00b
17:15	<b>Type</b> Type of operating condition being described. 000b = PME Auxiliary 001b = Auxiliary 010b = Idle 011b = Sustained 111b = Maximum All other encodings are <i>Reserved</i> .	Upstream	RO	Yes	000b
	<i>Reserved</i>	Downstream	RsvdP	No	000b
20:18	<b>Power Rail</b> Power Rail of the operating condition being described. 000b = Power 12V 001b = Power 3.3V 010b = Power 1.8V 111b = Thermal All other encodings are <i>Reserved</i> .	Upstream	RO	Yes	000b
	<i>Reserved</i>	Downstream	RsvdP	No	000b
31:21	<i>Reserved</i>		RsvdP	No	0-0h

**Register 13-55. 144h Power Budget Capability  
(Upstream Port(s))**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>System Allocated</b> 1 = Power budget for the device is included within the system power budget	Upstream	HwInit	Yes	1
	<i>Reserved</i>	Downstream	RsvdP	No	0
31:1	<i>Reserved</i>		RsvdP	No	0-0h

### 13.15 Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)

This section details the Virtual Channel Extended Capability registers, which are duplicated for each Port, as well as the WRR Port Arbitration Table registers. [Table 13-15](#) defines the register map for one Port.

**Table 13-15. Virtual Channel Extended Capability Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																
Next Capability Offset (E00h)												Capability Version (1h)				PCI Express Extended Capability ID (0002h)																148h
Port VC Capability 1																																14Ch
<i>Reserved</i>																				Port VC Capability 2												150h
Port VC Status ( <i>Reserved</i> )																Port VC Control																154h
VC0 Resource Capability																																158h
VC0 Resource Control																																15Ch
VC0 Resource Status																<i>Reserved</i>																160h
<i>Reserved</i>																																164h –
WRR Port Arbitration Table Registers (Offsets 178h – 1BCh)																																178h
																																...
																																1BCh

### Register 13-56. 148h Virtual Channel Extended Capability Header (All Ports)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>PCI Express Extended Capability ID</b> Program to 0002h, as required by the <i>PCI Express Base r3.0</i> .	RO	No	0002h
19:16	<b>Capability Version</b> Program to 1h, as required by the <i>PCI Express Base r3.0</i> .	RO	No	1h
31:20	<b>Next Capability Offset</b> Next extended capability is the <b>Multicast Extended Capability</b> structure, offset E00h.	RO	No	E00h

**Register 13-57. 14Ch Port VC Capability 1  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
2:0	<b>Extended VC Counter</b> 000b = Port supports only one Virtual Channel, VC0 001b = <i>Reserved</i> , because the PEX 8748 supports only one VC All other encodings are <i>Reserved</i> .		RO	No	000b
3	<i>Reserved</i>		RsvdP	No	0
6:4	<b>Low-Priority Extended VC Counter</b> For Strict Priority arbitration, indicates the quantity of extended Virtual Channels (VCs) (those in addition to VC0) that belong to the Low-Priority VC group for this Port. 000b = For this Port, only VC0 belongs to the Low-Priority VC group 001b = <i>Reserved</i> , because the PEX 8748 supports only one VC All other encodings are <i>Reserved</i> .		RO	No	000b
7	<i>Reserved</i>		RsvdP	No	0
9:8	<b>Reference Clock</b> <i>Reserved</i>		RsvdP	No	00b
11:10	<b>Port Arbitration Table Entry Size</b> 00b = Port Arbitration Table entry size is 1 bit 11b = Port Arbitration Table entry size is 8 bits All other encodings are <i>Reserved</i> .	Upstream	RO	Yes	11b
	<i>Reserved</i>	Downstream	RsvdP	No	00b
31:12	<i>Reserved</i>		RsvdP	No	0000_0h

**Register 13-58. 150h Port VC Capability 2  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>VC Arbitration Capability</b> Indicates the type of VC arbitration supported by the device for the LPVC (Low-Priority Extended VC) group.  00h = Indicates that the Round-Robin (Hardware-Fixed) Arbitration scheme is not supported 01h = <i>Reserved</i> , because the PEX 8748 supports only one VC ( <b>Port VC Capability 2</b> register <i>Low-Priority Extended VC Counter</i> field, offset 14Ch[6:4]=000b)	RO	No	00h
31:8	<i>Reserved</i>	RsvdP	No	0000_00h

**Register 13-59. 154h Port VC Status and Control  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Port VC Control</b>				
0	<b>Load VC Arbitration Table</b> <i>Not supported</i> The PEX 8748 supports only one Virtual Channel, VC0; therefore, a VC Arbitration Table is not present ( <b>Port VC Capability 2</b> register <i>Port Arbitration Table Offset</i> field (offset 150h[31:24]) is Cleared). Reads always return a value of 0.	RsvdP	No	0
3:1	<b>VC Arbitration Select</b> Selects the Port's VC arbitration type, as per the supported arbitration type indicated by the <b>Port VC Capability 2</b> register <i>VC Arbitration Capability</i> field (offset 150h[7:0]) value.  000b = Bit 0; Round-Robin (Hardware-Fixed) arbitration scheme  All other encodings are <i>Reserved</i> .	RW	Yes	000b
15:4	<i>Reserved</i>	RsvdP	No	000h
<b>Port VC Status</b>				
16	<b>VC Arbitration Table Status</b> <i>Reserved</i>	RsvdP	No	0
31:17	<i>Reserved</i>	RsvdP	No	0-0h

**Register 13-60. 158h VC0 Resource Capability  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
2:0	<b>Port Arbitration Capability</b> Bit 0 = 1 – Non-configurable Round-Robin (Hardware-Fixed) arbitration Bit 1 = 1 – Weighted Round-Robin (WRR) arbitration with 32 Phases Bit 2 = 1 – WRR arbitration with 64 Phases	Upstream	RO	No	100b
		Downstream	RO	No	001b
13:3	<b>Reserved</b>		RsvdP	No	0-0h
14	<b>Advanced Packet Switching</b>		RsvdP	No	0
15	<b>Reject Snoop Transactions</b> Not a PCI Express switch feature; therefore, this bit is Cleared.		RsvdP	No	0
22:16	<b>Maximum Time Slots</b> <b>Reserved</b>		RsvdP	No	000_0000b
23	<b>Reserved</b>		RsvdP	No	0
31:24	<b>Port Arbitration Table Offset</b> Offset of the Port Arbitration Table, as the quantity of DQWords from the Base address of the <b>Virtual Channel Extended Capability</b> structure. (Refer to <a href="#">Section 13.15.1</a> for further details.)  00h = Port Arbitration Table is not present 03h = Port Arbitration Table is located at register offset <a href="#">178h</a>	Upstream	RO	No	03h
	<b>Reserved</b>	Downstream	RO	No	00h

**Register 13-61. 15Ch VC0 Resource Control  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC Map</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped to VC0. Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.		RO	No	1
7:1			RW	Yes	7Fh
15:8	<b>Reserved</b>		RsvdP	No	00h
16	<b>Load Port Arbitration Table</b> Software writes this bit, to load the updated WRR Port Arbitration Table value to the internal logic. Software Read always returns a value of 0.	Upstream	RW	Yes	0
	<b>Reserved</b>	Downstream	RsvdP	No	0
19:17	<b>Port Arbitration Select</b> Selects the Port's Port Arbitration type. Indicates the bit number in the <b>VC0 Resource Capability</b> register <i>Port Arbitration Capability</i> field (offset 158h[2:0]) that corresponds to the arbitration type. Allowed values: <ul style="list-style-type: none"> <li>• 000b if the Port Arbitration Table is not present</li> <li>• 000b or 010b if the Port Arbitration Table is present</li> </ul> 000b = Fair Bandwidth (Hardware-Fixed Arbitration) 010b = Weighted Round-Robin with 64 Phases <i>Note: If software programs other values, hardware ignores the value.</i>	Upstream	RW	Yes	010b
		Downstream	RW	Yes	000b
23:20	<b>Reserved</b>		RsvdP	No	0h
26:24	<b>VC0 ID</b> Defines the Port's VC0 ID code. 000b = VC0 (default; VC0 is the only/default VC) All other encodings are <b>Reserved</b> .		RO	No	000b
30:27	<b>Reserved</b>		RsvdP	No	0h
31	<b>VC0 Enable</b> 0 = Not allowed 1 = Enables the Port's VC0		RO	No	1

**Register 13-62. 160h VC0 Resource Status  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<i>Reserved</i>		RsvdP	No	0000h
16	<b>Port Arbitration Table Status</b> 0 = Hardware has finished loading values stored in the Port Arbitration Table, after software Sets the <b>VC0 Resource Control</b> register <i>Load Port Arbitration Table</i> bit (offset 15Ch[16]), or if the Port Arbitration Table is not implemented, then this bit is <i>Reserved</i> 1 = Port Arbitration Table entry was written to by software	Upstream	RO	No	0
	<i>Reserved</i>	Downstream	RsvdP	No	0
17	<b>VC0 Negotiation Pending</b> 0 = Port's VC0 negotiation is complete 1 = Port's VC0 initialization is not complete		RO	Yes	1
31:18	<i>Reserved</i>		RsvdP	No	0-0h



### 13.15.1 WRR Port Arbitration Table Registers (Offsets 178h – 1BCh)

This section details the WRR Port Arbitration Table registers. Port Arbitration Table phases are used to determine Port weighting during “Weighted Round-Robin (WRR) with 64 Phases” Port arbitration.

Table 13-16 defines the register map. The numbers along the top of the register map table indicate the 8-bit fields of each 32-bit register. There are 64 phases, and any active Port Number can go into each Port *x* Phase *x* box.

**Note:** *The Port Arbitration Table is used only by the Upstream Port(s) when Weighted Round-Robin with 64-Phase Port Arbitration is selected, by way of the VC0 Resource Control register Port Arbitration Select field (offset 15Ch[19:17]=010b).*

**Table 13-16. WRR Port Arbitration Table Register Map**  
(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
Port <i>x</i> , Phase 3								Port <i>x</i> , Phase 2								Port <i>x</i> , Phase 1								Port <i>x</i> , Phase 0								178h							
Port <i>x</i> , Phase 7								Port <i>x</i> , Phase 6								Port <i>x</i> , Phase 5								Port <i>x</i> , Phase 4								17Ch							
Port <i>x</i> , Phase 11								Port <i>x</i> , Phase 10								Port <i>x</i> , Phase 9								Port <i>x</i> , Phase 8								180h							
Port <i>x</i> , Phase 15								Port <i>x</i> , Phase 14								Port <i>x</i> , Phase 13								Port <i>x</i> , Phase 12								184h							
Port <i>x</i> , Phase 19								Port <i>x</i> , Phase 18								Port <i>x</i> , Phase 17								Port <i>x</i> , Phase 16								188h							
Port <i>x</i> , Phase 23								Port <i>x</i> , Phase 22								Port <i>x</i> , Phase 21								Port <i>x</i> , Phase 20								18Ch							
Port <i>x</i> , Phase 27								Port <i>x</i> , Phase 26								Port <i>x</i> , Phase 25								Port <i>x</i> , Phase 24								190h							
Port <i>x</i> , Phase 31								Port <i>x</i> , Phase 30								Port <i>x</i> , Phase 29								Port <i>x</i> , Phase 28								194h							
Port <i>x</i> , Phase 35								Port <i>x</i> , Phase 34								Port <i>x</i> , Phase 33								Port <i>x</i> , Phase 32								198h							
Port <i>x</i> , Phase 39								Port <i>x</i> , Phase 38								Port <i>x</i> , Phase 37								Port <i>x</i> , Phase 36								19Ch							
Port <i>x</i> , Phase 43								Port <i>x</i> , Phase 42								Port <i>x</i> , Phase 41								Port <i>x</i> , Phase 40								1A0h							
Port <i>x</i> , Phase 47								Port <i>x</i> , Phase 46								Port <i>x</i> , Phase 45								Port <i>x</i> , Phase 44								1A4h							
Port <i>x</i> , Phase 51								Port <i>x</i> , Phase 50								Port <i>x</i> , Phase 49								Port <i>x</i> , Phase 48								1A8h							
Port <i>x</i> , Phase 55								Port <i>x</i> , Phase 54								Port <i>x</i> , Phase 53								Port <i>x</i> , Phase 52								1ACh							
Port <i>x</i> , Phase 59								Port <i>x</i> , Phase 58								Port <i>x</i> , Phase 57								Port <i>x</i> , Phase 56								1B0h							
Port <i>x</i> , Phase 63								Port <i>x</i> , Phase 62								Port <i>x</i> , Phase 61								Port <i>x</i> , Phase 60								1B4h							
Reserved																																1B8h –				1BCh			

**Register 13-63. 178h Port Arbitration Table Phases 0 to 3**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	<b>Port Arbitration Table Phase 0</b>	RW	Yes	Value is based upon the Port's <b>Negotiated Link Width</b> and <b>Current Link Speed</b> (offset 78h[25:20 and 19:16], respectively)
7:5	<i>Reserved</i>	RsvdP	No	000b
12:8	<b>Port Arbitration Table Phase 1</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	<i>Reserved</i>	RsvdP	No	000b
20:16	<b>Port Arbitration Table Phase 2</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	<i>Reserved</i>	RsvdP	No	000b
28:24	<b>Port Arbitration Table Phase 3</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	<i>Reserved</i>	RsvdP	No	000b

**Register 13-64. 17Ch Port Arbitration Table Phases 4 to 7**

**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	<b>Port Arbitration Table Phase 4</b>	RW	Yes	Value is based upon the Port's <a href="#">Negotiated Link Width</a> and <a href="#">Current Link Speed</a> (offset 78h[25:20 and 19:16], respectively)
7:5	<i>Reserved</i>	RsvdP	No	000b
12:8	<b>Port Arbitration Table Phase 5</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	<i>Reserved</i>	RsvdP	No	000b
20:16	<b>Port Arbitration Table Phase 6</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	<i>Reserved</i>	RsvdP	No	000b
28:24	<b>Port Arbitration Table Phase 7</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	<i>Reserved</i>	RsvdP	No	000b

**Register 13-65. 180h Port Arbitration Table Phases 8 to 11**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	Port Arbitration Table Phase 8	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
7:5	Reserved	RsvdP	No	000b
12:8	Port Arbitration Table Phase 9	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	Reserved	RsvdP	No	000b
20:16	Port Arbitration Table Phase 10	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	Reserved	RsvdP	No	000b
28:24	Port Arbitration Table Phase 11	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	Reserved	RsvdP	No	000b

**Register 13-66. 184h Port Arbitration Table Phases 12 to 15**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	Port Arbitration Table Phase 12	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
7:5	Reserved	RsvdP	No	000b
12:8	Port Arbitration Table Phase 13	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	Reserved	RsvdP	No	000b
20:16	Port Arbitration Table Phase 14	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	Reserved	RsvdP	No	000b
28:24	Port Arbitration Table Phase 15	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	Reserved	RsvdP	No	000b

**Register 13-67. 188h Port Arbitration Table Phases 16 to 19**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	Port Arbitration Table Phase 16	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
7:5	Reserved	RsvdP	No	000b
12:8	Port Arbitration Table Phase 17	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	Reserved	RsvdP	No	000b
20:16	Port Arbitration Table Phase 18	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	Reserved	RsvdP	No	000b
28:24	Port Arbitration Table Phase 19	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	Reserved	RsvdP	No	000b

**Register 13-68. 18Ch Port Arbitration Table Phases 20 to 23**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	Port Arbitration Table Phase 20	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
7:5	Reserved	RsvdP	No	000b
12:8	Port Arbitration Table Phase 21	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	Reserved	RsvdP	No	000b
20:16	Port Arbitration Table Phase 22	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	Reserved	RsvdP	No	000b
28:24	Port Arbitration Table Phase 23	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	Reserved	RsvdP	No	000b

**Register 13-69. 190h Port Arbitration Table Phases 24 to 27**

**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	<b>Port Arbitration Table Phase 24</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
7:5	<i>Reserved</i>	RsvdP	No	000b
12:8	<b>Port Arbitration Table Phase 25</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	<i>Reserved</i>	RsvdP	No	000b
20:16	<b>Port Arbitration Table Phase 26</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	<i>Reserved</i>	RsvdP	No	000b
28:24	<b>Port Arbitration Table Phase 27</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	<i>Reserved</i>	RsvdP	No	000b



**Register 13-70. 194h Port Arbitration Table Phases 28 to 31**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	Port Arbitration Table Phase 28	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
7:5	Reserved	RsvdP	No	000b
12:8	Port Arbitration Table Phase 29	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	Reserved	RsvdP	No	000b
20:16	Port Arbitration Table Phase 30	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	Reserved	RsvdP	No	000b
28:24	Port Arbitration Table Phase 31	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	Reserved	RsvdP	No	000b

**Register 13-71. 198h Port Arbitration Table Phases 32 to 35**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	<b>Port Arbitration Table Phase 32</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
7:5	<i>Reserved</i>	RsvdP	No	000b
12:8	<b>Port Arbitration Table Phase 33</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	<i>Reserved</i>	RsvdP	No	000b
20:16	<b>Port Arbitration Table Phase 34</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	<i>Reserved</i>	RsvdP	No	000b
28:24	<b>Port Arbitration Table Phase 35</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	<i>Reserved</i>	RsvdP	No	000b

**Register 13-72. 19Ch Port Arbitration Table Phases 36 to 39**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	Port Arbitration Table Phase 36	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
7:5	Reserved	RsvdP	No	000b
12:8	Port Arbitration Table Phase 37	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	Reserved	RsvdP	No	000b
20:16	Port Arbitration Table Phase 38	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	Reserved	RsvdP	No	000b
28:24	Port Arbitration Table Phase 39	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	Reserved	RsvdP	No	000b

**Register 13-73. 1A0h Port Arbitration Table Phases 40 to 43**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	<b>Port Arbitration Table Phase 40</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
7:5	<i>Reserved</i>	RsvdP	No	000b
12:8	<b>Port Arbitration Table Phase 41</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	<i>Reserved</i>	RsvdP	No	000b
20:16	<b>Port Arbitration Table Phase 42</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	<i>Reserved</i>	RsvdP	No	000b
28:24	<b>Port Arbitration Table Phase 43</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	<i>Reserved</i>	RsvdP	No	000b

**Register 13-74. 1A4h Port Arbitration Table Phases 44 to 47**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	Port Arbitration Table Phase 44	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
7:5	Reserved	RsvdP	No	000b
12:8	Port Arbitration Table Phase 45	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	Reserved	RsvdP	No	000b
20:16	Port Arbitration Table Phase 46	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	Reserved	RsvdP	No	000b
28:24	Port Arbitration Table Phase 47	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	Reserved	RsvdP	No	000b

**Register 13-75. 1A8h Port Arbitration Table Phases 48 to 51**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	<b>Port Arbitration Table Phase 48</b>	RW	Yes	Value is based upon the Port's <b>Negotiated Link Width</b> and <b>Current Link Speed</b> (offset 78h[25:20 and 19:16], respectively)
7:5	<i>Reserved</i>	RsvdP	No	000b
12:8	<b>Port Arbitration Table Phase 49</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	<i>Reserved</i>	RsvdP	No	000b
20:16	<b>Port Arbitration Table Phase 50</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	<i>Reserved</i>	RsvdP	No	000b
28:24	<b>Port Arbitration Table Phase 51</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	<i>Reserved</i>	RsvdP	No	000b

**Register 13-76. 1ACh Port Arbitration Table Phases 52 to 55**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	Port Arbitration Table Phase 52	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
7:5	Reserved	RsvdP	No	000b
12:8	Port Arbitration Table Phase 53	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	Reserved	RsvdP	No	000b
20:16	Port Arbitration Table Phase 54	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	Reserved	RsvdP	No	000b
28:24	Port Arbitration Table Phase 55	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	Reserved	RsvdP	No	000b

**Register 13-77. 1B0h Port Arbitration Table Phases 56 to 59**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	<b>Port Arbitration Table Phase 56</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
7:5	<i>Reserved</i>	RsvdP	No	000b
12:8	<b>Port Arbitration Table Phase 57</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	<i>Reserved</i>	RsvdP	No	000b
20:16	<b>Port Arbitration Table Phase 58</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	<i>Reserved</i>	RsvdP	No	000b
28:24	<b>Port Arbitration Table Phase 59</b>	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	<i>Reserved</i>	RsvdP	No	000b



**Register 13-78. 1B4h Port Arbitration Table Phases 60 to 63**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	Port Arbitration Table Phase 60	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
7:5	Reserved	RsvdP	No	000b
12:8	Port Arbitration Table Phase 61	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
15:13	Reserved	RsvdP	No	000b
20:16	Port Arbitration Table Phase 62	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
23:21	Reserved	RsvdP	No	000b
28:24	Port Arbitration Table Phase 63	RW	Yes	Value is based upon the Port's Negotiated Link Width and Current Link Speed (offset 78h[25:20 and 19:16], respectively)
31:29	Reserved	RsvdP	No	000b

## 13.16 Device-Specific Registers (Offsets 1C0h – A74h)

This section details the Device-Specific registers located at offsets 1C0h through A74h. Device-Specific registers are unique to the PEX 8748 and not referenced in the *PCI Express Base r3.0*. [Table 13-17](#) defines the register map.

Other Device-Specific registers are detailed in:

- [Section 13.18, “Device-Specific Registers \(Offsets B70h – DFCh\)”](#)
- [Section 13.20, “Device-Specific Registers – Virtual Switch \(Offsets F00h – F20h\), Virtual Switch Mode”](#)
- [Section 13.22, “Device-Specific Registers \(Offsets F30h – FB0h\)”](#)

*Note: It is recommended that these registers not be changed from their default values.*

**Table 13-17. Device-Specific Register Map  
(Offsets 1C0h – A74h)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
<i>Reserved</i>		1C0h – 1CCh
Device-Specific Registers – Read Pacing (Offsets 1D0h – 1D8h)		1D0h
		...
		1D8h
Device-Specific Registers – Captured Bus and Device Numbers (Offsets 1DCh – 1E4h)		1DCh
		...
		1E4h
Device-Specific Registers – Read Pacing (Offset 1E8h)		1E8h
<i>Reserved</i>		1ECh – 1FCh
Device-Specific Registers – Physical Layer (Offsets 200h – 25Ch)		200h
		...
		25Ch
Device-Specific Registers – Serial EEPROM (Offsets 260h – 270h)		260h
		...
		270h
<i>Reserved</i>		274h – 28Ch
Device-Specific Registers – I2C and SMBus Slave Interfaces (Offsets 290h – 2FCh)		290h
		...
		2FCh
Device-Specific Registers – Port Configuration and Lane Status (Offsets 300h – 350h)		300h
		...
		350h
Device-Specific Registers – Port Configuration (Offsets 354h – 46Ch)		354h
		...
		46Ch

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

**Table 13-17. Device-Specific Register Map  
(Offsets 1C0h – A74h) (Cont.)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																
<i>Factory Test Only/Reserved</i>																470h –	5FCh
Device-Specific Registers – General-Purpose Input/Output (Offsets 600h – 68Ch)																	600h
																	...
																	68Ch
<i>Factory Test Only</i>																690h –	69Ch
Device-Specific Registers – PORT_GOOD Selector (Offsets 6A0h – 6A8h)																	6A0h
																	...
																	6A8h
<i>Reserved</i>																6ACh –	6FCh
Device-Specific Registers – Error Checking and Debug (Offsets 700h – 75Ch)																	700h
																	...
																	75Ch
Device-Specific Registers – Control (Offsets 760h – 774h)																	760h
																	...
																	774h
Device-Specific Registers – Soft Error (Offsets 778h – 8FCh)																	778h
																	...
																	8FCh
<i>Reserved</i> (Base Mode)																	900h
Device-Specific Registers – Virtual Switch (Offsets 900h – 9E8h), Virtual Switch Mode (Virtual Switch Mode)																	...
																	9E8h
Device-Specific Registers – Ingress Credit Handler (Offsets 9ECh – A2Ch)																	9ECh
																	...
																	A2Ch
Device-Specific Registers – Virtual Switch Debug and GPIO Status and Control (Offsets A30h – A74h)																	A30h
																	...
																	A74h

13.16.1     **Device-Specific Registers – Read Pacing  
(Offsets 1D0h – 1D8h)**

*Note: Source Queuing and Read Pacing should not be concurrently enabled. The two features are incompatible and enabling both concurrently can result in Fatal errors.*

This section details the Device-Specific Read Pacing registers located at offsets 1D0h through 1D8h. [Table 13-18](#) defines the register map.

Another Device-Specific Read Pacing register is detailed in [Section 13.16.3, “Device-Specific Registers – Read Pacing \(Offset 1E8h\).”](#)

Read Pacing is described, in detail, in [Section 8.5, “Read Pacing.”](#)

**Table 13-18.    Device-Specific Read Pacing Register Map  
(Offsets 1D0h – 1D8h) (Base mode – Ports 0, 8, and 16; Virtual Switch  
mode – Ports 0, 8, and 16, accessible through the Management Port)**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Read Pacing Control																																1D0h
Read Pacing Threshold 1																																1D4h
Read Pacing Threshold 2																																1D8h

**Register 13-79. 1D0h Read Pacing Control**

**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name.</p> <p>Read Pacing and Source Queuing should not be concurrently enabled. The two features are incompatible and enabling both concurrently can result in Fatal errors.</p> <p>Read Pacing must be enabled for Read Spreading to be enabled. (That is, for a Port to have Read Spreading enabled, the corresponding Port bits within this register, for both Read Pacing and Read Spreading, must both be Cleared.)</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
0	<b>Port x Read Pacing Disable</b>	0, 1, 8, 9, 16, 17	RWS	Yes	1
1	0 = Read Pacing is enabled for this Port 1 = Read Pacing is disabled for this Port	2, 3, 10, 11, 18, 19	RWS	Yes	1
2	<b>Factory Test Only</b>		RsvdP	No	0
3	<b>Reserved</b>		ROS	No	1
11:4	<b>Reserved</b>		RsvdP	No	00h
12	<b>Port x Memory Read Outstanding Counter Reset</b> Provides a mechanism for Clearing the Counter when an error condition occurs, with either the device that issued the Read Request, or the device that was to provide the Read Completions, where a System Level Reset will not be issued. If the Counter is not Cleared after an error condition such as this, the threshold will not be accurate. Read returns a value of 0.	0, 1, 8, 9, 16, 17	RZ	Yes	0
13	0 = Read Outstanding Counter value increments, with each outstanding Read 1 = Resets Read Outstanding Counter	2, 3, 10, 11, 18, 19	RZ	Yes	0
14	<b>Factory Test Only</b>		RsvdP	No	0
15	<b>Reserved</b>		RZ	Yes	0

**Register 13-79. 1D0h Read Pacing Control**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (*Cont.*)

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Port x Memory Read Spreading Disable</b> 0 = Memory Read Spreading is enabled for this Port 1 = Memory Read Spreading is disabled for this Port	0, 1, 8, 9, 16, 17	RWS	Yes	0
17		2, 3, 10, 11, 18, 19	RWS	Yes	0
18	<b>Factory Test Only</b>		RsvdP	No	0
19	<b>Reserved</b>		ROS	No	0
27:20	<b>Reserved</b>		RsvdP	No	00h
31:28	<b>Maximum Read Response Time</b> 0h = Disabled 1h = 5 ms (default) 2h = 10 ms 3h = 15 ms 4h = 100 ms 5h = 200 ms 6h = 300 ms 7h = 500 ms 8h = 1.0s 9h = 2.0s All other encodings are <b>Reserved</b> .		RWS	Yes	1h

**Register 13-80. 1D4h Read Pacing Threshold 1****(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Notes:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports. The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.				
12:0	<b>x16 Port Memory Read Outstanding Threshold</b> Specified in DWords. Default value of 800h programs the threshold to 8 KB.	RWS	Yes	800h
15:13	<b>Reserved</b>	RsvdP	No	000b
28:16	<b>x8 Port Memory Read Outstanding Threshold</b> Specified in DWords. Default value of 600h programs the threshold to 6 KB.	RWS	Yes	600h
31:29	<b>Reserved</b>	RsvdP	No	000b

**Register 13-81. 1D8h Read Pacing Threshold 2****(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Descriptions	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Notes:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports. The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.				
12:0	<b>x4 Port Memory Read Outstanding Threshold</b> Specified in DWords. Default value of 400h programs the threshold to 4 KB.	RWS	Yes	400h
15:13	<b>Reserved</b>	RsvdP	No	000b
28:16	<b>x2 Port Memory Read Outstanding Threshold</b> Specified in DWords. Default value of 200h programs the threshold to 2 KB.	RWS	Yes	200h
31:29	<b>Reserved</b>	RsvdP	No	000b

## 13.16.2 Device-Specific Registers – Captured Bus and Device Numbers (Offsets 1DCh – 1E4h)

This section details the Device-Specific Captured Bus and Device Numbers register. [Table 13-19](#) defines the register map.

**Table 13-19. Device-Specific Captured Bus and Device Numbers Register Map (All Ports)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
Captured Bus and Device Numbers		1DCh
Reserved		1E0h
Factory Test Only		1E4h

**Register 13-82. 1DCh Captured Bus and Device Numbers (All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Captured Bus Number</b> Captured Bus Number value for the Port. The value of this field can be overwritten, if bit 31 ( <i>Captured BusDev Number Override</i> ) is Set prior to changing the Captured Bus Number. <i>Note: Overwriting the Captured Bus Number value is not recommended.</i>		RW	Yes	00h
12:8	<b>Captured Device Number</b> Captured Device Number value for the Port. The value of this field can be overwritten, if bit 31 ( <i>Captured BusDev Number Override</i> ) is Set prior to changing the Captured Device Number. <i>Note: Overwriting the Captured Device Number value is not recommended.</i>	Upstream	RW	Yes	0-0h
		Downstream	RO	No	0-0h
30:13	Reserved		RsvdP	No	0-0h
31	<b>Captured BusDev Number Override</b> 1 = Enables the Captured Bus Number and Device Number to be overridden		RW	Yes	0



### 13.16.3 Device-Specific Registers – Read Pacing (Offset 1E8h)

*Note: Source Queuing and Read Pacing should not be concurrently enabled. The two features are incompatible and enabling both concurrently can result in Fatal errors.*

This section details one of the Device-Specific Read Pacing registers, at offset 1E8h. [Table 13-20](#) defines the register map.

Other Device-Specific Read Pacing registers are detailed in [Section 13.16.1, “Device-Specific Registers – Read Pacing \(Offsets 1D0h – 1D8h\).”](#)

Read Pacing is described, in detail, in [Section 8.5, “Read Pacing.”](#)

**Table 13-20. Device-Specific Read Pacing Register Map (Offset 1E8h) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read Pacing Watermark																															

1E8h

**Register 13-83. 1E8h Read Pacing Watermark (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12:0	Memory Read Outstanding Upper Watermark	ROS	No	000h
15:13	<i>Reserved</i>	RsvdP	No	000b
28:16	Memory Read Outstanding Lower Watermark	ROS	No	0-0h
30:29	Memory Read Outstanding Counter Selection	RWS	Yes	00b
31	Memory Read Outstanding Threshold Reset	RW	Yes	0

### 13.16.4 Device-Specific Registers – Physical Layer (Offsets 200h – 25Ch)

This section details the Device-Specific Physical Layer (PHY) registers located at offsets 200h through 25Ch. [Table 13-21](#) defines the register map.

Other Device-Specific PHY registers are detailed in [Section 13.18.2, “Device-Specific Registers – Physical Layer \(Offsets B80h – C88h\).”](#)

**Table 13-21. Device-Specific PHY Register Map**  
(Offsets 200h – 25Ch) (Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																	
Physical Layer Receiver Detect Status																Physical Layer Electrical Idle for Compliance Mask																200h	
Physical Layer Receiver Not Detected Mask																Physical Layer Electrical Idle Detect Mask																204h	
Port Control																																208h	
Physical Layer User Test Pattern, Bytes 0 through 3																																20Ch	
Physical Layer User Test Pattern, Bytes 4 through 7																																210h	
Physical Layer User Test Pattern, Bytes 8 through 11																																214h	
Physical Layer User Test Pattern, Bytes 12 through 15																																218h	
Physical Layer Command and Status																																21Ch	
Physical Layer Function Control																																220h	
Physical Layer Test 0																																224h	
Physical Layer Test 1																								<i>Reserved</i>								228h	
Physical Layer Safety Bits																																22Ch	
<i>Reserved</i>								<i>Factory Test Only</i>								Physical Layer Port Command																230h	
SKIP Ordered-Set Interval																																234h	
SerDes Quad 0 Diagnostic Data																																238h	
SerDes Quad 1 Diagnostic Data																																23Ch	
SerDes Quad 2 Diagnostic Data																																240h	
SerDes Quad 3 Diagnostic Data																																244h	
Target Link Width																																248h	
<i>Factory Test Only</i>																																24Ch –	250h
Physical Layer Additional Status																																254h	
<i>Factory Test Only/Reserved</i>																Physical Layer Additional Control																258h	
<i>Reserved</i>								<i>Factory Test Only</i>								Physical Layer Error Injection Control																25Ch	

**Register 13-84. 200h Physical Layer Receiver Detect Status and Electrical Idle for Compliance Mask (Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	SerDes <i>x</i> /Lane <i>x</i>	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register is used for specifying the pre-determined quantity of Lanes that detected a Receiver during an LTSSM <i>Detect</i> state, but never detected an exit from Electrical Idle. Because the PEX 8748 has multiple Port configurations, a Mask register is used, rather than programming a specific quantity. When multiple bits are Set, and they correspond to Lanes that belong to the same Port, any of those specified Lanes can cause entry into the LTSSM <i>Polling.Compliance</i> substate.</p> <p><b>Notes:</b> The <i>SerDes x/Lane x</i> column is provided to indicate the SerDes/Lane Number associated with bits/fields that include “SerDes <i>x</i>” or “Lane <i>x</i>” in their name.</p> <p><i>Table 13-5</i> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
<b>Physical Layer Electrical Idle for Compliance Mask</b>					
This register allows masking that specifies which Lanes must never exit Electrical Idle, for entry to the LTSSM <i>Polling.Compliance</i> substate to occur.					
0	<b>Electrical Idle on SerDes <i>x</i> Causes Entry to Compliance State</b>  When all the bits are Cleared, the LTSSM <i>Polling.Compliance</i> substate cannot be entered, due to the Electrical Idle condition.  1 = Lane must have detected a Receiver during an LTSSM <i>Detect</i> state, and must not see an exit from Electrical Idle during the <i>Polling.Active</i> substate, to cause entry to the <i>Polling.Compliance</i> substate	0, 16, or 32	RWS	Yes	1
1		1, 17, or 33	RWS	Yes	1
2		2, 18, or 34	RWS	Yes	1
3		3, 19, or 35	RWS	Yes	1
4		4, 20, or 36	RWS	Yes	1
5		5, 21, or 37	RWS	Yes	1
6		6, 22, or 38	RWS	Yes	1
7		7, 23, or 39	RWS	Yes	1
8		8, 24, or 40	RWS	Yes	1
9		9, 25, or 41	RWS	Yes	1
10		10, 26, or 42	RWS	Yes	1
11		11, 27, or 43	RWS	Yes	1
12		12, 28, or 44	RWS	Yes	1
13		13, 29, or 45	RWS	Yes	1
14		14, 30, or 46	RWS	Yes	1
15		15, 31, or 47	RWS	Yes	1

**Register 13-84. 200h Physical Layer Receiver Detect Status and Electrical Idle for Compliance Mask (Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (Cont.)**

Bit(s)	Description	SerDes <i>x</i> /Lane <i>x</i>	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Physical Layer Receiver Detect Status</b>					
This register returns the Receiver's LTSSM <i>Detect</i> state status for all Lanes within the Station.					
16	<b>Receiver Detected on Lane <i>x</i></b> Reads back as 1 when a Receiver is detected on the Lane.	0, 16, or 32	ROS	No	Set by SerDes
17		1, 17, or 33	ROS	No	Set by SerDes
18		2, 18, or 34	ROS	No	Set by SerDes
19		3, 19, or 35	ROS	No	Set by SerDes
20		4, 20, or 36	ROS	No	Set by SerDes
21		5, 21, or 37	ROS	No	Set by SerDes
22		6, 22, or 38	ROS	No	Set by SerDes
23		7, 23, or 39	ROS	No	Set by SerDes
24		8, 24, or 40	ROS	No	Set by SerDes
25		9, 25, or 41	ROS	No	Set by SerDes
26		10, 26, or 42	ROS	No	Set by SerDes
27		11, 27, or 43	ROS	No	Set by SerDes
28		12, 28, or 44	ROS	No	Set by SerDes
29		13, 29, or 45	ROS	No	Set by SerDes
30		14, 30, or 46	ROS	No	Set by SerDes
31		15, 31, or 47	ROS	No	Set by SerDes

**Register 13-85. 204h Electrical Idle Detect/Receiver Detect Mask**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	SerDes x	Type	Serial EEPROM and I <sup>2</sup> C	Default
Masking Electrical Idle detect will not affect the inferred Electrical Idle detection.					
<i>Notes: Use this register with caution.</i>					
<i>The SerDes x column is provided to indicate the Port Number associated with bits/fields that include “SerDes x” in their name. Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes.</i>					
<i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i>					
<b>Physical Layer Electrical Idle Detect Mask</b>					
Never Detect Electrical Idle mask. This register allows masking of the Electrical Idle Detect function, on a per-SerDes basis. When the bits in this register are Set, the Lane’s Electrical Idle Condition flag does not assert, regardless of the actual presence of Electrical Idle. Masking Electrical Idle detect does not affect the inferred Electrical Idle detection.					
0	<b>SerDes x Mask Electrical Idle Detect</b> 1 = Masks the Electrical Idle Detect for the SerDes, by Station – the corresponding Lane will never detect Electrical Idle	0, 16, or 32	RWS	Yes	0
1		1, 17, or 33	RWS	Yes	0
2		2, 18, or 34	RWS	Yes	0
3		3, 19, or 35	RWS	Yes	0
4		4, 20, or 36	RWS	Yes	0
5		5, 21, or 37	RWS	Yes	0
6		6, 22, or 38	RWS	Yes	0
7		7, 23, or 39	RWS	Yes	0
8		8, 24, or 40	RWS	Yes	0
9		9, 25, or 41	RWS	Yes	0
10		10, 26, or 42	RWS	Yes	0
11		11, 27, or 43	RWS	Yes	0
12		12, 28, or 44	RWS	Yes	0
13		13, 29, or 45	RWS	Yes	0
14		14, 30, or 46	RWS	Yes	0
15		15, 31, or 47	RWS	Yes	0

**Register 13-85. 204h Electrical Idle Detect/Receiver Detect Mask**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	SerDes x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Physical Layer Receiver Not Detected Mask</b>					
Always Detect a Receiver mask. This register allows masking of the Receiver Detect function, on a per SerDes basis. When the bits in this register are Set, the PHY functions as if the Lane detected a Receiver, regardless of the actual presence of a Receiver.					
16	<b>SerDes x Mask Receiver Not Detected</b> 1 = Masks the Receiver Not Detected for the SerDes, by Station – the corresponding Lane will always detect a Receiver	0, 16, or 32	RWS	Yes	0
17		1, 17, or 33	RWS	Yes	0
18		2, 18, or 34	RWS	Yes	0
19		3, 19, or 35	RWS	Yes	0
20		4, 20, or 36	RWS	Yes	0
21		5, 21, or 37	RWS	Yes	0
22		6, 22, or 38	RWS	Yes	0
23		7, 23, or 39	RWS	Yes	0
24		8, 24, or 40	RWS	Yes	0
25		9, 25, or 41	RWS	Yes	0
26		10, 26, or 42	RWS	Yes	0
27		11, 27, or 43	RWS	Yes	0
28		12, 28, or 44	RWS	Yes	0
29		13, 29, or 45	RWS	Yes	0
30		14, 30, or 46	RWS	Yes	0
31		15, 31, or 47	RWS	Yes	0

**Register 13-86. 208h Port Control****(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register is used to disable or enable the LTSSM within individual Ports. The Port Control bits are intended to be used in lieu of placing the Port into the <i>Loopback.Active</i> substate as a Loopback Master. These bits enable the test patterns to be transmitted, with or without a device attached at the far end. Recommended usage is as follows:</p> <ol style="list-style-type: none"> <li>Set the Port's <i>Disable Port x</i> and <i>Hold Port x Quiet</i> bits (bits [3:0 and 11:8], respectively). <ul style="list-style-type: none"> <li>Setting the Port's <i>Disable Port x</i> bit forces the Port into the <i>Detect.Quiet</i> substate. If no device is attached, it is not necessary to Set the Port's <i>Disable Port x</i> bit.</li> <li>If 8.0 GT/s is needed, also Set the Port's <i>Port x Test Pattern x Rate</i> bit (bits [25:24]).</li> </ul> </li> <li>If Set, Clear the Port's <i>Disable Port x</i> bit.</li> <li>Load the UTP registers and enable UTP transmission, or just enable PRBS transmission.</li> </ol> <p><b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name. <i>Table 13-5</i> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
0	<b>Disable Port x</b> While the Port is disabled, Receiver termination is disabled and the SerDes that belong to the disabled Port are placed into the L1 Link PM state.	0, 8, 16	RWS	Yes	0
1	<b>Note:</b> Software should not Set a Port's <i>Disable Port x</i> bit while that Port's Link is in the L2/L3 Ready Link PM state. 0 = Enables the Port's Link Training operation.	1, 9, 17	RWS	Yes	0
2	1 = LTSSM remains in the <i>Detect.Quiet</i> substate on the Port if it is currently in, or returns to, that substate. Unconditionally disables the Port. This is different from an LTSSM <i>Disabled</i> state, in that the Port does not attempt to enter this state. If the Port is idle, it ceases attempting to detect a Receiver. If the Port is up, it immediately returns to the <i>Detect.Quiet</i> substate and remains there. No EIOS is sent, which could force any connected device to the <i>Recovery</i> state, and then to an LTSSM <i>Detect</i> state. The Port remains disabled until its <i>Disable Port x</i> bit is Cleared.	2, 10, 18	RWS	Yes	0
3		3, 11, 19	RWS	Yes	0
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>Hold Port x Quiet</b> 0 = No effect on the LTSSM.	0, 8, 16	RWS	Yes	0
9	1 = Port remains in the <i>Detect.Quiet</i> substate once it returns there. These bits do not force the LTSSM into the <i>Detect.Quiet</i> substate. (Refer to the <i>Disable Port x</i> bits.)	1, 9, 17	RWS	Yes	0
10	After entering the <i>Detect.Quiet</i> substate, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. The Port can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> substate.	2, 10, 18	RWS	Yes	0
11	<b>Note:</b> Use these bits when it is necessary to transmit some data pattern, without first entering the <i>Loopback.Active</i> substate as a Loopback Master.	3, 11, 19	RWS	Yes	0

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 13-86. 208h Port Control**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (*Cont.*)

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b
16	<b>Port x Bypass UTP Alignment Pattern</b> Must be Set when the following conditions are met: <ul style="list-style-type: none"> <li>Link width of the Port being tested is wider than x4, and</li> <li>Port being tested is a Loopback Master transmitting the User Test Pattern, and</li> <li>Loopback Slave is in a different clock domain</li> </ul> 0 = UTP Transmitter continuously transmits the UTP alignment pattern (K28.5 D3.2 D18.2 D13.2), to allow the corresponding symbol alignment logic to find the symbol boundaries before sending the sync pattern (D3.2 D18.2 D13.2 D17.1) and programmed UTP. 1 = UTP transmitter sends one UTP alignment pattern (K28.5 D3.2 D18.2 D13.2), one sync pattern (D3.2 D18.2 D13.2 D17.1), and then the programmed UTP.	0, 8, 16	RWS	Yes	0
17		1, 9, 17	RWS	Yes	0
18		2, 10, 18	RWS	Yes	0
19		3, 11, 19	RWS	Yes	0
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
23:22	<b>Reserved</b>		RsvdP	No	00b
24	<b>Port x Test Pattern x Rate</b> The Port transmits the selected test pattern (PRBS or UTP) at 8.0 GT/s, if the Port's <i>Hold Port x Quiet</i> bit (bits [11:8]) is also Set (manual rate selection is enabled only when the Port's <i>Hold Port x Quiet</i> bit is Set). Set when the Link must transmit at Gen 3 Link speed.  0 = UTP is transmitted at the Link's current speed 1 = UTP is transmitted at a rate of 8.0 GT/s  <i>Note: If the Port's Port x Bypass UTP Alignment Pattern bit (bits [19:16]) is Set, this bit (for that Port) cannot be used.</i>	0, 1, 8, 9, 16, 17	RWS	Yes	0
25		2, 3, 10, 11, 18, 19	RWS	Yes	0
26	<b>Factory Test Only</b>		RsvdP	No	0
27	<b>Reserved</b>		RsvdP	No	0
28	<b>Test Pattern x Rate Port x Select</b> When Set along with the Port's <i>Hold Port x Quiet</i> bit (bits [11:8]), the selected test pattern is transmitted at 5.0 GT/s, on the selected Port. Only values 000b through 101b are valid. If 110b or 111b is written to this field, uninitialized values are returned in the Port's <i>Port x Test Pattern x Rate</i> bit (bits [11:8]) when this register is read.	0, 1, 8, 9, 16, 17	RWS	Yes	0
29		2, 3, 10, 11, 18, 19	RWS	Yes	0
30	<b>Factory Test Only</b>		RsvdP	No	0
31	<b>Reserved</b>		RsvdP	No	0



**Register 13-87. 20Ch Physical Layer User Test Pattern, Bytes 0 through 3**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>UTP Bytes 0 through 3. Used for Digital Far-End Loopback testing.</p> <p>A 16-byte test pattern can be written to register offsets 20Ch through 218h. When User Test Pattern (UTP) transmission is enabled, Byte 0 of register offset 20Ch is transmitted first and Byte 3 (Byte 15 of the UTP) of register offset 218h is transmitted last. (Refer to <a href="#">Section 17.2.4, “Digital Loopback Master Mode,”</a> for further details.) In Gen 1 and Gen 2 modes, every byte of the UTP can be a Control or Data character. Illegal Control characters can be specified. In Gen 3 mode, the UTP generator sends a Data block (Sync Header 10b), with the UTP register data as the content.</p> <p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
7:0	Byte 0 of the UTP. This is the first byte transferred.	RWS	Yes	00h
15:8	Byte 1 of the UTP.	RWS	Yes	00h
23:16	Byte 2 of the UTP.	RWS	Yes	00h
31:24	Byte 3 of the UTP.	RWS	Yes	00h

**Register 13-88. 210h Physical Layer User Test Pattern, Bytes 4 through 7**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>UTP Bytes 4 through 7. Used for Digital Far-End Loopback testing.</p> <p>A 16-byte test pattern can be written to register offsets 20Ch through 218h. When User Test Pattern (UTP) transmission is enabled, Byte 0 of register offset 20Ch is transmitted first and Byte 3 (Byte 15 of the UTP) of register offset 218h is transmitted last. (Refer to <a href="#">Section 17.2.4, “Digital Loopback Master Mode,”</a> for further details.) In Gen 1 and Gen 2 modes, every byte of the UTP can be a Control or Data character. Illegal Control characters can be specified. In Gen 3 mode, the UTP generator sends a Data block (Sync Header 10b), with the UTP register data as the content.</p> <p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
7:0	Byte 4 of the UTP. This is the fifth byte transferred.	RWS	Yes	00h
15:8	Byte 5 of the UTP.	RWS	Yes	00h
23:16	Byte 6 of the UTP.	RWS	Yes	00h
31:24	Byte 7 of the UTP.	RWS	Yes	00h

**Register 13-89. 214h Physical Layer User Test Pattern, Bytes 8 through 11**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>UTP Bytes 8 through 11. Used for Digital Far-End Loopback testing.</p> <p>A 16-byte test pattern can be written to register offsets 20Ch through 218h. When User Test Pattern (UTP) transmission is enabled, Byte 0 of register offset 20Ch is transmitted first and Byte 3 (Byte 15 of the UTP) of register offset 218h is transmitted last. (Refer to <a href="#">Section 17.2.4, “Digital Loopback Master Mode,”</a> for further details.) In Gen 1 and Gen 2 modes, every byte of the UTP can be a Control or Data character. Illegal Control characters can be specified. In Gen 3 mode, the UTP generator sends a Data block (Sync Header 10b), with the UTP register data as the content.</p> <p><b>Notes:</b> <a href="#">Table 13-5</a> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>				
7:0	Byte 8 of the UTP. This is the ninth byte transferred.	RWS	Yes	00h
15:8	Byte 9 of the UTP.	RWS	Yes	00h
23:16	Byte 10 of the UTP.	RWS	Yes	00h
31:24	Byte 11 of the UTP.	RWS	Yes	00h

**Register 13-90. 218h Physical Layer User Test Pattern, Bytes 12 through 15**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>UTP Bytes 12 through 15. Used for Digital Far-End Loopback testing.</p> <p>A 16-byte test pattern can be written to register offsets 20Ch through 218h. When User Test Pattern (UTP) transmission is enabled, Byte 0 of register offset 20Ch is transmitted first and Byte 3 (Byte 15 of the UTP) of register offset 218h is transmitted last. (Refer to <a href="#">Section 17.2.4, “Digital Loopback Master Mode,”</a> for further details.) In Gen 1 and Gen 2 modes, every byte of the UTP can be a Control or Data character. Illegal Control characters can be specified. In Gen 3 mode, the UTP generator sends a Data block (Sync Header 10b), with the UTP register data as the content.</p> <p><b>Notes:</b> <a href="#">Table 13-5</a> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>				
7:0	Byte 12 of the UTP. This is the thirteenth byte transferred.	RWS	Yes	00h
15:8	Byte 13 of the UTP.	RWS	Yes	00h
23:16	Byte 14 of the UTP.	RWS	Yes	00h
31:24	Byte 15 of the UTP.	RWS	Yes	00h

**Register 13-91. 21Ch Physical Layer Command and Status**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register provides various Command and Status bits for Physical Layer operation.</p> <p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
3:0	<p><b>Number of Ports Available in the Station</b></p> <p>Returns the quantity of enabled Ports within this Station, based upon the selected Port configuration.</p>	ROS	No	<p>Defined by STRAP_STNx_PORTCFGx input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station x</i> field(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6, 5:3, and/or 2:0])</p>
4	<p><b>Upstream Cross-Link Enable</b></p> <p>0 = Disables Upstream cross-link, Upstream Port(s) cannot be connected to other Upstream Port(s)  1 = Enables Upstream cross-link, Upstream Port(s) can be connected to other Upstream Port(s)</p>	ROS	Yes	1
5	<p><b>Downstream Cross-Link Enable</b></p> <p>0 = Disables Downstream cross-link, Downstream Ports cannot be connected to other Downstream Ports  1 = Enables Downstream cross-link, Downstream Ports can be connected to other Downstream Ports</p>	ROS	Yes	1
6	<p><b>Lane Reversal Disable</b></p> <p>0 = Enables Lane reversal on all Ports  1 = Disables Lane reversal on all Ports</p>	ROS	Yes	0
7	<b>Reserved</b>	RsvdP	No	0

**Register 13-91. 21Ch Physical Layer Command and Status**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:8	<b>LTSSM Watch Dog Timer Disable</b> Disables the Port's LTSSM Watchdog Timer.	RWS	Yes	00h
31:16	<b>User Test Pattern Control/Data</b> The UTP generators send out a set of 16 bytes of User Programmable data. For Gen 1 and Gen 2 Link speeds (2.5 and 5.0 GT/s, respectively), a k-code bit can be Set for each byte. Bit 16 corresponds to Byte 0 of the User Test Pattern. ... Bit 31 corresponds to Byte 15 of the User Test Pattern. 0 = Corresponding byte of the User Test Pattern is transmitted as a Data character 1 = Corresponding byte of the User Test Pattern is transmitted as a Control character  <i><b>Notes: Use caution when Setting bits within this field, because UTP logic does not check the validity of Control characters.</b></i>  <i>In Gen 3 mode, the UTP generator sends a Data block (Sync Header 10b), with the UTP register data as the content.</i>	RWS	Yes	0000h

**Register 13-92. 220h Physical Layer Function Control**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register allows for the configuration of various functions within the PHY logic.</p> <p><b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name. <a href="#">Table 13-5</a> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
3:0	<p><b>Configuration Fail Count</b></p> <p>This field is used only if the SPARE2 input is Low, to modify Link training for a non-compliant Gen 1 device, that would otherwise fail Link training if the <i>Data Rate</i> bits in the TS are Set.</p> <p>Specifies the quantity of times that Link training fails (in either of the LTSSM <i>Polling</i> or <i>Configuration</i> states), after which the Port re-starts Link training (from the <i>Detect</i> state), advertising only a Gen 1 data rate.</p> <p>The initial value of this register is determined by the <b>Strap Configuration</b> register <i>Gen 1 Compatible Disable Status</i> bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 46Ch[10]) value, which defaults to the inverse state of the SPARE2 input. If the bit value is 0 when reset de-asserts, the initial value of this field is 1h; otherwise, the initial value is 0h.</p>		RWS	Yes	<p>0h (offset 46Ch[10]=1)</p> <p>1h (offset 46Ch[10]=0)</p>
7:4	<b>Reserved</b>		RsvdP	No	0h
9:8	<p><b>Detect.Quiet Wait Time Select Code [1:0]</b></p> <p>Selects the amount of time to wait during the <i>Detect.Quiet</i> substate, before starting the Receiver Detect operation, when a break from Electrical Idle is detected. If Electrical Idle is detected on all Lanes, the wait time is 12 ms.</p> <p>00b = 0 ms            01b = 4 ms (default)            10b = 8 ms            11b = 12 ms</p>		RWS	Yes	01b
11:10	<b>Reserved</b>		RsvdP	No	00b

**Register 13-92. 220h Physical Layer Function Control**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Elastic Buffer Sync Reset Enable</b> 1 = Enable Elastic Buffer Reset upon entry to the <i>Recovery</i> state (if bit 14 ( <i>Elastic Buffer Sync Reset on Rx Recovery Enable</i> ) is Set) or entry to the <i>Configuration</i> state from the <i>Recovery</i> state for upconfigure		RWS	Yes	1
13	<b>Symbol-Block Align Sync Reset Enable</b> 1 = Enable Block and Symbol Aligner Sync Reset with Elastic Buffer Sync Reset		RWS	Yes	1
14	<b>Elastic Buffer Sync Reset on Rx Recovery Enable</b> 1 = Reset Elastic buffer when Receivers are detected entering the <i>Recovery</i> state		RWS	Yes	1
15	<b>Inferred Electrical Idle Exit Type</b> Selects the method used to detect exit from Electrical Idle, after Electrical Idle has been inferred. 0 = <b>Fast Method</b> – Type 0 Exit mode is used, which uses conventional analog Electrical Idle Exit Detection circuitry 1 = <b>Slow Method</b> – Type 1 Exit mode is used, which uses the Symbol Framing Detection Time Select Code and Inferred Electrical Idle Exit Time Select Code Timers		RWS	Yes	0
16	<b>Port x Electrical Idle Inference on EIOS Receipt Enable</b> Electrical Idle Inference on Electrical Idle Ordered-Set (EIOS) Receipt enable, for the Port.	0, 8, 16	RWS	Yes	0
17	0 = Electrical Idle inference is enabled upon EIOS receipt, if the <b>Physical Layer Electrical Idle Detect Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 204h[15:0]) are Set, for the SerDes associated with the Port	1, 9, 17	RWS	Yes	0
18		2, 10, 18	RWS	Yes	0
19	1 = Electrical Idle will be inferred as soon as an EIOS is received on any Lane of the Port	3, 11, 19	RWS	Yes	0
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
23:22	<b>Reserved</b>		RsvdP	No	00b

**Register 13-92. 220h Physical Layer Function Control**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Port x Electrical Idle Inference Disable</b> 0 = Electrical Idle inference is enabled, if the <b>Physical Layer Electrical Idle Detect Mask</b> register <i>SerDes x Mask Electrical Idle Detect</i> bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 204h[15:0]) are Set, for the SerDes associated with the Port. 1 = Port's overall Electrical Idle inference logic is disabled. Electrical Idle inference during the <i>Recovery.Speed</i> substate is not affected and will continue to operate.	0, 8, 16	RWS	Yes	0
25		1, 9, 17	RWS	Yes	0
26		2, 10, 18	RWS	Yes	0
27		3, 11, 19	RWS	Yes	0
30:28	<b>Factory Test Only</b>		RsvdP	No	000b
31	<b>Reserved</b>		RsvdP	No	0

**Register 13-93. 224h Physical Layer Test 0**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register provides controls to enable various PHY test modes.					
<b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name. Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports. The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.					
0	<b>Port x Timer Test Mode Enable</b> 0 = Normal PHY Timer parameters are used 1 = Millisecond scale timers in the Port’s LTSSM are reduced to microsecond scale	0, 8, 16	RWS	Yes	0
1		1, 9, 17	RWS	Yes	0
2		2, 10, 18	RWS	Yes	0
3		3, 11, 19	RWS	Yes	0
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>SKIP Timer Test Mode Enable</b> 0 = Disables Skip Timer Test mode. 1 = Enables Skip Timer Test mode. SKIP Ordered-Sets are transmitted every 256 symbol times, on all Ports, regardless of the <b>SKIP Ordered-Set Interval</b> register <i>SKIP Ordered-Set Interval in Clocks</i> (Gen 1 and Gen 2 Link speeds) or <i>SKIP Ordered-Set Interval in Blocks</i> (Gen 3 Link speeds) field (offset 234h[11:0 or 27:16], respectively) value.		RW	Yes	0
9	<b>Ignore Compliance Receive TCB</b> Ignore the <i>Compliance Receive</i> Training Control Bit (TCB) field in Training Sets. 1 = Causes the PHY to ignore the <i>Compliance Receive</i> TCB when it is Set in received Training Sets		RWS	Yes	0
10	<b>Analog Loopback Enable</b> 0 = PEX 8748 enters Digital Loopback Slave mode if an external device sends at least two consecutive TS1 Ordered-Sets that have the <i>Loopback</i> bit exclusively Set in the TS1 Training Control symbol. The PEX 8748 then loops back data through the Elastic buffer, and decoder and encoder (8b/10b for Gen 1 and Gen 2 Link speeds, or 128b/130b for Gen 3 Link speed). 1 = When operating as a Loopback Slave, the Loopback point of all Ports will be before the Elastic buffer in the Recovered Receive Clock domain.		RWS	Yes	0
11	<b>Factory Test Only</b>		RW	Yes	0
19:12	<b>Factory Test Only</b>		RWS	Yes	00h
31:20	<b>Reserved</b>		RsvdP	No	000h



**Register 13-94. 228h Physical Layer Test 1**

**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Port x or SerDes Quad x/SerDes Pair x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register provides controls to enable various PHY test modes.</p> <p><b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name. The <b>SerDes Quad x/SerDes Pair x</b> column is provided to indicate the SerDes quad and SerDes pair(s) associated with bits/fields that include “SerDes Quad x/SerDes Pair x” in their name.</p> <p><i>Table 13-5</i> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
7:0	<b>Reserved</b>		RWS	No	00h
8	<b>Port x Serial Loopback Path Enable</b> 1 = Enables serial loopback (NES) on the Port	0, 8, 16	RWS	Yes	0
9		1, 9, 17	RWS	Yes	0
10		2, 10, 18	RWS	Yes	0
11		3, 11, 19	RWS	Yes	0
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b
16	<b>Port x Parallel Loopback Path Enable</b> 1 = Enables the Parallel loopback (LB) path on the Port (analog loopback = 0 LB_PIPE, analog loopback = 1 LB_FEP)	0, 8, 16	RWS	Yes	0
17		1, 9, 17	RWS	Yes	0
18		2, 10, 18	RWS	Yes	0
19		3, 11, 19	RWS	Yes	0
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
23:22	<b>Reserved</b>		RsvdP	No	00b

**Register 13-94. 228h Physical Layer Test 1**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Port x or SerDes Quad x/SerDes Pair x	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>SerDes Quad x Pair x User Test Pattern Enable</b> Enables the User Test Pattern generator for SerDes Pair x, by Station.	SerDes Quad 0 (Lower Pair) SerDes[0, 1]/[16, 17]/[32, 33]	RWS	Yes	0
25	0 = Disables transmission of the 128-bit test pattern 1 = Enables transmission of the 128-bit test pattern ( <b>Physical Layer User Test Pattern, Bytes x through y</b> registers (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offsets <b>20Ch</b> through <b>218h</b> )) on SerDes Pair x, by Station, in Digital Far-End Loopback Master mode	SerDes Quad 0 (Upper Pair) SerDes[2, 3]/[18, 19]/[34, 35]	RWS	Yes	0
26	<b>Notes:</b> UTP transmission should be enabled only when operating as a Loopback Master, or when the LTSSM has returned to the Detect.Quiet substate and the Port's <b>Port Control</b> register <b>Hold Port x Quiet</b> bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <b>208h[11:8]</b> ) is Set. The Port's <b>Port Control</b> register <b>Port x Bypass UTP Alignment Pattern</b> bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <b>208h[19:16]</b> ) must be Set if the Link width of the Port under test is wider than x4, the Port under test is a Loopback Master, and the Loopback Slave is in a different clock domain.	SerDes Quad 1 (Lower Pair) SerDes[4, 5]/[20, 21]/[36, 37]	RWS	Yes	0
27		SerDes Quad 1 (Upper Pair) SerDes[6, 7]/[22, 23]/[38, 39]	RWS	Yes	0
28		SerDes Quad 2 (Lower Pair) SerDes[8, 9]/[24, 25]/[40, 41]	RWS	Yes	0
29		SerDes Quad 2 (Upper Pair) SerDes[10, 11]/[26, 27]/[42, 43]	RWS	Yes	0
30		SerDes Quad 3 (Lower Pair) SerDes[12, 13]/[28, 29]/[44, 45]	RWS	Yes	0
31		SerDes Quad 3 (Upper Pair) SerDes[14, 15]/[30, 31]/[46, 47]	RWS	Yes	0

**Register 13-95. 22Ch Physical Layer Safety Bits**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	<b>Factory Test Only</b>	RWS	Yes	0-0h
5	<b>Reserved</b>	RWS	Yes	0
6	<b>Upconfigure Capability Disable</b> 0 = Upconfigure capability is advertised on all Ports 1 = Upconfigure capability is not advertised on all Ports	RWS	Yes	0
19:7	<b>Reserved</b>	RWS	Yes	0-0h
24:20	<b>Factory Test Only</b>	RWS	Yes	0-0h
27:25	<b>Reserved</b>	RWS	Yes	000b
30:28	<b>Factory Test Only</b>	RWS	Yes	111b
31	<b>Reserved</b>	RWS	Yes	1

**Register 13-96. 230h Physical Layer Port Command**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register provides the Loopback, Scrambler Disable, and Compliance Receive commands, and Ready as Loopback Master status, for each Port.</p> <p><b>Notes:</b> <i>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
0	<p><b>Port 0, 8, 16 Loopback Command</b></p> <p>0 = Port is not enabled to go to the <i>Loopback</i> Master state.</p> <p>1 = Port attempts to enter the <i>Loopback</i> state as a Loopback Master. If this bit is Set before the <i>Configuration</i> state is reached, the <i>Configuration.Linkwidth.Start to Loopback</i> path is used. If this bit is Set later, the <i>Recovery.Idle to Loopback</i> path is used.</p>	RWS	Yes	0
1	<p><b>Port 0, 8, 16 Scrambler Disable Command</b></p> <p>When Set, unconditionally disables the data scramblers on the Port's Lane(s), and causes the <i>Scramble Disable</i> Training Control Bit to be Set in the transmitted Training Sets. There is one bit for each Port in the associated Station.</p> <p>If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> substate.</p> <p>If software Sets this bit when the Link is in the Up state, hardware disables its scrambler without executing the Link Training protocol. This scrambler disable takes effect after the Link passes through <i>Configuration</i> again. The Upstream/Downstream device scrambler will not be disabled.</p> <p>0 = Port's scrambler is enabled</p> <p>1 = Port's scrambler is disabled</p>	RWS	Yes	0
2	<p><b>Port 0, 8, 16 Compliance Receive Command</b></p> <p>0 = When the Port transmits TS1 Ordered-Sets, the <i>Compliance Receive</i> Training Control Bit within these Ordered-Sets is not Set during the <i>Polling.Active</i> nor <i>Loopback.Entry</i> substate</p> <p>1 = When the Port transmits TS1 Ordered-Sets, the <i>Compliance Receive</i> Training Control Bit within these Ordered-Sets is Set during the <i>Polling.Active</i> or <i>Loopback.Entry</i> substate</p>	RWS	Yes	0
3	<p><b>Port 0, 8, 16 Ready as Loopback Master</b></p> <p>Link Training and Status State Machine (LTSSM) established Loopback as a Master for the Port.</p> <p>0 = Port is not in Loopback Master mode.</p> <p>1 = Indicates that the Port has successfully transitioned to the <i>Loopback.Active</i> substate as a Loopback Master. The LTSSM remains in this substate, until bit 0 (<i>Port 0, 8, 16 Loopback Command</i>) is Cleared. This bit is Cleared when the PEX 8748 exits the <i>Loopback.Active</i> substate.</p>	ROS	No	0

**Register 13-96. 230h Physical Layer Port Command**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>Port 1, 9, 17 Loopback Command</b> 0 = Port is not enabled to go to the <i>Loopback</i> Master state. 1 = Port attempts to enter the <i>Loopback</i> state as a Loopback Master. If this bit is Set before the <i>Configuration</i> state is reached, the <i>Configuration.Linkwidth.Start</i> to <i>Loopback</i> path is used. If this bit is Set later, the <i>Recovery.Idle</i> to <i>Loopback</i> path is used.	ROS	Yes	0
5	<b>Port 1, 9, 17 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane(s), and causes the <i>Scramble Disable</i> Training Control Bit to be Set in the transmitted Training Sets. There is one bit for each Port in the associated Station. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> substate. If software Sets this bit when the Link is in the Up state, hardware disables its scrambler without executing the Link Training protocol. This scrambler disable takes effect after the Link passes through <i>Configuration</i> again. The Upstream/Downstream device scrambler will not be disabled. 0 = Port's scrambler is enabled 1 = Port's scrambler is disabled	ROS	Yes	0
6	<b>Port 1, 9, 17 Compliance Receive Command</b> 0 = When the Port transmits TS1 Ordered-Sets, the <i>Compliance Receive</i> Training Control Bit within these Ordered-Sets is not Set during the <i>Polling.Active</i> nor <i>Loopback.Entry</i> substate 1 = When the Port transmits TS1 Ordered-Sets, the <i>Compliance Receive</i> Training Control Bit within these Ordered-Sets is Set during the <i>Polling.Active</i> or <i>Loopback.Entry</i> substate	ROS	Yes	0
7	<b>Port 1, 9, 17 Ready as Loopback Master</b> LTSSM established Loopback as a Master for the Port. 0 = Port is not in Loopback Master mode. 1 = Indicates that the Port has successfully transitioned to the <i>Loopback.Active</i> substate as a Loopback Master. The LTSSM remains in this substate, until bit 4 ( <i>Port 1, 9, 17 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8748 exits the <i>Loopback.Active</i> substate.	ROS	No	0

**Register 13-96. 230h Physical Layer Port Command**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Port 2, 10, 18 Loopback Command</b> 0 = Port is not enabled to go to the <i>Loopback</i> Master state. 1 = Port attempts to enter the <i>Loopback</i> state as a Loopback Master. If this bit is Set before the <i>Configuration</i> state is reached, the <i>Configuration.Linkwidth.Start</i> to <i>Loopback</i> path is used. If this bit is Set later, the <i>Recovery.Idle</i> to <i>Loopback</i> path is used.	ROS	Yes	0
9	<b>Port 2, 10, 18 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane(s), and causes the <i>Scramble Disable</i> Training Control Bit to be Set in the transmitted Training Sets. There is one bit for each Port in the associated Station. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> substate. If software Sets this bit when the Link is in the Up state, hardware disables its scrambler without executing the Link Training protocol. This scrambler disable takes effect after the Link passes through <i>Configuration</i> again. The Upstream/Downstream device scrambler will not be disabled. 0 = Port's scrambler is enabled 1 = Port's scrambler is disabled	ROS	Yes	0
10	<b>Port 2, 10, 18 Compliance Receive Command</b> 0 = When the Port transmits TS1 Ordered-Sets, the <i>Compliance Receive</i> Training Control Bit within these Ordered-Sets is not Set during the <i>Polling.Active</i> nor <i>Loopback.Entry</i> substate 1 = When the Port transmits TS1 Ordered-Sets, the <i>Compliance Receive</i> Training Control Bit within these Ordered-Sets is Set during the <i>Polling.Active</i> or <i>Loopback.Entry</i> substate	ROS	Yes	0
11	<b>Port 2, 10, 18 Ready as Loopback Master</b> LTSSM established Loopback as a Master for the Port. 0 = Port is not in Loopback Master mode. 1 = Indicates that the Port has successfully transitioned to the <i>Loopback.Active</i> substate as a Loopback Master. The LTSSM remains in this substate, until bit 8 ( <i>Port 2, 10, 18 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8748 exits the <i>Loopback.Active</i> substate.	ROS	No	0

**Register 13-96. 230h Physical Layer Port Command**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Port 3, 11, 19 Loopback Command</b> 0 = Port is not enabled to go to the <i>Loopback</i> Master state. 1 = Port attempts to enter the <i>Loopback</i> state as a Loopback Master. If this bit is Set before the <i>Configuration</i> state is reached, the <i>Configuration.Linkwidth.Start</i> to <i>Loopback</i> path is used. If this bit is Set later, the <i>Recovery.Idle</i> to <i>Loopback</i> path is used.	ROS	Yes	0
13	<b>Port 3, 11, 19 Scrambler Disable Command</b> When Set, unconditionally disables the data scramblers on the Port's Lane(s), and causes the <i>Scramble Disable</i> Training Control Bit to be Set in the transmitted Training Sets. There is one bit for each Port in the associated Station. If a serial EEPROM load Sets this bit, the scrambler is disabled in a <i>Configuration.Complete</i> substate. If software Sets this bit when the Link is in the Up state, hardware disables its scrambler without executing the Link Training protocol. This scrambler disable takes effect after the Link passes through <i>Configuration</i> again. The Upstream/Downstream device scrambler will not be disabled. 0 = Port's scrambler is enabled 1 = Port's scrambler is disabled	ROS	Yes	0
14	<b>Port 3, 11, 19 Compliance Receive Command</b> 0 = When the Port transmits TS1 Ordered-Sets, the <i>Compliance Receive</i> Training Control Bit within these Ordered-Sets is not Set during the <i>Polling.Active</i> nor <i>Loopback.Entry</i> substate 1 = When the Port transmits TS1 Ordered-Sets, the <i>Compliance Receive</i> Training Control Bit within these Ordered-Sets is Set during the <i>Polling.Active</i> or <i>Loopback.Entry</i> substate	ROS	Yes	0
15	<b>Port 3, 11, 19 Ready as Loopback Master</b> LTSSM established Loopback as a Master for the Port. 0 = Port is not in Loopback Master mode. 1 = Indicates that the Port has successfully transitioned to the <i>Loopback.Active</i> substate as a Loopback Master. The LTSSM remains in this substate, until bit 12 ( <i>Port 3, 11, 19 Loopback Command</i> ) is Cleared. This bit is Cleared when the PEX 8748 exits the <i>Loopback.Active</i> substate.	ROS	No	0
23:16	<b>Factory Test Only</b>	ROS	No	00h
31:24	<b>Reserved</b>	ROS	No	00h

**Register 13-97. 234h SKIP Ordered-Set Interval**

**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register is used to adjust the distance between SKIP Ordered-Sets.</p> <p><b>Notes:</b> <i>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
11:0	<p><b>SKIP Ordered-Set Interval in Clocks</b>  <b>Valid only for Gen 1 and Gen 2 Link speeds (2.5 and 5.0 GT/s, respectively).</b></p> <p>Specifies the SKIP Ordered-Set interval (in symbol times). When a value of 000h is written, SKIP Ordered-Set transmission is disabled.</p> <p>000h = SKIP Ordered-Set transmission is disabled  49Ch = Minimum interval (1,180 symbol times)  602h = Maximum interval (1,538 symbol times)</p> <p><b>Notes:</b> <i>A high value (such as FFFh) can cause the Link to fail.</i>  <i>For Gen 3 Link speed, refer to field [27:16] (SKIP Ordered-Set Interval in Blocks).</i></p>	RWS	Yes	49Ch
15:12	<b>Reserved</b>	RsvdP	No	0h
27:16	<p><b>SKIP Ordered-Set Interval in Blocks</b>  <b>Valid only for Gen 3 Link speed (8.0 GT/s).</b></p> <p>Specifies the SKIP Ordered-Set interval (in block times). When a value of 000h is written, SKIP Ordered-Set transmission is disabled.</p> <p>000h = SKIP Ordered-Set transmission is disabled  173h = Minimum interval (371 block times)  <b>TBD</b> = Maximum interval (<b>TBD</b> block times)</p> <p><b>Notes:</b> <i>A high value (such as FFFh) can cause the Link to fail.</i>  <i>For Gen 1 and Gen 2 Link speeds, refer to field [11:0] (SKIP Ordered-Set Interval in Clocks).</i></p>	RWS	Yes	173h
31:28	<b>Reserved</b>	RsvdP	No	0h



**Register 13-98. 238h SerDes Quad 0 Diagnostic Data**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>There are four <b>SerDes Quad x Diagnostic Data</b> (Diagnostic Data) registers, one per SerDes quad, at offsets 238h through 244h, in each Station. The Diagnostic Data register contents reflect the performance of the SerDes that are selected by the registers' <i>SerDes Diagnostic Data Select</i> bits (field [25:24]). For example, if offset 23Ch[25:24] is programmed to 10b, the information in that Diagnostic Data register is for SerDes 2 of Quad 1 of that Station (SerDes 6, 22, or 38 in Stations 0, 1, and 2, respectively).</p> <p>This register is used to retrieve Diagnostic Test results for SerDes[0-3]/[16-19]/[32-35].</p> <p><b>Notes:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>				
7:0	<b>UTP Expected Data</b> When User Test Pattern (UTP) is enabled, if an error is detected in the received UTP data, this field returns the expected data.	ROS	No	00h
15:8	<b>UTP Actual Data</b> When UTP is enabled, if an error is detected in the received UTP data, this field returns the actual data that was received.	ROS	No	00h
23:16	<b>UTP/PRBS Error Counter</b> <i>The PRBS function of this field is not used.</i> Receiver Detected flags. Returns the quantity of errors detected by the UTP Data Checker. The Error Counter saturates at 255, and is Cleared when UTP is disabled. To Clear the Counter, disable UTP mode by Clearing the SerDes Quad 0 <b>Physical Layer Test 1</b> register <i>SerDes Quad x Pair x User Test Pattern Enable</i> bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 228h[25:24]).	ROS	No	00h
25:24	<b>SerDes Diagnostic Data Select</b> Used to select the SerDes (SerDes[0-3]/[16-19]/[32-35]) to which the diagnostic data in this SerDes quad pertains. Status selection code for the fields representing RO bits [23:0] of this register. The binary code represents a status selection for one of the SerDes Quad 0 Lanes. The test results for physical device Lanes [0-3]/[16-19]/[32-35] are selected with corresponding binary codes from 0-3. <i>Note: To obtain diagnostic data on all SerDes within the quad, run a test, then cycle these bits.</i>	RW	Yes	00b
29:26	<b>Reserved</b>	ROS	No	0h
30	<b>PRBS Counter/-UTP Counter</b> <i>The PRBS function of this bit is not used.</i> 0 = Indicates that field [23:16] (UTP/PRBS Error Counter) is the UTP Error Counter (diagnostic data is from the UTP Data Checkers) 1 = <b>Reserved</b>	ROS	No	0
31	<b>UTP Sync</b> UTP Sync indicator.	ROS	No	0

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 13-99. 23Ch SerDes Quad 1 Diagnostic Data**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>There are four <b>SerDes Quad x Diagnostic Data</b> (Diagnostic Data) registers, one for each SerDes quad, at offsets <a href="#">238h</a> through <a href="#">244h</a>, in each Station. The contents of the Diagnostic Data registers reflect the performance of the SerDes that are selected by the registers' <i>SerDes Diagnostic Data Select</i> bits (field [25:24]). <i>For example</i>, if field [25:24] of this register is programmed to 10b, the information in that Diagnostic Data register is for SerDes 2 of Quad 1 of that Station (SerDes 6, 22, or 38 in Stations 0, 1, and 2, respectively).</p> <p>This register is used to retrieve Diagnostic Test results for SerDes[4-7]/[20-23]/[36-39].</p> <p><b>Notes:</b> <a href="#">Table 13-5</a> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>				
7:0	<b>UTP Expected Data</b> When User Test Pattern (UTP) is enabled, if an error is detected in the received UTP data, this field returns the expected data.	ROS	No	00h
15:8	<b>UTP Actual Data</b> When UTP is enabled, if an error is detected in the received UTP data, this field returns the actual data that was received.	ROS	No	00h
23:16	<b>UTP/PRBS Error Counter</b> <i>The PRBS function of this field is not used.</i> Receiver Detected flags. Returns the quantity of errors detected by the UTP Data Checker. The Error Counter saturates at 255, and is Cleared when UTP is disabled. To Clear the Counter, disable UTP mode by Clearing the SerDes Quad 1 <b>Physical Layer Test 1</b> register <i>SerDes Quad x Pair x User Test Pattern Enable</i> bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <a href="#">228h</a> [27:26]).	ROS	No	00h
25:24	<b>SerDes Diagnostic Data Select</b> Used to select the SerDes (SerDes[4-7]/[20-23]/[36-39]) to which the diagnostic data in this SerDes quad pertains. Status selection code for the fields representing RO bits [23:0] of this register. The binary code represents a status selection for one of the SerDes Quad 1 Lanes. The test results for physical device Lanes [4-7]/[20-23]/[36-39] are selected with corresponding binary codes from 0-3. <i>Note: To obtain diagnostic data on all SerDes within the quad, run a test, then cycle these bits.</i>	RW	Yes	00b
29:26	<b>Reserved</b>	ROS	No	0h
30	<b>PRBS Counter/-UTP Counter</b> <i>The PRBS function of this bit is not used.</i> 0 = Indicates that field [23:16] ( <i>UTP/PRBS Error Counter</i> ) is the UTP Error Counter (diagnostic data is from the UTP Data Checkers) 1 = <b>Reserved</b>	ROS	No	0
31	<b>UTP Sync</b> UTP Sync indicator.	ROS	No	0

**Register 13-100. 240h SerDes Quad 2 Diagnostic Data**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>There are four <b>SerDes Quad x Diagnostic Data</b> (Diagnostic Data) registers, one for each SerDes quad, at offsets <b>238h</b> through <b>244h</b>, in each Station. The contents of the Diagnostic Data registers reflect the performance of the SerDes that are selected by the registers' <i>SerDes Diagnostic Data Select</i> bits (field [25:24]). <i>For example</i>, if offset <b>23Ch</b>[25:24] is programmed to 10b, the information in that Diagnostic Data register is for SerDes 2 of Quad 1 of that Station (SerDes 6, 22, or 38 in Stations 0, 1, and 2, respectively).</p> <p>This register is used to retrieve Diagnostic Test results for SerDes[8-11]/[24-27]/[40-43].</p> <p><b>Notes:</b> <i>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
7:0	<b>UTP Expected Data</b> When User Test Pattern (UTP) is enabled, if an error is detected in the received UTP data, this field returns the expected data.	ROS	No	00h
15:8	<b>UTP Actual Data</b> When UTP is enabled, if an error is detected in the received UTP data, this field returns the actual data that was received.	ROS	No	00h
23:16	<b>UTP/PRBS Error Counter</b> <i>The PRBS function of this field is not used.</i> Receiver Detected flags. Returns the quantity of errors detected by the UTP Data Checker. The Error Counter saturates at 255, and is Cleared when UTP is disabled. To Clear the Counter, disable UTP mode by Clearing the SerDes Quad 2 <b>Physical Layer Test 1</b> register <i>SerDes Quad x Pair x User Test Pattern Enable</i> bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <b>228h</b> [29:28]).	ROS	No	00h
25:24	<b>SerDes Diagnostic Data Select</b> Used to select the SerDes (SerDes[8-11]/[24-27]/[40-43]) to which the diagnostic data in this SerDes quad pertains. Status selection code for the fields representing RO bits [23:0] of this register. The binary code represents a status selection for one of the SerDes Quad 2 Lanes. The test results for physical device Lanes [8-11]/[24-27]/[40-43] are selected with corresponding binary codes from 0-3. <i>Note: To obtain diagnostic data on all SerDes within the quad, run a test, then cycle these bits.</i>	RW	Yes	00b
29:26	<b>Reserved</b>	ROS	No	0h
30	<b>PRBS Counter/-UTP Counter</b> <i>The PRBS function of this bit is not used.</i> 0 = Indicates that field [23:16] ( <i>UTP/PRBS Error Counter</i> ) is the UTP Error Counter (diagnostic data is from the UTP Data Checkers) 1 = <b>Reserved</b>	ROS	No	0
31	<b>UTP Sync</b> UTP Sync indicator.	ROS	No	0

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 13-101. 244h SerDes Quad 3 Diagnostic Data**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>There are four <b>SerDes Quad x Diagnostic Data</b> (Diagnostic Data) registers, one for each SerDes quad, at offsets <a href="#">238h</a> through 244h, in each Station. The contents of the Diagnostic Data registers reflect the performance of the SerDes that are selected by the registers' <i>SerDes Diagnostic Data Select</i> bits (field [25:24]). <i>For example</i>, if offset <a href="#">23Ch</a>[25:24] is programmed to 10b, the information in that Diagnostic Data register is for SerDes 2 of Quad 1 of that Station (SerDes 6, 22, or 38 in Stations 0, 1, and 2, respectively).</p> <p>This register is used to retrieve Diagnostic Test results for SerDes[12-15]/[28-31]/[44-47].</p> <p><b>Notes:</b> <a href="#">Table 13-5</a> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>				
7:0	<b>UTP Expected Data</b> When User Test Pattern (UTP) is enabled, if an error is detected in the received UTP data, this field returns the expected data.	ROS	No	00h
15:8	<b>UTP Actual Data</b> When UTP is enabled, if an error is detected in the received UTP data, this field returns the actual data that was received.	ROS	No	00h
23:16	<b>UTP/PRBS Error Counter</b> <i>The PRBS function of this field is not used.</i> Receiver Detected flags. Returns the quantity of errors detected by the UTP Data Checker. The Error Counter saturates at 255, and is Cleared when UTP is disabled. To Clear the Counter, disable UTP mode by Clearing the SerDes Quad 3 <b>Physical Layer Test 1</b> register <i>SerDes Quad x Pair x User Test Pattern Enable</i> bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <a href="#">228h</a> [31:30]).	ROS	No	00h
25:24	<b>SerDes Diagnostic Data Select</b> Used to select the SerDes (SerDes[12-15]/[28-31]/[44-47]) to which the diagnostic data in this SerDes quad pertains. Status selection code for the fields representing RO bits [23:0] of this register. The binary code represents a status selection for one of the SerDes Quad 3 Lanes. The test results for physical device Lanes [12-15]/[28-31]/[44-47] are selected with corresponding binary codes from 0-3. <i>Note: To obtain diagnostic data on all SerDes within the quad, run a test, then cycle these bits.</i>	RW	Yes	00b
29:26	<b>Reserved</b>	ROS	No	0h
30	<b>PRBS Counter/-UTP Counter</b> <i>The PRBS function of this bit is not used.</i> 0 = Indicates that field [23:16] ( <i>UTP/PRBS Error Counter</i> ) is the UTP Error Counter (diagnostic data is from the UTP Data Checker) 1 = <b>Reserved</b>	ROS	No	0
31	<b>UTP Sync</b> UTP Sync indicator.	ROS	No	0

**Register 13-102. 248h Target Link Width**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register is provided to provide software with the ability to direct Link Width Up/Down configuration.</p> <p>The Port's <i>Target Link Width</i> field in this register is initially loaded with the Port's <i>Negotiated Link Width</i> (offset 78h[25:20]), the first time that the Port's LTSSM transitions from the <i>Configuration</i> state to the L0 Link PM state. The value remains 0 for Ports that are not enabled.</p> <p>Software can be used to retrain a Link to a wider or narrower width than the current width, <i>such as</i> to conserve power when maximum bandwidth is not required. Devices advertise such support, by transmitting the appropriate value for the Data Rate Identifier symbol within TS2 Ordered-Sets.</p> <p>If the Port's <i>Upconfigure Capability Received</i> bit is Set (indicating that during the previous Link training, the Port received Upconfigure Capability notification from the connected device), software can cause the Port to:</p> <ul style="list-style-type: none"> <li>• Upconfigure the Link to a previously negotiated Link width, –or–</li> <li>• Downconfigure the Link to a narrower width</li> </ul> <p>by writing the needed Target Link Width value to this register, followed by Setting the Port's <i>Link Control</i> register <i>Retrain Link</i> bit (Downstream Ports, offset 78h[5]). If the Target Link Width is not equal to the current Link width, the LTSSM transitions from <i>Recovery</i> to <i>Configuration</i>, then renegotiates the Link width.</p> <p><b>Notes:</b> <i>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
4:0	<p><b>Port 0, 8, 16 Target Link Width</b></p> <p>Provides the capability to allow software to cause Link retraining to a wider or narrower width than the current width, <i>such as</i> to conserve power when maximum bandwidth is not required. Devices advertise such support, by transmitting the appropriate value for the Data Rate Identifier symbol within TS2 Ordered-Sets.</p> <p>Written with the Port's Target Link width, to support Link Width Upconfiguration. The initial value of this field is the Port's <i>Negotiated Link Width</i> (offset 78h[25:20]).</p>	RWS	No	Defined by STRAP_STNx_PORTCFGx input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station x</i> field(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6, 5:3, and/or 2:0])
6:5	<b>Reserved</b>	RsvdP	No	00b
7	<p><b>Port 0, 8, 16 Upconfigure Capability Received</b></p> <p>Set during Link training, if the Port received an Upconfigure Capability notification from the connected device.</p> <p>0 = Device connected to the Port does not indicate that it is capable of Link Width Upconfiguration.</p> <p>1 = Device connected to the Port indicates that it is capable of Link Width Upconfiguration. Software can cause the Port to either upconfigure the Link to a previously negotiated Link width, or downconfigure the Link to a narrower width, by writing the needed Target Link Width value to this register, followed by Setting the Port's <i>Link Control</i> register <i>Retrain Link</i> bit (Downstream Ports, offset 78h[5]).</p>	ROS	No	0

**Register 13-102. 248h Target Link Width**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
11:8	<b>Port 1, 9, 17 Target Link Width</b> Provides the capability to allow software to cause Link retraining to a wider or narrower width than the current width, <i>such as</i> to conserve power when maximum bandwidth is not required. Devices advertise such support, by transmitting the appropriate value for the Data Rate Identifier symbol within TS2 Ordered-Sets. Written with the Port's Target Link width, to support Link Width Upconfiguration. The initial value of this field is the Port's <a href="#">Negotiated Link Width</a> (offset 78h[25:20]).	RWS	No	Defined by STRAP_STNx_PORTCFGx input states, or programmed by serial EEPROM value for the <a href="#">Port Configuration</a> register <i>Port Configuration for Station x</i> field(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6, 5:3, and/or 2:0])
14:12	<b>Reserved</b>	RsvdP	No	000b
15	<b>Port 1, 9, 17 Upconfigure Capability Received</b> Set during Link training, if the Port received an Upconfigure Capability notification from the connected device. 0 = Device connected to the Port does not indicate that it is capable of Link Width Upconfiguration. 1 = Device connected to the Port indicates that it is capable of Link Width Upconfiguration. Software can cause the Port to either upconfigure the Link to a previously negotiated Link width, or downconfigure the Link to a narrower width, by writing the needed Target Link Width value to this register, followed by Setting the Port's <a href="#">Link Control</a> register <a href="#">Retrain Link</a> bit (Downstream Ports, offset 78h[5]).	ROS	No	0
18:16	<b>Port 2, 10, 18 Target Link Width</b> Provides the capability to allow software to cause Link retraining to a wider or narrower width than the current width, <i>such as</i> to conserve power when maximum bandwidth is not required. Devices advertise such support, by transmitting the appropriate value for the Data Rate Identifier symbol within TS2 Ordered-Sets. Written with the Port's Target Link width, to support Link Width Upconfiguration. The initial value of this field is the Port's <a href="#">Negotiated Link Width</a> (offset 78h[25:20]).	RWS	No	Defined by STRAP_STNx_PORTCFGx input states, or programmed by serial EEPROM value for the <a href="#">Port Configuration</a> register <i>Port Configuration for Station x</i> field(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6, 5:3, and/or 2:0])
22:19	<b>Reserved</b>	RsvdP	No	0h
23	<b>Port 2, 10, 18 Upconfigure Capability Received</b> Set during Link training, if the Port received an Upconfigure Capability notification from the connected device. 0 = Device connected to the Port does not indicate that it is capable of Link Width Upconfiguration. 1 = Device connected to the Port indicates that it is capable of Link Width Upconfiguration. Software can cause the Port to either upconfigure the Link to a previously negotiated Link width, or downconfigure the Link to a narrower width, by writing the needed Target Link Width value to this register, followed by Setting the Port's <a href="#">Link Control</a> register <a href="#">Retrain Link</a> bit (Downstream Ports, offset 78h[5]).	ROS	No	0



**Register 13-102. 248h Target Link Width**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (Cont.)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26:24	<b>Port 3, 11, 19 Target Link Width</b> Provides the capability to allow software to cause Link retraining to a wider or narrower width than the current width, <i>such as</i> to conserve power when maximum bandwidth is not required. Devices advertise such support, by transmitting the appropriate value for the Data Rate Identifier symbol within TS2 Ordered-Sets. Written with the Target Link width for the Port, to support Link Width Upconfiguration. The initial value of this field is the Port's <a href="#">Negotiated Link Width</a> (offset 78h[25:20]).	RWS	No	Defined by STRAP_STN <sub>x</sub> _PORTCFG <sub>x</sub> input states, or programmed by serial EEPROM value for the <a href="#">Port Configuration</a> register <i>Port Configuration for Station x</i> field(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6, 5:3, and/or 2:0])
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>Port 3, 11, 19 Upconfigure Capability Received</b> Set during Link training, if the Port received an Upconfigure Capability notification from the connected device. 0 = Device connected to the Port does not indicate that it is capable of Link Width Upconfiguration. 1 = Device connected to the Port indicates that it is capable of Link Width Upconfiguration. Software can cause the Port to either upconfigure the Link to a previously negotiated Link width, or downconfigure the Link to a narrower width, by writing the needed Target Link Width value to this register, followed by Setting the Port's <a href="#">Link Control</a> register <a href="#">Retrain Link</a> bit (Downstream Ports, offset 78h[5]).	ROS	No	0

**Register 13-103. 254h Physical Layer Additional Status**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name. <a href="#">Table 13-5</a> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
0	<b>Port x Loopback Master Entry Failed</b> 1 = Indicates that the Port failed to enter the Loopback state as a Loopback Master, and abandoned the attempt by returning the LTSSM to the Detect state  <i>Note:</i> If this bit and the Port's <b>Physical Layer Port Command</b> register Port x Ready as Loopback Master bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 230h[3, 7, 11, or 15]) are both Set, the Loopback state was entered after the initial failure from the Configuration state.	0, 8, 16	RW1C	Yes	0
1		1, 9, 17	RW1C	Yes	0
2		2, 10, 18	RW1C	Yes	0
3		3, 11, 19	RW1C	Yes	0
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
9:8	<b>Link Speed</b> Reflects the STRAP_GEN1_GEN2 input state, which indicates the Link speed.		ROS	No	STRAP_GEN1_GEN2 input state
	<b>Field Value</b>	<b>STRAP_GEN1_GEN2 Input State</b>			
	00b	0			
	01b	Z			
	10b	1			
	11b	–			<b>Reserved</b>
11:10	<b>Reserved</b>		RsvdP	No	00b



**Register 13-103. 254h Physical Layer Additional Status****(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
14:12	<b>Link Speed Status Port Select</b> Selects the Port within the Station for which to return Link speed status, in the <i>Link Speed</i> bits (bits [9:8]).  000b = Port 0 of the Station 001b = Port 1 of the Station 010b = Port 2 of the Station 011b = Port 3 of the Station  All other encodings are <b>Reserved</b> .  <i>Note: If the Port Number indicated is not enabled within the Station, the associated encoding is Reserved.</i>		RW	No	000b
15	<b>Reserved</b>		RsvdP	No	0
22:16	<b>Received Modified Compliance Error Counter</b> Returns the value received in the Modified Compliance Pattern's <i>ERR</i> field, for the Lane(s) selected by the <i>Port x Received Modified Compliance Lane Select</i> bits (bits [27:24]).		RO	No	0-0h
23	<b>Received Modified Compliance Pattern Lock</b> 1 = Indicates that the Modified Compliance Pattern has been locked onto by the Lane(s) selected by the <i>Port x Received Modified Compliance Lane Select</i> bits (bits [27:24]).		RO	No	0
24	<b>Port x Received Modified Compliance Lane Select</b>  Selects which Lane receives the Modified Compliance Error Counter. Status is read from the <i>Received Modified Compliance Pattern Lock</i> bit and <i>Received Modified Compliance Error Counter</i> field (bits [23 and 22:16], respectively).	0, 8, 16	RW	Yes	0
25		1, 9, 17	RW	Yes	0
26		2, 10, 18	RW	Yes	0
27		3, 11, 19	RW	Yes	0
31:28	<b>Reserved</b>		RsvdP	No	0h

**Register 13-104. 258h Physical Layer Additional Control**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name. <i>Table 13-5</i> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
0	<b>Port x External Loopback Enable</b>  1 = Allows the Port to reach Link Up status, when receiving its own Training Sets during Link training. It is necessary to Set this bit when a Port’s Receivers are directly connected, externally, to its Transmitters.	0, 8, 16	RWS	No	0
1		1, 9, 17	RWS	No	0
2		2, 10, 18	RWS	No	0
3		3, 11, 19	RWS	No	0
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>Port x 2<sup>nd</sup> Receiver Detect Disable</b>  1 = Prevents the Port from waiting 12 ms and Retrying the Receiver Detect operation, when Receivers are detected on a subset of Lanes after the first Receiver Detect operation. Instead, the LTSSM progresses to the <i>Polling</i> state, and operates only on the Lanes that detected Receivers.	0, 8, 16	RWS	Yes	0
9		1, 9, 17	RWS	Yes	0
10		2, 10, 18	RWS	Yes	0
11		3, 11, 19	RWS	Yes	0
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b
16	<b>Factory Test Only</b>		RWS	Yes	0
17	<b>Reserved</b>		RsvdP	No	0
19:18	<b>Tx Idle Minimum Time Select</b> 00b = 20 ns (default) 01b = 30 ns 10b = 40 ns 11b = 80 ns		RWS	Yes	00b
31:20	<b>Reserved</b>		RsvdP	No	000h

**Register 13-105. 25Ch Physical Layer Error Injection Control**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register provides 1-bit error injection control.</p> <p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
1:0	<b>Port 0, 8, 16 1-Bit Error/DLLP Error Injection</b> 00b = Disabled (default) 01b = 1-bit error every 1 $\mu$ s 10b = 1-bit error every 1 ms 11b = Error is in the next DLLP (this value self-Clears after an error is injected in a DLLP)	RW	Yes	00b
3:2	<b>Port 0, 8, 16 TLP Error Injection</b> 00b = Disabled (default) 01b = Every TLP 10b = Every other TLP 11b = Every third TLP	RW	Yes	00b
5:4	<b>Port 1, 9, 17 1-Bit Error/DLLP Error Injection</b> 00b = Disabled (default) 01b = 1-bit error every 1 $\mu$ s 10b = 1-bit error every 1 ms 11b = Error is in the next DLLP (this value self-Clears after an error is injected in a DLLP)	RW	Yes	00b
7:6	<b>Port 1, 9, 17 TLP Error Injection</b> 00b = Disabled (default) 01b = Every TLP 10b = Every other TLP 11b = Every third TLP	RW	Yes	00b

**Register 13-105. 25Ch Physical Layer Error Injection Control**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
9:8	<b>Port 2, 10, 18 1-Bit Error/DLLP Error Injection</b> 00b = Disabled (default) 01b = 1-bit error every 1 $\mu$ s 10b = 1-bit error every 1 ms 11b = Error is in the next DLLP (this value self-Clears after an error is injected in a DLLP)	RW	Yes	00b
11:10	<b>Port 2, 10, 18 TLP Error Injection</b> 00b = Disabled (default) 01b = Every TLP 10b = Every other TLP 11b = Every third TLP	RW	Yes	00b
13:12	<b>Port 3, 11, 19 1-Bit Error/DLLP Error Injection</b> 00b = Disabled (default) 01b = 1-bit error every 1 $\mu$ s 10b = 1-bit error every 1 ms 11b = Error is in the next DLLP (this value self-Clears after an error is injected in a DLLP)	RW	Yes	00b
15:14	<b>Port 3, 11, 19 TLP Error Injection</b> 00b = Disabled (default) 01b = Every TLP 10b = Every other TLP 11b = Every third TLP	RW	Yes	00b
23:16	<b>Factory Test Only</b>	RW	Yes	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

### 13.16.5 Device-Specific Registers – Serial EEPROM (Offsets 260h – 270h)

This section details the Device-Specific Serial EEPROM registers. [Table 13-22](#) defines the register map.

**Table 13-22. Device-Specific Serial EEPROM Register Map  
(Base mode – Port 0; Virtual Switch mode – Port 0,  
accessible through the Management Port)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																
Status Data from Serial EEPROM								Serial EEPROM Status								Serial EEPROM Control																260h
Serial EEPROM Buffer																																264h
Serial EEPROM Clock Frequency																																268h
Expansion ROM Base Address																Reserved								Serial EEPROM 3rd Address Byte								26Ch
Serial EEPROM CRC Value																																270h

**Register 13-106. 260h Serial EEPROM Status and Control**  
(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM	Default
<b>Serial EEPROM Control</b>				
12:0	<b>EepBlkAddr</b> Serial EEPROM Block Address for 32 KB.	RW	Yes	000h
15:13	<b>EepCmd[2:0]</b> Commands to the Serial EEPROM Controller.  000b = <i>Reserved</i> 001b = Data from bits [31:24] ( <b>Status Data from Serial EEPROM</b> register) is written to the serial EEPROM's internal <b>Status</b> register 010b = Write four bytes of data from the <b>EepBuf</b> into the memory location pointed to by field [12:0] ( <i>EepBlkAddr</i> ) 011b = Read four bytes of data from the memory location pointed to by field [12:0] ( <i>EepBlkAddr</i> ) into the <b>EepBuf</b> 100b = Reset Write Enable latch 101b = Data from the serial EEPROM's internal <b>Status</b> register is written to bits [31:24] ( <b>Status Data from Serial EEPROM</b> register) 110b = Set Write Enable latch 111b = <i>Reserved</i>  <i><b>Note:</b> For value of 001b, only bits [31, 27:26] can be written into the serial EEPROM's internal <b>Status</b> register.</i>	RW	Yes	000b

**Register 13-106. 260h Serial EEPROM Status and Control**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM	Default	
Serial EEPROM Status					
17:16	<b>EepPrsnt[1:0]</b> Serial EEPROM Present status.  00b = Not present 01b = Serial EEPROM is present – <a href="#">Validation Signature</a> is verified 10b = <i>Reserved</i> 11b = Serial EEPROM is present – <a href="#">Validation Signature</a> is not verified	ROS	No	00b	
18	<b>EepCmdStatus</b> Serial EEPROM Command status.  0 = Serial EEPROM Command is complete 1 = Serial EEPROM Command is not complete	ROS	No	0	
19	<b>EepCRCErr</b> CRC error detected status.	RW1CS	No	0	
20	<b>EepBlkAddr Upper Bit</b> Serial EEPROM Block Address upper bit 13. Extends the serial EEPROM to 64 KB.	RW	Yes	0	
21	<b>EepAddrWidth Override</b> 0 = Field [23:22] ( <i>EepAddrWidth</i> ) is RO 1 = Field [23:22] ( <i>EepAddrWidth</i> ) is software-writable	RW	Yes	0	
23:22	<b>EepAddrWidth</b> Serial EEPROM Address width. If the addressing width cannot be determined, 00b is returned. A non-zero value is reported only if the <a href="#">Validation Signature (5Ah)</a> is successfully read from the first serial EEPROM location.  This field is usually ROS; however, it is RW if bit 21 ( <i>EepAddrWidth Override</i> ) is Set.  00b = Undetermined 01b = 1 byte 10b = 2 bytes 11b = 3 bytes	Bit 21 = 0	ROS	No	00b
	Bit 21 = 1	RW	No	00b	

**Register 13-106. 260h Serial EEPROM Status and Control****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM	Default					
Status Data from Serial EEPROM <sup>a</sup>									
24	<b>EepRdy</b> Serial EEPROM RDY#. 0 = Serial EEPROM is ready to transmit data 1 = Write cycle is in-progress	RW	Yes	0					
25	<b>EepWen</b> Serial EEPROM Write enable. 0 = Serial EEPROM Write is disabled 1 = Serial EEPROM Write is enabled	RW	Yes	0					
27:26	<b>EepBp[1:0]</b> Serial EEPROM Block-Write Protect bits. Block Protection options protect the top ¼, top ½, or the entire serial EEPROM. PEX 8748 Configuration data is stored in the lower addresses; therefore, when using Block Protection, the entire serial EEPROM should be protected with BP[1:0]=11b.		RW	Yes	00b				
	<b>BP[1:0]</b>	<b>Level</b>				<b>Array Addresses Protected, by Device Size</b>			
						<b>8 KB</b>	<b>16 KB</b>	<b>32 KB</b>	<b>64 KB</b>
	00b	0				None	None	None	None
	01b	1 (top ¼)				1800h – 1FFFh	3000h – 3FFFh	6000h – 7FFFh	–
	10b	2 (top ½)				1000h – 1FFFh	2000h – 3FFFh	4000h – 7FFFh	–
	11b	3 (All)	0000h – 1FFFh	0000h – 3FFFh	0000h – 7FFFh	–			

**Register 13-106. 260h Serial EEPROM Status and Control**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM	Default
30:28	<b>EepWrStatus</b> Serial EEPROM Write status. Value is 000b when the serial EEPROM is not in an internal Write cycle. <i>Note: The definition of this field varies among serial EEPROM manufacturers. Reads of the serial EEPROM's internal <b>Status</b> register can return a value of 000b or 111b, depending upon the serial EEPROM that is used.</i>	RW	No	000b
31	<b>EepWpen</b> Serial EEPROM Write Protect enable. Overrides the internal serial EEPROM Write Protect WP# input and enables/disables Writes to the <b>Serial EEPROM Status</b> register (bits [23:16] of this register): <ul style="list-style-type: none"> <li>When WP#=H or this bit is Cleared, and bit 25 (<i>EepWen</i>) is Set, the <b>Serial EEPROM Status</b> register is writable</li> <li>When WP#=L and this bit is Set, or bit 25 (<i>EepWen</i>) is Cleared, the <b>Serial EEPROM Status</b> register is write-protected</li> </ul> <i>Notes: If the internal serial EEPROM Write Protect WP# input is Low, after software Sets the EepWen bit to write-protect the <b>Serial EEPROM Status</b> register, the EepWen value cannot be Cleared, nor can the <i>EepBp[1:0]</i> field be Cleared to disable Block Protection, until the WP# input is High.</i> <i>This bit is <b>not</b> implemented in certain serial EEPROMs. Refer to the serial EEPROM manufacturer's data sheet.</i>	RW	Yes	0

a. Within the serial EEPROM's internal **Status** register, only bits [31, 27:26] can be written.

**Register 13-107. 264h Serial EEPROM Buffer**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>EepBuf</b> Serial EEPROM RW buffer. Read/Write command to the <b>Serial EEPROM Control</b> register (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 260h) results in a 4-byte Read/Write from/to the serial EEPROM device.	RW	Yes	0000_0000h



**Register 13-108. 268h Serial EEPROM Clock Frequency**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
2:0	<b>EepFreq[2:0]</b> Serial EEPROM clock ( <a href="#">EE_SK</a> ) frequency control.  000b = 1 MHz (default) 001b = 1.98 MHz 010b = 5 MHz 011b = 9.62 MHz 100b = 12.5 MHz 101b = 15.6 MHz 110b = 17.86 MHz 111b = <i>Reserved</i>	RW	Yes	000b
7:3	<i>Reserved</i>	RsvdP	No	0-0h
10:8	<b>EepCsStHld[2:0]</b> CS to SCLK setup and hold timing, provided as a quantity of ½ <a href="#">EE_SK</a> Clock cycles.  000b = Use default timing for <a href="#">EE_CS#</a> setup and <a href="#">EE_CS#</a> hold timing to the serial EEPROM, for <a href="#">EE_CS#</a> active to <a href="#">EE_SK</a> active delay, and <a href="#">EE_SK</a> inactive to <a href="#">EE_CS#</a> inactive delay, respectively 001b = Non-zero value adds that quantity of ½ <a href="#">EE_SK</a> clocks delay to the default setup and hold timing, between <a href="#">EE_CS#</a> active and <a href="#">EE_SK</a> active, and between <a href="#">EE_SK</a> inactive and <a href="#">EE_CS#</a> inactive	RW	Yes	000b
15:11	<i>Reserved</i>	RsvdP	No	0-0h
16	<b>Expansion ROM Size</b> Indicates the size of the NT Port Expansion ROM.  0 = 16 KB 1 = 32 KB	RW	Yes	0
31:17	<i>Reserved</i>	RsvdP	No	0-0h

**Register 13-109. 26Ch Serial EEPROM 3<sup>rd</sup> Address Byte**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Serial EEPROM 3<sup>rd</sup> Address Byte</b> Contains the upper Byte address when reading from or writing to a 3-address byte serial EEPROM.	RW	Yes	00h
15:8	<b>Reserved</b>	RsvdP	No	00h
31:16	<b>Expansion ROM Base Address</b> Indicates the NT Port Expansion ROM Base address within the serial EEPROM. The value is dependent upon the <a href="#">Serial EEPROM Clock Frequency</a> register <a href="#">Expansion ROM Size</a> bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 268h[16]) value. The lower six bits, [21:16], map to serial EEPROM byte Address bits [15:10] (aligned to a 256-DWord (1-KB) boundary). The NT Port Expansion ROM must not straddle a 64-KB boundary within the serial EEPROM.  0020h = Default Base address in serial EEPROM for a 16-KB NT Port Expansion ROM ( <i>Expansion ROM Size</i> bit is Cleared) is 2000h (8 KB). The serial EEPROM size must be at least 32 KB.  0040h = Default Base address in serial EEPROM for a 32-KB NT Port Expansion ROM ( <i>Expansion ROM Size</i> bit is Set) is 4000h (16 KB). The serial EEPROM size must be at least 64 KB.	RW	Yes	0020h

**Register 13-110. 270h Serial EEPROM CRC Value**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>EepCRCValue</b> Calculated serial EEPROM CRC value. There is an optional CRC checking at the end of serial EEPROM auto load. Bit 7 of the second byte (after the <a href="#">5Ah Validation Signature</a> ) determines whether CRC checking is to be performed. The calculated CRC value is always placed into this register after serial EEPROM auto load. If CRC checking is enabled, and the stored 32-bit CRC value read from the end the serial EEPROM content does not match the calculated value, an Uncorrectable Internal error is triggered at Port 0 of Station 0.	HwInit	Yes	FFFF_FFFFh

### 13.16.6 Device-Specific Registers – I<sup>2</sup>C and SMBus Slave Interfaces (Offsets 290h – 2FCh)

This section details the Device-Specific I<sup>2</sup>C and SMBus Slave Interface registers. [Table 13-23](#) defines the register map.

The I<sup>2</sup>C and SMBus Slave Interfaces are described, in detail, in [Chapter 7, “I<sup>2</sup>C/SMBus Slave Interface Operation.”](#)

**Table 13-23. Device-Specific I<sup>2</sup>C and SMBus Slave Interfaces Register Map**  
(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
<i>Factory Test Only</i>																290h															
<a href="#">I2C Configuration</a>																294h															
<i>Factory Test Only</i>																298h – 2C4h															
<a href="#">SMBus Configuration</a>																2C8h															
<i>Reserved</i>																2CCh – 2FCh															

**Register 13-111. 294h I<sup>2</sup>C Configuration**  
(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default								
6:0	<b>Slave Address</b> Default I <sup>2</sup> C Slave/SMBus Device address, which is dependent upon the I2C_ADDR0 input state.	RWS	Yes	Refer to Table								
	<table><tr><th>I2C_ADDR0 Input State</th><th>I<sup>2</sup>C Slave/SMBus Device Address</th></tr><tr><td>0</td><td>0111_000b</td></tr><tr><td>Z</td><td>0111_001b</td></tr><tr><td>1</td><td>0111_010b</td></tr></table>				I2C_ADDR0 Input State	I <sup>2</sup> C Slave/SMBus Device Address	0	0111_000b	Z	0111_001b	1	0111_010b
	I2C_ADDR0 Input State				I <sup>2</sup> C Slave/SMBus Device Address							
	0				0111_000b							
	Z				0111_001b							
	1				0111_010b							
Other values can be programmed as well, by serial EEPROM or Memory Write. Additionally, Silicon Revision CA supports access to this field, through Configuration Requests.												
<i>Note: The I<sup>2</sup>C Slave/SMBus Device address must not be changed by an I<sup>2</sup>C/SMBus Write command.</i>												
9:7	<i>Reserved</i>	RsvdP	No	000b								
10	<i>Factory Test Only</i>	RWS	Yes	0								
30:11	<i>Reserved</i>	RWS	Yes	0-0h								
31	<i>Factory Test Only</i>	ROS	No	0								

**Register 13-112. 2C8h SMBus Configuration****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>SMBus Enable</b> Initially loaded from the <a href="#">STRAP_I2C_SMBUS_EN</a> input state. The value can later be changed by software, serial EEPROM, I <sup>2</sup> C, and/or SMBus. 0 = Disables SMBus for device configuration (I <sup>2</sup> C mode is enabled) 1 = Enables SMBus for device configuration (SMBus mode is enabled)	RWS	Yes	0 (STRAP_I2C_SMBUS_EN=Z or 1) 1 (STRAP_I2C_SMBUS_EN=0)
7:1	<b>SMBus Device Address</b> Defined by the Address Resolution Protocol (ARP) Master, if ARP is enabled. Defaults to 0011_100b if ARP is disabled (STRAP_I2C_SMBUS_EN=0 and bit 8 ( <a href="#">ARP Disable</a> ) is Set). When disabled, the serial EEPROM should simultaneously: <ul style="list-style-type: none"> <li>Set bits [15, 11, and 10 (<a href="#">SMBus Parameter Reload</a>, <a href="#">AR Flag</a>, and <a href="#">AV Flag</a>, respectively), and,</li> <li>Program this field to the SMBus address for this device (<i>such as</i> to 70h, to match the I<sup>2</sup>C Slave Address (<a href="#">I2C Configuration</a> register <a href="#">Slave Address</a> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 294h[6:0])) default value)</li> </ul> An 8 <sup>th</sup> bit is transmitted immediately following the 7-bit address, to signal whether the instruction is Write (0) or Read (1). Therefore, when accessing the default I <sup>2</sup> C Slave address <a href="#">0111_000b</a> , the transmitted byte value is 70h for a Write instruction and 71h for a Read instruction.	RWS	Yes	0000_000b (STRAP_I2C_SMBUS_EN=Z or 1) 0011_100b (STRAP_I2C_SMBUS_EN=0 and bit 8 ( <a href="#">ARP Disable</a> ) is Set)
8	<b>ARP Disable</b> 0 = Device under test is able to respond to ARP commands 1 = Device under test is unable to respond to ARP commands	RWS	Yes	1
9	<b>PEC Check Disable</b> 0 = Enable PEC checking on all packets 1 = Disables Packet Error Checks (PECs) checking on all packets; packets with the wrong PECs are accepted	RWS	Yes	0
10	<b>AV Flag</b> Address Valid (AV) flag. Set, by default, when ARP is disabled (STRAP_I2C_SMBUS_EN=0 and bit 8 ( <a href="#">ARP Disable</a> ) is Set).	RWS	Yes	1
11	<b>AR Flag</b> Address Resolved (AR) flag.	RWS	Yes	0

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

**Register 13-112. 2C8h SMBus Configuration****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
13:12	<b>UDID Address Type</b> Unique Device Identifier (UDID) Address type. 00b = ARP is disabled (STRAP_I2C_SMBUS_EN=0 and bit 8 (ARP Disable) is Set) 10b = ARP is enabled (STRAP_I2C_SMBUS_EN=Z or 1) All other encodings are <i>Reserved</i> .	RWS	Yes	00b (STRAP_I2C_SMBUS_EN=0 and bit 8 (ARP Disable) is Set) 10b (STRAP_I2C_SMBUS_EN=Z or 1)
14	<b>UDID PEC Support</b> 1 = Sets the <i>PEC Support</i> bit in the UDID	RWS	Yes	1
15	<b>SMBus Parameter Reload</b> Software must Set this bit if bits [10, 8, and/or 7:1] (AV Flag, ARP Disable, and/or SMBus Device Address, respectively) at runtime. Effective only when bit 28 (SMBus Command In-Progress) is Cleared.	ROS	No	0
23:16	<b>UDID Vendor-Specific ID</b> Defines the MSB of the UDID Vendor-Specific ID. Bits [23:20] of this field are defined by the I2C_ADDR0 input state, which provides the following ID values. 0 = 7000_0000h Z = B000_0000h 1 = D000_0000h	RWS	Yes	Defined by I2C_ADDR0 input state
26:24	<b>UDID Version</b> UDID version defined for the SMBus v2.0.	RWS	Yes	001b
27	<b>Factory Test Only</b>	RWS	Yes	0
28	<b>SMBus Command In-Progress</b> 0 = SMBus state machine is idle 1 = SMBus state machine is active (not idle)	ROS	No	0
29	<b>PEC Check Failed</b> 0 = PEC check successfully completed when receiving a packet 1 = PEC check failed when receiving a packet	RW1C	No	0
30	<b>Unsupported SMBus Command</b> 0 = Command received from SMBus is a supported command 1 = Command received from SMBus is an unsupported command	RW1C	No	0
31	<b>SMBus Error Detected</b> 0 = No error detected in STOP condition (as generated by the SMBus Master, to terminate the Data transfer) 1 = STOP detected was not on a byte boundary	RW1C	Yes	0

NDA CONFIDENTIAL -- Inventec | King Michael

### 13.16.7 Device-Specific Registers – Port Configuration and Lane Status (Offsets 300h – 350h)

This section details Device-Specific Port Configuration and Lane Status registers at offsets 300h through 350h. [Table 13-24](#) defines the register map.

Other Device-Specific Port Configuration and Lane Status registers are detailed in [Section 13.16.8](#), “Device-Specific Registers – Port Configuration (Offsets 354h – 46Ch).”

**Table 13-24. Device-Specific Port Configuration and Lane Status Register Map (Offsets 300h – 350h) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																
Port Configuration																																300h
Reserved								x1 Port Configuration																								304h
Reserved								x2 Port Configuration																Reserved								308h
Factory Test Only																																30Ch – 310h
Clock Enable																																314h
Factory Test Only																																318h – 32Ch
Station 0/1 Lane Status																																330h
Reserved																Station 2 Lane Status																334h
Factory Test Only																																338h – 350h

**Register 13-113. 300h Port Configuration****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> <a href="#">Table 13-5</a> lists the Port configurations for each Station. The table also indicates which Ports and SerDes/Lanes are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p> <p>In the values listed for each field, the second value is the Port Configuration Number, as listed in <a href="#">Table 13-5</a> (for example, 001b is equivalent to Port Configuration 1).</p>				
2:0	<p><b>Port Configuration for Station 0</b></p> <p>Port Configuration Link width, per Port, for Station 0. The serial EEPROM bit values always override the STRAP_STN0_PORTCFG[1:0] input states (if the serial EEPROM values are loaded; refer to <a href="#">Table 13-5</a>).</p> <p>This register is reset only by a Fundamental Reset (PEX_PERST# assertion).</p> <p>001b = 1 = x16 010b = 2 = x8x8 011b = 3 = x8x4x4 100b = 4 = x4x4x4x4</p> <p>All other encodings are <b>Reserved</b>.</p>	ROS	Yes	Defined by STRAP_STN0_PORTCFG[1:0] input states
5:3	<p><b>Port Configuration for Station 1</b></p> <p>Port Configuration Link width, per Port, for Station 1. The serial EEPROM bit values always override the STRAP_STN1_PORTCFG[1:0] input states (if the serial EEPROM values are loaded; refer to <a href="#">Table 13-5</a>).</p> <p>This register is reset only by a Fundamental Reset (PEX_PERST# assertion).</p> <p>001b = 1 = x16 010b = 2 = x8x8 011b = 3 = x8x4x4 100b = 4 = x4x4x4x4</p> <p>All other encodings are <b>Reserved</b>.</p>	ROS	Yes	Defined by STRAP_STN1_PORTCFG[1:0] input states

**Register 13-113. 300h Port Configuration**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8:6	<p><b>Port Configuration for Station 2</b></p> <p>Port Configuration Link width, per Port, for Station 2. The serial EEPROM bit values always override the STRAP_STN2_PORTCFG[1:0] input states (if the serial EEPROM values are loaded; refer to <a href="#">Table 13-5</a>).</p> <p>This register is reset only by a Fundamental Reset (PEX_PERST# assertion).</p> <p>001b = 1 = x16  010b = 2 = x8x8  011b = 3 = x8x4x4  100b = 4 = x4x4x4x4</p> <p>All other encodings are <b>Reserved</b>.</p>	ROS	Yes	Defined by STRAP_STN2_PORTCFG[1:0] input states
31:9	<b>Reserved</b>	RsvdP	No	0-0h



**Register 13-114. 304h x1 Port Configuration****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register forces the Port to linkup with a x1 negotiated Link width, regardless of the quantity of Lanes connected to the Port.</p> <p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Ports, SerDes modules, and Lanes.</i></p> <p><i>“X” is “Don’t Care.”</i></p>				
7:0	<b>x1 Only for Station 0</b> 0000_0000b = Indicates that none of the Ports are configured as x1 only, and x1 operation is allowed XXXX_XXX1b = Indicates that Port 0 is configured as x1 only XXXX_XX1Xb = Indicates that Port 1 is configured as x1 only XXXX_X1XXb = Indicates that Port 2 is configured as x1 only XXXX_1XXXb = Indicates that Port 3 is configured as x1 only All other encodings are <b>Reserved</b> .	ROS	Yes	0000_0000b
15:8	<b>x1 Only for Station 1</b> 0000_0000b = Indicates that none of the Ports are configured as x1 only, and x1 operation is allowed XXXX_XXX1b = Indicates that Port 8 is configured as x1 only XXXX_XX1Xb = Indicates that Port 9 is configured as x1 only XXXX_X1XXb = Indicates that Port 10 is configured as x1 only XXXX_1XXXb = Indicates that Port 11 is configured as x1 only All other encodings are <b>Reserved</b> .	ROS	Yes	0000_0000b
23:16	<b>x1 Only for Station 2</b> 0000_0000b = Indicates that none of the Ports are configured as x1 only, and x1 operation is allowed XXXX_XXX1b = Indicates that Port 16 is configured as x1 only XXXX_XX1Xb = Indicates that Port 17 is configured as x1 only XXXX_X1XXb = Indicates that Port 18 is configured as x1 only XXXX_1XXXb = Indicates that Port 19 is configured as x1 only All other encodings are <b>Reserved</b> .	ROS	Yes	0000_0000b
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 13-115. 308h x2 Port Configuration****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register forces the Port to linkup with a x2 negotiated Link width, regardless of the quantity of Lanes connected to the Port.</p> <p><b>Notes:</b> <i>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Ports, SerDes modules, and Lanes.</i></p> <p><i>“X” is “Don’t Care.”</i></p>				
7:0	<b>Reserved</b>	RsvdP	No	00h
15:8	<p><b>x2 Only for Station 1</b></p> <p>0000_0000b = Indicates that none of the Ports are configured as x2 only, and x2 operation is allowed</p> <p>XXXX_XXX1b = Indicates that Port 8 is configured as x2 only</p> <p>XXXX_XX1Xb = Indicates that Port 9 is configured as x2 only</p> <p>XXXX_X1XXb = Indicates that Port 10 is configured as x2 only</p> <p>XXXX_1XXXb = Indicates that Port 11 is configured as x2 only</p> <p>All other encodings are <b>Reserved</b>.</p>	ROS	Yes	0000_0000b
23:16	<p><b>x2 Only for Station 2</b></p> <p>0000_0000b = Indicates that none of the Ports are configured as x2 only, and x2 operation is allowed</p> <p>XXXX_XXX1b = Indicates that Port 16 is configured as x2 only</p> <p>XXXX_XX1Xb = Indicates that Port 17 is configured as x2 only</p> <p>XXXX_X1XXb = Indicates that Port 18 is configured as x2 only</p> <p>XXXX_1XXXb = Indicates that Port 19 is configured as x2 only</p> <p>All other encodings are <b>Reserved</b>.</p>	ROS	Yes	0000_0000b
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 13-116. 314h Clock Enable****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>Ports and Stations are automatically enabled, according to the Port configuration defined by the STRAP_STNx_PORTCFGx 3-state inputs, which can be overridden by programming the Station's <b>Port Configuration</b> register <i>Port Configuration for Station x</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6, 5:3, and/or 2:0]).</p> <p>An enabled Port can be selectively disabled, however, by Clearing the Port's <i>Port x Clock Enable</i> bit within this register. Port 0 must always remain enabled, and Ports 8 and 16 (containing Station-specific registers) must remain enabled, if other Ports within their respective Stations are enabled.</p> <p><b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name. <i>Table 13-5</i> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p> <p><i>It is not possible to enable more Ports than the maximum specified for the device.</i></p>					
0	<b>Port x Clock Enable</b> 0 = Disables 1 = Enables	0	RWS	Yes	1
1		1	RWS	Yes	Defined by STRAP_STN0_PORTCFG[1:0] input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station 0</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[2:0])
2		2	RWS	Yes	
3		3	RWS	Yes	
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>Port x Clock Enable</b> 0 = Disables 1 = Enables	8	RWS	Yes	Defined by STRAP_STN1_PORTCFG[1:0] input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station 1</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[5:3])
9		9	RWS	Yes	
10		10	RWS	Yes	
11		11	RWS	Yes	
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b
16	<b>Port x Clock Enable</b> 0 = Disables 1 = Enables	16	RWS	Yes	Defined by STRAP_STN2_PORTCFG[1:0] input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station 2</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6])
17		17	RWS	Yes	
18		18	RWS	Yes	
19		19	RWS	Yes	
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
23:22	<b>Reserved</b>		RsvdP	No	00b

**Register 13-116. 314h Clock Enable****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Station 1 Root Clock Enable</b>		RWS	Yes	Defined by STRAP_STN1_PORTCFG[1:0] input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station 1</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[5:3])
25	<b>Station 2 Root Clock Enable</b>		RWS	Yes	Defined by STRAP_STN2_PORTCFG[1:0] input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station 2</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6])
31:26	<b><i>Factory Test Only/Reserved</i></b>		RsvdP	No	0-0h

**Register 13-117. 330h Station 0/1 Lane Status****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Lane x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Notes: The <b>Lane x</b> column is provided to indicate the Lane Number associated with each bit.</i> <i>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Ports, SerDes modules, and Lanes.</i>					
0	<b>Lane x Up Status</b> 0 = Lane is Down 1 = Lane is Up	0	ROS	Yes	1
1		1	ROS	Yes	1
2		2	ROS	Yes	1
3		3	ROS	Yes	1
4		4	ROS	Yes	1
5		5	ROS	Yes	1
6		6	ROS	Yes	1
7		7	ROS	Yes	1
8		8	ROS	Yes	1
9		9	ROS	Yes	1
10		10	ROS	Yes	1
11		11	ROS	Yes	1
12		12	ROS	Yes	1
13		13	ROS	Yes	1
14		14	ROS	Yes	1
15		15	ROS	Yes	1

**Register 13-117. 330h Station 0/1 Lane Status****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Lane x	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Lane x Up Status</b> 0 = Lane is Down 1 = Lane is Up	16	ROS	Yes	1
17		17	ROS	Yes	1
18		18	ROS	Yes	1
19		19	ROS	Yes	1
20		20	ROS	Yes	1
21		21	ROS	Yes	1
22		22	ROS	Yes	1
23		23	ROS	Yes	1
24		24	ROS	Yes	1
25		25	ROS	Yes	1
26		26	ROS	Yes	1
27		27	ROS	Yes	1
28		28	ROS	Yes	1
29		29	ROS	Yes	1
30		30	ROS	Yes	1
31		31	ROS	Yes	1

**Register 13-118. 334h Station 2 Lane Status****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Lane x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Notes: The <b>Lane x</b> column is provided to indicate the Lane Number associated with each bit.</i> <i>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Ports, SerDes modules, and Lanes.</i>					
0	<b>Lane x Up Status</b> 0 = Lane is Down 1 = Lane is Up	32	ROS	Yes	1
1		33	ROS	Yes	1
2		34	ROS	Yes	1
3		35	ROS	Yes	1
4		36	ROS	Yes	1
5		37	ROS	Yes	1
6		38	ROS	Yes	1
7		39	ROS	Yes	1
8		40	ROS	Yes	1
9		41	ROS	Yes	1
10		42	ROS	Yes	1
11		43	ROS	Yes	1
12		44	ROS	Yes	1
13		45	ROS	Yes	1
14		46	ROS	Yes	1
15		47	ROS	Yes	1
31:16	<b>Reserved</b>		RsvdP	No	0000h

## 13.16.8 Device-Specific Registers – Port Configuration (Offsets 354h – 46Ch)

This section details the Device-Specific Port Configuration registers located at offsets 354h through 46Ch. In particular, these registers are related to the Management Port and Virtual Switches. [Table 13-25](#) defines the register map.

Additional information regarding programming of the Virtual Switch Table, as it relates to these registers, is provided in [Section 5.3.3.2, “Virtual Switch Table Programming Sequence.”](#)

Other Device-Specific Port Configuration registers are detailed in [Section 13.16.7, “Device-Specific Registers – Port Configuration and Lane Status \(Offsets 300h – 350h\).”](#)

**Table 13-25. Device-Specific Port Configuration Register Map (Offsets 354h – 46Ch)**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
<i>Reserved</i> (Base Mode)																																354h
Management Port Control (Virtual Switch Mode)																																
<i>Reserved</i> (Base Mode)																																358h
Virtual Switch Enable (Virtual Switch Mode)																																
<i>Reserved</i>																																35Ch
VS0 Upstream																																360h
<i>Reserved</i> (Base Mode)																																364h
VS1 Upstream (Virtual Switch Mode)																																
<i>Reserved</i> (Base Mode)																																368h
VS2 Upstream (Virtual Switch Mode)																																
<i>Reserved</i> (Base Mode)																																36Ch
VS3 Upstream (Virtual Switch Mode)																																
<i>Reserved</i> (Base Mode)																																370h
VS4 Upstream (Virtual Switch Mode)																																
<i>Reserved</i> (Base Mode)																																374h
VS5 Upstream (Virtual Switch Mode)																																
<i>Factory Test Only</i>																																378h – 37Ch



**Table 13-25. Device-Specific Port Configuration Register Map  
(Offsets 354h – 46Ch) (Cont.)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																						
VS0 Port Vector																																						380h
<i>Reserved</i> (Base Mode) VS1 Port Vector (Virtual Switch Mode)																																						384h
<i>Reserved</i> (Base Mode) VS2 Port Vector (Virtual Switch Mode)																																						388h
<i>Reserved</i> (Base Mode) VS3 Port Vector (Virtual Switch Mode)																																						38Ch
<i>Reserved</i> (Base Mode) VS4 Port Vector (Virtual Switch Mode)																																						390h
<i>Reserved</i> (Base Mode) VS5 Port Vector (Virtual Switch Mode)																																						394h
<i>Factory Test Only</i>																																				398h –	39Ch	
Port Reset																																						3A0h
<i>Reserved</i>								Parallel Hot Plug Control																														3A4h
VSx_PERST# and NT_PERST# Status																																						3A8h
Configuration Release																																						3ACh
<i>Factory Test Only/Reserved</i>																																				3B0h –	468h	
Strap Configuration																																						46Ch

**Register 13-119. 354h Management Port Control**  
**(Virtual Switch mode – Port 0, accessible through the Management Port and Redundant Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default																																							
<p>This register can be accessed by Management-capable Ports (Active and Redundant Management Ports), Strapping balls, and/or the I<sup>2</sup>C Bus. The <b>Reserved</b> register bits return zeros (0) during Reads. Writes to <b>Reserved</b> register bits do not affect the register. This register, in combination with the <b>VSx Upstream</b> and <b>VSx Port Vector</b> registers (offsets 360h through 374h, and 380h through 394h, respectively), comprise the Virtual Switch Table. For further details, refer to <a href="#">Section 5.3.3, “Virtual Switch Table.”</a></p> <p><b>Note:</b> <a href="#">Table 13-5</a> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>																																											
4:0	<p><b>Active Management Port</b></p> <p>Indicates the Port Number of the Active Management Port. The value of this field is latched in, upon reset de-assertion, from the STRAP_UPSTRM_PORTSEL[2:0] input states.</p> <table><thead><tr><th>Field Value</th><th>Strapping Input States</th><th>Port Number</th></tr></thead><tbody><tr><td>0_0000b</td><td>000</td><td>0</td></tr><tr><td>0_0001b</td><td>00Z</td><td>1</td></tr><tr><td>0_0010b</td><td>001</td><td>2</td></tr><tr><td>0_0011b</td><td>0Z0</td><td>3</td></tr><tr><td>0_1000b</td><td>011</td><td>8</td></tr><tr><td>0_1001b</td><td>Z00</td><td>9</td></tr><tr><td>0_1010b</td><td>Z0Z</td><td>10</td></tr><tr><td>0_1011b</td><td>Z01</td><td>11</td></tr><tr><td>1_0000b</td><td>Z1Z</td><td>16</td></tr><tr><td>1_0001b</td><td>Z11</td><td>17</td></tr><tr><td>1_0010b</td><td>100</td><td>18</td></tr><tr><td>1_0011b</td><td>10Z</td><td>19</td></tr></tbody></table> <p>All other encodings are <b>Reserved</b>.</p>	Field Value	Strapping Input States	Port Number	0_0000b	000	0	0_0001b	00Z	1	0_0010b	001	2	0_0011b	0Z0	3	0_1000b	011	8	0_1001b	Z00	9	0_1010b	Z0Z	10	0_1011b	Z01	11	1_0000b	Z1Z	16	1_0001b	Z11	17	1_0010b	100	18	1_0011b	10Z	19	RWS	Yes	Defined by STRAP_UPSTRM_PORTSEL[2:0] input states
Field Value	Strapping Input States	Port Number																																									
0_0000b	000	0																																									
0_0001b	00Z	1																																									
0_0010b	001	2																																									
0_0011b	0Z0	3																																									
0_1000b	011	8																																									
0_1001b	Z00	9																																									
0_1010b	Z0Z	10																																									
0_1011b	Z01	11																																									
1_0000b	Z1Z	16																																									
1_0001b	Z11	17																																									
1_0010b	100	18																																									
1_0011b	10Z	19																																									
5	<p><b>Active Management Port Enable</b></p> <p>Enables the Active Management Port. The value of this bit is latched in, upon reset de-assertion, from the STRAP_MGMT_PORT_EN input state.</p> <p>0 = Management Port is disabled (STRAP_MGMT_PORT_EN=1) 1 = Management Port is enabled (STRAP_MGMT_PORT_EN=0 or Z)</p> <p>If STRAP_MGMT_PORT_EN=0, the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, offset 3ACh[0]) is Set, and all Ports linkup (unless the serial EEPROM Clears the <i>Initiate Configuration</i> bit).</p> <p>If STRAP_MGMT_PORT_EN=Z, the <i>Initiate Configuration</i> bit value is 0, and only the Management Port links up. Other Ports linkup when the <i>Initiate Configuration</i> bit is Set by software, serial EEPROM, and/or I<sup>2</sup>C/SMBus.</p>	RWS	Yes	Set by STRAP_MGMT_PORT_EN input state																																							

**Register 13-119. 354h Management Port Control**  
**(Virtual Switch mode – Port 0, accessible through the Management Port and Redundant Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
6	<p><b>Active Management Port EEPROM Load on Hot Reset for Chip and Station Registers Enable</b>  <i>Valid only for the Management Port.</i></p> <p>After the Management Port receives a Hot Reset or DL_Down condition, the serial EEPROM reloads registers, as described below.</p> <p>0 = Serial EEPROM reloads Management Port Port-specific registers (default)  1 = Serial EEPROM reloads:</p> <ul style="list-style-type: none"> <li>• Chip-specific registers (might affect all virtual switches),</li> <li>• Station-specific registers (might affect all virtual switches), and,</li> <li>• Management Port Port-specific registers</li> </ul>	RWS	Yes	0
7	<b>Reserved</b>	RsvdP	No	0
12:8	<p><b>Redundant Management Port</b>  Indicates the Port Number of the Redundant Management Port.</p> <p>0_0000b = Port 0 (default)  0_0001b = Port 1  0_0010b = Port 2  0_0011b = Port 3  0_1000b = Port 8  0_1001b = Port 9  0_1010b = Port 10  0_1011b = Port 11  1_0000b = Port 16  1_0001b = Port 17  1_0010b = Port 18  1_0011b = Port 19</p> <p>All other encodings are <b>Reserved</b>.</p>	RWS	Yes	0_0000b
13	<p><b>Redundant Management Port Enable</b>  Enables the Redundant Management Port.</p>	RWS	Yes	0
15:14	<b>Reserved</b>	RsvdP	No	00b
18:16	<p><b>NT VS</b>  Indicates the virtual switch to which the NT Port belongs.</p> <p>000b = VS0 (default)  001b = VS1  010b = VS2  011b = VS3  100b = VS4  101b = VS5</p> <p>All other encodings are <b>Reserved</b>.</p>	RsvdP	No	000b
31:19	<b>Reserved</b>	RsvdP	No	0-0h

**Register 13-120. 358h Virtual Switch Enable**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
--------	-------------	------	------------------------------------	---------

This register indicates which virtual switches are enabled. The initial value of this register corresponds to the STRAP\_TESTMODE[2:0] and VS\_MODE input states.

STRAP_TESTMODE[2:0] Input States	VS_MODE Input State	# of Enabled Virtual Switches	VSx
0ZZ, 0Z1, 010, or 01Z	0	1	VS0
	Z	2	VS0, VS1
	1	3	VS0, VS1, VS2
011, Z00, Z0Z, or Z01	0	4	VS0, VS1, VS2, VS3
	Z	5	VS0, VS1, VS2, VS3, VS4
	1	6	VS0, VS1, VS2, VS3, VS4, VS5

All other encoding combinations are *Reserved*.

Table 13-6 lists the default Upstream and Downstream Port assignments of each virtual switch.

This register can be programmed to enable or disable virtual switches, prior to linkup, using any of the following methods:

- Serial EEPROM
- I<sup>2</sup>C, if STRAP\_I2C\_SMBUS\_CFG\_EN#=L, followed by an I<sup>2</sup>C Write that Sets the **Configuration Release** register *Initiate Configuration* bit (Port 0, accessible through the Management Port, offset 3ACh[0])
- Software, followed by a Hot Reset

0	<b>VS0 Enable</b> 0 = Disables VS0 1 = Enables VS0	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
1	<b>VS1 Enable</b> 0 = Disables VS1 1 = Enables VS1	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
2	<b>VS2 Enable</b> 0 = Disables VS2 1 = Enables VS2	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
3	<b>VS3 Enable</b> 0 = Disables VS3 1 = Enables VS3	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
4	<b>VS4 Enable</b> 0 = Disables VS4 1 = Enables VS4	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
5	<b>VS5 Enable</b> 0 = Disables VS5 1 = Enables VS5	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
31:6	<b>Reserved</b>	RsvdP	No	0-0h

**Register 13-121. 360h VS0 Upstream****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default																																							
<p>This register can be programmed, prior to linkup, using any of the following methods:</p> <ul style="list-style-type: none"><li>Serial EEPROM</li><li>I<sup>2</sup>C, if <b>STRAP_I2C_SMBUS_CFG_EN#</b>=L, followed by an I<sup>2</sup>C Write that Sets the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 3ACh[0])</li><li>Software, followed by a Hot Reset</li></ul> <p>In Base mode, this register defines the VS0 Upstream Port and NT Port, and enables NT mode.</p> <p>In Virtual Switch mode, this register defines the VS0 Upstream Port and NT Port, and enables NT mode. The <b>VSx Upstream</b> register(s) (offset(s) 360h through 374h), in combination with the <b>Virtual Switch Enable</b> and <b>VSx Port Vector</b> registers (offsets 354h, and 380h through 394h, respectively), comprise the Virtual Switch Table. For further details, refer to <a href="#">Section 5.3.3, “Virtual Switch Table.”</a></p> <p><b>Note:</b> <a href="#">Table 13-5</a> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>																																											
4:0	<p><b>VS0 Upstream Port</b></p> <p><b>Base Mode</b></p> <p>Identifies the VS0 Upstream Port, based upon the STRAP_UPSTRM_PORTSEL[2:0] input states.</p> <table><thead><tr><th>Field Value</th><th>STRAP_UPSTRM_PORTSEL[2:0] Input States</th><th>Port Number</th></tr></thead><tbody><tr><td>0_0000b</td><td>000</td><td>0</td></tr><tr><td>0_0001b</td><td>00Z</td><td>1</td></tr><tr><td>0_0010b</td><td>001</td><td>2</td></tr><tr><td>0_0011b</td><td>0Z0</td><td>3</td></tr><tr><td>0_1000b</td><td>011</td><td>8</td></tr><tr><td>0_1001b</td><td>Z00</td><td>9</td></tr><tr><td>0_1010b</td><td>Z0Z</td><td>10</td></tr><tr><td>0_1011b</td><td>Z01</td><td>11</td></tr><tr><td>1_0000b</td><td>Z1Z</td><td>16</td></tr><tr><td>1_0001b</td><td>Z11</td><td>17</td></tr><tr><td>1_0010b</td><td>100</td><td>18</td></tr><tr><td>1_0011b</td><td>10Z</td><td>19</td></tr></tbody></table> <p>All other encodings are <b>Reserved</b>.</p>	Field Value	STRAP_UPSTRM_PORTSEL[2:0] Input States	Port Number	0_0000b	000	0	0_0001b	00Z	1	0_0010b	001	2	0_0011b	0Z0	3	0_1000b	011	8	0_1001b	Z00	9	0_1010b	Z0Z	10	0_1011b	Z01	11	1_0000b	Z1Z	16	1_0001b	Z11	17	1_0010b	100	18	1_0011b	10Z	19	RWS	Yes	Defined by STRAP_UPSTRM_PORTSEL[2:0] input states
Field Value	STRAP_UPSTRM_PORTSEL[2:0] Input States	Port Number																																									
0_0000b	000	0																																									
0_0001b	00Z	1																																									
0_0010b	001	2																																									
0_0011b	0Z0	3																																									
0_1000b	011	8																																									
0_1001b	Z00	9																																									
0_1010b	Z0Z	10																																									
0_1011b	Z01	11																																									
1_0000b	Z1Z	16																																									
1_0001b	Z11	17																																									
1_0010b	100	18																																									
1_0011b	10Z	19																																									
	<p><b>Virtual Switch Mode</b></p> <p><a href="#">Table 13-6</a> lists the default Upstream and Downstream Port assignments of each virtual switch.</p>	RWS	Yes	0h																																							
7:5	<b>Reserved</b>	RsvdP	No	000b																																							

**Register 13-121. 360h VS0 Upstream****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default																																																								
12:8	<b>NT Port</b> The NT Port Number value is selected by the STRAP_NT_UPSTRM_PORTSEL[2:0] input states, and dependent upon whether the NT Port is enabled (bit 13, <i>NT Enable</i> ).	RWS	Yes	Defined by STRAP_NT_UPSTRM_P ORTSEL[2:0] input states																																																								
	<table><tr><th>Field Value</th><th>Bit 13 Value</th><th>STRAP_NT_UPSTRM_PORTSEL[2:0] Input States</th><th>Port Number</th></tr><tr><td>0_0000b</td><td>1</td><td>000</td><td>0</td></tr><tr><td>0_0001b</td><td>1</td><td>00Z</td><td>1</td></tr><tr><td>0_0010b</td><td>1</td><td>001</td><td>2</td></tr><tr><td>0_0011b</td><td>1</td><td>0Z0</td><td>3</td></tr><tr><td>0_1000b</td><td>1</td><td>011</td><td>8</td></tr><tr><td>0_1001b</td><td>1</td><td>Z00</td><td>9</td></tr><tr><td>0_1010b</td><td>1</td><td>Z0Z</td><td>10</td></tr><tr><td>0_1011b</td><td>1</td><td>Z01</td><td>11</td></tr><tr><td>1_0000b</td><td>1</td><td>Z1Z</td><td>16</td></tr><tr><td>1_0001b</td><td>1</td><td>Z11</td><td>17</td></tr><tr><td>1_0010b</td><td>1</td><td>100</td><td>18</td></tr><tr><td>1_0011b</td><td>1</td><td>10Z</td><td>19</td></tr><tr><td>1_1111b</td><td>0</td><td>111</td><td>— (NT mode is disabled)</td></tr></table>				Field Value	Bit 13 Value	STRAP_NT_UPSTRM_PORTSEL[2:0] Input States	Port Number	0_0000b	1	000	0	0_0001b	1	00Z	1	0_0010b	1	001	2	0_0011b	1	0Z0	3	0_1000b	1	011	8	0_1001b	1	Z00	9	0_1010b	1	Z0Z	10	0_1011b	1	Z01	11	1_0000b	1	Z1Z	16	1_0001b	1	Z11	17	1_0010b	1	100	18	1_0011b	1	10Z	19	1_1111b	0	111	— (NT mode is disabled)
	Field Value				Bit 13 Value	STRAP_NT_UPSTRM_PORTSEL[2:0] Input States	Port Number																																																					
	0_0000b				1	000	0																																																					
	0_0001b				1	00Z	1																																																					
	0_0010b				1	001	2																																																					
	0_0011b				1	0Z0	3																																																					
	0_1000b				1	011	8																																																					
	0_1001b				1	Z00	9																																																					
	0_1010b				1	Z0Z	10																																																					
	0_1011b				1	Z01	11																																																					
	1_0000b				1	Z1Z	16																																																					
	1_0001b				1	Z11	17																																																					
	1_0010b				1	100	18																																																					
	1_0011b				1	10Z	19																																																					
	1_1111b				0	111	— (NT mode is disabled)																																																					
All other encodings are <i>Reserved</i> .																																																												
13	<b>NT Enable</b> 0 = STRAP_NT_UPSTRM_PORTSEL[2:0] = 111, NT mode is disabled 1 = STRAP_NT_UPSTRM_PORTSEL[2:0] = Port Number, NT mode is enabled	RWS	Yes	Defined by STRAP_NT_UPSTRM_P ORTSEL[2:0] input states																																																								
25:14	<b>Factory Test Only</b> <i>Note:</i> When NT mode is disabled ( <i>STRAP_NT_UPSTRM_PORTSEL[2:0]=111</i> ), the value of bits [23:8] is 1A1Ah.	RWS	Yes	Refer to Note																																																								
29:26	<b>Reserved</b>	RsvdP	No	0h																																																								
31:30	<b>Factory Test Only</b>	RWS	Yes	00b																																																								

**Register 13-122. 364h VS1 Upstream**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register can be programmed, prior to linkup, using any of the following methods:</p> <ul style="list-style-type: none"> <li>Serial EEPROM</li> <li>I<sup>2</sup>C, if <b>STRAP_I2C_SMBUS_CFG_EN#</b>=L, followed by an I<sup>2</sup>C Write that Sets the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, accessible through the Management Port, offset 3ACh[0])</li> <li>Software, followed by a Hot Reset</li> </ul> <p>The <b>VSx Upstream</b> register(s) (offset(s) 360h through 374h), in combination with the <b>Virtual Switch Enable</b> and <b>VSx Port Vector</b> registers (offsets 354h, and 380h through 394h, respectively), comprise the Virtual Switch Table. For further details, refer to <a href="#">Section 5.3.3, “Virtual Switch Table.”</a></p>				
4:0	<b>VS1 Upstream Port</b> <b>Table 13-6</b> lists the default Upstream and Downstream Port assignments of each virtual switch.	RWS	Yes	Defined by STRAP_NT_UPSTRM_PORTSEL[2:0] input states
31:5	<b>Reserved</b>	RsvdP	No	0-0h

**Register 13-123. 368h VS2 Upstream**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register can be programmed, prior to linkup, using any of the following methods:</p> <ul style="list-style-type: none"> <li>Serial EEPROM</li> <li>I<sup>2</sup>C, if <b>STRAP_I2C_SMBUS_CFG_EN#</b>=L, followed by an I<sup>2</sup>C Write that Sets the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, accessible through the Management Port, offset 3ACh[0])</li> <li>Software, followed by a Hot Reset</li> </ul> <p>The <b>VSx Upstream</b> register(s) (offset(s) 360h through 374h), in combination with the <b>Virtual Switch Enable</b> and <b>VSx Port Vector</b> registers (offsets 354h, and 380h through 394h, respectively), comprise the Virtual Switch Table. For further details, refer to <a href="#">Section 5.3.3, “Virtual Switch Table.”</a></p>				
4:0	<b>VS2 Upstream Port</b> <b>Table 13-6</b> lists the default Upstream and Downstream Port assignments of each virtual switch.	RWS	Yes	Defined by STRAP_NT_UPSTRM_PORTSEL[2:0] input states
31:5	<b>Reserved</b>	RsvdP	No	0-0h

**Register 13-124. 36Ch VS3 Upstream**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register can be programmed, prior to linkup, using any of the following methods:</p> <ul style="list-style-type: none"> <li>Serial EEPROM</li> <li>I<sup>2</sup>C, if <b>STRAP_I2C_SMBUS_CFG_EN#</b>=L, followed by an I<sup>2</sup>C Write that Sets the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, accessible through the Management Port, offset <b>3ACh</b>[0])</li> <li>Software, followed by a Hot Reset</li> </ul> <p>The <b>VSx Upstream</b> register(s) (offset(s) <b>360h</b> through <b>374h</b>), in combination with the <b>Virtual Switch Enable</b> and <b>VSx Port Vector</b> registers (offsets <b>354h</b>, and <b>380h</b> through <b>394h</b>, respectively), comprise the Virtual Switch Table. For further details, refer to <a href="#">Section 5.3.3, “Virtual Switch Table.”</a></p>				
4:0	<b>VS3 Upstream Port</b> <b>Table 13-6</b> lists the default Upstream and Downstream Port assignments of each virtual switch.	RWS	Yes	Defined by STRAP_NT_UPSTRM_PORTSEL[2:0] input states
31:5	<b>Reserved</b>	RsvdP	No	0-0h



**Register 13-125. 370h VS4 Upstream**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register can be programmed, prior to linkup, using any of the following methods:</p> <ul style="list-style-type: none"> <li>Serial EEPROM</li> <li>I<sup>2</sup>C, if <b>STRAP_I2C_SMBUS_CFG_EN#</b>=L, followed by an I<sup>2</sup>C Write that Sets the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, accessible through the Management Port, offset 3ACh[0])</li> <li>Software, followed by a Hot Reset</li> </ul> <p>The <b>VSx Upstream</b> register(s) (offset(s) 360h through 374h), in combination with the <b>Virtual Switch Enable</b> and <b>VSx Port Vector</b> registers (offsets 354h, and 380h through 394h, respectively), comprise the Virtual Switch Table. For further details, refer to <a href="#">Section 5.3.3, “Virtual Switch Table.”</a></p>				
4:0	<b>VS4 Upstream Port</b> Table 13-6 lists the default Upstream and Downstream Port assignments of each virtual switch.	RWS	Yes	Defined by STRAP_NT_UPSTRM_PORTSEL[2:0] input states
31:5	<i>Reserved</i>	RsvdP	No	0-0h

**Register 13-126. 374h VS5 Upstream**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register can be programmed, prior to linkup, using any of the following methods:</p> <ul style="list-style-type: none"> <li>Serial EEPROM</li> <li>I<sup>2</sup>C, if <b>STRAP_I2C_SMBUS_CFG_EN#</b>=L, followed by an I<sup>2</sup>C Write that Sets the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, accessible through the Management Port, offset 3ACh[0])</li> <li>Software, followed by a Hot Reset</li> </ul> <p>The <b>VSx Upstream</b> register(s) (offset(s) 360h through 374h), in combination with the <b>Virtual Switch Enable</b> and <b>VSx Port Vector</b> registers (offsets 354h, and 380h through 394h, respectively), comprise the Virtual Switch Table. For further details, refer to <a href="#">Section 5.3.3, “Virtual Switch Table.”</a></p>				
4:0	<b>VS5 Upstream Port</b> Table 13-6 lists the default Upstream and Downstream Port assignments of each virtual switch.	RWS	Yes	Defined by STRAP_NT_UPSTRM_PORTSEL[2:0] input states
31:5	<i>Reserved</i>	RsvdP	No	0-0h

**Register 13-127. 380h VS0 Port Vector****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
--------	-------------	--------	------	------------------------------------	---------

This register can be programmed, prior to linkup, using any of the following methods:

- Serial EEPROM
- I<sup>2</sup>C, if **STRAP\_I2C\_SMBUS\_CFG\_EN#**=L, followed by an I<sup>2</sup>C Write that Sets the **Configuration Release** register *Initiate Configuration* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset **3ACh**[0])
- Software, followed by a Hot Reset

In Base mode, the default value of this register is 00FF\_FFFFh, indicating that all Ports are assigned to VS0.

In Virtual Switch mode:

- This register defines which Upstream and Downstream Ports are active for VS0, and is initially programmed (along with other Virtual Switch Table registers; refer to the next bullet) according to the values of the **STRAP\_TESTMODE**[2:0] and **VS\_MODE** 3-state inputs, and/or by serial EEPROM. Immediately following **PEX\_PERST#** input de-assertion, if the **Configuration Release** register *Initiate Configuration* bit (Port 0, accessible through the Management Port, offset **3ACh**[0]) is Cleared (by **STRAP\_I2C\_SMBUS\_CFG\_EN#**=L, **STRAP\_MGMT\_PORT\_EN**=Z, and/or serial EEPROM), to allow I<sup>2</sup>C and/or Management Port software to program switch configuration prior to Setting the *Initiate Configuration* bit, the default value of this register is 00FF\_FFFFh, indicating that all Ports are initially assigned to VS0.
- VSx Port Vector** register(s) (offset(s) **380h** through **394h**), in combination with the **Virtual Switch Enable** and **VSx Upstream** registers (offsets **354h**, and **360h** through **374h**, respectively), comprise the Virtual Switch Table. For further details, refer to **Section 5.3.3, “Virtual Switch Table.”**
- Attempted access through virtual switch Upstream Ports, that are not the Management Port, is handled as NOP (Reads return a value of 0h, Writes have no effect).

**Table 13-6** lists the default Upstream and Downstream Port assignments of each virtual switch.

**Notes:** The **Port x** column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name.

**Table 13-5** indicates which Ports are enabled/used, and when, based upon the **STRAP\_STNx\_PORTCFGx** input states and/or serial EEPROM programming.

0	<b>VS0 Active Port x</b>	0	RWS	Yes	Based upon <b>STRAP_TESTMODE</b> [2:0] and <b>VS_MODE</b> input states
1		1	RWS	Yes	
2		2	RWS	Yes	
3		3	RWS	Yes	
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>VS0 Active Port x</b>	8	RWS	Yes	Based upon <b>STRAP_TESTMODE</b> [2:0] and <b>VS_MODE</b> input states
9		9	RWS	Yes	
10		10	RWS	Yes	
11		11	RWS	Yes	
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b

**Register 13-127. 380h VS0 Port Vector**

(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>VS0 Active Ports</b>	16	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
17		17	RWS	Yes	
18		18	RWS	Yes	
19		19	RWS	Yes	
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
31:22	<b>Reserved</b>		RsvdP	No	0-0h

**Register 13-128. 384h VS1 Port Vector**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register defines which Upstream and Downstream Ports are associated with VS1. The initial value of this register corresponds to the STRAP_TESTMODE[2:0] and VS_MODE input states.</p> <p>Attempted access through virtual switch Upstream Ports, that are not the Management Port, is handled as NOP (Reads return a value of 0h, Writes have no effect).</p> <p>Table 13-6 lists the default Upstream and Downstream Port assignments of each virtual switch.</p> <p>This register can be programmed, prior to linkup, using any of the following methods:</p> <ul style="list-style-type: none"> <li>Serial EEPROM</li> <li>I<sup>2</sup>C, if STRAP_I2C_SMBUS_CFG_EN#=L, followed by an I<sup>2</sup>C Write that Sets the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, accessible through the Management Port, offset 3ACh[0])</li> <li>Software, followed by a Hot Reset</li> </ul> <p>The <b>VSx Port Vector</b> register(s) (offset(s) 380h through 394h), in combination with the <b>Virtual Switch Enable</b> and <b>VSx Upstream</b> registers (offsets 354h, and 360h through 374h, respectively), comprise the Virtual Switch Table. For further details, refer to Section 5.3.3, “Virtual Switch Table.”</p> <p><b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name. Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>					
0	<b>VS1 Active Port x</b>	0	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
1		1	RWS	Yes	
2		2	RWS	Yes	
3		3	RWS	Yes	
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>VS1 Active Port x</b>	8	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
9		9	RWS	Yes	
10		10	RWS	Yes	
11		11	RWS	Yes	
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b
16	<b>VS1 Active Port x</b>	16	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
17		17	RWS	Yes	
18		18	RWS	Yes	
19		19	RWS	Yes	
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
31:22	<b>Reserved</b>		RsvdP	No	0-0h

**Register 13-129. 388h VS2 Port Vector**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register defines which Upstream and Downstream Ports are associated with VS2. The initial value of this register corresponds to the STRAP_TESTMODE[2:0] and VS_MODE input states.</p> <p>Attempted access through virtual switch Upstream Ports, that are not the Management Port, is handled as NOP (Reads return a value of 0h, Writes have no effect).</p> <p>Table 13-6 lists the default Upstream and Downstream Port assignments of each virtual switch.</p> <p>This register can be programmed, prior to linkup, using any of the following methods:</p> <ul style="list-style-type: none"> <li>Serial EEPROM</li> <li>I<sup>2</sup>C, if STRAP_I2C_SMBUS_CFG_EN# = L, followed by an I<sup>2</sup>C Write that Sets the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, accessible through the Management Port, offset 3ACh[0])</li> <li>Software, followed by a Hot Reset</li> </ul> <p>The <b>VSx Port Vector</b> register(s) (offset(s) 380h through 394h), in combination with the <b>Virtual Switch Enable</b> and <b>VSx Upstream</b> registers (offsets 354h, and 360h through 374h, respectively), comprise the Virtual Switch Table. For further details, refer to Section 5.3.3, “Virtual Switch Table.”</p> <p><b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name. Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>					
0	<b>VS2 Active Port x</b>	0	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
1		1	RWS	Yes	
2		2	RWS	Yes	
3		3	RWS	Yes	
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>VS2 Active Port x</b>	8	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
9		9	RWS	Yes	
10		10	RWS	Yes	
11		11	RWS	Yes	
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b
16	<b>VS2 Active Port x</b>	16	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
17		17	RWS	Yes	
18		18	RWS	Yes	
19		19	RWS	Yes	
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
31:22	<b>Reserved</b>		RsvdP	No	0-0h

**Register 13-130. 38Ch VS3 Port Vector**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register defines which Upstream and Downstream Ports are associated with VS3. The initial value of this register corresponds to the STRAP_TESTMODE[2:0] and VS_MODE input states.</p> <p>Attempted access through virtual switch Upstream Ports, that are not the Management Port, is handled as NOP (Reads return a value of 0h, Writes have no effect).</p> <p>Table 13-6 lists the default Upstream and Downstream Port assignments of each virtual switch.</p> <p>This register can be programmed, prior to linkup, using any of the following methods:</p> <ul style="list-style-type: none"> <li>Serial EEPROM</li> <li>I<sup>2</sup>C, if STRAP_I2C_SMBUS_CFG_EN#=L, followed by an I<sup>2</sup>C Write that Sets the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, accessible through the Management Port, offset 3ACh[0])</li> <li>Software, followed by a Hot Reset</li> </ul> <p>The <b>VSx Port Vector</b> register(s) (offset(s) 380h through 394h), in combination with the <b>Virtual Switch Enable</b> and <b>VSx Upstream</b> registers (offsets 354h, and 360h through 374h, respectively), comprise the Virtual Switch Table. For further details, refer to Section 5.3.3, “Virtual Switch Table.”</p> <p><b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name. Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>					
0	<b>VS3 Active Port x</b>	0	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
1		1	RWS	Yes	
2		2	RWS	Yes	
3		3	RWS	Yes	
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>VS3 Active Port x</b>	8	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
9		9	RWS	Yes	
10		10	RWS	Yes	
11		11	RWS	Yes	
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b
16	<b>VS3 Active Port x</b>	16	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
17		17	RWS	Yes	
18		18	RWS	Yes	
19		19	RWS	Yes	
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
31:22	<b>Reserved</b>		RsvdP	No	0-0h

**Register 13-131. 390h VS4 Port Vector**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register defines which Upstream and Downstream Ports are associated with VS4. The initial value of this register corresponds to the STRAP_TESTMODE[2:0] and VS_MODE input states.</p> <p>Attempted access through virtual switch Upstream Ports, that are not the Management Port, is handled as NOP (Reads return a value of 0h, Writes have no effect).</p> <p>Table 13-6 lists the default Upstream and Downstream Port assignments of each virtual switch.</p> <p>This register can be programmed, prior to linkup, using any of the following methods:</p> <ul style="list-style-type: none"> <li>Serial EEPROM</li> <li>I<sup>2</sup>C, if STRAP_I2C_SMBUS_CFG_EN# = L, followed by an I<sup>2</sup>C Write that Sets the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, accessible through the Management Port, offset 3ACh[0])</li> <li>Software, followed by a Hot Reset</li> </ul> <p>The <b>VSx Port Vector</b> register(s) (offset(s) 380h through 394h), in combination with the <b>Virtual Switch Enable</b> and <b>VSx Upstream</b> registers (offsets 354h, and 360h through 374h, respectively), comprise the Virtual Switch Table. For further details, refer to Section 5.3.3, “Virtual Switch Table.”</p> <p><b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name. Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>					
0	<b>VS4 Active Port x</b>	0	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
1		1	RWS	Yes	
2		2	RWS	Yes	
3		3	RWS	Yes	
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>VS4 Active Port x</b>	8	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
9		9	RWS	Yes	
10		10	RWS	Yes	
11		11	RWS	Yes	
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b
16	<b>VS4 Active Port x</b>	16	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
17		17	RWS	Yes	
18		18	RWS	Yes	
19		19	RWS	Yes	
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
31:22	<b>Reserved</b>		RsvdP	No	0-0h

**Register 13-132. 394h VS5 Port Vector**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register defines which Upstream and Downstream Ports are associated with VS5. The initial value of this register corresponds to the STRAP_TESTMODE[2:0] and VS_MODE input states.</p> <p>Attempted access through virtual switch Upstream Ports, that are not the Management Port, is handled as NOP (Reads return a value of 0h, Writes have no effect).</p> <p>Table 13-6 lists the default Upstream and Downstream Port assignments of each virtual switch.</p> <p>This register can be programmed, prior to linkup, using any of the following methods:</p> <ul style="list-style-type: none"> <li>Serial EEPROM</li> <li>I<sup>2</sup>C, if STRAP_I2C_SMBUS_CFG_EN#=L, followed by an I<sup>2</sup>C Write that Sets the <b>Configuration Release</b> register <i>Initiate Configuration</i> bit (Port 0, accessible through the Management Port, offset 3ACh[0])</li> <li>Software, followed by a Hot Reset</li> </ul> <p>The <b>VSx Port Vector</b> register(s) (offset(s) 380h through 394h), in combination with the <b>Virtual Switch Enable</b> and <b>VSx Upstream</b> registers (offsets 354h, and 360h through 374h, respectively), comprise the Virtual Switch Table. For further details, refer to Section 5.3.3, “Virtual Switch Table.”</p> <p><b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name. Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>					
0	<b>VS5 Active Port x</b>	0	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
1		1	RWS	Yes	
2		2	RWS	Yes	
3		3	RWS	Yes	
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>VS5 Active Port x</b>	8	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
9		9	RWS	Yes	
10		10	RWS	Yes	
11		11	RWS	Yes	
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b
16	<b>VS5 Active Port x</b>	16	RWS	Yes	Based upon STRAP_TESTMODE[2:0] and VS_MODE input states
17		17	RWS	Yes	
18		18	RWS	Yes	
19		19	RWS	Yes	
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
31:22	<b>Reserved</b>		RsvdP	No	0-0h



**Register 13-133. 3A0h Port Reset****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Downstream Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
When driven with a value of 1, this register holds the Port in reset, including the Port PCI-to-PCI bridge and the hierarchy below it. A value of 0 indicates to un-reset the PCI-to-PCI bridge and the hierarchy below it.					
<i>Notes: The <b>Downstream Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name.</i>					
<i>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</i>					
0	<b>Reset Port x Vector</b> 0 = Port is not reset 1 = Port is reset	0	RWS	Yes	0
1		1	RWS	Yes	0
2		2	RWS	Yes	0
3		3	RWS	Yes	0
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>Reset Port x Vector</b> 0 = Port is not reset 1 = Port is reset	8	RWS	Yes	0
9		9	RWS	Yes	0
10		10	RWS	Yes	0
11		11	RWS	Yes	0
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b
16	<b>Reset Port x Vector</b> 0 = Port is not reset 1 = Port is reset	16	RWS	Yes	0
17		17	RWS	Yes	0
18		18	RWS	Yes	0
19		19	RWS	Yes	0
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
31:22	<b>Reserved</b>		RsvdP	No	0-0h

**Register 13-134. 3A4h Parallel Hot Plug Control****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
Any Downstream Transparent Port can be configured as a Parallel Hot Plug Port or Serial Hot Plug Port. Hot Plug Port assignment is described in <a href="#">Section 10.9, “Default Parallel Hot Plug Ports.”</a>				
0_0000b = Port 0				
0_0001b = Port 1				
0_0010b = Port 2				
0_0011b = Port 3				
0_1000b = Port 8				
0_1001b = Port 9				
0_1010b = Port 10				
0_1011b = Port 11				
1_0000b = Port 16				
1_0001b = Port 17				
1_0010b = Port 18				
1_0011b = Port 19				
All other encodings are <i>Reserved</i> .				
<b>Note:</b> <a href="#">Table 13-5</a> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.				
4:0	<b>Parallel Hot Plug Port A</b> Value specifies which Port is assigned Parallel Hot Plug Controller A.	RWS	Yes	<i>TBD</i>
6:5	<i>Reserved</i>	RsvdP	No	00b
7	<b>Parallel Hot Plug Port A Enable</b> 0 = Hot Plug Port A is not enabled 1 = Hot Plug Port A is enabled	RWS	Yes	1
12:8	<b>Parallel Hot Plug Port B</b> Value specifies which Port is assigned Parallel Hot Plug Controller B.	RWS	Yes	<i>TBD</i>
14:13	<i>Reserved</i>	RsvdP	No	00b
15	<b>Parallel Hot Plug Port B Enable</b> 0 = Hot Plug Port B is not enabled 1 = Hot Plug Port B is enabled	RWS	Yes	1
20:16	<b>Parallel Hot Plug Port C</b> Value specifies which Port is assigned Parallel Hot Plug Controller C.	RWS	Yes	<i>TBD</i>
22:21	<i>Reserved</i>	RsvdP	No	00b
23	<b>Parallel Hot Plug Port C Enable</b> 0 = Hot Plug Port C is not enabled 1 = Hot Plug Port C is enabled	RWS	Yes	1
31:24	<i>Reserved</i>	RsvdP	No	00h

**Register 13-135. 3A8h VSx\_PERST# and NT\_PERST# Status**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The VSx_PERST#-related bits are valid only in Virtual Switch mode. The PEX_NT_PERST#-related bits are valid in both Base mode and Virtual Switch mode.				
0	<b>VS0_PERST# Input Value</b> 0 = VS0_PERST# is Low 1 = VS0_PERST# is High	ROS	No	Reflects the VS0_PERST# input value
1	<b>VS1_PERST# Input Value</b> 0 = VS1_PERST# is Low 1 = VS1_PERST# is High	ROS	No	Reflects the VS1_PERST# input value
2	<b>VS2_PERST# Input Value</b> 0 = VS2_PERST# is Low 1 = VS2_PERST# is High	ROS	No	Reflects the VS2_PERST# input value
3	<b>VS3_PERST# Input Value</b> 0 = VS3_PERST# is Low 1 = VS3_PERST# is High	ROS	No	Reflects the VS3_PERST# input value
4	<b>VS4_PERST# Input Value</b> 0 = VS4_PERST# is Low 1 = VS4_PERST# is High	ROS	No	Reflects the VS4_PERST# input value
5	<b>VS5_PERST# Input Value</b> 0 = VS5_PERST# is Low 1 = VS5_PERST# is High	ROS	No	Reflects the VS5_PERST# input value
7:6	<b>Factory Test Only/Reserved</b>	RsvdP	No	00b
8	<b>VS0_PERST# Control</b> 0 = Writing 0 Clears the VS0_PERST# reset condition 1 = Causes VS0_PERST# to receive a <i>Fundamental</i> Reset	RWS	Yes	0
9	<b>VS1_PERST# Control</b> 0 = Writing 0 Clears the VS1_PERST# reset condition 1 = Causes VS1_PERST# Reset	RWS	Yes	0
10	<b>VS2_PERST# Control</b> 0 = Writing 0 Clears the VS2_PERST# reset condition 1 = Causes VS2_PERST# Reset	RWS	Yes	0
11	<b>VS3_PERST# Control</b> 0 = Writing 0 Clears the VS3_PERST# reset condition 1 = Causes VS3_PERST# Reset	RWS	Yes	0
12	<b>VS4_PERST# Control</b> 0 = Writing 0 Clears the VS4_PERST# reset condition 1 = Causes VS4_PERST# Reset	RWS	Yes	0
13	<b>VS5_PERST# Control</b> 0 = Writing 0 Clears the VS5_PERST# reset condition 1 = Causes VS5_PERST# Reset	RWS	Yes	0
15:14	<b>Factory Test Only/Reserved</b>	RsvdP	No	00b

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 13-135. 3A8h VSx\_PERST# and NT\_PERST# Status**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the**  
**Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>PEX_NT_PERST# Ball Value</b> 0 = PEX_NT_PERST# input is Low 1 = PEX_NT_PERST# input is High	ROS	No	Reflects the <a href="#">PEX_NT_PERST#</a> input value
23:17	<b>Reserved</b>	RsvdP	No	0-0h
24	<b>PEX_NT_PERST# Control</b> 0 = Writing 0 Clears the PEX_NT_PERST# reset condition 1 = Causes PEX_NT_PERST# Reset	RWS	Yes	0
31:25	<b>Reserved</b>	RsvdP	No	0-0h

**Register 13-136. 3ACh Configuration Release****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<p><b>Initiate Configuration</b>  0 = STRAP_I2C_SMBUS_CFG_EN#=0, –or– STRAP_MGMT_PORT_EN=Z  1 = STRAP_I2C_SMBUS_CFG_EN#=1, –or– STRAP_MGMT_PORT_EN=0 or 1, releases the hold that prevents the Links from coming up</p> <p><b>Base Mode and Virtual Switch Mode</b>  If the STRAP_I2C_SMBUS_CFG_EN# input is pulled or tied Low to VSS (Ground), I<sup>2</sup>C is being used to configure the PEX 8748. After I<sup>2</sup>C has completed its configuration, it must write to this bit, to release the hold that is preventing the Links from coming up.</p> <p><b>Virtual Switch Mode</b>  The STRAP_MGMT_PORT_EN 3-state input is used to control Bring-Up Options 1 and 2:</p> <ul style="list-style-type: none"> <li>• <b>Option 1</b> – STRAP_MGMT_PORT_EN=Z. After the serial EEPROM (if present) is loaded, the Management Port comes up first, to configure the PEX 8748. When the Management Port has completed its configuration, Management Port software and/or I<sup>2</sup>C/SMBus must Set this bit, to release the hold that is preventing the remaining Links from coming up (assuming this hold is not de-asserted by the serial EEPROM Setting this bit).</li> <li>• <b>Option 2</b> – STRAP_MGMT_PORT_EN=0. After the serial EEPROM (if present) is loaded, all Ports come up concurrently (provided that the serial EEPROM does <i>not</i> Clear this bit).</li> </ul>	HwInit	Yes	<p>0  (STRAP_I2C_SMBUS_CFG_EN#=0, –or–  if in Virtual Switch mode, STRAP_MGMT_PORT_EN=Z)  Otherwise, 1</p>
31:1	<b>Reserved</b>	RsvdP	No	0-0h

**Register 13-137. 46Ch Strap Configuration****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4:0	<i>Factory Test Only</i>	HwInit	Yes	0-0h
5	<b>I<sup>2</sup>C Configuration Enable Status</b> 0 = I <sup>2</sup> C/SMBus Configuration is disabled (STRAP_I2C_SMBUS_CFG_EN#=H) 1 = I <sup>2</sup> C/SMBus Configuration is enabled (STRAP_I2C_SMBUS_CFG_EN#=L)	ROS	Yes	PCFG
6	<b>SMBus Enable Status</b> 0 = I <sup>2</sup> C protocol is enabled (STRAP_I2C_SMBUS_EN=0) 1 = SMBus protocol is enabled (STRAP_I2C_SMBUS_EN=Z or 1)	ROS	Yes	PCFG
7	<i>Reserved</i>	RsvdP	No	0
9:8	<i>Factory Test Only</i>	ROS	Yes	11b
10	<b>Gen 1 Compatible Disable Status</b> Value is dependent upon the SPARE2 input. 0 = Enable Gen 1 Compatible mode, to accommodate a connected, non-compliant Gen 1 device (SPARE2=L) 1 = Disable Gen 1 Compatible mode, for non-compliant Gen 1 device (SPARE2=H)	ROS	Yes	0 (SPARE2=L) 1 (SPARE2=H)
11	<b>Gen 1 Mode Status</b> Value is dependent upon the STRAP_GEN1_GEN2 3-state input. 0 = Link speed is 2.5 GT/s (STRAP_GEN1_GEN2=0) 1 = Maximum Link speed is 5.0 or 8.0 GT/s (STRAP_GEN1_GEN2=Z or 1, respectively)	ROS	Yes	PCFG

**Register 13-137. 46Ch Strap Configuration****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Gen 2 Mode Status</b> Value is dependent upon the STRAP_GEN1_GEN2 3-state input. 0 = Maximum Link speed is 5.0 GT/s (STRAP_GEN1_GEN2=Z) 1 = Maximum Link speed is 8.0 GT/s (STRAP_GEN1_GEN2=1)	ROS	Yes	PCFG
15:13	<b>VS Mode Status</b> Indicates the HP_PWR_GOOD_A_VS_MODE input state, primarily for the purpose of allowing software to check the board connection. The VS_MODE 3-state input is sampled and latched at PEX_PERST# and/or VSx_PERST# input de-assertion). While the serial EEPROM and/or I <sup>2</sup> C/SMBus could change the value, Virtual Switch mode must be enabled by programming the Virtual Switch Table registers, rather than this field. (Virtual Switch Table registers are comprised of the <b>Virtual Switch Enable</b> , <b>VSx Upstream</b> , and <b>VSx Port Vector</b> registers (offsets 354h, 360h through 374h, and 380h through 394h, respectively.) 000b = Base mode, no virtual switches are enabled 001b = Two virtual switches are enabled (VS0, VS1) 010b = Three virtual switches are enabled (VS0, VS1, VS2) 011b = Four virtual switches are enabled (VS0, VS1, VS2, and VS3) 100b = Five virtual switches are enabled (VS0, VS1, VS2, VS3, and VS4) 101b = Six virtual switches are enabled (VS0, VS1, VS2, VS3, VS4, and VS5) All other encodings are <b>Reserved</b> .	ROS	Yes	PCFG

**Register 13-137. 46Ch Strap Configuration**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20:16	<i>Factory Test Only</i>	ROS	No	PCFG
23:21	<i>Reserved</i>	RsvdP	No	000b
28:24	<b>Strap Test Mode Status</b> Reflects the STRAP_TESTMODE[2:0] input states.  00000b = 000 00001b = 00Z 00010b = 001 00011b = 0Z0  00100b = 0ZZ 00101b = 0Z1 00110b = 010 00111b = 01Z  01000b = 011 01001b = Z00 01010b = Z0Z 01011b = Z01  Encodings 01100b through 11010b are <i>Factory Test Only</i> . Encodings 11010b and above are <i>Reserved</i> .	ROS	No	Based upon STRAP_TESTMODE[2:0] input states
31:29	<i>Reserved</i>	RsvdP	No	000b



### 13.16.9 Device-Specific Registers – General-Purpose Input/Output (Offsets 600h – 68Ch)

This section details the Device-Specific General-Purpose Input/Output (GPIO) registers. [Table 13-26](#) defines the register map.

**Note:** In Virtual Switch mode, each virtual switch uses its own local set of **GPIO** registers (VS Upstream Port(s), offsets [A34h](#) through [A54h](#)).

**Table 13-26. Device-Specific GPIO Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		
GPIO 0_9 Direction Control		600h
GPIO 10_11 Direction Control		604h
Reserved		608h –
GPIO 0_11 Input De-Bounce		614h
Reserved		618h
GPIO 0_11 Input Data		61Ch
Reserved		620h
GPIO 0_11 Output Data		624h
Reserved		628h
GPIO 0_11 Interrupt Polarity		62Ch
Reserved		630h
GPIO 0_11 Interrupt Status		634h
Reserved		638h
GPIO 0_11 Interrupt Mask		63Ch
Reserved		640h –
Reserved (Base Mode) Virtual Switch GPIO Update (Virtual Switch Mode)		64Ch
Reserved (Base Mode) VS0 GPIO_PG 0_11 Assignment (Virtual Switch Mode)		650h
Reserved (Base Mode) VS1 GPIO_PG 0_11 Assignment (Virtual Switch Mode)		654h
Reserved (Base Mode) VS2 GPIO_PG 0_11 Assignment (Virtual Switch Mode)		658h
Reserved (Base Mode) VS3 GPIO_PG 0_11 Assignment (Virtual Switch Mode)		65Ch
Reserved (Base Mode) VS4 GPIO_PG 0_11 Assignment (Virtual Switch Mode)		660h
Reserved (Base Mode) VS5 GPIO_PG 0_11 Assignment (Virtual Switch Mode)		664h
Factory Test Only/Reserved		668h – 68Ch

**Register 13-138. 600h GPIO 0\_9 Direction Control****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>The <b>GPIO 0_9 Direction Control</b> registers control the direction, source, and destination of the PORT_GOOD[0:9]# signals, respectively.</p> <p><b>Notes:</b> Register offsets <b>61Ch</b> and <b>624h</b>, referenced within this register, are located as follows – Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port.</p> <p>PORT_GOOD[9:0]# correspond to Ports 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) <b>6A0h</b> through <b>6A8h</b>.</p> <p><i>Table 13-5</i> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
1:0	<b>PORT_GOOD0# Source/Destination</b> <b>As Input:</b> 00b = To <b>PORT_GOOD0# Input Data</b> register (offset <b>61Ch</b> [0]) 01b = General interrupt (INTx, MSI, or <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode)) 10b, 11b = <b>Reserved</b> <b>As Output:</b> 00b = From <b>PORT_GOOD0# Output Data</b> register (offset <b>624h</b> [0]) 01b = PORT_GOOD0# 10b = SHP_PERST0# output 11b = <b>Reserved</b>	RWS	Yes	<b>TBD</b>
2	<b>PORT_GOOD0# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	<b>TBD</b>
4:3	<b>PORT_GOOD1# Source/Destination</b> <b>As Input:</b> 00b = To <b>PORT_GOOD1# Input Data</b> register (offset <b>61Ch</b> [1]) 01b = General interrupt (INTx, MSI, or <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode)) 10b, 11b = <b>Reserved</b> <b>As Output:</b> 00b = From <b>PORT_GOOD1# Output Data</b> register (offset <b>624h</b> [1]) 01b = PORT_GOOD1# 10b = SHP_PERST1# output 11b = <b>Reserved</b>	RWS	Yes	<b>TBD</b>
5	<b>PORT_GOOD1# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	<b>TBD</b>

**Register 13-138. 600h GPIO 0\_9 Direction Control**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:6	<b>PORT_GOOD2# Source/Destination</b> <b>As Input:</b> 00b = To <b>PORT_GOOD2# Input Data</b> register (offset 61Ch[2]) 01b = General interrupt (INTx, MSI, or <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode)) 10b, 11b = <b>Reserved</b> <b>As Output:</b> 00b = From <b>PORT_GOOD2# Output Data</b> register (offset 624h[2]) 01b = <b>PORT_GOOD2#</b> 10b = <b>SHP_PERST2#</b> output 11b = <b>Reserved</b>	RWS	Yes	<b>TBD</b>
8	<b>PORT_GOOD2# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	<b>TBD</b>
10:9	<b>PORT_GOOD3# Source/Destination</b> <b>As Input:</b> 00b = To <b>PORT_GOOD3# Input Data</b> register (offset 61Ch[3]) 01b = General interrupt (INTx, MSI, or <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode)) 10b, 11b = <b>Reserved</b> <b>As Output:</b> 00b = From <b>PORT_GOOD3# Output Data</b> register (offset 624h[3]) 01b = <b>PORT_GOOD3#</b> 10b = <b>SHP_PERST3#</b> output 11b = <b>Reserved</b>	RWS	Yes	<b>TBD</b>
11	<b>PORT_GOOD3# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	<b>TBD</b>

**Register 13-138. 600h GPIO 0\_9 Direction Control**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
13:12	<b>PORT_GOOD4# Source/Destination</b> <b>As Input:</b> 00b = To <b>PORT_GOOD4# Input Data</b> register (offset 61Ch[4]) 01b = General interrupt (INTx, MSI, or <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode)) 10b, 11b = <i>Reserved</i> <b>As Output:</b> 00b = From <b>PORT_GOOD4# Output Data</b> register (offset 624h[4]) 01b = <b>PORT_GOOD4#</b> 10b = <b>SHP_PERST4#</b> output 11b = <i>Reserved</i>	RWS	Yes	<i>TBD</i>
14	<b>PORT_GOOD4# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	<i>TBD</i>
16:15	<b>PORT_GOOD5# Source/Destination</b> <b>As Input:</b> 00b = To <b>PORT_GOOD5# Input Data</b> register (offset 61Ch[5]) 01b = General interrupt (INTx, MSI, or <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode)) 10b, 11b = <i>Reserved</i> <b>As Output:</b> 00b = From <b>PORT_GOOD5# Output Data</b> register (offset 624h[5]) 01b = <b>PORT_GOOD5#</b> 10b = <b>SHP_PERST5#</b> output 11b = <i>Reserved</i>	RWS	Yes	<i>TBD</i>
17	<b>PORT_GOOD5# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	<i>TBD</i>

**Register 13-138. 600h GPIO 0\_9 Direction Control**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:18	<b>PORT_GOOD6# Source/Destination</b> <b>As Input:</b> 00b = To <b>PORT_GOOD6# Input Data</b> register (offset 61Ch[6]) 01b = General interrupt (INTx, MSI, or <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode)) 10b, 11b = <b>Reserved</b> <b>As Output:</b> 00b = From <b>PORT_GOOD6# Output Data</b> register (offset 624h[6]) 01b = <b>PORT_GOOD6#</b> 10b = <b>SHP_PERST6#</b> output 11b = <b>Reserved</b>	RWS	Yes	<b>TBD</b>
20	<b>PORT_GOOD6# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	<b>TBD</b>
22:21	<b>PORT_GOOD7# Source/Destination</b> <b>As Input:</b> 00b = To <b>PORT_GOOD7# Input Data</b> register (offset 61Ch[7]) 01b = General interrupt (INTx, MSI, or <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode)) 10b, 11b = <b>Reserved</b> <b>As Output:</b> 00b = From <b>PORT_GOOD7# Output Data</b> register (offset 624h[7]) 01b = <b>PORT_GOOD7#</b> 10b = <b>SHP_PERST7#</b> output 11b = <b>Reserved</b>	RWS	Yes	<b>TBD</b>
23	<b>PORT_GOOD7# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	<b>TBD</b>

**Register 13-138. 600h GPIO 0\_9 Direction Control**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
25:24	<b>PORT_GOOD8# Source/Destination</b> <b>As Input:</b> 00b = To <b>PORT_GOOD8# Input Data</b> register (offset 61Ch[8]) 01b = General interrupt (INTx, MSI, or <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode)) 10b, 11b = <b>Reserved</b> <b>As Output:</b> 00b = From <b>PORT_GOOD8# Output Data</b> register (offset 624h[8]) 01b = PORT_GOOD8# 10b = SHP_PERST8# output 11b = <b>Reserved</b>	RWS	Yes	<b>TBD</b>
26	<b>PORT_GOOD8# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	<b>TBD</b>
28:27	<b>PORT_GOOD9# Source/Destination</b> <b>As Input:</b> 00b = To <b>PORT_GOOD9# Input Data</b> register (offset 61Ch[9]) 01b = General interrupt (INTx, MSI, or <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode)) 10b, 11b = <b>Reserved</b> <b>As Output:</b> 00b = From <b>PORT_GOOD9# Output Data</b> register (offset 624h[9]) 01b = PORT_GOOD9# 10b = SHP_PERST9# output 11b = <b>Reserved</b>	RWS	Yes	<b>TBD</b>
29	<b>PORT_GOOD9# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	<b>TBD</b>
31:30	<b>Reserved</b>	RsvdP	No	00b

**Register 13-139. 604h GPIO 10\_11 Direction Control**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>The <b>GPIO 10_11 Direction Control</b> registers controls the direction, source, and destination of the PORT_GOOD[10:11]# signals, respectively.</p> <p><i>Notes:</i> Register offsets <b>61Ch</b> and <b>624h</b>, referenced within this register, are located as follows – Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port.</p> <p><i>PORT_GOOD[11:10]# correspond to Ports 19 and 18, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) <b>6A0h</b> through <b>6A8h</b>.</i></p> <p><i>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</i></p>				
1:0	<p><b>PORT_GOOD10# Source/Destination</b></p> <p><b>As Input:</b>            00b = To <b>PORT_GOOD10# Input Data</b> register (offset <b>61Ch[10]</b>)            01b = General interrupt (INTx, MSI, or <b>PEX_INTA#</b> (Base mode) or VSx_PEX_INTA# (Virtual Switch mode))            10b, 11b = <b>Reserved</b></p> <p><b>As Output:</b>            00b = From <b>PORT_GOOD10# Output Data</b> register (offset <b>624h[10]</b>)            01b = PORT_GOOD10#            10b = SHP_PERST10# output            11b = <b>Reserved</b></p>	RWS	Yes	<b>TBD</b>
2	<p><b>PORT_GOOD10# Direction Control</b></p> <p>0 = Input            1 = Output</p>	RWS	Yes	<b>TBD</b>

**Register 13-139. 604h GPIO 10\_11 Direction Control**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4:3	<b>PORT_GOOD11# Source/Destination</b> <i>Note: This field is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this field is <b>Reserved</b>.</i> 00b = To <b>PORT_GOOD11# Input Data</b> register (offset 61Ch[11]) 01b = General interrupt (INTx, MSI, or <b>PEX_INTA#</b> (Base mode) or VSx_PEX_INTA# (Virtual Switch mode)) 10b, 11b = <b>Reserved</b> <b>As Output:</b> 00b = From <b>PORT_GOOD11# Output Data</b> register (offset 624h[11]) 01b = PORT_GOOD11# 10b = SHP_PERST11# output 11b = <b>Reserved</b>	RWS	Yes	<b>TBD</b>
5	<b>PORT_GOOD11# Direction Control</b> <i>Note: This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> 0 = Input 1 = Output	RWS	Yes	<b>TBD</b>
31:6	<b>Reserved</b>	RsvdP	No	0-0h



**Register 13-140. 614h GPIO 0\_11 Input De-Bounce**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>The PORT_GOOD[0:11]# signals are de-bounced, using the <b>GPIO 0_11 Input De-Bounce</b> registers, respectively.</p> <p><i>Notes:</i> Register offsets <b>600h</b> and <b>604h</b>, referenced within this register, are located as follows – Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port.</p> <p>PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) <b>6A0h</b> through <b>6A8h</b>.</p> <p><i>Table 13-5</i> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
0	<p><b>PORT_GOOD0# Input De-Bounce Control</b></p> <p>Controls de-bounce when PORT_GOOD0# is configured as an input (offset <b>600h[2]</b>, is Cleared).</p> <p>0 = PORT_GOOD0# input is not de-bounced</p> <p>1 = PORT_GOOD0# input is de-bounced; de-bounce time is approximately 1.3 ms</p>	RWS	Yes	0
1	<p><b>PORT_GOOD1# Input De-Bounce Control</b></p> <p>Controls de-bounce when PORT_GOOD1# is configured as an input (offset <b>600h[5]</b>, is Cleared).</p> <p>0 = PORT_GOOD1# input is not de-bounced</p> <p>1 = PORT_GOOD1# input is de-bounced; de-bounce time is approximately 1.3 ms</p>	RWS	Yes	0
2	<p><b>PORT_GOOD2# Input De-Bounce Control</b></p> <p>Controls de-bounce when PORT_GOOD2# is configured as an input (offset <b>600h[8]</b>, is Cleared).</p> <p>0 = PORT_GOOD2# input is not de-bounced</p> <p>1 = PORT_GOOD2# input is de-bounced; de-bounce time is approximately 1.3 ms</p>	RWS	Yes	0
3	<p><b>PORT_GOOD3# Input De-Bounce Control</b></p> <p>Controls de-bounce when PORT_GOOD3# is configured as an input (offset <b>600h[11]</b>, is Cleared).</p> <p>0 = PORT_GOOD3# input is not de-bounced</p> <p>1 = PORT_GOOD3# input is de-bounced; de-bounce time is approximately 1.3 ms</p>	RWS	Yes	0

**Register 13-140. 614h GPIO 0\_11 Input De-Bounce**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the**  
**Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>PORT_GOOD4# Input De-Bounce Control</b> Controls de-bounce when PORT_GOOD4# is configured as an input (offset 600h[14], is Cleared). 0 = PORT_GOOD4# input is not de-bounced 1 = PORT_GOOD4# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
5	<b>PORT_GOOD5# Input De-Bounce Control</b> Controls de-bounce when PORT_GOOD5# is configured as an input (offset 600h[17], is Cleared). 0 = PORT_GOOD5# input is not de-bounced 1 = PORT_GOOD5# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
6	<b>PORT_GOOD6# Input De-Bounce Control</b> Controls de-bounce when PORT_GOOD6# is configured as an input (offset 600h[20], is Cleared). 0 = PORT_GOOD6# input is not de-bounced 1 = PORT_GOOD6# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
7	<b>PORT_GOOD7# Input De-Bounce Control</b> Controls de-bounce when PORT_GOOD7# is configured as an input (offset 600h[23], is Cleared). 0 = PORT_GOOD7# input is not de-bounced 1 = PORT_GOOD7# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0

**Register 13-140. 614h GPIO 0\_11 Input De-Bounce**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>PORT_GOOD8# Input De-Bounce Control</b> Controls de-bounce when PORT_GOOD8# is configured as an input (offset 600h[26], is Cleared). 0 = PORT_GOOD8# input is not de-bounced 1 = PORT_GOOD8# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
9	<b>PORT_GOOD9# Input De-Bounce Control</b> Controls de-bounce when PORT_GOOD9# is configured as an input (offset 600h[29], is Cleared). 0 = PORT_GOOD9# input is not de-bounced 1 = PORT_GOOD9# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
10	<b>PORT_GOOD10# Input De-Bounce Control</b> Controls de-bounce when PORT_GOOD10# is configured as an input (offset 604h[2], is Cleared). 0 = PORT_GOOD10# input is not de-bounced 1 = PORT_GOOD10# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
11	<b>PORT_GOOD11# Input De-Bounce Control</b> <i><b>Note:</b> This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> Controls de-bounce when PORT_GOOD11# is configured as an input (offset 604h[5], is Cleared). 0 = PORT_GOOD11# input is not de-bounced 1 = PORT_GOOD11# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

**Register 13-141. 61Ch GPIO 0\_11 Input Data****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>The PORT_GOOD[0:11]# input values are updated into the <b>GPIO 0_11 Input Data</b> registers, respectively.</p> <p><i>Notes:</i> Register offsets <b>600h</b> and <b>604h</b>, referenced within this register, are located as follows – Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port.</p> <p>PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) <b>6A0h</b> through <b>6A8h</b>.</p> <p><i>Table 13-5</i> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
0	<b>PORT_GOOD0# Input Data</b> If PORT_GOOD0# is configured as an output (offset 600h[2], is Set), Reads return a value of 0. If PORT_GOOD0# is configured as an input (offset 600h[2], is Cleared), Reads return the logic value of the voltage on PORT_GOOD0#.	ROS	No	0
1	<b>PORT_GOOD1# Input Data</b> If PORT_GOOD1# is configured as an output (offset 600h[5], is Set), Reads return a value of 0. If PORT_GOOD1# is configured as an input (offset 600h[5], is Cleared), Reads return the logic value of the voltage on PORT_GOOD1#.	ROS	No	0
2	<b>PORT_GOOD2# Input Data</b> If PORT_GOOD2# is configured as an output (offset 600h[8], is Set), Reads return a value of 0. If PORT_GOOD2# is configured as an input (offset 600h[8], is Cleared), Reads return the logic value of the voltage on PORT_GOOD2#.	ROS	No	0
3	<b>PORT_GOOD3# Input Data</b> If PORT_GOOD3# is configured as an output (offset 600h[11], is Set), Reads return a value of 0. If PORT_GOOD3# is configured as an input (offset 600h[11], is Cleared), Reads return the logic value of the voltage on PORT_GOOD3#.	ROS	No	0

**Register 13-141. 61Ch GPIO 0\_11 Input Data****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>PORT_GOOD4# Input Data</b> If PORT_GOOD4# is configured as an output (offset 600h[14], is Set), Reads return a value of 0. If PORT_GOOD4# is configured as an input (offset 600h[14], is Cleared), Reads return the logic value of the voltage on PORT_GOOD4#.	ROS	No	0
5	<b>PORT_GOOD5# Input Data</b> If PORT_GOOD5# is configured as an output (offset 600h[17], is Set), Reads return a value of 0. If PORT_GOOD5# is configured as an input (offset 600h[17], is Cleared), Reads return the logic value of the voltage on PORT_GOOD5#.	ROS	No	0
6	<b>PORT_GOOD6# Input Data</b> If PORT_GOOD6# is configured as an output (offset 600h[20], is Set), Reads return a value of 0. If PORT_GOOD6# is configured as an input (offset 600h[20], is Cleared), Reads return the logic value of the voltage on PORT_GOOD6#.	ROS	No	0
7	<b>PORT_GOOD7# Input Data</b> If PORT_GOOD7# is configured as an output (offset 600h[23], is Set), Reads return a value of 0. If PORT_GOOD7# is configured as an input (offset 600h[23], is Cleared), Reads return the logic value of the voltage on PORT_GOOD7#.	ROS	No	0
8	<b>PORT_GOOD8# Input Data</b> If PORT_GOOD8# is configured as an output (offset 600h[26], is Set), Reads return a value of 0. If PORT_GOOD8# is configured as an input (offset 600h[26], is Cleared), Reads return the logic value of the voltage on PORT_GOOD8#.	ROS	No	0
9	<b>PORT_GOOD9# Input Data</b> If PORT_GOOD9# is configured as an output (offset 600h[29], is Set), Reads return a value of 0. If PORT_GOOD9# is configured as an input (offset 600h[29], is Cleared), Reads return the logic value of the voltage on PORT_GOOD9#.	ROS	No	0
10	<b>PORT_GOOD10# Input Data</b> If PORT_GOOD10# is configured as an output (offset 604h[2], is Set), Reads return a value of 0. If PORT_GOOD10# is configured as an input (offset 604h[2], is Cleared), Reads return the logic value of the voltage on PORT_GOOD10#.	ROS	No	0
11	<b>PORT_GOOD11# Input Data</b> <i>Note: This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> If PORT_GOOD11# is configured as an output (offset 604h[5], is Set), Reads return a value of 0. If PORT_GOOD11# is configured as an input (offset 604h[5], is Cleared), Reads return the logic value of the voltage on PORT_GOOD11#.	ROS	No	0
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

**Register 13-142. 624h GPIO 0\_11 Output Data**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>The <b>GPIO 0_11 Output Data</b> registers control the PORT_GOOD[0:11]# outputs, respectively.</p> <p><i>Notes:</i> Register offsets <b>600h</b> and <b>604h</b>, referenced within this register, are located as follows – Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port.</p> <p>PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) <b>6A0h</b> through <b>6A8h</b>.</p> <p><i>Table 13-5</i> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
0	<b>PORT_GOOD0# Output Data</b> If PORT_GOOD0# is configured as an output (offset 600h[2], is Set), the value written to this bit is immediately driven to the PORT_GOOD0# output. Reads return the value written.	RWS	Yes	0
1	<b>PORT_GOOD1# Output Data</b> If PORT_GOOD1# is configured as an output (offset 600h[5], is Set), the value written to this bit is immediately driven to the PORT_GOOD1# output. Reads return the value written.	RWS	Yes	0
2	<b>PORT_GOOD2# Output Data</b> If PORT_GOOD2# is configured as an output (offset 600h[8], is Set), the value written to this bit is immediately driven to the PORT_GOOD2# output. Reads return the value written.	RWS	Yes	0
3	<b>PORT_GOOD3# Output Data</b> If PORT_GOOD3# is configured as an output (offset 600h[11], is Set), the value written to this bit is immediately driven to the PORT_GOOD3# output. Reads return the value written.	RWS	Yes	0
4	<b>PORT_GOOD4# Output Data</b> If PORT_GOOD4# is configured as an output (offset 600h[14], is Set), the value written to this bit is immediately driven to the PORT_GOOD4# output. Reads return the value written.	RWS	Yes	0
5	<b>PORT_GOOD5# Output Data</b> If PORT_GOOD5# is configured as an output (offset 600h[17], is Set), the value written to this bit is immediately driven to the PORT_GOOD5# output. Reads return the value written.	RWS	Yes	0
6	<b>PORT_GOOD6# Output Data</b> If PORT_GOOD6# is configured as an output (offset 600h[20], is Set), the value written to this bit is immediately driven to the PORT_GOOD6# output. Reads return the value written.	RWS	Yes	0
7	<b>PORT_GOOD7# Output Data</b> If PORT_GOOD7# is configured as an output (offset 600h[23], is Set), the value written to this bit is immediately driven to the PORT_GOOD7# output. Reads return the value written.	RWS	Yes	0

**Register 13-142. 624h GPIO 0\_11 Output Data****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>PORT_GOOD8# Output Data</b> If PORT_GOOD8# is configured as an output (offset 600h[26], is Set), the value written to this bit is immediately driven to the PORT_GOOD8# output. Reads return the value written.	RWS	Yes	0
9	<b>PORT_GOOD9# Output Data</b> If PORT_GOOD9# is configured as an output (offset 600h[29], is Set), the value written to this bit is immediately driven to the PORT_GOOD9# output. Reads return the value written.	RWS	Yes	0
10	<b>PORT_GOOD10# Output Data</b> If PORT_GOOD10# is configured as an output (offset 604h[2], is Set), the value written to this bit is immediately driven to the PORT_GOOD10# output. Reads return the value written.	RWS	Yes	0
11	<b>PORT_GOOD11# Output Data</b> <i>Note: This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> If PORT_GOOD11# is configured as an output (offset 604h[5], is Set), the value written to this bit is immediately driven to the PORT_GOOD11# output. Reads return the value written.	RWS	Yes	0
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

**Register 13-143. 62Ch GPIO 0\_11 Interrupt Polarity**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>The <b>GPIO 0_11 Interrupt Polarity</b> registers control the PORT_GOOD[0:11]# Interrupt input polarities, respectively.</p> <p><i>Notes:</i> PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) 6A0h through 6A8h).</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
0	<b>PORT_GOOD0# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High. 0 = PORT_GOOD0# Interrupt input is Active-Low 1 = PORT_GOOD0# Interrupt input is Active-High	RWS	Yes	0
1	<b>PORT_GOOD1# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High. 0 = PORT_GOOD1# Interrupt input is Active-Low 1 = PORT_GOOD1# Interrupt input is Active-High	RWS	Yes	0
2	<b>PORT_GOOD2# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High. 0 = PORT_GOOD2# Interrupt input is Active-Low 1 = PORT_GOOD2# Interrupt input is Active-High	RWS	Yes	0
3	<b>PORT_GOOD3# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High. 0 = PORT_GOOD3# Interrupt input is Active-Low 1 = PORT_GOOD3# Interrupt input is Active-High	RWS	Yes	0
4	<b>PORT_GOOD4# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High. 0 = PORT_GOOD4# Interrupt input is Active-Low 1 = PORT_GOOD4# Interrupt input is Active-High	RWS	Yes	0
5	<b>PORT_GOOD5# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High. 0 = PORT_GOOD5# Interrupt input is Active-Low 1 = PORT_GOOD5# Interrupt input is Active-High	RWS	Yes	0
6	<b>PORT_GOOD6# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High. 0 = PORT_GOOD6# Interrupt input is Active-Low 1 = PORT_GOOD6# Interrupt input is Active-High	RWS	Yes	0
7	<b>PORT_GOOD7# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High. 0 = PORT_GOOD7# Interrupt input is Active-Low 1 = PORT_GOOD7# Interrupt input is Active-High	RWS	Yes	0



**Register 13-143. 62Ch GPIO 0\_11 Interrupt Polarity**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>PORT_GOOD8# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High. 0 = PORT_GOOD8# Interrupt input is Active-Low 1 = PORT_GOOD8# Interrupt input is Active-High	RWS	Yes	0
9	<b>PORT_GOOD9# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High. 0 = PORT_GOOD9# Interrupt input is Active-Low 1 = PORT_GOOD9# Interrupt input is Active-High	RWS	Yes	0
10	<b>PORT_GOOD10# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High. 0 = PORT_GOOD10# Interrupt input is Active-Low 1 = PORT_GOOD10# Interrupt input is Active-High	RWS	Yes	0
11	<b>PORT_GOOD11# Interrupt Polarity</b> <i><b>Note:</b> This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> Controls whether GPIO Interrupt input is Active-Low or Active-High. 0 = PORT_GOOD11# Interrupt input is Active-Low 1 = PORT_GOOD11# Interrupt input is Active-High	RWS	Yes	0
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

**Register 13-144. 634h GPIO 0\_11 Interrupt Status**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>The PORT_GOOD[0:11]# Interrupt input status values are updated into the <b>GPIO 0_11 Interrupt Status</b> registers, respectively.</p> <p>Interrupt status remains Set, as long the corresponding PORT_GOODx# signal is asserted, and Clears on its own when the corresponding PORT_GOODx# input de-asserts to the inactive state.</p> <p>The active state of each interrupt is controlled by its respective <b>GPIO 0_11 Interrupt Polarity</b> register bits (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port. offset 62Ch).</p> <p><b>Notes:</b> The bits in this register can be masked by their respective <b>GPIO 0_11 Interrupt Mask</b> register bits (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port. offset 63Ch).</p> <p>PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) 6A0h through 6A8h).</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
0	<b>PORT_GOOD0# Interrupt Status</b> Indicates whether the GPIO interrupt is inactive or active. 0 = PORT_GOOD0# interrupt is inactive 1 = PORT_GOOD0# interrupt is active	ROS	No	0
1	<b>PORT_GOOD1# Interrupt Status</b> Indicates whether the GPIO interrupt is inactive or active. 0 = PORT_GOOD1# interrupt is inactive 1 = PORT_GOOD1# interrupt is active	ROS	No	0
2	<b>PORT_GOOD2# Interrupt Status</b> Indicates whether the GPIO interrupt is inactive or active. 0 = PORT_GOOD2# interrupt is inactive 1 = PORT_GOOD2# interrupt is active	ROS	No	0
3	<b>PORT_GOOD3# Interrupt Status</b> Indicates whether the GPIO interrupt is inactive or active. 0 = PORT_GOOD3# interrupt is inactive 1 = PORT_GOOD3# interrupt is active	ROS	No	0

**Register 13-144. 634h GPIO 0\_11 Interrupt Status**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>PORT_GOOD4# Interrupt Status</b> Indicates whether the GPIO interrupt is inactive or active. 0 = PORT_GOOD4# interrupt is inactive 1 = PORT_GOOD4# interrupt is active	ROS	No	0
5	<b>PORT_GOOD5# Interrupt Status</b> Indicates whether the GPIO interrupt is inactive or active. 0 = PORT_GOOD5# interrupt is inactive 1 = PORT_GOOD5# interrupt is active	ROS	No	0
6	<b>PORT_GOOD6# Interrupt Status</b> Indicates whether the GPIO interrupt is inactive or active. 0 = PORT_GOOD6# interrupt is inactive 1 = PORT_GOOD6# interrupt is active	ROS	No	0
7	<b>PORT_GOOD7# Interrupt Status</b> Indicates whether the GPIO interrupt is inactive or active. 0 = PORT_GOOD7# interrupt is inactive 1 = PORT_GOOD7# interrupt is active	ROS	No	0
8	<b>PORT_GOOD8# Interrupt Status</b> Indicates whether the GPIO interrupt is inactive or active. 0 = PORT_GOOD8# interrupt is inactive 1 = PORT_GOOD8# interrupt is active	ROS	No	0
9	<b>PORT_GOOD9# Interrupt Status</b> Indicates whether the GPIO interrupt is inactive or active. 0 = PORT_GOOD9# interrupt is inactive 1 = PORT_GOOD9# interrupt is active	ROS	No	0
10	<b>PORT_GOOD10# Interrupt Status</b> Indicates whether the GPIO interrupt is inactive or active. 0 = PORT_GOOD10# interrupt is inactive 1 = PORT_GOOD10# interrupt is active	ROS	No	0
11	<b>PORT_GOOD11# Interrupt Status</b> <i>Note: This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> Indicates whether the GPIO interrupt is inactive or active. 0 = PORT_GOOD11# interrupt is inactive 1 = PORT_GOOD11# interrupt is active	ROS	No	0
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

**Register 13-145. 63Ch GPIO 0\_11 Interrupt Mask**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>The PORT_GOOD[0:11]# Interrupt inputs are masked, using the <b>GPIO 0_11 Interrupt Mask</b> registers, respectively.</p> <p><b>Notes:</b> The bits in this register can be used to mask their respective <b>GPIO 0_11 Interrupt Status</b> register bits (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port. offset 634h).</p> <p>PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) 6A0h through 6A8h).</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
0	<p><b>PORT_GOOD0# Interrupt Mask</b>  Indicates whether the GPIO interrupt is masked or not masked.</p> <p>0 = PORT_GOOD0# interrupt is not masked.  1 = PORT_GOOD0# interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.</p>	RWS	Yes	1
1	<p><b>PORT_GOOD1# Interrupt Mask</b>  Indicates whether the GPIO interrupt is masked or not masked.</p> <p>0 = PORT_GOOD1# interrupt is not masked.  1 = PORT_GOOD1# interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.</p>	RWS	Yes	1
2	<p><b>PORT_GOOD2# Interrupt Mask</b>  Indicates whether the GPIO interrupt is masked or not masked.</p> <p>0 = PORT_GOOD2# interrupt is not masked.  1 = PORT_GOOD2# interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.</p>	RWS	Yes	1
3	<p><b>PORT_GOOD3# Interrupt Mask</b>  Indicates whether the GPIO interrupt is masked or not masked.</p> <p>0 = PORT_GOOD3# interrupt is not masked.  1 = PORT_GOOD3# interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.</p>	RWS	Yes	1

**Register 13-145. 63Ch GPIO 0\_11 Interrupt Mask**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>PORT_GOOD4# Interrupt Mask</b> Indicates whether the GPIO interrupt is masked or not masked. 0 = PORT_GOOD4# interrupt is not masked. 1 = PORT_GOOD4# interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
5	<b>PORT_GOOD5# Interrupt Mask</b> Indicates whether the GPIO interrupt is masked or not masked. 0 = PORT_GOOD5# interrupt is not masked. 1 = PORT_GOOD5# interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
6	<b>PORT_GOOD6# Interrupt Mask</b> Indicates whether the GPIO interrupt is masked or not masked. 0 = PORT_GOOD6# interrupt is not masked. 1 = PORT_GOOD6# interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
7	<b>PORT_GOOD7# Interrupt Mask</b> Indicates whether the GPIO interrupt is masked or not masked. 0 = PORT_GOOD7# interrupt is not masked. 1 = PORT_GOOD7# interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1

**Register 13-145. 63Ch GPIO 0\_11 Interrupt Mask**  
**(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>PORT_GOOD8# Interrupt Mask</b> Indicates whether the GPIO interrupt is masked or not masked. 0 = PORT_GOOD8# interrupt is not masked. 1 = PORT_GOOD8# interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
9	<b>PORT_GOOD9# Interrupt Mask</b> Indicates whether the GPIO interrupt is masked or not masked. 0 = PORT_GOOD9# interrupt is not masked. 1 = PORT_GOOD9# interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
10	<b>PORT_GOOD10# Interrupt Mask</b> Indicates whether the GPIO interrupt is masked or not masked. 0 = PORT_GOOD10# interrupt is not masked. 1 = PORT_GOOD10# interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
11	<b>PORT_GOOD11# Interrupt Mask</b> <i><b>Note:</b> This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> Indicates whether the GPIO interrupt is masked or not masked. 0 = PORT_GOOD11# interrupt is not masked. 1 = PORT_GOOD11# interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

**Register 13-146. 64Ch Virtual Switch GPIO Update**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>VS GPIOs Update</b> After the Management Port, serial EEPROM, and/or I <sup>2</sup> C/SMBus configures the PORT_GOODx# signals, and assigns these individual signals to various virtual switches, by programming the <b>VSx GPIO_PG 0_11 Assignment</b> register(s) (Port 0, accessible through the Management Port, offset(s) 650h through 664h, respectively), Set this bit to cause the GPIO assignments to take effect.	RW1CS	Yes	0
31:1	<b>Reserved</b>	RsvdP	No	0-0h

**Register 13-147. 650h VS0 GPIO\_PG 0\_11 Assignment**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>Each of bits [11:0] in this register uniquely corresponds to one of 12 PORT_GOODx# signals, where <i>x</i> in the signal name matches the bit position within this register. If bit(s) in this register are Set, the corresponding PORT_GOODx# signal(s) is (are) assigned to VS0. After GPIO assignment is complete (by the Management Port or I<sup>2</sup>C/SMBus Setting the <b>Virtual Switch GPIO Update</b> register <b>VS GPIOs Update</b> bit (Port 0, accessible through the Management Port, offset 64Ch[0]), the <b>Virtual Switch GPIO_PG 0_11 Availability</b> register <b>Number of GPIO_PGs Available</b> field (VS Upstream Port(s), offset A3Ch[3:0]) indicates the total quantity of PORT_GOODx# signals that are assigned to VS0 (same as the quantity of bits that are Set in this register).</p> <p><b>Notes:</b> Each virtual switch uses its own local set of <b>GPIO</b> registers (VS Upstream Port(s), offsets A34h through A54h). PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) 6A0h through 6A8h).</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
0	<b>VS0 GPIO_PG 0 Assignment</b> 0 = PORT_GOOD0# is not assigned to VS0 1 = PORT_GOOD0# is assigned to VS0	RWS	Yes	BALL
1	<b>VS0 GPIO_PG 1 Assignment</b> 0 = PORT_GOOD1# is not assigned to VS0 1 = PORT_GOOD1# is assigned to VS0	RWS	Yes	BALL
2	<b>VS0 GPIO_PG 2 Assignment</b> 0 = PORT_GOOD2# is not assigned to VS0 1 = PORT_GOOD2# is assigned to VS0	RWS	Yes	BALL
3	<b>VS0 GPIO_PG 3 Assignment</b> 0 = PORT_GOOD3# is not assigned to VS0 1 = PORT_GOOD3# is assigned to VS0	RWS	Yes	BALL

**Register 13-147. 650h VS0 GPIO\_PG 0\_11 Assignment**  
**(Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>VS0 GPIO_PG 4 Assignment</b> 0 = PORT_GOOD4# is not assigned to VS0 1 = PORT_GOOD4# is assigned to VS0	RWS	Yes	BALL
5	<b>VS0 GPIO_PG 5 Assignment</b> 0 = PORT_GOOD5# is not assigned to VS0 1 = PORT_GOOD5# is assigned to VS0	RWS	Yes	BALL
6	<b>VS0 GPIO_PG 6 Assignment</b> 0 = PORT_GOOD6# is not assigned to VS0 1 = PORT_GOOD6# is assigned to VS0	RWS	Yes	BALL
7	<b>VS0 GPIO_PG 7 Assignment</b> 0 = PORT_GOOD7# is not assigned to VS0 1 = PORT_GOOD7# is assigned to VS0	RWS	Yes	BALL
8	<b>VS0 GPIO_PG 8 Assignment</b> 0 = PORT_GOOD8# is not assigned to VS0 1 = PORT_GOOD8# is assigned to VS0	RWS	Yes	BALL
9	<b>VS0 GPIO_PG 9 Assignment</b> 0 = PORT_GOOD9# is not assigned to VS0 1 = PORT_GOOD9# is assigned to VS0	RWS	Yes	BALL
10	<b>VS0 GPIO_PG 10 Assignment</b> 0 = PORT_GOOD10# is not assigned to VS0 1 = PORT_GOOD10# is assigned to VS0	RWS	Yes	BALL
11	<b>VS0 GPIO_PG 11 Assignment</b> <i><b>Note:</b> This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> 0 = PORT_GOOD11# is not assigned to VS0 1 = PORT_GOOD11# is assigned to VS0	RWS	Yes	BALL
31:12	<b>Reserved</b>	RsvdP	No	0000_0h



**Register 13-148. 654h VS1 GPIO\_PG 0\_11 Assignment**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>Each of bits [11:0] in this register uniquely corresponds to one of 12 PORT_GOODx# signals, where <i>x</i> in the signal name matches the bit position within this register. If bit(s) in this register are Set, the corresponding PORT_GOODx# signal(s) is (are) assigned to VS0. After GPIO assignment is complete (by the Management Port or I<sup>2</sup>C/SMBus Setting the <b>Virtual Switch GPIO Update</b> register <b>VS GPIOs Update</b> bit (Port 0, accessible through the Management Port, offset 64Ch[0]), the <b>Virtual Switch GPIO_PG 0_11 Availability</b> register <b>Number of GPIO_PGs Available</b> field (VS Upstream Port(s), offset A3Ch[3:0]) indicates the total quantity of PORT_GOODx# signals that are assigned to VS1 (same as the quantity of bits that are Set in this register).</p> <p><b>Notes:</b> Each virtual switch uses its own local set of <b>GPIO</b> registers (VS Upstream Port(s), offsets A34h through A54h). PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) 6A0h through 6A8h).</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
0	<b>VS1 GPIO_PG 0 Assignment</b> 0 = PORT_GOOD0# is not assigned to VS1 1 = PORT_GOOD0# is assigned to VS1	RWS	Yes	BALL
1	<b>VS1 GPIO_PG 1 Assignment</b> 0 = PORT_GOOD1# is not assigned to VS1 1 = PORT_GOOD1# is assigned to VS1	RWS	Yes	BALL
2	<b>VS1 GPIO_PG 2 Assignment</b> 0 = PORT_GOOD2# is not assigned to VS1 1 = PORT_GOOD2# is assigned to VS1	RWS	Yes	BALL
3	<b>VS1 GPIO_PG 3 Assignment</b> 0 = PORT_GOOD3# is not assigned to VS1 1 = PORT_GOOD3# is assigned to VS1	RWS	Yes	BALL
4	<b>VS1 GPIO_PG 4 Assignment</b> 0 = PORT_GOOD4# is not assigned to VS1 1 = PORT_GOOD4# is assigned to VS1	RWS	Yes	BALL
5	<b>VS1 GPIO_PG 5 Assignment</b> 0 = PORT_GOOD5# is not assigned to VS1 1 = PORT_GOOD5# is assigned to VS1	RWS	Yes	BALL
6	<b>VS1 GPIO_PG 6 Assignment</b> 0 = PORT_GOOD6# is not assigned to VS1 1 = PORT_GOOD6# is assigned to VS1	RWS	Yes	BALL
7	<b>VS1 GPIO_PG 7 Assignment</b> 0 = PORT_GOOD7# is not assigned to VS1 1 = PORT_GOOD7# is assigned to VS1	RWS	Yes	BALL

**Register 13-148. 654h VS1 GPIO\_PG 0\_11 Assignment**  
**(Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>VS1 GPIO_PG 8 Assignment</b> 0 = PORT_GOOD8# is not assigned to VS1 1 = PORT_GOOD8# is assigned to VS1	RWS	Yes	BALL
9	<b>VS1 GPIO_PG 9 Assignment</b> 0 = PORT_GOOD9# is not assigned to VS1 1 = PORT_GOOD9# is assigned to VS1	RWS	Yes	BALL
10	<b>VS1 GPIO_PG 10 Assignment</b> 0 = PORT_GOOD10# is not assigned to VS1 1 = PORT_GOOD10# is assigned to VS1	RWS	Yes	BALL
11	<b>VS1 GPIO_PG 11 Assignment</b> <i><b>Note:</b> This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> 0 = PORT_GOOD11# is not assigned to VS1 1 = PORT_GOOD11# is assigned to VS1	RWS	Yes	BALL
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

**Register 13-149. 658h VS2 GPIO\_PG 0\_11 Assignment**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>Each of bits [11:0] in this register uniquely corresponds to one of 12 PORT_GOODx# signals, where <i>x</i> in the signal name matches the bit position within this register. If bit(s) in this register are Set, the corresponding PORT_GOODx# signal(s) is (are) assigned to VS0. After GPIO assignment is complete (by the Management Port or I<sup>2</sup>C/SMBus Setting the <b>Virtual Switch GPIO Update</b> register <b>VS GPIOs Update</b> bit (Port 0, accessible through the Management Port, offset 64Ch[0]), the <b>Virtual Switch GPIO_PG 0_11 Availability</b> register <b>Number of GPIO_PGs Available</b> field (VS Upstream Port(s), offset A3Ch[3:0]) indicates the total quantity of PORT_GOODx# signals that are assigned to VS2 (same as the quantity of bits that are Set in this register).</p> <p><b>Notes:</b> Each virtual switch uses its own local set of <b>GPIO</b> registers (VS Upstream Port(s), offsets A34h through A54h). PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) 6A0h through 6A8h).</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
0	<b>VS2 GPIO_PG 0 Assignment</b> 0 = PORT_GOOD0# is not assigned to VS2 1 = PORT_GOOD0# is assigned to VS2	RWS	Yes	BALL
1	<b>VS2 GPIO_PG 1 Assignment</b> 0 = PORT_GOOD1# is not assigned to VS2 1 = PORT_GOOD1# is assigned to VS2	RWS	Yes	BALL
2	<b>VS2 GPIO_PG 2 Assignment</b> 0 = PORT_GOOD2# is not assigned to VS2 1 = PORT_GOOD2# is assigned to VS2	RWS	Yes	BALL
3	<b>VS2 GPIO_PG 3 Assignment</b> 0 = PORT_GOOD3# is not assigned to VS2 1 = PORT_GOOD3# is assigned to VS2	RWS	Yes	BALL
4	<b>VS2 GPIO_PG 4 Assignment</b> 0 = PORT_GOOD4# is not assigned to VS2 1 = PORT_GOOD4# is assigned to VS2	RWS	Yes	BALL
5	<b>VS2 GPIO_PG 5 Assignment</b> 0 = PORT_GOOD5# is not assigned to VS2 1 = PORT_GOOD5# is assigned to VS2	RWS	Yes	BALL
6	<b>VS2 GPIO_PG 6 Assignment</b> 0 = PORT_GOOD6# is not assigned to VS2 1 = PORT_GOOD6# is assigned to VS2	RWS	Yes	BALL
7	<b>VS2 GPIO_PG 7 Assignment</b> 0 = PORT_GOOD7# is not assigned to VS2 1 = PORT_GOOD7# is assigned to VS2	RWS	Yes	BALL

**Register 13-149. 658h VS2 GPIO\_PG 0\_11 Assignment**  
**(Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>VS2 GPIO_PG 8 Assignment</b> 0 = PORT_GOOD8# is not assigned to VS2 1 = PORT_GOOD8# is assigned to VS2	RWS	Yes	BALL
9	<b>VS2 GPIO_PG 9 Assignment</b> 0 = PORT_GOOD9# is not assigned to VS2 1 = PORT_GOOD9# is assigned to VS2	RWS	Yes	BALL
10	<b>VS2 GPIO_PG 10 Assignment</b> 0 = PORT_GOOD10# is not assigned to VS2 1 = PORT_GOOD10# is assigned to VS2	RWS	Yes	BALL
11	<b>VS2 GPIO_PG 11 Assignment</b> <i><b>Note:</b> This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> 0 = PORT_GOOD11# is not assigned to VS2 1 = PORT_GOOD11# is assigned to VS2	RWS	Yes	BALL
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

**Register 13-150. 65Ch VS3 GPIO\_PG 0\_11 Assignment**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>Each of bits [11:0] in this register uniquely corresponds to one of 12 PORT_GOODx# signals, where <i>x</i> in the signal name matches the bit position within this register. If bit(s) in this register are Set, the corresponding PORT_GOODx# signal(s) is (are) assigned to VS0. After GPIO assignment is complete (by the Management Port or I<sup>2</sup>C/SMBus Setting the <b>Virtual Switch GPIO Update</b> register <b>VS GPIOs Update</b> bit (Port 0, accessible through the Management Port, offset 64Ch[0]), the <b>Virtual Switch GPIO_PG 0_11 Availability</b> register <b>Number of GPIO_PGs Available</b> field (VS Upstream Port(s), offset A3Ch[3:0]) indicates the total quantity of PORT_GOODx# signals that are assigned to VS3 (same as the quantity of bits that are Set in this register).</p> <p><b>Notes:</b> Each virtual switch uses its own local set of <b>GPIO</b> registers (VS Upstream Port(s), offsets A34h through A54h). PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) 6A0h through 6A8h).</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
0	<b>VS3 GPIO_PG 0 Assignment</b> 0 = PORT_GOOD0# is not assigned to VS3 1 = PORT_GOOD0# is assigned to VS3	RWS	Yes	BALL
1	<b>VS3 GPIO_PG 1 Assignment</b> 0 = PORT_GOOD1# is not assigned to VS3 1 = PORT_GOOD1# is assigned to VS3	RWS	Yes	BALL
2	<b>VS3 GPIO_PG 2 Assignment</b> 0 = PORT_GOOD2# is not assigned to VS3 1 = PORT_GOOD2# is assigned to VS3	RWS	Yes	BALL
3	<b>VS3 GPIO_PG 3 Assignment</b> 0 = PORT_GOOD3# is not assigned to VS3 1 = PORT_GOOD3# is assigned to VS3	RWS	Yes	BALL
4	<b>VS3 GPIO_PG 4 Assignment</b> 0 = PORT_GOOD4# is not assigned to VS3 1 = PORT_GOOD4# is assigned to VS3	RWS	Yes	BALL
5	<b>VS3 GPIO_PG 5 Assignment</b> 0 = PORT_GOOD5# is not assigned to VS3 1 = PORT_GOOD5# is assigned to VS3	RWS	Yes	BALL
6	<b>VS3 GPIO_PG 6 Assignment</b> 0 = PORT_GOOD6# is not assigned to VS3 1 = PORT_GOOD6# is assigned to VS3	RWS	Yes	BALL
7	<b>VS3 GPIO_PG 7 Assignment</b> 0 = PORT_GOOD7# is not assigned to VS3 1 = PORT_GOOD7# is assigned to VS3	RWS	Yes	BALL

**Register 13-150. 65Ch VS3 GPIO\_PG 0\_11 Assignment**  
**(Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>VS3 GPIO_PG 8 Assignment</b> 0 = PORT_GOOD8# is not assigned to VS3 1 = PORT_GOOD8# is assigned to VS3	RWS	Yes	BALL
9	<b>VS3 GPIO_PG 9 Assignment</b> 0 = PORT_GOOD9# is not assigned to VS3 1 = PORT_GOOD9# is assigned to VS3	RWS	Yes	BALL
10	<b>VS3 GPIO_PG 10 Assignment</b> 0 = PORT_GOOD10# is not assigned to VS3 1 = PORT_GOOD10# is assigned to VS3	RWS	Yes	BALL
11	<b>VS3 GPIO_PG 11 Assignment</b> <i><b>Note:</b> This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> 0 = PORT_GOOD11# is not assigned to VS3 1 = PORT_GOOD11# is assigned to VS3	RWS	Yes	BALL
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

**Register 13-151. 660h VS4 GPIO\_PG 0\_11 Assignment**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>Each of bits [11:0] in this register uniquely corresponds to one of 12 PORT_GOODx# signals, where <i>x</i> in the signal name matches the bit position within this register. If bit(s) in this register are Set, the corresponding PORT_GOODx# signal(s) is (are) assigned to VS0. After GPIO assignment is complete (by the Management Port or I<sup>2</sup>C/SMBus Setting the <b>Virtual Switch GPIO Update</b> register <b>VS GPIOs Update</b> bit (Port 0, accessible through the Management Port, offset 64Ch[0]), the <b>Virtual Switch GPIO_PG 0_11 Availability</b> register <b>Number of GPIO_PGs Available</b> field (VS Upstream Port(s), offset A3Ch[3:0]) indicates the total quantity of PORT_GOODx# signals that are assigned to VS4 (same as the quantity of bits that are Set in this register).</p> <p><b>Notes:</b> Each virtual switch uses its own local set of <b>GPIO</b> registers (VS Upstream Port(s), offsets A34h through A54h). PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) 6A0h through 6A8h).</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
0	<b>VS4 GPIO_PG 0 Assignment</b> 0 = PORT_GOOD0# is not assigned to VS4 1 = PORT_GOOD0# is assigned to VS4	RWS	Yes	BALL
1	<b>VS4 GPIO_PG 1 Assignment</b> 0 = PORT_GOOD1# is not assigned to VS4 1 = PORT_GOOD1# is assigned to VS4	RWS	Yes	BALL
2	<b>VS4 GPIO_PG 2 Assignment</b> 0 = PORT_GOOD2# is not assigned to VS4 1 = PORT_GOOD2# is assigned to VS4	RWS	Yes	BALL
3	<b>VS4 GPIO_PG 3 Assignment</b> 0 = PORT_GOOD3# is not assigned to VS4 1 = PORT_GOOD3# is assigned to VS4	RWS	Yes	BALL
4	<b>VS4 GPIO_PG 4 Assignment</b> 0 = PORT_GOOD4# is not assigned to VS4 1 = PORT_GOOD4# is assigned to VS4	RWS	Yes	BALL
5	<b>VS4 GPIO_PG 5 Assignment</b> 0 = PORT_GOOD5# is not assigned to VS4 1 = PORT_GOOD5# is assigned to VS4	RWS	Yes	BALL
6	<b>VS4 GPIO_PG 6 Assignment</b> 0 = PORT_GOOD6# is not assigned to VS4 1 = PORT_GOOD6# is assigned to VS4	RWS	Yes	BALL
7	<b>VS4 GPIO_PG 7 Assignment</b> 0 = PORT_GOOD7# is not assigned to VS4 1 = PORT_GOOD7# is assigned to VS4	RWS	Yes	BALL

**Register 13-151. 660h VS4 GPIO\_PG 0\_11 Assignment**  
**(Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>VS4 GPIO_PG 8 Assignment</b> 0 = PORT_GOOD8# is not assigned to VS4 1 = PORT_GOOD8# is assigned to VS4	RWS	Yes	BALL
9	<b>VS4 GPIO_PG 9 Assignment</b> 0 = PORT_GOOD9# is not assigned to VS4 1 = PORT_GOOD9# is assigned to VS4	RWS	Yes	BALL
10	<b>VS4 GPIO_PG 10 Assignment</b> 0 = PORT_GOOD10# is not assigned to VS4 1 = PORT_GOOD10# is assigned to VS4	RWS	Yes	BALL
11	<b>VS4 GPIO_PG 11 Assignment</b> <i><b>Note:</b> This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> 0 = PORT_GOOD11# is not assigned to VS4 1 = PORT_GOOD11# is assigned to VS4	RWS	Yes	BALL
31:12	<b>Reserved</b>	RsvdP	No	0000_0h



**Register 13-152. 664h VS5 GPIO\_PG 0\_11 Assignment**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>Each of bits [11:0] in this register uniquely corresponds to one of 12 PORT_GOODx# signals, where <i>x</i> in the signal name matches the bit position within this register. If bit(s) in this register are Set, the corresponding PORT_GOODx# signal(s) is (are) assigned to VS0. After GPIO assignment is complete (by the Management Port or I<sup>2</sup>C/SMBus Setting the <b>Virtual Switch GPIO Update</b> register <b>VS GPIOs Update</b> bit (Port 0, accessible through the Management Port, offset 64Ch[0]), the <b>Virtual Switch GPIO_PG 0_11 Availability</b> register <b>Number of GPIO_PGs Available</b> field (VS Upstream Port(s), offset A3Ch[3:0]) indicates the total quantity of PORT_GOODx# signals that are assigned to VS5 (same as the quantity of bits that are Set in this register).</p> <p><b>Notes:</b> Each virtual switch uses its own local set of <b>GPIO</b> registers (VS Upstream Port(s), offsets A34h through A54h). PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) 6A0h through 6A8h).</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
0	<b>VS5 GPIO_PG 0 Assignment</b> 0 = PORT_GOOD0# is not assigned to VS5 1 = PORT_GOOD0# is assigned to VS5	RWS	Yes	BALL
1	<b>VS5 GPIO_PG 1 Assignment</b> 0 = PORT_GOOD1# is not assigned to VS5 1 = PORT_GOOD1# is assigned to VS5	RWS	Yes	BALL
2	<b>VS5 GPIO_PG 2 Assignment</b> 0 = PORT_GOOD2# is not assigned to VS5 1 = PORT_GOOD2# is assigned to VS5	RWS	Yes	BALL
3	<b>VS5 GPIO_PG 3 Assignment</b> 0 = PORT_GOOD3# is not assigned to VS5 1 = PORT_GOOD3# is assigned to VS5	RWS	Yes	BALL
4	<b>VS5 GPIO_PG 4 Assignment</b> 0 = PORT_GOOD4# is not assigned to VS5 1 = PORT_GOOD4# is assigned to VS5	RWS	Yes	BALL
5	<b>VS5 GPIO_PG 5 Assignment</b> 0 = PORT_GOOD5# is not assigned to VS5 1 = PORT_GOOD5# is assigned to VS5	RWS	Yes	BALL
6	<b>VS5 GPIO_PG 6 Assignment</b> 0 = PORT_GOOD6# is not assigned to VS5 1 = PORT_GOOD6# is assigned to VS5	RWS	Yes	BALL
7	<b>VS5 GPIO_PG 7 Assignment</b> 0 = PORT_GOOD7# is not assigned to VS5 1 = PORT_GOOD7# is assigned to VS5	RWS	Yes	BALL

**Register 13-152. 664h VS5 GPIO\_PG 0\_11 Assignment**  
**(Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>VS5 GPIO_PG 8 Assignment</b> 0 = PORT_GOOD8# is not assigned to VS5 1 = PORT_GOOD8# is assigned to VS5	RWS	Yes	BALL
9	<b>VS5 GPIO_PG 9 Assignment</b> 0 = PORT_GOOD9# is not assigned to VS5 1 = PORT_GOOD9# is assigned to VS5	RWS	Yes	BALL
10	<b>VS5 GPIO_PG 10 Assignment</b> 0 = PORT_GOOD10# is not assigned to VS5 1 = PORT_GOOD10# is assigned to VS5	RWS	Yes	BALL
11	<b>VS5 GPIO_PG 11 Assignment</b> <i><b>Note:</b> This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> 0 = PORT_GOOD11# is not assigned to VS5 1 = PORT_GOOD11# is assigned to VS5	RWS	Yes	BALL
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

13.16.10    **Device-Specific Registers – PORT\_GOOD Selector (Offsets 6A0h – 6A8h)**

This section details the Device-Specific PORT\_GOOD Selector registers. [Table 13-27](#) defines the register map.

**Table 13-27.    Device-Specific PORT\_GOOD Selector Register Map (Offsets 6A0h – 6A8h) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT_GOOD Selector 0																																6A0h
PORT_GOOD Selector 1																																6A4h
PORT_GOOD Selector 2																																6A8h

**Register 13-153. 6A0h PORT\_GOOD Selector 0****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register assigns PORT_GOODx#/GPIOx signal(s) that are enabled for PORT_GOOD output functionality (in the <b>GPIO 0_x Direction Control</b> register <i>Direction Control</i> bit(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset(s) 600h and/or 604h), in which default values are globally defined, according to the STRAP_TESTMODE[2:0] input states), to designated Port(s). This register has no effect on PORT_GOODx#/GPIOx signals that are enabled for Serial Hot Plug (SHP_PERSTx#) or GPIO functionality.</p> <p>0_0000b = Port 0  0_0001b = Port 1  0_0010b = Port 2  0_0011b = Port 3  0_1000b = Port 8  0_1001b = Port 9  0_1010b = Port 10  0_1011b = Port 11  1_0000b = Port 16  1_0001b = Port 17  1_0010b = Port 18  1_0011b = Port 19</p> <p>All other <i>Port Up Select x</i> field encodings are <b>Reserved</b>.</p> <p><b>Notes:</b> PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) 6A0h through 6A8h).</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
4:0	<b>Port Up Select 0</b> Use this field to program a different Port to be associated with PORT_GOOD0#. The default for PORT_GOOD0# is Port 0.	RWS	Yes	0_0000b
7:5	<b>Reserved</b>	RsvdP	No	000b
12:8	<b>Port Up Select 1</b> Use this field to program a different Port to be associated with PORT_GOOD1#. The default for PORT_GOOD1# is Port 1.	RWS	Yes	0_0001b
15:13	<b>Reserved</b>	RsvdP	No	000b
20:16	<b>Port Up Select 2</b> Use this field to program a different Port to be associated with PORT_GOOD2#. The default for PORT_GOOD2# is Port 2.	RWS	Yes	0_0010b
23:21	<b>Reserved</b>	RsvdP	No	000b
28:24	<b>Port Up Select 3</b> Use this field to program a different Port to be associated with PORT_GOOD3#. The default for PORT_GOOD3# is Port 3.	RWS	Yes	0_0011b
31:29	<b>Reserved</b>	RsvdP	No	000b

**Register 13-154. 6A4h PORT\_GOOD Selector 1****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register assigns PORT_GOOD<sub>x</sub>/GPIO<sub>x</sub> signal(s) that are enabled for PORT_GOOD output functionality (in the <b>GPIO 0<sub>x</sub> Direction Control</b> register <i>Direction Control</i> bit(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset(s) 600h and/or 604h), in which default values are globally defined, according to the STRAP_TESTMODE[2:0] input states), to designated Port(s). This register has no effect on PORT_GOOD<sub>x</sub>/GPIO<sub>x</sub> signals that are enabled for Serial Hot Plug (SHP_PERST<sub>x</sub>) or GPIO functionality.</p> <p>0_0000b = Port 0  0_0001b = Port 1  0_0010b = Port 2  0_0011b = Port 3  0_1000b = Port 8  0_1001b = Port 9  0_1010b = Port 10  0_1011b = Port 11  1_0000b = Port 16  1_0001b = Port 17  1_0010b = Port 18  1_0011b = Port 19</p> <p>All other <i>Port Up Select x</i> field encodings are <b>Reserved</b>.</p> <p><b>Notes:</b> PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOOD<sub>x</sub> signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) 6A0h through 6A8h).</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STN<sub>x</sub>_PORTCFG<sub>x</sub> input states and/or serial EEPROM programming.</p>				
4:0	<b>Port Up Select 4</b> Use this field to program a different Port to be associated with PORT_GOOD4#. The default for PORT_GOOD4# is Port 8.	RWS	Yes	0_1000b
7:5	<b>Reserved</b>	RsvdP	No	000b
12:8	<b>Port Up Select 5</b> Use this field to program a different Port to be associated with PORT_GOOD5#. The default for PORT_GOOD5# is Port 9.	RWS	Yes	0_1001b
15:13	<b>Reserved</b>	RsvdP	No	000b
20:16	<b>Port Up Select 6</b> Use this field to program a different Port to be associated with PORT_GOOD6#. The default for PORT_GOOD6# is Port 10.	RWS	Yes	0_1010b
23:21	<b>Reserved</b>	RsvdP	No	000b
28:24	<b>Port Up Select 7</b> Use this field to program a different Port to be associated with PORT_GOOD7#. The default for PORT_GOOD7# is Port 11.	RWS	Yes	0_1011b
31:29	<b>Reserved</b>	RsvdP	No	000b

**Register 13-155. 6A8h PORT\_GOOD Selector 2****(Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register assigns PORT_GOODx#/GPIOx signal(s) that are enabled for PORT_GOOD output functionality (in the <b>GPIO 0_x Direction Control</b> register <i>Direction Control</i> bit(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset(s) <b>600h</b> and <b>604h</b>), in which default values are globally defined, according to the STRAP_TESTMODE[2:0] input states), to designated Port(s). This register has no effect on PORT_GOODx#/GPIOx signals that are enabled for Serial Hot Plug (SHP_PERSTx#) or GPIO functionality.</p> <p>0_0000b = Port 0  0_0001b = Port 1  0_0010b = Port 2  0_0011b = Port 3  0_1000b = Port 8  0_1001b = Port 9  0_1010b = Port 10  0_1011b = Port 11  1_0000b = Port 16  1_0001b = Port 17  1_0010b = Port 18  1_0011b = Port 19</p> <p>All other <i>Port Up Select x</i> field encodings are <b>Reserved</b>.</p> <p><b>Notes:</b> PORT_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively, by default. PORT_GOODx# signals can be re-assigned to any Port, however, by programming the <b>Port Good Selector x</b> register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) <b>6A0h</b> through <b>6A8h</b>).</p> <p><i>Table 13-5</i> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>				
4:0	<b>Port Up Select 8</b> Use this field to program a different Port to be associated with PORT_GOOD8#. The default for PORT_GOOD8# is Port 16.	RWS	Yes	1_0000b
7:5	<b>Reserved</b>	RsvdP	No	000b
12:8	<b>Port Up Select 9</b> Use this field to program a different Port to be associated with PORT_GOOD9#. The default for PORT_GOOD9# is Port 17.	RWS	Yes	1_0001b
15:13	<b>Reserved</b>	RsvdP	No	000b
20:16	<b>Port Up Select 10</b> Use this field to program a different Port to be associated with PORT_GOOD10#. The default for PORT_GOOD10# is Port 18.	RWS	Yes	1_0010b
23:21	<b>Reserved</b>	RsvdP	No	000b
28:24	<b>Port Up Select 11</b> <i>Note: This field is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this field is Reserved.</i> Use this field to program a different Port to be associated with PORT_GOOD11#. The default for PORT_GOOD11# is Port 19.	RWS	Yes	1_0011b
31:29	<b>Reserved</b>	RsvdP	No	000b

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

### 13.16.11 Device-Specific Registers – Error Checking and Debug (Offsets 700h – 75Ch)

This section details the Device-Specific Error Checking and Debug registers located at offsets 700h through 75Ch. [Table 13-28](#) defines the register map.

Other Device-Specific Error Checking and Debug registers are detailed in [Section 13.22.3, “Device-Specific Registers – Error Checking and Debug \(Offsets F70h – FB0h\).”](#)

**Table 13-28. Device-Specific Error Checking and Debug Register Map (Offsets 700h – 75Ch) (Ports<sup>a</sup>)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																
Device-Specific Error Status 1																																700h
Device-Specific Error Mask 1																																704h
Device-Specific Error Status 2																																708h
Device-Specific Error Mask 2																																70Ch
Reserved								Device-Specific Error Status 3																								710h
Reserved								Device-Specific Error Mask 3																								714h
Reserved								Device-Specific Error Status 4																								718h
Reserved								Device-Specific Error Mask 4																								71Ch
ECC Error Check Disable																																720h
Reserved																Gen 3 Framing Error Status																724h
Reserved																Gen 3 Framing Error Mask																728h
Reserved																Gen 3 Framing Error Disable																72Ch
Reserved																																730h – 75Ch

a. *Certain registers are Port-specific, others are Station-specific or Chip-specific; all are Device-specific.*

**Register 13-156. 700h Device-Specific Error Status 1**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>ECC is checked by the egress Port, and any ECC error is reported by the egress Station (containing the egress Port), when that Port reads the Packet data from the ingress Station RAM. <i>For example</i>, when Port 8 receives a packet, the data is stored in Station 1 RAM; then, if the egress Port is in Station 0 and an ECC error is detected when the Port reads the Station 1 RAM, the error is flagged in the Port 0 register bit that reflects Station 1 error status.</p> <p><b>Notes:</b> The bits in this register can be masked by their respective <i>Device-Specific Error Mask 1</i> register bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 704h).</p> <p>All errors in this register generate ERR_FATAL or ERR_NONFATAL Messages, if enabled by the following:</p> <ul style="list-style-type: none"> <li>• <i>Device-Specific Error Mask 1</i> register</li> <li>• <i>Uncorrectable Error Status</i> register <i>Uncorrectable Internal Error Status</i> bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset FB8h[22], is Set)</li> <li>• Port's <i>Device Status</i> register <i>Fatal Error Reporting Enable</i> and <i>Non-Fatal Error Reporting Enable</i> bits (offset 70h[2:1], respectively)</li> </ul> <p>The Stations are defined in Table 13-5. The Port 0 bits are for Station 0. The Port 8 bits are for Station 1. The Port 16 bits are for Station 2.</p>					
0	<b>Factory Test Only</b>		RW1CS	Yes	0
1	<b>Factory Test Only</b>	0	RW1CS	No	0
2	<b>Station 0 Egress Packet Link List RAM 1-Bit ECC Error Detected</b> 0 = No 1-Bit ECC error is detected 1 = Soft error is detected		RW1CS	Yes	0
3	<b>Station 1 Egress Packet Link List RAM 1-Bit ECC Error Detected</b> 0 = No 1-Bit ECC error is detected 1 = Soft error is detected		RW1CS	Yes	0
4	<b>Station 2 Egress Packet Link List RAM 1-Bit ECC Error Detected</b> 0 = No 1-Bit ECC error is detected 1 = Soft error is detected		RW1CS	Yes	0
7:5	<b>Factory Test Only</b>		RW1CS	Yes	000b
8	<b>Station 0 Header RAM0 Instance 0 Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0
9	<b>Station 1 Header RAM0 Instance 0 Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0
10	<b>Station 2 Header RAM0 Instance 0 Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0
11	<b>Station 0 Header RAM0 Instance 1 Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1



**Register 13-156. 700h Device-Specific Error Status 1****(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Station 1 Header RAM0 Instance 1 Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0
13	<b>Station 2 Header RAM0 Instance 1 Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0
14	<b>Station 0 Header RAM1 Instance 0 Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0
15	<b>Station 1 Header RAM1 Instance 0 Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0
16	<b>Station 2 Header RAM1 Instance 0 Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0
17	<b>Station 0 Header RAM1 Instance 1 Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0
18	<b>Station 1 Header RAM1 Instance 1 Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0
19	<b>Station 2 Header RAM1 Instance 1 Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0
22:20	<b>Factory Test Only</b>		RsvdP	No	000b
23	<b>Destination Queue Link List RAM 1-Bit ECC Error Detected</b> 0 = No 1-Bit ECC error is detected 1 = Soft error is detected		RW1CS	Yes	0

**Register 13-156. 700h Device-Specific Error Status 1**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Source Queue Link List RAM 1-Bit ECC Error Detected</b> 0 = No 1-Bit ECC error is detected 1 = Soft error is detected		RW1CS	Yes	0
25	<b>Retry Buffer 1-Bit ECC Error Detected</b> 0 = No 1-Bit ECC error is detected 1 = Soft error is detected		RW1CS	Yes	0
26	<b>Ingress Link List RAM Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0
27	<b>Source Queue Link List RAM2 Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0
28	<b>Destination Queue Data RAM Soft Error Counter Overflow Detected</b> 0 = No overflow is detected 1 = Soft error is detected		RW1CS	Yes	0
31:29	<b>Reserved</b>		RsvdP	No	000b

**Register 13-157. 704h Device-Specific Error Mask 1**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The bits in this register can be used to mask their respective <i>Device-Specific Error Status 1</i> register bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 700h).</p> <p>The Stations are defined in <a href="#">Table 13-5</a>. The Port 0 bits are for Station 0. The Port 8 bits are for Station 1. The Port 16 bits are for Station 2.</p>					
0	<b>Factory Test Only</b>		RWS	Yes	1
1	<b>Factory Test Only</b>	0	RWS	Yes	1
2	<b>Station 0 Egress Packet Link List RAM 1-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Egress Packet Link List RAM 1-Bit ECC Error Detected</i> bit is masked/disabled		RWS	Yes	1
3	<b>Station 1 Egress Packet Link List RAM 1-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Egress Packet Link List RAM 1-Bit ECC Error Detected</i> bit is masked/disabled		RWS	Yes	1
4	<b>Station 2 Egress Packet Link List RAM 1-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Egress Packet Link List RAM 1-Bit ECC Error Detected</i> bit is masked/disabled		RWS	Yes	1
7:5	<b>Factory Test Only</b>		RWS	Yes	000b
8	<b>Station 0 Header RAM0 Instance 0 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Header RAM0 Instance 0 Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1
9	<b>Station 1 Header RAM0 Instance 0 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Header RAM0 Instance 0 Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1
10	<b>Station 2 Header RAM0 Instance 0 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Header RAM0 Instance 0 Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1
11	<b>Station 0 Header RAM0 Instance 1 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Header RAM0 Instance 1 Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1

**Register 13-157. 704h Device-Specific Error Mask 1**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Station 1 Header RAM0 Instance 1 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Header RAM0 Instance 1 Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1
13	<b>Station 2 Header RAM0 Instance 1 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Header RAM0 Instance 1 Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1
14	<b>Station 0 Header RAM1 Instance 0 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Header RAM1 Instance 0 Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1
15	<b>Station 1 Header RAM1 Instance 0 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Header RAM1 Instance 0 Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1
16	<b>Station 2 Header RAM1 Instance 0 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Header RAM1 Instance 0 Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1
17	<b>Station 0 Header RAM1 Instance 1 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Header RAM1 Instance 1 Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1
18	<b>Station 1 Header RAM1 Instance 1 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Header RAM1 Instance 1 Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1
19	<b>Station 2 Header RAM1 Instance 1 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Header RAM1 Instance 1 Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1
22:20	<b>Factory Test Only</b>		RsvdP	No	000b
23	<b>Destination Queue Link List RAM 1-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Destination Queue Link List RAM 1-Bit ECC Error Detected</i> bit is masked/disabled		RWS	Yes	1

**Register 13-157. 704h Device-Specific Error Mask 1****(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Source Queue Link List RAM 1-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Source Queue Link List RAM 1-Bit ECC Error Detected</i> bit is masked/disabled		RWS	Yes	1
25	<b>Retry Buffer 1-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Retry Buffer 1-Bit ECC Error Detected</i> bit is masked/disabled		RWS	Yes	1
26	<b>Ingress Link List RAM Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Ingress Link List RAM Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1
27	<b>Source Queue Link List RAM2 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Source Queue Link List RAM2 Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1
28	<b>Destination Queue Data RAM Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Destination Queue Data RAM Soft Error Counter Overflow Detected</i> bit is masked/disabled		RWS	Yes	1
31:29	<b>Reserved</b>		RsvdP	No	000b

**Register 13-158. 708h Device-Specific Error Status 2**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>ECC is checked by the egress Port, and any ECC error is reported by the egress Station (containing the egress Port), when that Port reads the Packet data from the ingress Station RAM. <i>For example</i>, when Port 8 receives a packet, the data is stored in Station 1 RAM; then, if the egress Port is in Station 0 and an ECC error is detected when the Port reads the Station 1 RAM, the error is flagged in the Port 0 register bit that reflects Station 1 error status.</p> <p><b>Notes:</b> The bits in this register can be masked by their respective <b>Device-Specific Error Mask 2</b> register bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 70Ch).</p> <p>All errors in this register generate ERR_FATAL or ERR_NONFATAL Messages, if enabled by the following:</p> <ul style="list-style-type: none"> <li>• <b>Device-Specific Error Mask 2</b> register</li> <li>• <b>Uncorrectable Error Status</b> register <b>Uncorrectable Internal Error Status</b> bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset FB8h[22], is Set)</li> <li>• Port's <b>Device Status</b> register <b>Fatal Error Reporting Enable</b> and <b>Non-Fatal Error Reporting Enable</b> bits (offset 70h[2:1], respectively)</li> </ul> <p>The Stations are defined in Table 13-5. The Port 0 bits are for Station 0. The Port 8 bits are for Station 1. The Port 16 bits are for Station 2.</p>				
0	<b>Reserved</b>	RsvdP	No	0
1	<b>Ingress Freelist Underrun Error Detected</b> 0 = No error is detected 1 = Underrun error is detected	RW1CS	Yes	0
2	<b>Station 0 Egress Packet Link List RAM 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
3	<b>Station 1 Egress Packet Link List RAM 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
4	<b>Station 2 Egress Packet Link List RAM 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
7:5	<b>Factory Test Only</b>	RW1CS	Yes	000b
8	<b>Station 0 Header RAM0 Instance 0 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
9	<b>Station 1 Header RAM0 Instance 0 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
10	<b>Station 2 Header RAM0 Instance 0 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
11	<b>Station 0 Header RAM0 Instance 1 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 13-158. 708h Device-Specific Error Status 2****(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Station 1 Header RAM0 Instance 1 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
13	<b>Station 2 Header RAM0 Instance 1 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
14	<b>Station 0 Header RAM1 Instance 0 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
15	<b>Station 1 Header RAM1 Instance 0 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
16	<b>Station 2 Header RAM1 Instance 0 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
17	<b>Station 0 Header RAM1 Instance 1 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
18	<b>Station 1 Header RAM1 Instance 1 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
19	<b>Station 2 Header RAM1 Instance 1 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
22:20	<b>Factory Test Only</b>	RsvdP	No	000b
23	<b>Destination Queue Link List RAM 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0

**Register 13-158. 708h Device-Specific Error Status 2**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Source Queue Link List RAM 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
25	<b>Retry Buffer 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
26	<b>Ingress Link List RAM 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
27	<b>Source Queue Link List RAM2 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
28	<b>Destination Queue Data RAM 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
29	<b>Ingress Parity Error Detected</b> 0 = No error is detected 1 = Ingress Parity error detected	RW1CS	Yes	0
30	<b>Egress Parity Error Detected</b> 0 = No error is detected 1 = Egress Parity error detected	RW1CS	Yes	0
31	<b>Reserved</b>	RsvdP	No	0



**Register 13-159. 70Ch Device-Specific Error Mask 2**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The bits in this register can be used to mask their respective <i>Device-Specific Error Status 2</i> register bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 708h).</p> <p>The Stations are defined in <a href="#">Table 13-5</a>. The Port 0 bits are for Station 0. The Port 8 bits are for Station 1. The Port 16 bits are for Station 2.</p>				
0	<i>Reserved</i>	RsvdP	No	0
1	<b>Ingress Freelist Underrun Error Mask</b> 0 = No effect on reporting activity 1 = <i>Ingress Freelist Underrun Error Detected</i> bit is masked/disabled	RWS	Yes	1
2	<b>Station 0 Egress Packet Link List RAM 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Egress Packet Link List RAM 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
3	<b>Station 1 Egress Packet Link List RAM 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Egress Packet Link List RAM 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
4	<b>Station 2 Egress Packet Link List RAM 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Egress Packet Link List RAM 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
7:5	<i>Factory Test Only</i>	RWS	Yes	000b
8	<b>Station 0 Header RAM0 Instance 0 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Header RAM0 Instance 0 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
9	<b>Station 1 Header RAM0 Instance 0 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Header RAM0 Instance 0 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
10	<b>Station 2 Header RAM0 Instance 0 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Header RAM0 Instance 0 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
11	<b>Station 0 Header RAM0 Instance 1 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Header RAM0 Instance 1 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1

**Register 13-159. 70Ch Device-Specific Error Mask 2**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Station 1 Header RAM0 Instance 1 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Header RAM0 Instance 1 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
13	<b>Station 2 Header RAM0 Instance 1 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Header RAM0 Instance 1 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
14	<b>Station 0 Header RAM1 Instance 0 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Header RAM1 Instance 0 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
15	<b>Station 1 Header RAM1 Instance 0 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Header RAM1 Instance 0 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
16	<b>Station 2 Header RAM1 Instance 0 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Header RAM1 Instance 0 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
17	<b>Station 0 Header RAM1 Instance 1 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Header RAM1 Instance 1 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
18	<b>Station 1 Header RAM1 Instance 1 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Header RAM1 Instance 1 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
19	<b>Station 2 Header RAM1 Instance 1 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Header RAM1 Instance 1 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
22:20	<b>Factory Test Only</b>	RsvdP	No	000b
23	<b>Destination Queue Link List RAM 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Destination Queue Link List RAM 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1

**Register 13-159. 70Ch Device-Specific Error Mask 2**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Source Queue Link List RAM 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Source Queue Link List RAM 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
25	<b>Retry Buffer 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Retry Buffer 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
26	<b>Ingress Link List RAM 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Ingress Link List RAM 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
27	<b>Source Queue Link List RAM2 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Source Queue Link List RAM2 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
28	<b>Destination Queue Data RAM 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Destination Queue Data RAM 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
29	<b>Ingress Parity Error Mask</b> 0 = No effect on reporting activity 1 = <i>Ingress Parity Error Detected</i> bit is masked/disabled	RWS	Yes	1
30	<b>Egress Parity Error Mask</b> 0 = No effect on reporting activity 1 = <i>Egress Parity Error Detected</i> bit is masked/disabled	RWS	Yes	1
31	<b>Reserved</b>	RsvdP	No	0

**Register 13-160. 710h Device-Specific Error Status 3**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>ECC is checked by the egress Port, and any ECC error is reported by the egress Station (containing the egress Port), when that Port reads the Packet data from the ingress Station RAM. <i>For example</i>, when Port 8 receives a packet, the data is stored in Station 1 RAM; then, if the egress Port is in Station 0 and an ECC error is detected when the Port reads the Station 1 RAM, the error is flagged in the Port 0 register bit that reflects Station 1 error status.</p> <p><b>Notes:</b> The bits in this register can be masked by their respective <a href="#">Device-Specific Error Mask 3</a> register bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 714h).</p> <p>All errors in this register generate ERR_FATAL or ERR_NONFATAL Messages, if enabled by the following:</p> <ul style="list-style-type: none"> <li>• <a href="#">Device-Specific Error Mask 3</a> register</li> <li>• <a href="#">Uncorrectable Error Status</a> register <a href="#">Uncorrectable Internal Error Status</a> bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset FB8h[22], is Set)</li> <li>• Port's <a href="#">Device Status</a> register <a href="#">Fatal Error Reporting Enable</a> and <a href="#">Non-Fatal Error Reporting Enable</a> bits (offset 70h[2:1], respectively)</li> </ul> <p>The Stations are defined in <a href="#">Table 13-5</a>. The Port 0 bits are for Station 0. The Port 8 bits are for Station 1. The Port 16 bits are for Station 2.</p>				
0	<b>Station 0 Packet RAM0 Instance 0 Soft Error Counter Overflow Detected</b> 0 = No error is detected 1 = Soft error is detected	RW1CS	Yes	0
1	<b>Station 1 Packet RAM0 Instance 0 Soft Error Counter Overflow Detected</b> 0 = No error is detected 1 = Soft error is detected	RW1CS	Yes	0
2	<b>Station 2 Packet RAM0 Instance 0 Soft Error Counter Overflow Detected</b> 0 = No error is detected 1 = Soft error is detected	RW1CS	Yes	0
5:3	<b>Factory Test Only</b>	RW1CS	Yes	000b
6	<b>Station 0 Packet RAM0 Instance 1 Soft Error Counter Overflow Detected</b> 0 = No error is detected 1 = Soft error is detected	RW1CS	Yes	0
7	<b>Station 1 Packet RAM0 Instance 1 Soft Error Counter Overflow Detected</b> 0 = No error is detected 1 = Soft error is detected	RW1CS	Yes	0
8	<b>Station 2 Packet RAM0 Instance 1 Soft Error Counter Overflow Detected</b> 0 = No error is detected 1 = Soft error is detected	RW1CS	Yes	0
11:9	<b>Factory Test Only</b>	RW1CS	Yes	000b

**Register 13-160. 710h Device-Specific Error Status 3**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Station 0 Packet RAM1 Instance 0 Soft Error Counter Overflow Detected</b> 0 = No error is detected 1 = Soft error is detected	RW1CS	Yes	0
13	<b>Station 1 Packet RAM1 Instance 0 Soft Error Counter Overflow Detected</b> 0 = No error is detected 1 = Soft error is detected	RW1CS	Yes	0
14	<b>Station 2 Packet RAM1 Instance 0 Soft Error Counter Overflow Detected</b> 0 = No error is detected 1 = Soft error is detected	RW1CS	Yes	0
17:15	<b>Factory Test Only</b>	RW1CS	Yes	000b
18	<b>Station 0 Packet RAM1 Instance 1 Soft Error Counter Overflow Detected</b> 0 = No error is detected 1 = Soft error is detected	RW1CS	Yes	0
19	<b>Station 1 Packet RAM1 Instance 1 Soft Error Counter Overflow Detected</b> 0 = No error is detected 1 = Soft error is detected	RW1CS	Yes	0
20	<b>Station 2 Packet RAM1 Instance 1 Soft Error Counter Overflow Detected</b> 0 = No error is detected 1 = Soft error is detected	RW1CS	Yes	0
23:21	<b>Factory Test Only</b>	RsvdP	No	000b
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 13-161. 714h Device-Specific Error Mask 3**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The bits in this register can be used to mask their respective <b>Device-Specific Error Status 3</b> register bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 710h).</p> <p>The Stations are defined in <a href="#">Table 13-5</a>. The Port 0 bits are for Station 0. The Port 8 bits are for Station 1. The Port 16 bits are for Station 2.</p>				
0	<b>Station 0 Packet RAM0 Instance 0 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Packet RAM0 Instance 0 Soft Error Counter Overflow Detected</i> bit is masked/disabled	RWS	Yes	1
1	<b>Station 1 Packet RAM0 Instance 0 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Packet RAM0 Instance 0 Soft Error Counter Overflow Detected</i> bit is masked/disabled	RWS	Yes	1
2	<b>Station 2 Packet RAM0 Instance 0 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Packet RAM0 Instance 0 Soft Error Counter Overflow Detected</i> bit is masked/disabled	RWS	Yes	1
5:3	<b>Factory Test Only</b>	RWS	Yes	000b
6	<b>Station 0 Packet RAM0 Instance 1 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Packet RAM0 Instance 1 Soft Error Counter Overflow Detected</i> bit is masked/disabled	RWS	Yes	1
7	<b>Station 1 Packet RAM0 Instance 1 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Packet RAM0 Instance 1 Soft Error Counter Overflow Detected</i> bit is masked/disabled	RWS	Yes	1
8	<b>Station 2 Packet RAM0 Instance 1 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Packet RAM0 Instance 1 Soft Error Counter Overflow Detected</i> bit is masked/disabled	RWS	Yes	1
11:9	<b>Factory Test Only</b>	RWS	Yes	000b

**Register 13-161. 714h Device-Specific Error Mask 3****(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Station 0 Packet RAM1 Instance 0 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Packet RAM1 Instance 0 Soft Error Counter Overflow Detected</i> bit is masked/disabled	RWS	Yes	1
13	<b>Station 1 Packet RAM1 Instance 0 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Packet RAM1 Instance 0 Soft Error Counter Overflow Detected</i> bit is masked/disabled	RWS	Yes	1
14	<b>Station 2 Packet RAM1 Instance 0 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Packet RAM1 Instance 0 Soft Error Counter Overflow Detected</i> bit is masked/disabled	RWS	Yes	1
17:15	<b>Factory Test Only</b>	RWS	Yes	000b
18	<b>Station 0 Packet RAM1 Instance 1 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Packet RAM1 Instance 1 Soft Error Counter Overflow Detected</i> bit is masked/disabled	RWS	Yes	1
19	<b>Station 1 Packet RAM1 Instance 1 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Packet RAM1 Instance 1 Soft Error Counter Overflow Detected</i> bit is masked/disabled	RWS	Yes	1
20	<b>Station 2 Packet RAM1 Instance 1 Soft Error Counter Overflow Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Packet RAM1 Instance 1 Soft Error Counter Overflow Detected</i> bit is masked/disabled	RWS	Yes	1
23:21	<b>Factory Test Only</b>	RsvdP	No	000b
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 13-162. 718h Device-Specific Error Status 4**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>ECC is checked by the egress Port, and any ECC error is reported by the egress Station (containing the egress Port), when that Port reads the Packet data from the ingress Station RAM. <i>For example</i>, when Port 8 receives a packet, the data is stored in Station 1 RAM; then, if the egress Port is in Station 0 and an ECC error is detected when the Port reads the Station 1 RAM, the error is flagged in the Port 0 register bit that reflects Station 1 error status.</p> <p><b>Notes:</b> The bits in this register can be masked by their respective <i>Device-Specific Error Mask 4</i> register bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 71Ch).</p> <p>All errors in this register generate ERR_FATAL or ERR_NONFATAL Messages, if enabled by the following:</p> <ul style="list-style-type: none"> <li>• <i>Device-Specific Error Mask 4</i> register</li> <li>• <i>Uncorrectable Error Status</i> register <i>Uncorrectable Internal Error Status</i> bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset FB8h[22], is Set)</li> <li>• Port's <i>Device Status</i> register <i>Fatal Error Reporting Enable</i> and <i>Non-Fatal Error Reporting Enable</i> bits (offset 70h[2:1], respectively)</li> </ul> <p>The Stations are defined in Table 13-5. The Port 0 bits are for Station 0. The Port 8 bits are for Station 1. The Port 16 bits are for Station 2.</p>				
0	<b>Station 0 Packet RAM0 Instance 0 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
1	<b>Station 1 Packet RAM0 Instance 0 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
2	<b>Station 2 Packet RAM0 Instance 0 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
5:3	<b>Factory Test Only</b>	RW1CS	Yes	000b
6	<b>Station 0 Packet RAM0 Instance 1 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
7	<b>Station 1 Packet RAM0 Instance 1 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
8	<b>Station 2 Packet RAM0 Instance 1 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
11:9	<b>Factory Test Only</b>	RW1CS	Yes	000b



**Register 13-162. 718h Device-Specific Error Status 4**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Station 0 Packet RAM1 Instance 0 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
13	<b>Station 1 Packet RAM1 Instance 0 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
14	<b>Station 2 Packet RAM1 Instance 0 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
17:15	<b>Factory Test Only</b>	RW1CS	Yes	000b
18	<b>Station 0 Packet RAM1 Instance 1 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
19	<b>Station 1 Packet RAM1 Instance 1 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
20	<b>Station 2 Packet RAM1 Instance 1 2-Bit ECC Error Detected</b> 0 = No error is detected 1 = 2-bit ECC error is detected	RW1CS	Yes	0
23:21	<b>Factory Test Only</b>	RsvdP	No	000b
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 13-163. 71Ch Device-Specific Error Mask 4**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The bits in this register can be used to mask their respective <b>Device-Specific Error Status 4</b> register bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 718h).</p> <p>The Stations are defined in <a href="#">Table 13-5</a>. The Port 0 bits are for Station 0. The Port 8 bits are for Station 1. The Port 16 bits are for Station 2.</p>				
0	<b>Station 0 Packet RAM0 Instance 0 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Packet RAM0 Instance 0 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
1	<b>Station 1 Packet RAM0 Instance 0 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Packet RAM0 Instance 0 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
2	<b>Station 2 Packet RAM0 Instance 0 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Packet RAM0 Instance 0 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
5:3	<b>Factory Test Only</b>	RWS	Yes	000b
6	<b>Station 0 Packet RAM0 Instance 1 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Packet RAM0 Instance 1 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
7	<b>Station 1 Packet RAM0 Instance 1 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Packet RAM0 Instance 1 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
8	<b>Station 2 Packet RAM0 Instance 1 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Packet RAM0 Instance 1 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
11:9	<b>Factory Test Only</b>	RWS	Yes	000b

**Register 13-163. 71Ch Device-Specific Error Mask 4****(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Station 0 Packet RAM1 Instance 0 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Packet RAM1 Instance 0 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
13	<b>Station 1 Packet RAM1 Instance 0 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Packet RAM1 Instance 0 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
14	<b>Station 2 Packet RAM1 Instance 0 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Packet RAM1 Instance 0 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
17:15	<b>Factory Test Only</b>	RWS	Yes	000b
18	<b>Station 0 Packet RAM1 Instance 1 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 0 Packet RAM1 Instance 1 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
19	<b>Station 1 Packet RAM1 Instance 1 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 1 Packet RAM1 Instance 1 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
20	<b>Station 2 Packet RAM1 Instance 1 2-Bit ECC Error Mask</b> 0 = No effect on reporting activity 1 = <i>Station 2 Packet RAM1 Instance 1 2-Bit ECC Error Detected</i> bit is masked/disabled	RWS	Yes	1
23:21	<b>Factory Test Only</b>	RsvdP	No	000b
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 13-164. 720h ECC Error Check Disable**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> PEX_INTA# is used only in Base mode. VSx_PEX_INTA# is used only in Virtual Switch mode.</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
0	<b>ECC 1-Bit Error Check Disable</b> 0 = RAM 1-Bit Soft Error Check enabled 1 = Disables RAM 1-Bit Soft Error Check	<b>Base Mode</b> 0  <b>Virtual Switch Mode</b> 0, accessible through the Management Port	RWS	Yes	0
1	<b>ECC 2-Bit Error Check Disable</b> 0 = RAM 2-Bit Soft Error Check enabled 1 = Disables RAM 2-Bit Soft Error Check	<b>Base Mode</b> 0  <b>Virtual Switch Mode</b> 0, accessible through the Management Port	RWS	Yes	0
2	<b>Software Force Error Enable</b> 1 = <b>Correctable Error Status</b> and <b>Uncorrectable Error Status</b> registers (offsets FC4h and FB8h, respectively) change from RW1CS to RW. Software Sets an <b>Uncorrectable Error Status</b> register bit when that register is Cleared, the <b>Advanced Error Capabilities and Control</b> register <i>First Error Pointer</i> field (offset FCCh[4:0]) is updated; however, the <b>Header Log x</b> registers (offsets FD0h through FDCh) remain unchanged.	Transparent Downstream	RWS	Yes	0
		Otherwise	RsvdP	No	0
3	<b>Software Force Non-Posted Request</b> Used to select software-forced errors to be associated with Posted or Non-Posted TLPs, because some errors are handled differently, depending upon the TLP type (Posted or Non-Posted).  0 = Handle software-forced errors as if the errors are associated with Posted TLPs 1 = Enables handling of errors associated with Posted TLPs as if those errors are associated with Non-Posted TLPs		RWS	Yes	0

**Register 13-164. 720h ECC Error Check Disable**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<p><b>Enable PEX_INTA# or VSx_PEX_INTA# Interrupt Output for Hot Plug or Link State Event-Triggered Interrupts</b></p> <p>0 = Hot Plug or Link State Event Interrupt Requests send an INTx Message (and do not assert <b>PEX_INTA#</b> (Base mode) nor <b>VSx_PEX_INTA#</b> (Virtual Switch mode))</p> <p>1 = Hot Plug or Link State Event Interrupt Requests assert <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode) (and do not send an INTx Message)</p>		RWS	Yes	0
5	<p><b>Enable PEX_INTA# or VSx_PEX_INTA# Interrupt Output for Device-Specific NT-Link Port Event-Triggered Interrupts</b></p> <p><i>This bit is valid only in NT mode.</i></p> <p>Enables <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode), or INTx interrupt, signaling for the following NT-Link Interface events, defined in the <b>Link Error Status Virtual</b> and <b>Link Error Mask Virtual</b> registers (NT Port Virtual Interface, offsets <b>FE0h</b> and <b>FE4h</b>, respectively):</p> <ul style="list-style-type: none"> <li>• NT-Link Port Correctable error (NT Port Link Interface, offset <b>FC4h</b>)</li> <li>• NT-Link Port Uncorrectable error (NT Port Link Interface, offset <b>FB8h</b>)</li> <li>• NT-Link Port Data Link Layer State change</li> <li>• NT-Link Port received (and consequently dropped) a Fatal or Non-Fatal Error Message</li> </ul> <p>0 = Device-Specific NT-Link Port Event Interrupt Requests send an INTx Message (and do not assert <b>PEX_INTA#</b> nor <b>VSx_PEX_INTA#</b>)</p> <p>1 = Device-Specific NT-Link Port Event Interrupt Requests assert <b>PEX_INTA#</b> or <b>VSx_PEX_INTA#</b> (and do not send an INTx Message)</p>	<p><b>Base Mode</b> 0</p> <p><b>Virtual Switch Mode</b> 0, accessible through the Management Port</p>	RWS	Yes	0
	<b>Reserved</b>	Otherwise	RsvdP	No	0

**Register 13-164. 720h ECC Error Check Disable**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (*Cont.*)

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
6	<b>Enable PEX_INTA# or VSx_PEX_INTA# Interrupt Output for GPIO-Generated Interrupts</b> 0 = General-Purpose Input/Output (GPIO) Interrupt Requests send an INT <sub>x</sub> Message (and do not assert PEX_INTA# (Base mode) nor VS <sub>x</sub> _PEX_INTA# (Virtual Switch mode)) 1 = GPIO Interrupt Requests assert PEX_INTA# (Base mode) or VS <sub>x</sub> _PEX_INTA# (Virtual Switch mode) (and do not send an INT <sub>x</sub> Message)		RWS	Yes	0
7	<b>Enable PEX_INTA# or VSx_PEX_INTA# Interrupt Output for NT-Virtual Doorbell-Generated Interrupts</b> <i>This bit is valid only in NT mode.</i> Enables PEX_INTA# (Base mode) or VS <sub>x</sub> _PEX_INTA# (Virtual Switch mode), or INT <sub>x</sub> Messages, for NT-Virtual Doorbell interrupts (NT Port Virtual Interface, offsets C4Ch through C58h). 0 = NT-Virtual Doorbell Interrupt Requests send an INT <sub>x</sub> Message (and do not assert PEX_INTA# nor VS <sub>x</sub> _PEX_INTA#) 1 = NT-Virtual Doorbell Interrupt Requests assert PEX_INTA# or VS <sub>x</sub> _PEX_INTA# (and do not send an INT <sub>x</sub> Message)	<b>Base Mode</b> 0  <b>Virtual Switch Mode</b> 0, accessible through the Management Port	RWS	Yes	0
	<i>Reserved</i>	Otherwise	RsvdP	No	0

**Register 13-164. 720h ECC Error Check Disable**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Base Mode</b> <i>Reserved</i>		RsvdP	No	0
	<b>Virtual Switch Mode</b> <b>Enable VS<sub>x</sub>_PEX_INTA# Interrupt Output for Management Port Doorbell-Generated Interrupts</b> 0 = Management Port Doorbell Interrupt Requests send an INT <sub>x</sub> Message (and do not assert VS <sub>x</sub> _PEX_INTA#) 1 = Management Port Doorbell Interrupt Requests assert VS <sub>x</sub> _PEX_INTA# (and do not send an INT <sub>x</sub> Message)	0, accessible through the Management Port	RWS	Yes	0
9	<b>Base Mode</b> <i>Reserved</i>		RsvdP	No	0
	<b>Virtual Switch Mode</b> <b>Enable VS<sub>x</sub>_PEX_INTA# Interrupt Output for Management Link Status Event-Generated Interrupts</b> 0 = Management Link Status Event Interrupt Requests send an INT <sub>x</sub> Message (and do not assert VS <sub>x</sub> _PEX_INTA#) 1 = Management Link Status Event Interrupt Requests assert VS <sub>x</sub> _PEX_INTA# (and do not send an INT <sub>x</sub> Message)	0, accessible through the Management Port	RWS	Yes	0
10	<b>Disable Sending MSI if MSI Is Enabled after Interrupt Status Set</b> 0 = Does not disable sending an MSI, if MSIs are enabled after an <i>Interrupt Status</i> bit is Set 1 = Disables sending an MSI, if MSIs are enabled after an <i>Interrupt Status</i> bit is Set <i>Note: This bit must remain Cleared, for compliance to specifications governing the MSI Capability.</i>		RWS	Yes	0
11	<b>Egress Completion FIFO Overflow Mask</b> <i>Note: This bit can be used to mask bit 12 (Egress Completion FIFO Overflow Status).</i>		RWS	Yes	1
12	<b>Egress Completion FIFO Overflow Status</b> <i>Note: This bit can be masked by bit 11 (Egress Completion FIFO Overflow Mask).</i>		RW1CS	Yes	0
31:13	<i>Reserved</i>		RsvdP	No	0-0h

**Register 13-165. 724h Gen 3 Framing Error Status  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The bits in this register can be masked by their respective <b>Gen 3 Framing Error Mask</b> register bits (offset 728h).				
0	<b>Invalid Sync Header Status</b>	RW1CS	Yes	0
1	<b>Sync Header Not Same on All Lanes Status</b>	RW1CS	Yes	0
2	<b>Bad Framing Symbol in Data Block Status</b>	RW1CS	Yes	0
3	<b>Second Symbol of DLLP Not ACh Status</b> 1 = Second symbol of the Data Link Layer Packet (DLLP) is not ACh	RW1CS	Yes	0
4	<b>EDB Not following TLP Status</b>	RW1CS	Yes	0
5	<b>Length CRC/Parity Error Status</b> Length Cyclic Redundancy Check (CRC)/Parity Error status.	RW1CS	Yes	0
6	<b>End of Data Stream Not Followed by Ordered-Set Block Status</b>	RW1CS	Yes	0
7	<b>End of Data Stream Not in Last DWord of Data Block Status</b>	RW1CS	Yes	0
8	<b>Ordered-Set Block Not Followed by Data Block Status</b>	RW1CS	Yes	0
9	<b>FTS Not Followed by Data Block Status</b>	RW1CS	Yes	0
10	<b>Invalid Skip Ordered-Set Status</b>	RW1CS	Yes	0
11	<b>Ordered-Set Not the Same on All Lanes Status</b>	RW1CS	Yes	0
12	<b>Not All Subsequent Lanes Have LIDL or End of Data Stream when LIDL Status</b>	RW1CS	Yes	0
13	<b>Last TLP or DLLP Has No LIDL Symbols Status</b> For x8 or x16 Gen 3 Links, if a TLP or DLLP ends on Lane K, and is not followed by another TLP or DLLP, then subsequent Lanes should have LIDL symbols.	RW1CS	Yes	0
14	<b>Frame Error 14 Status</b>	RW1CS	Yes	0
15	<b>Frame Error 15 Status</b>	RW1CS	Yes	0
31:16	<b>Reserved</b>	RsvdP	No	0000h



**Register 13-166. 728h Gen 3 Framing Error Mask  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The bits in this register can be used to mask their respective <b>Gen 3 Framing Error Status</b> register bits (offset 724h).				
0	Invalid Sync Header Mask	RWS	Yes	0
1	Sync Header Not Same on All Lanes Mask	RWS	Yes	0
2	Bad Framing Symbol in Data Block Mask	RWS	Yes	0
3	Second Symbol of DLLP Not ACh Mask	RWS	Yes	0
4	EDB Not following TLP Mask	RWS	Yes	0
5	Length CRC/Parity Error Mask Length Cyclic Redundancy Check (CRC)/Parity Error mask.	RWS	Yes	0
6	End of Data Stream Not Followed by Ordered-Set Block Mask	RWS	Yes	0
7	End of Data Stream Not in Last DWord of Data Block Mask	RWS	Yes	0
8	Ordered-Set Block Not Followed by Data Block Mask	RWS	Yes	0
9	FTS Not Followed by Data Block Mask	RWS	Yes	0
10	Invalid Skip Ordered-Set Mask	RWS	Yes	0
11	Ordered-Set Not the Same on All Lanes Mask	RWS	Yes	0
12	Not All Subsequent Lanes Have LIDL or End of Data Stream when LIDL Mask	RWS	Yes	0
13	Last TLP or DLLP Has No LIDL Symbols Mask For x8 or x16 Gen 3 Links, if TLP or DLLP ends on Lane K, and not followed by another TLP or DLLP, then subsequent Lanes should have LIDL symbols.	RWS	Yes	0
14	Frame Error 14 Mask	RWS	Yes	0
15	Frame Error 15 Mask	RWS	Yes	0
31:16	Reserved	RsvdP	No	0000h

**Register 13-167. 72Ch Gen 3 Framing Error Disable**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
0	Sync Header Not Same on All Lanes Disable	RWS	Yes	0
1	Invalid Block Type Disable	RWS	Yes	0
2	Not SKIP, EIOS, or EIEOS Ordered-Set in Block following EDS Disable	RWS	Yes	0
3	Any Ordered-Set Block following SDS Ordered-Set Disable	RWS	Yes	0
4	Ordered-Set Block Received Without EDS Token in Preceding Data Block Disable	RWS	Yes	0
5	Block Containing EDS Token Followed by Data Block Instead of Ordered-Set Disable	RWS	Yes	0
6	Ordered-Sets Not the Same on All Lanes Disable	RWS	Yes	1
7	TLP Length Field CRC/Parity Disable	RWS	Yes	0
8	All Four Symbols of EDB Token Not the Same Disable	RWS	Yes	0
9	EDB Token Not Immediately following a TLP Disable	RWS	Yes	0
10	TLP Length Field = 0 Disable	RWS	Yes	1
11	Received More than One STP Token in a Single Symbol Time Disable	RWS	Yes	1
12	Received More than One SDP Token in a Single Symbol Time Disable	RWS	Yes	1
13	Lanes between IDL and Next Potential Starting Lane are Not All IDL Disable	RWS	Yes	1
14	Invalid Framing Token Detected when Searching for Framing Tokens Disable	RWS	Yes	0
15	Frame Error 15 Disable	RWS	Yes	1
31:16	Reserved	RsvdP	No	0000h

### 13.16.12 Device-Specific Registers – Control (Offsets 760h – 774h)

This section details the Device-Specific Control registers. [Table 13-29](#) defines the register map.

**Table 13-29. Device-Specific Control Register Map**  
(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
Station-Based Control		760h
Ingress Chip Control		764h
Factory Test Only		768h – 774h

**Register 13-168. 760h Station-Based Control**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>				
0	<b>Link List RAM 1-Bit Error Injection</b> The ECC Counter for this bit is located in the <b>Ingress PLL RAM ECC 1-Bit Counter</b> register <i>1-Bit ECC Counter for PLL RAM Read from Ingress Block</i> field (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 778h[7:0]).	RWS	Yes	0
1	<b>Link List RAM 2-Bit Error Injection</b>	RWS	Yes	0
2	<b>Link List RAM Error Injection Field</b> 0 = Error injection is in the <i>ECC Code</i> field 1 = Error injection is in the <i>Data</i> field	RWS	Yes	0
4:3	<b>Link List RAM Port Selector for Error Injection</b>	RWS	Yes	00b
5	<b>Accumulator x12 Pointer Reset Disable</b> 0 = Enables 1 = Disables	RWS	Yes	0
6	<i>Not used</i>	RWS	Yes	0
7	<b>RAM Select for ECC Injection</b> 0 = Select Payload RAM for ECC injection 0 = Select Header RAM for ECC injection	RWS	Yes	0
8	<b>Header RAM 1-Bit Error Injection</b>	RWS	Yes	0
9	<b>Header RAM 2-Bit Error Injection</b>	RWS	Yes	0
10	<b>Header RAM Error Injection Field</b> 0 = Error injection is in the <i>ECC Code</i> field 1 = Error injection is in the <i>Data</i> field	RWS	Yes	0
12:11	<b>Header RAM Port Selector for Error Injection</b>	RWS	Yes	00b
15:13	<i>Factory Test Only</i>	RWS	Yes	000b

**Register 13-168. 760h Station-Based Control**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (Cont.)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Silicon Revision BA</b> <b>Payload RAM 1-Bit Error Injection</b>	RWS	Yes	0
	<b>Silicon Revision CA</b> <b>Device-Specific Register Access by Extended Configuration Request Enable</b>  <i>Note: Although this bit exists within each Station; however, it is used only in the Station(s) that have an Upstream or NT Port.</i>  This bit controls whether Extended Configuration Requests can access Device-Specific registers within the following offset ranges, of each Port: <ul style="list-style-type: none"> <li>• 200h through AFCh</li> <li>• B0Ch through B6Ch</li> <li>• B80h through C30h</li> </ul> Because this bit exists within the Device-Specific register range listed above, it cannot be Set by an Extended Configuration Request. Serial EEPROM, I <sup>2</sup> C/SMBus, and Memory-Mapped access can initially Set this bit. For NT Port Configuration access: <ul style="list-style-type: none"> <li>• <b>NT Port Virtual Interface access</b> – Set this bit in the Station(s) that have an Upstream Port</li> <li>• <b>NT Port Link Interface access</b> – Set this bit in the Station that has the NT Port</li> </ul> 0 = Extended Configuration Requests cannot access Device-Specific registers (default) 1 = Extended Configuration Requests can access Device-Specific registers	RWS	Yes	0
20	<b>Source Queue Link List RAM 1-Bit Error Injection</b>	RWS	Yes	0
21	<b>Source Queue Link List RAM 2-Bit Error Injection</b>	RWS	Yes	0
22	<b>Source Queue Link List RAM Error Injection Field</b> 0 = Error injection is in the <i>ECC Code</i> field 1 = Error injection is in the <i>Data</i> field	RWS	Yes	0
23	<b>Source Queue Link List RAM Selector for Error Injection</b> 0 = Port-A, normal traffic 1 = Port-B, Read-Pacing traffic	RWS	Yes	0

**Register 13-168. 760h Station-Based Control**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (*Cont.*)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Disable Credit Re-Balancing</b> 1 = No Credit re-balancing	RWS	Yes	0
25	<b>Use Serial EEPROM Values for Ingress Credit Initialization</b> Allow Configuration with a Device Number that is not 0, that is accessing Downstream devices, to be forwarded. When the Device Number is 0, the Configuration terminates in a UR.  0 = Use default values for ingress credit initialization 1 = Use serial EEPROM values for ingress credit initialization	RWS	Yes	0
26	<b>INCH Credit Reserve Flag</b> A transition from 0 to 1 indicates that I <sup>2</sup> C has finished with all CSR to INCH Initialization registers, and credit reservation can proceed.	RWS	Yes	0
28:27	<b>Factory Test Only</b>	RWS	Yes	00b
29	<b>No Special Treatment for Relaxed Ordering Traffic</b> The PEX 8748 supports Relaxed Ordering for Completions. By default, if the RO attribute is Set within a Completion, that Completion can bypass Posted transactions, if Posted TLPs are blocked at the egress Port (due to insufficient Posted credits from the connected device). This behavior can be disabled by Setting this bit, in each Station.  1 = Device-Specific Relaxed Ordering Completion will not be flagged to the egress block  <i>Note: When Set, affects all Ports within the Station.</i>	RWS	Yes	0
30	<b>Factory Test Only</b>	RWS	Yes	0
31	<b>Not used</b>	RWS	Yes	0

**Register 13-169. 764h Ingress Chip Control**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>				
0	<b>Expansion ROM Virtual Side</b> 0 = Expansion ROM is located on the NT Port Link Interface 1 = Expansion ROM is located on the NT Port Virtual Interface	RWS	Yes	0
1	<b>Reserved</b>	RsvdP	No	0
2	<b>LUT Format</b> Use of this bit is described in Section 15.3.2, “Device-Specific Memory-Mapped Configuration Mechanism.”	RWS	Yes	0
16:3	<b>Factory Test Only</b>	RWS	Yes	0-0h
17	<b>NT Error Message Drop</b> 0 = If the NT Port Link Interface receives an Uncorrectable Error Message that is routed to the Root Complex, the NT Port Link Interface reports a Malformed TLP error. 1 = Do not malformed a Fatal Error Message received on the NT Port Link Interface with routing equal to 0, and instead, drop the packet and log the error into the <a href="#">Link Error Status Virtual</a> register <a href="#">Link Side Uncorrectable Error Message Drop Status</a> bit (NT Port Virtual Interface, offset FE0h[3]). If the corresponding <a href="#">Link Error Mask Virtual</a> register <a href="#">Link Side Uncorrectable Error Message Drop Mask</a> bit (NT Port Virtual Interface, offset FE4h[3]) is Set, the NT Port Virtual Interface signals an Interrupt (INTx, MSI, or PEX_INTA# (Base mode) or VSx_PEX_INTA# (Virtual Switch mode)) to the Virtual side Host through the Upstream Port(s), if interrupts are enabled.	RWS	Yes	0
18	<b>Reserved</b>	RsvdP	No	0
19	<b>Not used</b>	RWS	Yes	0
20	<b>Ingress MWr32 Counter Disable</b> 0 = Enables Ingress Memory Write 32-Bit Counter 1 = Disables Ingress Memory Write 32-Bit Counter	RWS	Yes	0
21	<b>Ingress MWr64 Counter Disable</b> 0 = Enables Ingress Memory Write 64-Bit Counter 1 = Disables Ingress Memory Write 64-Bit Counter	RWS	Yes	0
22	<b>Ingress MSG Counter Disable</b> 0 = Enables Ingress Message Counter 1 = Disables Ingress Message Counter	RWS	Yes	0
23	<b>Ingress MRd32 Counter Disable</b> 0 = Enables Ingress Memory Read 32-Bit Counter 1 = Disables Ingress Memory Read 32-Bit Counter	RWS	Yes	0

**Register 13-169. 764h Ingress Chip Control**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,  
accessible through the Management Port) (*Cont.*)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Ingress MRd64 Counter Disable</b> 0 = Enables Ingress Memory Read 64-Bit Counter 1 = Disables Ingress Memory Read 64-Bit Counter	RWS	Yes	0
25	<b>Ingress Other Non-Posted Counter Disable</b> 0 = Enables Ingress Other Non-Posted Counter 1 = Disables Ingress Other Non-Posted Counter	RWS	Yes	0
26	<b>Ingress and Egress DLLP ACK Counter Disable</b> 0 = Enables Ingress and Egress DLLP ACK Counter 1 = Disables Ingress and Egress DLLP ACK Counter	RWS	Yes	0
27	<b>Ingress and Egress DLLP UpdateFC-P Counter Disable</b> 0 = Enables Ingress and Egress DLLP UpdateFC-Posted Counter 1 = Disables Ingress and Egress DLLP UpdateFC-Posted Counter	RWS	Yes	0
28	<b>Ingress and Egress DLLP UpdateFC-NP Counter Disable</b> 0 = Enables Ingress and Egress DLLP UpdateFC-Non-Posted Counter 1 = Disables Ingress and Egress DLLP UpdateFC-Non-Posted Counter	RWS	Yes	0
29	<b>Ingress and Egress DLLP UpdateFC-CPL Counter Disable</b> 0 = Enables Ingress and Egress DLLP UpdateFC-Completion Counter 1 = Disables Ingress and Egress DLLP UpdateFC-Completion Counter	RWS	Yes	0
30	<b>Parity Error Inject on Sch Bus</b>	RWS	Yes	0
31	<i>Factory Test Only</i>	RO	Yes	0



### 13.16.13 Device-Specific Registers – Soft Error (Offsets 778h – 8FCh)

This section details the Device-Specific Soft Error registers. [Table 13-30](#) defines the register map.

**Table 13-30. Device-Specific Soft Error Register Map**  
(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Reserved																Ingress PLL RAM ECC 1-Bit Counter												778h				
Accumulator Debug NAK																																77Ch
Ingress Exerciser Completion Status																																780h
Ingress Exerciser Completion Data																																784h
Reserved																														788h –		7FCh
Egress Station 0 Payload RAM Soft Error Counters																																800h
Egress Station 1 Payload RAM Soft Error Counters																																804h
Egress Station 2 Payload RAM Soft Error Counters																																808h
Factory Test Only																																80Ch
Reserved				Egress Header RAM Soft Error Counters 1																												810h
Reserved				Egress Header RAM Soft Error Counters 2																												814h
Reserved				Egress Header RAM Soft Error Counters 3																												818h
Reserved				Egress Header RAM Soft Error Counters 4																												81Ch
Reserved				Egress PLL RAM Soft Error Counters 1																												820h
Egress PLL RAM Soft Error Counters 2																Reserved																824h
Egress Miscellaneous RAM Soft Error Counters																																828h
Soft Error Injection																																82Ch
Reserved																														830h –		8FCh

**Register 13-170. 778h Ingress PLL RAM ECC 1-Bit Counter**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Notes:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports. The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.				
7:0	<b>1-Bit ECC Counter for PLL RAM Read from Ingress Block</b> A Write of 0 to bit 0 Clears the ECC Counter.	ROS	No	00h
31:8	<b>Reserved</b>	RsvdP	No	0000_00h

**Register 13-171. 77Ch Accumulator Debug NAK**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
<i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i>					
<i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i>					
18:0	Accumulator Debug NAK		RWS	Yes	0-0h
	Bit(s)	Description/Function			
	0	Initiate NAK Substitution for ACK			
	3:1	Port Number for NAK Substitution 000b = Port 0 of the Station 001b = Port 1 of the Station 010b = Port 2 of the Station 011b = Port 3 of the Station  All other encodings are <i>Reserved</i> .  <i>Note: If the Port Number indicated is not enabled, the corresponding encoding is <b>Reserved</b>.</i>			
	15:4	Sequence Number for NAK Substitution			
	18:16	Number of NAK Substitutions to Perform			
	31:19	Not used			

**Register 13-172. 780h Ingress Exerciser Completion Status**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
0	<b>Received a Completion</b> 1 = New Completion received	RW1C	Yes	0
1	<b>Completion Received with EP</b>	ROS	No	0
2	<b>Completion Received with ECRC Error</b>	ROS	No	0
3	<b>Reserved</b>	RsvdP	No	0
6:4	<b>Completion Status</b> 1 = Received Completion TLPs status	ROS	No	000b
31:7	<b>Reserved</b>	RsvdP	No	0-0h

**Register 13-173. 784h Ingress Exerciser Completion Data**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
31:0	<b>Completion Data</b> 1 = Received Completion data	ROS	No	0000_0000h

**Register 13-174. 800h Egress Station 0 Payload RAM Soft Error Counters**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Notes:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports. The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.				
7:0	Station 0 Payload RAM Bus 0 Instance 0 1-Bit Soft Error Counter Value	ROS	No	00h
15:8	Station 0 Payload RAM Bus 0 Instance 1 1-Bit Soft Error Counter Value	ROS	No	00h
23:16	Station 0 Payload RAM Bus 1 Instance 0 1-Bit Soft Error Counter Value	ROS	No	00h
31:24	Station 0 Payload RAM Bus 1 Instance 1 1-Bit Soft Error Counter Value	ROS	No	00h

**Register 13-175. 804h Egress Station 1 Payload RAM Soft Error Counters**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Notes:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports. The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.				
7:0	Station 1 Payload RAM Bus 0 Instance 0 1-Bit Soft Error Counter Value	ROS	No	00h
15:8	Station 1 Payload RAM Bus 0 Instance 1 1-Bit Soft Error Counter Value	ROS	No	00h
23:16	Station 1 Payload RAM Bus 1 Instance 0 1-Bit Soft Error Counter Value	ROS	No	00h
31:24	Station 1 Payload RAM Bus 1 Instance 1 1-Bit Soft Error Counter Value	ROS	No	00h

**Register 13-176. 808h Egress Station 2 Payload RAM Soft Error Counters**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Notes:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports. The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.				
7:0	Station 2 Payload RAM Bus 0 Instance 0 1-Bit Soft Error Counter Value	ROS	No	00h
15:8	Station 2 Payload RAM Bus 0 Instance 1 1-Bit Soft Error Counter Value	ROS	No	00h
23:16	Station 2 Payload RAM Bus 1 Instance 0 1-Bit Soft Error Counter Value	ROS	No	00h
31:24	Station 2 Payload RAM Bus 1 Instance 1 1-Bit Soft Error Counter Value	ROS	No	00h

**Register 13-177. 810h Egress Header RAM Soft Error Counters 1**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
7:0	Station 0 Header RAM Bus 0 Instance 0 1-Bit Soft Error Counter Value	ROS	No	00h
15:8	Station 1 Header RAM Bus 0 Instance 0 1-Bit Soft Error Counter Value	ROS	No	00h
23:16	Station 2 Header RAM Bus 0 Instance 0 1-Bit Soft Error Counter Value	ROS	No	00h
31:24	Reserved	RsvdP	No	00h

**Register 13-178. 814h Egress Header RAM Soft Error Counters 2**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
7:0	Station 0 Header RAM Bus 0 Instance 1 1-Bit Soft Error Counter Value	ROS	No	00h
15:8	Station 1 Header RAM Bus 0 Instance 1 1-Bit Soft Error Counter Value	ROS	No	00h
23:16	Station 2 Header RAM Bus 0 Instance 1 1-Bit Soft Error Counter Value	ROS	No	00h
31:24	Reserved	RsvdP	No	00h

**Register 13-179. 818h Egress Header RAM Soft Error Counters 3**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
7:0	Station 0 Header RAM Bus 1 Instance 0 1-Bit Soft Error Counter Value	ROS	No	00h
15:8	Station 1 Header RAM Bus 1 Instance 0 1-Bit Soft Error Counter Value	ROS	No	00h
23:16	Station 2 Header RAM Bus 1 Instance 0 1-Bit Soft Error Counter Value	ROS	No	00h
31:24	Reserved	RsvdP	No	00h

**Register 13-180. 81Ch Egress Header RAM Soft Error Counters 4**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
7:0	Station 0 Header RAM Bus 1 Instance 1 1-Bit Soft Error Counter Value	ROS	No	00h
15:8	Station 1 Header RAM Bus 1 Instance 1 1-Bit Soft Error Counter Value	ROS	No	00h
23:16	Station 2 Header RAM Bus 1 Instance 1 1-Bit Soft Error Counter Value	ROS	No	00h
31:24	Reserved	RsvdP	No	00h

**Register 13-181. 820h Egress PLL RAM Soft Error Counters 1**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
7:0	Station 0 PLL RAM 1-Bit Soft Error Counter Value	ROS	No	00h
15:8	Station 1 PLL RAM 1-Bit Soft Error Counter Value	ROS	No	00h
23:16	Station 2 PLL RAM 1-Bit Soft Error Counter Value	ROS	No	00h
31:24	Reserved	RsvdP	No	00h

**Register 13-182. 824h Egress PLL RAM Soft Error Counters 2**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
7:0	Source Queue Link List RAM Instance 1 1-Bit Soft Error Counter Value Reserved	RsvdP	No	00h
15:8	Reserved	RsvdP	No	00h
23:16	Destination Queue Link List RAM Instance 1 1-Bit Soft Error Counter Value	ROS	No	00h
31:24	Destination Queue Link List RAM Instance 2 1-Bit Soft Error Counter Value	ROS	No	00h

**Register 13-183. 828h Egress Miscellaneous RAM Soft Error Counters**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i><b>Notes:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
7:0	Destination Queue Data RAM 1-Bit Soft Error Counter Value	ROS	No	00h
15:8	Destination Queue Link List RAM Instance 0 1-Bit Soft Error Counter Value	ROS	No	00h
23:16	Source Queue Link List RAM 1-Bit Soft Error Counter Value	ROS	No	00h
31:24	Retry Buffer 1-Bit Soft Error Counter Value	ROS	No	00h



**Register 13-184. 82Ch Soft Error Injection**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
0	<b>Destination Queue Data RAM 1-Bit Soft Error Injection</b> Writing 1 injects one error.	RWS	Yes	0
1	<b>Destination Queue Data RAM 2-Bit Soft Error Injection</b> Writing 1 injects one error.	RWS	Yes	0
2	<b>Destination Queue Data RAM Error Injection Select</b> 0 = Inject Soft error in the <i>ECC Code</i> field 1 = Inject Soft error in the <i>Data</i> field	RWS	Yes	0
3	<b>Destination Queue Link List RAM Instance 0 1-Bit Soft Error Injection</b> Writing 1 injects one error.	RWS	Yes	0
4	<b>Destination Queue Link List RAM Instance 0 2-Bit Soft Error Injection</b> Writing 1 injects one error.	RWS	Yes	0
5	<b>Destination Queue Link List RAM Instance 0 Error Injection Select</b> 0 = Inject Soft error in the <i>ECC Code</i> field 1 = Inject Soft error in the <i>Data</i> field	RWS	Yes	0
6	<b>Retry Buffer 1-Bit Soft Error Injection</b> Writing 1 injects one error.	RWS	Yes	0
7	<b>Retry Buffer 2-Bit Soft Error Injection</b> Writing 1 injects one error.	RWS	Yes	0

**Register 13-184. 82Ch Soft Error Injection**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Retry Buffer Error Injection Select</b> 0 = Inject Soft error in the <i>ECC Code</i> field 1 = Inject Soft error in the <i>Data</i> field	RWS	Yes	0
9	<b>Destination Queue Link List RAM Instance 1 1-Bit Soft Error Injection</b> Writing 1 injects one error.	RWS	Yes	0
10	<b>Destination Queue Link List RAM Instance 1 2-Bit Soft Error Injection</b> Writing 1 injects one error.	RWS	Yes	0
11	<b>Destination Queue Link List RAM Instance 1 Error Injection Select</b> 0 = Inject Soft error in the <i>ECC Code</i> field 1 = Inject Soft error in the <i>Data</i> field	RWS	Yes	0
12	<b>Destination Queue Link List RAM Instance 2 1-Bit Soft Error Injection</b> Writing 1 injects one error.	RWS	Yes	0
13	<b>Destination Queue Link List RAM Instance 2 2-Bit Soft Error Injection</b> Writing 1 injects one error.	RWS	Yes	0
14	<b>Destination Queue Link List RAM Instance 2 Error Injection Select</b> 0 = Inject Soft error in the <i>ECC Code</i> field 1 = Inject Soft error in the <i>Data</i> field	RWS	Yes	0
19:15	<i>Reserved</i>	RsvdP	No	0-0h

**Register 13-184. 82Ch Soft Error Injection**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (*Cont.*)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Ingress Payload RAM0 Instance 0 ECC Counter Reset</b> Writing 1 Clears the Counter in the <b>Egress Station x Payload RAM Soft Error Counters</b> register <i>Station x Payload RAM0 Instance 0 1-Bit Soft Error Counter Value</i> field(s) (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offsets 800h, 804h, and 808h[7:0]). Reads always return a value of 0.	RZ	Yes	0
21	<b>Ingress Payload RAM0 Instance 1 ECC Counter Reset</b> Writing 1 Clears the Counter in the <b>Egress Station x Payload RAM Soft Error Counters</b> register <i>Station x Payload RAM0 Instance 1 1-Bit Soft Error Counter Value</i> field(s) (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offsets 800h, 804h, and 808h[15:8]). Reads always return a value of 0.	RZ	Yes	0
22	<b>Ingress Payload RAM1 Instance 0 ECC Counter Reset</b> Writing 1 Clears the Counter in the <b>Egress Station x Payload RAM Soft Error Counters</b> register <i>Station x Payload RAM1 Instance 0 1-Bit Soft Error Counter Value</i> field(s) (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offsets 800h, 804h, and 808h[23:16]). Reads always return a value of 0.	RZ	Yes	0
23	<b>Ingress Payload RAM1 Instance 1 ECC Counter Reset</b> Writing 1 Clears the Counter in the <b>Egress Station x Payload RAM Soft Error Counters</b> register <i>Station x Payload RAM1 Instance 1 1-Bit Soft Error Counter Value</i> field(s) (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offsets 800h, 804h, and 808h[31:24]). Reads always return a value of 0.	RZ	Yes	0
24	<b>Header RAM ECC Counter Reset</b> Writing 1 Clears the Counter in the <b>Egress Header RAM Soft Error Counters x</b> register <i>Station x Header RAM 1-Bit Soft Error Counter Value</i> field(s) (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 810h[7:0, 15:8, and 23:16], respectively). Reads always return a value of 0.	RZ	Yes	0
28:25	<b>Factory Test Only</b>	RZ	Yes	0h
31:29	<b>Station Number to Reset the ECC Counter</b> Selects the ECC Counter that corresponds to the Station Number. 000b = Station 0 (default) 001b = Station 1 010b = Station 2 All other encodings are <i>Reserved</i> .	RZ	Yes	000b

### 13.16.14 Device-Specific Registers – Virtual Switch (Offsets 900h – 9E8h), Virtual Switch Mode

**Note:** In Base mode, this entire structure is **Reserved**, RsvdP, not serial EEPROM nor I<sup>2</sup>C writable, and has a default value of 0h.

This section details the Device-Specific Virtual Switch Support registers located at offsets 900h through 9E8h. These registers are implemented only in Virtual Switch mode. Additionally, the registers are implemented only in Port 0, accessible through the Management Port (offsets 900h through 90Ch), or VS Upstream Port(s) and Management Port (offsets 910h through 93Ch), as indicated in the registers that follow. Table 13-31 defines the register map.

Other Device-Specific Virtual Switch registers are detailed in [Section 13.20, “Device-Specific Registers – Virtual Switch \(Offsets F00h – F20h\), Virtual Switch Mode.”](#)

**Table 13-31. Device-Specific Virtual Switch Register Map (Offsets 900h – 9E8h)**

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0	
Switch Link Up				900h
Switch Link Down				904h
Switch Link Event Mask				908h
Switch Link Status				90Ch
VS Upstream to Management Upstream Doorbell Request				910h
VS Upstream to Management Upstream Doorbell Mask				914h
VS Upstream to Management Upstream Scratchpad 1				918h
VS Upstream to Management Upstream Scratchpad 2				91Ch
VS Upstream to Management Upstream Scratchpad 3				920h
VS Upstream to Management Upstream Scratchpad 4				924h
Management Upstream to VS Upstream Doorbell Request				928h
Management Upstream to VS Upstream Doorbell Mask				92Ch
Management Upstream to VS Upstream Scratchpad 1				930h
Management Upstream to VS Upstream Scratchpad 2				934h
Management Upstream to VS Upstream Scratchpad 3				938h
Management Upstream to VS Upstream Scratchpad 4				93Ch
<i>Reserved</i>				940h – 9E8h

**Register 13-185. 900h Switch Link Up**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Port	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</i>					
0	<b>Port 0 Link Up</b> 1 = Port 0 Link transitioned from the <i>DL_Inactive</i> state to the <i>DL_Active</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
1	<b>Port 1 Link Up</b> 1 = Port 1 Link transitioned from the <i>DL_Inactive</i> state to the <i>DL_Active</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
2	<b>Port 2 Link Up</b> 1 = Port 2 Link transitioned from the <i>DL_Inactive</i> state to the <i>DL_Active</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
3	<b>Port 3 Link Up</b> 1 = Port 3 Link transitioned from the <i>DL_Inactive</i> state to the <i>DL_Active</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
5:4	<i>Factory Test Only</i>		RsvdP	No	00b
7:6	<i>Reserved</i>		RsvdP	No	00b
8	<b>Port 8 Link Up</b> 1 = Port 8 Link transitioned from the <i>DL_Inactive</i> state to the <i>DL_Active</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
9	<b>Port 9 Link Up</b> 1 = Port 9 Link transitioned from the <i>DL_Inactive</i> state to the <i>DL_Active</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
10	<b>Port 10 Link Up</b> 1 = Port 10 Link transitioned from the <i>DL_Inactive</i> state to the <i>DL_Active</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
11	<b>Port 11 Link Up</b> 1 = Port 11 Link transitioned from the <i>DL_Inactive</i> state to the <i>DL_Active</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
13:12	<i>Factory Test Only</i>		RsvdP	No	00b
15:14	<i>Reserved</i>		RsvdP	No	00b

**Register 13-185. 900h Switch Link Up**  
**(Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Port	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Port 16 Link Up</b> 1 = Port 16 Link transitioned from the <i>DL_Inactive</i> state to the <i>DL_Active</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<b>Reserved</b>	Otherwise	RsvdP	No	0
17	<b>Port 17 Link Up</b> 1 = Port 17 Link transitioned from the <i>DL_Inactive</i> state to the <i>DL_Active</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<b>Reserved</b>	Otherwise	RsvdP	No	0
18	<b>Port 18 Link Up</b> 1 = Port 18 Link transitioned from the <i>DL_Inactive</i> state to the <i>DL_Active</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<b>Reserved</b>	Otherwise	RsvdP	No	0
19	<b>Port 19 Link Up</b> 1 = Port 19 Link transitioned from the <i>DL_Inactive</i> state to the <i>DL_Active</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<b>Reserved</b>	Otherwise	RsvdP	No	0
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
31:22	<b>Reserved</b>		RsvdP	No	0-0h

**Register 13-186. 904h Switch Link Down**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Port	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.					
0	<b>Port 0 Link Down</b> 1 = Port 0 Link transitioned from the <i>DL_Active</i> state to the <i>DL_Inactive</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
1	<b>Port 1 Link Down</b> 1 = Port 1 Link transitioned from the <i>DL_Active</i> state to the <i>DL_Inactive</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
2	<b>Port 2 Link Down</b> 1 = Port 2 Link transitioned from the <i>DL_Active</i> state to the <i>DL_Inactive</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
3	<b>Port 3 Link Down</b> 1 = Port 3 Link transitioned from the <i>DL_Active</i> state to the <i>DL_Inactive</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
5:4	<i>Factory Test Only</i>		RsvdP	No	00b
7:6	<i>Reserved</i>		RsvdP	No	00b
8	<b>Port 8 Link Down</b> 1 = Port 8 Link transitioned from the <i>DL_Active</i> state to the <i>DL_Inactive</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
9	<b>Port 9 Link Down</b> 1 = Port 9 Link transitioned from the <i>DL_Active</i> state to the <i>DL_Inactive</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
10	<b>Port 10 Link Down</b> 1 = Port 10 Link transitioned from the <i>DL_Active</i> state to the <i>DL_Inactive</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
11	<b>Port 11 Link Down</b> 1 = Port 11 Link transitioned from the <i>DL_Active</i> state to the <i>DL_Inactive</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
13:12	<i>Factory Test Only</i>		RsvdP	No	00b
15:14	<i>Reserved</i>		RsvdP	No	00b

**Register 13-186. 904h Switch Link Down**  
**(Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Port	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Port 16 Link Down</b> 1 = Port 16 Link transitioned from the <i>DL_Active</i> state to the <i>DL_Inactive</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<b>Reserved</b>	Otherwise	RsvdP	No	0
17	<b>Port 17 Link Down</b> 1 = Port 17 Link transitioned from the <i>DL_Active</i> state to the <i>DL_Inactive</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<b>Reserved</b>	Otherwise	RsvdP	No	0
18	<b>Port 18 Link Down</b> 1 = Port 18 Link transitioned from the <i>DL_Active</i> state to the <i>DL_Inactive</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<b>Reserved</b>	Otherwise	RsvdP	No	0
19	<b>Port 19 Link Down</b> 1 = Port 19 Link transitioned from the <i>DL_Active</i> state to the <i>DL_Inactive</i> state	0, accessible through the Management Port	RW1C	Yes	PCFG
	<b>Reserved</b>	Otherwise	RsvdP	No	0
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
31:22	<b>Reserved</b>		RsvdP	No	0-0h



**Register 13-187. 908h Switch Link Event Mask**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Port	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes:</i> The bits in this register can be used to mask their respective <b>Switch Link Status</b> register bits (Port 0, accessible through the Management Port, offset 90Ch).</p> <p><i>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</i></p>					
0	<b>Port 0 Link Event Mask</b> 0 = Interrupt for Port 0 due to a Link Up or Link Down event is not masked 1 = Interrupt for Port 0 due to a Link Up or Link Down event is masked	0, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	0
1	<b>Port 1 Link Event Mask</b> 0 = Interrupt for Port 1 due to a Link Up or Link Down event is not masked 1 = Interrupt for Port 1 due to a Link Up or Link Down event is masked	0, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	0
2	<b>Port 2 Link Event Mask</b> 0 = Interrupt for Port 2 due to a Link Up or Link Down event is not masked 1 = Interrupt for Port 2 due to a Link Up or Link Down event is masked	0, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	0
3	<b>Port 3 Link Event Mask</b> 0 = Interrupt for Port 3 due to a Link Up or Link Down event is not masked 1 = Interrupt for Port 3 due to a Link Up or Link Down event is masked	0, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	0
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b

**Register 13-187. 908h Switch Link Event Mask**  
**(Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Port	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Port 8 Link Event Mask</b> 0 = Interrupt for Port 8 due to a Link Up or Link Down event is not masked 1 = Interrupt for Port 8 due to a Link Up or Link Down event is masked	0, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	0
9	<b>Port 9 Link Event Mask</b> 0 = Interrupt for Port 9 due to a Link Up or Link Down event is not masked 1 = Interrupt for Port 9 due to a Link Up or Link Down event is masked	0, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	0
10	<b>Port 10 Link Event Mask</b> 0 = Interrupt for Port 10 due to a Link Up or Link Down event is not masked 1 = Interrupt for Port 10 due to a Link Up or Link Down event is masked	0, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	0
11	<b>Port 11 Link Event Mask</b> 0 = Interrupt for Port 11 due to a Link Up or Link Down event is not masked 1 = Interrupt for Port 11 due to a Link Up or Link Down event is masked	0, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	0
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b

**Register 13-187. 908h Switch Link Event Mask**  
**(Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Port	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Port 16 Link Event Mask</b> 0 = Interrupt for Port 16 due to a Link Up or Link Down event is not masked 1 = Interrupt for Port 16 due to a Link Up or Link Down event is masked	0, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	0
17	<b>Port 17 Link Event Mask</b> 0 = Interrupt for Port 17 due to a Link Up or Link Down event is not masked 1 = Interrupt for Port 17 due to a Link Up or Link Down event is masked	0, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	0
18	<b>Port 18 Link Event Mask</b> 0 = Interrupt for Port 18 due to a Link Up or Link Down event is not masked 1 = Interrupt for Port 18 due to a Link Up or Link Down event is masked	0, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	0
19	<b>Port 19 Link Event Mask</b> 0 = Interrupt for Port 19 due to a Link Up or Link Down event is not masked 1 = 1 = Interrupt for Port 19 due to a Link Up or Link Down event is masked	0, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	0
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
31:22	<b>Reserved</b>		RsvdP	No	0-0h

**Register 13-188. 90Ch Switch Link Status**  
**(Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Port	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Notes:</b> The bits in this register can be masked by their respective <i>Switch Link Event Mask</i> register bits (Port 0, accessible through the Management Port, offset 908h). <i>Table 13-5</i> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.					
0	<b>Port 0 Link Status</b> 0 = Indicates that Port 0 is in a <i>DL_Inactive</i> state 1 = Indicates that Port 0 is in a <i>DL_Active</i> state	0, accessible through the Management Port	RO	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
1	<b>Port 1 Link Status</b> 0 = Indicates that Port 1 is in a <i>DL_Inactive</i> state 1 = Indicates that Port 1 is in a <i>DL_Active</i> state	0, accessible through the Management Port	RO	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
2	<b>Port 2 Link Status</b> 0 = Indicates that Port 2 is in a <i>DL_Inactive</i> state 1 = Indicates that Port 2 is in a <i>DL_Active</i> state	0, accessible through the Management Port	RO	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
3	<b>Port 3 Link Status</b> 0 = Indicates that Port 3 is in a <i>DL_Inactive</i> state 1 = Indicates that Port 3 is in a <i>DL_Active</i> state	0, accessible through the Management Port	RO	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
5:4	<i>Factory Test Only</i>		RsvdP	No	00b
7:6	<i>Reserved</i>		RsvdP	No	00b
8	<b>Port 8 Link Status</b> 0 = Indicates that Port 8 is in a <i>DL_Inactive</i> state 1 = Indicates that Port 8 is in a <i>DL_Active</i> state	0, accessible through the Management Port	RO	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
9	<b>Port 9 Link Status</b> 0 = Indicates that Port 9 is in a <i>DL_Inactive</i> state 1 = Indicates that Port 9 is in a <i>DL_Active</i> state	0, accessible through the Management Port	RO	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
10	<b>Port 10 Link Status</b> 0 = Indicates that Port 10 is in a <i>DL_Inactive</i> state 1 = Indicates that Port 10 is in a <i>DL_Active</i> state	0, accessible through the Management Port	RO	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
11	<b>Port 11 Link Status</b> 0 = Indicates that Port 11 is in a <i>DL_Inactive</i> state 1 = Indicates that Port 11 is in a <i>DL_Active</i> state	0, accessible through the Management Port	RO	Yes	PCFG
	<i>Reserved</i>	Otherwise	RsvdP	No	0
13:12	<i>Factory Test Only</i>		RsvdP	No	00b
15:14	<i>Reserved</i>		RsvdP	No	00b

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

**Register 13-188. 90Ch Switch Link Status****(Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Port	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Port 16 Link Status</b> 0 = Indicates that Port 16 is in a <i>DL_Inactive</i> state 1 = Indicates that Port 16 is in a <i>DL_Active</i> state	0, accessible through the Management Port	RO	Yes	PCFG
	<b>Reserved</b>	Otherwise	RsvdP	No	0
17	<b>Port 17 Link Status</b> 0 = Indicates that Port 17 is in a <i>DL_Inactive</i> state 1 = Indicates that Port 17 is in a <i>DL_Active</i> state	0, accessible through the Management Port	RO	Yes	PCFG
	<b>Reserved</b>	Otherwise	RsvdP	No	0
18	<b>Port 18 Link Status</b> 0 = Indicates that Port 18 is in a <i>DL_Inactive</i> state 1 = Indicates that Port 18 is in a <i>DL_Active</i> state	0, accessible through the Management Port	RO	Yes	PCFG
	<b>Reserved</b>	Otherwise	RsvdP	No	0
19	<b>Port 19 Link Status</b> 0 = Indicates that Port 19 is in a <i>DL_Inactive</i> state 1 = Indicates that Port 19 is in a <i>DL_Active</i> state	0, accessible through the Management Port	RO	Yes	PCFG
	<b>Reserved</b>	Otherwise	RsvdP	No	0
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
31:22	<b>Reserved</b>		RsvdP	No	0-0h

**Register 13-189. 910h VS Upstream to Management Upstream Doorbell Request  
(Virtual Switch mode – VS Upstream Port(s) and Management Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The bits in this register can be masked by their respective <b>VS Upstream to Management Upstream Doorbell Mask</b> register bits (VS Upstream Port(s) and Management Port, offset 914h).					
3:0	<b>Doorbell</b> Writing 1 to any bit within this field, in the Non-Management VS Upstream Port(s), signals an interrupt to the Management Upstream Port. The Management Port Clears the interrupt(s), by writing 1 to the bit(s) that are Set.	Non-Management VS Upstream Port	RW	Yes	0h
		Management Port	RW1C	Yes	0h
	<b>Reserved</b>	Otherwise	RsvdP	No	0h
31:4	<b>Reserved</b>		RsvdP	No	0000_000h

**Register 13-190. 914h VS Upstream to Management Upstream Doorbell Mask  
(Virtual Switch mode – VS Upstream Port(s) and Management Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note:</i> The bits in this register can be used to mask their respective <b>VS Upstream to Management Upstream Doorbell Request</b> register bits (VS Upstream Port(s) and Management Port, offset 910h).					
3:0	<b>Doorbell Interrupt Mask</b> 0 = Doorbell interrupts to the Management Upstream Port are not masked 1 = Doorbell interrupts to the Management Upstream Port are masked	Non-Management VS Upstream Port	RW	Yes	Fh
		Management Port	RW1C	Yes	0h
	<b>Reserved</b>	Otherwise	RsvdP	No	0h
31:4	<b>Reserved</b>		RsvdP	No	0000_000h

**Register 13-191. 918h VS Upstream to Management Upstream Scratchpad 1  
(Virtual Switch mode – VS Upstream Port(s) and Management Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 1</b> 32-bit Scratchpad 1 register.	Non-Management VS Upstream Port	RW	Yes	0000_0000h
		Management Port	ROS	Yes	0000_0000h
	<i>Reserved</i>	Otherwise	RsvdP	No	0000_0000h

**Register 13-192. 91Ch VS Upstream to Management Upstream Scratchpad 2  
(Virtual Switch mode – VS Upstream Port(s) and Management Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 2</b> 32-bit Scratchpad 2 register.	Non-Management VS Upstream Port	RW	Yes	0000_0000h
		Management Port	ROS	Yes	0000_0000h
	<i>Reserved</i>	Otherwise	RsvdP	No	0000_0000h

**Register 13-193. 920h VS Upstream to Management Upstream Scratchpad 3  
(Virtual Switch mode – VS Upstream Port(s) and Management Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 3</b> 32-bit Scratchpad 3 register.	Non-Management VS Upstream Port	RW	Yes	0000_0000h
		Management Port	ROS	Yes	0000_0000h
	<i>Reserved</i>	Otherwise	RsvdP	No	0000_0000h

**Register 13-194. 924h VS Upstream to Management Upstream Scratchpad 4  
(Virtual Switch mode – VS Upstream Port(s) and Management Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 4</b> 32-bit Scratchpad 4 register.	Non-Management VS Upstream Port	RW	Yes	0000_0000h
		Management Port	ROS	Yes	0000_0000h
	<i>Reserved</i>	Otherwise	RsvdP	No	0000_0000h

**Register 13-195. 928h Management Upstream to VS Upstream Doorbell Request**  
**(Virtual Switch mode – VS Upstream Port(s) and Management Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i><b>Note:</b> The bits in this register can be masked by their respective <a href="#">Management Upstream to VS Upstream Doorbell Mask</a> register bits (VS Upstream Port(s) and Management Port, offset 92Ch).</i>					
3:0	<b>Doorbell</b> Writing 1 to any bit within this field, in the Management Port, signals an interrupt to the VS Upstream Port(s). The Non-Management VS Upstream Port(s) Clear(s) the interrupt(s), by writing 1 to the bit(s) that are Set.	Non-Management VS Upstream Port	RW1C	Yes	0h
		Management Port	RW	Yes	0h
	<b>Reserved</b>	Otherwise	RsvdP	No	0h
31:4	<b>Reserved</b>		RsvdP	No	0000_000h

**Register 13-196. 92Ch Management Upstream to VS Upstream Doorbell Mask**  
**(Virtual Switch mode – VS Upstream Port(s) and Management Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i><b>Note:</b> The bits in this register can be used to mask their respective <a href="#">Management Upstream to VS Upstream Doorbell Request</a> register bits (VS Upstream Port(s) and Management Port, offset 928h).</i>					
3:0	<b>Doorbell Interrupt Mask</b> 0 = Doorbell interrupts to the VS Upstream Port(s) are not masked 1 = Doorbell interrupts to the VS Upstream Port(s) are masked	Non-Management VS Upstream Port	RW1C	Yes	Fh
		Management Port	RW	Yes	0h
	<b>Reserved</b>	Otherwise	RsvdP	No	0h
31:4	<b>Reserved</b>		RsvdP	No	0000_000h



**Register 13-197. 930h Management Upstream to VS Upstream Scratchpad 1  
(Virtual Switch mode – VS Upstream Port(s) and Management Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 1</b> 32-bit Scratchpad 1 register.	Non-Management VS Upstream Port	ROS	Yes	0000_0000h
		Management Port	RW	Yes	0000_0000h
	<i>Reserved</i>	Otherwise	RsvdP	No	0000_0000h

**Register 13-198. 934h Management Upstream to VS Upstream Scratchpad 2  
(Virtual Switch mode – VS Upstream Port(s) and Management Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 2</b> 32-bit Scratchpad 2 register.	Non-Management VS Upstream Port	ROS	Yes	0000_0000h
		Management Port	RW	Yes	0000_0000h
	<i>Reserved</i>	Otherwise	RsvdP	No	0000_0000h

**Register 13-199. 938h Management Upstream to VS Upstream Scratchpad 3  
(Virtual Switch mode – VS Upstream Port(s) and Management Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 3</b> 32-bit Scratchpad 3 register.	Non-Management VS Upstream Port	ROS	Yes	0000_0000h
		Management Port	RW	Yes	0000_0000h
	<i>Reserved</i>	Otherwise	RsvdP	No	0000_0000h

**Register 13-200. 93Ch Management Upstream to VS Upstream Scratchpad 4  
(Virtual Switch mode – VS Upstream Port(s) and Management Port)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 4</b> 32-bit Scratchpad 4 register.	Non-Management VS Upstream Port	ROS	Yes	0000_0000h
		Management Port	RW	Yes	0000_0000h
	<i>Reserved</i>	Otherwise	RsvdP	No	0000_0000h

13.16.15 Device-Specific Registers – Ingress Credit Handler (Offsets 9ECh – A2Ch)

This section details the Device-Specific Ingress Credit Handler (INCH) registers. Table 13-32 defines the register map.

Table 13-32. Device-Specific INCH Register Map (Ports<sup>a</sup>)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Reserved																																9ECh	
INCH Station Pool Values																																9F0h	
Factory Test Only																																9F4h	
INCH Reserve Pool																																9F8h	
Reserved																								INCH Port Pool								9FCh	
INCH Threshold VC0 Posted																																A00h	
Factory Test Only								INCH Threshold VC0 Non-Posted																								A04h	
INCH Threshold VC0 Completion																																A08h	
Reserved																																A0Ch –	A2Ch

a. Certain registers are Port-specific, others are Station-specific; all are Device-specific.

**Register 13-201. 9F0h INCH Station Pool Values**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
INCH Reserve pool.  <i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i>  <i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i>				
8:0	<b>Current Value of Header Pool for the Station</b>	ROS	Yes	–
15:9	<i>Reserved</i>	RsvdP	No	0-0h
25:16	<b>Current Value of Payload Link Pool for the Station</b>	ROS	Yes	–
31:26	<i>Reserved</i>	RsvdP	No	0-0h

**Register 13-202. 9F8h INCH Reserve Pool**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i>  <i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i>				
7:0	<b>Header Count to Remove from Station Header Pool</b> Value of Header to remove from initial Header pool.	RW	Yes	00h
15:8	<i>Reserved</i>	RsvdP	No	00h
24:16	<b>Payload Link Count to Remove from Station Payload Pool</b> Value of Payload Links to remove from initial Payload Link pool.	RW	Yes	0-0h
31:25	<i>Reserved</i>	RsvdP	No	0-0h

**Register 13-203. 9FCh INCH Port Pool**

(Base mode – All Ports; Virtual Switch mode – Port 0, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note: Consider the INCH Port Pool register to be Reserved and only change the credit Settings, using the INCH Threshold registers (offsets <b>A00h</b> through <b>A08h</b>). Do not change the INCH Port Pool register from its default value, unless directed otherwise by PLX Technical Support.</i>				
2:0	<b>Port Payload Pool</b> Payload credits (other than the initial credits) for Posted/Completion TLPs that are dedicated to the Port.  000b = 0 001b = 32 010b = 64 011b = 96 100b = 128 101b = 192 110b, 111b = 256	RWS	Yes	000b
3	<b>Unused 0</b> Keep value at 0. Additional bit for the Port Payload Pool.	RWS	Yes	0
6:4	<b>Port Header Pool</b> Combined Header credits (other than the initial credits) that are dedicated to the Port.  000b = 0 TLP 001b = 4 TLPs 010b = 8 TLPs 011b = 16 TLPs 100b = 32 TLPs 101b = 48 TLPs 110b, 111b = 64 TLPs	RWS	Yes	000b
7	<b>Unused 1</b> Keep value at 0. Additional bit for the Port Header Pool.	RWS	Yes	0
31:8	<b>Reserved</b>	RsvdP	No	0000_00h

**Register 13-204. A00h INCH Threshold VC0 Posted**  
**(Base mode – All Ports; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
Posted credits are used for VC0 Memory Write and Message transactions.				
<i>Note: Changing credit values from default register values must be done carefully; otherwise the PEX 8748 will not properly function.</i>				
2:0	<b>Reserved</b>	RsvdP	No	
8:3	<b>Posted Payload Credit</b> Default advertised Posted Payload credit. Actual value is dependent upon Link width and Port configuration, and the advertised value can be 0 to 3 more Payload credits than the register value. Bit resolution is in units of 8. Each increment provides 8 Posted Payload credits ( <i>for example</i> , Ah = 80 Posted Payload credits). Each credit means that 16 bytes of storage are reserved for Posted TLP Payload data.	RWS	Yes	Upstream Payload (decimal): x16: 504 x8: 256 x4: 136 x2: 72  Downstream Payload (decimal): x16: 504 x8: 256 x4: 136 x2: 72  Upstream: x16: C3F8h x8: 63000h x4: 3288h x2: 1A48h
15:9	<b>Posted Header Credit</b> Default advertised Posted Header credit. Actual value is dependent upon Link width and Port configuration. Bit resolution is 1 for 1. Each increment provides 1 Posted Header credit ( <i>for example</i> , Ah = 10 Posted Header credits). Each credit means that storage is reserved for the entire Header of a Posted TLP.	RWS	Yes	Upstream Header (decimal): x16: 97 x8: 49 x4: 25 x2: 13  Downstream Header (decimal): x16: 124 x8: 113 x4: 57 x2: 29  Downstream: x16: F9F8h x8: E300h x4: 7288h x2: 3A48h

**Register 13-204. A00h INCH Threshold VC0 Posted**  
**(Base mode – All Ports; Virtual Switch mode – Port 0, accessible through the**  
**Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
17:16	<b>UpdateFC High-Priority Threshold for Posted Payload Credit</b> 00b = 75% (default) 01b = 50% 10b = 25% 11b = 100%	RWS	Yes	00b
19:18	<b>UpdateFC High-Priority Threshold for Posted Header Credit</b> 00b = 75% (default) 01b = 50% 10b = 25% 11b = 100%	RWS	Yes	00b
22:20	<b>Congested Port Weight</b> If the effective rate Setting times the negotiated Port Link width equates to less than x1 or greater than x8, the internal Credit Allocation logic rounds to x1 or x8, respectively.  000b = Request is weighted, based upon the Port's Link width relative to the effective Link widths of the other Stations' Ports  001b = Increases the weight of a Request by 2x 010b = Increases the weight of a Request by 4x 011b = Increases the weight of a Request by 8x  100b = Port receives no credit out of the common pool, until a decongested state is reached  101b = Decreases the weight of a Request by 2x 110b = Decreases the weight of a Request by 4x 111b = Decreases the weight of a Request by 8x	RWS	Yes	000b
31:23	<b>Reserved</b>	RsvdP	No	0-0h

**Register 13-205. A04h INCH Threshold VC0 Non-Posted**  
**(Base mode – All Ports; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
Non-Posted credits are used for VC0 Memory Read, I/O Read, I/O Write, Configuration Read, and Configuration Write transactions. <i>Note: Changing credit values from default register values must be done carefully; otherwise the PEX 8748 will not properly function.</i>				
7:0	<b>Non-Posted Payload Credit</b> The Non-Posted Payload is stored with the Non-Posted Header; therefore Non-Posted Payload credit is always available.	RWS	Yes	Upstream Payload (decimal): x16: 8 x8: 8 x4: 8 x2: 8  Downstream Payload (decimal): x16: 8 x8: 8 x4: 8 x2: 8  Upstream: x16: 6108h x8: 3108h x4: 1908h x2: 0D08h
8	<b>Reserved</b>	RsvdP	No	0
15:9	<b>Non-Posted Header Credit</b> Default advertised Posted Header credit. Actual value is dependent upon Link width and Port configuration. Bit resolution is 1 for 1. Each increment provides 1 Non-Posted Header credit ( <i>for example</i> , Ah = 10 Non-Posted Header credits). Each credit means that storage is reserved for the entire Header of a Non-Posted TLP.	RWS	Yes	Upstream Header (decimal): x16: 97 x8: 49 x4: 25 x2: 13  Downstream Header (decimal): x16: 124 x8: 62 x4: 31 x2: 15  Downstream: x16: 7C08h x8: 3E08h x4: 1F08h x2: 0F08h

**Register 13-205. A04h INCH Threshold VC0 Non-Posted**  
**(Base mode – All Ports; Virtual Switch mode – Port 0, accessible through the**  
**Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
17:16	<b>UpdateFC High-Priority Threshold for Non-Posted Payload Credit</b> 00b = 75% (default) 01b = 50% 10b = 25% 11b = 100%	RWS	Yes	00b
19:18	<b>UpdateFC High-Priority Threshold for Non-Posted Header Credit</b> 00b = 75% (default) 01b = 50% 10b = 25% 11b = 100%	RWS	Yes	00b
22:20	<i>Not used</i>	RWS	Yes	000b
23	<i>Reserved</i>	RsvdP	No	0
29:24	<i>Factory Test Only</i>	RWS	Yes	0-0h
31:30	<i>Factory Test Only</i>	RW1C	No	00b



**Register 13-206. A08h INCH Threshold VC0 Completion**  
**(Base mode – All Ports; Virtual Switch mode – Port 0, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
Completion credits are used for VC0 Memory Read, I/O Read, I/O Write, Configuration Read, and Configuration Write transaction Completions.				
<i>Note: Changing credit values from default register values must be done carefully; otherwise the PEX 8748 will not properly function.</i>				
2:0	<b>Reserved</b>	RsvdP	No	Upstream Payload (decimal): x16: 480 x8: 240 x4: 136 x2: 72
8:3	<b>Completion Payload Credit</b> Default advertised Completion Payload credit. Actual value is dependent upon Link width and Port configuration, and the advertised value can be 0 to 3 more Payload credits than the register value. Bit resolution is in units of 8. Each increment provides 8 Completion Payload credits ( <i>for example</i> , Ah = 80 Completion Payload credits). Each credit means that 16 bytes of storage are reserved for Completion TLP Payload data.	RWS	Yes	Downstream Payload (decimal): x16: 480 x8: 240 x4: 136 x2: 72  Upstream: x16: F9E0h x8: E2F0h x4: 7288h x2: 3A48h
15:9	<b>Completion Header Credit</b> Default advertised Completion Header credit. Actual value is dependent upon Link width and Port configuration. Bit resolution is 1 for 1. Each increment provides 1 Completion Header credit ( <i>for example</i> , Ah = 10 Completion Header credits). Each credit means that storage is reserved for the entire Header of a Completion TLP.	RWS	Yes	Upstream Header (decimal): x16: 124 x8: 71 x4: 57 x2: 29  Downstream Header (decimal): x16: 70 x8: 36 x4: 19 x2: 11  Downstream: x16: 8DE0h x8: 48F0h x4: 2688h x2: 1648h

**Register 13-206. A08h INCH Threshold VC0 Completion**  
**(Base mode – All Ports; Virtual Switch mode – Port 0, accessible through the Management Port) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
17:16	<b>UpdateFC High-Priority Threshold for Completion Payload Credit</b> 00b = 75% (default) 01b = 50% 10b = 25% 11b = 100%	RWS	Yes	00b
19:18	<b>UpdateFC High-Priority Threshold for Completion Header Credit</b> 00b = 75% (default) 01b = 50% 10b = 25% 11b = 100%	RWS	Yes	00b
22:20	<i>Not used</i>	RWS	Yes	000b
31:23	<i>Reserved</i>	RsvdP	No	0-0h

### 13.16.16 Device-Specific Registers – Virtual Switch Debug and GPIO Status and Control (Offsets A30h – A74h)

This section details the Device-Specific Virtual Switch Debug and GPIO Status and Control registers located at offsets A30h through A74h. These registers are implemented only in the Upstream Port(s). [Table 13-33](#) defines the register map.

**Table 13-33. Device-Specific Virtual Switch Debug and GPIO Status and Control Register Map (Offsets A30h – A74h) (Upstream Port(s))**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
Virtual Switch Debug		A30h
<b>Reserved</b> (Base Mode)		A34h
Virtual Switch GPIO_PG 0_9 Direction Control (Virtual Switch Mode)		
<b>Reserved</b> (Base Mode)		A38h
Virtual Switch GPIO_PG 10_11 Direction Control (Virtual Switch Mode)		
<b>Reserved</b> (Base Mode)		A3Ch
Virtual Switch GPIO_PG 0_11 Availability (Virtual Switch Mode)		
<b>Reserved</b> (Base Mode)		A40h
Virtual Switch GPIO_PG 0_11 Input De-Bounce (Virtual Switch Mode)		
<b>Reserved</b> (Base Mode)		A44h
Virtual Switch GPIO_PG 0_11 Input Data (Virtual Switch Mode)		
<b>Reserved</b> (Base Mode)		A48h
Virtual Switch GPIO_PG 0_11 Output Data (Virtual Switch Mode)		
<b>Reserved</b> (Base Mode)		A4Ch
Virtual Switch GPIO_PG 0_11 Interrupt Polarity (Virtual Switch Mode)		
<b>Reserved</b> (Base Mode)		A50h
Virtual Switch GPIO_PG 0_11 Interrupt Status (Virtual Switch Mode)		
<b>Reserved</b> (Base Mode)		A54h
Virtual Switch GPIO_PG 0_11 Interrupt Mask (Virtual Switch Mode)		
<b>Reserved</b>		A58h – A74h

**Register 13-207. A30h Virtual Switch Debug  
(Upstream Port(s))**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<p><b>Interrupt Fencing Mode Select</b></p> <p><i>Note:</i> A Fundamental Reset is needed to recover from Fencing errors.</p> <p><b>Mode 1 (Default)</b></p> <ol style="list-style-type: none"> <li>When the PEX 8748 receives a packet with a Fatal error (Malformed, DLL Protocol error) from an external device, or the device detects a Credit Overflow, Receiver Overflow, or Surprise Link Down, the switch logs the Header on the corresponding Port, sends a Fatal Error Message to the Host, then asserts <b>FATAL_ERR#</b> (Base mode) or <b>VSx_FATAL_ERR#</b> (Virtual Switch mode).</li> <li>When the PEX 8748 detects an internal Fatal error (ECC failure), the switch sends a Fatal Interrupt Message to the Host and asserts <b>FATAL_ERR#</b> (Base mode) or <b>VSx_FATAL_ERR#</b> (Virtual Switch mode). In certain situations, delivery of the interrupt is not guaranteed; however, the signal is always asserted upon a Fatal event.</li> </ol> <p><b>Mode 2 (Generate Internal Reset) – Base Mode</b></p> <p>Upon Fatal error (internal or external) detection, an internal Chip Level reset is asserted (equivalent to an In-Band Reset from the Upstream Port). No Error Messages are generated, and no attempt is made to block packets in transit.</p> <p><b>Mode 3 (Block All Packet Transmission)</b></p> <p>Upon Fatal error (internal or external) detection, the Port logs the error in the <b>Uncorrectable Error Status</b> register (offset <b>FB8h</b>), then asserts <b>FATAL_ERR#</b> (Base mode) or <b>VSx_FATAL_ERR#</b> (Virtual Switch mode). This Fatal error detection blocks all the Ports from sending out TLPs. No Error Messages are generated. If a packet is already in transmission, an EDB is inserted to cancel the packet.</p> <p><b>Mode 4 (Block All Packet Transmission and Create Surprise Down)</b></p> <p>In addition to the Mode 3 actions, the PEX 8748 forces the Upstream Link to go down, thus causing a Surprise Down event on the Link, so that the Host is notified.</p> <p>00b = Mode 1 (default)  01b = Mode 2 – Generate Internal Reset  10b = Mode 3 – Block All Packet Transmission  11b = Mode 4 – Block All Packet Transmission and Create Surprise Down</p>	RWS	Yes	00b

**Register 13-207. A30h Virtual Switch Debug (Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
2	<b>Upstream Hot Reset Control</b> 0 = Reset all logic, except Sticky bits and Device-Specific registers 1 = Reset only the Configuration Space registers of all Ports defined by the <i>PCI Express Base r3.0</i> <i>Note: Only a Fundamental Reset serial EEPROM load affects this bit.</i>	RWS	Yes	0
3	<b>Disable Serial EEPROM Load on Hot Reset</b> 0 = Enables serial EEPROM load upon VS Upstream Port Hot Reset or <i>DL_Down</i> state. Port-specific registers for other virtual switches are <i>not</i> reloaded. (default) <b>Virtual Switch mode</b> – Chip- and Station-specific register reload from serial EEPROM (which can affect other virtual switches) can be enabled by Setting the <b>Management Port Control</b> register <i>Active Management Port EEPROM Load on Hot Reset for Chip and Station Registers Enable</i> bit (Port 0, accessible through the Management Port and Redundant Management Port, offset 354h[6]). 1 = Disables serial EEPROM load upon VS Upstream Port Hot Reset or <i>DL_Down</i> state	RWS	Yes	0
4	<b>Upstream Port and NT-Link Port DL_Down Reset Propagation Disable</b> Setting this bit: <ul style="list-style-type: none"> <li>Enables the Upstream and NT-Link Ports to ignore a Hot Reset training sequence,</li> <li>Blocks the PEX 8748 from manifesting an internal reset due to a <i>DL_Down</i> event</li> <li>Blocks the PEX 8748 NT Port Link Interface from manifesting an internal reset due to a <i>DL_Down</i> event on the NT Port Link, and</li> <li>Prevents the Downstream Ports from issuing a Hot Reset to Downstream devices when a Hot Reset or <i>DL_Down</i> event occurs on the Upstream Link</li> </ul>	RWS	Yes	0
5	<b>Factory Test Only</b>	RWS	Yes	0
23:6	<b>Reserved</b>	RsvdP	No	0-0h

**Register 13-207. A30h Virtual Switch Debug  
(Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Virtual Interface Access Enable</b> <i>This bit is used only in NT mode.</i>  When the serial EEPROM is not present, the default value is 1; otherwise, the default value is 0.  0 = Retries Type 0 Configuration TLP received on the NT Port Virtual Interface 1 = Accepts Type 0 Configuration TLP on the NT Port Virtual Interface  <i>Notes: This bit does not affect the PEX 8748 in Transparent mode,            nor does it affect other transaction types.</i>  <i>Set this bit to enable Configuration access to the NT Port            Virtual Interface.</i>	RW	Yes	1
25	<b>Link Interface Access Enable</b> <i>This bit is used only in NT mode.</i>  0 = Retries Type 0 Configuration Request received on the NT Port Link Interface 1 = Accepts Type 0 Configuration Request on the NT Port Link Interface  <i>Notes: This bit does not affect the PEX 8748 in Transparent mode.</i>  <i>Set this bit to enable Configuration access to the NT Port            Link Interface.</i>	RW	Yes	0

**Register 13-207. A30h Virtual Switch Debug (Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
26	<p><b>Inhibit EEPROM NT-Link Load on Hot Reset</b>  <i>This bit is used only in NT mode.</i></p> <p>Inhibits serial EEPROM load of NT Port Link Interface registers when any one of the following conditions are met:</p> <ul style="list-style-type: none"> <li>• <b>Upstream Port Hot Reset</b> – Bits [3:2] (<i>Disable Serial EEPROM Load on Hot Reset</i> and <i>Upstream Hot Reset Control</i>, respectively) are Cleared</li> <li>• <b>Upstream Port DL_Down state</b> – Bits [4:2] (<i>Upstream Port and NT-Link Port DL_Down Reset Propagation Disable</i>, <i>Disable Serial EEPROM Load on Hot Reset</i>, and <i>Upstream Hot Reset Control</i>, respectively) are Cleared</li> <li>• <b>NT Port Link Interface Hot Reset or DL_Down state</b> – Bit 3 (<i>Disable Serial EEPROM Load on Hot Reset</i>) is Cleared</li> </ul> <p>Refer also to <a href="#">Section 6.8, “Serial EEPROM Loading of NT Port Link Interface Registers – NT Mode,”</a> for further details.</p>	RW	Yes	0
27	<p><b>Load Only EEPROM NT-Link on Hot Reset</b>  <i>This bit is used only in NT mode.</i></p> <p>Load only serial EEPROM NT Port Link Interface register entries when any one of the following conditions are met:</p> <ul style="list-style-type: none"> <li>• <b>Upstream Port Hot Reset</b> – Bits [3:2] (<i>Disable Serial EEPROM Load on Hot Reset</i> and <i>Upstream Hot Reset Control</i>, respectively) are Cleared</li> <li>• <b>Upstream Port DL_Down state</b> – Bits [4:2] (<i>Upstream Port and NT-Link Port DL_Down Reset Propagation Disable</i>, <i>Disable Serial EEPROM Load on Hot Reset</i>, and <i>Upstream Hot Reset Control</i>, respectively) are Cleared</li> <li>• <b>NT Port Link Interface Hot Reset or DL_Down state</b> – Bit 3 (<i>Disable Serial EEPROM Load on Hot Reset</i>) is Cleared</li> </ul> <p>Refer also to <a href="#">Section 6.8, “Serial EEPROM Loading of NT Port Link Interface Registers – NT Mode,”</a> for further details.</p>	RW	Yes	0
31:28	<b>Reserved</b>	RsvdP	No	0h

**Register 13-208. A34h Virtual Switch GPIO\_PG 0\_9 Direction Control**  
**(Virtual Switch mode – VS Upstream Port(s))**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register provides a virtualized copy of the <b>GPIO 0_9 Direction Control</b> register (Port 0, accessible through the Management Port, offset <b>600h</b>), to support the virtual switches. The main difference is that the virtual switches have no knowledge of which PORT_GOODx# signal is being used.</p> <p>A single GPIO ball/signal cannot be assigned to more than one virtual switch.</p> <p><b>Note:</b> Register offsets <b>A44h</b> and <b>A48h</b>, referenced within this register, are located as follows – Virtual Switch mode – VS Upstream Port(s).</p>				
1:0	<b>VS GPIO_PG 0 PORT_GOODx# Source/Destination</b> <b>As Input:</b> 00b = To <b>VS GPIO_PG 0 PORT_GOODx# Input Data</b> register (offset <b>A44h</b> [0]) 01b = General interrupt (INTx, MSI, or VSx_PEX_INTA#) 10b, 11b = <i>Reserved</i> <b>As Output:</b> 00b = From <b>VS GPIO_PG 0 PORT_GOODx# Output Data</b> register (offset <b>A48h</b> [0]) 01b = PORT_GOODx# 10b = SHP_PERSTx# output 11b = <i>Reserved</i>	RWS	Yes	00b
2	<b>VS GPIO_PG 0 PORT_GOODx# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
4:3	<b>VS GPIO_PG 1 PORT_GOODx# Source/Destination</b> <b>As Input:</b> 00b = To <b>VS GPIO_PG 1 PORT_GOODx# Input Data</b> register (offset <b>A44h</b> [1]) 01b = General interrupt (INTx, MSI, or VSx_PEX_INTA#) 10b, 11b = <i>Reserved</i> <b>As Output:</b> 00b = From <b>VS GPIO_PG 1 PORT_GOODx# Output Data</b> register (offset <b>A48h</b> [1]) 01b = PORT_GOODx# 10b = SHP_PERSTx# output 11b = <i>Reserved</i>	RWS	Yes	00b
5	<b>VS GPIO_PG 1 PORT_GOODx# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0



**Register 13-208. A34h Virtual Switch GPIO\_PG 0\_9 Direction Control (Virtual Switch mode – VS Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:6	<b>VS GPIO_PG 2 PORT_GOODx# Source/Destination</b> <b>As Input:</b> 00b = To <b>VS GPIO_PG 2 PORT_GOODx# Input Data</b> register (offset A44h[2]) 01b = General interrupt (INTx, MSI, or VSx_PEX_INTA#) 10b, 11b = <i>Reserved</i> <b>As Output:</b> 00b = From <b>VS GPIO_PG 2 PORT_GOODx# Output Data</b> register (offset A48h[2]) 01b = PORT_GOODx# 10b = SHP_PERSTx# output 11b = <i>Reserved</i>	RWS	Yes	00b
8	<b>VS GPIO_PG 2 PORT_GOODx# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
10:9	<b>VS GPIO_PG 3 PORT_GOODx# Source/Destination</b> <b>As Input:</b> 00b = To <b>VS GPIO_PG 3 PORT_GOODx# Input Data</b> register (offset A44h[3]) 01b = General interrupt (INTx, MSI, or VSx_PEX_INTA#) 10b, 11b = <i>Reserved</i> <b>As Output:</b> 00b = From <b>VS GPIO_PG 3 PORT_GOODx# Output Data</b> register (offset A48h[3]) 01b = PORT_GOODx# 10b = SHP_PERSTx# output 11b = <i>Reserved</i>	RWS	Yes	00b
11	<b>VS GPIO_PG 3 PORT_GOODx# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
13:12	<b>VS GPIO_PG 4 PORT_GOODx# Source/Destination</b> <b>As Input:</b> 00b = To <b>VS GPIO_PG 4 PORT_GOODx# Input Data</b> register (offset A44h[4]) 01b = General interrupt (INTx, MSI, or VSx_PEX_INTA#) 10b, 11b = <i>Reserved</i> <b>As Output:</b> 00b = From <b>VS GPIO_PG 4 PORT_GOODx# Output Data</b> register (offset A48h[4]) 01b = PORT_GOODx# 10b = SHP_PERSTx# output 11b = <i>Reserved</i>	RWS	Yes	00b
14	<b>VS GPIO_PG 4 PORT_GOODx# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 13-208. A34h Virtual Switch GPIO\_PG 0\_9 Direction Control  
(Virtual Switch mode – VS Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16:15	<b>VS GPIO_PG 5 PORT_GOODx# Source/Destination</b> <b>As Input:</b> 00b = To <b>VS GPIO_PG 5 PORT_GOODx# Input Data</b> register (offset A44h[5]) 01b = General interrupt (INTx, MSI, or VSx_PEX_INTA#) 10b, 11b = <i>Reserved</i> <b>As Output:</b> 00b = From <b>VS GPIO_PG 5 PORT_GOODx# Output Data</b> register (offset A48h[5]) 01b = PORT_GOODx# 10b = SHP_PERSTx# output 11b = <i>Reserved</i>	RWS	Yes	00b
17	<b>VS GPIO_PG 5 PORT_GOODx# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
19:18	<b>VS GPIO_PG 6 PORT_GOODx# Source/Destination</b> <b>As Input:</b> 00b = To <b>VS GPIO_PG 6 PORT_GOODx# Input Data</b> register (offset A44h[6]) 01b = General interrupt (INTx, MSI, or VSx_PEX_INTA#) 10b, 11b = <i>Reserved</i> <b>As Output:</b> 00b = From <b>VS GPIO_PG 6 PORT_GOODx# Output Data</b> register (offset A48h[6]) 01b = PORT_GOODx# 10b = SHP_PERSTx# output 11b = <i>Reserved</i>	RWS	Yes	00b
20	<b>VS GPIO_PG 6 PORT_GOODx# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
22:21	<b>VS GPIO_PG 7 PORT_GOODx# Source/Destination</b> <b>As Input:</b> 00b = To <b>VS GPIO_PG 7 PORT_GOODx# Input Data</b> register (offset A44h[7]) 01b = General interrupt (INTx, MSI, or VSx_PEX_INTA#) 10b, 11b = <i>Reserved</i> <b>As Output:</b> 00b = From <b>VS GPIO_PG 7 PORT_GOODx# Output Data</b> register (offset A48h[7]) 01b = PORT_GOODx# or GPIOx 10b = SHP_PERSTx# output 11b = <i>Reserved</i>	RWS	Yes	00b
23	<b>VS GPIO_PG 7 PORT_GOODx# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

**Register 13-208. A34h Virtual Switch GPIO\_PG 0\_9 Direction Control (Virtual Switch mode – VS Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
25:24	<b>VS GPIO_PG 8 PORT_GOODx# Source/Destination</b> <b>As Input:</b> 00b = To <b>VS GPIO_PG 8 PORT_GOODx# Input Data</b> register (offset A44h[8]) 01b = General interrupt (INTx, MSI, or VSx_PEX_INTA#) 10b, 11b = <i>Reserved</i> <b>As Output:</b> 00b = From <b>VS GPIO_PG 8 PORT_GOODx# Output Data</b> register (offset A48h[8]) 01b = PORT_GOODx# 10b = SHP_PERSTx# output 11b = <i>Reserved</i>	RWS	Yes	00b
26	<b>VS GPIO_PG 8 PORT_GOODx# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
28:27	<b>VS GPIO_PG 9 PORT_GOODx# Source/Destination</b> <b>As Input:</b> 00b = To <b>VS GPIO_PG 9 PORT_GOODx# Input Data</b> register (offset A44h[9]) 01b = General interrupt (INTx, MSI, or VSx_PEX_INTA#) 10b, 11b = <i>Reserved</i> <b>As Output:</b> 00b = From <b>VS GPIO_PG 9 PORT_GOODx# Output Data</b> register (offset A48h[9]) 01b = PORT_GOODx# 10b = SHP_PERSTx# output 11b = <i>Reserved</i>	RWS	Yes	00b
29	<b>VS GPIO_PG 9 PORT_GOODx# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
31:30	<i>Reserved</i>	RsvdP	No	00b

**Register 13-209. A38h Virtual Switch GPIO\_PG 10\_11 Direction Control**  
**(Virtual Switch mode – VS Upstream Port(s))**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register provides a virtualized copy of the <b>GPIO 10_11 Direction Control</b> register (Port 0, accessible through the Management Port, offset 604h), to support the virtual switches. The main difference is that the virtual switches have no knowledge of which PORT_GOODx# signal is being used.</p> <p>A single GPIO ball/signal cannot be assigned to more than one virtual switch.</p> <p><i>Note:</i> Register offsets A44h and A48h, referenced within this register, are located as follows – Virtual Switch mode – VS Upstream Port(s).</p>				
1:0	<b>VS GPIO_PG 10 PORT_GOODx# Source/Destination</b> <b>As Input:</b> 00b = To <b>VS GPIO_PG 10 PORT_GOODx# Input Data</b> register (offset A44h[10]) 01b = General interrupt (INTx, MSI, or VSx_PEX_INTA#) 10b, 11b = <b>Reserved</b> <b>As Output:</b> 00b = From <b>VS GPIO_PG 10 PORT_GOODx# Output Data</b> register (offset A48h[10]) 01b = PORT_GOODx# 10b = SHP_PERSTx# output 11b = <b>Reserved</b>	RWS	Yes	00b
2	<b>VS GPIO_PG 10 PORT_GOODx# Direction Control</b> 0 = Input 1 = Output	RWS	Yes	0
4:3	<b>VS GPIO_PG 11 PORT_GOODx# Source/Destination</b> <i>Note:</i> This field is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this field is reserved. <b>As Input:</b> 00b = To <b>VS GPIO_PG 11 PORT_GOODx# Input Data</b> register (offset A44h[11]) 01b = General interrupt (INTx, MSI, or VSx_PEX_INTA#) 10b, 11b = <b>Reserved</b> <b>As Output:</b> 00b = From <b>VS GPIO_PG 11 PORT_GOODx# Output Data</b> register (offset A48h[11]) 01b = PORT_GOODx# 10b = SHP_PERSTx# output 11b = <b>Reserved</b>	RWS	Yes	00b
5	<b>VS GPIO_PG 11 PORT_GOODx# Direction Control</b> <i>Note:</i> This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b> . 0 = Input 1 = Output	RWS	Yes	0
31:6	<b>Reserved</b>	RsvdP	No	0-0h

**Register 13-210. A3Ch Virtual Switch GPIO\_PG 0\_11 Availability (Virtual Switch mode – VS Upstream Port(s))**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<p><b>Number of GPIO_PGs Available</b></p> <p>Indicates the quantity of PORT_GOODx# signals assigned to each virtual switch. The value corresponds to the quantity of bits that are Set in the <b>VSx GPIO_PG 0_11 Assignment</b> registers (Port 0, accessible through the Management Port, offsets <a href="#">650h</a> through <a href="#">664h</a>). A single GPIO ball/signal cannot be assigned to more than one virtual switch.</p> <p>0h = 1 (GPIO_PG 0)  1h = 2 (GPIO_PG 0_1)  2h = 3 (GPIO_PG 0_2)  3h = 4 (GPIO_PG 0_3)  4h = 5 (GPIO_PG 0_4)  5h = 6 (GPIO_PG 0_5)  6h = 7 (GPIO_PG 0_6)  7h = 8 (GPIO_PG 0_7)  8h = 9 (GPIO_PG 0_8)  9h = 10 (GPIO_PG 0_9)  Ah = 11 (GPIO_PG 0_10)  Bh = 12 (GPIO_PG 0_11)</p> <p>All other encodings are <b>Reserved</b>.</p> <p><i>Note: Although this register is programmable, it should not be written.</i></p>	RWS	Yes	0h
31:4	<b>Reserved</b>	RsvdP	No	0000_000h

**Register 13-211. A40h Virtual Switch GPIO\_PG 0\_11 Input De-Bounce**  
**(Virtual Switch mode – VS Upstream Port(s))**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register provides virtualized copies of the <b>GPIO 0_11 Input De-Bounce</b> register (Port 0, accessible through the Management Port, offset <a href="#">614h</a>), to support the virtual switches. The main difference is that the virtual switches have no knowledge of which PORT_GOODx# signal is being used.</p> <p>A single GPIO ball/signal cannot be assigned to more than one virtual switch.</p> <p><b>Note:</b> Register offsets <a href="#">A34h</a> and <a href="#">A38h</a>, referenced within this register, are located as follows – Virtual Switch mode – VS Upstream Port(s).</p>				
0	<b>VS GPIO_PG 0 PORT_GOODx# Input De-Bounce Control</b> Controls de-bounce when PORT_GOODx# is configured as an input (offset <a href="#">A34h[2]</a> , is Cleared).  0 = PORT_GOODx# input is not de-bounced 1 = PORT_GOODx# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
1	<b>VS GPIO_PG 1 PORT_GOODx# Input De-Bounce Control</b> Controls de-bounce when PORT_GOODx# is configured as an input (offset <a href="#">A34h[5]</a> , is Cleared).  0 = PORT_GOODx# input is not de-bounced 1 = PORT_GOODx# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
2	<b>VS GPIO_PG 2 PORT_GOODx# Input De-Bounce Control</b> Controls de-bounce when PORT_GOODx# is configured as an input (offset <a href="#">A34h[8]</a> , is Cleared).  0 = PORT_GOODx# input is not de-bounced 1 = PORT_GOODx# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
3	<b>VS GPIO_PG 3 PORT_GOODx# Input De-Bounce Control</b> Controls de-bounce when PORT_GOODx# is configured as an input (offset <a href="#">A34h[11]</a> , is Cleared).  0 = PORT_GOODx# input is not de-bounced 1 = PORT_GOODx# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0

**Register 13-211. A40h Virtual Switch GPIO\_PG 0\_11 Input De-Bounce (Virtual Switch mode – VS Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>VS GPIO_PG 4 PORT_GOODx# Input De-Bounce Control</b> Controls de-bounce when PORT_GOODx# is configured as an input (offset A34h[14], is Cleared). 0 = PORT_GOODx# input is not de-bounced 1 = PORT_GOODx# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
5	<b>VS GPIO_PG 5 PORT_GOODx# Input De-Bounce Control</b> Controls de-bounce when PORT_GOODx# is configured as an input (offset A34h[17], is Cleared). 0 = PORT_GOODx# input is not de-bounced 1 = PORT_GOODx# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
6	<b>VS GPIO_PG 6 PORT_GOODx# Input De-Bounce Control</b> Controls de-bounce when PORT_GOODx# is configured as an input (offset A34h[20], is Cleared). 0 = PORT_GOODx# input is not de-bounced 1 = PORT_GOODx# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
7	<b>VS GPIO_PG 7 PORT_GOODx# Input De-Bounce Control</b> Controls de-bounce when PORT_GOODx# is configured as an input (offset A34h[23], is Cleared). 0 = PORT_GOODx# input is not de-bounced 1 = PORT_GOODx# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0

**Register 13-211. A40h Virtual Switch GPIO\_PG 0\_11 Input De-Bounce  
(Virtual Switch mode – VS Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>VS GPIO_PG 8 PORT_GOODx# Input De-Bounce Control</b> Controls de-bounce when PORT_GOODx# is configured as an input (offset A34h[26], is Cleared). 0 = PORT_GOODx# input is not de-bounced 1 = PORT_GOODx# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
9	<b>VS GPIO_PG 9 PORT_GOODx# Input De-Bounce Control</b> Controls de-bounce when PORT_GOODx# is configured as an input (offset A34h[29], is Cleared). 0 = PORT_GOODx# input is not de-bounced 1 = PORT_GOODx# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
10	<b>VS GPIO_PG 10 PORT_GOODx# Input De-Bounce Control</b> Controls de-bounce when PORT_GOODx# is configured as an input (offset A38h[2], is Cleared). 0 = PORT_GOODx# input is not de-bounced 1 = PORT_GOODx# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
11	<b>VS GPIO_PG 11 PORT_GOODx# Input De-Bounce Control</b> <i>Note: This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> Controls de-bounce when PORT_GOODx# is configured as an input (offset A38h[5], is Cleared). 0 = PORT_GOODx# input is not de-bounced 1 = PORT_GOODx# input is de-bounced; de-bounce time is approximately 1.3 ms	RWS	Yes	0
31:12	<b>Reserved</b>	RsvdP	No	0000_0h



**Register 13-212. A44h Virtual Switch GPIO\_PG 0\_11 Input Data (Virtual Switch mode – VS Upstream Port(s))**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register provides virtualized copies of the <b>GPIO 0_11 Input Data</b> register (Port 0, accessible through the Management Port, offset <b>61Ch</b>), to support the virtual switches. The main difference is that the virtual switches have no knowledge of which PORT_GOODx# signal is being used.</p> <p>A single GPIO ball/signal cannot be assigned to more than one virtual switch.</p> <p><b>Note:</b> Register offsets <b>A34h</b> and <b>A38h</b>, referenced within this register, are located as follows – Virtual Switch mode – VS Upstream Port(s).</p>				
0	<b>VS GPIO_PG 0 PORT_GOODx# Input Data</b> If PORT_GOODx# is configured as an output (offset <b>A34h[2]</b> , is Set), Reads return a value of 0. If PORT_GOODx# is configured as an input (offset <b>A34h[2]</b> , is Cleared), Reads return the logic value of the voltage on PORT_GOODx# or GPIOx.	ROS	No	0
1	<b>VS GPIO_PG 1 PORT_GOODx# Input Data</b> If PORT_GOODx# is configured as an output (offset <b>A34h[5]</b> , is Set), Reads return a value of 0. If PORT_GOODx# is configured as an input (offset <b>A34h[5]</b> , is Cleared), Reads return the logic value of the voltage on PORT_GOODx# or GPIOx.	ROS	No	0
2	<b>VS GPIO_PG 2 PORT_GOODx# Input Data</b> If PORT_GOODx# is configured as an output (offset <b>A34h[8]</b> , is Set), Reads return a value of 0. If PORT_GOODx# is configured as an input (offset <b>A34h[8]</b> , is Cleared), Reads return the logic value of the voltage on PORT_GOODx# or GPIOx.	ROS	No	0
3	<b>VS GPIO_PG 3 PORT_GOODx# Input Data</b> If PORT_GOODx# is configured as an output (offset <b>A34h[11]</b> , is Set), Reads return a value of 0. If PORT_GOODx# is configured as an input (offset <b>A34h[11]</b> , is Cleared), Reads return the logic value of the voltage on PORT_GOODx# or GPIOx.	ROS	No	0

**Register 13-212. A44h Virtual Switch GPIO\_PG 0\_11 Input Data  
(Virtual Switch mode – VS Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>VS GPIO_PG 4 PORT_GOODx# Input Data</b> If PORT_GOODx# is configured as an output (offset A34h[14], is Set), Reads return a value of 0. If PORT_GOODx# is configured as an input (offset A34h[14], is Cleared), Reads return the logic value of the voltage on PORT_GOODx# or GPIOx.	ROS	No	0
5	<b>VS GPIO_PG 5 PORT_GOODx# Input Data</b> If PORT_GOODx# is configured as an output (offset A34h[17], is Set), Reads return a value of 0. If PORT_GOODx# is configured as an input (offset A34h[17], is Cleared), Reads return the logic value of the voltage on PORT_GOODx# or GPIOx.	ROS	No	0
6	<b>VS GPIO_PG 6 PORT_GOODx# Input Data</b> If PORT_GOODx# is configured as an output (offset A34h[20], is Set), Reads return a value of 0. If PORT_GOODx# is configured as an input (offset A34h[20], is Cleared), Reads return the logic value of the voltage on PORT_GOODx# or GPIOx.	ROS	No	0
7	<b>VS GPIO_PG 7 PORT_GOODx# Input Data</b> If PORT_GOODx# is configured as an output (offset A34h[23], is Set), Reads return a value of 0. If PORT_GOODx# is configured as an input (offset A34h[23], is Cleared), Reads return the logic value of the voltage on PORT_GOODx# or GPIOx.	ROS	No	0

**Register 13-212. A44h Virtual Switch GPIO\_PG 0\_11 Input Data (Virtual Switch mode – VS Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>VS GPIO_PG 8 PORT_GOODx# Input Data</b> If PORT_GOODx# is configured as an output (offset A34h[26], is Set), Reads return a value of 0. If PORT_GOODx# is configured as an input (offset A34h[26], is Cleared), Reads return the logic value of the voltage on PORT_GOODx# or GPIOx.	ROS	No	0
9	<b>VS GPIO_PG 9 PORT_GOODx# Input Data</b> If PORT_GOODx# is configured as an output (offset A34h[29], is Set), Reads return a value of 0. If PORT_GOODx# is configured as an input (offset A34h[29], is Cleared), Reads return the logic value of the voltage on PORT_GOODx# or GPIOx.	ROS	No	0
10	<b>VS GPIO_PG 10 PORT_GOODx# Input Data</b> If PORT_GOODx# is configured as an output (offset A38h[2], is Set), Reads return a value of 0. If PORT_GOODx# is configured as an input (offset A38h[2], is Cleared), Reads return the logic value of the voltage on PORT_GOODx# or GPIOx.	ROS	No	0
11	<b>VS GPIO_PG 11 PORT_GOODx# Input Data</b> <i>Note: This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> If PORT_GOODx# is configured as an output (offset A38h[5], is Set), Reads return a value of 0. If PORT_GOODx# is configured as an input (offset A38h[5], is Cleared), Reads return the logic value of the voltage on PORT_GOODx# or GPIOx.	ROS	No	0
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

**Register 13-213. A48h Virtual Switch GPIO\_PG 0\_11 Output Data  
(Virtual Switch mode – VS Upstream Port(s))**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register provides virtualized copies of the <b>GPIO 0_11 Output Data</b> register (Port 0, accessible through the Management Port, offset 624h), to support the virtual switches. The main difference is that the virtual switches have no knowledge of which PORT_GOODx# signal is being used.</p> <p>A single GPIO ball/signal cannot be assigned to more than one virtual switch.</p> <p><b>Note:</b> Register offsets A34h and A38h, referenced within this register, are located as follows – Virtual Switch mode – VS Upstream Port(s).</p>				
0	<b>VS GPIO_PG 0 PORT_GOODx# Output Data</b> If PORT_GOODx# is configured as an output (offset A34h[2], is Set), the value written to this bit is immediately driven to the PORT_GOODx# output. Reads return the value written.	RWS	Yes	0
1	<b>VS GPIO_PG 1 PORT_GOODx# Output Data</b> If PORT_GOODx# is configured as an output (offset A34h[5], is Set), the value written to this bit is immediately driven to the PORT_GOODx# output. Reads return the value written.	RWS	Yes	0
2	<b>VS GPIO_PG 2 PORT_GOODx# Output Data</b> If PORT_GOODx# is configured as an output (offset A34h[8], is Set), the value written to this bit is immediately driven to the PORT_GOODx# output. Reads return the value written.	RWS	Yes	0
3	<b>VS GPIO_PG 3 PORT_GOODx# Output Data</b> If PORT_GOODx# is configured as an output (offset A34h[11], is Set), the value written to this bit is immediately driven to the PORT_GOODx# output. Reads return the value written.	RWS	Yes	0
4	<b>VS GPIO_PG 4 PORT_GOODx# Output Data</b> If PORT_GOODx# is configured as an output (offset A34h[14], is Set), the value written to this bit is immediately driven to the PORT_GOODx# output. Reads return the value written.	RWS	Yes	0
5	<b>VS GPIO_PG 5 PORT_GOODx# Output Data</b> If PORT_GOODx# is configured as an output (offset A34h[17], is Set), the value written to this bit is immediately driven to the PORT_GOODx# output. Reads return the value written.	RWS	Yes	0
6	<b>VS GPIO_PG 6 PORT_GOODx# Output Data</b> If PORT_GOODx# is configured as an output (offset A34h[20], is Set), the value written to this bit is immediately driven to the PORT_GOODx# output. Reads return the value written.	RWS	Yes	0
7	<b>VS GPIO_PG 7 PORT_GOODx# Output Data</b> If PORT_GOODx# is configured as an output (offset A34h[23], is Set), the value written to this bit is immediately driven to the PORT_GOODx# output. Reads return the value written.	RWS	Yes	0

**Register 13-213. A48h Virtual Switch GPIO\_PG 0\_11 Output Data (Virtual Switch mode – VS Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>VS GPIO_PG 8 PORT_GOODx# Output Data</b> If PORT_GOODx# is configured as an output (offset A34h[26], is Set), the value written to this bit is immediately driven to the PORT_GOODx# output. Reads return the value written.	RWS	Yes	0
9	<b>VS GPIO_PG 9 PORT_GOODx# Output Data</b> If PORT_GOODx# is configured as an output (offset A34h[29], is Set), the value written to this bit is immediately driven to the PORT_GOODx# output. Reads return the value written.	RWS	Yes	0
10	<b>VS GPIO_PG 10 PORT_GOODx# Output Data</b> If PORT_GOODx# is configured as an output (offset A38h[2], is Set), the value written to this bit is immediately driven to the PORT_GOODx# output. Reads return the value written.	RWS	Yes	0
11	<b>VS GPIO_PG 11 PORT_GOODx# Output Data</b> <i>Note: This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> If PORT_GOODx# is configured as an output (offset A38h[5], is Set), the value written to this bit is immediately driven to the PORT_GOODx# output. Reads return the value written.	RWS	Yes	0
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

**Register 13-214. A4Ch Virtual Switch GPIO\_PG 0\_11 Interrupt Polarity  
(Virtual Switch mode – VS Upstream Port(s))**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register provides virtualized copies of the <b>GPIO 0_11 Interrupt Polarity</b> register (Port 0, accessible through the Management Port, offset <b>62Ch</b>), to support the virtual switches. The main difference is that the virtual switches have no knowledge of which PORT_GOODx# signal is being used.</p> <p>A single GPIO ball/signal cannot be assigned to more than one virtual switch.</p>				
0	<b>VS GPIO_PG 0 PORT_GOODx# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High for the PORT_GOODx# signal. 0 = GPIO Interrupt input is Active-Low 1 = GPIO Interrupt input is Active-High	RWS	Yes	0
1	<b>VS GPIO_PG 1 PORT_GOODx# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High for the PORT_GOODx# signal. 0 = GPIO Interrupt input is Active-Low 1 = GPIO Interrupt input is Active-High	RWS	Yes	0
2	<b>VS GPIO_PG 2 PORT_GOODx# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High for the PORT_GOODx# signal. 0 = GPIO Interrupt input is Active-Low 1 = GPIO Interrupt input is Active-High	RWS	Yes	0
3	<b>VS GPIO_PG 3 PORT_GOODx# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High for the PORT_GOODx# signal. 0 = GPIO Interrupt input is Active-Low 1 = GPIO Interrupt input is Active-High	RWS	Yes	0

**Register 13-214. A4Ch Virtual Switch GPIO\_PG 0\_11 Interrupt Polarity (Virtual Switch mode – VS Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>VS GPIO_PG 4 PORT_GOODx# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High for the PORT_GOODx# signal. 0 = GPIO Interrupt input is Active-Low 1 = GPIO Interrupt input is Active-High	RWS	Yes	0
5	<b>VS GPIO_PG 5 PORT_GOODx# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High for the PORT_GOODx# signal. 0 = GPIO Interrupt input is Active-Low 1 = GPIO Interrupt input is Active-High	RWS	Yes	0
6	<b>VS GPIO_PG 6 PORT_GOODx# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High for the PORT_GOODx# signal. 0 = GPIO Interrupt input is Active-Low 1 = GPIO Interrupt input is Active-High	RWS	Yes	0
7	<b>VS GPIO_PG 7 PORT_GOODx# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High for the PORT_GOODx# signal. 0 = GPIO Interrupt input is Active-Low 1 = GPIO Interrupt input is Active-High	RWS	Yes	0
8	<b>VS GPIO_PG 8 PORT_GOODx# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High for the PORT_GOODx# signal. 0 = GPIO Interrupt input is Active-Low 1 = GPIO Interrupt input is Active-High	RWS	Yes	0
9	<b>VS GPIO_PG 9 PORT_GOODx# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High for the PORT_GOODx# signal. 0 = GPIO Interrupt input is Active-Low 1 = GPIO Interrupt input is Active-High	RWS	Yes	0
10	<b>VS GPIO_PG 10 PORT_GOODx# Interrupt Polarity</b> Controls whether GPIO Interrupt input is Active-Low or Active-High for the PORT_GOODx# signal. 0 = GPIO Interrupt input is Active-Low 1 = GPIO Interrupt input is Active-High	RWS	Yes	0
11	<b>VS GPIO_PG 11 PORT_GOODx# Interrupt Polarity</b> <i>Note: This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> Controls whether GPIO Interrupt input is Active-Low or Active-High for the PORT_GOODx# signal. 0 = GPIO Interrupt input is Active-Low 1 = GPIO Interrupt input is Active-High	RWS	Yes	0
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 13-215. A50h Virtual Switch GPIO\_PG 0\_11 Interrupt Status**  
**(Virtual Switch mode – VS Upstream Port(s))**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register provides virtualized copies of the <b>GPIO 0_11 Interrupt Status</b> register (Port 0, accessible through the Management Port, offset <b>634h</b>), to support the virtual switches. The main difference is that the virtual switches have no knowledge of which PORT_GOODx# signal is being used.</p> <p>Interrupt status remains Set, as long the corresponding PORT_GOODx# signal is asserted, and Clears on its own when the corresponding PORT_GOODx# input de-asserts to the inactive state.</p> <p>The active state of each interrupt is controlled by its respective <b>Virtual Switch GPIO_PG 0_11 Interrupt Polarity</b> register bits (offset <b>A4Ch</b>).</p> <p>A single GPIO ball/signal cannot be assigned to more than one virtual switch.</p> <p><b>Note:</b> The bits in this register can be masked by their respective <b>Virtual Switch GPIO_PG 0_11 Interrupt Mask</b> register bits (offset <b>A54h</b>).</p>				
0	<b>VS GPIO_PG 0 PORT_GOODx# Interrupt Status</b> Indicates whether GPIO interrupts are inactive or active for the PORT_GOODx# signal. 0 = GPIO interrupt is inactive 1 = GPIO interrupt is active	ROS	No	0
1	<b>VS GPIO_PG 1 PORT_GOODx# Interrupt Status</b> Indicates whether GPIO interrupts are inactive or active for the PORT_GOODx# signal. 0 = GPIO Interrupt input is inactive 1 = GPIO Interrupt input is active	ROS	No	0
2	<b>VS GPIO_PG 2 PORT_GOODx# Interrupt Status</b> Indicates whether GPIO interrupts are inactive or active for the PORT_GOODx# signal. 0 = GPIO Interrupt input is inactive 1 = GPIO Interrupt input is active	ROS	No	0
3	<b>VS GPIO_PG 3 PORT_GOODx# Interrupt Status</b> Indicates whether GPIO interrupts are inactive or active for the PORT_GOODx# signal. 0 = GPIO Interrupt input is inactive 1 = GPIO Interrupt input is active	ROS	No	0



**Register 13-215. A50h Virtual Switch GPIO\_PG 0\_11 Interrupt Status (Virtual Switch mode – VS Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>VS GPIO_PG 4 PORT_GOODx# Interrupt Status</b> Indicates whether GPIO interrupts are inactive or active for the PORT_GOODx# signal. 0 = GPIO Interrupt input is inactive 1 = GPIO Interrupt input is active	ROS	No	0
5	<b>VS GPIO_PG 5 PORT_GOODx# Interrupt Status</b> Indicates whether GPIO interrupts are inactive or active for the PORT_GOODx# signal. 0 = GPIO Interrupt input is inactive 1 = GPIO Interrupt input is active	ROS	No	0
6	<b>VS GPIO_PG 6 PORT_GOODx# Interrupt Status</b> Indicates whether GPIO interrupts are inactive or active for the PORT_GOODx# signal. 0 = GPIO Interrupt input is inactive 1 = GPIO Interrupt input is active	ROS	No	0
7	<b>VS GPIO_PG 7 PORT_GOODx# Interrupt Status</b> Indicates whether GPIO interrupts are inactive or active for the PORT_GOODx# signal. 0 = GPIO Interrupt input is inactive 1 = GPIO Interrupt input is active	ROS	No	0
8	<b>VS GPIO_PG 8 PORT_GOODx# Interrupt Status</b> Indicates whether GPIO interrupts are inactive or active for the PORT_GOODx# signal. 0 = GPIO Interrupt input is inactive 1 = GPIO Interrupt input is active	ROS	No	0
9	<b>VS GPIO_PG 9 PORT_GOODx# Interrupt Status</b> Indicates whether GPIO interrupts are inactive or active for the PORT_GOODx# signal. 0 = GPIO Interrupt input is inactive 1 = GPIO Interrupt input is active	ROS	No	0
10	<b>VS GPIO_PG 10 PORT_GOODx# Interrupt Status</b> Indicates whether GPIO interrupts are inactive or active for the PORT_GOODx# signal. 0 = GPIO Interrupt input is inactive 1 = GPIO Interrupt input is active	ROS	No	0
11	<b>VS GPIO_PG 11 PORT_GOODx# Interrupt Status</b> <i>Note: This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> Indicates whether GPIO interrupts are inactive or active for the PORT_GOODx# signal. 0 = GPIO Interrupt input is inactive 1 = GPIO Interrupt input is active	ROS	No	0
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 13-216. A54h Virtual Switch GPIO\_PG 0\_11 Interrupt Mask**  
**(Virtual Switch mode – VS Upstream Port(s))**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register provides virtualized copies of the <b>GPIO 0_11 Interrupt Mask</b> register (Port 0, accessible through the Management Port, offset <b>63Ch</b>), to support the virtual switches. The main difference is that the virtual switches have no knowledge of which <b>PORT_GOODx#</b> signal is being used.</p> <p>A single GPIO ball/signal cannot be assigned to more than one virtual switch.</p> <p><b>Note:</b> The bits in this register can be used to mask their respective <b>Virtual Switch GPIO_PG 0_11 Interrupt Status</b> register bits (offset <b>A50h</b>).</p>				
0	<b>VS GPIO_PG 0 PORT_GOODx# Interrupt Mask</b> Indicates whether GPIO interrupts are not masked or masked for the <b>PORT_GOODx#</b> signal. 0 = GPIO interrupt is not masked. 1 = GPIO interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
1	<b>VS GPIO_PG 1 PORT_GOODx# Interrupt Mask</b> Indicates whether GPIO interrupts are not masked or masked for the <b>PORT_GOODx#</b> signal. 0 = GPIO interrupt is not masked. 1 = GPIO interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
2	<b>VS GPIO_PG 2 PORT_GOODx# Interrupt Mask</b> Indicates whether GPIO interrupts are not masked or masked for the <b>PORT_GOODx#</b> signal. 0 = GPIO interrupt is not masked. 1 = GPIO interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
3	<b>VS GPIO_PG 3 PORT_GOODx# Interrupt Mask</b> Indicates whether GPIO interrupts are not masked or masked for the <b>PORT_GOODx#</b> signal. 0 = GPIO interrupt is not masked. 1 = GPIO interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1

**Register 13-216. A54h Virtual Switch GPIO\_PG 0\_11 Interrupt Mask (Virtual Switch mode – VS Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4	<b>VS GPIO_PG 4 PORT_GOODx# Interrupt Mask</b> Indicates whether GPIO interrupts are not masked or masked for the PORT_GOODx# signal. 0 = GPIO interrupt is not masked. 1 = GPIO interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
5	<b>VS GPIO_PG 5 PORT_GOODx# Interrupt Mask</b> Indicates whether GPIO interrupts are not masked or masked for the PORT_GOODx# signal. 0 = GPIO interrupt is not masked. 1 = GPIO interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
6	<b>VS GPIO_PG 6 PORT_GOODx# Interrupt Mask</b> Indicates whether GPIO interrupts are not masked or masked for the PORT_GOODx# signal. 0 = GPIO interrupt is not masked. 1 = GPIO interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
7	<b>VS GPIO_PG 7 PORT_GOODx# Interrupt Mask</b> Indicates whether GPIO interrupts are not masked or masked for the PORT_GOODx# signal. 0 = GPIO interrupt is not masked. 1 = GPIO interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1

**Register 13-216. A54h Virtual Switch GPIO\_PG 0\_11 Interrupt Mask  
(Virtual Switch mode – VS Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>VS GPIO_PG 8 PORT_GOODx# Interrupt Mask</b> Indicates whether GPIO interrupts are not masked or masked for the PORT_GOODx# signal. 0 = GPIO interrupt is not masked. 1 = GPIO interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
9	<b>VS GPIO_PG 9 PORT_GOODx# Interrupt Mask</b> Indicates whether GPIO interrupts are not masked or masked for the PORT_GOODx# signal. 0 = GPIO interrupt is not masked. 1 = GPIO interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
10	<b>VS GPIO_PG 10 PORT_GOODx# Interrupt Mask</b> Indicates whether GPIO interrupts are not masked or masked for the PORT_GOODx# signal. 0 = GPIO interrupt is not masked. 1 = GPIO interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
11	<b>VS GPIO_PG 11 PORT_GOODx# Interrupt Mask</b> <i><b>Note:</b> This bit is used only in Virtual Switch mode, when four, five, or six virtual switches are enabled; otherwise, this bit is <b>Reserved</b>.</i> Indicates whether GPIO interrupts are not masked or masked for the PORT_GOODx# signal. 0 = GPIO interrupt is not masked. 1 = GPIO interrupt is masked. When an interrupt is masked, the corresponding <b>Interrupt Status</b> register bit is not updated.	RWS	Yes	1
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

This section details the Latency Tolerance Reporting (LTR) Extended Capability registers. [Table 13-34](#) defines the register map.

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

Next Capability Offset (B70h) (Upstream)	Capability Version (1h) (Upstream)	PCI Express Extended Capability ID (0018h) (Upstream)	B00h
<i>Reserved</i> (Downstream)			
Max No-Snoop Latency (Upstream) <i>Reserved</i> (Downstream)	Max Snoop Latency (Upstream) <i>Reserved</i> (Downstream)		B04h
LTR Message Received Status (Upstream) <i>Reserved</i> (Downstream)			B08h

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>PCI Express Extended Capability ID</b> Program to 0018h, as required by the <i>PCI Express Base r3.0</i> .	Upstream	ROS	Yes	0018h
	<b>Reserved</b>	Downstream	RsvdP	No	0000h
19:16	<b>Capability Version</b> Program to 1h, as required by the <i>PCI Express Base r3.0</i> .	Upstream	ROS	Yes	1h
	<b>Reserved</b>	Downstream	RsvdP	No	0h
31:20	<b>Next Capability Offset</b> Program to B70h, which addresses the <b>Vendor-Specific Extended Capability 2</b> structure.	Upstream	ROS	Yes	<a href="#">B70h</a>
	<b>Reserved</b>	Downstream	RsvdP	No	000h

**Register 13-218. B04h Max Snoop/No-Snoop Latency (Upstream Port(s))**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Max Snoop Latency</b>					
9:0	<b>Max Snoop LatencyValue</b> Along with field [12:10] ( <i>Max Snoop LatencyScale</i> ), this register specifies the maximum Snoop latency that the PEX 8748 is allowed to request. Software should program this field to the platform's maximum supported latency, or less.	Upstream	RW	Yes	0-0h
	<b>Reserved</b>	Downstream	RsvdP	No	0-0h
12:10	<b>Max Snoop LatencyScale</b> Provides a scale for the value contained within field [9:0] ( <i>Max Snoop LatencyValue</i> ). The encoding is the same as the <i>LatencyScale</i> fields in the LTR Message. (Refer to the <i>PCI Express Base r3.0</i> , Section 6.18.) Hardware operation is undefined if software writes a Not Permitted value to this field.	Upstream	RW	Yes	000b
	<b>Reserved</b>	Downstream	RsvdP	No	000b
15:13	<b>Reserved</b>		RsvdP	No	000b
<b>Max No-Snoop Latency</b>					
25:16	<b>Max No-Snoop LatencyValue</b> Along with field [28:26] ( <i>Max No-Snoop LatencyScale</i> ), this register specifies the maximum No-Snoop latency that the PEX 8748 is allowed to request. Software should program this field to the platform's maximum supported latency, or less.	Upstream	RW	Yes	0-0h
	<b>Reserved</b>	Downstream	RsvdP	No	0-0h
28:26	<b>Max No-Snoop LatencyScale</b> Provides a scale for the value contained within field [25:16] ( <i>Max No-Snoop LatencyValue</i> ). The encoding is the same as the <i>LatencyScale</i> fields in the LTR Message. (Refer to the <i>PCI Express Base r3.0</i> , Section 6.18.) Hardware operation is undefined if software writes a Not Permitted value to this field.	Upstream	RW	Yes	000b
	<b>Reserved</b>	Downstream	RsvdP	No	000b
31:29	<b>Reserved</b>		RsvdP	No	000b

**Register 13-219. B08h LTR Message Received Status (Upstream Port(s))**

Bit(s)	Description	Upstream Port <i>x</i>	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>When driven with a value of 1, this register holds the Port in reset, including the Port PCI-to-PCI bridge and the hierarchy below it. A value of 0 indicates to un-reset the PCI-to-PCI bridge and the hierarchy below it.</p> <p><b>Notes:</b> The Downstream Port bits are <b>Reserved</b>, RsvdP, not serial EEPROM nor I<sup>2</sup>C writable, and have a default value of 0.</p> <p>The <b>Upstream Port <i>x</i></b> column is provided to indicate the Port Number associated with bits/fields that include “Port <i>x</i>” in their name.</p> <p><a href="#">Table 13-5</a> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>					
0	<b>Port <i>x</i> LTR Initiate or Received</b>	0	RO	Yes	0
1		1	RO	Yes	0
2		2	RO	Yes	0
3		3	RO	Yes	0
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>Port <i>x</i> LTR Initiate or Received</b>	8	RO	Yes	0
9		9	RO	Yes	0
10		10	RO	Yes	0
11		11	RO	Yes	0
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b
16	<b>Port <i>x</i> LTR Initiate or Received</b>	16	RO	Yes	0
17		17	RO	Yes	0
18		18	RO	Yes	0
19		19	RO	Yes	0
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
31:22	<b>Reserved</b>		RsvdP	No	0-0h

# 13.18 Device-Specific Registers (Offsets B70h – DFCh)

This section details the Device-Specific registers located at offsets B70h through DFCh. Device-Specific registers are unique to the PEX 8748 and not referenced in the *PCI Express Base r3.0*. Table 13-17 defines the register map.

Other Device-Specific registers are detailed in:

- Section 13.16, “Device-Specific Registers (Offsets 1C0h – A74h)”
- Section 13.20, “Device-Specific Registers – Virtual Switch (Offsets F00h – F20h), Virtual Switch Mode”
- Section 13.22, “Device-Specific Registers (Offsets F30h – FB0h)”

*Note: It is recommended that these registers not be changed from their default values.*

Table 13-35. Device-Specific Register Map (Offsets B70h – DFCh)

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																																				
Next Capability Offset 2 (000h)																1h																PCI Express Extended Capability ID 2 (000Bh)																																				B70h
Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets B70h – B7Ch)																																																				...																
																																																				B7Ch																
Device-Specific Registers – Physical Layer (Offsets B80h – C88h)																																																				B80h																
																																																				...																
Device-Specific Registers – Requester ID Read Back (Offsets C8Ch – C90h)																																																				C88h																
																																																				C8Ch																
Device-Specific Registers – Requester ID Read Back (Offsets C8Ch – C90h)																																																				...																
																																																				C90h																
Reserved																																																				C94h –	DFCh															



### 13.18.1 Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets B70h – B7Ch)

This section details the Device-Specific Vendor-Specific Extended Capability 2 registers. [Table 13-36](#) defines the register map.

**Table 13-36. Device-Specific, Vendor-Specific Extended Capability 2 Register Map (All Ports)**

<div> <div>31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16</div> <div>15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</div> </div>																																
Next Capability Offset 2 (000h)																Capability Version 2 (1h)				PCI Express Extended Capability ID 2 (000Bh)												B70h
Vendor-Specific Header 2																																B74h
Hardwired Device ID																Hardwired Vendor ID																B78h
<i>Reserved</i>																								Hardwired Revision ID								B7Ch

### Register 13-220. B70h Vendor-Specific Extended Capability 2 (All Ports)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>PCI Express Extended Capability ID 2</b> Program to 000Bh, to indicate that the Capability structure is the <b>Vendor-Specific Extended Capability</b> structure.	ROS	Yes	000Bh
19:16	<b>Capability Version 2</b>	ROS	Yes	1h
31:20	<b>Next Capability Offset 2</b> 000h = This extended capability is the last extended capability in the PEX 8748 Extended Capabilities list	ROS	Yes	000h

### Register 13-221. B74h Vendor-Specific Header 2 (All Ports)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Vendor-Specific ID 2</b> ID Number of this Extended Capability structure.	ROS	Yes	0001h
19:16	<b>Vendor-Specific Rev 2</b> Version Number of this structure.	ROS	Yes	0h
31:20	<b>Vendor-Specific Length 2</b> Quantity of bytes in the entire structure.	ROS	Yes	010h

**Register 13-222. B78h PLX Hardwired Configuration ID  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Hardwired Vendor ID</b> Always returns the PLX PCI-SIG-assigned <b>Vendor ID</b> value, 10B5h.	ROS	No	10B5h
31:16	<b>Hardwired Device ID</b> Always returns the PEX 8748 default <b>Device ID</b> value, 8748h.	ROS	No	8748h

**Register 13-223. B7Ch PLX Hardwired Revision ID  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Hardwired Revision ID</b> Always returns the PEX 8748 default <b>Revision ID</b> value – BAh.	ROS	No	Current Rev # (BAh)
31:8	<b>Reserved</b>	RsvdP	No	0000_00h

## 13.18.2 Device-Specific Registers – Physical Layer (Offsets B80h – C88h)

This section details the Device-Specific Physical Layer (PHY) registers located at offsets B80h through C88h. [Table 13-37](#) defines the register map.

Other Device-Specific PHY registers are detailed in [Section 13.16.4, “Device-Specific Registers – Physical Layer \(Offsets 200h – 25Ch\).”](#)

**Table 13-37. Device-Specific PHY Register Map (Offsets B80h – C88h) (Ports<sup>a</sup>)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																
<i>Factory Test Only/Reserved</i>																SerDes Control																																B80h
Synchronous Advertised N_FTS																																B84h																
<i>Reserved</i>																																B88h																
Asynchronous Advertised N_FTS																																B8Ch																
<i>Reserved</i>																																B90h																
<i>Reserved</i>																<i>Factory Test Only</i>																PIPE Transmit Parameter Control																B94h
2.5 GT/s -3.5 dB Transmit Parameters																																B98h																
5.0 GT/s -6 dB Transmit Parameters																																B9Ch																
5.0 GT/s -3.5 dB Transmit Parameters																																BA0h																
8.0 GT/s 0 dB Transmit Parameters																																BA4h																
<i>Reserved</i>																																BA8h –	BB4h															
<i>Reserved</i>																DC Balance Control																																BB8h
Trained Coefficient Status																																BBCh																
Port Accumulation Control																																BC0h																
Recovery Diagnostic																																BC4h																
User Lane Equalization Control																																BC8h																
<i>Reserved</i>																																BCCh –	BD0h															
Gen 3 Coefficient Values																																BD4h																
Equalization Control																																BD8h																
<i>Factory Test Only/Reserved</i>																Signal Detect Level																BDCh																
Gen 3 LFSR Seed																																BE0h																
<i>Factory Test Only</i>																																BE4h –	BE8h															
TLP Round Trip Timer Control																																BECh																
Port Receiver Error Counter																																BF0h																
<i>Reserved</i>																																BF4h																
<i>Factory Test Only</i>																																BF8h																
AHB Access Control																																BFCh																
<i>Reserved</i>																																C00h –	C88h															

a. Certain registers are Port-specific, others are Station-specific; all are Device-specific.

**Register 13-224. B80h SerDes Control**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register controls SerDes logic parameters.</p> <p><b>Notes:</b> <i>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
2:0	<b>Electrical Idle Inference Time Select</b> Selects the amount of time to wait until no SKIP Ordered-Sets are detected, for Electrical Idle to be inferred. Does not affect Electrical Idle inference during the <i>Recovery.Speed</i> substate.  000b = 4 $\mu$ s 001b = 6 $\mu$ s 010b = 8 $\mu$ s 011b = 16 $\mu$ s 100b = 32 $\mu$ s 101b = 64 $\mu$ s 110b = 128 $\mu$ s (default) 111b = 256 $\mu$ s	RWS	Yes	110b
3	<b>Reserved</b>	RsvdP	No	0
5:4	<b>Recovery.Speed Electrical Idle Inference Time Select</b>	RWS	Yes	00b
7:6	<b>Reserved</b>	RsvdP	No	00b
8	<b>Force SerDes Out</b> Transmit Data (TD) [19:0] Force enable.  1 = All bits of the TD[19:0] inputs, of the SerDes in this Station, are forced to the state specified by bit 9 ( <i>SerDes Out Data Force State</i> )	RWS	Yes	0
9	<b>SerDes Out Data Force State</b> TD[19:0] Force state. Specifies the state to which the TD[19:0] inputs are forced, when bit 8 ( <i>Force SerDes Out</i> ) is Set.	RWS	Yes	0
11:10	<b>Reserved</b>	RsvdP	No	00b
14:12	<b>Tx Parameter Port Select</b> Selects the Port to which the value written in the <b>Tx Parameter Control</b> registers will be applied.	RWS	Yes	000b
15	<b>Reserved</b>	RsvdP	No	0
21:16	<b>Tx Margin Override</b>	RWS	Yes	0-0h
23:22	<b>Reserved</b>	RsvdP	No	00b
27:24	<b>Factory Test Only</b>	RWS	Yes	0h
31:28	<b>Reserved</b>	RsvdP	No	0h

**Register 13-225. B84h Synchronous Advertised N\_FTS**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register advertises the Number of Fast Training Sets (N_FTS) values for synchronous clocking.					
<i>Notes: The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name.   <b>Table 13-5</b> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.   The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i>					
7:0	<b>Port x Synchronous Tx N_FTS</b>	0, 8, 16	RWS	Yes	80h
15:8	When clocking is synchronous, specifies the <i>N_FTS</i> field value in Training Sets transmitted by the Port, when the Port's <b>Link Control</b> register <i>Common Clock Configuration</i> bit (offset 78h[6]) is Set.	1, 9, 17	RWS	Yes	80h
23:16		2, 10, 18	RWS	Yes	80h
31:24		3, 11, 19	RWS	Yes	80h

**Register 13-226. B8Ch Asynchronous Advertised N\_FTS**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
This register advertises the Number of Fast Training Sets (N_FTS) values for asynchronous clocking.					
<i>Notes: The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name.   <b>Table 13-5</b> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.   The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i>					
7:0	<b>Port x Asynchronous Tx N_FTS</b>	0, 8, 16	RWS	Yes	A0h
15:8	When clocking is asynchronous, specifies the <i>N_FTS</i> field value in Training Sets transmitted by the Port, when the Port's <b>Link Control</b> register <i>Common Clock Configuration</i> bit (offset 78h[6]) is Cleared.	1, 9, 17	RWS	Yes	A0h
23:16		2, 10, 18	RWS	Yes	A0h
31:24		3, 11, 19	RWS	Yes	A0h

**Register 13-227. B94h PIPE Transmit Parameter Control**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
2:0	<b>Port 0, 8, 16 Tx Margin</b> Applied when TxMargin Override0 = 1.	RWS	Yes	000b
3	<b>Port 0, 8, 16 Tx Swing</b>	RWS	Yes	0
6:4	<b>Port 1, 9, 17 Tx Margin</b> Applied when TxMargin Override1 = 1.	RWS	Yes	000b
7	<b>Port 1, 9, 17 Tx Swing</b>	RWS	Yes	0
10:8	<b>Port 2, 10, 18 Tx Margin</b> Applied when TxMargin Override2 = 1.	RWS	Yes	000b
11	<b>Port 2, 10, 18 Tx Swing</b>	RWS	Yes	0
14:12	<b>Port 3, 11, 19 Tx Margin</b> Applied when TxMargin Override3 = 1.	RWS	Yes	000b
15	<b>Port 3, 11, 19 Tx Swing</b>	RWS	Yes	0
23:16	<b>Factory Test Only</b>	RsvdP	No	00h
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 13-228. B98h 2.5 GT/s -3.5 dB Transmit Parameters**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
2:0	<b>TX_PREDRV_DLY</b> Pre-drive delay.	RWS	Yes	000b
5:3	<b>TXEQ_D2</b> Transmit equalization.	RWS	Yes	000b
7:6	<b>TX_EQ_E</b> Transmit equalization.	RWS	Yes	00b
10:8	<b>TXEQ_MODE</b> Transmit equalization mode.	RWS	Yes	000b
13:11	<b>TX_ETLEV</b>	RWS	Yes	000b
16:14	<b>TX_D2LEV</b>	RWS	Yes	000b
20:17	<b>TX_LEV</b> Transmit amplitude control.	RWS	Yes	0h
23:21	<b>TX_SR_SET</b>	RWS	Yes	000b
26:24	<b>TX_DTLEV</b> Transmit de-emphasis control.	RWS	Yes	000b
28:27	<b>TX_RT_SET</b>	RWS	Yes	00b
30:29	<i>Reserved</i>	RsvdP	Yes	00b
31	<b>Transmit Parameter Override Enable</b>	RWS	Yes	0

**Register 13-229. B9Ch 5.0 GT/s -6 dB Transmit Parameters**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
2:0	<b>TX_PREDRV_DLY</b> Pre-drive delay.	RWS	Yes	000b
5:3	<b>TXEQ_D2</b> Transmit equalization.	RWS	Yes	000b
7:6	<b>TX_EQ_E</b> Transmit equalization.	RWS	Yes	00b
10:8	<b>TXEQ_MODE</b> Transmit equalization mode.	RWS	Yes	000b
13:11	<b>TX_ETLEV</b>	RWS	Yes	000b
16:14	<b>TX_D2LEV</b>	RWS	Yes	000b
20:17	<b>TX_LEV</b> Transmit amplitude control.	RWS	Yes	0h
23:21	<b>TX_SR_SET</b>	RWS	Yes	000b
26:24	<b>TX_DTLEV</b> Transmit de-emphasis control.	RWS	Yes	000b
28:27	<b>TX_RT_SET</b>	RWS	Yes	00b
30:29	<i>Reserved</i>	RsvdP	Yes	00b
31	<b>Transmit Parameter Override Enable</b>	RWS	Yes	0



**Register 13-230. BA0h 5.0 GT/s -3.5 dB Transmit Parameters**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
2:0	<b>TX_PREDRV_DLY</b> Pre-drive delay.	RWS	Yes	000b
5:3	<b>TXEQ_D2</b> Transmit equalization.	RWS	Yes	000b
7:6	<b>TX_EQ_E</b> Transmit equalization.	RWS	Yes	00b
10:8	<b>TXEQ_MODE</b> Transmit equalization mode.	RWS	Yes	000b
13:11	<b>TX_ETLEV</b>	RWS	Yes	000b
16:14	<b>TX_D2LEV</b>	RWS	Yes	000b
20:17	<b>TX_LEV</b> Transmit amplitude control.	RWS	Yes	0h
23:21	<b>TX_SR_SET</b>	RWS	Yes	000b
26:24	<b>TX_DTLEV</b> Transmit de-emphasis control.	RWS	Yes	000b
28:27	<b>TX_RT_SET</b>	RWS	Yes	00b
30:29	<i>Reserved</i>	RsvdP	Yes	00b
31	<b>Transmit Parameter Override Enable</b>	RWS	Yes	0

**Register 13-231. BA4h 8.0 GT/s 0 dB Transmit Parameters**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
2:0	<b>TX_PREDRV_DLY</b> Pre-drive delay.	RWS	Yes	000b
5:3	<b>TXEQ_D2</b> Transmit equalization.	RWS	Yes	000b
7:6	<b>TX_EQ_E</b> Transmit equalization.	RWS	Yes	00b
10:8	<b>TXEQ_MODE</b> Transmit equalization mode.	RWS	Yes	000b
13:11	<b>TX_ETLEV</b>	RWS	Yes	000b
16:14	<b>TX_D2LEV</b>	RWS	Yes	000b
20:17	<b>TX_LEV</b> Transmit amplitude control.	RWS	Yes	0h
23:21	<b>TX_SR_SET</b>	RWS	Yes	000b
26:24	<b>TX_DTLEV</b> Transmit de-emphasis control.	RWS	Yes	000b
28:27	<b>TX_RT_SET</b>	RWS	Yes	00b
30:29	<i>Reserved</i>	RsvdP	Yes	00b
31	<b>Transmit Parameter Override Enable</b>	RWS	Yes	0

**Register 13-232. BB8h DC Balance Control**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
5:0	Port DC Balance Enable	RWS	Yes	3Fh
7:6	Reserved	RsvdP	No	00b
15:8	DC Balance Symbol 14 Threshold	RWS	Yes	20h
23:16	DC Balance Symbol 15 Threshold	RWS	Yes	10h
31:24	Reserved	RsvdP	No	00h

**Register 13-233. BBCh Trained Coefficient Status**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
5:0	Pre-Cursor Coefficient	ROS	No	0-0h
7:6	Reserved	RsvdP	No	00b
13:8	Main Cursor Coefficient	ROS	No	0-0h
15:14	Reserved	RsvdP	No	00b
21:16	Post-Cursor Coefficient	ROS	No	0-0h
23:22	Reserved	RsvdP	No	00b
24	Near-End/Far-End Select	RWS	Yes	0
27:25	Reserved	RsvdP	No	000b
31:28	Lane Select	RWS	Yes	0h

**Register 13-234. BC0h Port Accumulation Control**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> <i>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
1:0	<b>Lock Down Mode</b> Selects the Station's Port 0 Lock Down mode. Bit 0 determines the final Port configuration. Bit 1 enables/disables PIPE Reversal mode.	RWS	Yes	00b
2	<b>Lock Down Mode Enable</b> Enables the Station's Port 0 Lock Down mode when Autonomous Port Accumulation mode is also enabled (bit 7 ( <i>Autonomous Port Accumulation Mode Enable</i> ) is Set).  0 = Disables Lock Down mode 1 = Enables Lock Down mode	RWS	Yes	0
6:3	<b>Reserved</b>	RsvdP	No	0h
7	<b>Autonomous Port Accumulation Mode Enable</b> 0 = Disables Autonomous Port Accumulation mode 1 = Enables Autonomous Port Accumulation mode	RWS	Yes (Serial EEPROM only)	0
10:8	<b>Auto Port Configuration</b> Returns the current Port configuration selected by the Port Accumulation logic.  <i>Note: Refer to Table 13-5 for the Port Configurations associated with these values.</i>  001b = 1 010b = 2 011b = 3 100b = 4  All other encodings are <b>Reserved</b> .	ROS	No	PCFG
11	<b>PIPE Reverse</b> Returns the PIPE Reversal mode selected by the Port Accumulation logic.  0 = PIPE Reversal mode is disabled (bit 1 of field [1:0] ( <i>Lock Down Mode</i> ) is Cleared) 1 = PIPE Reversal mode is enabled (bit 1 of field [1:0] ( <i>Lock Down Mode</i> ) is Set)	ROS	No	PCFG
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

**Register 13-235. BC4h Recovery Diagnostic**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes:</i> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>				
15:0	<b>Recovery Counter</b> Quantity of times that the Port entered the <i>Recovery</i> state since the Counter was last Cleared.	ROS	No	PCFG
23:16	<b>Reserved</b>	RsvdP	No	00h
26:24	<b>Port Select</b> Selects the Port for which to return the Recovery Count, or the Port's Recovery Counter to Clear.	RW	Yes	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>Clear Recovery Count</b> Writing 1 to this bit Clears the Port's Recovery Counter.	RZ	Yes	0

**Register 13-236. BC8h User Lane Equalization Control**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes:</i> The <b>Port x</b> column is provided to indicate the Port Number associated with fields that include “Port x” in their name.</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
7:0	<b>User Lane Equalization Control Data for Port x</b> These fields allow MemWr commands (rather than the serial EEPROM) to load the <b>Lane x Equalization Control</b> register(s) (offset(s) 118h through 134h) if this register is written. When this register is used to load the <b>Lane x Equalization Control</b> register(s), all Lanes that belong to the same Port have the same Settings.	0, 8, 16	RWS	Yes	00h
15:8		1, 9, 17	RWS	Yes	00h
23:16		2, 10, 18	RWS	Yes	00h
31:24		3, 11, 19	RWS	Yes	00h

**Register 13-237. BD4h Gen 3 Coefficient Values**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i>					
<i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i>					
17:0	Preset 0 Coefficients		RWS	Yes	1_0BC0h
	Bit(s)	Description			
	5:0	Preshoot			
	11:6	Main			
	17:12	De-Emphasis			
23:18	Gen 3 FS Value		RWS	Yes	3Fh
27:24	Preset Select		RWS	Yes	0h
29:28	Factory Test Only		RWS	Yes	00b
30	Reserved		RsvdP	No	0
31	Coefficient Write Enable Control Must be Set to enable modification of bits [23:0].		RWS	Yes	0

**Register 13-238. BD8h Equalization Control**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name. <a href="#">Table 13-5</a> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
0	<b>Port x Bypass Equalization</b> Directs Downstream Ports to bypass the Transmitter Equalization phases.	0, 8, 16	RWS	Yes	0
1		1, 9, 17	RWS	Yes	0
2		2, 10, 18	RWS	Yes	0
3		3, 11, 19	RWS	Yes	0
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>Set start_equalization_w_preset</b> Directs Downstream Ports to Set the start_equalization_w_preset variable to 1 at the appropriate time.	0, 8, 16	RWS	Yes	0
9		1, 9, 17	RWS	Yes	0
10		2, 10, 18	RWS	Yes	0
11		3, 11, 19	RWS	Yes	0
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b

**Register 13-238. BD8h Equalization Control**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port) (*Cont.*)

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Reset EIEOS Counter</b> Directs Ports to Set the <i>Reset EIEOS Counter</i> bit in the TS1 Ordered-Sets transmitted during its Equalization phase.	0, 8, 16	RWS	Yes	0
17		1, 9, 17	RWS	Yes	0
18		2, 10, 18	RWS	Yes	0
19		3, 11, 19	RWS	Yes	0
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
23:22	<b>Reserved</b>		RsvdP	No	00b
26:24	<b>Receiver Evaluation Period</b> Selects the length of time to use for the Receiver Evaluation period. <b>Silicon Revision BA</b> 000b = 1 $\mu$ s 001b = 2 $\mu$ s 010b = 4 $\mu$ s 011b = 8 $\mu$ s 100b = 26 $\mu$ s (default) 101b = 32 $\mu$ s 110b = 64 $\mu$ s 111b = 128 $\mu$ s <b>Silicon Revision CA</b> 000b = 4 $\mu$ s 001b = 8 $\mu$ s 010b = 16 $\mu$ s 011b = 32 $\mu$ s 100b = 64 $\mu$ s (default) 101b = 128 $\mu$ s 110b = 256 $\mu$ s 111b = 512 $\mu$ s		RWS	Yes	100b
31:27	<b>Receiver Evaluation Threshold</b> Selects the maximum quantity of Receiver Evaluation periods per Transmitter Tuning phase.		RWS	Yes	19h



**Register 13-239. BDCh Signal Detect Level**

**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>This register programs the Signal Detect Level for 2.5 and 8.0 GT/s; writes to this register do not affect the Signal Detect Level for 5.0 GT/s. The Signal Detect Level for individual Lanes and/or SerDes quads can be programmed, independently, for 2.5, 5.0 and 8.0 GT/s, through the <b>AHB Access Control</b> register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BFCh).</p> <p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
2:0	<b>Port 0, 8, 16 Signal Detect Level</b>	RWS	Yes	010b
3	<i>Reserved</i>	RsvdP	No	0
6:4	<b>Port 1, 9, 17 Signal Detect Level</b>	RWS	Yes	010b
7	<i>Reserved</i>	RsvdP	No	0
10:8	<b>Port 2, 10, 18 Signal Detect Level</b>	RWS	Yes	010b
11	<i>Reserved</i>	RsvdP	No	0
14:12	<b>Port 3, 11, 19 Signal Detect Level</b>	RWS	Yes	010b
22:15	<i>Factory Test Only/Reserved</i>	RsvdP	No	0-0h
31:23	<i>Reserved</i>	RsvdP	No	0-0h

**Register 13-240. BE0h Gen 3 LFSR Seed**

(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes.</p> <p>The Port 0 bits are for Lanes [0-15]. The Port 8 bits are for Lanes [16-31]. The Port 16 bits are for Lanes [32-47].</p>				
22:0	<b>LFSR Seed</b> Linear Feedback Shift register (LFSR) seed. Used as the Lane-specific scrambler seed for the Gen 3 scrambler/de-scrambler.	RWS	Yes	1D_BFBCh
23	<b>Reserved</b>	RsvdP	No	0
26:24	<b>Lane Select</b> Specifies which of eight register banks in field [22:0] ( <i>LFSR Seed</i> ) is to be accessed. (Lanes are by Station.)  000b = Lanes 0, 8 (default) 001b = Lanes 1, 9 010b = Lanes 2, 10 011b = Lanes 3, 11 100b = Lanes 4, 12 101b = Lanes 5, 13 110b = Lanes 6, 14 111b = Lanes 7, 15	RWS	Yes	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>LFSR Seed Write Enable Control</b> Must be Set to enable modification of field [22:0] ( <i>LFSR Seed</i> ).	RWS	Yes	0

**Register 13-241. BECh TLP Round Trip Timer Control**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><i>Notes: Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</i></p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
0	<b>TRT Enable</b> Enable TLP Round Trip (TRT) Timer. 0 = TLP Round Trip Timer is disabled 1 = TLP Round Trip Timer is enabled	RW	Yes	0
1	<b>TRT Clear</b> Always returns a value of 0 when read. 1 = Clear minimum and maximum registers	RW	Yes	0
2	<b>TRT Read Maximum</b> 0 = Return minimum time in field [31:16] ( <i>TRT Time</i> ) 1 = Return maximum time in field [31:16] ( <i>TRT Time</i> )	RW	Yes	0
3	<b>TRT Active</b> 1 = TRT Timer is active, time update is pending	RO	No	0
6:4	<b>TRT Port Select</b>	RW	Yes	000b
7	<i>Reserved</i>	RsvdP	No	0
14:8	<b>TRT TLP Type and Format Select</b>		RWS	0-0h
	<b>Bit(s)</b>	<b>Description/Function</b>		
	12:8	<b>Non-Posted TLP Type</b>		
	14:13	<b>Non-Posted TLP Format</b>		
15	<b>TRT Ignore Format Field</b> 1 = Detect egress TLPs, by matching only bits [12:8] ( <i>Non-Posted TLP Type</i> ) of field [14:8]	RW	Yes	0
31:16	<b>TRT Time</b> Returns the minimum or maximum round trip time, depending upon the bit 2 ( <i>TRT Read Maximum</i> ) state.	RO	No	FFFFh

**Register 13-242. BF0h Port Receiver Error Counter**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16,**  
**accessible through the Management Port)**

Bit(s)	Description	Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The <b>Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name. <a href="#">Table 13-5</a> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
7:0	<b>Port x Receiver Error Counter</b>	0, 8, 16	RW1C	No	00h
15:8	When read, returns the quantity of Receiver errors that the Port detected (Receiver Error Counter). The Error Counter saturates at 255.  The Counter is Cleared with any Write to the corresponding byte in this register; otherwise, this field is RO.	1, 9, 17	RW1C	No	00h
23:16		2, 10, 18	RW1C	No	00h
31:24		3, 11, 19	RW1C	No	00h

**Register 13-243. BFCh AHB Access Control****(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>In addition to the functions defined for each bit/field, this register can also be used to program the Signal Detect Level, per Lane or SerDes Quad, independently, for each of the three data rates (2.5, 5.0, and 8.0 GT/s), by Station. (Refer to <a href="#">Table 13-38</a>.) Signal Detect Level is defined by 3-bit values (48 fields, per Station). Because the AHB Data window is 8 bits wide, a Read/Modify/Write sequence is required, to preserve the values of other bits within the register. The 3-bit field for 5.0 GT/s exists within one 8-bit window; therefore, only one Read/Modify/Write sequence is needed. However the 3-bit fields for 2.5 and 8.0 GT/s are each split between two offsets; therefore, two Read/Modify/Write sequences are required to modify a single 3-bit field.</p> <p><i>For example, to program SerDes Quad 0 to use value 3 during 8.0 GT/S, software performs the following actions:</i></p> <ol style="list-style-type: none"> <li>1. Writes the value 000E_5800h to offset BFCh.</li> <li>2. Reads the register (to preserve bits [7:1]).</li> <li>3. Clears bit 0 (msb of 3-bit Signal Detect value).</li> <li>4. ORs the field [7:0] value with 008E_5800h.</li> <li>5. Writes the result back to offset BFCh.</li> <li>6. Writes the value 000E_5700h to offset BFCh.</li> <li>7. Reads the register (to preserve bits [5:0]).</li> <li>8. Changes the value of bits [7:6] to 10b (lsb of Signal Detect value).</li> <li>9. ORs the field [7:0] value with 008E_5700h.</li> <li>10. Writes the result back to offset BFCh.</li> </ol> <p><b>Notes:</b> <a href="#">Table 13-5</a> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p><i>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</i></p>				
7:0	<b>AHB Data</b> When an <b>AHB</b> register is being written, the data in this field is the Write data. When an <b>AHB</b> register is read, the response (Read) data is returned in this field.	RWS	Yes	00h
20:8	<b>AHB Address</b> Specifies the address of the <b>AHB</b> register being accessed.	RWS	Yes	0-0h
22:21	<b>AHB Quad Select</b> Specifies the SerDes quad to which the AHB access is directed.	RWS	Yes	00b
23	<b>AHB -Read/+Write</b> Specifies the AHB access type. 0 = Read 1 = Write	RWS	Yes	0
29:24	<b>Reserved</b>	RsvdP	No	0-0h
30	<b>AHB Access Timeout</b> 1 = An AHB register access did not return AHB_Ready	RW1C	No	0
31	<b>AHB Ready</b> 0 = Indicates that the AHB Controller is processing the previous access or client accesses 1 = Indicates that the AHB Controller is ready to accept a new access	ROS	No	1

**Table 13-38. Signal Level Detect Read/Modify/Write Values for AHB Access Control Register**  
**(Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible**  
**through the Management Port, offset BFCh)<sup>a</sup>**

Quad	Lane(s)	Gen 1 (2.5 GT/s)						Gen 2 (5.0 GT/s)			Gen 3 (8.0 GT/s)					
		Read/Modify/Write			Read/Modify/Write			Read/Modify/Write			Read/Modify/Write			Read/Modify/Write		
		Write (bit 23 = 0)	Read (bit 23 = 1) [7:1]	Write (bit 23 = 1) OR [7:1] OR	Write (bit 23 = 0)	Read (bit 23 = 1) [5:0]	Write (bit 23 = 1) OR [5:0] OR	Write (bit 23 = 0)	Read (bit 23 = 1) [4:0]	Write (bit 23 = 1) OR [4:0] OR	Write (bit 23 = 0)	Read (bit 23 = 1) [7:1]	Write (bit 23 = 1) OR [7:1] OR	Write (bit 23 = 0)	Read (bit 23 = 1) [5:0]	Write (bit 23 = 1) OR [5:0] OR
				[0] = msb)			[7:6] = lsb)			(bits [7:5])			[0] = msb)			[7:6] = lsb)
0	(All 4)	000E1300h		008E13_h	000E1200h		008E12_h	000E4F00h		008E4F_h	000E5800h		008E58_h	000E5700h		008E57_h
	0	00021300h		008213_h	00021200h		008212_h	00024F00h		00824F_h	00025800h		008258_h	00025700h		008257_h
	1	00041300h		008413_h	00041200h		008412_h	00044F00h		00844F_h	00045800h		008458_h	00045700h		008457_h
	2	00061300h		008613_h	00061200h		008612_h	00064F00h		00864F_h	00065800h		008658_h	00065700h		008657_h
	3	00081300h		008813_h	00081200h		008812_h	00084F00h		00884F_h	00085800h		008858_h	00085700h		008857_h
1	(All 4)	002E1300h		00AE13_h	002E1200h		00AE12_h	002E4F00h		00AE4F_h	002E5800h		00AE58_h	002E5700h		00AE57_h
	4	00221300h		00A213_h	00221200h		00A212_h	00224F00h		00A24F_h	00225800h		00A258_h	00225700h		00A257_h
	5	00241300h		00A413_h	00241200h		00A412_h	00244F00h		00A44F_h	00245800h		00A458_h	00245700h		00A457_h
	6	00261300h		00A613_h	00261200h		00A612_h	00264F00h		00A64F_h	00265800h		00A658_h	00265700h		00A657_h
	7	00281300h		00A813_h	00281200h		00A812_h	00284F00h		00A84F_h	00285800h		00A858_h	00285700h		00A857_h
2	(All 4)	004E1300h		00CE13_h	004E1200h		00CE12_h	004E4F00h		00CE4F_h	004E5800h		00CE58_h	004E5700h		00CE57_h
	8	00421300h		00C213_h	00421200h		00C212_h	00424F00h		00C24F_h	00425800h		00C258_h	00425700h		00C257_h
	9	00441300h		00C413_h	00441200h		00C412_h	00444F00h		00C44F_h	00445800h		00C458_h	00445700h		00C457_h
	10	00461300h		00C613_h	00461200h		00C612_h	00464F00h		00C64F_h	00465800h		00C658_h	00465700h		00C657_h
	11	00481300h		00C813_h	00481200h		00C812_h	00484F00h		00C84F_h	00485800h		00C858_h	00485700h		00C857_h
3	(All 4)	006E1300h		00EE13_h	006E1200h		00EE12_h	006E4F00h		00EE4F_h	006E5800h		00EE58_h	006E5700h		00EE57_h
	12	00621300h		00E213_h	00621200h		00E212_h	00624F00h		00E24F_h	00625800h		00E258_h	00625700h		00E257_h
	13	00641300h		00E413_h	00641200h		00E412_h	00644F00h		00E44F_h	00645800h		00E458_h	00645700h		00E457_h
	14	00661300h		00E613_h	00661200h		00E612_h	00664F00h		00E64F_h	00665800h		00E658_h	00665700h		00E657_h
	15	00681300h		00E813_h	00681200h		00E812_h	00684F00h		00E84F_h	00685800h		00E858_h	00685700h		00E857_h

a. Throughout this data book, 32-bit hex values usually include an underscore character (\_), separating the combined 16-bit values. In this table, however, that underscore is not used, to avoid confusion with the underscores used for placeholder values.

### 13.18.3 Device-Specific Registers – Requester ID Read Back (Offsets C8Ch – C90h)

This section details the Device-Specific Requester ID Read back register. [Table 13-39](#) defines the register map.

**Table 13-39. Device-Specific Requester ID Register Map (Offsets C8Ch – C90h) (All Ports)**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
<i>Reserved</i>																																C8Ch
<i>Reserved</i> (Upstream)																Requester ID Read Back (Upstream)																C90h
<i>Reserved</i> (Downstream)																																

**Register 13-244. C90h Requester ID Read Back (All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Requester ID Read Back</b> Memory Read to this location returns the Requester ID of that TLP.  Determines the Requester ID values to program into the NT Port Requester ID Lookup Tables (NT Port Link Interface, offsets <a href="#">DB4h</a> to <a href="#">DF0h</a> , and NT Port Virtual Interface, offsets <a href="#">D94h</a> to <a href="#">DD0h</a> ), in the Station that contains the NT Port.	Upstream	ROS	No	0000h
	<i>Reserved</i>	Downstream	RsvdP	No	0000h
31:16	<i>Reserved</i>		RsvdP	No	0000h

### 13.19 Multicast Extended Capability Registers (Offsets E00h – E2Ch)

This section details the Multicast Extended Capability registers. [Table 13-40](#) defines the register map. Multicast is described, in detail, in [Section 8.6, “Multicast.”](#)

**Table 13-40. Multicast Extended Capability Register Map (All Ports)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																				
Next Capability Offset (B00h or F24h)																Capability Version (1h)				PCI Express Extended Capability ID (0012h)																E00h
Multicast Control																Multicast Extended Capability																E04h				
Multicast BAR0																																E08h				
Multicast BAR1																																E0Ch				
Multicast Receive 0																																E10h				
Multicast Receive 1																																E14h				
Multicast Block All 0																																E18h				
Multicast Block All 1																																E1Ch				
Multicast Block Untranslated 0																																E20h				
Multicast Block Untranslated 1																																E24h				
Multicast Overlay BAR0																																E28h				
Multicast Overlay BAR1																																E2Ch				



**Register 13-245. E00h Multicast Extended Capability Header  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>PCI Express Extended Capability ID</b> Program to 0012h, to indicate that the Extended Capability structure is the <b>Multicast Extended Capability</b> structure.		ROS	Yes	0012h
19:16	<b>Capability Version</b>		ROS	Yes	1h
31:20	<b>Next Capability Offset</b> Next extended capability is the <b>LTR Extended Capability</b> structure, offset B00h.	Upstream	ROS	Yes	B00h
	Next extended capability is the <b>ACS Extended Capability</b> structure, offset F24h.	Downstream	ROS	Yes	F24h

**Register 13-246. E04h Multicast Extended Capability and Control  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Multicast Extended Capability</b>				
5:0	<b>MC Max Group</b> 00h = Indicates that one Multicast Group is supported 3Fh = Indicates the maximum quantity of Multicast Groups that the component supports, encoded as $M-1$	ROS	Yes	3Fh
14:6	<i>Reserved</i>	RsvdP	No	0-0h
15	<b>MC ECRC Generation Supported</b> 0 = ECRC generation is not supported in Multicast 1 = ECRC generation is supported in Multicast	ROS	Yes	1
<b>Multicast Control</b>				
21:16	<b>MC Num Group</b> Indicates the quantity of Multicast Groups configured for use, encoded as $N-1$ . The behavior of this field is undefined if its value exceeds the value indicated by field [5:0] ( <i>MC Max Group</i> ). This parameter indirectly defines the upper limit of the Multicast Address range. This field is ignored if bit 31 ( <i>MC Enable</i> ) is Cleared. 00h = Indicates that one Multicast Group is configured for use	RW	Yes	0-0h
30:22	<i>Reserved</i>	RsvdP	No	0-0h
31	<b>MC Enable</b> 0 = Disables Multicast 1 = Enables Multicast	RW	Yes	0

**Register 13-247. E08h Multicast BAR0  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
5:0	<b>MC Index Position</b> The Multicast Group Number LSB location within the address. The behavior of this field is undefined if its value is less than 12 (Ch) and the <b>Multicast Control</b> register <i>MC Enable</i> bit (offset E04h[31]) is Set.	RW	Yes	0-0h
11:6	<b>Reserved</b>	RsvdP	No	0-0h
31:12	<b>MC Base Address</b> Multicast Lower Base Address[31:12]. Base address of the Multicast Address range. The behavior is undefined if: <ul style="list-style-type: none"> <li><b>Multicast Control</b> register <i>MC Enable</i> bit (offset E04h[31]) is Set, and</li> <li>Bits in this field corresponding to Address bits that contain the Multicast Group number, or Address bits less than the value indicated by field [5:0] (<i>MC Index Position</i>), are non-zero.</li> </ul>	RW	Yes	0000_0h

**Register 13-248. E0Ch Multicast BAR1  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>MC Upper Base Address</b> Multicast Upper Base Address[63:32]. Base address of the Multicast Address range. The behavior is undefined if: <ul style="list-style-type: none"> <li><b>Multicast Control</b> register <i>MC Enable</i> bit (offset E04h[31]) is Set, and</li> <li>Bits in this field corresponding to Address bits that contain the Multicast Group number, or Address bits less than the value indicated by the <b>Multicast BAR0</b> register <i>MC Index Position</i> field (offset E04h[5:0]), are non-zero.</li> </ul>	RW	Yes	0000_0000h

**Register 13-249. E10h Multicast Receive 0  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>MC Receive</b> Multicast Receive[31:0]. Provides a bit Vector that denotes to which Multicast Groups the Port should forward Multicast TLPs. For each bit that is Set, this Port receives a copy of any Multicast TLPs that exist for the associated Multicast Group. Bits above MC Num Group ( <b>Multicast Extended Capability</b> register <i>MC Num Group</i> field, offset E04h[21:16]) are ignored by hardware.	RW	Yes	0000_0000h

**Register 13-250. E14h Multicast Receive 1  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>MC Receive</b> Multicast Receive[63:32]. Provides a bit Vector that denotes to which Multicast Groups the Port should forward Multicast TLPs. For each bit that is Set, this Port receives a copy of any Multicast TLPs that exist for the associated Multicast Group. Bits above MC Num Group ( <b>Multicast Extended Capability</b> register <i>MC Num Group</i> field, offset E04h[21:16]) are ignored by hardware.	RW	Yes	0000_0000h

**Register 13-251. E18h Multicast Block All 0  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>MC Block All</b> Multicast Block All[31:0]. Provides a bit Vector that denotes which Multicast Groups the Multicast function should block. For each bit that is Set, this Port is blocked from sending TLPs to the associated Multicast Group. Bits above MC Num Group ( <b>Multicast Extended Capability</b> register <i>MC Num Group</i> field, offset E04h[21:16]) are ignored by hardware.	RW	Yes	0000_0000h

**Register 13-252. E1Ch Multicast Block All 1  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>MC Block All</b> Multicast Block All[63:32]. Provides a bit Vector that denotes which Multicast Groups the Multicast function should block. For each bit that is Set, this Port is blocked from sending TLPs to the associated Multicast Group. Bits above MC Num Group ( <b>Multicast Extended Capability</b> register <i>MC Num Group</i> field, offset E04h[21:16]) are ignored by hardware.	RW	Yes	0000_0000h

**Register 13-253. E20h Multicast Block Untranslated 0  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>MC Block Untranslated</b> Multicast Block Untranslated[31:0]. Used to determine whether a TLP that includes an Untranslated Address should be blocked. For each bit that is Set, this Port is blocked from sending TLPs containing Untranslated addresses to the associated Multicast Group. Bits above MC Num Group ( <b>Multicast Extended Capability</b> register <i>MC Num Group</i> field (offset E04h[21:16])) are ignored by hardware.	RW	Yes	0000_0000h

**Register 13-254. E24h Multicast Block Untranslated 1  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>MC Block Untranslated</b> Multicast Block Untranslated[63:32]. Used to determine whether a TLP that includes an Untranslated Address should be blocked. For each bit that is Set, this Port is blocked from sending TLPs containing Untranslated addresses to the associated Multicast Group. Bits above MC Num Group ( <b>Multicast Extended Capability</b> register <i>MC Num Group</i> field (offset E04h[21:16])) are ignored by hardware.	RW	Yes	0000_0000h

**Register 13-255. E28h Multicast Overlay BAR0  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
Specifies the Base address of a window in Unicast space onto which Multicast TLPs output from an egress Port are overlaid by a process of address replacement. This allows a single BAR in an endpoint attached to the PEX 8748 to be used for both Unicast and Multicast traffic. At the Upstream Port(s), this register allows the Multicast Address range, or a portion of it, to be overlaid onto Host memory.				
5:0	<b>MC Overlay Size</b> Less than 06h = Disables the Overlay mechanism 06h or greater = Specifies the size (in bytes) of the Overlay Address range, as a power of 2	RW	Yes	0-0h
31:6	<b>MC Overlay BAR</b> Multicast Overlay Lower Base Address[31:6]. Specifies the Base address of the window onto which Multicast TLPs passing through the Multicast function will be overlaid.	RW	Yes	0-0h

**Register 13-256. E2Ch Multicast Overlay BAR1  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
Specifies the Base address of a window in Unicast space onto which Multicast TLPs output from an egress Port are overlaid by a process of address replacement. This allows a single BAR in an endpoint attached to the PEX 8748 to be used for both Unicast and Multicast traffic. At the Upstream Port(s), this register allows the Multicast Address range, or a portion of it, to be overlaid onto Host memory.				
31:0	<b>MC Overlay Upper Base Address</b> Multicast Overlay Upper Base Address[63:32]. Specifies the Base address of the window onto which Multicast TLPs passing through the Multicast function will be overlaid.	RW	Yes	0000_0000h

## 13.20 Device-Specific Registers – Virtual Switch (Offsets F00h – F20h), Virtual Switch Mode

**Note:** In Base mode, this entire structure is **Reserved**, RsvdP, not serial EEPROM nor I<sup>2</sup>C writable, and has a default value of 0h.

This section details Device-Specific virtual switch registers, located at offsets F00h through F20h. Table 13-41 defines the register map.

Other Device-Specific Virtual Switch registers are detailed in Section 13.16.14, “Device-Specific Registers – Virtual Switch (Offsets 900h – 9E8h), Virtual Switch Mode.”

Other Device-Specific registers are detailed in:

- Section 13.16, “Device-Specific Registers (Offsets 1C0h – A74h)”
- Section 13.18, “Device-Specific Registers (Offsets B70h – DFCh)”
- Section 13.22, “Device-Specific Registers (Offsets F30h – FB0h)”

**Table 13-41. Device-Specific Virtual Switch Register Map (Offsets F00h – F20h) (Virtual Switch mode – VS Upstream Port(s))**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
VS Failover		F00h
Reserved		F04h – F1Ch
Port Cut-Thru Enable Status		F20h

**Register 13-257. F00h VS Failover**  
**(Virtual Switch mode – VS Upstream Port(s))**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
Provides failover support for virtual switches. The following values indicate the Port associated with fields [12:8 and 4:0]: 0_0000b = Port 0 0_0001b = Port 1 0_0010b = Port 2 0_0011b = Port 3 0_1000b = Port 8 0_1001b = Port 9 0_1010b = Port 10 0_1011b = Port 11 1_0000b = Port 16 1_0001b = Port 17 1_0010b = Port 18 1_0011b = Port 19 All other encodings are <i>Reserved</i> .				
4:0	<b>VS Upstream Port</b>	RW	Yes	PCFG
7:5	<i>Reserved</i>	RsvdP	No	000b
12:8	<b>VS NT Port</b>	RW	Yes	PCFG
13	<b>VS NT Enable</b>	RW	Yes	PCFG
31:14	<i>Reserved</i>	RsvdP	No	0-0h



**Register 13-258. F20h Port Cut-Thru Enable Status  
(Virtual Switch mode – VS Upstream Port(s))**

Bit(s)	Description	Upstream Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The Downstream Port bits are <b>Reserved</b>, RsvdP, not serial EEPROM nor I<sup>2</sup>C writable, and have a default value of 0.</p> <p>The <b>Upstream Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name.</p> <p><i>Table 13-5</i> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p>					
0	<b>Port x Cut-Thru Enable Status</b> Link Up status. 0 = Cut-Thru is disabled for this Port (Link is Down) 1 = Cut-Thru is enabled for this Port (Link is Up)	0	RO	No	Defined by STRAP_STN0_PORTCFG[1:0] input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station 0</i> field (Port 0, accessible through the Management Port, offset 300h[2:0])
1		1	RO	No	
2		2	RO	No	
3		3	RO	No	
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>Port x Cut-Thru Enable Status</b> Link Up status. 0 = Cut-Thru is disabled for this Port (Link is Down) 1 = Cut-Thru is enabled for this Port (Link is Up)	8	RO	No	Defined by STRAP_STN1_PORTCFG[1:0] input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station 1</i> field (Port 0, accessible through the Management Port, offset 300h[5:3])
9		9	RO	No	
10		10	RO	No	
11		11	RO	No	
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b
16	<b>Port x Cut-Thru Enable Status</b> Link Up status. 0 = Cut-Thru is disabled for this Port (Link is Down) 1 = Cut-Thru is enabled for this Port (Link is Up)	16	RO	No	Defined by STRAP_STN2_PORTCFG[1:0] input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station 2</i> field (Port 0, accessible through the Management Port, offset 300h[8:6])
17		17	RO	No	
18		18	RO	No	
19		19	RO	No	
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
31:22	<b>Reserved</b>		RsvdP	No	0-0h

## 13.21 ACS Extended Capability Registers (Offsets F24h – F2Ch)

This section details the ACS Extended Capability registers. [Table 13-42](#) defines the register map.

**Table 13-42. ACS Extended Capability Register Map  
(Downstream Ports)**

31 30 29 28 27 26 25 24 23 22 21 20	19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
<b>Reserved</b> (Upstream)			
Next Capability Offset (B70h) (Downstream)	Capability Version (1h) (Downstream)	PCI Express Extended Capability ID (000Dh) (Downstream)	F24h
<b>Reserved</b> (Upstream)			
ACS Control (Downstream)		ACS Capability (Downstream)	F28h
<b>Reserved</b> (Upstream) Egress Control Vector (Downstream)			
			F2Ch

**Register 13-259. F24h ACS Extended Capability Header  
(Downstream Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> Because this register is implemented as one physical register common to all Ports, the Upstream Port(s)' register (which is <b>Reserved</b> ) has the same value as the Downstream Ports' registers. However, in the Upstream Port(s), the ACS Extended Capability is excluded from the Linked List of PCI Express Extended Capabilities, and therefore, the Upstream Port(s)' register is effectively hidden from system software and the non-zero value has no significant consequence.					
15:0	<b>Reserved</b>	Upstream	RsvdP	No	0000h
	<b>PCI Express Extended Capability ID</b>	Downstream	ROS	Yes	000Dh
19:16	<b>Reserved</b>	Upstream	RsvdP	No	0h
	<b>Capability Version</b>	Downstream	ROS	Yes	1h
31:20	<b>Reserved</b>	Upstream	RsvdP	No	000h
	<b>Next Capability Offset</b> Program to B70h, which addresses the <b>Vendor-Specific Extended Capability 2</b> structure.	Downstream	ROS	Yes	<b>B70h</b>

**Register 13-260. F28h ACS Control and Capability  
(Downstream Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Capability</b>					
0	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ACS Source Validation</b>	Downstream	RO	Yes	1
1	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking</b>	Downstream	RO	Yes	1
2	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect</b> ACS Peer-to-Peer Request redirect.	Downstream	RO	Yes	1
3	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect</b> ACS Peer-to-Peer Completion redirect.	Downstream	RO	Yes	1
4	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding</b>	Downstream	RO	Yes	1
5	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control</b> ACS Peer-to-Peer Egress control.	Downstream	RO	Yes	1
6	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P</b> ACS Direct Translated Peer-to-Peer.	Downstream	RO	Yes	1
7	<i>Reserved</i>		RsvdP	No	0
12:8	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>Egress Control Vector Size</b> Encodings 01h through FFh directly indicate the number of each Port's <b>Egress Control Vector</b> register <i>Peer-to-Peer Port x Control</i> bit (offset F2Ch[19:16, 11:8, 3:0]).  <i>Note: The ACS Egress Control Vector Size value must be adjusted for the specific Port configuration.</i>	Downstream	HwInit	Yes	18h
15:13	<i>Reserved</i>		RsvdP	No	000b

**Register 13-260. F28h ACS Control and Capability  
(Downstream Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>ACS Control</b>					
16	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ACS Source Validation Enable</b> 0 = Disables 1 = Enables	Downstream	RW	Yes	0
17	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ACS Translation Blocking Enable</b> 0 = Disables 1 = Enables	Downstream	RW	Yes	0
18	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ACS P2P Request Redirect Enable</b> Enables or disables ACS Peer-to-Peer Request redirect. 0 = Disables 1 = Enables	Downstream	RW	Yes	0
19	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ACS P2P Completion Redirect Enable</b> Enables or disables ACS Peer-to-Peer Completion redirect. 0 = Disables 1 = Enables	Downstream	RW	Yes	0
20	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ACS Upstream Forwarding Enable</b> 0 = Disables 1 = Enables	Downstream	RW	Yes	0
21	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ACS P2P Egress Control Enable</b> Enables or disables ACS Peer-to-Peer Egress control. 0 = Disables 1 = Enables	Downstream	RW	Yes	0
22	<i>Reserved</i>	Upstream	RsvdP	No	0
	<b>ACS Direct Translated P2P Enable</b> Enables or disables ACS Direct Translated Peer-to-Peer. 0 = Disables 1 = Enables	Downstream	RW	Yes	0
31:23	<i>Reserved</i>		RsvdP	No	0-0h

**Register 13-261. F2Ch Egress Control Vector  
(Downstream Ports)**

Bit(s)	Description	Downstream Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The Upstream Port bits are <b>Reserved</b>, RsvdP, not serial EEPROM nor I<sup>2</sup>C writable, and have a default value of 0. The <b>Downstream Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name.</p> <p><i>Table 13-5</i> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming.</p> <p>The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
0	<b>Peer-to-Peer Port x Control</b> Valid when the <b>ACS Control</b> register <i>ACS P2P Egress Control Enable</i> bit (Downstream Ports, offset F28h[21]) is Set. 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	0	RW	Yes	0
1		1	RW	Yes	0
2		2	RW	Yes	0
3		3	RW	Yes	0
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>Peer-to-Peer Port x Control</b> Valid when the <b>ACS Control</b> register <i>ACS P2P Egress Control Enable</i> bit (Downstream Ports, offset F28h[21]) is Set. 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	8	RW	Yes	0
9		9	RW	Yes	0
10		10	RW	Yes	0
11		11	RW	Yes	0
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b
16	<b>Peer-to-Peer Port x Control</b> Valid when the <b>ACS Control</b> register <i>ACS P2P Egress Control Enable</i> bit (Downstream Ports, offset F28h[21]) is Set. 0 = No Peer-to-Peer control 1 = ACS Peer-to-Peer control	16	RW	Yes	0
17		17	RW	Yes	0
18		18	RW	Yes	0
19		19	RW	Yes	0
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
31:22	<b>Reserved</b>		RsvdP	No	0-0h

# 13.22 Device-Specific Registers (Offsets F30h – FB0h)

This section details the Device-Specific registers located at offsets F30h through FB0h. Device-Specific registers are unique to the PEX 8748 and not referenced in the *PCI Express Base r3.0*. [Table 13-43](#) defines the register map.

Other Device-Specific registers are detailed in:

- [Section 13.16, “Device-Specific Registers \(Offsets 1C0h – A74h\)”](#)
- [Section 13.18, “Device-Specific Registers \(Offsets B70h – DFCh\)”](#)
- [Section 13.20, “Device-Specific Registers – Virtual Switch \(Offsets F00h – F20h\), Virtual Switch Mode”](#)

*Note: It is recommended that these registers not be changed from their default values.*

**Table 13-43. Device-Specific Register Map (Offsets F30h – FB0h)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
Device-Specific Registers – Egress Control (Offsets F30h – F44h)		F30h ... F44h
Device-Specific Registers – Ingress Control and Port Enable (Offsets F48h – F6Ch)		F48h ... F6Ch
Device-Specific Registers – Error Checking and Debug (Offsets F70h – FB0h)		F70h ... FB0h

### 13.22.1 Device-Specific Registers – Egress Control (Offsets F30h – F44h)

This section details the Device-Specific Egress Control registers. [Table 13-44](#) defines the register map.

**Table 13-44. Device-Specific Egress Control Register Map (All Ports)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
Egress Control and Status		F30h
Reserved		F34h
Port Egress TLP Threshold		F38h
Reserved		F3Ch – F40h
Factory Test Only		F44h

**Register 13-262. F30h Egress Control and Status  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Notes:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports. For bits [12:9, 7, 5] – The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.					
1:0	<b>Reserved</b>		RsvdP	No	00b
4:2	<b>Factory Test Only</b>		RWS	Yes	000b
5	<b>Snoop Enable as Requester ID</b> The <i>Snoop Enable</i> bit must match the incoming NS attribute for LUT.	<b>Base Mode</b> 0, 8, 16  <b>Virtual Switch Mode</b> 0, 8, 16, accessible through the Management Port	RWS	Yes	0
		Otherwise	RsvdP	No	0
6	<b>Factory Test Only</b>		RWS	Yes	0
7	<b>Ex Tag 8 Bit</b> Exerciser Tag is 8 bits	<b>Base Mode</b> 0, 8, 16  <b>Virtual Switch Mode</b> 0, 8, 16, accessible through the Management Port	RWS	Yes	0
		Otherwise	RsvdP	No	0



**Register 13-262. F30h Egress Control and Status**  
**(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Rdr Hdr Prefetch Disable</b> 0 = Prefetch Headers on this Port 1 = Do not prefetch Headers on this Port		RWS	Yes	0
9	<b>Vendor-Defined Type 0 UR</b> 0 = Do not generate UR Vendor-Defined Type 0 Broadcast TLP in <i>DL_Down</i> state 1 = Generate UR Vendor-Defined Type 0 Broadcast TLP in <i>DL_Down</i> state	<b>Base Mode</b> 0, 8, 16  <b>Virtual Switch Mode</b> 0, 8, 16, accessible through the Management Port	RWS	Yes	0
10	<b>Egress Credit Timeout Enable</b> 0 = Egress Credit Timeout mechanism is disabled. 1 = Egress Credit Timeout mechanism is enabled. The timeout period is selected in field [12:11] ( <i>Egress Credit Timeout Value</i> ). Status is reflected in bit 16 ( <i>Egress Credit Timeout Status</i> ). If the Egress Credit Timer is enabled and expires (due to lack of Flow Control credits from the connected device), the Port brings down its Link. This event generates a Surprise Down Uncorrectable error, for Transparent Downstream Ports. For Upstream Port Egress Credit Timeout, the connected Upstream device detects the Surprise Down event.	<b>Base Mode</b> 0, 8, 16  <b>Virtual Switch Mode</b> 0, 8, 16, accessible through the Management Port	RWS	Yes	0
12:11	<b>Egress Credit Timeout Value</b> 00b = 1 ms 01b = 512 ms 10b = 1s 11b = <i>Reserved</i>	<b>Base Mode</b> 0, 8, 16  <b>Virtual Switch Mode</b> 0, 8, 16, accessible through the Management Port	RWS	Yes	00b
15:13	<i>Reserved</i>		RsvdP	No	000b
16	<b>Egress Credit Timeout Status</b> 0 = No timeout 1 = Timeout		RW1CS	No	0
18:17	<b>Egress Credit Timeout VC&amp;T</b> Egress Credit timeout for Virtual Channel and Type. 00b = Posted 01b = Non-Posted 10b = Completion 11b = <i>Reserved</i>		ROS	No	00b
30:19	<i>Reserved</i>		RsvdP	No	0-0h
31	<b>Port Activity</b> 0 = Port is idle 1 = Port has one or more pending TLPs to transmit		ROS	No	0

**Register 13-263. F38h Port Egress TLP Threshold  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<i>Note: Source Queuing and Read Pacing should not be concurrently enabled. The two features are incompatible and enabling both concurrently can result in Fatal errors.</i>					
11:0	<b>Port Lower TLP Counter</b> When Source Scheduling is disabled due to Threshold, it is re-enabled when the Port TLP Counter goes below this Threshold value.	Upstream	RWS	Yes	003h
		Downstream	RWS	Yes	FFFh
15:12	<b>Reserved</b>		RsvdP	No	0h
27:16	<b>Port Upper TLP Counter</b> When the Port TLP Counter is greater than or equal to this value, the Source Scheduler disables TLP scheduling to this egress Port.	Upstream	RWS	Yes	006h
		Downstream	RWS	Yes	FFFh
31:28	<b>Reserved</b>		RsvdP	No	0h

## 13.22.2 Device-Specific Registers – Ingress Control and Port Enable (Offsets F48h – F6Ch)

This section details the Device-Specific Ingress Control and Port Enable registers, which also include the **Negotiated Link Width** registers. Table 13-45 defines the register map.

**Table 13-45. Device-Specific Ingress Control and Port Enable Register Map (Ports<sup>a</sup>)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
Ingress Port-Based Control																															F48h
Port Enable Status																															F4Ch
Reserved								Factory Test Only								Negotiated Link Width for Ports 0, 1, 2, 3															F50h
Reserved								Factory Test Only								Negotiated Link Width for Ports 8, 9, 10, 11															F54h
Reserved								Factory Test Only								Negotiated Link Width for Ports 16, 17, 18, 19															F58h
Minimum/Maximum Status																															F5Ch
Ingress Control																							Factory Test Only/ Reserved/Not Used								F60h
Reserved																															F64h – F6Ch

a. Certain registers are Port-specific, others are Station-specific or Chip-specific; all are Device-specific.

### Register 13-264. F48h Ingress Port-Based Control (Base mode – All Ports; Virtual Switch mode – VS Upstream Port(s))

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>ACK TLP Counter Timeout</b> Sets the quantity of ingress TLP Acknowledges (ACKs) pending, which causes a high-priority ACK to be sent. 00b = 16 TLPs 01b = 8 TLPs 10b = 4 TLPs 11b = Feature is disabled	RWS	Yes	00b
31:2	Reserved	RsvdP	No	0-0h

**Register 13-265. F4Ch Port Enable Status (Upstream Port(s))**

Bit(s)	Description	Upstream Port x	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>The value of this register depends upon the Port configuration for each Station – Station 0 (bits [3:0]), Station 1 (bits [11:8]), and Station 2 (bits [19:16]). When a Port is enabled in the corresponding Station configuration, the bit for that Port is Set; otherwise, the bit is Cleared.</p> <p><b>Notes:</b> The Downstream Port bits are <b>Reserved</b>, RsvdP, not serial EEPROM nor I<sup>2</sup>C writable, and have a default value of 0.</p> <p>The <b>Upstream Port x</b> column is provided to indicate the Port Number associated with bits/fields that include “Port x” in their name.</p> <p><i>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations and Ports.</i></p>					
0	<b>Port x Enable Status</b> 0 = Port is disabled 1 = Port is enabled	0	RO	No	Defined by STRAP_STN0_PORTCFG[1:0] input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station 0</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[2:0])
1		1	RO	No	
2		2	RO	No	
3		3	RO	No	
5:4	<b>Factory Test Only</b>		RsvdP	No	00b
7:6	<b>Reserved</b>		RsvdP	No	00b
8	<b>Port x Enable Status</b> 0 = Port is disabled 1 = Port is enabled	8	RO	No	Defined by STRAP_STN1_PORTCFG[1:0] input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station 1</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[5:3])
9		9	RO	No	
10		10	RO	No	
11		11	RO	No	
13:12	<b>Factory Test Only</b>		RsvdP	No	00b
15:14	<b>Reserved</b>		RsvdP	No	00b
16	<b>Port x Enable Status</b> 0 = Port is disabled 1 = Port is enabled	16	RO	No	Defined by STRAP_STN2_PORTCFG[1:0] input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station 2</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6])
17		17	RO	No	
18		18	RO	No	
19		19	RO	No	
21:20	<b>Factory Test Only</b>		RsvdP	No	00b
23:22	<b>Reserved</b>		RsvdP	No	00b
28:24	<b>Upstream Port of VS port Is in</b>		ROS	No	0h
31:29	<b>Reserved</b>		RsvdP	No	000b

**Register 13-266. F50h Negotiated Link Width for Ports 0, 1, 2, 3 (Upstream Port(s))**

Bit(s)	Description	Upstream Port <i>x</i>	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The Downstream Port bits are <b>Reserved</b>, RsvdP, not serial EEPROM nor I<sup>2</sup>C writable, and have a default value of 0.</p> <p>The <b>Upstream Port <i>x</i></b> column is provided to indicate the Port Number associated with bits/fields that include “Port <i>x</i>” in their name.</p> <p><a href="#">Table 13-5</a> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes.</p>					
3:0	<b>Negotiated Link Width and Link Speed Encoding for Port <i>x</i></b> Remainder of value divided by 5 indicates the negotiated Link width. If the Link is Down, the value is 0h.  0h = x1, or the Port is in the <i>DL_Down</i> state 1h = x2 2h = x4 3h = x8 4h = x16  Quotient of division indicates the Link speed. 0h = Gen 1 (2.5 GT/s) 1h = Gen 2 (5.0 GT/s) 2h = Gen 3 (8.0 GT/s)  All other encodings are <b>Reserved</b> .	0	RO	No	PCFG
7:4		1	RO	No	
11:8		2	RO	No	
15:12		3	RO	No	
23:16	<b>Factory Test Only</b>		RsvdP	No	00h
31:24	<b>Reserved</b>		RsvdP	No	00h

**Register 13-267. F54h Negotiated Link Width for Ports 8, 9, 10, 11 (Upstream Port(s))**

Bit(s)	Description	Upstream Port <i>x</i>	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The Downstream Port bits are <b>Reserved</b>, RsvdP, not serial EEPROM nor I<sup>2</sup>C writable, and have a default value of 0.</p> <p>The <b>Upstream Port <i>x</i></b> column is provided to indicate the Port Number associated with bits/fields that include “Port <i>x</i>” in their name.</p> <p>Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes.</p>					
3:0	<b>Negotiated Link Width and Link Speed Encoding for Port <i>x</i></b> Remainder of value divided by 5 indicates the negotiated Link width. If the Link is Down, the value is 0h.	8	RO	No	PCFG
7:4	0h = x1, or the Port is in the <i>DL_Down</i> state 1h = x2 2h = x4 3h = x8 4h = x16	9	RO	No	
11:8	Quotient of division indicates the Link speed. 0h = Gen 1 (2.5 GT/s) 1h = Gen 2 (5.0 GT/s) 2h = Gen 3 (8.0 GT/s) All other encodings are <b>Reserved</b> .	10	RO	No	
15:12		11	RO	No	
23:16	<b>Factory Test Only</b>		RsvdP	No	00h
31:24	<b>Reserved</b>		RsvdP	No	00h

**Register 13-268. F58h Negotiated Link Width for Ports 16, 17, 18, 19 (Upstream Port(s))**

Bit(s)	Description	Upstream Port <i>x</i>	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The Downstream Port bits are <b>Reserved</b>, RsvdP, not serial EEPROM nor I<sup>2</sup>C writable, and have a default value of 0.</p> <p>The <b>Upstream Port <i>x</i></b> column is provided to indicate the Port Number associated with bits/fields that include “Port <i>x</i>” in their name.</p> <p><i>Table 13-5</i> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, Ports, SerDes modules, and Lanes.</p>					
3:0	<b>Negotiated Link Width and Link Speed Encoding for Port <i>x</i></b> Remainder of value divided by 5 indicates the negotiated Link width. If the Link is Down, the value is 0h.  0h = x1, or the Port is in the <i>DL_Down</i> state 1h = x2 2h = x4 3h = x8 4h = x16  Quotient of division indicates the Link speed.  0h = Gen 1 (2.5 GT/s) 1h = Gen 2 (5.0 GT/s) 2h = Gen 3 (8.0 GT/s)  All other encodings are <b>Reserved</b> .	16	RO	No	PCFG
7:4		17	RO	No	
11:8		18	RO	No	
15:12		19	RO	No	
23:16	<b>Factory Test Only</b>		RsvdP	No	00h
31:24	<b>Reserved</b>		RsvdP	No	00h

**Register 13-269. F5Ch Minimum/Maximum Status (Base mode – All Ports; Virtual Switch mode – VS Upstream Port(s))**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Minimum Cut-Thru Value</b> Bits [15:13] are tied to 000b.	RO	No	1FFFh
31:16	<b>Maximum Cut-Thru Value</b> Bits [31:29] are tied to 000b.	RO	No	0000h

**Register 13-270. F60h Ingress Control  
(Upstream Port(s))**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
3:0	<i>Factory Test Only</i>	RWS	Yes	2h
5:4	<i>Not used</i>	RWS	Yes	00b
6	<i>Reserved</i>	RsvdP	No	0
7	<i>Not used</i>	RsvdP	No	0
8	<b>Drop ECRC TLPs</b> Drop End-to-end Cyclic Redundancy Check (ECRC) TLPs. 1 = ECRC TLP was dropped	RWS	Yes	0
9	<b>Drop Poisoned TLPs</b> 0 = Nullify (invert Link Cyclic Redundancy Check (LCRC), add EDB) and forward any TLPs received as Poisoned. 1 = Discard Poisoned TLPs  <i>Note: The PEX 8748 never Sets the EP bit in the TLP Header. If a TLP is received with its EP bit Set (indicating Poisoned), the Port Sets its Detected Parity Error bit(s) (<a href="#">PCI Status</a> register <a href="#">Detected Parity Error</a> bit (offset 04h[31]) on the primary side, and/or <a href="#">Secondary Status</a> register <a href="#">Detected Parity Error</a> bit (offset 1Ch[31]) on the secondary side).</i>	RWS	Yes	0
10	<i>Factory Test Only</i>	RWS	Yes	0
12:11	<i>Not used</i>	RWS	Yes	00b
14:13	<i>Factory Test Only</i>	RWS	Yes	00b
15	<b>Disable NT Port Expansion ROM BAR</b> 0 = NT Port Expansion ROM BAR requests Memory space 1 = NT Port Expansion ROM BAR always reads 0, making the Expansion ROM not present	RWS	Yes	0
23:16	<i>Not used</i>	RWS	Yes	00h



**Register 13-270. F60h Ingress Control (Upstream Port(s)) (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
24	<b>Limit NT Port Link Interface Memory Accesses to NT Port Link Interface</b> 0 = NT Port Link Interface is allowed access to all PEX 8748 registers. 1 = Limit NT Port Link Interface Memory accesses that target PEX 8748 accesses to only the NT Port Link Interface Memory-Mapped registers. All Memory accesses that target registers in other PEX 8748 Ports are No Operation – Write data is ignored, Read operation returns zeros (0s).	RWS	Yes	0
25	<b>Factory Test Only</b>	RWS	Yes	0
26	<b>Disable Upstream Port BAR0 and BAR1</b> 0 = Enables all Upstream Port <b>Base Address 0</b> and <b>Base Address 1</b> registers ( <b>BAR0</b> and <b>BAR1</b> , Upstream Port(s), offsets <b>10h</b> and <b>14h</b> , respectively) 1 = Disables all Upstream Port <b>BAR0</b> and <b>BAR1</b>	RWS	Yes	0
27	<b>Flag Unexpected Completion Error</b> 0 = Flag unexpected Completion errors for Completions that hit the PEX 8748's internal virtual PCI Bus space 1 = Silently unexpected Completion errors for Completions that hit the PEX 8748's internal virtual PCI Bus space	RWS	Yes	0
28	<b>Disable VGA BIOS Memory Access Decoding</b> 0 = Enables the <b>Bridge Control</b> register <i>VGA 16-Bit Decode Enable</i> , <i>VGA Enable</i> , and <i>ISA Enable</i> bits (offset <b>3Ch[20:18]</b> , respectively), and enables decoding of PC ROM shadow addresses C_0000h to C_FFFFh (packets destined to these addresses are blocked) 1 = Disables the <b>Bridge Control</b> register <i>VGA 16-Bit Decode Enable</i> , <i>VGA Enable</i> , and <i>ISA Enable</i> bits (offset <b>3Ch[20:18]</b> , respectively), and disables decoding of PC ROM shadow addresses C_0000h to C_FFFFh (packets destined to these addresses are <i>not</i> blocked)	RWS	Yes	1
30:29	<b>Factory Test Only</b>	RWS	Yes	00b
31	<b>Not used</b>	RWS	Yes	0

13.22.3     **Device-Specific Registers – Error Checking and Debug (Offsets F70h – FB0h)**

This section details the Device-Specific Error Checking and Debug registers located at offsets F70h through FB0h. [Table 13-46](#) defines the register map.

Other Device-Specific Error Checking and Debug registers are detailed in [Section 13.16.11](#), “Device-Specific Registers – Error Checking and Debug (Offsets 700h – 75Ch).”

**Table 13-46.    Device-Specific Error Checking and Debug Register Map (Offsets F70h – FB0h) (All Ports)**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Power Management Hot Plug User Configuration																																F70h
Reserved																																F74h – FA4h
ACK Transmission Latency Limit																																FA8h
Bad TLP Count																																FACh
Bad DLLP Count																																FB0h

**Register 13-271. F70h Power Management Hot Plug User Configuration (All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>L0s Entry Idle Count</b> Traffic Idle time to meet, to enter the L0s Link PM state. 0 = Idle condition must last 1 $\mu$ s 1 = Idle condition must last 4 $\mu$ s		RWS	Yes	0
1	<b>Factory Test Only</b>		RWS	Yes	0
2	<b>Not enabled</b> Functionality associated with this bit is enabled only on Downstream Ports.	Upstream	RWS	Yes	0
	<b>HPC PME Turn-Off Enable</b> 1 = PME Turn-Off Message is transmitted before the Port is turned Off on a Downstream Port	Downstream	RWS	Yes	0
4:3	<b>Not enabled</b> Functionality associated with this field is enabled only on Downstream Ports.	Upstream	RWS	Yes	00b
	<b>HPC T<sub>pepv</sub></b> Hot Plug Port time from Power Enable to Power Valid. Indicates the delay from when HP_PWREN <sub>x</sub> is asserted High, to when power is valid at a slot. (Refer to <a href="#">Section 10.8.1.2, “Slot Power-Up Sequencing when Power Controller Present Bit Is Set,”</a> for details.) 00b = Feature is disabled, and HP_PWR_GOOD <sub>x</sub> inputs are used for Power Valid 01b = 128 ms 10b = 256 ms 11b = 512 ms	Downstream	RWS	Yes	00b
5	<b>Factory Test Only</b>		RWS	Yes	0
6	<b>Not enabled</b> Functionality associated with this bit is enabled only on Downstream Ports.	Upstream	RWS	Yes (Serial EEPROM only)	0
	<b>HP_PWR_GOOD<sub>x</sub> Active-Low Enable</b> Controls the HP_PWR_GOOD <sub>x</sub> input polarity. (Refer to <a href="#">Section 10.8.1.2, “Slot Power-Up Sequencing when Power Controller Present Bit Is Set,”</a> for details.) 0 = HP_PWR_GOOD <sub>x</sub> inputs are Active-High 1 = HP_PWR_GOOD <sub>x</sub> inputs are Active-Low	Downstream	RWS	Yes (Serial EEPROM only)	0
7	<b>Factory Test Only</b>		RWS	Yes	0

**Register 13-271. F70h Power Management Hot Plug User Configuration  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>DLLP Timeout Link Retrain Disable</b> Disable Link retraining when no Data Link Layer Packets (DLLPs) are received for more than 256 $\mu$ s. 0 = Enables Link retraining when no DLLPs are received for more than 256 $\mu$ s (default) 1 = DLLP Timeout is disabled		RWS	Yes	0
9	<b>Factory Test Only</b>		RWS	Yes	0
10	<b>L0s Entry Disable</b> 0 = Enables entry into the L0s Link PM state on a Port when the L0s idle conditions are met 1 = Disables entry into the L0s Link PM state on a Port when the L0s idle conditions are met		RWS	Yes	0
11	<b>Factory Test Only</b>		RWS	Yes	0
12	<b>Software-Controlled Hot Plug Enable</b> 1 = Hot Plug input functionality is disabled and software controls the input functionality		RWS	Yes	0
13	<b>Software-Controlled Power Good</b> When bit 12 ( <i>Software-Controlled Hot Plug Enable</i> ) is Set, the value in this bit controls HP_PWR_GOOD_x inputs to the PEX 8748		RWS	Yes	0
14	<b>Upstream Hot Plug Enable</b> 1 = Enables Presence Detect and its corresponding interrupt on the Upstream Port(s)	Upstream	RWS	Yes	0
		Downstream	RsvdP	No	0
15	<b>Port Is Serial Hot Plug Port</b> 1 = Indicates the Port is a Serial Hot Plug Port, using I/O Expanders		RO	No	0

**Register 13-271. F70h Power Management Hot Plug User Configuration  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>HPC Serial Expansion Controller Disable</b> 0 = Enables Serial Hot Plug capability on all Ports 1 = Disables Serial Hot Plug capability on all Ports	0	RWS	Yes	0
	<i>Not used</i>	Otherwise	RsvdP	No	0
17	<b>40-Pin I/O Expander Scan Disable</b> 0 = 40-pin I/O Expander scan 1 = Disables 40-pin I/O Expander scan		RWS	Yes	0
18	<b>Serial Hot Plug INTx De-Bounce Disable</b> 1 = Disables the 10-ms De-Bounce Counter in the Serial Hot Plug Controller, for I/O Expander Interrupt inputs		RWS	Yes	0
19	<b>Serial Hot Plug Override Parallel Disable</b> 0 = Defaults to Serial Hot Plug when a Port is both Parallel and Serial Hot Plug-capable 1 = Selects Parallel Hot Plug if a Port is both Parallel and Serial Hot Plug-capable		RWS	Yes	0
20	<b>HPC I/O Reload</b> 1 = Parallel Hot Plug Controller/Serial Hot Plug Controller (I <sup>2</sup> C I/O Expander) Output pin values are reloaded from field [26:21] ( <i>HPC Output Reload Value</i> ). After the action is complete, this bit is self-Clearing.		RWS	Yes	0
26:21	<b>HPC Output Reload Value</b> When bit 20 ( <i>HPC I/O Reload</i> ) is Set, values from this field are reloaded to the Hot Plug Controller outputs associated with the Port.		RWS	Yes	0-0h
	<b>Bit</b>	<b>Description/Function</b>			
	21	<b>HP_PWRLED_x# or Serial Hot Plug PWRLED#</b>			
	22	<b>HP_ATNLED_x# or Serial Hot Plug ATNLED#</b>			
	23	<b>HP_PWREN_x or Serial Hot Plug PWREN</b>			
	24	<b>HP_CLKEN_x# or Serial Hot Plug REFCLKEN#</b>			
	25	<b>HP_PERST_x# or Serial Hot Plug PERST#</b>			
26	<b>Serial Hot Plug INTERLOCK</b>				
27	<b>Software Present Detect State Value</b> Presence Detect state (Port's <b>Slot Status</b> register <i>Presence Detect State</i> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[22])) value is programmed from this register when bit 14 or 12 ( <i>Upstream Hot Plug Enable</i> or <i>Software-Controlled Hot Plug Enable</i> , respectively) is Set.		RWS	Yes	0

**Register 13-271. F70h Power Management Hot Plug User Configuration  
(All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
28	<b>Software MRL State Value</b> Manually operated Retention Latch (MRL) Sensor state (Port's <b>Slot Status</b> register <i>MRL Sensor State</i> bit (Base mode – Downstream Ports; Virtual Switch mode – Upstream and Downstream Ports, offset 80h[21])) value is programmed from this register when bit 14 or 12 ( <i>Upstream Hot Plug Enable</i> or <i>Software-Controlled Hot Plug Enable</i> , respectively) is Set.		RWS	Yes	0
29	<b>PHY Upstream Port Control Write Enable</b> 1 = Enables the following bits to be functional on Upstream Port(s): <ul style="list-style-type: none"> <li>• <b>Link Control</b> register <i>Link Disable</i> bit (offset 78h[4])</li> <li>• <b>Link Control</b> register <i>Retrain Link</i> bit (offset 78h[5])</li> <li>• <b>Link Control 2</b> register <i>Target Link Speed</i> field (offset 98h[3:0])</li> </ul>		RWS	Yes	0
31:30	<b>Factory Test Only</b>		RWS	Yes	00b

**Register 13-272. FA8h ACK Transmission Latency Limit  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The value of this register should be valid after Link negotiation.				
11:0	<b>ACK Transmission Latency Limit</b> Acknowledge Control Packet (ACK) Transmission Latency Limit. The value of this field changes, based upon the Port's <a href="#">Negotiated Link Width</a> (offset 78h[25:20]), <a href="#">Current Link Speed</a> (offset 78h[19:16]) after the Link is Up, and <a href="#">Maximum Payload Size</a> (offset 70h[7:5]). x1 = 255d x2 = 217d x4 = 118d x8 = 107d x16 = 100d	RWS	Yes	Defined by STRAP_STNx_PORTCFGx input states
15:12	<b>Reserved</b>	RsvdP	No	0h
23:16	<b>Upper 8 Bits of the Replay Timer Limit</b> If the serial EEPROM is not present, the value of this register changes based upon the negotiated Link width after the Link is Up. The value in this field is a multiplier of the default internal timer values that are compliant to the <i>PCI Express Base r3.0</i> . These bits should normally remain the default value, 00h.	RWS	Yes	00h
30:24	<b>Reserved</b>	RsvdP	No	00h
31	<b>ACK Transmission Latency Timer Status</b> Indicates the written status of field [11:0] ( <i>ACK Transmission Latency Limit</i> ). After the register is written, either by software and/or serial EEPROM, this bit is Set and Cleared only by a Fundamental Reset.	RO	No	0

**Register 13-273. FACH Bad TLP Count  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Bad TLP Count</b> Counts the quantity of TLPs received with bad LCRC, or quantity of TLPs with a Sequence Number Mismatch error. The Counter saturates at FFFF_FFFFh and does not roll over to 0000_0000h.	RWS	Yes	0000_0000h

**Register 13-274. FB0h Bad DLLP Count  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Bad DLLP Count</b> Counts the quantity of DLLPs received with bad LCRC, or quantity of DLLPs with a Sequence Number Mismatch error. The Counter saturates at FFFF_FFFFh and does not roll over to 0000_0000h.	RWS	Yes	0000_0000h



### 13.23 Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)

This section details the Advanced Error Reporting Extended Capability registers. [Table 13-47](#) defines the register map.

**Table 13-47. Advanced Error Reporting Extended Capability Register Map (All Ports)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																				
Next Capability Offset (138h or 148h)																Capability Version (1h)				PCI Express Extended Capability ID (0001h)																FB4h
Uncorrectable Error Status																																FB8h				
Uncorrectable Error Mask																																FBCh				
Uncorrectable Error Severity																																FC0h				
Reserved																Correctable Error Status																FC4h				
Reserved																Correctable Error Mask																FC8h				
Advanced Error Capabilities and Control																																FCCh				
Header Log 0																																FD0h				
Header Log 1																																FD4h				
Header Log 2																																FD8h				
Header Log 3																																FDCCh				

## Register 13-275. FB4h Advanced Error Reporting Extended Capability Header (All Ports)

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>PCI Express Extended Capability ID</b>		ROS	Yes	0001h
19:16	<b>Capability Version</b>		ROS	Yes	1h
31:20	<b>Next Capability Offset</b> Program to 138h, which addresses the Upstream Port <b>Power Budget Extended Capability</b> structure.	Upstream	ROS	Yes	138h
	Program to 148h, which addresses the <b>Virtual Channel Extended Capability</b> structure.	Downstream	ROS	Yes	148h

**Register 13-276. FB8h Uncorrectable Error Status (All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> If an individual error is masked (corresponding bit in the <b>Uncorrectable Error Mask</b> register (offset <b>FBCh</b>) is Set) when the error is detected, its Error Status bit is still updated; however, an error reporting Message (ERR_FATAL or ERR_NONFATAL) is <b>not</b> sent to the Root Complex, and the <b>Advanced Error Capabilities and Control</b> register <b>First Error Pointer</b> field (offset <b>FCCh[4:0]</b>) and <b>Header Log x</b> registers (offsets <b>FD0h</b> through <b>FDCh</b>) remain unchanged.</p> <p><b>Table 13-5</b> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>For bit 22 – The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
3:0	<b>Reserved</b>		RsvdP	No	0h
4	<b>Data Link Protocol Error Status</b> 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Surprise Down Error Status</b> 0 = No error is detected 1 = Error is detected	Downstream	RW1CS <sup>a</sup>	Yes	0
11:6	<b>Reserved</b>		RsvdP	No	0-0h
12	<b>Poisoned TLP Status</b> 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
13	<b>Flow Control Protocol Error Status</b> <b>Reserved/Not supported</b>		RsvdP	No	0
14	<b>Completion Timeout Status</b> <b>Not applicable to switches.</b>		RsvdP	No	0
15	<b>Completer Abort Status</b>		RW1CS <sup>a</sup>	Yes	0
16	<b>Unexpected Completion Status</b> 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
17	<b>Receiver Overflow Status</b> 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
18	<b>Malformed TLP Status</b> 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
19	<b>ECRC Error Status</b> 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0

**Register 13-276. FB8h Uncorrectable Error Status (All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Unsupported Request Error Status</b> 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
21	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Violation Error Status</b> 0 = No violation is detected 1 = Violation is detected	Downstream	RW1CS <sup>a</sup>	Yes	0
22	<b>Uncorrectable Internal Error Status</b> 0 = No error is detected 1 = Error is detected	<b>Base Mode</b> 0, 8, 16  <b>Virtual Switch Mode</b> 0, 8, 16, accessible through the Management Port	RW1CS <sup>a</sup>	Yes	0
	<b>Reserved</b>	Otherwise	RsvdP	No	0
23	<b>MC Blocked TLP Status</b> Multicast blocked TLP status. 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
24	<b>Atomic Operation Egress Blocked Status</b> 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
31:25	<b>Reserved</b>		RsvdP	No	0-0h

- a. When the **ECC Error Check Disable** register **Software Force Error Enable** bit (Transparent Downstream Ports, offset 720h[2]) is Set, Type changes from RW1CS to RW.

**Register 13-277. FBCh Uncorrectable Error Mask  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The bits in this register can be used to mask their respective <b>Uncorrectable Error Status</b> register bits (offset <b>FB8h</b>).</p> <p><b>Table 13-5</b> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>For bit 22 – The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
3:0	<b>Reserved</b>		RsvdP	No	0h
4	<b>Data Link Protocol Error Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error		RWS	Yes	0
5	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>Surprise Down Error Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error	Downstream	RWS	Yes	0
11:6	<b>Reserved</b>		RsvdP	No	0-0h
12	<b>Poisoned TLP Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error		RWS	Yes	0
13	<b>Flow Control Protocol Error Mask</b> <b>Reserved/Not supported</b>		RsvdP	No	0
14	<b>Completion Timeout Mask</b> <b>Not applicable to switches.</b>		RsvdP	No	0
15	<b>Completer Abort Mask</b>		RWS	Yes	0

**Register 13-277. FBCh Uncorrectable Error Mask (All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Unexpected Completion Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error		RWS	Yes	0
17	<b>Receiver Overflow Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error		RWS	Yes	0
18	<b>Malformed TLP Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error		RWS	Yes	0
19	<b>ECRC Error Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error		RWS	Yes	0
20	<b>Unsupported Request Error Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error		RWS	Yes	0
21	<b>Reserved</b>	Upstream	RsvdP	No	0
	<b>ACS Violation Error Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error	Downstream	RWS	Yes	0
22	<b>Uncorrectable Internal Error Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error	<b>Base Mode</b> 0, 8, 16  <b>Virtual Switch Mode</b> 0, 8, 16, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	1
23	<b>MC Blocked TLP Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error		RWS	Yes	0
24	<b>Atomic Operation Egress Blocked Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error		RWS	Yes	0
31:25	<b>Reserved</b>		RsvdP	No	0-0h

**Register 13-278. FC0h Uncorrectable Error Severity (All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Notes:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports. For bit 22 – The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.					
3:0	<b>Reserved</b>		RsvdP	No	0h
4	<b>Data Link Protocol Error Severity</b> 0 = Error is reported as non-fatal 1 = Error is reported as fatal		RWS	Yes	1
5	<b>Surprise Down Error Severity</b> 0 = Error is reported as non-fatal 1 = Error is reported as fatal	Upstream	ROS	No	1
		Downstream	RWS	Yes	1
11:6	<b>Reserved</b>		RsvdP	No	0-0h
12	<b>Poisoned TLP Severity</b> 0 = Error is handled as an Advisory Non-Fatal error, and reported as a Correctable error 1 = Error is reported as fatal		RWS	Yes	0
13	<b>Flow Control Protocol Error Severity</b> <b>Reserved/Not supported</b>		RO	No	1
14	<b>Completion Timeout Severity</b> <b>Not applicable to switches.</b> Because the Status and Mask are both <b>Reserved</b> for this bit, Severity can be ignored.		RsvdP	No	0
15	<b>Completer Abort Severity</b> 0 = Error is handled as an Advisory Non-Fatal error, and reported as a Correctable error 1 = Error is reported as fatal		RWS	Yes	0
16	<b>Unexpected Completion Severity</b> 0 = Error is handled as an Advisory Non-Fatal error, and reported as a Correctable error 1 = Error is reported as fatal		RWS	Yes	0
17	<b>Receiver Overflow Severity</b> 0 = Error is reported as non-fatal 1 = Error is reported as fatal		RWS	Yes	1
18	<b>Malformed TLP Severity</b> 0 = Error is reported as non-fatal 1 = Error is reported as fatal		RWS	Yes	1
19	<b>ECRC Error Severity</b> 0 = Error is handled as an Advisory Non-Fatal error, and reported as a Correctable error 1 = Error is reported as fatal		RWS	Yes	0

**Register 13-278. FC0h Uncorrectable Error Severity (All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Unsupported Request Error Severity</b> 0 = Error is handled as an Advisory Non-Fatal error, and reported as a Correctable error 1 = Error is reported as fatal		RWS	Yes	0
21	<b>Reserved</b>	Upstream	RsvdP	Yes	0
	<b>ACS Violation Error Severity</b> 0 = Error is handled as an Advisory Non-Fatal error, and reported as a Correctable error 1 = Error is reported as fatal	Downstream	RWS	Yes	0
22	<b>Uncorrectable Internal Error Severity</b> 0 = Error is reported as non-fatal 1 = Error is reported as fatal	<b>Base Mode</b> 0, 8, 16  <b>Virtual Switch Mode</b> 0, 8, 16, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	1
23	<b>MC Blocked TLP Severity</b> 0 = Error is reported as non-fatal 1 = Error is reported as fatal		RWS	Yes	0
24	<b>Atomic Operation Egress Blocked Severity</b> 0 = Error is reported as non-fatal 1 = Error is reported as fatal		RWS	Yes	0
31:25	<b>Reserved</b>		RsvdP	No	0-0h

**Register 13-279. FC4h Correctable Error Status  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> If an individual error is masked (corresponding bit in the <b>Correctable Error Mask</b> register (offset <b>FC8h</b>) is Set) when the error is detected, its Error Status bit is still updated; however, an error reporting Message (ERR_COR) is <b>not</b> sent to the Root Complex.</p> <p><b>Table 13-5</b> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>For bit 14 – The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
0	<b>Receiver Error Status</b> Indicates that one or more of the following Receiver errors was detected: <ul style="list-style-type: none"> <li>• Code (symbol) error</li> <li>• Disparity error</li> <li>• Elastic Buffer overflow/underflow</li> <li>• Gen 3 Framing error</li> </ul> 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
5:1	<b>Reserved</b>		RsvdP	No	0-0h
6	<b>Bad TLP Status</b> 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
7	<b>Bad DLLP Status</b> 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
8	<b>REPLAY NUM Rollover Status</b> Replay Number Rollover status. 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
11:9	<b>Reserved</b>		RsvdP	No	000b



**Register 13-279. FC4h Correctable Error Status (All Ports) (Cont.)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
12	<b>Replay Timer Timeout Status</b> 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
13	<b>Advisory Non-Fatal Error Status</b> 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
14	<b>Corrected Internal Error Status</b> 0 = No error is detected 1 = Error is detected	<b>Base Mode</b> 0, 8, 16  <b>Virtual Switch Mode</b> 0, 8, 16, accessible through the Management Port	RW1CS <sup>a</sup>	Yes	0
	<b>Reserved</b>	Otherwise			
15	<b>Header Log Overflow Status</b> 0 = No error is detected 1 = Error is detected		RW1CS <sup>a</sup>	Yes	0
31:16	<b>Reserved</b>		RsvdP	No	0000h

- a. When the **ECC Error Check Disable** register **Software Force Error Enable** bit (Transparent Downstream Ports, offset 720h[2]) is Set, Type changes from RW1CS to RW.

**Register 13-280. FC8h Correctable Error Mask  
(All Ports)**

Bit(s)	Description	Ports	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Notes:</b> The bits in this register can be used to mask their respective <b>Correctable Error Status</b> register bits (offset <b>FC4h</b>).</p> <p><b>Table 13-5</b> indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Stations, Station Ports, and Ports.</p> <p>For bit 14 – The Port 0 bits are for Ports 0, 1, 2, and 3. The Port 8 bits are for Ports 8, 9, 10, and 11. The Port 16 bits are for Ports 16, 17, 18, and 19.</p>					
0	<b>Receiver Error Mask</b> 0 = Error reporting is not masked 1 = Error reporting is masked		RWS	Yes	0
5:1	<b>Reserved</b>		RsvdP	No	0-0h
6	<b>Bad TLP Mask</b> 0 = Error reporting is not masked 1 = Error reporting is masked		RWS	Yes	0
7	<b>Bad DLLP Mask</b> 0 = Error reporting is not masked 1 = Error reporting is masked		RWS	Yes	0
8	<b>REPLAY NUM Rollover Mask</b> Replay Number Rollover mask. 0 = Error reporting is not masked 1 = Error reporting is masked		RWS	Yes	0
11:9	<b>Reserved</b>		RsvdP	No	000b
12	<b>Replay Timer Timeout Mask</b> 0 = Error reporting is not masked 1 = Error reporting is masked		RWS	Yes	0
13	<b>Advisory Non-Fatal Error Mask</b> 0 = Error reporting is not masked 1 = Error reporting is masked		RWS	Yes	1
14	<b>Corrected Internal Error Mask</b> 0 = Error reporting is not masked 1 = Error reporting is masked	<b>Base Mode</b> 0, 8, 16  <b>Virtual Switch Mode</b> 0, 8, 16, accessible through the Management Port	RWS	Yes	1
	<b>Reserved</b>	Otherwise	RsvdP	No	1
15	<b>Header Log Overflow Mask</b> 0 = Error reporting is not masked 1 = Error reporting is masked		RWS	Yes	1
31:16	<b>Reserved</b>		RsvdP	No	0000h

**Register 13-281. FCCh Advanced Error Capabilities and Control (All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
4:0	<b>First Error Pointer</b> Identifies the bit position of the first error reported in the <b>Uncorrectable Error Status</b> register (offset FB8h).	ROS	No	1Fh
5	<b>ECRC Generation Capable</b> 0 = ECRC generation is not supported 1 = ECRC generation is supported, but must be enabled	ROS	Yes	1
6	<b>ECRC Generation Enable</b> 0 = ECRC generation is disabled 1 = ECRC generation is enabled	RWS	Yes	0
7	<b>ECRC Check Capable</b> 0 = ECRC checking is not supported 1 = ECRC checking is supported, but must be enabled	ROS	Yes	1
8	<b>ECRC Check Enable</b> 0 = ECRC checking is disabled 1 = ECRC checking is enabled	RWS	Yes	0
31:9	<b>Reserved</b>	RsvdP	No	0-0h

**Register 13-282. FD0h Header Log 0  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>TLP Header 0</b> First DWord Header. TLP Header associated with error.	ROS	Yes	0000_0000h

**Register 13-283. FD4h Header Log 1  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>TLP Header 1</b> Second DWord Header. TLP Header associated with error.	ROS	Yes	0000_0000h

**Register 13-284. FD8h Header Log 2  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>TLP Header 2</b> Third DWord Header. TLP Header associated with error.	ROS	Yes	0000_0000h

**Register 13-285. FDCh Header Log 3  
(All Ports)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>TLP Header 3</b> Fourth DWord Header. TLP Header associated with error.	ROS	Yes	0000_0000h

### 14.1 Introduction

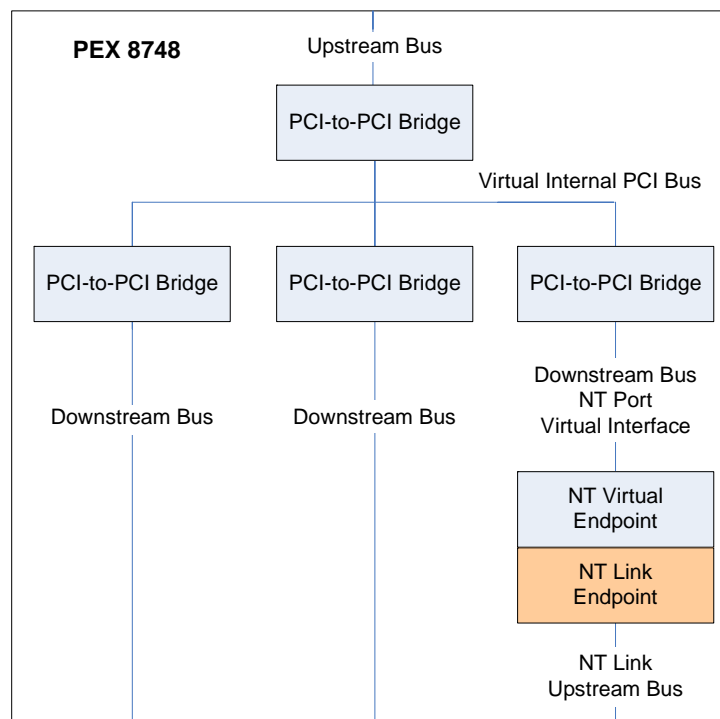
**Note:** Check the latest design guides, application notes and errata list for Non-Transparent (NT) usage.

A Non-Transparent bridge (NT bridge, or NTB) consists of two back-to-back PCI Express endpoints, a Virtual side endpoint, and a Link side endpoint. In the PEX 8748, the Virtual side endpoint shows up to software behind two PCI-to-PCI bridges on the switch (a switch is a hierarchy of PCI-to-PCI bridges) while the Link side endpoint is all that the Link side Host “finds,” as illustrated in [Figure 14-1](#).

The NTB is used to connect two independent address/Host domains, and can be used to implement High-Availability systems or Intelligent I/O modules using PCI Express technology. NT bridges allow systems to isolate Address spaces, by appearing as an endpoint to each Host.

The NTB exposes a Type 0 Configuration Space register (CSR) Header to each side – Virtual and Link – and forwards transactions from one domain to the other, using Address and Requester ID translation through Memory windows. The NTB also includes **Doorbell** registers, for transmitting interrupts from one side of the bridge to the other, and **Scratchpad** registers, accessible from both domains for inter-Host communication. The PEX 8748, with a single Port, supports the [Intelligent Adapter Mode](#) system model.

**Figure 14-1. NT Bridge – NT Mode in Base Mode**



## 14.1.1 NT Device Type Identification

An NT bridge identifies itself as **PCI Class Code** “other,” 068000h, with a Type 0 Header. A Type 0 Header is used by PCI endpoints. In contrast, a Transparent PCI-to-PCI bridge identifies itself as a **PCI Class Code** 060400h, with a Type 1 Header (PCI-to-PCI bridge).

The **PCI Express Capability** register includes a **Device/Port Type** field (offset 68h[23:20]). In this register, a Transparent bridge/switch Port identifies itself as an *Upstream Port* or *Downstream Port*, while an NTB/NT Port identifies itself as a *PCI Express endpoint*.

## 14.1.2 Enabling Non-Transparent Bridging

NT mode is enabled if the **STRAP\_NT\_UPSTRM\_PORTSEL[2:0]** 3-state inputs are not all pulled or tied High to **VDD18**. Valid values on these 3-state inputs are 0, Z (floating), and 1, and when used together, provide 27 possible combinations, of which values within the ranges of 0 through 3, 8 through 11, and 16 through 19 represent corresponding Port Numbers. The selected value determines which Port is assigned to be the NT Port. The Port Numbers listed above do not exist in all Port configurations, and the programmed value must correspond to a valid Port Number; otherwise, that value is **Reserved**. **Table 14-1** lists the 27 combinations for the three 3-state inputs, and their equivalent binary and decimal values (indicating register values and Port Numbers), in incrementing order of the 3-state encoding.

Alternately, I<sup>2</sup>C, software, and/or an optional serial EEPROM (if present) can enable NT, by Setting the **VS0 Upstream** register **NT Enable** bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 360h[13]). The NT Port is determined by the register’s **NT Port** field (field [12:8]).

**Table 14-1. 3-State-Encoded STRAP\_NT\_UPSTRM\_PORTSEL[2:0] Values**

STRAP_NT_UPSTRM_PORTSEL[2:0] Input States	Binary Value	Port Number/ Decimal Value <sup>a</sup>
000	0_0000b	0
00Z	0_0001b	1
001	0_0010b	2
0Z0	0_0011b	3
0ZZ, 0Z1, 010, 01Z	<b>Reserved</b>	<b>Reserved</b>
011	0_1000b	8
Z00	0_1001b	9
Z0Z	0_1010b	10
Z01	0_1011b	11
ZZ0, ZZZ, ZZ1, Z10	<b>Reserved</b>	<b>Reserved</b>
Z1Z	1_0000b	16
Z11	1_0001b	17
100	1_0010b	18
10Z	1_0011b	19
1ZZ, 1Z1, 110, 11Z	<b>Reserved</b>	<b>Reserved</b>
111	1_1010b	Disabled

- a. **Table 13-5** indicates which Ports are enabled/used, and when, based upon the **STRAP\_STNx\_PORTCFGx** input states and/or serial EEPROM programming.

### 14.1.3 NT Port Features

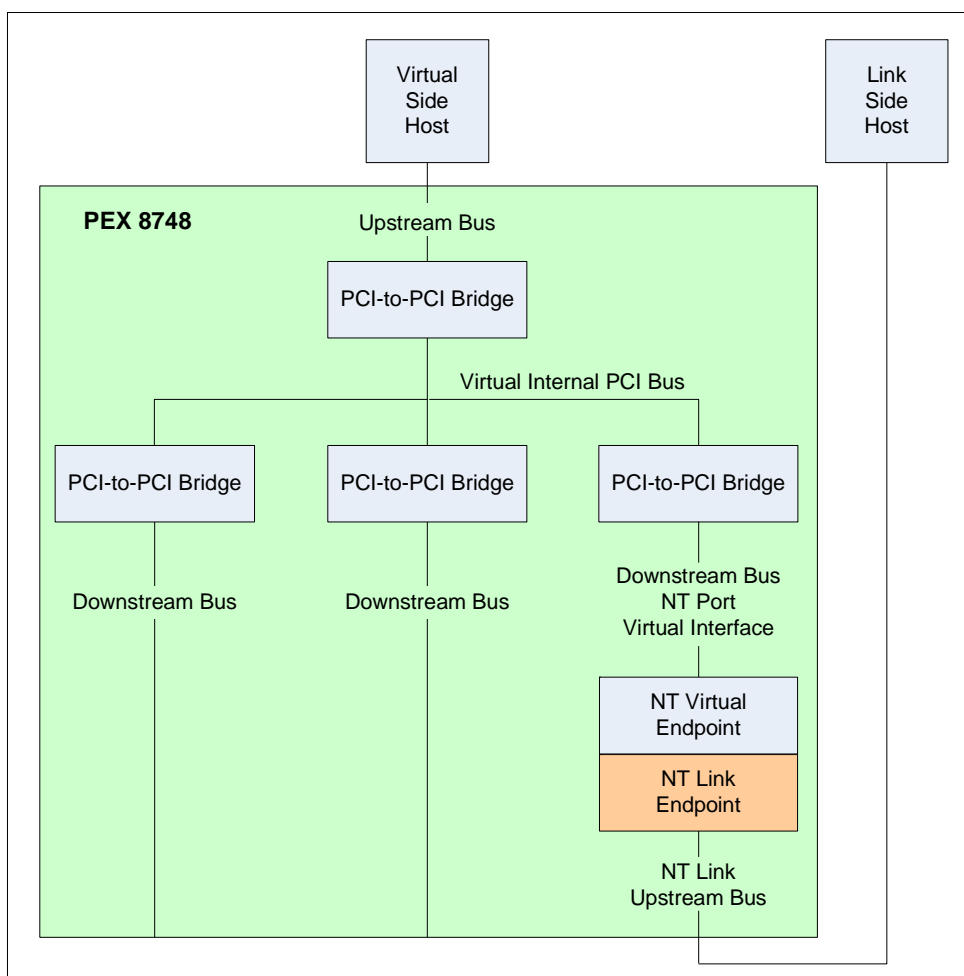
- Maps PEX 8748 Configuration registers into either 32- or 64-bit Memory space
- Base Address registers (BARs)
  - Implements four 32-bit, two 32-bit and one 64-bit, or two 64-bit BARs
  - Supports BAR Size programming, through the **BARx Setup** register(s)
  - Allows BARs to be individually disabled, including Memory-Mapped BARs
- Supports [Direct Address Translation](#)
  - 32-to-32-bit address conversion
  - 32-to-64-bit address conversion
  - 64-to-32-bit address conversion
  - 64-to-64-bit address conversion
  - Requester ID (Bus Number, Device Number, and Function Number) conversion across the NT bridge
- 16 **Doorbell** registers
- 8 **Scratchpad** registers
- Supports Requester ID and Completion ID translation
- NT Port Link Interface *DL\_Active* state change generates interrupt to Virtual side Host
- Supports Cursor mechanism
- Supports Expansion ROM on either NT Port interface
- Supports End-to-end Cyclic Redundancy Check (ECRC)
- Provides ability to Clear *No Snoop* Transaction Layer Packet (TLP) attribute (if enabled)
- Programmable Upstream Port(s) and NT Port for the enabling of High-Availability systems (Failover and Redundant systems)
- Ability to bring down the NT Port Link side, when the Virtual side domain is down
- Supports the Fencing mechanism
- Signals Device-Specific interrupt to the Virtual side Host when the NT Port Link Interface detects TLP errors
- Signals Device-Specific interrupt to the Virtual side Host when the NT Port Link Interface receives Error Messages (Safety bit-controllable)
- Ability to disable NT Port Link Interface Hot Reset from resetting NT Link side registers
- Supports Configuration Space access control
- NT Link side Reset input ([PEX\\_NT\\_PERST#](#))
- Programmable **Captured Bus and Device Numbers** register
- Requester ID translation table supports 32 IDs, in each direction

## 14.1.4 Intelligent Adapter Mode

The diagram provided in [Figure 14-2](#) has two Type 0 CSR Headers in the NTB. The one nearer the internal virtual PCI Bus is referred to as the *Virtual endpoint*, and is accessed by way of the *Virtual Interface*. The one nearer the PCI Express Link is referred to as the *Link endpoint*, and is accessed by way of the *Link Interface*.

The Intelligent Adapter model includes the NT Port Link endpoint that is connected to the System domain. The Link side Host manages only the NT Port Link endpoint Type 0 function. The Virtual side Host manages all PEX 8748 Transparent Port Type 1 and NT Port Virtual endpoint Type 0 functions. Cross-domain traffic is routed through an Address Translation mechanism. (Refer to [Section 14.1.11.](#))

**Figure 14-2. Intelligent Adapter Software Model – Base Mode**





## 14.1.5 NT Port Reset

The section discusses NT mode exceptions and enhancements to Transparent mode PCI Express (standard) reset behavior.

In discussing reset, it is useful to consider three different types of registers:

- **Setup** registers (**BARx Setup**, **Memory BARx Address Translation**, and Requester ID LUT), for transaction forwarding and address translation
- **Doorbell** registers, for interrupt-based synchronization between Host domains, and
- **Scratchpad** registers, for information exchange

### 14.1.5.1 Fundamental Reset (PERST#)

The Fundamental Reset input, **PEX\_PERST#**, resets all PEX 8748 states, including all NT Port states, to an initial power-up value. This includes resetting all Sticky bits and all **Setup**, **Doorbell**, and **Scratchpad** registers.

VSx\_PERST# provides a similar Fundamental Reset, but only to its specific virtual switch, and does not reset states nor registers that are used by other virtual switches.

**PEX\_NT\_PERST#** resets the NT Port Link Interface.

### 14.1.5.2 Virtual Side Hot Reset

The Virtual side of the NTB sees a Hot Reset, if any of the following occurs:

- Transparent Upstream Port goes down or receives an in-band reset (and reset propagation is not blocked)
- Transparent Upstream Port(s)'s PCI-to-PCI bridge Sets the **Bridge Control** register *Secondary Bus Reset* bit (offset 3Ch[22])
- PCI-to-PCI bridge above the Virtual side endpoint Sets the *Secondary Bus Reset* bit

A Virtual side Hot Reset resets all the NT Port Virtual Interface states. Hot Reset does not propagate through the NT Port; therefore, the NT Port Link Interface is *not* reset by the Virtual side Hot Reset.

The PEX 8748 supports options that allow the Virtual side Hot Reset conditions to have no effect:

- Reset, from either Hot Reset being received at the PEX 8748 Upstream Port, or from an Upstream Port DL\_Down, both of which usually reset all Ports, and can be blocked by Setting the **Virtual Switch Debug** register *Upstream Port and NT-Link Port DL\_Down Reset Propagation Disable* bit (Upstream Port(s), offset A30h[4]).
- Hot Reset can be blocked at the NT Port Virtual Interface, by Setting the **Link Control** register *Link Disable* bit (offset 78h[4]) in the PCI-to-PCI bridge that is logically connected to the NT Port Virtual Interface.

The NT Port Virtual Interface **BARx Setup** and **Doorbell** registers are reset, while the NT Port Link Interface **BARx Setup** and **Doorbell** registers are unaffected. The **Scratchpad** registers on both sides are also unaffected.

### 14.1.5.3 Link Side Hot Reset

The Link side of the NTB sees a Hot Reset, if any of the following occurs:

- NT Link goes down
- NT Link receives an in-band reset

When the Link side receives a Hot Reset, the NT Port Link Interface registers are reset.

The PEX 8748 supports an option to bring down the NT Port Link, which resets the NT link side logic, by Setting the Port's **Port Control** register *Disable Port x* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 208h[3:0]). The identified Port (one Hot) goes to, and remains in, the *Detect.Quiet* substate, until the bit is Cleared. No EIOS is sent, which should force any connected device to the *Recovery*, and then *Detect*, state.

The Link side **Setup** and **Doorbell** registers are reset, while the Virtual side **Setup** and **Doorbell** registers are unaffected. The **Scratchpad** registers on both sides are also unaffected.

### 14.1.5.4 Reset Propagation

Reset propagation, during a Hot Reset or using the **Bridge Control** register *Secondary Bus Reset* mechanism, is limited to Transparent Downstream Ports. In the NTB, this reset cannot be propagated across the bridge (across the NT Port). (Refer to [Section 5.1, “Reset,”](#) for details regarding PEX 8748 Transparent mode reset behavior.)

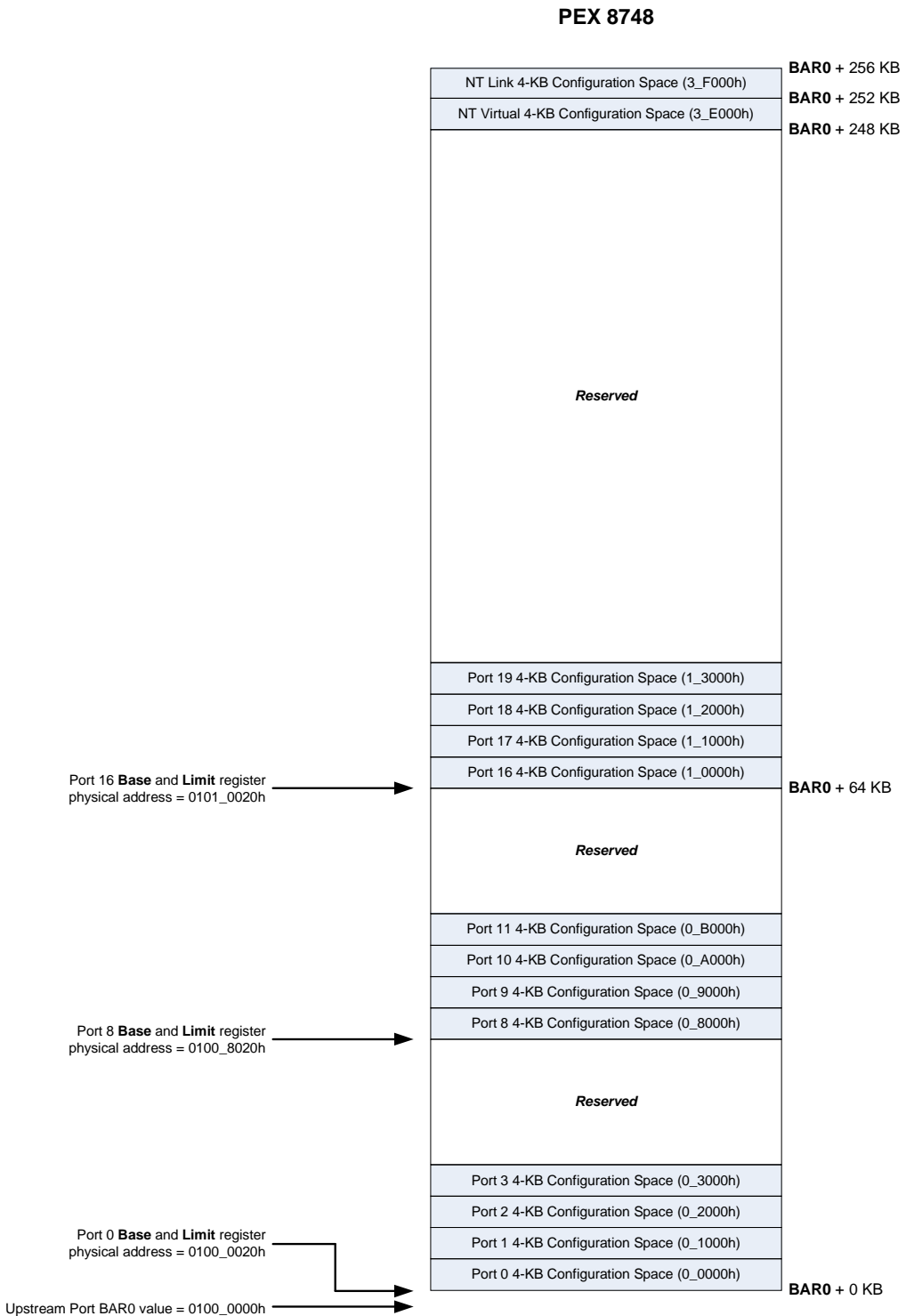
## 14.1.6 NT Port Memory-Mapped Base Address Registers

The NT Port Virtual and Link Interfaces individually claim 256 KB of memory, using **BAR0** and **BAR1**. The 256-KB space contains the Configuration Space registers for all PEX 8748 Ports. **BAR0** and **BAR1** can be programmed as one of the following:

- 32-bit BAR (**BAR0** is used, **BAR1** is **Reserved**; default mode)
- 64-bit BAR, by programming the Configuration **BAR0/1 Setup** register  
(NT Port Virtual Interface, offset 3\_E0D0h or NT Port Link Interface, offset 3\_F0E4h)
- **BAR0** and **BAR1** can be completely disabled

Figure 14-3 provides a memory-mapped view of the PEX 8748 Configuration Space registers. This view is the same from the Upstream Port(s), NT Port Virtual Interface, or NT Port Link Interface, although the Base address can be different in each case. Each Port has a 4-KB Memory space for Configuration registers. Starting at **BAR0**, Port 0 is at offset 0000h, Port 1 is at offset 1000h, and so forth, with 4-KB increments, per Port. There are also NT function registers at the top of the space, with the NT Port Virtual Interface at 3\_E000h and NT Port Link Interface at 3\_F000h. **Reserved** spaces should *not* be used.

Figure 14-3. NT Mode Configuration Register Mapping to Memory-Mapped BAR



*Note:* Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP\_STNx\_PORTCFGx input states and/or serial EEPROM programming.

## 14.1.7 Doorbell Registers

**Doorbell** registers are used to signal interrupts from one side of the NTB to the other. This section describes a typical set of Doorbell Control registers.

A 16-bit software-controlled Interrupt Request register and associated 16-bit Mask register are implemented for the NT Port Virtual and Link Interfaces. The Doorbell mechanisms consist of the following registers:

- **Virtual Interface IRQ Set** (offset C4Ch)
- **Virtual Interface IRQ Clear** (offset C50h)
- **Virtual Interface IRQ Mask Set** (offset C54h)
- **Virtual Interface IRQ Mask Clear** (offset C58h)
- **Link Interface IRQ Set** (offset C5Ch)
- **Link Interface IRQ Clear** (offset C60h)
- **Link Interface IRQ Mask Set** (offset C64h)
- **Link Interface IRQ Mask Clear** (offset C68h)

***Note:** There is redundancy here. Both the Virtual and Link Interfaces have a copy of the same register, which all perform the same function for all eight of these registers.*

Each NT Port interface has its own register (*that is*, there are two of each register). The offsets are all located in NT 4-KB space. *For example*, offset C4Ch can be 3\_EC4Ch (NT Port Virtual Interface) and 3\_FC4Ch (NT Port Link Interface).

The Virtual Interface IRQ is for interrupts that exit the NT Port Virtual Interface. An interrupt is asserted on the NT Port Virtual Interface when one or more of the **Virtual Interface IRQ Set** register bits are Set by the NT Port Link Interface and their corresponding **Virtual Interface IRQ Mask Set** register bits are Cleared. An interrupt is de-asserted on the NT Port Virtual Interface when one or more of the **Virtual Interface IRQ Clear** register bits are Set from the NT Port Virtual Interface or their corresponding **Virtual Interface IRQ Mask Clear** register bits are Cleared. The interrupt is de-asserted when all Set bits are masked or Cleared.

The Link Interface IRQ is for interrupts that exit the NT Port Link Interface. An interrupt is asserted on the NT Port Link Interface when one or more of the **Link Interface IRQ Set** register bits are Set by the NT Port Virtual Interface and their corresponding **Link Interface IRQ Mask Set** register bits are Cleared. An interrupt is de-asserted on the NT Port Link Interface when one or more of the **Link Interface IRQ Clear** register bits are Set from the NT Port Link Interface or their corresponding **Link Interface IRQ Mask Clear** register bits are Cleared. The interrupt is de-asserted when all Set bits are masked or Cleared.

Because Memory Requests can access both sets of NT-Virtual and NT-Link Doorbell registers, software in either domain can generate Doorbell interrupts to both domains.

Internally, the **Set IRQ** and **Clear IRQ** registers are the same register. One location is used to Set bits and the other is used to Clear bits. The status can be read from either register.

In a PCI Express switch, interrupt state transitions (from Setting to Clearing, or vice versa) result in packets being transmitted Upstream on the appropriate side of the bridge, when INTx are enabled (**PCI Command** register *Interrupt Disable* bit, offset 04h[10], is Cleared). Standard PCI Express Capability structures allow these interrupts to be configured as INTx or MSIs. When MSIs are enabled (**MSI Control** register *MSI Enable* bit, offset 48h[16], is Set), packets are transmitted only when interrupts transition from Clear IRQ to Set IRQ.

NT Port Virtual Interface Doorbell interrupts can optionally use the **PEX\_INTA#** (Base mode) or **VSx\_PEX\_INTA#** (Virtual Switch mode) output for interrupt signaling, instead of the INTx or MSI signaling mechanisms, by Setting the **ECC Error Check Disable** register *Enable PEX\_INTA# or VSx\_PEX\_INTA# Interrupt Output for NT-Virtual Doorbell-Generated Interrupts* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 720h[7]).

The PEX 8748 Virtual interrupts are de-asserted when the NT Port goes to the *DL\_Down* state.

### 14.1.8 Scratchpad Registers

**Scratchpad** registers are readable and writable from both sides of the NTB, providing a generic means for inter-Host communication. A block of eight registers are provided, accessible in Memory space from the NT Port Virtual and Link Interfaces. These registers pass Control and Status information between Virtual and Link Interface devices, or they can be generic RW registers. Reading from or writing to **Scratchpad** registers does not cause interrupts to assert – **Doorbell** registers are used for that purpose. **Scratchpad** registers are reset only by a Fundamental Reset (**PEX\_PERST#**, Serial Hot Plug **PERST#** (SHP\_PERSTx#), VSx\_PERST#, and/or **PEX\_NT\_PERST#**).

Some non-active registers can also be used for **Scratchpad** registers, to extend the number available. The **Physical Layer User Test Pattern, Bytes x through y** registers (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offsets 20Ch through 218h), provide four additional registers, per Port.

## 14.1.9 NT Base Address Registers

There are two sets of NT Base Address registers (BARs) – one each for the NT Port Virtual and Link Interfaces. Each BAR has its own **Setup** and **Address Translation** register:

- **BARx Setup** registers enable/disable the BAR and define the window size and type. Program the **BARx Setup** registers prior to allowing configuration software to assign a resource for these BARs. (Discussed further in [Section 14.1.9.1](#).)
- **Memory BARx Address Translation** registers allow for an address change on the upper bits (up to the size of the space). Program the **Memory BARx Address Translation** registers, before generating traffic across the NT Port. This programming is typically performed by information downloaded from I<sup>2</sup>C, software, or an optional serial EEPROM (if present) on the destination side. The source side does not need to know what the Address Translation is.
- The address could change size. *For example*, the PEX 8748 NT Port allows a 32-bit device to communicate to a 64-bit device, and vice versa. (The same is true when the addresses are the same size, as well – a 32-bit device can communicate with a 32-bit device, and a 64-bit can communicate with a 64-bit device.)

### 14.1.9.1 NT BARx Setup Registers

PCI defines Base Address Registers (BARs) for a PCI device to claim Address space. For the PEX 8748, **BAR0** and **BAR1** provide Memory-Mapped access to the CSRs, while **BAR2**, **BAR3**, **BAR4**, and **BAR5** provide programmable window sizes to the other side of the NTB.

The **BARx Setup** registers are used to program the window size of each BAR. [Table 14-2](#) briefly describes each NT Port BAR. **BAR2**, **BAR3**, **BAR4**, and **BAR5** can be configured for accessing the Address space across the NT Port Virtual and Link Interfaces.

Each **BARx Setup** register defines the memory window size that is to be assigned by a system enumerator (*that is*, BIOS or firmware). The smallest size allowed by PCI rules is 1 MB. Furthermore, the **BARx Setup** registers define a space as a 32- or 64-bit Memory space, and whether the space is I/O, Non-Prefetchable Memory, or Prefetchable Memory space.

*For example*, if the window size must be 1 MB in a Prefetchable Memory space, the **BARx Setup** register in 32-bit Address space is FFF0\_0008h. The “FFF” value on bits [31:20] provides the size, 1 MB in this case. The 0\_000 on bits [19:4] are **Reserved**. Bit 3 denotes whether the space is prefetchable – a value of 1 indicates that it is prefetchable (*that is*, cacheable). Bits [2:1] allow a choice of 32- or 64-bit Address space, 00b indicates 32 bit.

**Note:** Only the **BAR2** and **BAR4 Setup** registers are programmable for bits [2:1] – if programmed for 64-bit Address space, then **BAR2** and **BAR3** work together as **BAR2/3**, and/or **BAR4** and **BAR5** work together as **BAR4/5**.

In most cases, the **BARx Setup** registers are programmed using an optional serial EEPROM (if present), before the BIOS or firmware allocates the resources (because resource enumeration is done before the system software can access these devices). I<sup>2</sup>C can also be used to program the **BARx Setup** registers, as long as it finishes before PCI Express enumeration completes.

Table 14-2. NT Port Virtual and Link Interface BARs

BAR	NT Port Virtual Interface Description	NT Port Link Interface Description
<b>BAR0</b>	<p>All PEX 8748 Port Configuration registers are mapped into Memory space, using <b>BAR0</b> and <b>BAR1</b>. The Virtual side Host, connected to the Transparent Upstream Port(s), can use the Transparent Upstream Port(s) <b>BAR0/1</b> or NT Port Virtual Interface <b>BAR0/1</b> to access the PEX 8748 Port Configuration registers. The <b>NT Port Virtual Interface BAR0/1 Setup</b> register controls the <b>BAR0</b> and <b>BAR1</b> implementation, as follows:</p> <ul style="list-style-type: none"> <li>Disables <b>BAR0</b> and <b>BAR1</b></li> <li>Enables <b>BAR0</b> and disables <b>BAR1</b> (<b>BAR0</b> is a 32-bit BAR)</li> <li>Enables <b>BAR0</b> and <b>BAR1</b> (<b>BAR0/1</b> is a 64-bit BAR)</li> </ul> <p><b>BAR0</b> and <b>BAR1</b> claim 256-KB Memory space to the system.</p>	<p>All PEX 8748 Port Configuration registers are mapped into Memory space, using <b>BAR0</b> and <b>BAR1</b>. The Link side Host, connected to the NT Port, can use <b>BAR0/1</b> to access the PEX 8748 Port Configuration registers. The <b>NT Port Link Interface BAR0/1 Setup</b> register controls the <b>BAR0</b> and <b>BAR1</b> implementation, as follows:</p> <ul style="list-style-type: none"> <li>Disables <b>BAR0</b> and <b>BAR1</b></li> <li>Enables <b>BAR0</b> and disables <b>BAR1</b> (<b>BAR0</b> is a 32-bit BAR)</li> <li>Enables <b>BAR0</b> and <b>BAR1</b> (<b>BAR0/1</b> is a 64-bit BAR)</li> </ul> <p><b>BAR0</b> and <b>BAR1</b> claim 256-KB Memory space to the system.</p>
<b>BAR1</b>	Configured by the <b>NT Port Virtual Interface BAR0/1 Setup</b> register. <b>BAR1</b> is implemented as an upper 32-bit address of the NT Port Virtual Interface memory-mapped 64-bit BAR; otherwise, it is <i>Reserved</i> .	Configured by the <b>NT Port Link Interface BAR0/1 Setup</b> register. <b>BAR1</b> is implemented as an upper 32-bit address of the NT Port Link Interface memory-mapped 64-bit BAR; otherwise, it is <i>Reserved</i> .
<b>BAR2</b>	Configured by the <b>NT Port Virtual Interface Memory BAR2 Setup</b> register. <b>BAR2</b> can be implemented as a 32-bit BAR or lower half of a 64-bit BAR, by combining it with <b>BAR3</b> ( <b>BAR2/3</b> ). <b>BAR2</b> uses Direct Address Translation.	Configured by the <b>NT Port Link Interface Memory BAR2 Setup</b> register. <b>BAR2</b> can be implemented as a 32-bit BAR or lower half of a 64-bit BAR, by combining it with <b>BAR3</b> ( <b>BAR2/3</b> ). <b>BAR2</b> uses Direct Address Translation.
<b>BAR3</b>	Configured by the <b>NT Port Virtual Interface Memory BAR2/3 Setup</b> register. <b>BAR3</b> can be implemented as a 32-bit BAR or upper half of a 64-bit BAR, by combining it with <b>BAR2</b> ( <b>BAR2/3</b> ). <b>BAR3</b> uses Direct Address Translation.	Configured by the <b>NT Port Link Interface Memory BAR2/3 Setup</b> register. <b>BAR3</b> can be implemented as a 32-bit BAR or upper half of a 64-bit BAR, by combining it with <b>BAR2</b> ( <b>BAR2/3</b> ). <b>BAR3</b> uses Direct Address Translation.
<b>BAR4</b>	Configured by the <b>NT Port Virtual Interface Memory BAR4 Setup</b> register. <b>BAR4</b> can be implemented as a 32-bit BAR or lower half of a 64-bit BAR, by combining it with <b>BAR5</b> ( <b>BAR4/5</b> ). <b>BAR4</b> uses Direct Address Translation.	Configured by the <b>NT Port Link Interface Memory BAR4 Setup</b> register. <b>BAR4</b> can be implemented as a 32-bit BAR or lower half of a 64-bit BAR, by combining it with <b>BAR5</b> ( <b>BAR4/5</b> ). <b>BAR4</b> uses Direct Address Translation.
<b>BAR5</b>	Configured by the <b>NT Port Virtual Interface Memory BAR4/5 Setup</b> register. <b>BAR5</b> can be implemented as a 32-bit BAR or upper half of a 64-bit BAR, by combining it with <b>BAR4</b> ( <b>BAR4/5</b> ). <b>BAR5</b> uses Direct Address Translation.	Configured by the <b>NT Port Link Interface Memory BAR4/5 Setup</b> register. <b>BAR5</b> can be implemented as a 32-bit BAR or upper half of a 64-bit BAR, by combining it with <b>BAR4</b> ( <b>BAR4/5</b> ). <b>BAR5</b> uses Direct Address Translation.



## 14.1.10 TLP Routing

To understand what structures are required in an NTB, a brief overview of PCI routing is provided.

Memory and I/O Requests are routed by comparing an address against the PCI-to-PCI bridge **Base** and **Limit** registers. Any Memory or I/O Requests, that fall within the Base and Limit, route out the Downstream side of that one PCI-to-PCI bridge. All Downstream devices must be mapped in contiguous address regions, such that a single Address range (between the Base and Limit, inclusive) in each space is sufficient.

Completions and Configuration Requests are routed, using a Bus Number that is within the range of the PCI-to-PCI bridge's **Secondary Bus Number** and **Subordinate Bus Number** values (offsets 19h and 1Ah, respectively, within the **Bus Number** register (offset 18h)). PCI uses a tree of Bus Numbers, with the Upstream Port(s) having the lowest Bus Number. Each PCI-to-PCI bridge has:

- An Upstream Primary Bus Number,
- A Downstream Secondary Bus Number, and
- A range of other Downstream buses, up to a Subordinate Bus Number.

Similar to the Base and Limit comparison for Memory and I/O Requests, any Completions or Configuration Requests, that fall within the Secondary and Subordinate Bus Number range, route out the Downstream side of that one PCI-to-PCI bridge.

TLPs that target the PEX 8748, however, are treated a bit differently. Memory Requests must hit the BAR space. Completions or Configuration Requests must exactly match the Bus Number of a PCI-to-PCI bridge within the PEX 8748.

Upstream routing is simply the negation of Downstream and switch routing. If a packet does not route Downstream or hit the PEX 8748, the packet must route Upstream.

### 14.1.11 Address Translation

In multi-domain systems, each Host domain has its own Address space and Bus Numbering scheme, independent from that of other Host domain(s). Hence, any transaction crossing the inter-domain boundary, by way of an NTB or other means, must support address, as well as Bus Number, translations.

Before a transaction (PCI Express packet) can go through the NTB (either from the Virtual to Link side, or from the Link to Virtual side) in a multi-domain system, one or more sets of Memory resources must be assigned to the NTB. To request this resource from the system enumerator (BIOS or firmware), the NTB must be programmed with the **BARx Setup** register(s). (Refer to [Section 14.1.9.1](#).) The **BARx Setup** register(s) requests the window space size, memory type, 32- or 64-bit Address space, and prefetchable or non-prefetchable area, using one 32-bit register for 32-bit Address space or the two 32-bit registers (**BAR2/3** or **BAR4/5**) for 64-bit Address space. In return, the system enumerator assigns resources to the NTB in **BAR0** through **BAR5**. Any transactions that target **BAR2** through **BAR5** on the NT Port Link Interface result in a transaction across the NTB, to the secondary address domain.

The NTB must be enumerated to accommodate the resources assigned to the NT endpoint, to allow packets to logically traverse the bridge. Its Device Number (on the Downstream Internal Virtual PCI Bus) value is the NT Port Number. Device enumeration minimally includes the **PCI Command**, **Bus Number**, and **Memory Base and Limit** and/or **Prefetchable Memory Base and Limit** registers (offsets 04h, 18h, and 20h, and/or 24h). The Internal NT Virtual Bus (connecting the NT PCI-to-PCI secondary interface and NT endpoint, as illustrated in [Figure 14-2](#)) can be assigned any available Bus Number within the Upstream Port's range of Subordinate Bus Numbers.

In addition to the BARs, re-maps for the other side (Virtual or Link) must be programmed. The **Memory BARx Address Translation** registers provide memory translation, with a one-to-one correspondence between **BAR2** through **BAR5**. This is fairly straightforward:

address{Host1\_abc} changes to address{Host2\_abc}

where *abc* represents a lower Address bit offset from the BAR. As previously mentioned, the Host1 and Host2 Address spaces can independently be a 32- or 64-bit Address space; the address translation accounts for any alteration.

The Requester ID Lookup Table (RID-LUT) is a little more complicated, because it processes more information. The TLP's Requester ID is used to select an entry in the RID-LUT to translate. The Requester ID is composed of the following:

- 8-bit Bus Number,
- 5-bit Device Number, and
- 3-bit Tag

The Bus Number must change as it crosses Host domains, and there must be a mechanism to change it back (for Completions); therefore, the egress TLP needs a way to route back to the initial Requester. That way is the RID-LUT, where the egress TLP is sent with an index to the RID-LUT for the return trip Bus Number recovery. Because there is an association of initial Requester ID into the RID-LUT, a Requester can be prevented from crossing the NTB if it does not have an entry enabled in the RID-LUT, adding security to the NTB, by limiting the devices that can generate transactions across the NTB.

The Address Translation registers and RID-LUT can be changed during runtime, as long as they are not being used by pending transactions.

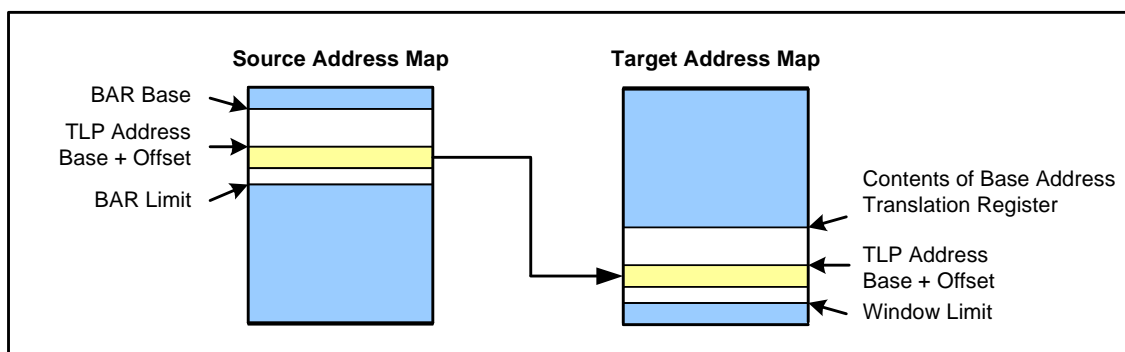
The NT Port Virtual and Link Interfaces support [Direct Address Translation](#), described in the following section.

### 14.1.11.1 Direct Address Translation

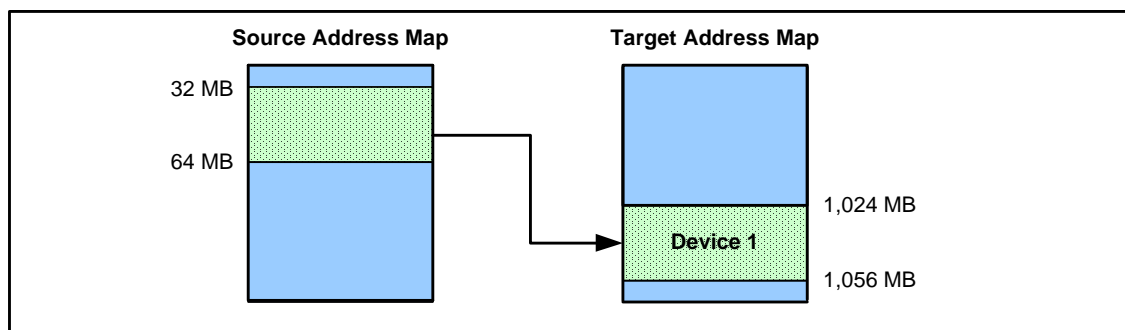
The **BAR<sub>x</sub> Setup** registers define a mask that splits the address into an upper *Base* field and a lower *Offset* field. Translation then consists of replacing, under the maskable portion of the **BAR<sub>x</sub> Setup** register, the Address Base register bits with the corresponding Address Translation register bits. Accordingly, the Address Translation register value must be a multiple of the corresponding BAR size. [Figure 14-4](#) illustrates Direct Address Translation.

The device(s) on the originating Host domain can communicate to a single device or multiple devices mapped to consecutive Memory Address space on the Target Host domain, by using the Direct Address Translation mechanism. [Figure 14-5](#) illustrates the entire Address map, claimed by the NT Port, mapped into the single target device. [Figure 14-6](#) illustrates the entire Address map claimed by the NT Port, mapped into multiple target devices. Multiple devices must be in contiguous Memory Address ranges.

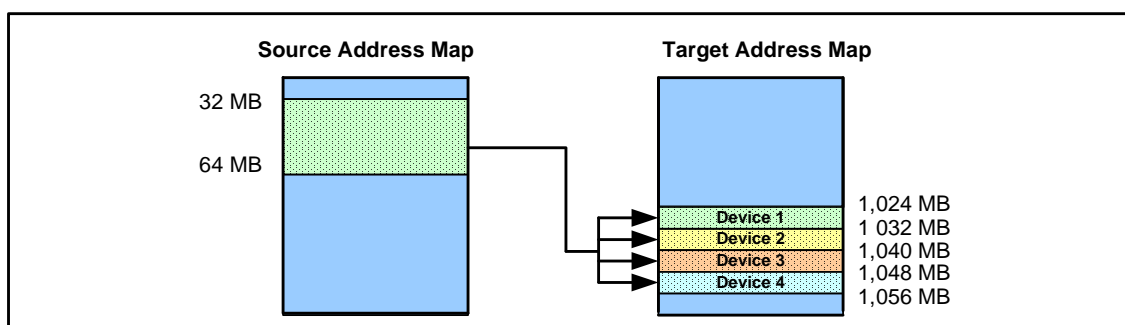
**Figure 14-4. Direct Address Translation**



**Figure 14-5. NT Port Mapped into Single Target Device**



**Figure 14-6. NT Port Mapped into Multiple Target Devices**



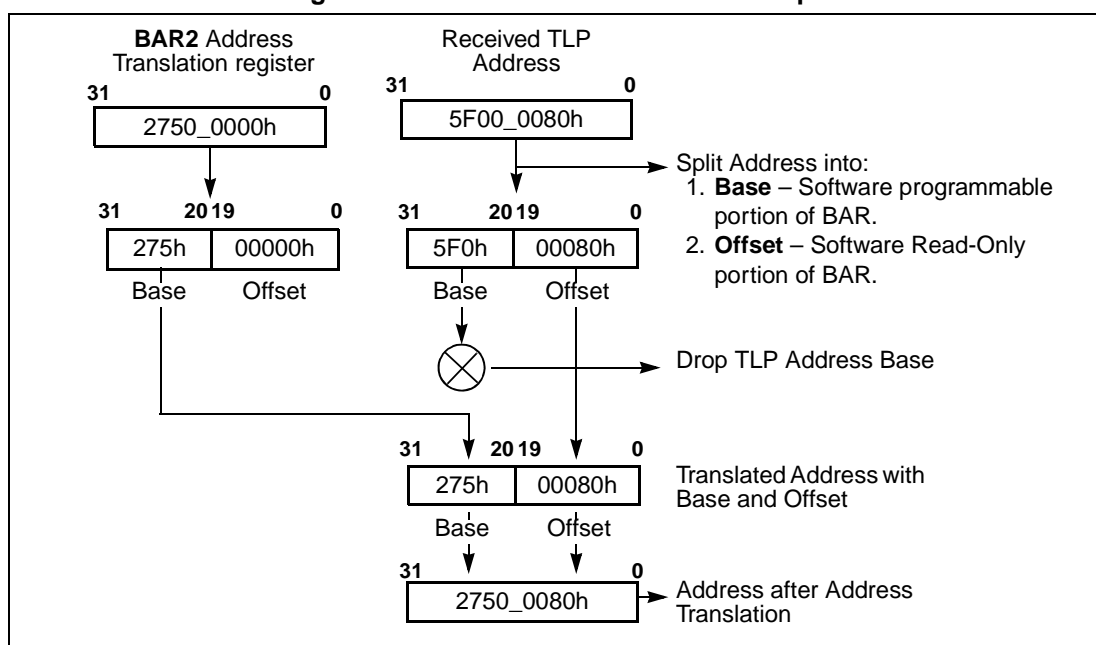
**Address Translation Example**

Assume the following initial conditions:

1. NT Port Virtual Interface **BAR2** claims 1-MB Memory space, by Setting bits [31:20] (**BAR2 Setup** register = FFF0\_0000h).
2. Enumeration software on the Virtual side assigns the 5F00\_0000h address value to NT Port Virtual Interface **BAR2** (an endpoint from the software point of view) and makes the PCI-to-PCI bridges Upstream of that endpoint have a Base and Limit that include that **BAR2** space.
3. Device driver software programs the **BAR2** Address Translation register to 2750\_0000h.

The PEX 8748 receives a transaction to the NT Port Virtual Interface, with address 5F00\_0080h. The received transaction address is hitting the NT Port Virtual Interface **BAR2**. The PEX 8748 claims the transaction and executes the address translation described in Figure 14-7.

**Figure 14-7. Address Translation Example**



## 14.2 Requester ID Translation

Some Messages, and Configuration and Completion transactions, are Bus Number-routed, instead of address-routed. Of these, the NT Port forwards only the Completion transaction between the two Host domains. PCI Express switches and bridges use the Requester ID (defined in the Completion TLP Header) to route these packets.

The Requester ID consists of the following:

- Requester's PCI Bus Number
- Device Number
- Function Number

The Completer ID consists of the following:

- Completer's PCI Bus Number
- Device Number
- Function Number

**Note:** The PCI Bus Number is unique for each Host domain.

Figure 14-8 illustrates the Memory Request TLP Header format. Figure 14-9 illustrates the Completion TLP Header format.

**Figure 14-8. Memory Request TLP Header Format**

	Byte 0								Byte 1								Byte 2								Byte 3							
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Bytes 0-3	R	Fmtx1		Type				R	TC			R			TD	EP	Attr		R		Length											
Bytes 4-7	Requester ID												Tag						Last DW BE				1st DW BE									
Bytes 8-11	Address[63:32]																															
Bytes 12-15	Address[31:0]																															R

**Figure 14-9. Completion TLP Header Format**

	Byte 0								Byte 1								Byte 2								Byte 3							
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Bytes 0-3	R	Fmt		Type					R	TC			R			TD	EP	Attr		R		Length										
Bytes 4-7	Completer ID																Completer Status		BCM	Byte Count												
Bytes 8-11	Requester ID																Tag						R	Lower Address								

## 14.2.1 Transaction Sequence

*To implement a memory sequence from a Requester in Host1 space to a Target in Host2 space:*

1. Hardware inserts its Requester ID information into the Memory Read TLP that it generates on the initiating Host1 domain.
2. Switches and bridges between the transaction initiator and NT Port route this Memory Read TLP, based upon the Memory address.
3. NT Port replaces the Memory Read TLP Requester ID with its own Requester ID, and conducts the address translation. The NT Port's Requester ID consists of the Host2 Bus Number of the NT Port and an index to the RID-LUT, as well as three original bits. This modified TLP is sent to the target Host, Host2, domain.
4. Switches and bridges between the NT Port and target device route this Memory Read TLP, based upon the address and using Host2 Address space.
5. A Host2 target is found, and a Completion is generated. When the target device generates the Completion TLP, it copies the Memory Read TLP Requester ID (which was the NT Port Bus Number with RID-LUT index) into the corresponding Completion's *TLP Requester ID* field and inserts its ID into the *TLP Completer ID* field.
6. Switches and bridges between the target device and NT Port route the Completion TLP, based upon the Host2 Bus Numbering scheme and Bus Number in the *Requester ID* field, back to the NT Port.
7. NT Port restores the original Requester ID value from the RID-LUT pointed to by the RID-LUT index, replaces the Completer ID with the NT Port's Host1 Completer ID (using the Host1 Bus Number), and forwards the modified Completion TLP to the original Requester in the Host1 domain.
8. Switches and bridges between the NT Port and Requester route the Completion TLP, based upon the Bus Number in the *Requester ID* field.
9. Requester accepts and processes the Completion TLP.

Transactions originating in both the Virtual- and Link-side Host domains use the same mechanism and have the same hardware resources. Therefore, any transactions use the same processes.

Both directions have their own unique RID-LUT with 32 entries, as detailed in:

- [Section 15.13.2, “NT Port Virtual Interface NT Bridging-Specific Registers – Requester ID Translation Lookup Table Entry \(Addresses D94h – DD0h\)”](#)
- [Section 16.13.2, “NT Bridging-Specific Registers – Requester ID Translation Lookup Table Entry \(Addresses DB4h – DF0h\)”](#)

This data structure supports up to 32 devices, each of which can simultaneously transmit Requests through the associated NT Port. Because the Function Number is not used in the RID-LUT association, a separate RID-LUT entry is not required for each requesting or phantom function device.

The RID-LUT must be configured before transmitting Requests through the NT Port. This Requester registration process, which cannot be accomplished by a peer, is an effective security and protection mechanism.

When a Request is received and routed to the NT Port, its Requester ID is translated – Bus Number and Device Number, but not Function Number. The 3-bit Function Number remains the same on both sides of the NTB. The remainder of the Requester ID is translated, based upon the RID-LUT. The received Memory Request TLP Requester ID is associated into the RID-LUT, to find a single matching entry. The address of this matching entry (RxIndex) is substituted into the 5-bit *Device Number* field of the Memory Request's TLP *Requester ID* field. The NT Port's Bus Number is substituted into the 8-bit *Bus Number* field.

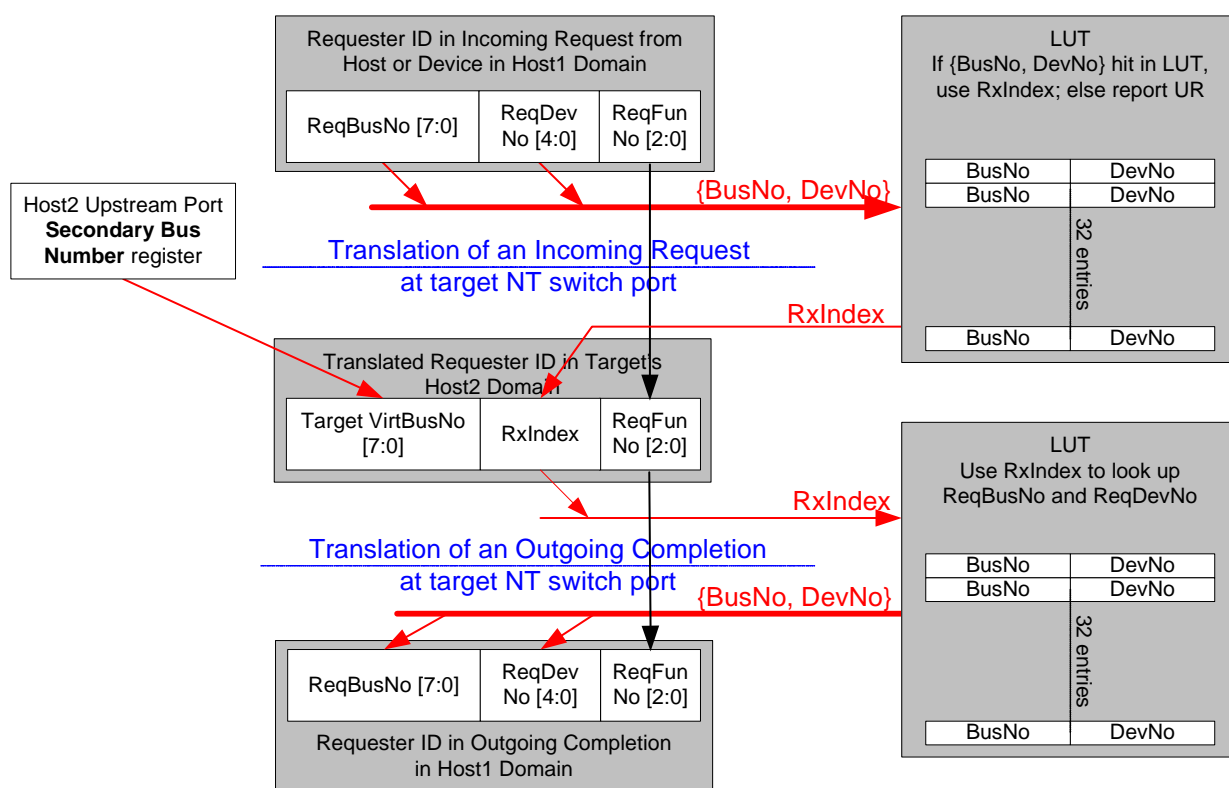
If no match is found, or the matched entry is not enabled, the Request receives an Unsupported Request (UR). If a match is found, and matched entry is enabled, the translated Memory Request TLP is address-translated and forwarded into the Virtual side Host domain.

The NT Port's Bus Number is sufficient to route the Completion from the Completer back to the NT Port in the Completer's domain, because the NT Port is the only possible Requester on that bus segment (because the NT Port is an endpoint). Elsewhere in the PCI Express hierarchy, the Bus Number is sufficient to route the Completion back into the switch that contains the NT Port.

The inverse translation occurs, when a Completion passes through the NTB. The RxIndex is retrieved from the *Device Number* field of the received Completion TLP *Requester ID* Header field and used to look up the 32-entry RID-LUT. The Completion TLP *Requester ID* values for the *Bus Number* and *Device Number* fields are replaced by the decoded RID-LUT-entry Bus Number and Device Number values, if the entry is valid; otherwise, an Unexpected Completion is reported.

The Completion TLP Completer ID is replaced by the NT Port Bus Number, Device Number, and Function Number values before forwarding the Completion TLP to the Link side Host domain. (Refer to Figure 14-10.)

**Figure 14-10. Requester ID Translation for Request Originating in Host1 Domain**



## 14.3 NT Port Power Management Handling

### 14.3.1 Active State Power Management

The NT Port Link Interface endpoint supports the ASPM L0s and L1 Link PM states, as well as the PCI Bus PM L2/L3 Ready state. The NT Port Virtual Interface endpoint implements the Configuration Space registers for ASPM support. However, it does not enter into the low-power states, because there is no physical Link associated with it.

***Note:** The NT Port Virtual Interface has no Link. Therefore, it never enters any Link PM states. From the Upstream scoreboard point of view, the NT Port Virtual Interface emulates an endpoint device in the DL\_Down condition.*

#### 14.3.1.1 L0s Link PM State Transition – NT Port Link Interface

L0s Link PM state entry is mandatory, and this state is allowable only in Active State Link Power Management (ASPM). Each device reports its support to L0s entry through the Port's **Link Capability** register *Active State Power Management (ASPM) Support* field (offset 74h[11:10]). Software enables L0s state entry, by writing into Active state Link PM Control register. The L0s Link PM state is optimized for short entry and exit latencies, while providing substantial power savings. If the L0s state is enabled in a device, transmit Links that are not being used must be brought into the L0s state.

Link Layer Timers are *not* affected by a transition to the L0 Link PM state. When in the L0s state, Links return to the L0 state, regardless of whether there are any TLP traffic situations to transmit Flow Control (FC) updates, which occurs every 30  $\mu$ s minimum.

##### ***Tx L0s Link PM State Entry Conditions – NT Port Link Interface***

The conditions for the NT Port Link Interface to enter the L0s state are similar to an endpoint's L0s state entry conditions:

1. No pending TLPs (including Retry TLPs), or there are no credits available to transmit pending TLPs.
2. No DLLP pending.
3. Enabled in the **Link Capability** register *Active State Power Management (ASPM) Support* field (NT Port Link Interface, offset 74h[11:10], programmed to 01b or 11b), for going into the L0s state.

The PM module directs the NT Link side Port to go into the L0s state when Conditions 1 and 2 are met for a specified period of time and Condition 3 is satisfied. The idle time can be selected, as either 1  $\mu$ s (default) or 4  $\mu$ s, in the **Power Management Hot Plug User Configuration** register *L0s Entry Idle Count* bit (NT Port Link Interface, offset F70h[0]).

##### ***L0s Link PM State Exit Conditions – NT Port Link Interface***

Components from either end of a PCI Express Link can initiate exit from the L0s Link state. The following conditions must be met for the NT Port Link Interface to exit the L0s state:

- If there are pending DLLPs or TLPs to transmit
- Link enters retrain



### 14.3.1.2 L1 Link PM State Entry – NT Port Link Interface

The L1 state is a deeper power-saving state than the L0s Link PM state, with both Tx and Rx Lanes settling into Electrical Idle after L1 negotiation is complete. Entry into the L1 state is always initiated by an Upstream Port or Downstream device. ASPM L1 state is optional, and software-controlled, by writing into the *Active State Power Management (ASPM)* field. However, L1 state entry is also possible in the PCI Express PCI-PM, when an Upstream Port is placed into a state other than the PCI Express PCI-PM D0 state. This transition is required. Conditions for entering the ASPM L1 state are implementation-specific. Downstream Ports never initiate L1 state entry.

A Port cannot directly go from the L0s to L1 state while in ASPM. In ASPM, the transition is L0s -> L0 -> L1. After L1 state entry, Flow Control Update Timers are suspended.

PCI-PM-compatible L1 entry protocol uses the PM\_enter\_L1\_DLLP; however, ASPM uses PM\_active\_state\_request\_L1\_Dllp. Downstream Ports never NAK the PM\_enter\_L1\_DLLP received from a Downstream device; however, it can NAK PM\_active\_state\_request\_L1\_Dllp, if there is something to transmit.

The NT Port Link Interface requests PCI Express PCI-PM L1 state entry, when the following six conditions are met:

- Software places the NT Port Link Interface into the PCI Express PCI-PM D3hot state.
- NT Port Link Interface suspends the scheduling of new TLPs, after a Completion for that particular packet is transmitted Upstream.
- NT Port Link Interface waits until all previously transmitted TLPs are acknowledged. During this time, the NT Port Link Interface receives TLPs and DLLPs. Also transmits ACK/NAK and FC DLLPs.
- NT Port Link Interface waits for a sufficient quantity of FC credits, for all types of TLP packets.
- NT Port Link Interface waits for all the Retry TLPs to complete.
- After the NT Port Link Interface transmits all pending TLPs, it transmits a PM\_EnterL1\_Dllp Upstream, with no more than a 4-symbol time gap.

When the NT Port receives the PM\_Request\_Ack DLLP, it directs the Link to enter the L1 state.

When in the L1 state, if an Electrical Idle exit is detected, or a TLP must be transmitted, the NT Port exits the L1 state, transmits the pending TLP(s), if required, then re-enters the PCI Express PCI-PM L1 state after a 16-μs delay. The NT Port must detect a continuous 16 μs of the PHY L0 state for the next state entry.

### ***L1 Link PM State Entry Negotiation Interruption – ASPM and PCI Express PCI-PM – NT Port Link Interface***

Per the *PCI Express Base r3.0*, when L1 state entry negotiation is interrupted, the state machine on both ends of the PCI Express Link should return to an Idle state. A Port can be in any of the states described in the sections that follow, when L1 state entry negotiation is interrupted, usually as the result of a trip through the *Recovery* state.

What occurs after L1 state entry negotiation is interrupted on the NT Port Link Interface is dependent upon the following:

- **Started requesting L1 entry, using PM\_request DLLPs** – When the Tx or Rx *Recovery* state occurs, the Port stops receiving Requests, returns to an Idle state, then waits 10  $\mu$ s before making another L1 state entry Request, –or– the Tx Lane should enter the L0s Link PM state.
- **Waiting for a PM request ACK** – When the Tx or Rx *Recovery* state occurs, the Port returns to an Idle state, then waits 10  $\mu$ s before making another L1 state entry Request, –or– the Tx Lane should enter the L0s Link PM state.
- **After receiving a PM Request ACK, directs the PHY to the L1 state, and the PHY has not reached the L1 state and the PHY enters the Tx or Rx *Recovery* state** – The Port returns to an Idle state, then waits 10  $\mu$ s before making another L1 state entry Request, –or– the Tx Lane should enter the L0s Link PM state.
- **After the PHY transmits an EIOS and settles in the Tx L1 state, the Rx has not yet reached the L1 state, and an Rx *Recovery* state occurs or the Port's Link Control register *Retrain Link* bit (Downstream Ports, offset 78h[5]) is Set** – The Port directs the PHY Tx to exit the L1 state.

### 14.3.1.3 L2/L3 Link PM State Entry – NT Port Link Interface

The L2 and L3 states are the staging points for removing main power. Support for the L2/L3 Ready transition protocol is required.

The L2 and L3 states are related to the PCI Express PCI-PM D-state transitions. L2/L3 Ready is the state in which a given Link enters when the platform is preparing to enter its system sleep state. Following completion of the L2/L3 Ready state transition protocol, the Link enters the L2/L3 Ready state. After main power is removed, the Link settles into the L3 state. The time for the L2/L3 Ready state entry transition is indicated by completion of the PME\_Turn\_Off/PME\_TO\_Ack handshake sequence. Any actions on the part of the Downstream component necessary to ready itself for power loss must be completed prior to initiating the transition to the L2/L3 Ready state. After all preparations for loss of power and clock are complete, the Downstream component initiates L2/L3 Ready state entry, by transmitting the PM\_EnterL23 DLLP Upstream.

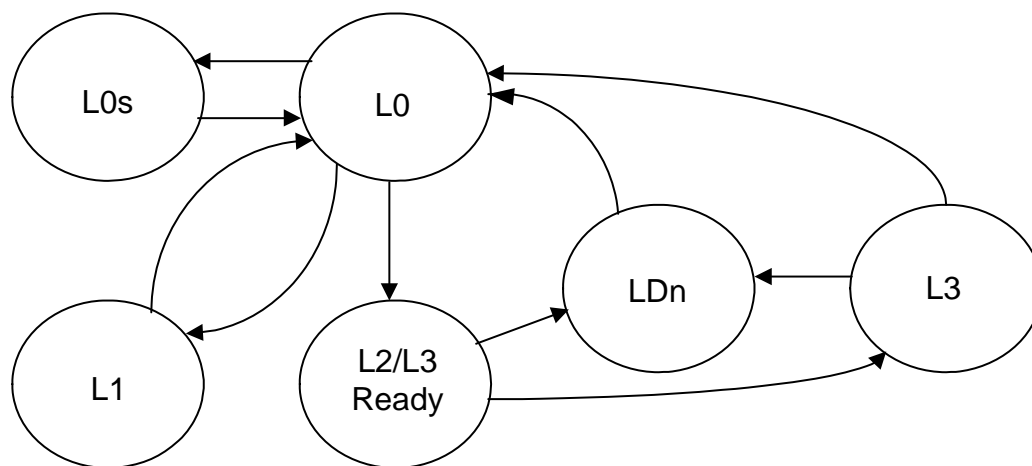
Exit from the L2/L3 Ready state, back to the L0 Link PM state, can be initiated only by an Upstream-initiated transaction that targets the Downstream component (Upstream Electrical Idle exit) in the same manner that an Upstream-initiated transaction trigger would trigger the transition from the L1 to L0 Link PM states.

When the NT Port Link Interface receives a PME\_TOFF\_MSG, it behaves the same way as an endpoint. The NT Port finishes the pending TLPS and ACK/NAK DLLPs, transmits a PME\_TO\_Ack and subsequently transmits L2/L3 Request DLLPs, then enters the L2/L3 Ready state.

When the NT Port Link Interface is in the L2/L3 Ready state, Rx Electrical Idle exit causes the Link to transition to the LDn state, then subsequently enter the *Detect* state.

Figure 14-11 illustrates the Link PM state transitions.

**Figure 14-11. Link PM State Transitions**



## 14.3.2 PCI-PM and PME Turn Off Support

When NT mode is enabled, the NT Port Link Interface behaves like any other endpoint in the PCI Express PCI-PM D3hot state. After entering the D3hot state, the NT Port Link Interface requests PCI-PM L1 Link PM state entry, then finally settles into the L1 Link PM state. Only Configuration accesses and Messages to the NT Port Link Interface are supported in the D3hot state. The Root Complex transmits PME\_Turn\_Off Messages when the NT Host decides to turn Off the main power and Reference Clock. The NT Port Link Interface indicates its readiness to lose power, by transmitting a PME\_TO\_Ack Message to the Upstream device. The PME\_TO\_Ack Message is transmitted when there are no pending TLPs to transmit Upstream, toward the NT Port Link Interface. The Port requests the L2/L3 Ready Link PM state, by transmitting PM\_Enter\_L23 Data Link Layer Packets (DLLPs) to the Upstream device after transmitting a PME\_TO\_Ack TLP. The Port settles into the L3 Link PM state when the Power Controller removes the main power and Reference Clock.

When the PME\_Turn\_Off Message is received on the Transparent Upstream Port(s), the Port broadcasts this Message to all Downstream devices, including the NT Port Virtual Interface. After the PME\_TO\_Ack Message is received from all Downstream devices and the NT Port Virtual Interface, the Transparent Upstream Port(s) transmit(s) an aggregated PME\_TO\_Ack Message to the Upstream component after it finishes transmitting all pending TLPs to the Upstream component. When NT mode is enabled, the Transparent Downstream Ports allow the attached devices to enter the PCI-PM-compatible L1 Link PM state. The NT Port Virtual Interface never enters the PCI-PM L1 Link PM state.

## 14.3.3 Message Generation

The NT Port Link Interface never generates PM\_PME Messages. The NT Port Virtual Interface never receives Set\_Slot\_Power\_Limit Messages nor generates PM\_PME Messages.

## 14.4 Expansion ROM

The NT Port Link Interface supports Expansion ROM, by default. Expansion ROM support can be moved from the NT Port Link Interface to the NT Port Virtual Interface, by Setting the **Ingress Chip Control** register *Expansion ROM Virtual Side* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 764h[0]).

The NT Port supports 16- or 32-KB-sized Expansion ROM, based upon the **Serial EEPROM Clock Frequency** register *Expansion ROM Size* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 268h[16]) value.

***Note:** Expansion ROM can be enabled in either the NT Port Virtual or Link Interface, but not both simultaneously. Expansion ROM is enabled, by default, in the NT Port Link Interface (**Ingress Chip Control** register *Expansion ROM Virtual Side* bit is Cleared).*

*Expansion ROM can be disabled, by Setting the **Ingress Control** register *Disable NT Port Expansion ROM BAR* bit (Upstream Port(s), offset F60h[15]).*

## 14.5 NT Port Interrupts

The NT Port Virtual and Link Interfaces can both generate interrupts in response to specific events. The NT Port must not receive any INT<sub>x</sub> Message Requests. If the NT Port receives an INT<sub>x</sub> Message Request, the Request is handled as a Malformed TLP error.

### 14.5.1 NT Port Virtual Interface Interrupts

The NT Port Virtual Interface generates interrupts to the Virtual side Host for the following reasons (all are masked, by default, and must not be masked to be enabled):

- Doorbell interrupts
- NT Port Link Interface detected an Correctable TLP error
- NT Port Link Interface detected an Uncorrectable TLP error (option to signal Fatal and/or Non-Fatal)
- NT Port Link Interface *DL\_Active* state change
- NT Port Link Interface received an Uncorrectable Error Message

NT-Virtual Doorbell interrupts and NT-Link errors and events can use the INT<sub>x</sub>, MSI, or **PEX\_INTA#** (Base mode) or **VSx\_PEX\_INTA#** (Virtual Switch mode) signaling mechanisms (all mutually exclusive), as follows:

- **PEX\_INTA#** or **VSx\_PEX\_INTA#** output can be enabled for NT Port Virtual Interface Doorbell interrupts, by Setting the **ECC Error Check Disable** register *Enable PEX\_INTA# or VSx\_PEX\_INTA# Interrupt Output for NT-Virtual Doorbell-Generated Interrupts* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 720h[7])
- **PEX\_INTA#** or **VSx\_PEX\_INTA#** output can be enabled for NT-Link Error and Event interrupts, by Setting the **ECC Error Check Disable** register *Enable PEX\_INTA# or VSx\_PEX\_INTA# Interrupt Output for Device-Specific NT-Link Port Event-Triggered Interrupts* bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 720h[5])

Refer to [Section 14.1.7](#) for Doorbell interrupt details.

The NT Port Virtual Interface de-asserts INTx, or PEX\_INTA# or VSx\_PEX\_INTA#, interrupts in response to one or more of the following conditions:

- **PCI Command** register *Interrupt Disable* bit (NT Port Virtual Interface, offset 04h[10]) is Set
- Corresponding *Interrupt Mask* bit is Set
- NT Port Link goes down (DL\_Down condition) or the NT Port Link Interface receives a Hot Reset
- Software Clears the *Interrupt Status* bit, or Sets the *Doorbell Interrupt Request Clear* bit

For tracking purposes, an NT Port Virtual Interface-generated interrupt is treated like an event that is external to the PCI-to-PCI bridge. If software asserts a **Secondary Bus Reset** from this PCI-to-PCI bridge, the PCI-to-PCI bridge de-asserts the NT Port Virtual Interface interrupt.

When the NT Port Link Interface detects Correctable TLP errors, the NT Port Virtual Interface signals the interrupt to the Virtual side Host, if the interrupt signaling is enabled and not masked (**Link Error Status Virtual** register *Link Side Correctable Error Status* bit is Set, and **Link Error Mask Virtual** register *Link Side Correctable Error Mask* bit is Cleared (NT Port Virtual Interface, offsets FE0h[0] and FE4h[0]), respectively).

When the NT Port Link Interface detects Uncorrectable TLP errors, the NT Port Virtual Interface signals the interrupt to the Virtual side Host, if the interrupt signaling is enabled and not masked (**Link Error Status Virtual** register *Link Side Uncorrectable Error Status* bit is Set, and **Link Error Mask Virtual** register *Link Side Uncorrectable Error Mask* bit is Cleared (NT Port Virtual Interface, offsets FE0h[1] and FE4h[1]), respectively).

An NT Port Link Interface *DL\_Active* state change occurs upon detection of an NT Port Link Interface *DL\_Down* state rise edge and fall edge. This signals the interrupt to the Virtual side Host, if the interrupt signaling is enabled and not masked (**Link Error Status Virtual** register *Link Side DL Active Change Status* bit is Set, and **Link Error Mask Virtual** register *Link Side DL Active Change Mask* bit is Cleared (NT Port Virtual Interface, offsets FE0h[2] and FE4h[2]), respectively).

When the NT Port Link Interface receives an Uncorrectable Error Message, the NT Port Virtual Interface signals the interrupt to the Virtual side Host, if the interrupt signaling is enabled and not masked (**Link Error Status Virtual** register *Link Side Uncorrectable Error Message Drop Status* bit is Set, and **Link Error Mask Virtual** register *Link Side Uncorrectable Error Message Drop Mask* bit is Cleared (NT Port Virtual Interface, offsets FE0h[3] and FE4h[3]), respectively).

## 14.5.2 NT Port Link Interface Interrupts

The NT Port Link Interface generates interrupts to the Link side Host for NT-Link Doorbell interrupts detected at the NT Port Link Interface ingress Port (interrupts are masked, by default). The NT Port Link Interface should not detect any Device-Specific errors (because these errors are reported to the Upstream Port Host to handle).

NT-Link Doorbell interrupts and Device-Specific errors can use either INTx or MSI signaling mechanisms (both mutually exclusive). Refer to [Section 14.1.7](#) for Doorbell interrupt details.

The NT Port Link Interface de-asserts INTx interrupts in response to one or more of the following conditions:

- **PCI Command** register *Interrupt Disable* bit (NT Port Link Interface, offset 04h[10]) is Set
- Corresponding *Interrupt Mask* bit is Set
- NT Port Link goes down (DL\_Down condition) or the NT Port Link Interface receives a Hot Reset
- Software Clears the *Interrupt Status* bit, or Sets the *Doorbell Interrupt Request Clear* bit

## 14.6 NT Port Error Handling

The NT Port Virtual Interface endpoint logs TLP errors, for TLPs that travel from the NT Port Virtual Interface to the NT Port Link Interface. The PEX 8748 signals Error Messages to the Virtual side Host. The PEX 8748 provides an option to communicate this error condition to the Link side Host, by signaling an interrupt.

The NT Port Link Interface endpoint logs TLP errors, for TLPs that travel from the NT Port Link Interface to the NT Port Virtual Interface. The PEX 8748 signals Error Messages to the Link side Host.

When the PEX 8748 receives a TLP and before it transmits the TLP through the NT Port, the following is performed, in the sequence listed:

1. TLP integrity check.
2. Address decode.
3. Address translation.
4. Requester ID translation.
5. ECRC re-generation.

If the PEX 8748 detects an ECRC error, it corrupts the re-generated ECRC before transmitting the TLP through the NT Port. The PEX 8748 also provides options for dropping error-detected poisoned or ECRC TLPs (**Ingress Control** register *Drop Poisoned TLPs* and *Drop ECRC TLPs* bits (Upstream Port(s), offset F60h[9:8], respectively)).

The PEX 8748 does not generate the ECRC for a TLP that passes through the NT Port, if the received TLP does not have its *TD* bit Set.

The PEX 8748 drops all TLPs traveling from the NT Port Virtual Interface to the NT Port Link Interface, if the internal RAM Fatal Error-Correcting Code (ECC) error is detected, until the PEX 8748 receives a Hot Reset from a Virtual side Host.

### 14.6.1 NT Port Link Interface Error Handling

If the NT Port Link Interface receives an Uncorrectable Error Message, it reports a Malformed TLP error, by default. However, if the **Ingress Chip Control** register *NT Error Message Drop* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 764h[17]) is Set, the NT Port Link Interface drops the Message, and logs the error into the **Link Error Status Virtual** register *Link Side Uncorrectable Error Message Drop Status* bit (NT Port Virtual Interface, offset FE0h[3]). If the corresponding **Link Error Mask Virtual** register *Link Side Uncorrectable Error Message Drop Mask* bit (NT Port Virtual Interface, offset FE4h[3]) is Set, the NT Port Virtual Interface signals an interrupt (INTx, MSI, or **PEX\_INTA#** (Base mode) or **VSx\_PEX\_INTA#** (Virtual Switch mode)) to the Virtual side Host through the Upstream Port(s), if interrupts are enabled.

## 14.6.2 NT Virtual Side Error Handling

For the NT Port Virtual Interface to generate an Error Message, the following error-forwarding *Enable* bits, in both the NT PCI-to-PCI bridge and the Upstream Port(s), must be Set:

- **PCI Command** register (offset 04h)
  - *SERR# Enable* (bit 8)
- **Bridge Control** register (offset 3Ch)
  - *SERR# Enable* (bit 17)
- **Device Control** register (offset 70h)
  - *Correctable Error Reporting Enable* (bit 0)
  - *Non-Fatal Error Reporting Enable* (bit 1)
  - *Fatal Error Reporting Enable* (bit 2)

When the NT Port Virtual Interface generates an Error Message, the NT PCI-to-PCI bridge also logs the error status in the following NT mode registers, while sending the Message in the Upstream direction:

- **Secondary Status** register (offset 1Ch)
  - *Received System Error* (bit 30)
- **PCI Status** register (offset 04h)
  - *Signaled System Error* (bit 30)



## 14.7 Cursor Mechanism

A software application can use the [Device-Specific Cursor Mechanism](#) to access the PEX 8748 NT Port Configuration Space registers. The registers that support the Device-Specific Cursor Mechanism are the **Configuration Address Window** and **Configuration Data Window** registers (offsets **F8h** and **FCh**, respectively). A software application can also:

- Select the Configuration Register offset, by using the **Configuration Address Window** register
- Perform Read accesses to the **Configuration Data Window** register, to read to the selected Configuration register
- Perform Write accesses to the **Configuration Data Window** register, to write to the selected Configuration register

Configuration transactions have access to this Device-Specific Cursor Mechanism, if NT mode is enabled.

For details regarding the **Configuration Address Window** and **Configuration Data Window** registers, refer to:

- [Section 15.9, “NT Port Virtual Interface Vendor-Specific Capability 3 Registers \(Offsets C8h – FCh\)”](#)
- [Section 16.9, “NT Port Link Interface Vendor-Specific Capability 3 Registers \(Offsets C8h – FCh\)”](#)

## 14.8 Port Programmability

The PEX 8748 supports the capability of programming the VS0 Upstream Port and NT Port, by way of the **VS0 Upstream** register *VS0 Upstream Port* and *NT Port* fields (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset **360h**[4:0 and 12:8], respectively). This register is updated, based upon the external **STRAP\_NT\_UPSTRM\_PORTSEL**[2:0] input states, by default. I<sup>2</sup>C, software, and/or an optional serial EEPROM (if present) can be used to override the external Strap values.

A software application can change the VS0 Upstream Port and NT Port locations to another Port Number during runtime, or as part of a failover sequence. It is recommended that the PEX 8748 be in an Idle state (no traffic) when changing the VS0 Upstream Port and NT Port Numbers during runtime. During a failover sequence, application software must be able to handle all spurious TLPs that it receives as a result of the failover process.

## 14.9 NT Failover

To protect against a failing Host taking the entire system down, a backup Host can be in place, ready to take over. One method is to have a secondary Host behind an NTB and use an NT failover sequence.

This section describes four of the major failover scenarios:

- [Active-Passive NT Failover](#)
- [Active-Active NT Failover](#)
- [NT Failover for Virtual Switch Operation](#)
- [Failover Completion](#)

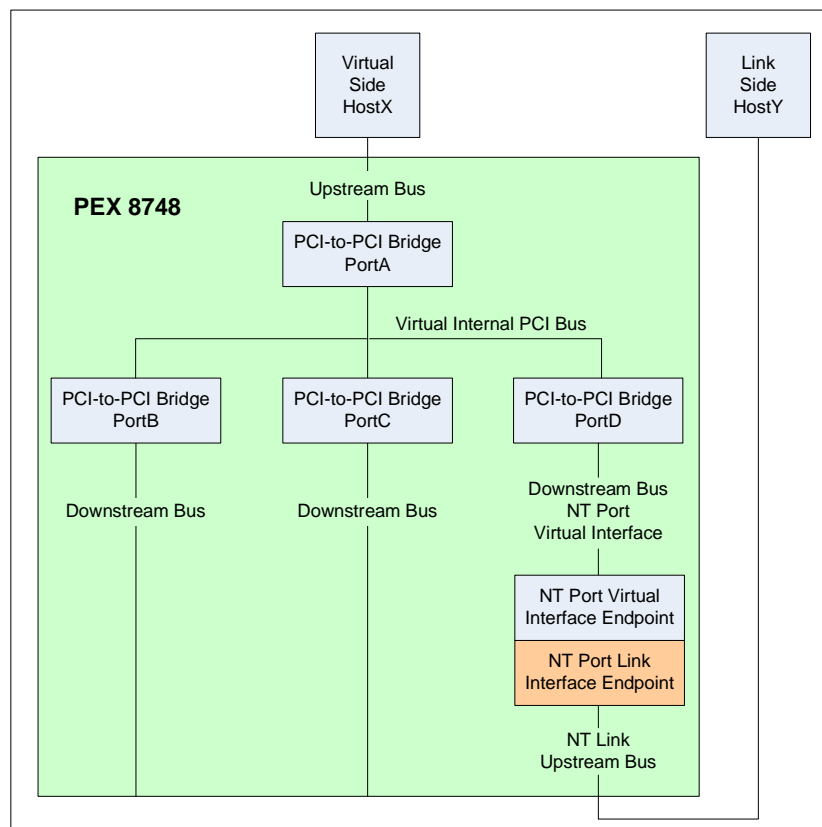
Additional failover information related to NT mode is discussed in [Section 12.5.3, “Virtual Switch Host to NT Host Failover.”](#)

### 14.9.1 Active-Passive NT Failover

In an active-passive Host scenario, the passive Host is waiting for the active Host to fail. The PEX 8748 might be configured, where initially the active HostX is the Virtual side Host connected to PortA and the passive HostY is the Link side Host connected to PortD. In this case (refer to [Figure 14-12](#)):

- HostX owns all the PEX 8748’s Downstream Ports and the NTB’s NT Port Virtual Interface endpoint
- HostY owns only the NTB’s Link side

**Figure 14-12. Example – Active-Passive NT Failover Host Scenario**

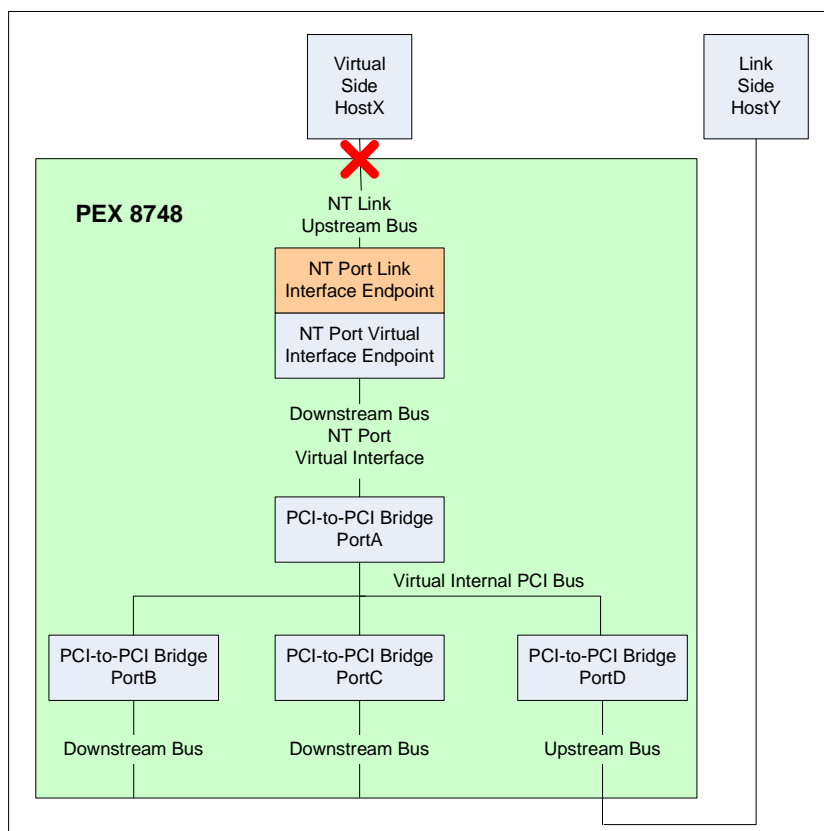


When the Virtual side Host (HostX) has failed and that failure is detected (by means beyond the scope of this data book) by the Link side Host (HostY), HostY initiates the following sequence:

1. Configure PortA to be the NT Port and PortD to be the Upstream Port.
  - This is a single CSR Write to the **VS Failover** register (NT Port Link Interface, offset F00h)
  - The register's **VS Upstream Port** field (field [4:0]) indicates the Upstream Port, and should be changed from the Port Number of PortA to the Port Number of PortD
  - The register's **VS NT Port** field (field [12:8]) indicates the NT Port, and should be changed from the Port Number of PortD to the Port Number of PortA
  - The register's **VS NT Enable** bit (bit 13) enables NT, and should remain Set
2. Reset the PEX 8748 through an Upstream Port's **Bridge Control** register *Secondary Bus Reset* bit (offset 3Ch[22]).
3. Re-enumerate the hierarchy and start operation.

Figure 14-13 illustrates how the PEX 8748 looks after this failover sequence. When HostX failure can be resolved, the red X goes away, and HostX becomes the passive standby backup processor, and the process is ready to repeat.

**Figure 14-13. Example – End Result of Active-Passive NT Failover Host Scenario**



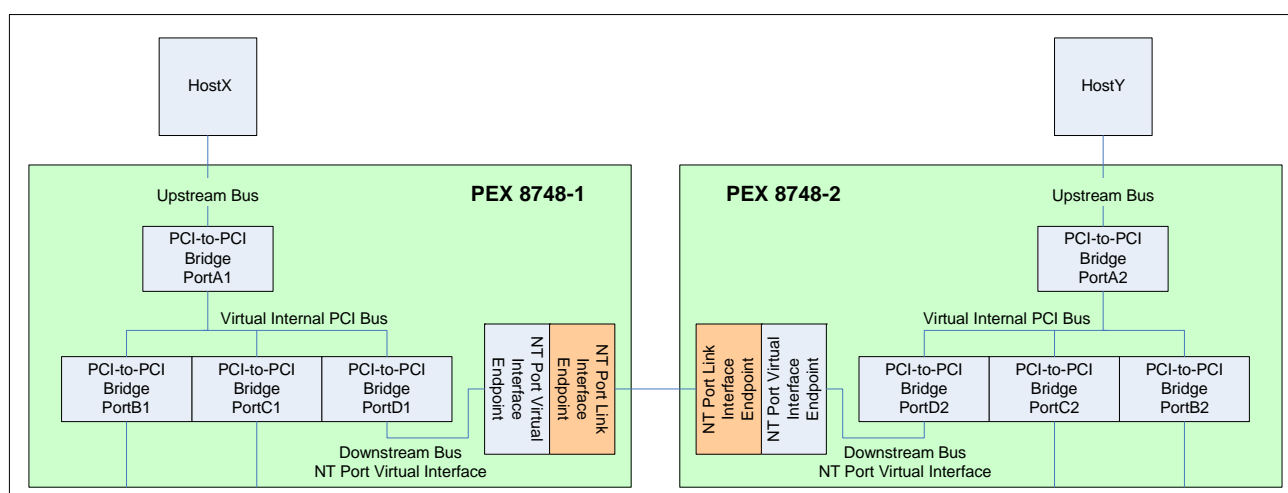
## 14.9.2 Active-Active NT Failover

In an active-active Host scenario, both Hosts are busy with their own hierarchies and ready to take over, should the other Host fail. A symmetric arrangement is the easiest to understand, but is not required. In [Figure 14-14](#), either Host can start the failover sequence. Initially:

- HostX owns all the PEX 8748-1 Downstream Ports and the NTB-1 NT Port Virtual Interface endpoint
- HostY owns all the PEX 8748-2 Downstream Ports and the NTB-2 NT Port Virtual Interface endpoint

Note that neither Host owns the Link between the two PEX 8748 switches nor the Link side endpoints. The default values for these endpoints use a Captured Bus Number of 0, which must be taken into consideration when programming the RID-LUT.

**Figure 14-14. Example – Active-Active NT Failover Host Scenario**

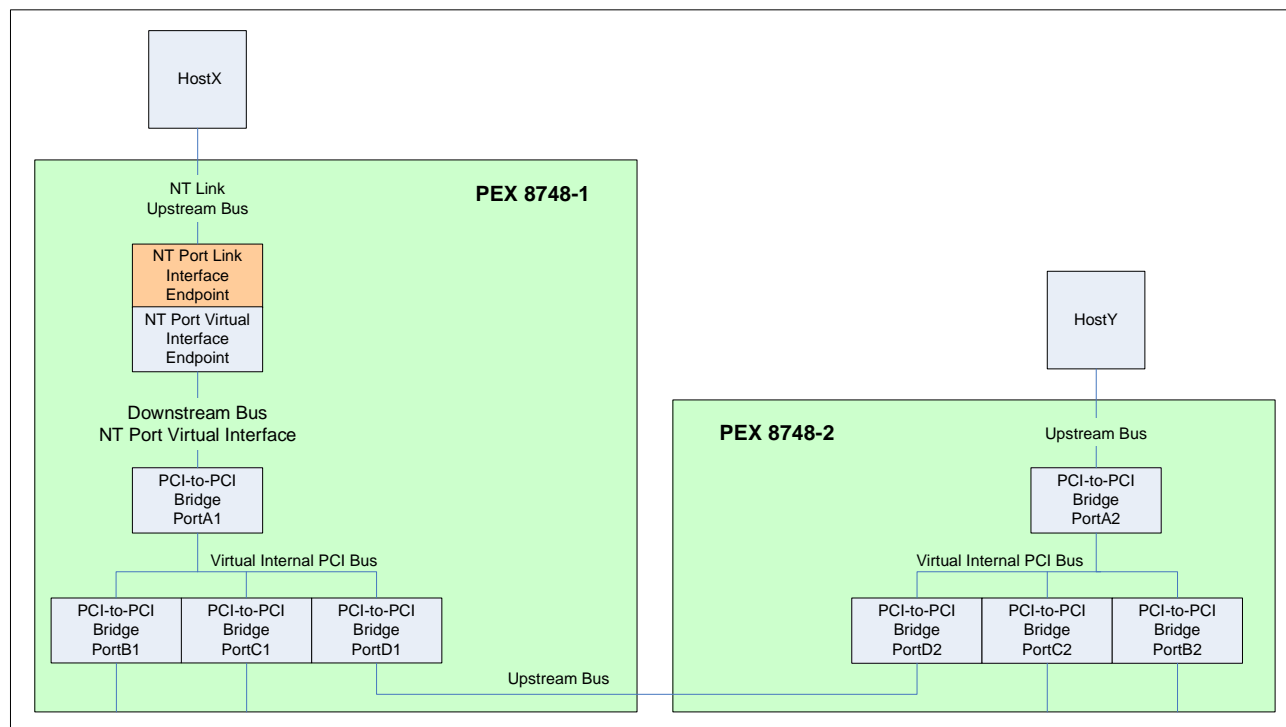


When one Host has determined that the other has failed (by means beyond the scope of this data book), it starts a failover sequence. For this example, assume HostX has failed.

1. HostY configures the PEX 8748-2 PortD2 to be a Transparent Downstream Port, by Clearing the **VS Failover** register *VS NT Enable* bit (Virtual Switch mode – VS Upstream Port(s), offset F00h[13]).
2. HostY configures the PEX 8748-1 PortD1 to be the Upstream Port, and PortA1 to be the NT Port, and NT enabled. This is a single CSR Write to the **VS Failover** register (Virtual Switch mode – NT Port Link Interface, on PEX 8748-1, offset F00h).
  - The register's *VS Upstream Port* field (field [4:0]) indicates the Upstream Port, and should be changed from A to D
  - The register's *VS NT Port* field (field [12:8]) indicates the NT Port, and should be changed from D to A
  - The register's *VS NT Enable* bit (bit 13) enables NT, and should remain Set
3. HostY resets the entire PEX 8748-1 hierarchy. It may also need to reset the PEX 8748-2 hierarchy, unless space in the Bus Numbering of PCI-to-PCI Bridge PortD2 was Reserved to inherit many new buses. The reset is Set through either the PEX 8748-1 Upstream Port's, or PEX 8748-2 PortD2's, **Bridge Control** register *Secondary Bus Reset* bit (offset 3Ch[22]).
4. Re-enumerate the hierarchy and start operation.

Post failover, the system resembles the example illustrated in [Figure 14-15](#). From the PEX 8748-1 perspective, it looks just like an active-passive failover. When HostX is repaired, it can negotiate with HostY when to resume control of the PEX 8748-1 hierarchy.

**Figure 14-15. Example – End Result of Active-Active NT Failover Host Scenario**



To return to failover protection, reverse the previous steps. Steps 4 through 6 can occur in parallel, because the two Hosts have independent scheduling.

1. HostX negotiates with HostY when to resume control of the PEX 8748-1 hierarchy, and thereby resume failover protection.
2. HostY resets the PEX 8748-2 PortD2 Downstream hierarchy, by Setting and then Clearing the PEX 8748-2 PortD2's **Bridge Control** register *Secondary Bus Reset* bit (offset 3Ch[22]).
3. HostX returns the PEX 8748-1 to its initial configuration, by programming PortA1 to be Upstream, PortD1 to be NT, and NT enabled. This is a single CSR Write to the PEX 8748-1's **VS Failover** register (Virtual Switch mode – NT Port Link Interface, offset F00h).
  - The register's *VS Upstream Port* field (field [4:0]) indicates the Upstream Port, and should be changed from D to A
  - The register's *VS NT Port* field (field [12:8]) indicates the NT Port, and should be changed from A to D
  - The register's *VS NT Enable* bit (bit 13) enables NT, and should remain Set
4. HostX resets the entire PEX 8748-1 hierarchy, by Setting and then Clearing the PEX 8748-1 PortA1's **Bridge Control** register *Secondary Bus Reset* bit (offset 3Ch[22]).
5. HostY adds the NT Port back into its hierarchy, by Setting the PEX 8748-2's **VS Failover** register *VS NT Enable* bit (Virtual Switch mode – VS Upstream Port(s), offset F00h[13]) (the NT Port assignment has not changed, only the enable has changed).
6. Both HostX and HostY re-enumerate their hierarchies and begin operation.

### 14.9.3 NT Failover for Virtual Switch Operation

In a single virtual switch system (Base mode), by definition, an NT Port and an Upstream Port exist in the same virtual switch hierarchy, because there is only one hierarchy. In a single virtual switch, an NT Port can automatically promote itself to be the Upstream Port, and it can assign a new NT Port to replace itself. This same functionality is available with multiple roots and multiple virtual switch hierarchies, with the one restriction being that the NT Port must belong to the same virtual switch as the Upstream Port it is trying to replace in the failover operation.

The first step for NT failover with multiple virtual switches then, is to have the NT Port belong to the same virtual switch as the failed Upstream Port (this process is described later in this section). The Management Port is the only Port that can control virtual switch assignment. The **Management Port Control** register (Port 0, accessible through the Management Port and Redundant Management Port, offset 354h) has bits that are used for assigning the virtual switch for the NT Port. If the NT Port needs to migrate to another virtual switch, the Management Port modifies the Virtual Switch Number for the NT Port.

The Management Port also assigns which other Port(s) belong to which virtual switch, in the **VSx Port Vector** register(s) (Port 0, accessible through the Management Port, offset(s) 380h through 394h), and controls which Ports are the Upstream Ports, per virtual switch. There are physically six separate **VSx Upstream** registers that control the Upstream and NT Port assignments, one per virtual switch. The Management Port sees all six registers.

After the NT and Upstream Port are in the same virtual switch, the NT failover process is identical to single virtual switch (Base mode) NT failover. The NT Port does not need to know to which virtual switch it belongs, or whether multiple virtual switches are enabled. Hardware will force the NT Port to see only the one **VSx Upstream** register that matches its virtual switch register. The hardware automatically links the NT register to the correct **VSx Upstream** register, because a Read on the **VSx Upstream** register sees the change that the NT Port makes on the **NT Upstream** register in NT Link space. The NT Port reads from and writes to the **VS Failover** register *VS Upstream Port* field (VS Upstream Port(s), offset F00h[4:0]), and hardware points it to the single correct **VSx Upstream** register.

The VS0 Upstream Port ID, NT Port ID, and NT Enable are all programmable, by any privileged source in that virtual switch, whether it is the Link side NT Host or Virtual side Upstream Host. (Refer to the **VS0 Upstream** register *VS0 Upstream Port*, *NT Port*, and *NT Enable* fields/bits (Port 0, accessible through the Management Port, offset 360h[4:0, 12:8, and 13], respectively, for further details.)

The NT Port, as the first step for failover, writes its NT Link side **VS Failover** register (VS Upstream Port(s), offset F00h), to simultaneously promote itself to be the Upstream Port while downgrading the (broken) Upstream Port to a Downstream Port or the new (replacement) NT Port. The new Upstream Port will commonly be the previous NT Port ID, which by definition is in the same virtual switch hierarchy, but hardware does not care which Port it is, as long as it is located within the same virtual switch. (Refer to [Section 14.9](#) for subsequent failover steps.)

## 14.9.4 Failover Completion

When the Management Port knows that traffic is quiesced for planned or unplanned events, it can re-program the owner of the Downstream Port(s). (The method used for determining the new Port assignments to the virtual switch is beyond the scope of this data book.) The Management Port does two Memory Writes to the **VSx Port Vector** register(s) (Port 0, accessible through the Management Port, offset(s) 380h through 394h) – one to disable the virtual switch that failed and one to enable the new virtual switch.

The Host that received the new Ports receives a Hot Plug Message, informing it of the new Ports. Alternately, the Management Host can message the applicable Host regarding the new Ports.

Depending upon the system, it may not be necessary to re-enumerate the recently acquired Ports. If control over the bus numbering is available, then each Downstream Port can be assigned unique Bus Number ranges (across all virtual switches), thus allowing any Host to own any Downstream Port without creating a Bus Number conflict.

If bus numbering control is not available, re-enumeration is required. It is possible that the entire virtual switch requires re-enumeration, from the Root Port down, because the Primary, Secondary, and Subordinate Bus Numbers of each PCI-to-PCI bridge might need to be changed.





## Chapter 15 NT Port Virtual Interface Registers – NT Mode

### 15.1 Introduction

**Note:** Check the latest design guides, application notes and errata list for Non-Transparent (NT) usage.

In NT mode, the NT Port includes two sets of Configuration, Capability, Control, and Status registers, to support the Virtual and Link Interfaces. This chapter defines the PEX 8748 NT Port Virtual Interface registers. Other registers are defined in:

- [Chapter 13, “Transparent Port Registers”](#)
- [Chapter 16, “NT Port Link Interface Registers – NT Mode”](#)

All PEX 8748 registers can be accessed by Configuration or Memory Requests.

For further details regarding register names and descriptions, refer to the following specifications:

- *PCI r3.0*
- *PCI Power Mgmt. r1.2*
- *PCI Express Base r3.0*

## 15.2 NT Port Virtual Interface Type 0 Register Map

Table 15-1 defines the NT Port Virtual Interface Type 0 register mapping.

**Table 15-1. NT Port Virtual Interface Type 0 Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		
NT Port Virtual Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch)			00h
			...
			34h
			...
			3Ch
		Next Capability Pointer (48h)	Capability ID (01h)
NT Port Virtual Interface PCI Power Management Capability Registers (Offsets 40h – 44h)			40h
		Next Capability Pointer (68h)	Capability ID (05h)
NT Port Virtual Interface MSI Capability Registers (Offsets 48h – 64h)			44h
			48h
			...
			64h
		Next Capability Pointer (C8h)	Capability ID (10h)
NT Port Virtual Interface PCI Express Capability Registers (Offsets 68h – A0h)			68h
			...
			A0h
<b>Reserved</b>			A4h – C4h
		Next Capability Pointer (00h)	Capability ID 3 (09h)
NT Port Virtual Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh)			C8h
			...
			FCh
Next Capability Offset (FB4h)	1h	PCI Express Extended Capability ID (0003h)	100h
Device Serial Number Extended Capability Registers (Offsets 100h – 108h)			...
			108h
<b>Reserved</b>			10Ch – 144h
Next Capability Offset 2 (C34h)	1h	PCI Express Extended Capability ID (0002h)	148h
NT Port Virtual Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)			...
			1BCh
NT Port Virtual Interface Device-Specific Registers (Offsets 1C0h – A74h)			1C0h
			...
			A74h
<b>Reserved</b>			A78h – B6Ch

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																				
Next Capability Offset 2 (000h)																1h				PCI Express Extended Capability ID 2 (000Bh)																B70h
NT Port Virtual Interface Device-Specific Registers (Offsets B70h – C88h)																																...				
Next Capability Offset 4 (B70h)																1h				PCI Express Extended Capability ID 4 (000Bh)																C34h
NT Port Virtual Interface Device-Specific Registers (Offsets B70h – C88h)																																...				
NT Port Virtual Interface NT Bridging-Specific Registers (Offsets C8Ch – DFCh)																																C88h				
NT Port Virtual Interface NT Bridging-Specific Registers (Offsets C8Ch – DFCh)																																C8Ch				
Multicast Extended Capability Registers (Offsets E00h – E2Ch)																																...				
Multicast Extended Capability Registers (Offsets E00h – E2Ch)																																E00h				
Multicast Extended Capability Registers (Offsets E00h – E2Ch)																																...				
Multicast Extended Capability Registers (Offsets E00h – E2Ch)																																E2Ch				
<i>Reserved</i>																																E30h –				
Next Capability Offset (148h)																1h				PCI Express Extended Capability ID (0001h)																FB0h
NT Port Virtual Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)																																FB4h				
NT Port Virtual Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)																																...				
NT Port Virtual Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)																																FDCh				
NT Port Virtual Interface Device-Specific Registers – Link Error (Offsets FE0h – FFCh)																																FE0h				
NT Port Virtual Interface Device-Specific Registers – Link Error (Offsets FE0h – FFCh)																																...				
NT Port Virtual Interface Device-Specific Registers – Link Error (Offsets FE0h – FFCh)																																FFCh				

## 15.3 Register Access

The PEX 8748 NT Port Virtual Interface implements a 4-KB Configuration Space. The lower 256 bytes (offsets 00h through FFh) comprise the PCI-compatible Configuration Space, and the upper 960 DWords (offsets 100h through FFFh) comprise the PCI Express Extended Configuration Space. The PEX 8748 supports three mechanisms for accessing the NT Port Virtual Interface registers:

- [PCI Express Base r3.0 Configuration Mechanism](#)
- [Device-Specific Memory-Mapped Configuration Mechanism](#)
- [Device-Specific Cursor Mechanism](#)

### 15.3.1 *PCI Express Base r3.0* Configuration Mechanism

The *PCI Express Base r3.0* Configuration mechanism is divided into two mechanisms:

- [PCI r3.0-Compatible Configuration Mechanism](#) – Provides Conventional PCI access to the first 256 bytes (the bytes at offsets 00h through FFh) of the NT Port Virtual Interface Configuration Register space
- [PCI Express Enhanced Configuration Access Mechanism](#) – Provides access to the entire 4-KB Configuration Space

Both are described in the sections that follow.

#### 15.3.1.1 *PCI r3.0-Compatible* Configuration Mechanism

The *PCI r3.0-Compatible* Configuration mechanism provides standard access to the PEX 8748 NT Port Virtual Interface's first 256 bytes (the bytes at offsets 00h through FFh) of the PCI Express Configuration Space. (Refer to [Figure 15-1](#).)

The mechanism uses PCI Type 0 and Type 1 Configuration transactions to access the PEX 8748 Configuration registers. All Ports capture the Bus Number and Device Number assigned by the Upstream device on the PCI Express Link attached to the PEX 8748 Upstream Port, as required by the *PCI Express Base r3.0*.

The PEX 8748 decodes all Type 1 Configuration accesses received on its Upstream Port, when any of the following conditions are met:

- If the Bus Number specified in the Configuration access is the Bus Number of the PEX 8748 internal virtual PCI Bus, the PEX 8748 automatically converts the Type 1 Configuration access into the appropriate Type 0 Configuration access for the specified device.
  - If the specified device corresponds to the NT Port Virtual Interface (or to the PCI-to-PCI bridge in one of the PEX 8748 Transparent Downstream Ports), the PEX 8748 processes the Read or Write Request
  - If the specified Device Number does not correspond to any of the PEX 8748 Downstream Port Device Numbers nor NT Port Number, the PEX 8748 responds with an Unsupported Request (UR)

This mechanism uses the same Request format as the [PCI Express Enhanced Configuration Access Mechanism](#). For PCI-compatible Configuration Requests, the *Extended Register Address* field must be all zeros (0).

Because the mechanism is limited to the first 256 bytes of the NT Port Virtual Interface Configuration register space, one of the following must be used to access beyond Byte FFh:

- [PCI Express Enhanced Configuration Access Mechanism](#)
- [Device-Specific Memory-Mapped Configuration Mechanism](#)
- [Device-Specific Cursor Mechanism](#)

NDA CONFIDENTIAL -- Inventec | King Michael

### 15.3.1.2 PCI Express Enhanced Configuration Access Mechanism

The PCI Express Enhanced Configuration Access mechanism (ECAM) uses a flat, Root Complex Memory-Mapped Address space to access the device Configuration registers. In this case, the Memory address determines the Configuration register accessed, and the Memory data returns the addressed register's contents. The Root Complex converts the Memory transaction into a Configuration transaction.

The mechanism can be used to access PEX 8748 registers that are defined by Specifications. This mechanism typically *cannot* access the Device-Specific registers (which are instead accessible by the Device-Specific Memory-Mapped Configuration Mechanism) in the following offset ranges, per Port:

- 200h through AFCh
- B0Ch through B6Ch
- B80h through C30h

Silicon Revision CA provides an option to enable ECAM access to all PEX 8748 registers, however, by Setting the **Station-Based Control** register *Device-Specific Register Access by Extended Configuration Request Enable* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 760h[16]), in the Station(s) that have an Upstream Port.

### 15.3.2 Device-Specific Memory-Mapped Configuration Mechanism

The Device-Specific Memory-Mapped Configuration mechanism provides a method to access the registers for all Ports within a single 256-KB Memory map, as illustrated in [Figure 15-1](#). This Memory map is identical for Upstream Port(s) **BAR0/1**, NT Port Virtual Interface **BAR0/1**, and NT Port Link Interface **BAR0/1**. The registers of each Port are located within a 4-KB range.

When the NT Port is enabled at Fundamental Reset, the NT Port Virtual and Link Interface registers use the *PCI r3.0* Type 0 Configuration Space Header. The NT PCI-to-PCI bridge (between the NT Port Virtual Interface and internal virtual PCI Bus) registers use the *PCI r3.0* Type 1 Configuration Space Header, and are mapped to the 4-KB Address space of the Port Number that is assigned as the NT Port (indicated in the **VS0 Upstream** register *NT Port* field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset **360h**[**12:8**])).

To use this mechanism, use the [PCI r3.0-Compatible Configuration Mechanism](#) to program the PEX 8748 NT Port Virtual Interface **Base Address 0** and **Base Address 1** registers (**BAR0** and **BAR1**, offsets **10h** and **14h**, respectively). After the PEX 8748 NT Port Virtual Interface Memory-Mapped register Base address is Set, the PEX 8748 Configuration Space registers are accessed, using Memory Reads and Writes to the 4-KB range, starting at offset 248 KB (**3\_E000h**, Virtual Interface) and offset 252 KB (**3\_F000h**, Link Interface).

**Figure 15-1. NT Mode Configuration Register Mapping to Memory-Mapped BAR**

<b>PEX 8748</b>	
NT Link 4-KB Configuration Space (3_F000h)	<b>BAR0 + 256 KB</b>
NT Virtual 4-KB Configuration Space (3_E000h)	<b>BAR0 + 252 KB</b>
	<b>BAR0 + 248 KB</b>
<b>Reserved</b>	
Port 19 4-KB Configuration Space (1_3000h)	
Port 18 4-KB Configuration Space (1_2000h)	
Port 17 4-KB Configuration Space (1_1000h)	
Port 16 4-KB Configuration Space (1_0000h)	<b>BAR0 + 64 KB</b>
<b>Reserved</b>	
Port 11 4-KB Configuration Space (0_B000h)	
Port 10 4-KB Configuration Space (0_A000h)	
Port 9 4-KB Configuration Space (0_9000h)	
Port 8 4-KB Configuration Space (0_8000h)	
<b>Reserved</b>	
Port 3 4-KB Configuration Space (0_3000h)	
Port 2 4-KB Configuration Space (0_2000h)	
Port 1 4-KB Configuration Space (0_1000h)	
Port 0 4-KB Configuration Space (0_0000h)	<b>BAR0 + 0 KB</b>

**Note:** *Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP\_STNx\_PORTCFGx input states and/or serial EEPROM programming.*

### 15.3.3 Device-Specific Cursor Mechanism

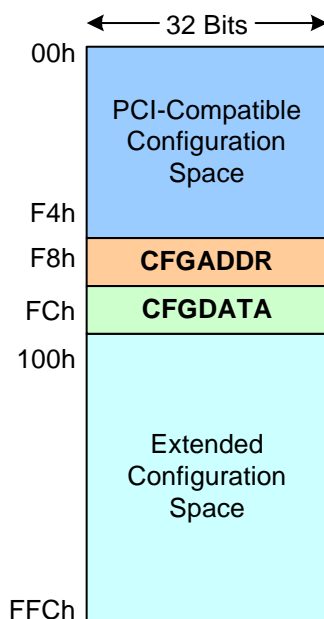
The Device-Specific Cursor mechanism is provided for use in development systems that can only generate *PCI r3.0* Configuration cycles (*that is*, the system cannot use either the [Device-Specific Memory-Mapped Configuration Mechanism](#), nor generate Extended Configuration Requests to access the Extended Configuration Space).

In [Figure 15-2](#), the software uses the **Configuration Address Window** (CFGADDR) register (offset **F8h**) to point to the NT Port Virtual or Link Interface Configuration Space registers, including the PCI Express Extended Configuration Space registers (offsets 100h through FFFh).

Software uses the **Configuration Data Window** (CFGDATA) register (offset **FCh**) to write to or read from the selected Configuration Space registers.

Refer to [Section 15.9](#), “NT Port Virtual Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh),” for the register descriptions.

**Figure 15-2. Configuration Space View**





## 15.4 Register Descriptions

The remainder of this chapter details the PEX 8748 NT Port Virtual Interface registers, including:

- Bit/field names
- Description of register functions in the PEX 8748 NT Port Virtual and Link Interfaces
- Type (*such as* RW or HwInit; refer to [Table 13-4, “Register Types, Grouped by User Accessibility,”](#) for Type descriptions)
- Whether the power-on/reset value can be modified, by way of the PEX 8748 serial EEPROM and/or I<sup>2</sup>C/SMBus Initialization feature
- Default power-on/reset value

## 15.5 NT Port Virtual Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch)

This section details the NT Port Virtual Interface PCI-Compatible Type 0 Configuration Header registers. [Table 15-2](#) defines the register map.

Table 15-2. NT Port Virtual Interface PCI-Compatible Type 0 Configuration Header Register Map

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																
Device ID																Vendor ID																00h
PCI Status																PCI Command																04h
PCI Class Code																PCI Revision ID																08h
PCI BIST <i>(Not Supported)</i>				PCI Header Type												Master Latency Timer <i>(Not Supported)</i>								Cache Line Size				0Ch				
Base Address 0																																10h
Base Address 1																																14h
Base Address 2																																18h
Base Address 3																																1Ch
Base Address 4																																20h
Base Address 5																																24h
<i>Reserved</i>																																28h
Subsystem ID																Subsystem Vendor ID																2Ch
Expansion ROM Base Address																																30h
<i>Reserved</i>																								Capability Pointer (40h)								34h
<i>Reserved</i>																																38h
Max_Lat <i>(Reserved)</i>								Min_Gnt <i>(Reserved)</i>								PCI Interrupt Pin								PCI Interrupt Line								3Ch

### Register 15-1. 00h PCI Configuration ID

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Vendor ID</b> Identifies the device manufacturer. Defaults to the PCI-SIG-issued Vendor ID of PLX (10B5h), if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	10B5h
31:16	<b>Device ID</b> Identifies the particular device. Defaults to the PLX part number for the PEX 8748, if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	87B0h

**Register 15-2. 04h PCI Command/Status**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Command</b>				
0	<b>I/O Access Enable</b> The PEX 8748 does not claim I/O resources, nor does it forward I/O transactions through the NT Port. The value of this register is “Don’t Care.”	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8748 ignores Memory Space Requests on the NT Port Virtual Interface 1 = PEX 8748 accepts Memory Space Requests received on the NT Port Virtual Interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8748 forwarding of Memory Requests Upstream. Does not affect Message forwarding nor Completions. 0 = PEX 8748 handles Memory Requests received on the NT Port Link Interface as Unsupported Requests (UR); for Non-Posted Requests, the PEX 8748 returns a Completion with UR Completion status 1 = PEX 8748 forwards Memory Requests in the Upstream direction	RW	Yes	0
3	<b>Special Cycle Enable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
4	<b>Memory Write and Invalidate Enable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
5	<b>VGA Palette Snoop</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
6	<b>Parity Error Response Enable</b> Controls bit 24 ( <i>Master Data Parity Error Detected</i> ).	RW	Yes	0
7	<b>IDSEL Stepping/Wait Cycle Control</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
8	<b>SERR# Enable</b> Controls bit 30 ( <i>Signaled System Error</i> ). 1 = Enables reporting of Fatal and Non-Fatal errors detected by the NT Port Virtual Interface to the Root Complex	RW	Yes	0
9	<b>Fast Back-to-Back Transactions Enable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
10	<b>Interrupt Disable</b> 0 = NT Port Virtual Interface is enabled to generate INTx Interrupt Messages 1 = NT Port Virtual Interface is prevented from generating INTx Interrupt Messages	RW	Yes	0
15:11	<b>Reserved</b>	RsvdP	No	0-0h

**Register 15-2. 04h PCI Command/Status (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Status</b>				
18:16	<b>Reserved</b>	RsvdP	No	000b
19	<b>Interrupt Status</b> 0 = No INTx interrupt is pending 1 = INTx interrupt is pending internally to the NT Port Virtual Interface, –or– PEX_INTA# (Base mode) or VSx_PEX_INTA# (Virtual Switch mode) (if enabled) is asserted	RO	No	0
20	<b>Capability List</b> Capability function is supported. Set, as required by the <i>PCI Express Base r3.0</i> .	RO	Yes	1
21	<b>66 MHz Capable</b> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
22	<b>Reserved</b>	RsvdP	No	0
23	<b>Fast Back-to-Back Transactions Capable</b> <b>Not supported</b> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
24	<b>Master Data Parity Error Detected</b> If bit 6 ( <i>Parity Error Response Enable</i> ) is Set, the NT Port Virtual Interface Sets this bit when the NT Port: <ul style="list-style-type: none"> <li>Forwards the poisoned Transaction Layer Packet (TLP) Write Request from the NT Port Link Interface to the NT Port Virtual Interface, –or–</li> <li>Receives a Completion marked as poisoned on the NT Port Virtual Interface</li> </ul> If the <i>Parity Error Response Enable</i> bit is Cleared, the PEX 8748 never Sets this bit. This error is natively reported by the <b>Uncorrectable Error Status</b> register <i>Poisoned TLP Status</i> bit (offset FB8h[12]), which is mapped to this bit for Conventional PCI backward compatibility.	RW1C	Yes	0
26:25	<b>DEVSEL# Timing</b> <b>Not supported</b> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	00b
27	<b>Signaled Target Abort</b> The NT Port Virtual Interface Sets this bit when any of the following conditions are met: <ul style="list-style-type: none"> <li>NT Port Virtual Interface receives a Completion (from a Transparent Port) that has a Completion status of Completer Abort (CA), –or–</li> <li>NT Port Virtual Interface receives a Memory Request targeting a PEX 8748 register, and the Payload Length (indicated within the Memory Request Header) is greater than 1 DWord</li> <li>NT Port Virtual Interface receives a Memory Request targeting a PEX 8748 register address within a non-existent Port</li> <li>NT Port Virtual Interface receives a Memory Write Request targeting enabled NT Port Expansion ROM Address space (Expansion ROM Base Address Register (BAR), offset 30h)</li> </ul> <b>Note:</b> When Set during a forwarded Completion, the <b>Uncorrectable Error Status</b> register <i>Completer Abort Status</i> bit (offset FB8h[15]) is not Set.	RW1C	Yes	0

**Register 15-2. 04h PCI Command/Status (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
28	<b>Received Target Abort</b> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
29	<b>Received Master Abort</b> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
30	<b>Signaled System Error</b> If bit 8 ( <i>SERR# Enable</i> ) is Set, the NT Port Virtual Interface Sets this bit when transmitting an ERR_FATAL or ERR_NONFATAL Message to the Upstream Port.	RW1C	Yes	0
31	<b>Detected Parity Error</b> This error is natively reported by the <b>Uncorrectable Error Status</b> register <i>Poisoned TLP Status</i> bit (offset FB8h[12]), which is mapped to this bit for Conventional PCI backward compatibility.  1 = NT Port Virtual Interface received a Poisoned TLP, regardless of the bit 6 ( <i>Parity Error Response Enable</i> ) state	RW1C	Yes	0

**Register 15-3. 08h PCI Class Code and Revision ID**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Revision ID</b>				
7:0	<b>Revision ID</b> Unless overwritten by the serial EEPROM, returns the Silicon Revision (BAh), the PLX-assigned Revision ID for this version of the PEX 8748. The PEX 8748 Serial EEPROM register Initialization capability is used to replace the PLX Revision ID with another Revision ID.	RO	Yes	BAh
<b>PCI Class Code</b>				<b>068000h</b>
15:8	<b>Register-Level Programming Interface</b>	RO	Yes	00h
23:16	<b>Sub-Class Code</b> Other bridge devices.	RO	Yes	80h
31:24	<b>Base Class Code</b> Bridge devices.	RO	Yes	06h

**Register 15-4. 0Ch Miscellaneous Control**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Cache Line Size</b>				
7:0	<b>Cache Line Size</b> System Cache Line Size. Implemented as a RW field for Conventional PCI compatibility purposes and does not impact PEX 8748 functionality.	RW	Yes	00h
<b>Master Latency Timer</b>				
15:8	<b>Master Latency Timer</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	00h
<b>PCI Header Type</b>				
22:16	<b>Configuration Layout Type</b> Type 0 Configuration Header for the NT Port.	RO	No	00h
23	<b>Multi-Function Device</b> 0 = Single-function device 1 = Indicates multiple (up to eight) functions (logical devices), each containing its own, individually addressable Configuration Space, 256 DWords in size	RO	No	0
<b>PCI BIST</b>				
31:24	<b>PCI BIST</b> <i>Not supported</i> Built-In Self-Test (BIST) Pass or Fail.	RsvdP	No	00h

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

**Register 15-5. 10h Base Address 0  
(NT Port Virtual Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> By default, NT Port Virtual Interface <b>BAR0</b> is enabled and <b>BAR1</b> is disabled, to provide a 32-bit <b>BAR0</b> for register access. <b>BAR1</b> can be enabled (by serial EEPROM and/or I <sup>2</sup> C/SMBus), to provide a 64-bit <b>BAR0/1</b> , by programming the <b>NT Port Virtual Interface BAR0/1 Setup</b> register <b>BAR0/1 Enable</b> field (NT Port Virtual Interface, offset <b>D0h[1:0]</b> ) to 11b (which enables both <b>BAR0</b> and <b>BAR1</b> ).				
0	<b>Memory Space Indicator</b> When enabled, the Base Address register maps PEX 8748 Port Configuration registers into Memory space. <b>Note:</b> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> 00b = Base Address register is 32 bits wide and can be mapped anywhere in the 32-bit Memory space 10b = Base Address register is 64 bits wide and can be mapped anywhere in the 64-bit Address space All other encodings are <b>Reserved</b> .	ROS	Yes	00b
3	<b>Prefetchable</b> 0 = Base Address register maps the PEX 8748 Port Configuration registers into Non-Prefetchable Memory space	ROS	Yes	0
17:4	<b>Reserved</b>	RsvdP	No	0-0h
31:18	<b>Base Address 0</b> 256-KB-aligned Base address used for Memory-Mapped access to the 256-KB block of all PEX 8748 registers (4 KB, per Port).	RW	Yes	0-0h

**Register 15-6. 14h Base Address 1  
(NT Port Virtual Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Upper 32-Bit Address for Memory-Mapped BAR</b> For 64-bit addressing ( <b>BAR0/1</b> ), Base Address 1 ( <b>BAR1</b> ) extends Base Address 0 ( <b>BAR0</b> ) to provide the upper 32 Address bits when the <b>Base Address 0</b> register <b>Memory Map Type</b> field (offset <b>10h[2:1]</b> ) is programmed to 10b.	RW	Yes	0000_0000h
	RO when the <b>Base Address 0</b> ( <b>BAR0</b> ) register is not enabled as a 64-bit BAR ( <b>Memory Map Type</b> field (offset <b>10h[2:1]</b> ) is not programmed to 10b).	RO	Yes	0000_0000h

**Register 15-7. 18h Base Address 2  
(NT Port Virtual Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Implemented as a Memory BAR	RO	No	0
2:1	<b>Memory Map Type</b> 00b = Base Address register is 32 bits wide and can be mapped anywhere in the 32-bit Memory space 10b = Base Address register is 64 bits wide and can be mapped anywhere in the 64-bit Address space All other encodings are <i>Reserved</i> .	ROS	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	ROS	Yes	0
19:4	<i>Reserved</i>	RsvdP	No	0_000h
31:20	<b>Base Address 2</b> Resolution is 1 MB.	RW	Yes	000h

**Register 15-8. 1Ch Base Address 3  
(NT Port Virtual Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
<i><b>Note:</b> This register has RW privilege if <b>BAR2/3</b> is configured as a 64-bit BAR (<b>Base Address 2</b> register <b>Memory Map Type</b> field (NT Port Virtual Interface, offset 18h[2:1]), is programmed to 10b).</i>					
0	<b>Memory Space Indicator</b> <b>BAR3</b> can be used as an independent 32-bit only BAR, or as the upper 32 bits of 64-bit <b>BAR2/3</b> . 0 = Memory BAR – only value supported	Offset 18h[2:1]=00b	RO	No	0
		Offset 18h[2:1]=10b	RW	Yes	0
2:1	<b>Memory Map Type</b> 00b = Base Address register is 32 bits wide and can be mapped anywhere in the 32-bit Memory space All other encodings are <i>Reserved</i> .	Offset 18h[2:1]=00b	RO	No	00b
		Offset 18h[2:1]=10b	RW	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	Offset 18h[2:1]=00b	ROS	No	0
		Offset 18h[2:1]=10b	RW	Yes	0
19:4	<i>Reserved</i>	Offset 18h[2:1]=00b	RsvdP	No	0_000h
	When <b>BAR2/3</b> are used as a 64-bit BAR, bits [31:0] (including bits [19:4]) are used as the upper 32 bits.	Offset 18h[2:1]=10b	RW	Yes	0_000h
31:20	<b>Base Address 3</b>	RW	Yes	000h	



**Register 15-9. 20h Base Address 4  
(NT Port Virtual Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Memory BAR – only value supported	RO	No	0
2:1	<b>Memory Map Type</b> 00b = Base Address register is 32 bits wide and can be mapped anywhere in the 32-bit Memory space 10b = Base Address register is 64 bits wide and can be mapped anywhere in the 64-bit Address space All other encodings are <i>Reserved</i> .	ROS	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	ROS	Yes	0
19:4	<i>Reserved</i>	RsvdP	No	0_000h
31:20	<b>Base Address 4</b>	RW	Yes	000h

**Register 15-10. 24h Base Address 5  
(NT Port Virtual Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
<i><b>Note:</b> This register has RW privilege if <b>BAR4/5</b> is configured as a 64-bit BAR (<b>Base Address 4</b> register <b>Memory Map Type</b> field (NT Port Virtual Interface, offset 20h[2:1]), is programmed to 10b).</i>					
0	<b>Memory Space Indicator</b> <b>BAR5</b> can be used as an independent 32-bit only BAR, or as the upper 32 bits of 64-bit <b>BAR4/5</b> . 0 = Memory BAR – only value supported	Offset 20h[2:1]=00b	RO	No	0
		Offset 20h[2:1]=10b	RW	Yes	0
2:1	<b>Memory Map Type</b> 00b = Base Address register is 32 bits wide and can be mapped anywhere in the 32-bit Memory space All other encodings are <i>Reserved</i> .	Offset 20h[2:1]=00b	RO	No	00b
		Offset 20h[2:1]=10b	RW	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	Offset 20h[2:1]=00b	ROS	Yes	0
		Offset 20h[2:1]=10b	RW	Yes	0
19:4	<i>Reserved</i>	Offset 20h[2:1]=00b	RsvdP	No	0_000h
	When <b>BAR4/5</b> are used as a 64-bit BAR, bits [31:0] (including bits [19:4]) are used as the upper 32 bits.	Offset 20h[2:1]=10b	RW	Yes	0_000h
31:20	<b>Base Address 5</b>	RW	Yes	000h	

**Register 15-11. 2Ch Subsystem ID and Subsystem Vendor ID**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Subsystem Vendor ID</b>				
15:0	<b>Subsystem Vendor ID</b> Identifies the device manufacturer. Defaults to the PCI-SIG-issued <b>Vendor ID</b> of PLX (10B5h), if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	10B5h
<b>Subsystem ID</b>				
31:16	<b>Subsystem ID</b> Identifies the particular device. Defaults to the PLX part number for the PEX 8748, if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	87B0h

**Register 15-12. 30h Expansion ROM Base Address**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
<i><b>Note:</b> Expansion ROM can be enabled in either the NT Port Virtual or Link Interface, but not both simultaneously. Expansion ROM is enabled, by default, in the NT Port Link Interface (<b>Ingress Chip Control</b> register <b>Expansion ROM Virtual Side</b> bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 764h[0]) is Cleared).</i> <i>Expansion ROM can be disabled, by Setting the <b>Ingress Control</b> register <b>Disable NT Port Expansion ROM BAR</b> bit (Base mode – All Ports; Virtual Switch mode – VS Upstream Port(s), offset F60h[15]).</i>					
0	<b>Expansion ROM Enable</b> 0 = NT Port Virtual Interface Expansion ROM is disabled	Upstream Port(s), offset F60h[15]=1	RsvdP	No	0
	1 = NT Port Virtual Interface Expansion ROM is enabled, and NT Port Link Interface Expansion ROM is disabled	Upstream Port(s), offset F60h[15]=0	RW	Yes	0
13:1	<b>Reserved</b>		RsvdP	No	0-0h
31:14	<b>Expansion ROM Base Address</b> If the <b>Serial EEPROM Clock Frequency</b> register <b>Expansion ROM Size</b> bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 268h[16]) value is 0, the NT Port Expansion ROM size is 16 KB (default value is FFFF_C001h). Bit 14 is RW. If the <b>Expansion ROM Size</b> bit value is 1, the NT Port Expansion ROM size is 32 KB (default value is FFFF_8001h). Bit 14 is RO.	Upstream Port(s), offset F60h[15]=1	RsvdP	No	0-0h
		Upstream Port(s), offset F60h[15]=0	RW	Yes	0-0h

**Register 15-13. 34h Capability Pointer**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Capability Pointer</b> Default 40h points to the <b>PCI Power Management Capability</b> structure.	RO	Yes	40h
31:8	<b>Reserved</b>	RsvdP	No	0000_00h

**Register 15-14. 3Ch PCI Interrupt**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>PCI Interrupt Line</b> The Interrupt Line Routing value communicates interrupt line routing information. Values in this register are programmed by system software and are system architecture-specific. The value is used by device drivers and operating systems.	RW	Yes	00h
15:8	<b>PCI Interrupt Pin</b> Identifies the Conventional PCI Interrupt Message(s) the device (or device function) uses. Only value 00h or 01h is allowed in the PEX 8748. 00h = Indicates that the device does not use Conventional PCI Interrupt Message(s) 01h, 02h, 03h, and 04h = Maps to Conventional PCI Interrupt Messages for INTA#, INTB#, INTC#, and INTD#, respectively	RO	Yes	01h
23:16	<b>Min_Gnt</b> <b>Reserved</b> Minimum Grant. Does not apply to PCI Express.	RsvdP	No	00h
31:24	<b>Max_Lat</b> <b>Reserved</b> Maximum Latency. Does not apply to PCI Express.	RsvdP	No	00h

## 15.6 NT Port Virtual Interface PCI Power Management Capability Registers (Offsets 40h – 44h)

This section details the NT Port Virtual Interface PCI Power Management Capability registers. [Table 15-3](#) defines the register map.

**Table 15-3. NT Port Virtual Interface PCI Power Management Capability Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																
PCI Power Management Capability																Next Capability Pointer (48h)								Capability ID (01h)								40h
PCI Power Management Data												PCI Power Management Control/Status Bridge Extensions ( <i>Reserved</i> )				PCI Power Management Status and Control																44h

### Register 15-15. 40h PCI Power Management Capability

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Capability ID</b> Default = 01h – only value allowed.	RO	Yes	01h
15:8	<b>Next Capability Pointer</b> Default 48h points to the <b>MSI Capability</b> structure.	RO	Yes	48h
18:16	<b>Version</b> Default = 011b – only value allowed.	RO	Yes	011b
19	<b>PME Clock</b> Power Management Event (PME) clock. Does not apply to PCI Express. Returns a value of 0.	RsvdP	No	0
20	<b>Reserved</b>	RsvdP	No	0
21	<b>Device-Specific Initialization</b> 0 = Device-Specific Initialization is <i>not</i> required	RO	Yes	0
24:22	<b>AUX Current</b> The PEX 8748 does <i>not support</i> PME generation from the D3cold Device Power Management (PM) state; therefore, the serial EEPROM value for this field should be 000b.	RO	Yes	000b
25	<b>D1 Support</b> <i>Not supported</i> 0 = PEX 8748 does <i>not support</i> the D1 Device PM state	RsvdP	No	0
26	<b>D2 Support</b> <i>Not supported</i> 0 = PEX 8748 does <i>not support</i> the D2 Device PM state	RsvdP	No	0
31:27	<b>PME Support</b> The default value is applied to bits [31, 30, and 27] only. PME Messages are disabled, by default.	RO	Yes	0000_0b

**Register 15-16. 44h PCI Power Management Status and Control**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Power Management Status and Control</b>				
1:0	<b>Power State</b> Used to determine the Port's current Device PM state, and to Set the Port into a new Device PM state.  00b = D0 01b = D1 – <i>Not supported</i> 10b = D2 – <i>Not supported</i> 11b = D3hot  If software attempts to write an unsupported state to this field, the Write operation completes normally; however, the data is discarded and no state change occurs.	RW	Yes	00b
2	<b>Reserved</b>	RsvdP	No	0
3	<b>No Soft Reset</b> 1 = Devices transitioning from the D3hot to D0 state, because of Power State commands, do not perform an internal reset	ROS	Yes	1
7:4	<b>Reserved</b>	RsvdP	No	0h
8	<b>PME Enable</b> Tied to 0, because the PEX 8748 does <i>not</i> generate PME in PCI Express mode.	RsvdP	No	0
12:9	<b>Data Select</b> Initially writable by serial EEPROM and/or I <sup>2</sup> C only <sup>a</sup> . This Configuration Space register (CSR) access privilege changes to RW after a Serial EEPROM and/or I <sup>2</sup> C Write occurs to this register. Selects the field [14:13] ( <i>Data Scale</i> ) and <b>PCI Power Management Data</b> register <i>Data</i> field (offset 44h[31:24]).  0h = D0 power consumed 3h = D3hot power consumed 4h = D0 power dissipated 7h = D3hot power dissipated  All other encodings are <b>Reserved</b> .	RO	Yes	0h
14:13	<b>Data Scale</b> Writable by serial EEPROM and/or I <sup>2</sup> C only <sup>a</sup> . Indicates the scaling factor to be used when interpreting the <b>PCI Power Management Data</b> register <i>Data</i> field (offset 44h[31:24]) value. The value and meaning of the <i>Data Scale</i> field varies, depending upon which data value is selected by field [12:9] ( <i>Data Select</i> ). There are four internal <i>Data Scale</i> fields (one each, per <i>Data Select</i> values 0h, 3h, 4h, and 7h). For other <i>Data Select</i> values, the <i>Data Scale</i> value returned is 0h.	RO	Yes	00b
15	<b>PME Status</b> 0 = PME is not being generated by the NT Port	RsvdP	No	0

**Register 15-16. 44h PCI Power Management Status and Control (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Power Management Control/Status Bridge Extensions</b>				
21:16	<i>Reserved</i>	RsvdP	No	0-0h
22	<b>B2/B3 Support</b> <i>Reserved</i> Cleared, as required by the <i>PCI Power Mgmt. r1.2</i> .	RsvdP	No	0
23	<b>Bus Power/Clock Control Enable</b> <i>Reserved</i> Cleared, as required by the <i>PCI Power Mgmt. r1.2</i> .	RsvdP	No	0
<b>PCI Power Management Data</b>				
31:24	<b>Data</b> Writable by serial EEPROM and/or I <sup>2</sup> C only <sup>a</sup> . There are four supported <i>Data Select</i> values (0h, 3h, 4h, and 7h). For other <i>Data Select</i> values, the <i>Data</i> value returned is 00h. Selected by the <b>PCI Power Management Status and Control</b> register <i>Data Select</i> field (offset 44h[12:9]).	RO	Yes	00h

- a. With no serial EEPROM nor previous I<sup>2</sup>C programming, Reads return a value of 00h for the **PCI Power Management Status and Control** register *Data Scale* and **Data** register *Data* fields (for all *Data Selects*).

## 15.7 NT Port Virtual Interface MSI Capability Registers (Offsets 48h – 64h)

The registers detailed in [Section 13.9, “Message Signaled Interrupt Capability Registers \(Offsets 48h – 64h\),”](#) are also applicable to the NT Port Virtual Interface, except as defined in [Table 15-4](#) (register map), and [Register 15-17](#) through [Register 15-19](#).

**Table 15-4. NT Port Virtual Interface MSI Capability Register Map<sup>a</sup>**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																				
MSI Control																Next Capability Pointer (68h)								Capability ID (05h)						48h						
MSI Address																																				4Ch
MSI Upper Address																																				50h
Reserved																MSI Data																				54h
MSI Mask																																				58h
MSI Status																																				5Ch
Reserved																																				60h – 64h

- a. Offsets 54h, 58h, and 5Ch change to 50h, 54h, and 58h, respectively, when the **MSI Control** register **MSI 64-Bit Address Capable** bit (offset 48h[23]) is Cleared.

**Register 15-17. 48h MSI Capability**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>MSI Capability Header</b>				
7:0	<b>Capability ID</b> Program to 05h, as required by the <i>PCI r3.0</i> .	RO	Yes	05h
15:8	<b>Next Capability Pointer</b> Program to 68h, to point to the <b>PCI Express Capability</b> structure.	RO	Yes	68h
<b>MSI Control</b>				
16	<b>MSI Enable</b> 0 = MSIs for the NT Port Virtual Interface are disabled 1 = MSIs for the NT Port Virtual Interface are enabled, and INT <sub>x</sub> Interrupt Messages and <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode) output assertion are disabled	RW	Yes	0
19:17	<b>Multiple Message Capable</b> 000b = NT Port Virtual Interface can request only one MSI Vector 001b = NT Port Virtual Interface can request two MSI Vectors 010b = NT Port Virtual Interface can request four MSI Vectors 011b = NT Port Virtual Interface can request eight MSI Vectors All other encodings are <b>Reserved</b> .	RO	Yes	011b
22:20	<b>Multiple Message Enable</b> 000b = NT Port Virtual Interface is allocated one MSI Vector, by default. 001b = NT Port Virtual Interface is allocated two MSI Vectors. 010b = NT Port Virtual Interface is allocated four MSI Vectors. 011b = NT Port Virtual Interface is allocated eight MSI Vectors. Up to six MSI Vectors are used; the upper two MSI Vectors (having the highest values of Message Data bits [2:0]) are <b>not</b> used. All other encodings are <b>Reserved</b> . <b>Note:</b> This field should not be programmed with a value larger than that of field [19:17] (Multiple Message Capable). If the value of this field is larger than that of field [19:17], the Multiple Message Capable value takes effect.	RW	Yes	000b
23	<b>MSI 64-Bit Address Capable</b> 0 = PEX 8748 is capable of generating MSI 32-bit addresses ( <b>MSI Address</b> register, offset 4Ch, is the Message address) 1 = PEX 8748 is capable of generating MSI 64-bit addresses ( <b>MSI Address</b> register, offset 4Ch, is the lower 32 bits of the Message address, and <b>MSI Upper Address</b> register, offset 50h, is the upper 32 bits of the Message address)	RO	Yes	1
24	<b>Per Vector Masking Capable</b> 0 = PEX 8748 does not have Per Vector Masking capability 1 = PEX 8748 has Per Vector Masking capability	RO	Yes	1
31:25	<b>Reserved</b>	RsvdP	No	0-0h



**Register 15-18. 58h MSI Mask**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
<p>NT Port Virtual Interface interrupt sources are grouped into four categories – Power Management/Link State events, Device-Specific NT-Link Port events, GPIO-generated interrupts, and NT-Virtual Doorbell-generated interrupts.</p> <p>The quantity of MSI Vectors that are generated is determined by the <b>MSI Control</b> register <i>Multiple Message Capable</i> and <i>Multiple Message Enable</i> fields (NT Port Virtual Interface, offset 48h[19:17 and 22:20], respectively):</p> <ul style="list-style-type: none"><li>• If <b>one</b> MSI Vector is enabled (default), all four interrupt events are combined into a single Vector</li><li>• If <b>two</b> MSI Vectors are allocated, the individual Vectors indicate:<ul style="list-style-type: none"><li>– Vector[0] Power Management/Link State events, GPIO-generated interrupts, and NT-Virtual Doorbell-generated interrupts</li><li>– Vector[1] Device-Specific NT-Link Port events</li></ul></li><li>• If <b>four</b> MSI Vectors are allocated, the individual Vectors indicate:<ul style="list-style-type: none"><li>– Vector[0] Power Management/Link State events, GPIO-generated interrupts, and NT-Virtual Doorbell-generated interrupts</li><li>– Vector[1] Device-Specific NT-Link Port events</li><li>– Vector[2] GPIO-generated interrupts</li><li>– Vector[3] NT-Virtual Doorbell-generated interrupts</li></ul></li></ul> <p><b>Notes:</b> The offset for this register changes from 58h, to 54h, when the <b>MSI Control</b> register <i>MSI 64-Bit Address Capable</i> bit (offset 48h[23]) is Cleared.</p> <p>The bits in this register can be used to mask their respective <b>MSI Status</b> register bits (offset 5Ch).</p>					
0	<b>MSI Mask for Link State Events</b> MSI mask for Power Management event- or Link State event-generated interrupts.	Offset 48h[22:20]≥010b	RW	Yes	0
	<b>MSI Mask for Shared Interrupt Sources</b> MSI mask for all interrupt sources when the <b>MSI Control</b> register <i>Multiple Message Enable</i> field indicates that the Host has allocated one or two Vectors.	Offset 48h[22:20]≤001b	RW	Yes	0
1	<b>MSI Mask for Device-Specific NT-Link Port Events</b> MSI mask for Device-Specific NT-Link Port event-generated interrupts. Enables MSIs for the following NT-Link Port events, defined in the <b>Link Error Status Virtual</b> and <b>Link Error Mask Virtual</b> registers (NT Port Virtual Interface, offsets FE0h and FE4h, respectively): <ul style="list-style-type: none"><li>• NT-Link Port Correctable error (NT Port Link Interface, offset FC4h)</li><li>• NT-Link Port Uncorrectable error (NT Port Link Interface, offset FB8h)</li><li>• NT-Link Port Data Link Layer State change</li><li>• NT-Link Port received (and consequently dropped) a Fatal or Non-Fatal Error Message</li></ul>	Offset 48h[22:20]≥001b	RW	Yes	0
	<b>Reserved</b>	Offset 48h[22:20]=000b	RsvdP	No	0

**Register 15-18. 58h MSI Mask (Cont.)**

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default
2	<b>MSI Mask for GPIO-Generated Interrupts</b>	Offset 48h[22:20]≥010b	RW	Yes	0
	<b>Reserved</b>	Offset 48h[22:20]≤001b	RsvdP	No	0
3	<b>MSI Mask for NT-Virtual Doorbell-Generated Interrupts</b> Refer to NT Port registers located at offsets C4Ch through C58h.	Offset 48h[22:20]≥010b	RW	Yes	0
	<b>Reserved</b>	Offset 48h[22:20]≤001b	RsvdP	No	0
31:4	<b>Reserved</b>		RsvdP	No	0000_000h

**Register 15-19. 5Ch MSI Status**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
NT Port Virtual Interface interrupt sources are grouped into four categories – Power Management/Link State events, Device-Specific NT-Link Port events, GPIO-generated interrupts, and NT-Virtual Doorbell-generated interrupts. The quantity of MSI Vectors that are generated is determined by the <b>MSI Control</b> register <i>Multiple Message Capable</i> and <i>Multiple Message Enable</i> fields (NT Port Virtual Interface, offset 48h[19:17 and 22:20], respectively):					
<ul style="list-style-type: none"><li>• If <b>one</b> MSI Vector is enabled (default), all four interrupt events are combined into a single Vector</li><li>• If <b>two</b> MSI Vectors are allocated, the individual Vectors indicate:<ul style="list-style-type: none"><li>– Vector[0] Power Management/Link State events, GPIO-generated interrupts, and NT-Virtual Doorbell-generated interrupts</li><li>– Vector[1] Device-Specific NT-Link Port events</li></ul></li><li>• If <b>four</b> MSI Vectors are allocated, the individual Vectors indicate:<ul style="list-style-type: none"><li>– Vector[0] Power Management/Link State events, GPIO-generated interrupts, and NT-Virtual Doorbell-generated interrupts</li><li>– Vector[1] Device-Specific NT-Link Port events</li><li>– Vector[2] GPIO-generated interrupts</li><li>– Vector[3] NT-Virtual Doorbell-generated interrupts</li></ul></li></ul>					
<i>Notes: The offset for this register changes from 5Ch, to 58h, when the <b>MSI Control</b> register <i>MSI 64-Bit Address Capable</i> bit (offset 48h[23]) is Cleared.</i>					
<i>The bits in this register can be masked by their respective <b>MSI Mask</b> register bits (offset 58h).</i>					
0	<b>MSI Pending Status for Link State Events</b> MSI pending status for Power Management event- or Link State event-generated interrupts.	Offset 48h[22:20]≥010b	RO	No	0
	<b>MSI Pending Status for Shared Interrupt Sources</b> MSI pending status for all interrupt sources when the <b>MSI Control</b> register <i>Multiple Message Enable</i> field indicates that the Host has allocated one or two Vectors.	Offset 48h[22:20]≤001b	RO	No	0
1	<b>MSI Pending Status for Device-Specific Error Triggered Event</b> Indicates the MSI pending status for the Device-Specific NT-Link Port event-generated interrupts defined in the <b>Link Error Status Virtual</b> and <b>Link Error Mask Virtual</b> registers (NT Port Virtual Interface, offsets FE0h and FE4h, respectively): <ul style="list-style-type: none"><li>• NT-Link Port Correctable error (NT Port Link Interface, offset FC4h)</li><li>• NT-Link Port Uncorrectable error (NT Port Link Interface, offset FB8h)</li><li>• NT-Link Port Data Link Layer State change</li><li>• NT-Link Port received (and consequently dropped) a Fatal or Non-Fatal Error Message</li></ul>	Offset 48h[22:20]≥001b	RO	No	0
	<b>Reserved</b>	Offset 48h[22:20]=000b	RsvdP	No	0

**Register 15-19. 5Ch MSI Status (Cont.)**

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default
2	<b>MSI Pending Status for GPIO-Generated Interrupts</b>	Offset 48h[22:20]≥010b	RO	No	0
	<b>Reserved</b>	Offset 48h[22:20]≤001b	RsvdP	No	0
3	<b>MSI Pending Status for NT-Virtual Doorbell-Generated Interrupts</b> Refer to NT Port registers located at offsets C4Ch through C58h.	Offset 48h[22:20]≥010b	RO	No	0
	<b>Reserved</b>	Offset 48h[22:20]≤001b	RsvdP	No	0
31:4	<b>Reserved</b>		RsvdP	No	0000_000h

## 15.8 NT Port Virtual Interface PCI Express Capability Registers (Offsets 68h – A0h)

The registers detailed in [Section 13.10, “PCI Express Capability Registers \(Offsets 68h – A0h\),”](#) are also applicable to the NT Port Virtual Interface, except as defined in [Table 15-5](#) (register map; offsets 7Ch and 80h are *Reserved*), and [Register 15-20](#) through [Register 15-28](#).

### Table 15-5. NT Port Virtual Interface PCI Express Capability Register Map

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0	
PCI Express Capability		Next Capability Pointer (C8h)		Capability ID (10h) 68h
Device Capability				6Ch
Device Status		Not Supported/Reserved		Device Control 70h
Link Capability				74h
Link Status		Reserved		Link Control 78h
Reserved				7Ch – 88h
Device Capability 2				8Ch
Device Status 2 (Reserved)		Device Control 2		90h
Reserved			Link Capability 2	94h
Link Status 2		Link Control 2		98h
Reserved				9Ch – A0h

**Register 15-20. 68h PCI Express Capability List and Capability**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Express Capability List</b>				
7:0	<b>Capability ID</b> Program to 10h, by default, as required by the <i>PCI Express Base r3.0</i> .	RO	Yes	10h
15:8	<b>Next Capability Pointer</b> Program to C8h, to point to the <b>Vendor-Specific Capability 3</b> structure.	RO	Yes	C8h
<b>PCI Express Capability</b>				
19:16	<b>Capability Version</b> The PEX 8748 NT Port Virtual Interface programs this field to 2h, as required by the <i>PCI Express Base r3.0</i> .	RO	Yes	2h
23:20	<b>Device/Port Type</b> Default = PCI Express endpoint device.	RO	Yes	0h
24	<b>Slot Implemented</b> <i>Not valid for PCI Express endpoint devices</i>	RsvdP	No	0
29:25	<b>Interrupt Message Number</b> The serial EEPROM writes 00_000b, because the Base Message and MSI Messages are the same.	RO	Yes	00_000b
31:30	<b>Reserved</b>	RsvdP	No	00b

**Register 15-21. 6Ch Device Capability**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
2:0	<b>Maximum Payload Size Supported</b>  <i><b>Note:</b> The default Maximum Payload Size supported is based upon the STRAP_STNx_PORTCFGx 3-state inputs. If the serial EEPROM or I<sup>2</sup>C/SMBus changes the Port configuration (from the strapped value), it must also program the value of this field, accordingly (with the value dependent upon the Station having the most enabled Ports).</i>  The value for all Ports is determined by the Station having the most enabled Ports: <ul style="list-style-type: none"> <li>One or two enabled Ports = 100b</li> <li>Three enabled Ports = 011b</li> <li>Four enabled Ports = 010b</li> </ul> 000b = NT Port Virtual Interface supports a 128-byte maximum payload 001b = NT Port Virtual Interface supports a 256-byte maximum payload 010b = NT Port Virtual Interface supports a 512-byte maximum payload 011b = NT Port Virtual Interface supports a 1,024-byte maximum payload 100b = NT Port Virtual Interface supports a 2,048-byte maximum payload  No other encodings are supported.	HwInit	Yes	Defined by STRAP_STNx_PORTCFGx input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station x</i> field(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6, 5:3, and/or 2:0])
4:3	<b>Phantom Functions Supported</b> <i>Not supported</i>	RO	Yes	00b
5	<b>Extended Tag Field Supported</b> 0 = Maximum <i>Tag</i> field is 5 bits 1 = Maximum <i>Tag</i> field is 8 bits	RO	Yes	0
8:6	<b>Endpoint L0s Acceptable Latency</b> 111b = No Limit	RO	Yes	111b
11:9	<b>Endpoint L1 Acceptable Latency</b> 111b = No Limit	RO	Yes	111b
14:12	<i>Reserved</i>	RsvdP	No	000b
15	<b>Role-Based Error Reporting</b>	RO	Yes	1

**Register 15-21. 6Ch Device Capability (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
17:16	<b>Reserved</b>	RsvdP	No	00b
25:18	<b>Captured Slot Power Limit Value</b> For the NT Port Virtual Interface, the upper limit on power supplied by the slot is determined by multiplying the value in this field by the value in field [27:26] ( <i>Captured Slot Power Limit Scale</i> ).	RO	Yes	00h
27:26	<b>Captured Slot Power Limit Scale</b> For the NT Port Virtual Interface, the upper limit on power supplied by the slot is determined by multiplying the value in this field by the value in field [25:18] ( <i>Captured Slot Power Limit Value</i> ). 00b = 1.0 01b = 0.1 10b = 0.01 11b = 0.001	RO	Yes	00b
31:28	<b>Reserved</b>	RsvdP	No	0h



**Register 15-22. 70h Device Status and Control**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Control</b>				
0	<b>Correctable Error Reporting Enable</b> 0 = Disables 1 = Enables the NT Port Virtual Interface to report Correctable errors to the Virtual side Host	RW	Yes	0
1	<b>Non-Fatal Error Reporting Enable</b> 0 = Disables 1 = Enables the NT Port Virtual Interface to report Non-Fatal errors to the Virtual side Host	RW	Yes	0
2	<b>Fatal Error Reporting Enable</b> 0 = Disables 1 = Enables the NT Port Virtual Interface to report Fatal errors to the Virtual side Host	RW	Yes	0
3	<b>Unsupported Request Reporting Enable</b> 0 = Disables 1 = Enables the NT Port Virtual Interface to report UR errors to the Virtual side Host	RW	Yes	0
4	<b>Enable Relaxed Ordering</b> <i>Not supported</i>	RW	Yes	1
7:5	<b>Maximum Payload Size</b> The NT Port Virtual Interface power-on/reset value is 000b, to support a Maximum Payload Size of 128 bytes. Software can change this field to configure the NT Port Virtual Interface to support other Payload sizes; however, software cannot change this field to a value larger than that indicated by the <b>Device Capability</b> register <i>Maximum Payload Size Supported</i> field (offset 6Ch[2:0]), for the NT Port Virtual and Link Interfaces. (Requester and Completer domains must possess the same Maximum Payload Size.)  000b = NT Port Virtual Interface supports a 128-byte maximum payload 001b = NT Port Virtual Interface supports a 256-byte maximum payload 010b = NT Port Virtual Interface supports a 512-byte maximum payload 011b = NT Port Virtual Interface supports a 1,024-byte maximum payload 100b = NT Port Virtual Interface supports a 2,048-byte maximum payload  No other encodings are supported. <b>Note:</b> Software must halt all transactions through the NT Port before changing this field.	RW	Yes	000b

**Register 15-22. 70h Device Status and Control (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Extended Tag Field Enable</b> <i>Not supported</i>	RsvdP	No	0
9	<b>Phantom Functions Enable</b> <i>Not supported</i>	RsvdP	No	0
10	<b>AUX Power PM Enable</b> <i>Not supported</i>	RsvdP	No	0
11	<b>Enable No Snoop</b> <i>Not supported</i>	RW	Yes	1
14:12	<b>Maximum Read Request Size</b> <i>Not supported</i>	RsvdP	No	000b
15	<b>Reserved</b>	RsvdP	No	0
<b>Device Status</b>				
16	<b>Correctable Error Detected</b> 0 = NT Port Virtual Interface did not detect a Correctable error 1 = NT Port Virtual Interface detected a Correctable error, regardless of the bit 0 ( <i>Correctable Error Reporting Enable</i> ) state	RW1C	Yes	0
17	<b>Non-Fatal Error Detected</b> 0 = NT Port Virtual Interface did not detect a Non-Fatal error 1 = NT Port Virtual Interface detected a Non-Fatal error, regardless of the bit 1 ( <i>Non-Fatal Error Reporting Enable</i> ) state	RW1C	Yes	0
18	<b>Fatal Error Detected</b> 0 = NT Port Virtual Interface did not detect a Fatal error 1 = NT Port Virtual Interface detected a Fatal error, regardless of the bit 2 ( <i>Fatal Error Reporting Enable</i> ) state	RW1C	Yes	0
19	<b>Unsupported Request Detected</b> 0 = NT Port Virtual Interface did not detect a UR 1 = NT Port Virtual Interface detected a UR, regardless of the bit 3 ( <i>Unsupported Request Reporting Enable</i> ) state	RW1C	Yes	0
20	<b>AUX Power Detected</b> <i>Not supported</i>	RsvdP	No	0
21	<b>Transactions Pending</b> <i>Not supported</i> Because the PEX 8748 NT Port is a bridging device, it does not track Completion for the corresponding Non-Posted transactions. Therefore, the NT Port Virtual Interface does not implement Transactions Pending.	RsvdP	No	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h

**Register 15-23. 74h Link Capability**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Ports and Lanes (Link widths).				
3:0	<b>Maximum Link Speed</b> Indicates the NT Port Virtual Interface's maximum Link speed. The encoding is the binary value of the bit location in the <b>Link Capability 2</b> register <i>Supported Link Speeds Vector</i> field (NT Port Virtual Interface, offset 94h[7:1]) that corresponds to the maximum Link speed. <i>For example</i> , the default value 3h indicates that the maximum Link speed is that which corresponds to bit 3 in the <i>Supported Link Speeds Vector</i> field, which is 8.0 GT/s. The default value for all Ports can be globally changed to 2h (Gen 2 maximum speed) or 1h (Gen 1 maximum speed), according to the STRAP_GEN1_GEN2 input state (Z = 2h (Gen 2), or 0 = 1h (Gen 1)).	RO	Yes	3h
9:4	<b>Maximum Link Width</b> The PEX 8748 maximum Link width is x16 = 01_0000b. Actual maximum Link width is defined by the STRAP_STNx_PORTCFGx 3-state inputs. 00_0000b = <i>Reserved</i> 00_0001b = x1 00_0010b = x2 00_0100b = x4 00_1000b = x8 01_0000b = x16 All other encodings are <i>not supported</i> .	Refer to Default value	No	Defined by STRAP_STNx_PORTCFGx input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> -related register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offsets 300h through 308h)
11:10	<b>Active State Power Management (ASPM) Support</b> Active State Link PM support. Indicates the level of ASPM supported by the Port. 01b = L0s Link PM state entry is supported 11b = L0s and L1 Link PM states are supported All other encodings are <i>Reserved</i> .	RO	Yes	01b

**Register 15-23. 74h Link Capability (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
14:12	<b>L0s Exit Latency</b> Indicates the L0s Link PM state exit latency for the Port's PCI Express Link. The value reported indicates the length of time this Port requires to complete the L0s to L0 Link PM state transition. If L0s is not supported, the value of this field is undefined; however, for the recommended value, refer to the Implementation Note in the <i>PCI Express Base r3.0</i> , Section 5.4.1.1, "Potential Issues with Legacy Software When L0s is Not Supported." 000b = Less than 64 ns 001b = 64 ns to less than 128 ns 010b = 128 ns to less than 256 ns 011b = 256 ns to less than 512 ns 100b = 512 ns to less than 1 $\mu$ s 101b = 1 $\mu$ s to less than 2 $\mu$ s 110b = 2 to 4 $\mu$ s (default) 111b = More than 4 $\mu$ s	RO	No	110b
17:15	<b>L1 Exit Latency</b> Indicates the L1 Link PM state exit latency for the Port's PCI Express Link. The value reported indicates the length of time this Port requires to complete the ASPM L1 to L0 Link PM state transition. If L0 is not supported, the value of this field is undefined. 000b = Less than 1 $\mu$ s 001b = 1 $\mu$ s to less than 2 $\mu$ s (5.0 GT/s) 010b = 2 $\mu$ s to less than 4 $\mu$ s (default, 2.5 GT/s) 011b = 4 $\mu$ s to less than 8 $\mu$ s 100b = 8 $\mu$ s to less than 16 $\mu$ s 101b = 16 $\mu$ s to less than 32 $\mu$ s 110b = 32 to 64 $\mu$ s 111b = More than 64 $\mu$ s <b>Note:</b> <i>The NT Port Virtual Interface never enters the L1 Link PM state, because there is no physical Link attached to it.</i>	RO	Yes	010b
18	<b>Clock Power Management</b>	RO	Yes	0
21:19	<b>Reserved</b>	RsvdP	No	000b
22	<b>ASPM Optionality Compliance</b>	HwInit	Yes	1
23	<b>Reserved</b>	RsvdP	No	0

**Register 15-23. 74h Link Capability (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default																																							
31:24	<b>NT Port Number</b> The NT Port Number value is determined by the STRAP_NT_UPSTRM_PORTSEL[2:0] input states, combined with the <b>VS0 Upstream</b> register <i>NT Port</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 360h[12:8]) value.	Refer to Default value	No	Defined by STRAP_NT_UPSTRM_PORTSEL[2:0] input states																																							
	<table><tr><th>STRAP_NT_UPSTRM_PORTSEL[2:0] Input States</th><th>Offset 360h[12:8] Value</th><th>Port Number</th></tr><tr><td>000</td><td>0_0000b</td><td>0</td></tr><tr><td>00Z</td><td>0_0001b</td><td>1</td></tr><tr><td>001</td><td>0_0010b</td><td>2</td></tr><tr><td>0Z0</td><td>0_0011b</td><td>3</td></tr><tr><td>011</td><td>0_1000b</td><td>8</td></tr><tr><td>Z00</td><td>0_1001b</td><td>9</td></tr><tr><td>Z0Z</td><td>0_1010b</td><td>10</td></tr><tr><td>Z01</td><td>0_1011b</td><td>11</td></tr><tr><td>Z1Z</td><td>1_0000b</td><td>16</td></tr><tr><td>Z11</td><td>1_0001b</td><td>17</td></tr><tr><td>100</td><td>1_0010b</td><td>18</td></tr><tr><td>10Z</td><td>1_0011b</td><td>19</td></tr></table>				STRAP_NT_UPSTRM_PORTSEL[2:0] Input States	Offset 360h[12:8] Value	Port Number	000	0_0000b	0	00Z	0_0001b	1	001	0_0010b	2	0Z0	0_0011b	3	011	0_1000b	8	Z00	0_1001b	9	Z0Z	0_1010b	10	Z01	0_1011b	11	Z1Z	1_0000b	16	Z11	1_0001b	17	100	1_0010b	18	10Z	1_0011b	19
	STRAP_NT_UPSTRM_PORTSEL[2:0] Input States				Offset 360h[12:8] Value	Port Number																																					
	000				0_0000b	0																																					
	00Z				0_0001b	1																																					
	001				0_0010b	2																																					
	0Z0				0_0011b	3																																					
	011				0_1000b	8																																					
	Z00				0_1001b	9																																					
	Z0Z				0_1010b	10																																					
	Z01				0_1011b	11																																					
	Z1Z				1_0000b	16																																					
	Z11				1_0001b	17																																					
	100				1_0010b	18																																					
	10Z				1_0011b	19																																					
	All other encodings are <i>Reserved</i> .																																										

**Register 15-24. 78h Link Status and Control**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Link Control</b>				
1:0	<b>Active State Power Management (ASPM) Control</b> The NT Port Virtual Interface ignores this register value, because no external Port connection exists.	RW	Yes	00b
2	<b>Reserved</b>	RsvdP	No	0
3	<b>Read Request Return Parameter Control</b> Read Request Return Parameter “R” control. Read Completion Boundary (RCB).	RO	Yes	0
4	<b>Link Disable</b> <b>Reserved</b> for the NT Port Virtual Interface.	RsvdP	No	0
5	<b>Retrain Link</b> <b>Reserved</b> for the NT Port Virtual Interface.	RsvdP	No	0
6	<b>Common Clock Configuration</b> The NT Port Virtual Interface ignores this register value, because no external Port connection exists.	RW	Yes	0
7	<b>Extended Sync</b> The NT Port Virtual Interface ignores this register value, because no external Port connection exists.	RW	Yes	0
15:8	<b>Reserved</b>	RsvdP	No	00h

**Register 15-24. 78h Link Status and Control (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Link Status</b>				
19:16	<b>Current Link Speed</b> Indicates the negotiated Link speed of the Port's PCI Express Link. The encoding is the binary value of the bit location in the <b>Link Capability 2</b> register <i>Supported Link Speeds Vector</i> field (NT Port Virtual Interface, offset 94h[7:1]) that corresponds to the current Link speed. <i>For example</i> , a value of 1h indicates that the current Link speed is that which corresponds to bit 1 in the <i>Supported Link Speeds Vector</i> field, which is 2.5 GT/s. If the Link is not up, the value of this field is undefined.	RO	No	1h
25:20	<b>Negotiated Link Width</b> Reports the Link status of the NT Port Link Interface. Dependent upon the configuration of the physical Ports. Link width is determined by the negotiated value with the attached Lane/Port. If the Link is not up, the value of this field is undefined. 00_0000b = Link is Down (default) 00_0001b = x1 00_0010b = x2 00_0100b = x4 00_1000b = x8 01_0000b = x16 All other encodings are <i>not supported</i> .	RO	No	00_0000b
26	<b>Reserved</b>	RsvdP	No	0
27	<b>Link Training</b> <i>Reserved</i> for the NT Port Virtual Interface. Always read as 0.	RsvdP	No	0
28	<b>Slot Clock Configuration</b> Because there is no external connection to the NT Port Virtual Interface, this bit is always Cleared, which indicates that the PEX 8748 uses an independent clock.	HwInit	Yes	0
31:29	<b>Reserved</b>	RsvdP	No	000b

**Register 15-25. 8Ch Device Capability 2**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
6:0	<b>Reserved</b>	RsvdP	No	0-0h
7	<b>AtomicOp 32-Bit Completer Supported</b> 1 = Supports Atomic 32-bit Completer	ROS	Yes	1
8	<b>AtomicOp 64-Bit Completer Supported</b> 1 = Supports Atomic 64-bit Completer	ROS	Yes	1
9	<b>AtomicOp 128-Bit Completer Supported</b> 1 = Supports Atomic 128-bit Completer	ROS	Yes	1
31:10	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 15-26. 90h Device Status and Control 2**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Control 2</b>				
5:0	<i>Reserved</i>	RsvdP	No	0-0h
6	<b>AtomicOp Requester Enable</b>	RW	Yes	0
15:7	<i>Reserved</i>	RsvdP	No	0-0h
<b>Device Status 2</b>				
31:16	<i>Reserved</i>	RsvdP	No	0000h

**Register 15-27. 94h Link Capability 2**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<i>Reserved</i>	RsvdP	No	0
7:1	<b>Supported Link Speeds Vector</b> Supports 2.5, 5.0, and 8.0 GT/s Link speeds. Indicates the supported Link speed(s) of the associated Port. For each bit, a value of 1 indicates that the corresponding Link speed is supported; otherwise, the Link speed is <i>not supported</i> .		RO	07h
	<b>Bit(s)</b>	<b>Description/Function</b>		
	1	2.5 GT/s		
	2	5.0 GT/s		
	3	8.0 GT/s		
	7:4	<i>Reserved</i>		
31:8	<i>Reserved</i>	RsvdP	No	0000_00h



**Register 15-28. 98h Link Status and Control 2**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Link Control 2</b>				
Bits in this register have no effect and are provided for compliance testing only.				
3:0	<b>Target Link Speed</b>	RWS	Yes	3h
4	<b>Enter Compliance</b>	RWS	Yes	0
5	<b>Hardware Autonomous Speed Disable</b> <i>Reserved</i>	RsvdP	No	0
6	<b>Selectable De-Emphasis</b> <i>Reserved</i>	RsvdP	Yes	0
9:7	<b>Transmit Margin</b>	RWS	Yes	000b
10	<b>Enter Modified Compliance</b>	RWS	Yes	0
11	<b>Compliance SOS</b>	RWS	Yes	0
15:12	<b>Compliance Preset/De-Emphasis</b>	RWS	Yes	0h
<b>Link Status 2</b>				
16	<b>Current De-Emphasis Level</b>	RO	Yes	0
31:17	<i>Reserved</i>	RsvdP	No	0-0h

## 15.9 NT Port Virtual Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh)

This section details the NT Port Virtual Interface Vendor-Specific Capability 3 registers, which include the **Memory BARx Setup** registers and **Configuration Address** and **Data Window** registers. Table 15-6 defines the register map used by the NT Port Virtual Interface.

The Cursor Mechanism registers at offsets F8h and FCh provide a means for accessing PCI Express Extended Configuration Space registers (offsets 100h through FFFh) within the NT Port Virtual and Link Interfaces, when only standard PCI Configuration transactions (that do not support the *Extended Register Number* field within the Completion Request Header) are available. The Cursor Mechanism can generally access only those registers that are defined by the *PCI Express Base r3.0*, and not the Device-Specific registers. However, if Port 0 is the NT Port, the Cursor Mechanism in the NT Port Virtual Interface registers can also access the Device-Specific registers.

**Table 15-6. NT Port Virtual Interface Vendor-Specific Capability 3 Register Map**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
<i>Reserved</i>																Next Capability Pointer (00h)								Capability ID 3 (09h)								C8h
<i>Reserved</i>																																CCh
NT Port Virtual Interface BAR0/1 Setup																																D0h
NT Port Virtual Interface Memory BAR2 Setup																																D4h
NT Port Virtual Interface Memory BAR2/3 Setup																																D8h
NT Port Virtual Interface Memory BAR4 Setup																																DCh
NT Port Virtual Interface Memory BAR4/5 Setup																																E0h
<i>Reserved</i>																														E4h –		F4h
Configuration Address Window																<i>Reserved</i>																F8h
Configuration Data Window																																FCh

**Register 15-29. C8h Vendor-Specific Capability 3**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Capability ID 3</b>	ROS	Yes	09h
15:8	<b>Next Capability Pointer</b> 00h = This capability is the last capability in the Linked List	ROS	Yes	00h
31:16	<i>Reserved</i>	RsvdP	No	0000h

**Register 15-30. D0h NT Port Virtual Interface BAR0/1 Setup**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>BAR0/1 Enable</b> 00b = Disables Virtual Interface <b>BAR0</b> and <b>BAR1</b> 01b = <i>Reserved</i> 10b = Enables Virtual Interface <b>BAR0</b> and disables <b>BAR1</b> ( <b>BAR0</b> is a 32-bit BAR) 11b = Enables Virtual Interface <b>BAR0</b> and <b>BAR1</b> ( <b>BAR0/1</b> is a 64-bit BAR)	RWS	Yes	10b
2	<b>BAR0 Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RWS	Yes	0
31:3	<i>Reserved</i>	RsvdP	No	0-0h

**Register 15-31. D4h NT Port Virtual Interface Memory BAR2 Setup**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
0	<b>Type Selector</b>	RsvdP	No	0	
2:1	<b>BAR2 Type</b> 00b = <b>BAR2</b> is implemented as a 32-bit Memory BAR 10b = <b>BAR2/3</b> is implemented as a 64-bit Memory BAR  No other encodings are allowed.	RWS	Yes	00b	
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RWS	Yes	0	
19:4	<i>Reserved</i>	RsvdP	No	0_000h	
30:20	<b>BAR2 Size</b> Specifies the Address Range size requested by <b>BAR2</b> .  0 = Corresponding <b>BAR2</b> bits are RO bits that always return a value of 0, and Writes are ignored 1 = Corresponding <b>BAR2</b> bits are RWS bits <i><b>Note:</b> If BAR[30:n] is the Base field (BAR size is 2<sup>n</sup>), BAR[30:n] should have all ones (1).</i>	RWS	Yes	0-0h	
31	<b>BAR2 Enable</b> 0 = <b>BAR2</b> is disabled, all <b>BAR2</b> bits read 0 1 = <b>BAR2</b> is enabled	Field [2:1] (BAR2 Type) = 00b	RWS	Yes	0
	<b>BAR2 Size</b> Included with field [30:20] when this BAR is used as a 64-bit BAR.	Field [2:1] (BAR2 Type) = 10b	RWS	Yes	0

**Register 15-32. D8h NT Port Virtual Interface Memory BAR2/3 Setup**

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Type Selector</b>		RsvdP	No	0
2:1	<b>BAR3 Type</b> 00b = Selects 32-bit Memory BAR ( <b>BAR3</b> ) No other encodings are allowed.		ROS	No	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable		RWS	Yes	0
19:4	<b>Reserved</b>	Offset D4h[2:1] ( <i>BAR2 Type</i> ) = 00b	RsvdP	No	0_000h
	When <b>BAR2/3</b> are used as a 64-bit BAR, bits [31:0] (including bits [19:4]) are used as the upper 32 bits.	Offset D4h[2:1] ( <i>BAR2 Type</i> ) = 10b	RWS	Yes	0_000h
30:20	<b>BAR3 Size</b> Specifies the Address Range size requested by <b>BAR3</b> . 0 = Corresponding <b>BAR3</b> bits are RO bits that always return a value of 0, and Writes are ignored 1 = Corresponding <b>BAR3</b> bits are RWS bits <i>Note: If BAR[30:n] is the Base field (BAR size is 2<sup>n</sup>), BAR[30:n] should have all ones (1).</i>		RWS	Yes	0-0h
31	<b>BAR3 Enable</b> <b>32-Bit BAR</b> 0 = <b>BAR3</b> is disabled 1 = <b>BAR3</b> is enabled as a 32-bit BAR	Offset D4h[2:1] ( <i>BAR2 Type</i> ) = 00b	RWS	Yes	0
	<b>64-Bit BAR</b> 0 = <b>BAR2/3</b> is disabled, all <b>BAR2/3</b> bits read 0 1 = <b>BAR2/3</b> is enabled as a 64-bit BAR	Offset D4h[2:1] ( <i>BAR2 Type</i> ) = 10b	RWS	Yes	0

**Register 15-33. DCh NT Port Virtual Interface Memory BAR4 Setup**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
0	<b>Type Selector</b>	RsvdP	No	0	
2:1	<b>BAR4 Type</b> 00b = <b>BAR4</b> is implemented as a 32-bit Memory BAR ( <b>BAR4</b> ) 10b = <b>BAR4/5</b> is implemented as a 64-bit Memory BAR ( <b>BAR4/5</b> ) No other encodings are allowed.	RWS	Yes	00b	
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RWS	Yes	0	
19:4	<i><b>Reserved</b></i>	RsvdP	No	0_000h	
30:20	<b>BAR4 Size</b> Specifies the Address Range size requested by <b>BAR4</b> . 0 = Corresponding <b>BAR4</b> bits are RO bits that always return a value of 0, and Writes are ignored 1 = Corresponding <b>BAR4</b> bits are RWS bits <i><b>Note:</b> If BAR[30:n] is the Base field (BAR size is 2<sup>n</sup>), BAR[30:n] should have all ones (1).</i>	RWS	Yes	0-0h	
31	<b>BAR4 Enable</b> 0 = <b>BAR4</b> is disabled, all <b>BAR4</b> bits read 0 1 = <b>BAR4</b> is enabled	Field [2:1] (BAR4 Type) = 00b	RWS	Yes	0
	<b>BAR4 Size</b> Included with field [30:20] when this BAR is used as a 64-bit BAR.	Field [2:1] (BAR4 Type) = 10b	RWS	Yes	0

**Register 15-34. E0h NT Port Virtual Interface Memory BAR4/5 Setup**

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Type Selector</b>		RsvdP	No	0
2:1	<b>BAR5 Type</b> 00b = Selects 32-bit Memory BAR ( <b>BAR5</b> ) No other encodings are allowed.		ROS	No	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable		RWS	Yes	0
19:4	<b>Reserved</b>	Offset DCh[2:1] ( <i>BAR4 Type</i> ) = 00b	RsvdP	No	0_000h
	When <b>BAR4/5</b> are used as a 64-bit BAR, bits [31:0] (including bits [19:4]) are used as the upper 32 bits.	Offset DCh[2:1] ( <i>BAR4 Type</i> ) = 10b	RWS	Yes	0_000h
30:20	<b>BAR5 Size</b> Specifies the Address Range size requested by <b>BAR5</b> . 0 = Corresponding <b>BAR5</b> bits are RO bits that always return a value of 0, and Writes are ignored 1 = Corresponding <b>BAR5</b> bits are RWS bits <i>Note: If BAR[30:n] is the Base field (BAR size is 2<sup>n</sup>), BAR[30:n] should have all ones (1).</i>		RWS	Yes	0-0h
31	<b>BAR5 Enable</b> <b>32-Bit BAR</b> 0 = <b>BAR5</b> is disabled 1 = <b>BAR5</b> is enabled as a 32-bit BAR	Offset DCh[2:1] ( <i>BAR4 Type</i> ) = 00b	RWS	Yes	0
	<b>64-Bit BAR</b> 0 = <b>BAR4/5</b> is disabled, all <b>BAR4/5</b> bits read 0 1 = <b>BAR4/5</b> is enabled as a 64-bit BAR	Offset DCh[2:1] ( <i>BAR4 Type</i> ) = 10b	RWS	Yes	0

**Register 15-35. F8h Configuration Address Window  
(Device-Specific Cursor Mechanism)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<i>Reserved</i>	RsvdP	No	0000h
25:16	<b>Register Offset</b>	RW	Yes	0-0h
30:26	<i>Reserved</i>	RsvdP	No	0-0h
31	<b>Interface Select</b> 0 = Access is to the NT Port Link Interface Type 0 Configuration Space register 1 = Access is to the NT Port Virtual Interface Type 0 Configuration Space register	RW	Yes	0

**Register 15-36. FCh Configuration Data Window  
(Device-Specific Cursor Mechanism)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Register Data</b> Software selects a register by writing into the NT Port Virtual Interface Configuration Address window, then reads from or writes to that register using this register.	RW	Yes	0000_0000h

## 15.10 NT Port Virtual Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)

The registers detailed in [Section 13.15, “Virtual Channel Extended Capability Registers \(Offsets 148h – 1BCh\),”](#) are also applicable to the NT Port Virtual Interface, except as defined in [Table 15-7](#) (register map; offsets 178h through 1B4h are *Reserved*), and [Register 15-37](#) through [Register 15-40](#).

**Table 15-7. NT Port Virtual Interface Virtual Channel Extended Capability Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																				
Next Capability Offset 2 (C34h)																Capability Version (1h)				PCI Express Extended Capability ID (0002h)																148h
Port VC Capability 1																																14Ch				
Port VC Capability 2																																150h				
Port VC Status ( <i>Reserved</i> )																Port VC Control																154h				
VC0 Resource Capability																																158h				
VC0 Resource Control																																15Ch				
VC0 Resource Status																<i>Reserved</i>																160h				
<i>Reserved</i>																																164h – 1BCh				



**Register 15-37. 148h Virtual Channel Extended Capability Header**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>PCI Express Extended Capability ID</b> Program to 0002h, as required by the <i>PCI Express Base r3.0</i> .	RO	No	0002h
19:16	<b>Capability Version</b> Program to 1h, as required by the <i>PCI Express Base r3.0</i> .	RO	No	1h
31:20	<b>Next Capability Offset 2</b> Next extended capability is the <b>Vendor-Specific Extended Capability 2</b> structure, offset C34h.	RO	Yes	C34h

**Register 15-38. 14Ch Port VC Capability 1**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
2:0	<b>Extended VC Counter</b> 000b = NT Port Virtual Interface supports only one Virtual Channel, VC0 001b = <b>Reserved</b> , because the PEX 8748 supports only one VC All other encodings are <b>Reserved</b> .	RO	No	000b
3	<b>Reserved</b>	RsvdP	No	0
6:4	<b>Low-Priority Extended VC Counter</b> For Strict Priority arbitration, indicates the quantity of extended Virtual Channels (VCs) (those in addition to VC0) that belong to the Low-Priority VC group for the NT Port Virtual Interface. 000b = For the NT Port Virtual Interface, only VC0 belongs to the Low-Priority VC group 001b = <b>Reserved</b> , because the PEX 8748 supports only one VC All other encodings are <b>Reserved</b> .	RO	Yes	000b
7	<b>Reserved</b>	RsvdP	No	0
9:8	<b>Reference Clock</b> <b>Reserved</b>	RsvdP	No	00b
11:10	<b>Port Arbitration Table Entry Size</b>	RO	Yes	00b
31:12	<b>Reserved</b>	RsvdP	No	0000_0h

**Register 15-39. 158h VC0 Resource Capability**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>Port Arbitration Capability</b> Bit 0 = 1 – Non-configurable Round-Robin (Hardware-Fixed) arbitration Bit 1 = 1 – Weighted Round-Robin (WRR) arbitration with 64 Phases	RO	No	00b
13:2	<b>Reserved</b>	RsvdP	No	0-0h
14	<b>Advanced Packet Switching</b>	RsvdP	No	0
15	<b>Reject Snoop Transactions</b> Not a PCI Express switch feature; therefore, this bit is Cleared.	RsvdP	No	0
22:16	<b>Maximum Time Slots</b> <b>Reserved</b>	RsvdP	No	000_0000b
23	<b>Reserved</b>	RsvdP	No	0
31:24	<b>Port Arbitration Table Offset</b>	RO	No	00h

**Register 15-40. 160h VC0 Resource Status**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Reserved</b>	RsvdP	No	0000h
16	<b>Port Arbitration Table Status</b> <i>Not implemented</i>	RO	No	0
17	<b>VC0 Negotiation Pending</b> 0 = VC0 negotiation completed (NT Port Virtual Interface Link to the internal virtual PCI Bus is up) 1 = VC0 initialization is not complete for the NT Port Link Interface	RO	Yes	0
31:18	<b>Reserved</b>	RsvdP	No	0-0h

## 15.11 NT Port Virtual Interface Device-Specific Registers (Offsets 1C0h – A74h)

The registers detailed in [Section 13.16, “Device-Specific Registers \(Offsets 1C0h – A74h\)”](#) are unique to the PEX 8748 and not referenced in the *PCI Express Base r3.0*. These registers are also applicable to the NT Port Virtual Interface, except as defined in [Table 15-8](#) (register map; offsets 1D0h through 1D8h, 200h through 270h, 290h through 46Ch, 600h through 68Ch, 6A0h through 6A8h, 760h through 8FCh, 900h through 93Ch, and 9E0h through A50h are **Reserved**) and [Table 15-9](#), and [Register 15-41](#) and [Register 15-42](#).

Other NT Port Virtual Interface Device-Specific registers are detailed in:

- [Section 15.12, “NT Port Virtual Interface Device-Specific Registers \(Offsets B70h – C88h\)”](#)
- [Section 15.15, “NT Port Virtual Interface Device-Specific Registers – Link Error \(Offsets FE0h – FFCh\)”](#)

**Note:** *It is recommended that these registers not be changed from their default values.*

**Table 15-8. NT Port Virtual Interface Device-Specific Register Map (Offsets 1C0h – A74h)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
<b>Reserved</b>		1C0h – 1D8h
NT Port Virtual Interface Device-Specific Registers – Captured Bus Number and Device Number (Offsets 1DCh – 1FCh)		1DCh ... 1FCh
<b>Factory Test Only/Reserved</b>		200h – A74h

15.11.1 NT Port Virtual Interface Device-Specific Registers – Captured Bus Number and Device Number (Offsets 1DCh – 1FCh)

The registers detailed in [Section 13.16.2, “Device-Specific Registers – Captured Bus and Device Numbers \(Offsets 1DCh – 1E4h\),”](#) are also applicable to the NT Port Virtual Interface, except as defined in [Table 15-9](#) (register map; offset 1E0h is not *Reserved*; offset 1E4h is *Reserved*), and [Register 15-41](#) and [Register 15-42](#).

Table 15-9. NT Port Virtual Interface Device-Specific Captured Bus Number and Device Number Register Map

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
NT Captured Bus Number																																1DCh
NT Captured Device Number																																1E0h
Reserved																																1E4h – 1FCh

**Register 15-41. 1DCh NT Captured Bus Number**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Captured Bus Number</b> Captured Bus Number value for the NT Port Virtual Interface. The value of this field can be overwritten, if bit 31 ( <i>Captured BusDev Number Override</i> ) is Set prior to changing the Captured Bus Number. <i>Note: Overwriting the Captured Bus Number value is not recommended.</i>	RW	Yes	00h
12:8	<b>Captured Device Number</b> Captured Device Number value for the NT Port Virtual Interface. The value of this field can be overwritten, if bit 31 ( <i>Captured BusDev Number Override</i> ) is Set prior to changing the Captured Device Number. <i>Note: Overwriting the Captured Device Number value is not recommended.</i>	RW	Yes	0-0h
30:13	<b>Reserved</b>	RsvdP	No	0-0h
31	<b>Captured BusDev Number Override</b> 1 = Enables the Captured Bus Number and Device Number to be overridden	RW	Yes	0

**Register 15-42. 1E0h NT Captured Device Number**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Captured Device Number</b> NT Port Virtual Interface Captured Device Number register value. <i>Note: Overwriting the Captured Device Number value is not recommended.</i>	RWS	Yes	00h
31:8	<b>Reserved</b>	RsvdP	No	0000_00h

## 15.12 NT Port Virtual Interface Device-Specific Registers (Offsets B70h – C88h)

The registers detailed in [Section 13.18, “Device-Specific Registers \(Offsets B70h – DFCh\)”](#) (for offsets B70h through C88h), are unique to the PEX 8748 and not referenced in the *PCI Express Base r3.0*. These registers are also applicable to the NT Port Virtual Interface, except as defined in [Table 15-10](#) (register map; offsets B80h through C30h are **Reserved**, and offsets C34h through C88h are associated with NT Port-specific registers) and [Table 15-11](#), and [Register 15-43](#) through [Register 15-64](#).

Other NT Port Virtual Interface Device-Specific registers are detailed in:

- Section 15.11, “NT Port Virtual Interface Device-Specific Registers (Offsets 1C0h – A74h)”
- Section 15.15, “NT Port Virtual Interface Device-Specific Registers – Link Error (Offsets FE0h – FFCh)”

**Note:** *It is recommended that these registers not be changed from their default values.*

**Table 15-10. NT Port Virtual Interface Device-Specific Register Map (Offsets B70h – C88h)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																				
Next Capability Offset 2 (000h)																1h				PCI Express Extended Capability ID 2 (000Bh)																B70h
Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets B70h – B7Ch)																																...				
Reserved																																B80h –				
Next Capability Offset 4 (B70h)																1h				PCI Express Extended Capability ID 4 (000Bh)																C34h
NT Port Virtual Interface Device-Specific Registers – Vendor-Specific Extended Capability 4 (Offsets C34h – C88h)																																...				
																																C88h				

### 15.12.1 NT Port Virtual Interface Device-Specific Registers – Vendor-Specific Extended Capability 4 (Offsets C34h – C88h)

This section details the NT Port Virtual Interface Device-Specific, Vendor-Specific Extended Capability 4 registers, which include the **Memory BARx Address Translation**, **Doorbell**, and **Scratchpad** registers. Table 15-11 defines the register map used by the NT Port Virtual Interface.

**Table 15-11. NT Port Virtual Interface Device-Specific, Vendor-Specific Extended Capability 4 Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																
Next Capability Offset 4 (B70h)												Capability Version 4 (1h)				PCI Express Extended Capability ID 4 (000Bh)																C34h
Vendor-Specific Header 4																																C38h
Memory BAR2 Address Translation Lower																																C3Ch
Memory BAR3 Address Translation Upper																																C40h
Memory BAR4 Address Translation Lower																																C44h
Memory BAR5 Address Translation Upper																																C48h
Reserved																Virtual Interface IRQ Set																C4Ch
Reserved																Virtual Interface IRQ Clear																C50h
Reserved																Virtual Interface IRQ Mask Set																C54h
Reserved																Virtual Interface IRQ Mask Clear																C58h
Reserved																Link Interface IRQ Set																C5Ch
Reserved																Link Interface IRQ Clear																C60h
Reserved																Link Interface IRQ Mask Set																C64h
Reserved																Link Interface IRQ Mask Clear																C68h
NT Port SCRATCH0																																C6Ch
NT Port SCRATCH1																																C70h
NT Port SCRATCH2																																C74h
NT Port SCRATCH3																																C78h
NT Port SCRATCH4																																C7Ch
NT Port SCRATCH5																																C80h
NT Port SCRATCH6																																C84h
NT Port SCRATCH7																																C88h

**Register 15-43. C34h Vendor-Specific Extended Capability 4**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>PCI Express Extended Capability ID 4</b> Program to 000Bh, to indicate that the Extended Capability structure is the <b>Vendor-Specific Extended Capability</b> structure.	ROS	Yes	000Bh
19:16	<b>Capability Version 4</b>	ROS	Yes	1h
31:20	<b>Next Capability Offset 4</b> Program to B70h, which addresses the <b>Vendor-Specific Extended Capability 4</b> structure.	ROS	Yes	B70h

**Register 15-44. C38h Vendor-Specific Header 4**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Vendor-Specific ID 4</b> ID Number of this Extended Capability structure.	ROS	Yes	0003h
19:16	<b>Vendor-Specific Rev 4</b> Version Number of this structure.	ROS	Yes	0h
31:20	<b>Vendor-Specific Length 4</b> Quantity of bytes in the entire structure.	ROS	Yes	078h



**Register 15-45. C3Ch Memory BAR2 Address Translation Lower**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<i>Reserved</i>	RsvdP	No	0_0000h
31:20	<b>NT Port Virtual-to-Link Interface BAR2 Base Translation Address</b> NT Port Virtual-to-Link Interface Base Translation address when <b>BAR2</b> is enabled ( <b>NT Port Virtual Interface Memory BAR2 Setup</b> register <b>BAR2 Enable</b> bit (NT Port Virtual Interface, offset D4h[31]) is Set).	RW	Yes	000h

**Register 15-46. C40h Memory BAR3 Address Translation Upper**

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<i>Reserved</i>	Offset D8h[31]=0	RsvdZ	No	0_0000h
	When <b>BAR2/3</b> are used as a 64-bit BAR, bits [31:0] (including bits [19:0]) are used as the upper 32 bits.	Offset D8h[31]=1	RW	Yes	0_0000h
31:20	<b>NT Port Virtual-to-Link Interface BAR3 Base Translation Address</b> NT Port Virtual-to-Link Interface Base Translation address when <b>BAR3</b> is enabled ( <b>NT Port Virtual Interface Memory BAR2/3 Setup</b> register <i>BAR3 Enable</i> bit (NT Port Virtual Interface, offset D8h[31]) is Set).		RW	Yes	000h

**Register 15-47. C44h Memory BAR4 Address Translation Lower**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<i>Reserved</i>	RsvdP	No	0_0000h
31:20	<b>NT Port Virtual-to-Link Interface BAR4 Base Translation Address</b> NT Port Virtual-to-Link Interface Base Translation address when <b>BAR4</b> is enabled ( <b>NT Port Virtual Interface Memory BAR4 Setup</b> register <b>BAR4 Enable</b> bit (NT Port Virtual Interface, offset DCh[31]) is Set).	RW	Yes	000h

**Register 15-48. C48h Memory BAR5 Address Translation Upper**

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<i>Reserved</i>	Offset E0h[31]=0	RsvdZ	No	0_0000h
	When <b>BAR4/5</b> are used as a 64-bit BAR, bits [31:0] (including bits [19:0]) are used as the upper 32 bits.	Offset E0h[31]=1	RW	Yes	0_0000h
31:20	<b>NT Port Virtual-to-Link Interface BAR5 Base Translation Address</b> NT Port Virtual-to-Link Interface Base Translation address when <b>BAR5</b> is enabled ( <b>NT Port Virtual Interface Memory BAR4/5 Setup</b> register <i>BAR5 Enable</i> bit (NT Port Virtual Interface, offset E0h[31]) is Set).		RW	Yes	000h

**Register 15-49. C4Ch Virtual Interface IRQ Set**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> The bits in this register can be masked by their respective <i>Virtual Interface IRQ Mask Set</i> register bits (offset C54h).				
15:0	<b>SET_IRQ</b> Set Virtual IRQ. Controls the state of the Virtual Interface Doorbell Interrupt Request. Reading returns the status of the bits. Writing 0 to a bit in the register has no effect. Writing 1 to a bit in the register Sets the corresponding Interrupt Request. The Virtual Interface interrupt is asserted when the following conditions are met: <ul style="list-style-type: none"> <li>• This register (offset C4Ch or C50h) value is non-zero, and,</li> <li>• Corresponding <b>Virtual Interface IRQ Mask Set</b> or <b>Virtual Interface IRQ Mask Clear</b> register (offset C54h or C58h, respectively) <i>Interrupt Mask</i> bit is not Set, and,</li> <li>• Interrupts (either INTx or MSIs) are enabled</li> </ul>	RW1S	Yes	0000h
31:16	<b>Reserved</b>	RsvdP	No	0000h

**Register 15-50. C50h Virtual Interface IRQ Clear**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> The bits in this register can be masked by their respective <i>Virtual Interface IRQ Mask Clear</i> register bits (offset C58h).				
15:0	<b>CLR_IRQ</b> Clear Virtual IRQ. Controls the state of the Virtual Interface Doorbell Interrupt Request. Reading returns the status of the bits. Writing 0 to a bit in the register has no effect. Writing 1 to a bit in the register Clears the corresponding Interrupt Request. The Virtual Interface interrupt is de-asserted when the following conditions are met: <ul style="list-style-type: none"> <li>• This register (offset C50h or C4Ch) value is zero (0), –or–</li> <li>• <b>Virtual Interface IRQ Mask Set</b> or <b>Virtual Interface IRQ Mask Clear</b> register (offset C54h or C58h, respectively) masks all its <b>Set</b> or <b>Clear</b> register (offset C50h or C4Ch) <i>Set</i> bits, and</li> <li>• INTx interrupts are enabled</li> </ul>	RW1C	Yes	0000h
31:16	<b>Reserved</b>	RsvdP	No	0000h

**Register 15-51. C54h Virtual Interface IRQ Mask Set**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> The bits in this register can be used to mask their respective <b>Virtual Interface IRQ Set</b> register bits (offset C4Ch).				
15:0	<b>SET_IRQM</b> Virtual Interface interrupt IRQ Mask Set. Reading returns the state of the <i>Interrupt Mask</i> bits. Writing 0 to a bit in the register has no effect. Writing 1 to a bit in the register Clears the corresponding <i>Interrupt Mask</i> bit. 0 = Corresponding <b>Virtual Interface IRQ Set</b> register (offset C4Ch) <i>Interrupt Request</i> bit is not masked 1 = Corresponding <b>Virtual Interface IRQ Set</b> register (offset C4Ch) <i>Interrupt Request</i> bit is masked/disabled	RW1S	Yes	FFFFh
31:16	<b>Reserved</b>	RsvdP	No	0000h

**Register 15-52. C58h Virtual Interface IRQ Mask Clear**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> The bits in this register can be used to mask their respective <b>Virtual Interface IRQ Clear</b> register bits (offset C50h).				
15:0	<b>CLR_IRQM</b> Clear Virtual IRQ Mask. Controls the state of the Virtual Interface <i>Interrupt Request</i> bits. Reading returns the state of the <i>Interrupt Mask</i> bits. Writing 0 to a bit in the register has no effect. Writing 1 to a bit in the register Clears the corresponding <i>Interrupt Mask</i> bit. 0 = Corresponding <b>Virtual Interface IRQ Clear</b> register (offset C50h) <i>Interrupt Request</i> bit is not masked 1 = Corresponding <b>Virtual Interface IRQ Clear</b> register (offset C50h) <i>Interrupt Request</i> bit is masked/disabled	RW1C	Yes	FFFFh
31:16	<b>Reserved</b>	RsvdP	No	0000h

**Register 15-53. C5Ch Link Interface IRQ Set**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> The bits in this register can be masked by their respective <i>Link Interface IRQ Mask Set</i> register bits (offset C64h).				
15:0	<b>SET_IRQ</b> Set Link IRQ. Controls the state of the Link Interface Doorbell Interrupt Request. Reading returns the status of the bits. Writing 0 to a bit in the register has no effect. Writing 1 to a bit in the register Sets the corresponding Interrupt Request. The Link Interface interrupt is asserted when the following conditions are met: <ul style="list-style-type: none"> <li>• This register (offset C5Ch or C60h) value is non-zero, and,</li> <li>• Corresponding <b>Link Interface IRQ Mask Set</b> or <b>Link Interface IRQ Mask Clear</b> register (offset C64h or C68h, respectively) <i>Interrupt Mask</i> bit is not Set, and,</li> <li>• Interrupts (either INTx or MSIs) are enabled</li> </ul>	RW1S	Yes	0000h
31:16	<b>Reserved</b>	RsvdP	No	0000h

**Register 15-54. C60h Link Interface IRQ Clear**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> The bits in this register can be masked by their respective <i>Link Interface IRQ Mask Clear</i> register bits (offset C68h).				
15:0	<b>CLR_IRQ</b> Clear Virtual IRQ. Controls the state of the Virtual Interface Doorbell Interrupt Request. Reading returns the status of the bits. Writing 0 to a bit in the register has no effect. Writing 1 to a bit in the register Clears the corresponding Interrupt Request. The Virtual Interface interrupt is de-asserted when the following conditions are met: <ul style="list-style-type: none"> <li>• This register (offset C60h or C5Ch) value is zero (0), –or–</li> <li>• <b>Link Interface IRQ Mask Set</b> or <b>Link Interface IRQ Mask Clear</b> register (offset C64h or C68h, respectively) masks all its <b>Set</b> or <b>Clear</b> register (offset C60h or C5Ch) <i>Set</i> bits, and</li> <li>• INTx interrupts are enabled</li> </ul>	RW1C	Yes	0000h
31:16	<b>Reserved</b>	RsvdP	No	0000h

**Register 15-55. C64h Link Interface IRQ Mask Set**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> The bits in this register can be used to mask their respective <b>Link Interface IRQ Set</b> register bits (offset C5Ch).				
15:0	<b>SET_IRQM</b> Link Interface Interrupt IRQ Mask Set. Reading returns the state of the <i>Interrupt Mask</i> bits. Writing 0 to a bit in the register has no effect. Writing 1 to a bit in the register Sets the corresponding <i>Interrupt Mask</i> bit. 0 = Corresponding <b>Link Interface IRQ Set</b> register (offset C5Ch) <i>Interrupt Request</i> bit is not masked 1 = Corresponding <b>Link Interface IRQ Set</b> register (offset C5Ch) <i>Interrupt Request</i> bit is masked/disabled	RW1S	Yes	FFFFh
31:16	<b>Reserved</b>	RsvdP	No	0000h

**Register 15-56. C68h Link Interface IRQ Mask Clear**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> The bits in this register can be used to mask their respective <b>Link Interface IRQ Clear</b> register bits (offset C60h).				
15:0	<b>CLR_IRQM</b> Link Interface Interrupt IRQ Mask Clear. Reading returns the state of the <i>Interrupt Mask</i> bits. Writing 0 to a bit in the register has no effect. Writing 1 to a bit in the register Clears the corresponding <i>Interrupt Mask</i> bit. 0 = Corresponding <b>Link Interface IRQ Clear</b> register (offset C60h) <i>Interrupt Request</i> bit is not masked 1 = Corresponding <b>Link Interface IRQ Clear</b> register (offset C60h) <i>Interrupt Request</i> bit is masked/disabled	RW1C	Yes	FFFFh
31:16	<b>Reserved</b>	RsvdP	No	0000h

**Register 15-57. C6Ch NT Port SCRATCH0**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 0</b> 32-bit Scratchpad 0 register. Reset only by Station Reset.	RW	Yes	0000_0000h

**Register 15-58. C70h NT Port SCRATCH1**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 1</b> 32-bit Scratchpad 1 register. Reset only by Station Reset.	RW	Yes	0000_0000h

**Register 15-59. C74h NT Port SCRATCH2**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 2</b> 32-bit Scratchpad 2 register. Reset only by Station Reset.	RW	Yes	0000_0000h

**Register 15-60. C78h NT Port SCRATCH3**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 3</b> 32-bit Scratchpad 3 register. Reset only by Station Reset.	RW	Yes	0000_0000h

**Register 15-61. C7Ch NT Port SCRATCH4**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 4</b> 32-bit Scratchpad 4 register. Reset only by Station Reset.	RW	Yes	0000_0000h

**Register 15-62. C80h NT Port SCRATCH5**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 5</b> 32-bit Scratchpad 5 register. Reset only by Station Reset.	RW	Yes	0000_0000h

**Register 15-63. C84h NT Port SCRATCH6**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 6</b> 32-bit Scratchpad 6 register. Reset only by Station Reset.	RW	Yes	0000_0000h

**Register 15-64. C88h NT Port SCRATCH7**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Scratchpad 7</b> 32-bit Scratchpad 7 register. Reset only by Station Reset.	RW	Yes	0000_0000h

# 15.13 NT Port Virtual Interface NT Bridging-Specific Registers (Offsets C8Ch – DFCh)

Table 15-12 defines the register map of the NT Port Virtual Interface NT Bridging-Specific registers.

Table 15-12. NT Port Virtual Interface NT Bridging-Specific Register Map

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
NT Bridging-Specific Registers – NT Port ID (Offsets C8Ch – C94h)		C8Ch ... C94h
Reserved		C98h – D90h
NT Port Virtual Interface NT Bridging-Specific Registers – Requester ID Translation Lookup Table Entry (Addresses D94h – DD0h)		D94h ... DD0h
NT Bridging-Specific Registers – Lookup Table Enable Link (Offsets DD4h – DDCh)		DD4h ... DDCh
Reserved		DE0h – DFCh



### 15.13.1 NT Bridging-Specific Registers – NT Port ID (Offsets C8Ch – C94h)

Table 15-13 defines the register map of the NT Port Virtual Interface NT Bridging-Specific NT Port ID register.

**Table 15-13. NT Port Virtual Interface NT Bridging-Specific NT Port ID Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
NT Port ID		C8Ch
Reserved		C90h
Factory Test Only		C94h

**Register 15-65. C8Ch NT Port ID**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>NT Port ID</b> 0 = NT Port 1 = <i>Reserved</i>	ROS	No	0
30:1	<i>Reserved</i>	RsvdP	No	0-0h
31	<b>Virtual Side Identifier</b>	ROS	No	0

### 15.13.2 NT Port Virtual Interface NT Bridging-Specific Registers – Requester ID Translation Lookup Table Entry (Addresses D94h – DD0h)

This section describes the NT Port Virtual Interface NT Bridging-Specific Requester ID Translation Lookup Table (LUT) Entry registers. The NT Port uses these registers for Requester ID translation when it forwards:

- Memory Requests from the NT Port Virtual Interface to the NT Port Link Interface, –or–
- Completion TLPs from the NT Port Link Interface to the NT Port Virtual Interface

If the application needs to send traffic through the NT Port Virtual Interface, program the registers listed in this group with the corresponding Requester's Requester ID, then Set the *LUT Entry\_n Enable* and *LUT Entry\_m Enable* bits (bits 0 and 16, respectively) for each LUT entry, as needed.

Table 15-14 defines the register and address locations, as they relate to Register 15-66.

The Requester ID LUT can be one of two formats, selected by the **Ingress Chip Control** register *LUT Format* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 764h[2]):

1. **LUT Format bit is Cleared** – LUT entry is not qualified by the TLP Requester ID Function Number. The LUT format is:
  - Two entries per 32-bit register
  - Each 16-bit entry is defined as follows:
    - Bits [15:8] = Bus Number
    - Bits [7:3] = Device Number
    - Bit 2 = *Not used*
    - Bit 1 = No Snoop Control
    - Bit 0 = Enable
2. **LUT Format bit is Set** – LUT entry can be qualified by the TLP Requester ID Function Number. The Table format is:
  - Two entries per 32-bit register
  - Each 16 bit-entry is defined as follows:
    - Bits [15:8] = Bus Number
    - Bits [7:3] = Device Number
    - bits [2:0] = Function Number
  - Separate 32-bit Requester ID Enable register (**Lookup Table Enable Virtual** register, offset DD4h), in which each bit is the *Enable* bit for the corresponding LUT entry
  - Separate 32-bit No Snoop register (**Lookup Table No Snoop Enable Virtual** register, offset DD8h), in which each bit is the *No Snoop Enable* bit for the corresponding LUT entry
  - A separate 32-bit Function Number Enable register (**Lookup Table Function Enable Virtual** register, offset DDCh), in which each bit is the *Enable* bit that specifies whether the Function Number is included in the decode

**Table 15-14. NT Port Virtual Interface NT Bridging-Specific Requester ID  
Translation Lookup Table Entry\_n\_m Register Locations**

ADDR Location	Lookup Table Entry_n_m	ADDR Location	Lookup Table Entry_n_m
D94h	0_1	DB4h	16_17
D98h	2_3	DB8h	18_19
D9Ch	4_5	DBCCh	20_21
DA0h	6_7	DC0h	22_23
DA4h	8_9	DC4h	24_25
DA8h	10_11	DC8h	26_27
DACCh	12_13	DCCCh	28_29
DB0h	14_15	DD0h	30_31

**Register 15-66. D94h – DD0h NT Port Virtual Interface Requester ID Translation**  
**LUT Entry<sub>n\_m</sub> (where  $n_m = 0\_1$  through 30\_31)**

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>LUT Entry<sub>n</sub> Enable</b> 0 = Disables 1 = Enables		RW	Yes	0
1	<b>LUT Entry<sub>n</sub> No Snoop Enable</b> If Set, the NT Port Clears the TLP <i>No Snoop</i> attribute bit for the Memory Request, then goes from the NT Port Virtual Interface to the NT Port Link Interface, and re-calculates the ECRC. If the original TLP has an ECRC error, the NT Port corrupts the re-calculated ECRC before transmitting to the other Host domain. The NT Port sets the <i>No Snoop</i> attribute bit when it forwards the Completion TLP from the NT Port Link Interface to the NT Port Virtual Interface if this bit is Set for the corresponding Requester ID entry. This ECRC rule applies to Completion TLPs as well. 0 = Disables 1 = Enables		RW	Yes	0
2	<b>LUT Entry<sub>n</sub> Function Number Bit 2</b>		RW	Yes	0
7:3	<b>Requester ID on Link Side</b>	<b>Device Number</b> LUT Entry <sub>n</sub> Requester Device Number.	RW	Yes	0000_0b
15:8		<b>Bus Number</b> LUT Entry <sub>n</sub> Requester Bus Number.	RW	Yes	00h
16	<b>LUT Entry<sub>m</sub> Enable</b> 0 = Disables 1 = Enables		RW	Yes	0
17	<b>LUT Entry<sub>m</sub> No Snoop Enable</b> If Set, the NT Port Clears the TLP <i>No Snoop</i> attribute bit for the Memory Request, then goes from the NT Port Virtual Interface to the NT Port Link Interface, and re-calculates the ECRC. If the original TLP has an ECRC error, the NT Port corrupts the re-calculated ECRC before transmitting to the other Host domain. The NT Port sets the <i>No Snoop</i> attribute bit when it forwards the Completion TLP from the NT Port Link Interface to the NT Port Virtual Interface if this bit is Set for the corresponding Requester ID entry. This ECRC rule applies to Completion TLPs as well. 0 = Disables 1 = Enables		RW	Yes	0
18	<b>LUT Entry<sub>n</sub> Function Number Bit 2</b>		RW	Yes	0
23:19	<b>Requester ID on Link Side</b>	<b>Device Number</b> LUT Entry <sub>m</sub> Requester Device Number.	RW	Yes	0000_0b
31:24		<b>Bus Number</b> LUT Entry <sub>m</sub> Requester Bus Number.	RW	Yes	00h

### 15.13.3 NT Bridging-Specific Registers – Lookup Table Enable Link (Offsets DD4h – DDCh)

Table 15-15 defines the register map of the NT Port Virtual Interface NT Bridging-Specific Lookup Table Enable Link registers. Information describing when these registers are enabled, and how they are used, is provided in Section 15.3.2.

**Table 15-15. NT Port Virtual Interface NT Bridging-Specific Lookup Table Enable Link Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
Lookup Table Enable Virtual		DD4h
Lookup Table No Snoop Enable Virtual		DD8h
Lookup Table Function Enable Virtual		DDCh

#### Register 15-67. DD4h Lookup Table Enable Virtual

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Lookup Table Enable</b> Enable bit per Lookup Table.	RWS	Yes	0000_0000h

#### Register 15-68. DD8h Lookup Table No Snoop Enable Virtual

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Lookup Table No Snoop Enable</b> No Snoop Enable bit per Lookup Table.	RWS	Yes	0000_0000h

#### Register 15-69. DDCh Lookup Table Function Enable Virtual

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Lookup Table Function Check Enable</b> Function Check Enable bit per Lookup Table.	RWS	Yes	0000_0000h

## 15.14 NT Port Virtual Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)

The registers detailed in [Section 13.23, “Advanced Error Reporting Extended Capability Registers \(Offsets FB4h – FDCh\),”](#) are also applicable to the NT Port Virtual Interface, except as defined in [Table 15-16](#) (register map), and [Register 15-70](#) through [Register 15-75](#).

**Table 15-16. NT Port Virtual Interface Advanced Error Reporting Extended Capability Register Map**

31 30 29 28 27 26 25 24 23 22 21 20																19 18 17 16				15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
Next Capability Offset (148h)												Capability Version (1h)				PCI Express Extended Capability ID (0001h)																FB4h			
Reserved				Uncorrectable Error Status																FB8h															
Reserved				Uncorrectable Error Mask																FBCh															
Reserved				Uncorrectable Error Severity																FC0h															
Reserved												Correctable Error Status																FC4h							
Reserved												Correctable Error Mask																FC8h							
Advanced Error Capabilities and Control																				FCCh															
Header Log 0																				FD0h															
Header Log 1																				FD4h															
Header Log 2																				FD8h															
Header Log 3																				FDCh															

**Register 15-70. FB4h Advanced Error Reporting Extended Capability Header**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	PCI Express Extended Capability ID	ROS	Yes	0001h
19:16	Capability Version	ROS	Yes	1h
31:20	Next Capability Offset Program to 148h, which addresses the <b>Virtual Channel Extended Capability</b> structure.	ROS	Yes	148h

**Register 15-71. FB8h Uncorrectable Error Status**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>The following PCI Express errors are not valid for the NT Port Virtual Interface:</p> <ul style="list-style-type: none"> <li>Data Link Protocol error</li> <li>Surprise Down error</li> </ul> <p><b>Note:</b> If an individual error is masked (corresponding bit in the <b>Uncorrectable Error Mask</b> register (offset <b>FBCh</b>) is Set) when the error is detected, its Error Status bit is still updated; however, an error reporting Message (<b>ERR_FATAL</b> or <b>ERR_NONFATAL</b>) is <b>not</b> sent to the Root Complex, and the <b>Advanced Error Capabilities and Control</b> register <b>First Error Pointer</b> field (offset <b>FCCh[4:0]</b>) and <b>Header Log x</b> registers (offsets <b>FD0h</b> through <b>FDCh</b>) remain unchanged.</p>				
3:0	<b>Reserved</b>	RsvdP	No	0h
4	<b>Data Link Protocol Error Status</b> 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0
5	<b>Surprise Down Error Status</b> <b>Reserved</b>	RsvdP	No	0
11:6	<b>Reserved</b>	RsvdP	No	0-0h
12	<b>Poisoned TLP Status</b> 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0
13	<b>Flow Control Protocol Error Status</b> <b>Reserved/Not supported</b>	RsvdP	No	0
14	<b>Completion Timeout Status</b> <b>Not applicable to switches.</b>	RsvdP	No	0
15	<b>Completer Abort Status</b>	RW1CS <sup>a</sup>	Yes	0
16	<b>Unexpected Completion Status</b> 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0
17	<b>Receiver Overflow Status</b> 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0
18	<b>Malformed TLP Status</b> 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0
19	<b>ECRC Error Status</b> 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0

**Register 15-71. FB8h Uncorrectable Error Status (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
20	<b>Unsupported Request Error Status</b> 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0
21	<b>Reserved</b>	RsvdP	No	0
22	<b>Uncorrectable Internal Error Status</b> <b>Reserved</b>	RsvdP	No	0
23	<b>MC Blocked TLP Status</b> Multicast blocked TLP status. 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0
31:24	<b>Reserved</b>	RsvdP	No	00h

- a. When the **ECC Error Check Disable** register **Software Force Error Enable** bit (Transparent Downstream Ports, offset 720h[2]) is Set, Type changes from RW1CS to RW.



**Register 15-72. FBCh Uncorrectable Error Mask**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>The following PCI Express errors are not valid for the NT Port Virtual Interface:</p> <ul style="list-style-type: none"> <li>• Data Link Protocol error</li> <li>• Surprise Down error</li> </ul> <p><b>Note:</b> The bits in this register can be used to mask their respective <i>Uncorrectable Error Status</i> register bits (offset FB8h).</p>				
3:0	<b>Reserved</b>	RsvdP	No	0h
4	<b>Data Link Protocol Error Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error	RWS	Yes	0
5	<b>Surprise Down Error Mask</b> <b>Reserved</b>	RsvdP	No	0
11:6	<b>Reserved</b>	RsvdP	No	0-0h
12	<b>Poisoned TLP Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error	RWS	Yes	0
13	<b>Flow Control Protocol Error Mask</b> <b>Reserved/Not supported</b>	RsvdP	No	0
14	<b>Completion Timeout Mask</b>	RWS	Yes	0
15	<b>Completer Abort Mask</b>	RWS	Yes	0

**Register 15-72. FBCh Uncorrectable Error Mask (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Unexpected Completion Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error	RWS	Yes	0
17	<b>Receiver Overflow Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error	RWS	Yes	0
18	<b>Malformed TLP Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error	RWS	Yes	0
19	<b>ECRC Error Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error	RWS	Yes	0
20	<b>Unsupported Request Error Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error	RWS	Yes	0
21	<i>Reserved</i>	RsvdP	No	0
22	<b>Uncorrectable Internal Error Mask</b> <i>Reserved</i>	RsvdP	No	0
23	<b>MC Blocked TLP Mask</b> 0 = No mask is Set 1 = Masks error reporting, first error update, and Header logging for this error	RWS	Yes	0
31:24	<i>Reserved</i>	RsvdP	No	00h

**Register 15-73. FC0h Uncorrectable Error Severity**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
The following PCI Express errors are not valid for the NT Port Virtual Interface: <ul style="list-style-type: none"> <li>Data Link Protocol error</li> <li>Surprise Down error</li> </ul>				
3:0	<b>Reserved</b>	RsvdP	No	0h
4	<b>Data Link Protocol Error Severity</b> 0 = Error is reported as non-fatal 1 = Error is reported as fatal	RWS	Yes	1
5	<b>Surprise Down Error Severity</b> 0 = Error is reported as non-fatal 1 = Error is reported as fatal	ROS	No	1
11:6	<b>Reserved</b>	RsvdP	No	0-0h
12	<b>Poisoned TLP Severity</b> 0 = Error is handled as an Advisory Non-Fatal error, and reported as a Correctable error 1 = Error is reported as fatal	RWS	Yes	0
13	<b>Flow Control Protocol Error Severity</b>	ROS	No	1
14	<b>Completion Timeout Severity</b> Because the Status and Mask are both <b>Reserved</b> for this bit, Severity can be ignored.	RsvdP	No	0
15	<b>Completer Abort Severity</b> 0 = Error is handled as an Advisory Non-Fatal error, and reported as a Correctable error 1 = Error is reported as fatal	RWS	Yes	0

**Register 15-73. FC0h Uncorrectable Error Severity (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
16	<b>Unexpected Completion Severity</b> 0 = Error is handled as an Advisory Non-Fatal error, and reported as a Correctable error 1 = Error is reported as fatal	RWS	Yes	0
17	<b>Receiver Overflow Severity</b> 0 = Error is reported as non-fatal 1 = Error is reported as fatal	RWS	Yes	1
18	<b>Malformed TLP Severity</b> 0 = Error is reported as non-fatal 1 = Error is reported as fatal	RWS	Yes	1
19	<b>ECRC Error Severity</b> 0 = Error is handled as an Advisory Non-Fatal error, and reported as a Correctable error 1 = Error is reported as fatal	RWS	Yes	0
20	<b>Unsupported Request Error Severity</b> 0 = Error is handled as an Advisory Non-Fatal error, and reported as a Correctable error 1 = Error is reported as fatal	RWS	Yes	0
21	<b>Reserved</b>	RsvdP	No	0
22	<b>Uncorrectable Internal Error Severity</b> <b>Reserved</b>	RsvdP	No	0
23	<b>MC Blocked TLP Severity</b> 0 = Error is reported as non-fatal 1 = Error is reported as fatal	RWS	Yes	0
31:24	<b>Reserved</b>	RsvdP	No	00h

**Register 15-74. FC4h Correctable Error Status**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>The following PCI Express errors are not valid for the NT Port Virtual Interface:</p> <ul style="list-style-type: none"> <li>• Receiver error</li> <li>• Bad TLP error</li> <li>• Bad Data Link Layer Packet (DLLP) error</li> <li>• Replay Number Rollover error</li> <li>• Replay Timer Timeout error</li> </ul> <p><b>Note:</b> If an individual error is masked (corresponding bit in the <b>Correctable Error Mask</b> register (offset <b>FC8h</b>) is Set) when the error is detected, its Error Status bit is still updated; however, an error reporting Message (ERR_COR) is <b>not</b> sent to the Root Complex.</p>				
0	<b>Receiver Error Status</b> 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0
5:1	<b>Reserved</b>	RsvdP	No	0-0h
6	<b>Bad TLP Status</b> 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0
7	<b>Bad DLLP Status</b> 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0
8	<b>REPLAY NUM Rollover Status</b> Replay Number Rollover status. 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0
11:9	<b>Reserved</b>	RsvdP	No	000b
12	<b>Replay Timer Timeout Status</b> 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0
13	<b>Advisory Non-Fatal Error Status</b> 0 = No error is detected 1 = Error is detected	RW1CS <sup>a</sup>	Yes	0
14	<b>Corrected Internal Error Status</b> <b>Reserved</b>	RsvdP	No	0
15	<b>Header Log Overflow Status</b> <b>Reserved</b>	RsvdP	No	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

a. When the **ECC Error Check Disable** register **Software Force Error Enable** bit (Transparent Downstream Ports, offset **720h[2]**) is Set, Type changes from RW1CS to RW.

**Register 15-75. FC8h Correctable Error Mask**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p>The following PCI Express errors are not valid for the NT Port Virtual Interface:</p> <ul style="list-style-type: none"> <li>• Receiver error</li> <li>• Bad TLP error</li> <li>• Bad DLLP error</li> <li>• Replay Number Rollover error</li> <li>• Replay Timer Timeout error</li> </ul> <p><b>Note:</b> The bits in this register can be used to mask their respective <i>Correctable Error Status</i> register bits (offset <i>FC4h</i>).</p>				
0	<b>Receiver Error Mask</b> 0 = Error reporting is not masked 1 = Error reporting is masked	RWS	Yes	0
5:1	<b>Reserved</b>	RsvdP	No	0-0h
6	<b>Bad TLP Mask</b> 0 = Error reporting is not masked 1 = Error reporting is masked	RWS	Yes	0
7	<b>Bad DLLP Mask</b> 0 = Error reporting is not masked 1 = Error reporting is masked	RWS	Yes	0
8	<b>REPLAY NUM Rollover Mask</b> Replay Number Rollover mask. 0 = Error reporting is not masked 1 = Error reporting is masked	RWS	Yes	0
11:9	<b>Reserved</b>	RsvdP	No	000b
12	<b>Replay Timer Timeout Mask</b> 0 = Error reporting is not masked 1 = Error reporting is masked	RWS	Yes	0
13	<b>Advisory Non-Fatal Error Mask</b> 0 = Error reporting is not masked 1 = Error reporting is masked	RWS	Yes	1
14	<b>Corrected Internal Error Mask</b> <b>Reserved</b>	RsvdP	No	0
15	<b>Header Log Overflow Mask</b> <b>Reserved</b>	RsvdP	No	0
31:16	<b>Reserved</b>	RsvdP	No	0000h

## 15.15 NT Port Virtual Interface Device-Specific Registers – Link Error (Offsets FE0h – FFCh)

This section details the NT Port Virtual Interface Device-Specific Link Error registers, located at offsets FE0h through FFCh. These Device-Specific registers are unique to the NT Port Virtual Interface and not referenced in the *PCI Express Base r3.0*. [Table 15-17](#) defines the register map used by the NT Port Virtual Interface.

Other NT Port Virtual Interface Device-Specific registers are detailed in:

- [Section 15.11, “NT Port Virtual Interface Device-Specific Registers \(Offsets 1C0h – A74h\)”](#)
- [Section 15.12, “NT Port Virtual Interface Device-Specific Registers \(Offsets B70h – C88h\)”](#)

**Note:** *It is recommended that these registers not be changed from their default values.*

**Table 15-17. NT Port Virtual Interface Device-Specific Register Map – Link Error (NT Port Virtual Interface)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
Link Error Status Virtual		FE0h
Link Error Mask Virtual		FE4h
Reserved		FE8h – FFCh

**Register 15-76. FE0h Link Error Status Virtual  
(NT Port Virtual Interface)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> The bits in this register can be masked by their respective bits in the <a href="#">Link Error Mask Virtual</a> register (NT Port Virtual Interface, offset <a href="#">FE4h</a> ).				
0	<b>Link Side Correctable Error Status</b> 1 = NT Port Link Interface detected a Correctable TLP error, and signaled the interrupt to the Virtual side Host	RW1CS	Yes	0
1	<b>Link Side Uncorrectable Error Status</b> 1 = NT Port Link Interface detected an Uncorrectable TLP error, and signaled the interrupt to the Virtual side Host	RW1CS	Yes	0
2	<b>Link Side DL Active Change Status</b> 1 = NT Port Link Interface <i>DL_Active</i> state change occurred upon detection of an NT Port Link Interface <i>DL_Down</i> state rise edge and fall edge	RW1CS	Yes	0
3	<b>Link Side Uncorrectable Error Message Drop Status</b> 1 = NT Port Link Interface received an Uncorrectable Error Message, and signaled the interrupt to the Virtual side Host	RW1CS	Yes	0
31:4	<b>Reserved</b>	RsvdP	No	0000_000h

**Register 15-77. FE4h Link Error Mask Virtual  
(NT Port Virtual Interface)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> The bits in this register can be used to mask their respective bits in the <a href="#">Link Error Status Virtual</a> register (NT Port Virtual Interface, offset <a href="#">FE0h</a> ).				
0	<b>Link Side Correctable Error Mask</b> 0 = No effect on reporting activity 1 = <a href="#">Link Side Correctable Error Status</a> bit is masked/disabled	RWS	Yes	1
1	<b>Link Side Uncorrectable Error Mask</b> 0 = No effect on reporting activity 1 = <a href="#">Link Side Uncorrectable Error Status</a> bit is masked/disabled	RWS	Yes	1
2	<b>Link Side DL Active Change Mask</b> 0 = No effect on reporting activity 1 = <a href="#">Link Side DL Active Change Status</a> bit is masked/disabled	RWS	Yes	1
3	<b>Link Side Uncorrectable Error Message Drop Mask</b> 0 = No effect on reporting activity 1 = <a href="#">Link Side Uncorrectable Error Message Drop Status</a> bit is masked/disabled	RWS	Yes	1
31:4	<b>Reserved</b>	RsvdP	No	0000_000h





## Chapter 16 NT Port Link Interface Registers – NT Mode

### 16.1 Introduction

**Note:** Check the latest design guides, application notes and errata list for Non-Transparent (NT) usage.

In NT mode, the NT Port includes two sets of Configuration, Capability, Control, and Status registers, to support the Link and Virtual Interfaces. This chapter defines the PEX 8748 NT Port Link Interface registers. Other registers are defined in:

- [Chapter 13, “Transparent Port Registers”](#)
- [Chapter 15, “NT Port Virtual Interface Registers – NT Mode”](#)

All PEX 8748 registers can be accessed by Configuration or Memory Requests.

For further details regarding register names and descriptions, refer to the following specifications:

- *PCI r3.0*
- *PCI Power Mgmt. r1.2*
- *PCI Express Base r3.0*

## 16.2 NT Port Link Interface Type 0 Register Map

Table 16-1 defines the NT Port Link Interface Type 0 register mapping.

**Table 16-1. NT Port Link Interface Type 0 Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																				
NT Port Link Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch)																																				00h
																																				...
																																				34h
NT Port Link Interface PCI Power Management Capability Registers (Offsets 40h – 44h)																																				...
																																				3Ch
NT Port Link Interface PCI Express Capability Registers (Offsets 68h – A0h)																Next Capability Pointer (48h)	Capability ID (01h)	40h																		
																NT Port Link Interface MSI Capability Registers (Offsets 48h – 64h)																Next Capability Pointer (68h)	Capability ID (05h)	44h		
NT Port Link Interface PCI Express Capability Registers (Offsets 68h – A0h)																																Next Capability Pointer (68h)	Capability ID (05h)	48h		
																NT Port Link Interface PCI Express Capability Registers (Offsets 68h – A0h)																Next Capability Pointer (C8h)	Capability ID (10h)	...		
NT Port Link Interface PCI Express Capability Registers (Offsets 68h – A0h)																																Next Capability Pointer (C8h)	Capability ID (10h)	64h		
																NT Port Link Interface PCI Express Capability Registers (Offsets 68h – A0h)																Next Capability Pointer (C8h)	Capability ID (10h)	68h		
NT Port Link Interface PCI Express Capability Registers (Offsets 68h – A0h)																																Next Capability Pointer (C8h)	Capability ID (10h)	...		
																NT Port Link Interface PCI Express Capability Registers (Offsets 68h – A0h)																Next Capability Pointer (C8h)	Capability ID (10h)	A0h		
Reserved																																				A4h –
NT Port Link Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh)																Next Capability Pointer (00h)	Capability ID 3 (09h)	C4h																		
																NT Port Link Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh)																Next Capability Pointer (00h)	Capability ID 3 (09h)	...		
NT Port Link Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh)																																Next Capability Pointer (00h)	Capability ID 3 (09h)	FCh		
																Next Capability Offset (FB4h)																1h	PCI Express Extended Capability ID (0003h)	100h		
Device Serial Number Extended Capability Registers (Offsets 100h – 108h)																																				...
Device Serial Number Extended Capability Registers (Offsets 100h – 108h)																																				108h
Next Capability Offset (148h)																1h	PCI Express Extended Capability ID (0019h)	10Ch																		
Secondary PCI Express Extended Capability Registers (Offsets 10Ch – 134h)																																				...
Secondary PCI Express Extended Capability Registers (Offsets 10Ch – 134h)																																				134h
Next Capability Offset (10Ch)																1h	PCI Express Extended Capability ID (0004h)	138h																		
Power Budget Extended Capability Registers (Offsets 138h – 144h)																																				...
Power Budget Extended Capability Registers (Offsets 138h – 144h)																																				144h
Next Capability Offset (C34h)																1h	PCI Express Extended Capability ID (0002h)	148h																		
NT Port Link Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)																																				...
NT Port Link Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)																																				1BCh

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																1C0h
NT Port Link Interface Device-Specific Registers (Offsets 1C0h – A74h)																																...
A74h																																
<i>Reserved</i>																																A78h – B6Ch
Next Capability Offset 2 (000h)																1h								PCI Express Extended Capability ID 2 (000Bh)								B70h
NT Port Link Interface Device-Specific Registers (Offsets B70h – C88h)																																...
Next Capability Offset 4 (B70h)																1h								PCI Express Extended Capability ID 4 (000Bh)								C34h
NT Port Link Interface Device-Specific Registers (Offsets B70h – C88h)																																...
C88h																																
NT Bridging-Specific Registers (Offsets C8Ch – EFCh)																																C8Ch
...																																EFCh
NT Port Link Interface Device-Specific Registers (Offsets F00h – FB0h)																																F00h
...																																FB0h
Next Capability Offset (138h)																1h								PCI Express Extended Capability ID (0001h)								FB4h
NT Port Link Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)																																...
FDCh																																
<i>Reserved</i>																																FE0h – FFCh

## 16.3 Register Access

The PEX 8748 NT Port Link Interface implements a 4-KB Configuration Space. The lower 256 bytes (offsets 00h through FFh) comprise the PCI-compatible Configuration Space, and the upper 960 DWords (offsets 100h through FFFh) comprise the PCI Express Extended Configuration Space. The PEX 8748 supports three mechanisms for accessing the NT Port Link Interface registers:

- [PCI Express Base r3.0 Configuration Mechanism](#)
- [Device-Specific Memory-Mapped Configuration Mechanism](#)
- [Device-Specific Cursor Mechanism](#)

### 16.3.1 *PCI Express Base r3.0* Configuration Mechanism

The *PCI Express Base r3.0* Configuration mechanism is divided into two mechanisms:

- [PCI r3.0-Compatible Configuration Mechanism](#) – Provides Conventional PCI access to the first 256 bytes (the bytes at offsets 00h through FFh) of the NT Port Link Interface Configuration Register space
- [PCI Express Enhanced Configuration Access Mechanism](#) – Provides access to the entire 4-KB Configuration Space

Both are described in the sections that follow.

The PEX 8748 decodes Type 0 Configuration transactions received on its NT Port Link Interface. The PEX 8748 reads from or writes to the NT Port Link Interface register, as specified in the original Type 0 Configuration access.

#### 16.3.1.1 *PCI r3.0-Compatible* Configuration Mechanism

The *PCI r3.0-Compatible* Configuration Space consists of the first 256 bytes of the NT Port Link Interface Configuration Space. (Refer to [Figure 16-1](#).) The *PCI r3.0-Compatible* Configuration mechanism provides standard access to the PEX 8748 NT Port Link Interface's first 256 bytes (the bytes at offsets 00h through FFh) of the PCI Express Configuration Space.

This mechanism uses the same Request format as the [PCI Express Enhanced Configuration Access Mechanism](#). For PCI-compatible Configuration Requests, the *Extended Register Address* field must be all zeros (0).

Because the mechanism is limited to the first 256 bytes of the NT Port Link Interface Configuration Register space, one of the following must be used to access beyond Byte FFh:

- [PCI Express Enhanced Configuration Access Mechanism](#)
- [Device-Specific Memory-Mapped Configuration Mechanism](#)
- [Device-Specific Cursor Mechanism](#)

### 16.3.1.2 PCI Express Enhanced Configuration Access Mechanism

The PCI Express Enhanced Configuration Access mechanism (ECAM) uses a flat, Root Complex Memory-Mapped Address space to access the device Configuration registers. In this case, the Memory address determines the Configuration register accessed, and Memory data returns the addressed register contents. The Root Complex converts the Memory transaction into a Configuration transaction before transmitting this access to the Downstream devices.

This mechanism is used to access the NT Port Link Interface Type 0 registers:

- NT Port Link Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch)
- NT Port Link Interface PCI Power Management Capability Registers (Offsets 40h – 44h)
- NT Port Link Interface MSI Capability Registers (Offsets 48h – 64h)
- NT Port Link Interface PCI Express Capability Registers (Offsets 68h – A0h)
- NT Port Link Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh)
- Device Serial Number Extended Capability Registers (Offsets 100h – 108h)
- Power Budget Extended Capability Registers (Offsets 138h – 144h)
- NT Port Link Interface Virtual Channel Extended Capability Registers (Offsets 148h – 1BCh)
- Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets B70h – B7Ch)
- NT Port Link Interface Device-Specific Registers – Vendor-Specific Extended Capability 4 (Offsets C34h – C88h)
- NT Port Link Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)

This mechanism typically *cannot* access the Device-Specific registers (which are instead accessible by the Device-Specific Memory-Mapped Configuration Mechanism) in the following offset ranges, per Port:

- 200h through AFCh
- B0Ch through B6Ch
- B80h through C30h

Silicon Revision CA provides an option to enable ECAM access to all PEX 8748 registers, however, by Setting the **Station-Based Control** register *Device-Specific Register Access by Extended Configuration Request Enable* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 760h[16]), in the Station that has the NT Port.

## 16.3.2 Device-Specific Memory-Mapped Configuration Mechanism

The Device-Specific Memory-Mapped Configuration mechanism provides a method to access the registers for all Ports within a single 256-KB Memory map, as illustrated in [Figure 16-1](#). This Memory map is identical for Upstream Port(s) **BAR0/1**, NT Port Virtual Interface **BAR0/1**, and NT Port Link Interface **BAR0/1**. The registers of each Port are located within a 4-KB range.

When the NT Port is enabled at Fundamental Reset, the NT Port Virtual and Link Interface registers use the *PCI r3.0* Type 0 Configuration Space Header. The NT PCI-to-PCI bridge (between the NT Port Virtual Interface and internal virtual PCI Bus) registers use the *PCI r3.0* Type 1 Configuration Space Header, and are mapped to the 4-KB Address space of the Port Number that is assigned as the NT Port (indicated in the **VS0 Upstream** register *NT Port* field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset **360h**[**12:8**])).

To use this mechanism, use the [PCI r3.0-Compatible Configuration Mechanism](#) to program the PEX 8748 Upstream Port(s) **Base Address 0** and **Base Address 1** registers (**BAR0** and **BAR1**, offsets **10h** and **14h**, respectively). After the PEX 8748 NT Port Link Interface Memory-Mapped register Base address is Set, the PEX 8748 Configuration Space registers are accessed, using Memory Reads and Writes to the 4-KB range, starting at offset 248 KB (**3\_E000h**, Virtual Interface) and offset 252 KB (**3\_F000h**, Link Interface).

**Figure 16-1. NT Mode Configuration Register Mapping to Memory-Mapped BAR**

<b>PEX 8748</b>	
NT Link 4-KB Configuration Space (3_F000h)	<b>BAR0 + 256 KB</b>
NT Virtual 4-KB Configuration Space (3_E000h)	<b>BAR0 + 252 KB</b>
	<b>BAR0 + 248 KB</b>
<b>Reserved</b>	
Port 19 4-KB Configuration Space (1_3000h)	
Port 18 4-KB Configuration Space (1_2000h)	
Port 17 4-KB Configuration Space (1_1000h)	
Port 16 4-KB Configuration Space (1_0000h)	<b>BAR0 + 64 KB</b>
<b>Reserved</b>	
Port 11 4-KB Configuration Space (0_B000h)	
Port 10 4-KB Configuration Space (0_A000h)	
Port 9 4-KB Configuration Space (0_9000h)	
Port 8 4-KB Configuration Space (0_8000h)	
<b>Reserved</b>	
Port 3 4-KB Configuration Space (0_3000h)	
Port 2 4-KB Configuration Space (0_2000h)	
Port 1 4-KB Configuration Space (0_1000h)	
Port 0 4-KB Configuration Space (0_0000h)	<b>BAR0 + 0 KB</b>

**Note:** *Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP\_STNx\_PORTCFGx input states and/or serial EEPROM programming.*

### 16.3.3 Device-Specific Cursor Mechanism

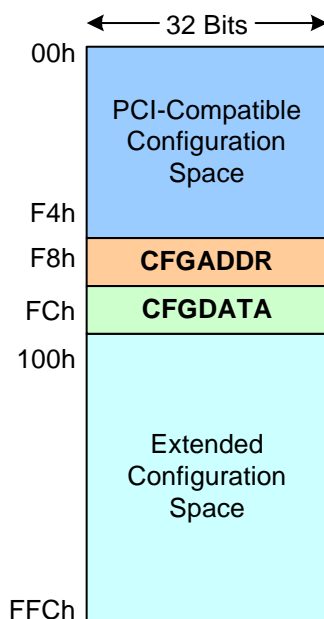
The Device-Specific Cursor mechanism is provided for use in development systems that can only generate *PCI r3.0* Configuration cycles (*that is*, the system cannot use either the [Device-Specific Memory-Mapped Configuration Mechanism](#), nor generate Extended Configuration Requests to access the Extended Configuration Space).

In [Figure 16-2](#), the software uses the **Configuration Address Window** (CFGADDR) register (offset **F8h**) to select the NT Port Link or Virtual Interface Configuration Space registers, including the PCI Express Extended Configuration Space registers (offsets 100h through FFFh).

Software uses the **Configuration Data Window** (CFGDATA) register (offset **FCh**) to write to or read from the selected Configuration Space registers.

Refer to [Section 16.9](#), “NT Port Link Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh),” for the register descriptions.

**Figure 16-2. Configuration Space View**





## 16.4 Register Descriptions

The remainder of this chapter details the PEX 8748 NT Port Link Interface registers, including:

- Bit/field names
- Description of register functions in the PEX 8748 NT Port Link and Virtual Interfaces
- Type (*such as* RW or HwInit; refer to [Table 13-4, “Register Types, Grouped by User Accessibility,”](#) for Type descriptions)
- Whether the power-on/reset value can be modified, by way of the PEX 8748 serial EEPROM and/or I<sup>2</sup>C/SMBus Initialization feature
- Default power-on/reset value

## 16.5 NT Port Link Interface PCI-Compatible Type 0 Configuration Header Registers (Offsets 00h – 3Ch)

This section details the NT Port Link Interface PCI-Compatible Type 0 Configuration Header registers. [Table 16-2](#) defines the register map.

**Table 16-2. NT Port Link Interface PCI-Compatible Type 0 Configuration Header Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																
Device ID																Vendor ID																00h
PCI Status																PCI Command																04h
PCI Class Code																PCI Revision ID																08h
PCI BIST <i>(Not Supported)</i>				PCI Header Type												Master Latency Timer <i>(Not Supported)</i>								Cache Line Size								0Ch
Base Address 0																																10h
Base Address 1																																14h
Base Address 2																																18h
Base Address 3																																1Ch
Base Address 4																																20h
Base Address 5																																24h
<i>Reserved</i>																																28h
Subsystem ID																Subsystem Vendor ID																2Ch
Expansion ROM Base Address																																30h
<i>Reserved</i>																								Capability Pointer (40h)								34h
<i>Reserved</i>																																38h
Max_Lat <i>(Reserved)</i>								Min_Gnt <i>(Reserved)</i>								PCI Interrupt Pin								PCI Interrupt Line								3Ch

### Register 16-1. 00h PCI Configuration ID

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Vendor ID</b> Identifies the device manufacturer. Defaults to the PCI-SIG-issued Vendor ID of PLX (10B5h), if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	10B5h
31:16	<b>Device ID</b> Identifies the particular device. Defaults to the PLX part number for the PEX 8748, if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	87A0h

**Register 16-2. 04h PCI Command/Status**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Command</b>				
0	<b>I/O Access Enable</b> The NT Port Link Interface ignores the value of this register, because it does <i>not support</i> I/O resources.	RW	Yes	0
1	<b>Memory Access Enable</b> 0 = PEX 8748 ignores Memory Space Requests received on the NT Port Link Interface 1 = PEX 8748 accepts Memory Space Requests received on the NT Port Link Interface	RW	Yes	0
2	<b>Bus Master Enable</b> Controls PEX 8748 forwarding of Memory Requests Upstream. Does not affect Message forwarding nor Completions. 0 = PEX 8748 handles Memory Requests received on the NT Port Virtual Interface as Unsupported Requests (UR); for Non-Posted Requests, the PEX 8748 returns a Completion with UR Completion status 1 = PEX 8748 forwards Memory Requests from the NT Port Virtual Interface to the NT Port Link Interface	RW	Yes	0
3	<b>Special Cycle Enable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
4	<b>Memory Write and Invalidate Enable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
5	<b>VGA Palette Snoop</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
6	<b>Parity Error Response Enable</b> Controls bit 24 ( <i>Master Data Parity Error Detected</i> ).	RW	Yes	0
7	<b>IDSEL Stepping/Wait Cycle Control</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
8	<b>SERR# Enable</b> Controls bit 30 ( <i>Signaled System Error</i> ). 1 = Enables reporting of Fatal and Non-Fatal errors detected by the NT Port Link Interface to the Root Complex	RW	Yes	0
9	<b>Fast Back-to-Back Transactions Enable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
10	<b>Interrupt Disable</b> 0 = NT Port Link Interface is enabled to generate INTx Interrupt Messages 1 = NT Port Link Interface is prevented from generating INTx Interrupt Messages	RW	Yes	0
15:11	<b>Reserved</b>	RsvdP	No	0-0h

NDA CONFIDENTIAL -- Inventec | King Michael

**Register 16-2. 04h PCI Command/Status (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Status</b>				
18:16	<i>Reserved</i>	RsvdP	No	000b
19	<b>Interrupt Status</b> 0 = No INTx interrupt is pending 1 = INTx interrupt is pending internally to the NT Port Link Interface, –or– PEX_INTA# (Base mode) or VSx_PEX_INTA# (Virtual Switch mode) (if enabled) is asserted	RO	No	0
20	<b>Capability List</b> Capability function is supported. Set, as required by the <i>PCI Express Base r3.0</i> .	RO	Yes	1
21	<b>66 MHz Capable</b> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
22	<i>Reserved</i>	RsvdP	No	0
23	<b>Fast Back-to-Back Transactions Capable</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
24	<b>Master Data Parity Error Detected</b> If bit 6 ( <i>Parity Error Response Enable</i> ) is Set, the NT Port Link Interface Sets this bit when the NT Port: <ul style="list-style-type: none"> <li>Forwards the poisoned Transaction Layer Packet (TLP) Write Request from the NT Port Virtual Interface to the NT Port Link Interface, –or–</li> <li>Receives a Completion marked as poisoned on the NT Port Link Interface</li> </ul> If the <i>Parity Error Response Enable</i> bit is Cleared, the PEX 8748 never Sets this bit. This error is natively reported by the <b>Uncorrectable Error Status</b> register <i>Poisoned TLP Status</i> bit (offset FB8h[12]), which is mapped to this bit for Conventional PCI backward compatibility.	RW1C	Yes	0
26:25	<b>DEVSEL# Timing</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	00b
27	<b>Signaled Target Abort</b> The NT Port Link Interface Sets this bit when any of the following conditions are met: <ul style="list-style-type: none"> <li>NT Port Link Interface receives a Completion (from a Transparent Port) that has a Completion status of Completer Abort (CA), –or–</li> <li>NT Port Link Interface receives a Memory Request targeting a PEX 8748 register, and the Payload Length (indicated within the Memory Request Header) is greater than 1 DWord</li> <li>NT Port Link Interface receives a Memory Request targeting a PEX 8748 register address within a non-existent Port</li> <li>NT Port Link Interface receives a Memory Write Request targeting enabled NT Port Expansion ROM Address space (Expansion ROM Base Address register (BAR), offset 30h)</li> </ul> <i>Note: When Set during a forwarded Completion, the <b>Uncorrectable Error Status</b> register <i>Completer Abort Status</i> bit (offset FB8h[15]) is not updated, because the NT Port does not log the corresponding Requests that it forwards.</i>	RW1C	Yes	0

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

**Register 16-2. 04h PCI Command/Status (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
28	<b>Received Target Abort</b> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
29	<b>Received Master Abort</b> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	0
30	<b>Signaled System Error</b> If bit 8 ( <i>SERR# Enable</i> ) is Set, the NT Port Link Interface Sets this bit when transmitting an ERR_FATAL or ERR_NONFATAL Message to its Upstream device.  This error is natively reported by the <b>Device Status</b> register <i>Fatal Error Detected</i> and <i>Non-Fatal Error Detected</i> bits (offset 70h[18:17], respectively), which are mapped to this bit for Conventional PCI backward compatibility.	RW1C	Yes	0
31	<b>Detected Parity Error</b> This error is natively reported by the <b>Uncorrectable Error Status</b> register <i>Poisoned TLP Status</i> bit (offset FB8h[12]), which is mapped to this bit for Conventional PCI backward compatibility.  1 = NT Port Link Interface received a Poisoned TLP, regardless of the bit 6 ( <i>Parity Error Response Enable</i> ) state	RW1C	Yes	0

**Register 16-3. 08h PCI Class Code and Revision ID**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Revision ID</b>				
7:0	<b>Revision ID</b> Unless overwritten by the serial EEPROM, returns the Silicon Revision (BAh), the PLX-assigned Revision ID for this version of the PEX 8748. The PEX 8748 Serial EEPROM register Initialization capability is used to replace the PLX Revision ID with another Revision ID.	RO	Yes	BAh
<b>PCI Class Code</b>				<b>068000h</b>
15:8	<b>Register-Level Programming Interface</b> Cleared, as required by the <i>PCI r3.0</i> for other bridge devices.	RO	Yes	00h
23:16	<b>Sub-Class Code</b> Other bridge devices.	RO	Yes	80h
31:24	<b>Base Class Code</b> Bridge devices.	RO	Yes	06h

**Register 16-4. 0Ch Miscellaneous Control**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Cache Line Size</b>				
7:0	<b>Cache Line Size</b> System Cache Line Size. Implemented as a RW field for Conventional PCI compatibility purposes and does not impact PEX 8748 functionality.	RW	Yes	00h
<b>Master Latency Timer</b>				
15:8	<b>Master Latency Timer</b> <i>Not supported</i> Cleared, as required by the <i>PCI Express Base r3.0</i> .	RsvdP	No	00h
<b>PCI Header Type</b>				
22:16	<b>Configuration Layout Type</b> Type 0 Configuration Header for the NT Port.	RO	Yes	00h
23	<b>Multi-Function Device</b> 0 = Single-function device 1 = Indicates multiple (up to eight) functions (logical devices), each containing its own, individually addressable Configuration Space, 256 DWords in size	RO	Yes	0
<b>PCI BIST</b>				
31:24	<b>PCI BIST</b> <i>Not supported</i> Built-In Self-Test (BIST) Pass or Fail.	RsvdP	No	00h

**Register 16-5. 10h Base Address 0  
(NT Port Link Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<p><b>Note:</b> By default, NT Port Link Interface <b>BAR0</b> is enabled and <b>BAR1</b> is disabled, to provide a 32-bit <b>BAR0</b> for register access. <b>BAR1</b> can be enabled (by serial EEPROM and/or I<sup>2</sup>C/SMBus), to provide a 64-bit <b>BAR0/1</b>, by programming the <b>NT Port Link Interface BAR0/1 Setup</b> register <b>BAR0/1 Enable</b> field (NT Port Link Interface, offset <b>E4h[1:0]</b>) to 11b (which enables both <b>BAR0</b> and <b>BAR1</b>).</p>				
0	<b>Memory Space Indicator</b> When enabled, the Base Address register maps PEX 8748 Port Configuration registers into Memory space. <b>Note:</b> Hardwired to 0.	RO	No	0
2:1	<b>Memory Map Type</b> 00b = Base Address register is 32 bits wide and can be mapped anywhere in the 32-bit Memory space 10b = Base Address register is 64 bits wide and can be mapped anywhere in the 64-bit Address space All other encodings are <i>Reserved</i> .	ROS	Yes	00b
3	<b>Prefetchable</b> 0 = Base Address register maps the PEX 8748 Port Configuration registers into Non-Prefetchable Memory space	ROS	Yes	0
17:4	<i>Reserved</i>	RsvdP	No	0-0h
31:18	<b>Base Address 0</b> 256-KB-aligned Base address used for Memory-Mapped access to the 256-KB block of all PEX 8748 registers (4 KB, per Port).	RW	Yes	0-0h

**Register 16-6. 14h Base Address 1  
(NT Port Link Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Upper 32-Bit Address for Memory-Mapped BAR</b> For 64-bit addressing ( <b>BAR0/1</b> ), Base Address 1 ( <b>BAR1</b> ) extends Base Address 0 ( <b>BAR0</b> ) to provide the upper 32 Address bits when the <b>Base Address 0</b> register <i>Memory Map Type</i> field (offset 10h[2:1]) is programmed to 10b.	RW	Yes	0000_0000h
	Read-Only when the <b>Base Address 0</b> ( <b>BAR0</b> ) register is not enabled as a 64-bit BAR ( <i>Memory Map Type</i> field (offset 10h[2:1]) is not programmed to 10b).	RO	Yes	0000_0000h

**Register 16-7. 18h Base Address 2  
(NT Port Link Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Implemented as a Memory BAR	RO	No	0
2:1	<b>Memory Map Type</b> 00b = Base Address register is 32 bits wide and can be mapped anywhere in the 32-bit Memory space 10b = Base Address register is 64 bits wide and can be mapped anywhere in the 64-bit Address space All other encodings are <i>Reserved</i> .	ROS	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	ROS	Yes	0
19:4	<i>Reserved</i>	RsvdP	No	0_000h
31:20	<b>Base Address 2</b> Resolution is 1 MB.	RW	Yes	000h

**Register 16-8. 1Ch Base Address 3  
(NT Port Link Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
<i><b>Note:</b> This register has RW privilege if <b>BAR2/3</b> is configured as a 64-bit BAR (<b>Base Address 2</b> register <b>Memory Map Type</b> field (NT Port Link Interface, offset 18h[2:1]), is programmed to 10b).</i>					
0	<b>Memory Space Indicator</b> <b>BAR3</b> can be used as an independent 32-bit only BAR, or as the upper 32 bits of 64-bit <b>BAR2/3</b> . 0 = Implemented as a Memory BAR in 32-Bit mode	Offset 18h[2:1]=00b	RO	No	0
		Offset 18h[2:1]=10b	RW	Yes	0
2:1	<b>Memory Map Type</b> 00b = Base Address register is 32 bits wide and can be mapped anywhere in the 32-bit Memory space All other encodings are <i>Reserved</i> .	Offset 18h[2:1]=00b	RO	No	00b
		Offset 18h[2:1]=10b	RW	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	Offset 18h[2:1]=00b	ROS	No	0
		Offset 18h[2:1]=10b	RW	Yes	0
19:4	<i>Reserved</i>	Offset 18h[2:1]=00b	RsvdP	No	0_000h
	When <b>BAR2/3</b> are used as a 64-bit BAR, bits [31:0] (including bits [19:4]) are used as the upper 32 bits.	Offset 18h[2:1]=10b	RW	Yes	0_000h
31:20	<b>Base Address 3</b>	RW	Yes	000h	



**Register 16-9. 20h Base Address 4  
(NT Port Link Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Memory Space Indicator</b> 0 = Implemented as a Memory BAR; otherwise, <i>Reserved</i>	RO	No	0
2:1	<b>Memory Map Type</b> 00b = Base Address register is 32 bits wide and can be mapped anywhere in the 32-bit Memory space 10b = Base Address register is 64 bits wide and can be mapped anywhere in the 64-bit Address space All other encodings are <i>Reserved</i> .	ROS	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	ROS	Yes	0
19:4	<i>Reserved</i>	RsvdP	No	0_000h
31:20	<b>Base Address 4</b>	RW	Yes	000h

**Register 16-10. 24h Base Address 5  
(NT Port Link Interface Memory Space)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
<i><b>Note:</b> This register has RW privilege if <b>BAR4/5</b> is configured as a 64-bit BAR (<b>Base Address 4</b> register <b>Memory Map Type</b> field (NT Port Link Interface, offset 20h[2:1]), is programmed to 10b).</i>					
0	<b>Memory Space Indicator</b> <b>BAR5</b> can be used as an independent 32-bit only BAR, or as the upper 32 bits of 64-bit <b>BAR4/5</b> . 0 = Implemented as a Memory BAR in 32-Bit mode	Offset 20h[2:1]=00b	RO	No	0
		Offset 20h[2:1]=10b	RW	Yes	0
2:1	<b>Memory Map Type</b> 00b = Base Address register is 32 bits wide and can be mapped anywhere in the 32-bit Memory space All other encodings are <i>Reserved</i> .	Offset 20h[2:1]=00b	RO	No	00b
		Offset 20h[2:1]=10b	RW	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	Offset 20h[2:1]=00b	ROS	No	0
		Offset 20h[2:1]=10b	RW	Yes	0
19:4	<i>Reserved</i>	Offset 20h[2:1]=00b	RsvdP	No	0_000h
	When <b>BAR4/5</b> are used as a 64-bit BAR, bits [31:0] (including bits [19:4]) are used as the upper 32 bits.	Offset 20h[2:1]=10b	RW	Yes	0_000h
31:20	<b>Base Address 5</b>	RW	Yes	000h	

**Register 16-11. 2Ch Subsystem ID and Subsystem Vendor ID**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Subsystem Vendor ID</b>				
15:0	<b>Subsystem Vendor ID</b> Identifies the device manufacturer. Defaults to the PCI-SIG-issued Vendor ID of PLX (10B5h), if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	10B5h
<b>Subsystem ID</b>				
31:16	<b>Subsystem ID</b> Identifies the particular device. Defaults to the PLX part number for the PEX 8748, if not overwritten by serial EEPROM and/or I <sup>2</sup> C.	RO	Yes	87A0h

**Register 16-12. 30h Expansion ROM Base Address**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
<b>Note:</b> Expansion ROM can be enabled in either the NT Port Link or Virtual Interface, but not both simultaneously. Expansion ROM is enabled, by default, in the NT Port Link Interface ( <b>Ingress Chip Control</b> register <i>Expansion ROM Virtual Side</i> bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 764h[0]) is Cleared). Expansion ROM can be disabled, by Setting the <b>Ingress Control</b> register <i>Disable NT Port Expansion ROM BAR</i> bit (Base mode – All Ports; Virtual Switch mode – VS Upstream Port(s), offset F60h[15]).					
0	<b>Expansion ROM Enable</b> 0 = NT Port Link Interface Expansion ROM is disabled	Upstream Port(s), offset F60h[15]=1	RsvdP	No	0
	1 = NT Port Link Interface Expansion ROM is enabled, and NT Port Virtual Interface Expansion ROM is disabled	Upstream Port(s), offset F60h[15]=0	RO	Yes	0
13:1	<b>Reserved</b>		RsvdP	No	0-0h
31:14	<b>Expansion ROM Base Address</b> If the <b>Serial EEPROM Clock Frequency</b> register <i>Expansion ROM Size</i> bit (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 268h[16]) value is 0, the NT Port Expansion ROM size is 16 KB (default value is FFFF_C001h). Bit 14 is RW.  If the <i>Expansion ROM Size</i> bit value is 1, the NT Port Expansion ROM size is 32 KB (default value is FFFF_8001h). Bit 14 is RO.	Upstream Port(s), offset F60h[15]=1	RsvdP	No	0-0h
		Upstream Port(s), offset F60h[15]=0	RW	Yes	0-0h

**Register 16-13. 34h Capability Pointer**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Capability Pointer</b> Default 40h points to the <b>PCI Power Management Capability</b> structure.	RO	Yes	40h
31:8	<b>Reserved</b>	RsvdP	No	0000_00h

**Register 16-14. 3Ch PCI Interrupt**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>PCI Interrupt Line</b> The Interrupt Line Routing Value communicates interrupt line routing information. Values in this register are programmed by system software and are system architecture-specific. The value is used by device drivers and operating systems.	RW	Yes	00h
15:8	<b>PCI Interrupt Pin</b> Identifies the Conventional PCI Interrupt Message(s) the device (or device function) uses. Only value 00h or 01h is allowed in the PEX 8748. 00h = Indicates that the device does not use Conventional PCI Interrupt Message(s) 01h, 02h, 03h, and 04h = Maps to Conventional PCI Interrupt Messages for INTA#, INTB#, INTC#, and INTD#, respectively	RO	Yes	01h
23:16	<b>Min_Gnt</b> Minimum Grant. <b>Reserved</b> Does not apply to PCI Express.	RsvdP	No	00h
31:24	<b>Max_Lat</b> Maximum Latency. <b>Reserved</b> Does not apply to PCI Express.	RsvdP	No	00h

## 16.6 NT Port Link Interface PCI Power Management Capability Registers (Offsets 40h – 44h)

This section details the NT Port Link Interface PCI Power Management Capability registers. [Table 16-3](#) defines the register map.

**Table 16-3. NT Port Link Interface PCI Power Management Capability Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																
PCI Power Management Capability																Next Capability Pointer (48h)								Capability ID (01h)								40h
PCI Power Management Data												PCI Power Management Control/Status Bridge Extensions ( <i>Reserved</i> )				PCI Power Management Status and Control																44h

### Register 16-15. 40h PCI Power Management Capability

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Capability ID</b> Default = 01h – only value allowed.	RO	Yes	01h
15:8	<b>Next Capability Pointer</b> Default 48h points to the <b>MSI Capability</b> structure.	RO	Yes	48h
18:16	<b>Version</b> Default = 011b – only value allowed.	RO	Yes	011b
19	<b>PME Clock</b> Power Management Event (PME) clock. Does not apply to PCI Express. Returns a value of 0.	RsvdP	No	0
20	<b>Reserved</b>	RsvdP	No	0
21	<b>Device-Specific Initialization</b> 0 = Device-Specific Initialization is <i>not</i> required	RO	Yes	0
24:22	<b>AUX Current</b> The PEX 8748 does <i>not support</i> PME generation from the D3cold Device Power Management (PM) state; therefore, the serial EEPROM value for this field should be 000b.	RO	Yes	000b
25	<b>D1 Support</b> <i>Not supported</i> 0 = PEX 8748 does <i>not support</i> the D1 Device PM state	RsvdP	No	0
26	<b>D2 Support</b> <i>Not supported</i> 0 = PEX 8748 does <i>not support</i> the D2 Device PM state	RsvdP	No	0
31:27	<b>PME Support</b> The default value is applied to bits [31, 30, and 27] only. PME Messages are disabled, by default.	RO	Yes	0000_0b

**Register 16-16. 44h PCI Power Management Status and Control**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Power Management Status and Control</b>				
1:0	<b>Power State</b> Used to determine the Port's current Device PM state, and to Set the Port into a new Device PM state.  00b = D0 01b = D1 – <i>Not supported</i> 10b = D2 – <i>Not supported</i> 11b = D3hot  If software attempts to write an unsupported state to this field, the Write operation completes normally; however, the data is discarded and no state change occurs.	RW	Yes	00b
2	<b>Reserved</b>	RsvdP	No	0
3	<b>No Soft Reset</b> 1 = Devices transitioning from the D3hot to D0 state, because of Power State commands, do not perform an internal reset	ROS	Yes	1
7:4	<b>Reserved</b>	RsvdP	No	0h
8	<b>PME Enable</b> Default value of 0 indicates that PME generation is disabled.	RsvdP	No	0
12:9	<b>Data Select</b> Initially writable by serial EEPROM and/or I <sup>2</sup> C only <sup>a</sup> . This Configuration Space register (CSR) access privilege changes to RW after a Serial EEPROM and/or I <sup>2</sup> C Write occurs to this register. Selects the field [14:13] ( <i>Data Scale</i> ) and <b>PCI Power Management Data</b> register <i>Data</i> field (offset 44h[31:24]).  0h = D0 power consumed 3h = D3hot power consumed 4h = D0 power dissipated 7h = D3hot power dissipated  All other encodings are <b>Reserved</b> .	RO	Yes	0h
14:13	<b>Data Scale</b> Writable by serial EEPROM and/or I <sup>2</sup> C only <sup>a</sup> . Indicates the scaling factor to be used when interpreting the <b>PCI Power Management Data</b> register <i>Data</i> field (offset 44h[31:24]) value. The value and meaning of the <i>Data Scale</i> field varies, depending upon which data value is selected by field [12:9] ( <i>Data Select</i> ). There are four internal <i>Data Scale</i> fields (one each, per <i>Data Select</i> values 0h, 3h, 4h, and 7h). For other <i>Data Select</i> values, the <i>Data Scale</i> value returned is 0h.	RO	Yes	00b
15	<b>PME Status</b> 0 = PME is not being generated by the NT Port	RsvdP	No	0

**Register 16-16. 44h PCI Power Management Status and Control (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Power Management Control/Status Bridge Extensions</b>				
21:16	<i>Reserved</i>	RsvdP	No	0-0h
22	<b>B2/B3 Support</b> <i>Reserved</i> Cleared, as required by the <i>PCI Power Mgmt. r1.2</i> .	RsvdP	No	0
23	<b>Bus Power/Clock Control Enable</b> <i>Reserved</i> Cleared, as required by the <i>PCI Power Mgmt. r1.2</i> .	RsvdP	No	0
<b>PCI Power Management Data</b>				
31:24	<b>Data</b> Writable by serial EEPROM and/or I <sup>2</sup> C only <sup>a</sup> . There are four supported <i>Data Select</i> values (0h, 3h, 4h, and 7h). For other <i>Data Select</i> values, the <i>Data</i> value returned is 00h. Selected by the <b>PCI Power Management Status and Control</b> register <i>Data Select</i> field (offset 44h[12:9]).	RO	Yes	00h

- a. With no serial EEPROM nor previous I<sup>2</sup>C programming, Reads return a value of 00h for the **PCI Power Management Status and Control** register *Data Scale* and **PCI Power Management Data** register *Data* fields (for all *Data Selects*).



**Register 16-17. 48h MSI Capability**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>MSI Capability Header</b>				
7:0	<b>Capability ID</b> Program to 05h, as required by the <i>PCI r3.0</i> .	RO	Yes	05h
15:8	<b>Next Capability Pointer</b> Program to 68h, to point to the <b>PCI Express Capability</b> structure.	RO	Yes	68h
<b>MSI Control</b>				
16	<b>MSI Enable</b> 0 = MSIs for the NT Port Link Interface are disabled 1 = MSIs for the NT Port Link Interface are enabled, and INTx Interrupt Messages and <b>PEX_INTA#</b> (Base mode) or <b>VSx_PEX_INTA#</b> (Virtual Switch mode) output assertion are disabled	RW	Yes	0
19:17	<b>Multiple Message Capable</b> 000b = NT Port Link Interface can request only one MSI Vector 001b = NT Port Link Interface can request two MSI Vectors 010b = NT Port Link Interface can request four MSI Vectors All other encodings are <i>Reserved</i> .	RO	Yes	010b
22:20	<b>Multiple Message Enable</b> 000b = NT Port Link Interface is allocated one MSI Vector, by default 001b = NT Port Link Interface is allocated two MSI Vectors 010b = NT Port Link Interface is allocated four MSI Vectors All other encodings are <i>Reserved</i> . <i>Note: This field should not be programmed with a value larger than that of field [19:17] (Multiple Message Capable). If the value of this field is larger than that of field [19:17], the Multiple Message Capable value takes effect.</i>	RW	Yes	000b
23	<b>MSI 64-Bit Address Capable</b> 0 = PEX 8748 is capable of generating MSI 32-bit addresses ( <b>MSI Address</b> register, offset 4Ch, is the Message address) 1 = PEX 8748 is capable of generating MSI 64-bit addresses ( <b>MSI Address</b> register, offset 4Ch, is the lower 32 bits of the Message address, and <b>MSI Upper Address</b> register, offset 50h, is the upper 32 bits of the Message address)	RO	Yes	1
24	<b>Per Vector Masking Capable</b> 0 = PEX 8748 does not have Per Vector Masking capability 1 = PEX 8748 has Per Vector Masking capability	RO	Yes	1
31:25	<i>Reserved</i>	RsvdP	No	0-0h



**Register 16-18. 58h MSI Mask**

Bit(s)	Description	Type	Serial EEPROM and I²C	Default	
<p>NT Port Link Interface interrupt sources are grouped into two categories – Power Management/Link State events and NT-Link Doorbell-generated interrupts.</p> <p>The quantity of MSI Vectors that are generated is determined by the <b>MSI Control</b> register <i>Multiple Message Capable</i> and <i>Multiple Message Enable</i> fields (NT Port Link Interface, offset 48h[19:17 and 22:20], respectively):</p> <ul style="list-style-type: none"> <li>• If <b>one</b> MSI Vector is enabled (default), both interrupt events are combined into a single Vector</li> <li>• If <b>two</b> MSI Vectors are allocated, the individual Vectors indicate: <ul style="list-style-type: none"> <li>– Vector[0] Power Management/Link State events</li> <li>– Vector[1] NT-Link Doorbell-generated interrupts</li> </ul> </li> </ul> <p><b>Notes:</b> The offset for this register changes from 58h, to 54h, when the <b>MSI Control</b> register <i>MSI 64-Bit Address Capable</i> bit (offset 48h[23]) is Cleared.</p> <p>The bits in this register can be used to mask their respective <b>MSI Status</b> register bits (offset 5Ch).</p>					
0	<b>MSI Mask for Link State Events</b> MSI mask for Power Management event- or Link State event-generated interrupts.	Offset 48h[22:20]=001b	RW	Yes	0
	<b>MSI Mask for Shared Interrupt Sources</b> MSI mask for all interrupt sources when the <b>MSI Control</b> register <i>Multiple Message Enable</i> field indicates that the Host has allocated one or two Vectors.	Offset 48h[22:20]=000b	RW	Yes	0
2:1	<b>Reserved</b>		RsvdP	No	00b
3	<b>MSI Mask for NT-Link Doorbell-Generated Interrupts</b> Refer to NT Port registers located at offsets C5Ch through C68h.	Offset 48h[22:20]=001b	RW	Yes	0
	<b>Reserved</b>	Offset 48h[22:20]=000b	RsvdP	No	0
31:4	<b>Reserved</b>		RsvdP	No	0000_000h

**Register 16-19. 5Ch MSI Status**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
<p>NT Port Link Interface interrupt sources are grouped into two categories – Power Management/Link State events and NT-Link Doorbell-generated interrupts.</p> <p>The quantity of MSI Vectors that are generated is determined by the <b>MSI Control</b> register <i>Multiple Message Capable</i> and <i>Multiple Message Enable</i> fields (NT Port Link Interface, offset 48h[19:17 and 22:20], respectively):</p> <ul style="list-style-type: none"> <li>If <b>one</b> MSI Vector is enabled (default), both interrupt events are combined into a single Vector</li> <li>If <b>two</b> MSI Vectors are allocated, the individual Vectors indicate: <ul style="list-style-type: none"> <li>Vector[0] Power Management/Link State events</li> <li>Vector[1] NT-Link Doorbell-generated interrupts</li> </ul> </li> </ul> <p><b>Notes:</b> The offset for this register changes from 5Ch, to 58h, when the <b>MSI Control</b> register <i>MSI 64-Bit Address Capable</i> bit (offset 48h[23]) is Cleared.</p> <p>The bits in this register can be masked by their respective <b>MSI Mask</b> register bits (offset 58h).</p>					
0	<b>MSI Pending Status for Link State Events</b> MSI pending status for Power Management event- or Link State event-generated interrupts.	Offset 48h[22:20]=001b	RO	No	0
	<b>MSI Pending Status for Shared Interrupt Sources</b> MSI pending status for all interrupt sources when the <b>MSI Control</b> register <i>Multiple Message Enable</i> field indicates that the Host has allocated one or two Vectors.	Offset 48h[22:20]=000b	RO	No	0
2:1	<b>Reserved</b>		RsvdP	No	00b
3	<b>MSI Pending Status for NT-Link Doorbell-Generated Interrupts</b> Refer to NT Port registers located at offsets C5Ch through C68h.	Offset 48h[22:20]=001b	RO	No	0
	<b>Reserved</b>	Offset 48h[22:20]=000b	RsvdP	No	0
31:4	<b>Reserved</b>		RsvdP	No	0000_000h

## 16.8 NT Port Link Interface PCI Express Capability Registers (Offsets 68h – A0h)

The registers detailed in [Section 13.10, “PCI Express Capability Registers \(Offsets 68h – A0h\),”](#) are also applicable to the NT Port Link Interface, except as defined in [Table 16-5](#) (register map; offsets 7Ch and 80h are **Reserved**), and [Register 16-20](#) through [Register 16-27](#).

**Table 16-5. NT Port Link Interface PCI Express Capability Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																											
PCI Express Capability																Next Capability Pointer (C8h)												Capability ID (10h)												68h			
Device Capability																																				6Ch							
Device Status																<i>Not Supported/Reserved</i>												Device Control												70h			
Link Capability																																				74h							
Link Status																<i>Reserved</i>												Link Control												78h			
<i>Reserved</i>																																				7Ch – 88h							
Device Capability 2																																				8Ch							
Device Status 2 ( <i>Reserved</i> )																Device Control 2																				90h							
Link Capability 2																																				94h							
Link Status 2																Link Control 2																				98h							
<i>Reserved</i>																																				9Ch – A0h							

**Register 16-20. 68h PCI Express Capability List and Capability**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>PCI Express Capability List</b>				
7:0	<b>Capability ID</b> Program to 10h, by default.	RO	Yes	10h
15:8	<b>Next Capability Pointer</b> Program to C8h, to point to the <b>Vendor-Specific Capability 3</b> structure.	RO	Yes	C8h
<b>PCI Express Capability</b>				
19:16	<b>Capability Version</b> The PEX 8748 NT Port Link Interface programs this field to 2h, as required by the <i>PCI Express Base r3.0</i> .	RO	Yes	2h
23:20	<b>Device/Port Type</b> Default = PCI Express endpoint device.	RO	No	0h
24	<b>Slot Implemented</b> <i>Not valid for PCI Express endpoint devices</i>	RsvdP	No	0
29:25	<b>Interrupt Message Number</b> The serial EEPROM writes 00_000b, because the Base Message and MSI Messages are the same.	RO	Yes	00_000b
31:30	<b>Reserved</b>	RsvdP	No	00b

**Register 16-21. 6Ch Device Capability**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
2:0	<b>Maximum Payload Size Supported</b>  <i>Note: The default Maximum Payload Size supported is based upon the STRAP_STNx_PORTCFGx 3-state inputs. If the serial EEPROM or I<sup>2</sup>C/SMBus changes the Port configuration (from the strapped value), it must also program the value of this field, accordingly (with the value dependent upon the Station having the most enabled Ports).</i>  The value for all Ports is determined by the Station having the most enabled Ports: <ul style="list-style-type: none"> <li>One or two enabled Ports = 100b</li> <li>Three enabled Ports = 011b</li> <li>Four enabled Ports = 010b</li> </ul> 000b = NT Port Link Interface supports a 128-byte maximum payload 001b = NT Port Link Interface supports a 256-byte maximum payload 010b = NT Port Link Interface supports a 512-byte maximum payload 011b = NT Port Link Interface supports a 1,024-byte maximum payload 100b = NT Port Link Interface supports a 2,048-byte maximum payload  No other encodings are supported.	HwInit	Yes	Defined by STRAP_STNx_PORTCFGx input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> register <i>Port Configuration for Station x</i> field(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 300h[8:6, 5:3, and/or 2:0])
4:3	<b>Phantom Functions Supported</b> <i>Not supported</i>	RO	Yes	00b
5	<b>Extended Tag Field Supported</b> 0 = Maximum <i>Tag</i> field is 5 bits 1 = Maximum <i>Tag</i> field is 8 bits	RO	Yes	0
8:6	<b>Endpoint L0s Acceptable Latency</b> 111b = No Limit	RO	Yes	111b
11:9	<b>Endpoint L1 Acceptable Latency</b> 111b = No Limit	RO	Yes	111b
14:12	<b>Reserved</b>	RsvdP	No	000b
15	<b>Role-Based Error Reporting</b>	RO	Yes	1

**Register 16-21. 6Ch Device Capability (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
17:16	<i>Reserved</i>	RsvdP	No	00b
25:18	<b>Captured Slot Power Limit Value</b> For the NT Port Link Interface, the upper limit on power supplied by the slot is determined by multiplying the value in this field by the value in field [27:26] ( <i>Captured Slot Power Limit Scale</i> ).	RO	Yes	00h
27:26	<b>Captured Slot Power Limit Scale</b> For the NT Port Link Interface, the upper limit on power supplied by the slot is determined by multiplying the value in this field by the value in field [25:18] ( <i>Captured Slot Power Limit Value</i> ).  00b = 1.0 01b = 0.1 10b = 0.01 11b = 0.001	RO	Yes	00b
31:28	<i>Reserved</i>	RsvdP	No	0h

**Register 16-22. 70h Device Status and Control**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Control</b>				
0	<b>Correctable Error Reporting Enable</b> 0 = Disables 1 = Enables the NT Port Link Interface to report Correctable errors to the Link side Host	RW	Yes	0
1	<b>Non-Fatal Error Reporting Enable</b> 0 = Disables 1 = Enables the NT Port Link Interface to report Non-Fatal errors to the Link side Host	RW	Yes	0
2	<b>Fatal Error Reporting Enable</b> 0 = Disables 1 = Enables the NT Port Link Interface to report Fatal errors to the Link side Host	RW	Yes	0
3	<b>Unsupported Request Reporting Enable</b> 0 = Disables 1 = Enables the NT Port Link Interface to report UR errors as Error Messages with a programmed uncorrectable error severity	RW	Yes	0
4	<b>Enable Relaxed Ordering</b> <i>Not supported</i>	RW	Yes	1
7:5	<b>Maximum Payload Size</b> The NT Port Link Interface power-on/reset value is 000b, to support a Maximum Payload Size of 128 bytes. Software can change this field to configure the NT Port Link Interface to support other Payload sizes; however, software cannot change this field to a value larger than that indicated by the <b>Device Capability</b> register <i>Maximum Payload Size Supported</i> field (offset 6Ch[2:0]), for the NT Port Virtual and Link Interfaces. (Requester and Completer domains must possess the same Maximum Payload Size.) 000b = NT Port Link Interface supports a 128-byte maximum payload 001b = NT Port Link Interface supports a 256-byte maximum payload 010b = NT Port Link Interface supports a 512-byte maximum payload 011b = NT Port Link Interface supports a 1,024-byte maximum payload 100b = NT Port Link Interface supports a 2,048-byte maximum payload No other encodings are supported. <b>Note:</b> Software must halt all transactions through the NT Port before changing this field.	RW	Yes	000b

**Register 16-22. 70h Device Status and Control (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
8	<b>Extended Tag Field Enable</b> <i>Not supported</i>	RsvdP	No	0
9	<b>Phantom Functions Enable</b> <i>Not supported</i>	RsvdP	No	0
10	<b>AUX Power PM Enable</b> <i>Not supported</i>	RsvdP	No	0
11	<b>Enable No Snoop</b> <i>Not supported</i>	RW	Yes	1
14:12	<b>Maximum Read Request Size</b> <i>Not supported</i>	RsvdP	No	000b
15	<b>Reserved</b>	RsvdP	No	0
<b>Device Status</b>				
16	<b>Correctable Error Detected</b> 0 = NT Port Link Interface did not detect a Correctable error 1 = NT Port Link Interface detected a Correctable error, regardless of the bit 0 ( <i>Correctable Error Reporting Enable</i> ) state.	RW1C	Yes	0
17	<b>Non-Fatal Error Detected</b> 0 = NT Port Link Interface did not detect a Non-Fatal error 1 = NT Port Link Interface detected a Non-Fatal error, regardless of the bit 1 ( <i>Non-Fatal Error Reporting Enable</i> ) state	RW1C	Yes	0
18	<b>Fatal Error Detected</b> 0 = NT Port Link Interface did not detect a Fatal error 1 = NT Port Link Interface detected a Fatal error, regardless of the bit 2 ( <i>Fatal Error Reporting Enable</i> ) state	RW1C	Yes	0
19	<b>Unsupported Request Detected</b> 0 = NT Port Link Interface did not detect a UR 1 = NT Port Link Interface detected a UR, regardless of the bit 3 ( <i>Unsupported Request Reporting Enable</i> ) state	RW1C	Yes	0
20	<b>AUX Power Detected</b> <i>Not supported</i>	RsvdP	No	0
21	<b>Transactions Pending</b> <i>Not supported</i> Because the PEX 8748 NT Port is a bridging device, it does not track Completion for the corresponding Non-Posted transactions. Therefore, the NT Port Link Interface does not implement this bit.	RsvdP	No	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h



**Register 16-23. 74h Link Capability**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Note:</b> Table 13-5 indicates which Ports are enabled/used, and when, based upon the STRAP_STNx_PORTCFGx input states and/or serial EEPROM programming. The table also lists the relationship between the Ports and Lanes (Link widths).				
3:0	<b>Maximum Link Speed</b> Indicates the NT Port Link Interface's maximum Link speed. The encoding is the binary value of the bit location in the <b>Link Capability 2</b> register <i>Supported Link Speeds Vector</i> field (NT Port Link Interface, offset 94h[7:1]) that corresponds to the maximum Link speed. <i>For example</i> , the default value 3h indicates that the maximum Link speed is that which corresponds to bit 3 in the <i>Supported Link Speeds Vector</i> field, which is 8.0 GT/s. The default value for all Ports can be globally changed to 2h (Gen 2 maximum speed) or 1h (Gen 1 maximum speed), according to the STRAP_GEN1_GEN2 input state (Z = 2h (Gen 2), or 0 = 1h (Gen 1)).	RO	Yes	3h
9:4	<b>Maximum Link Width</b> The PEX 8748 maximum Link width is x16 = 01_0000b. Actual maximum Link width is defined by the STRAP_STNx_PORTCFGx 3-state inputs. 00_0000b = <b>Reserved</b> 00_0001b = x1 00_0010b = x2 00_0100b = x4 00_1000b = x8 01_0000b = x16 All other encodings are <b>not supported</b> .	Refer to Default value	No	Defined by STRAP_STNx_PORTCFGx input states, or programmed by serial EEPROM value for the <b>Port Configuration</b> -related register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offsets 300h through 308h)
11:10	<b>Active State Power Management (ASPM) Support</b> Active State Link PM support. Indicates the level of ASPM supported by the Port. 01b = L0s Link PM state entry is supported 11b = L0s and L1 Link PM states are supported All other encodings are <b>Reserved</b> .	RO	Yes	11b

**Register 16-23. 74h Link Capability (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
14:12	<b>L0s Exit Latency</b> Indicates the L0s Link PM state exit latency for the Port's PCI Express Link. The value reported indicates the length of time this Port requires to complete the L0s to L0 Link PM state transition. If L0s is not supported, the value of this field is undefined; however, for the recommended value, refer to the Implementation Note in the <i>PCI Express Base r3.0</i> , Section 5.4.1.1, "Potential Issues with Legacy Software When L0s is Not Supported." 000b = Less than 64 ns 001b = 64 ns to less than 128 ns 010b = 128 ns to less than 256 ns 011b = 256 ns to less than 512 ns 100b = 512 ns to less than 1 $\mu$ s 101b = 1 $\mu$ s to less than 2 $\mu$ s 110b = 2 to 4 $\mu$ s (default) 111b = More than 4 $\mu$ s	RO	No	110b
17:15	<b>L1 Exit Latency</b> Indicates the L1 Link PM state exit latency for the Port's PCI Express Link. The value reported indicates the length of time this Port requires to complete the ASPM L1 to L0 Link PM state transition. If L0 is not supported, the value of this field is undefined. 000b = Less than 1 $\mu$ s 001b = 1 $\mu$ s to less than 2 $\mu$ s (5.0 GT/s) 010b = 2 $\mu$ s to less than 4 $\mu$ s (default, 2.5 GT/s) 011b = 4 $\mu$ s to less than 8 $\mu$ s 100b = 8 $\mu$ s to less than 16 $\mu$ s 101b = 16 $\mu$ s to less than 32 $\mu$ s 110b = 32 to 64 $\mu$ s 111b = More than 64 $\mu$ s	RO	Yes	010b
18	<b>Clock Power Management</b>	RO	Yes	0
21:19	<b>Reserved</b>	RsvdP	No	000b
22	<b>ASPM Optionality Compliance</b>	HwInit	Yes	1
23	<b>Reserved</b>	RsvdP	No	0

**Register 16-23. 74h Link Capability (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default																																							
31:24	<b>NT Port Number</b> The NT Port Number value is determined by the STRAP_NT_UPSTRM_PORTSEL[2:0] input states, combined with the <b>VS0 Upstream</b> register <i>NT Port</i> field (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset 360h[12:8]) value.	Refer to Default value	No	Defined by STRAP_NT_UPSTRM_PORTSEL[2:0] input states																																							
	<table><tr><th>STRAP_NT_UPSTRM_PORTSEL[2:0] Input States</th><th>Offset 360h[12:8] Value</th><th>Port Number</th></tr><tr><td>000</td><td>0_0000b</td><td>0</td></tr><tr><td>00Z</td><td>0_0001b</td><td>1</td></tr><tr><td>001</td><td>0_0010b</td><td>2</td></tr><tr><td>0Z0</td><td>0_0011b</td><td>3</td></tr><tr><td>011</td><td>0_1000b</td><td>8</td></tr><tr><td>Z00</td><td>0_1001b</td><td>9</td></tr><tr><td>Z0Z</td><td>0_1010b</td><td>10</td></tr><tr><td>Z01</td><td>0_1011b</td><td>11</td></tr><tr><td>Z1Z</td><td>1_0000b</td><td>16</td></tr><tr><td>Z11</td><td>1_0001b</td><td>17</td></tr><tr><td>100</td><td>1_0010b</td><td>18</td></tr><tr><td>10Z</td><td>1_0011b</td><td>19</td></tr></table>				STRAP_NT_UPSTRM_PORTSEL[2:0] Input States	Offset 360h[12:8] Value	Port Number	000	0_0000b	0	00Z	0_0001b	1	001	0_0010b	2	0Z0	0_0011b	3	011	0_1000b	8	Z00	0_1001b	9	Z0Z	0_1010b	10	Z01	0_1011b	11	Z1Z	1_0000b	16	Z11	1_0001b	17	100	1_0010b	18	10Z	1_0011b	19
	STRAP_NT_UPSTRM_PORTSEL[2:0] Input States				Offset 360h[12:8] Value	Port Number																																					
	000				0_0000b	0																																					
	00Z				0_0001b	1																																					
	001				0_0010b	2																																					
	0Z0				0_0011b	3																																					
	011				0_1000b	8																																					
	Z00				0_1001b	9																																					
	Z0Z				0_1010b	10																																					
	Z01				0_1011b	11																																					
	Z1Z				1_0000b	16																																					
	Z11				1_0001b	17																																					
	100				1_0010b	18																																					
	10Z				1_0011b	19																																					
All other encodings are <i>Reserved</i> .																																											

**Register 16-24. 78h Link Status and Control**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Link Control</b>				
1:0	<b>Active State Power Management (ASPM) Control</b> 00b = Disable <sup>a</sup> 01b = Enables only L0s Link PM state Entry 10b = Enables only L1 Link PM state Entry 11b = Enables both L0s and L1 Link PM state Entries	RW	Yes	00b
2	<b>Reserved</b>	RsvdP	No	0
3	<b>Read Request Return Parameter Control</b> Read Request Return Parameter “R” control. Read Completion Boundary (RCB).	RO	Yes	0
4	<b>Link Disable</b> <i>Reserved</i> for the NT Port Link Interface. Functionality is enabled, however, when the <b>Power Management Hot Plug User Configuration</b> register <i>PHY Upstream Port Control Write Enable</i> bit (NT Port Link Interface, offset F70h[11]) is Set.	RsvdP	No	0
5	<b>Retrain Link</b> <i>Reserved</i> for the NT Port Link Interface. Always read as 0. Functionality is enabled, however, when the <b>Power Management Hot Plug User Configuration</b> register <i>PHY Upstream Port Control Write Enable</i> bit (NT Port Link Interface, offset F70h[11]) is Set.	RsvdP	No	0
6	<b>Common Clock Configuration</b> 0 = NT Port Link Interface and the device at the other end of the Port’s PCI Express Link uses an asynchronous Reference Clock source 1 = NT Port Link Interface and the device at the other end of the Port’s PCI Express Link uses a common Reference Clock source (constant phase relationship)	RW	Yes	0
7	<b>Extended Sync</b> Setting this bit causes the NT Port Link Interface to transmit: <ul style="list-style-type: none"> <li>• 4,096 FTS Ordered-Sets in the L0s Link PM state,</li> <li>• Followed by a single SKIP Ordered-Set prior to entering the L0 Link PM state,</li> <li>• Finally, transmission of 1,024 TS1 Ordered-Sets in the Recovery state.</li> </ul>	RW	Yes	0
15:8	<b>Reserved</b>	RsvdP	No	00h

**Register 16-24. 78h Link Status and Control (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Link Status</b>				
19:16	<b>Current Link Speed</b> Indicates the negotiated Link speed of the Port's PCI Express Link. The encoding is the binary value of the bit location in the <b>Link Capability 2</b> register <i>Supported Link Speeds Vector</i> field (NT Port Link Interface, offset 94h[7:1]) that corresponds to the current Link speed. <i>For example</i> , a value of 1h indicates that the current Link speed is that which corresponds to bit 1 in the <i>Supported Link Speeds Vector</i> field, which is 2.5 GT/s. If the Link is not up, the value of this field is undefined.	RO	No	1h
25:20	<b>Negotiated Link Width</b> Dependent upon the configuration of the physical Ports. Link width is determined by the negotiated value with the attached Lane/Port. If the Link is not up, the value of this field is undefined. 00_0000b = Link is Down (default) 00_0001b = x1 00_0010b = x2 00_0100b = x4 00_1000b = x8 01_0000b = x16 All other encodings are <i>not supported</i> .	RO	No	00_0000b
26	<b>Reserved</b>	RsvdP	No	0
27	<b>Link Training</b> <i>Reserved</i> for the NT Port Link Interface.	RsvdP	No	0
28	<b>Slot Clock Configuration</b> Set by the Upstream Port(s) or NT Port Link Interface, but not both. 0 = Indicates that the PEX 8748 uses an independent clock 1 = Indicates that the PEX 8748 uses the same physical Reference Clock that the platform provides on the connector	HwInit	Yes	0
31:29	<b>Reserved</b>	RsvdP	No	000b

a. The Port Receiver must be capable of entering the L0s Link PM state, regardless of whether the state is disabled.

**Register 16-25. 8Ch Device Capability 2**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
6:0	<i>Reserved</i>	RsvdP	No	0-0h
7	<b>AtomicOp 32-Bit Completer Supported</b> 1 = Supports Atomic 32-bit Completer	ROS	Yes	1
8	<b>AtomicOp 64-Bit Completer Supported</b> 1 = Supports Atomic 64-bit Completer	ROS	Yes	1
9	<b>AtomicOp 128-Bit Completer Supported</b> 1 = Supports Atomic 128-bit Completer	ROS	Yes	1
31:10	<i>Reserved</i>	RsvdP	No	0000_01h

**Register 16-26. 90h Device Status and Control 2**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Device Control 2</b>				
5:0	<i>Reserved</i>	RsvdP	No	0-0h
6	<b>AtomicOp Requester Enable</b>	RW	Yes	0
15:7	<i>Reserved</i>	RsvdP	No	0-0h
<b>Device Status 2</b>				
31:16	<i>Reserved</i>	RsvdP	No	0000h

**Register 16-27. 98h Link Status and Control 2**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Link Control 2</b>				
3:0	<b>Target Link Speed</b> Sets the Target Compliance mode speed when software is using bit 4 ( <i>Enter Compliance</i> ) to force a Link into Compliance mode.	RWS	Yes	3h
4	<b>Enter Compliance</b>	RWS	Yes	0
5	<b>Hardware Autonomous Speed Disable</b> <i>Reserved</i> Initial transition to the highest supported common Link speed is not blocked by this bit.	RsvdP	No	0
6	<b>Selectable De-Emphasis</b> <i>Reserved</i>	RsvdP	Yes	0
9:7	<b>Transmit Margin</b> Intended for debug and compliance testing only.	RWS	Yes	000b
10	<b>Enter Modified Compliance</b> Intended for debug and compliance testing only.	RWS	Yes	0
11	<b>Compliance SOS</b> <i>This bit is applicable only when the Link is operating at 2.5 or 5.0 GT/s.</i> 1 = Link Training and Status State Machine (LTSSM) must periodically send SKIP Ordered-Sets between sequences when sending the Compliance Pattern or Modified Compliance Pattern	RWS	Yes	0
15:12	<b>Compliance Preset/De-Emphasis</b> Intended for debug and compliance testing only.  <b>8.0 GT/s Data Rate</b> Sets the Transmitter Preset in the <i>Polling.Compliance</i> substate, if the entry occurred due to bit 4 ( <i>Enter Compliance</i> ) being Set. The encodings are defined in the <i>PCI Express Base r3.0</i> , Section 4.2.3.2, “Encoding of Presets.” Results are undefined if a <i>Reserved</i> preset encoding is used when entering the <i>Polling.Compliance</i> substate in this way.  <b>5.0 GT/s Data Rate</b> Sets the de-emphasis level in the <i>Polling.Compliance</i> substate, if the entry occurred due to bit 4 ( <i>Enter Compliance</i> ) being Set.  0h = -6 dB (Link is operating at 5.0 GT/s) 1h = -3.5 dB (Link is operating at 2.5 GT/s)  <b>2.5 GT/s Data Rate</b> The programming of this field has no effect.	RWS	Yes	0h

**Register 16-27. 98h Link Status and Control 2 (Cont.)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
<b>Link Status 2</b>				
16	<b>Current De-Emphasis Level</b> Reflects the de-emphasis level when the Link is operating at 5.0 GT/s. 0 = -6 dB (Link is operating at 5.0 GT/s) 1 = Undefined	RO	No	0
17	<b>Equalization Complete</b> 1 = Indicates that the Transmitter Equalization procedure has completed	ROS	Yes	0
18	<b>Equalization Phase 1 Successful</b> 1 = Indicates that Phase 1 of the Transmitter Equalization procedure has successfully completed	ROS	Yes	0
19	<b>Equalization Phase 2 Successful</b> 1 = Indicates that Phase 2 of the Transmitter Equalization procedure has successfully completed	ROS	Yes	0
20	<b>Equalization Phase 3 Successful</b> 1 = Indicates that Phase 3 of the Transmitter Equalization procedure has successfully completed	ROS	Yes	0
21	<b>Link Equalization Request</b> Set by hardware, to request the Link equalization process to be performed on the Link.	RW1C	Yes	0
31:22	<b>Reserved</b>	RsvdP	No	0-0h



## 16.9 NT Port Link Interface Vendor-Specific Capability 3 Registers (Offsets C8h – FCh)

This section details the NT Port Link Interface Vendor-Specific Capability 3 registers. [Table 16-6](#) defines the register map used by the NT Port Link Interface.

The Cursor Mechanism registers at offsets **F8h** and **FCh** provide a means for accessing PCI Express Extended Configuration Space registers (offsets 100h through FFFh) within the NT Port Link and Virtual Interfaces, when only standard PCI Configuration transactions (that do not support the *Extended Register Number* field within the Completion Request Header) are available.

**Table 16-6. NT Port Link Interface Vendor-Specific Capability 3 Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
<b>Reserved</b>	Next Capability Pointer (00h)	Capability ID 3 (09h)
<b>Reserved</b>		CCh –
NT Port Link Interface BAR0/1 Setup		E4h
NT Port Link Interface Memory BAR2 Setup		E8h
NT Port Link Interface Memory BAR2/3 Setup		ECh
NT Port Link Interface Memory BAR4 Setup		F0h
NT Port Link Interface Memory BAR4/5 Setup		F4h
Configuration Address Window	<b>Reserved</b>	
Configuration Data Window		FCh

### Register 16-28. C8h Vendor-Specific Capability 3

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Capability ID 3</b>	ROS	Yes	09h
15:8	<b>Next Capability Pointer</b> 00h = This capability is the last capability in the Linked List	ROS	Yes	00h
31:16	<i>Reserved</i>	RsvdP	No	0000h

**Register 16-29. E4h NT Port Link Interface BAR0/1 Setup**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>BAR0/1 Enable</b> 00b = Disables Link Interface <b>BAR0</b> and <b>BAR1</b> 01b = <i>Reserved</i> 10b = Enables Link Interface <b>BAR0</b> and disables <b>BAR1</b> ( <b>BAR0</b> is a 32-bit BAR) 11b = Enables Link Interface <b>BAR0</b> and <b>BAR1</b> ( <b>BAR0/1</b> is a 64-bit BAR)	RWS	No	10b
2	<b>BAR0 Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RWS	No	0
31:3	<i>Reserved</i>	RsvdP	No	0-0h

**Register 16-30. E8h NT Port Link Interface Memory BAR2 Setup**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default	
0	<b>Type Selector</b>	RsvdP	No	0	
2:1	<b>BAR2 Type</b> 00b = <b>BAR2</b> is implemented as a 32-bit Memory BAR 10b = <b>BAR2/3</b> is implemented as a 64-bit Memory BAR No other encodings are allowed.	RWS	Yes	00b	
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	RWS	Yes	0	
19:4	<i>Reserved</i>	RsvdP	No	0_000h	
30:20	<b>BAR2 Size</b> Specifies the Address Range size requested by <b>BAR2</b> . 0 = Corresponding <b>BAR2</b> bits are RO bits that always return a value of 0, and Writes are ignored 1 = Corresponding <b>BAR2</b> bits are RWS bits	RWS	Yes	0-0h	
31	<b>BAR2 Enable</b> 0 = <b>BAR2</b> is disabled, all <b>BAR2</b> bits read 0 1 = <b>BAR2</b> is enabled	Field [2:1] ( <i>BAR2 Type</i> ) = 00b	RWS	Yes	0
	<b>BAR2 Size</b> Included with field [30:20] when this BAR is used as a 64-bit BAR.	Field [2:1] ( <i>BAR2 Type</i> ) = 10b	RWS	Yes	0

**Register 16-31. ECh NT Port Link Interface Memory BAR2/3 Setup**

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default
<i><b>Note:</b> This register has RWS privilege if <b>BAR2/3</b> is configured as a 64-bit BAR (<b>NT Port Link Interface Memory BAR2 Setup</b> register <b>BAR2 Type</b> field (NT Port Link Interface, offset <b>E8h[2:1]</b>), is programmed to 10b).</i>					
0	<b>Type Selector</b>	Offset E8h[2:1]=00b	RsvdP	No	0
		Offset E8h[2:1]=10b	RWS	Yes	0
2:1	<b>BAR3 Type</b> 00b = <b>BAR3</b> is implemented as a 32-bit Memory BAR No other encodings are allowed.	Offset E8h[2:1]=00b	RsvdP	No	00b
		Offset E8h[2:1]=10b	RWS	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable		RWS	Yes	0
19:4	<b>Reserved</b>	Offset E8h[2:1]=00b	RsvdP	No	0_000h
	When <b>BAR2/3</b> are used as a 64-bit BAR, bits [31:0] (including bits [19:4]) are used as the upper 32 bits.	Offset E8h[2:1]=10b	RWS	Yes	0_000h
30:20	<b>BAR3 Size</b> Specifies the Address Range size requested by <b>BAR3</b> . 0 = Corresponding <b>BAR3</b> bits are RO bits that always return a value of 0, and Writes are ignored 1 = Corresponding <b>BAR3</b> bits are RWS bits		RWS	Yes	0-0h
31	<b>BAR3 Enable</b> <b>32-Bit BAR</b> 0 = <b>BAR3</b> is disabled 1 = <b>BAR3</b> is enabled as a 32-bit BAR	Offset E8h[2:1]=00b	RWS	Yes	0
	<b>64-Bit BAR</b> 0 = <b>BAR2/3</b> is disabled, all <b>BAR2/3</b> bits read 0 1 = <b>BAR2/3</b> is enabled as a 64-bit BAR	Offset E8h[2:1]=10b	RWS	Yes	0

**Register 16-32. F0h NT Port Link Interface Memory BAR4 Setup**

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>Type Selector</b>		RsvdP	No	0
2:1	<b>BAR4 Type</b> 00b = <b>BAR4</b> is implemented as a 32-bit Memory BAR ( <b>BAR4</b> ) 10b = <b>BAR4/5</b> is implemented as a 64-bit Memory BAR ( <b>BAR4/5</b> ) No other encodings are allowed.		RWS	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable		RWS	Yes	0
19:4	<b>Reserved</b>		RsvdP	No	0_000h
30:20	<b>BAR4 Size</b> Specifies the Address Range size requested by <b>BAR4</b> . 0 = Corresponding <b>BAR4</b> bits are RO bits that always return a value of 0, and Writes are ignored 1 = Corresponding <b>BAR4</b> bits are RWS bits		RWS	Yes	0-0h
31	<b>BAR4 Enable</b> 0 = <b>BAR4</b> is disabled, all <b>BAR4</b> bits read 0 1 = <b>BAR4</b> is enabled	Field [2:1] ( <i>BAR4 Type</i> ) = 00b	RWS	Yes	0
	<b>BAR4 Size</b> Included with field [30:20] when this BAR is used as a 64-bit BAR.	Field [2:1] ( <i>BAR4 Type</i> ) = 10b	RWS	Yes	0

**Register 16-33. F4h NT Port Link Interface Memory BAR4/5 Setup**

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default
<i><b>Note:</b> This register has RWS privilege if <b>BAR4/5</b> is configured as a 64-bit BAR (<b>NT Port Link Interface Memory BAR4 Setup register BAR4 Type</b> field (NT Port Link Interface, offset <b>F0h[2:1]</b>), is programmed to 10b).</i>					
0	<b>Type Selector</b>	Offset F0h[2:1]=00b	RsvdP	No	0
		Offset F0h[2:1]=10b	RWS	Yes	0
2:1	<b>BAR5 Type</b> 00b = <b>BAR5</b> is implemented as a 32-bit Memory BAR No other encodings are allowed.	Offset F0h[2:1]=00b	RsvdP	No	00b
		Offset F0h[2:1]=10b	RWS	Yes	00b
3	<b>Prefetchable</b> 0 = Non-Prefetchable 1 = Prefetchable	Offset F0h[2:1]=00b	RsvdP	No	0
		Offset F0h[2:1]=10b	RWS	Yes	0
19:4	<b>Reserved</b>	Offset F0h[2:1]=00b	RsvdP	No	0_000h
	When <b>BAR4/5</b> are used as a 64-bit BAR, bits [31:0] (including bits [19:4]) are used as the upper 32 bits.	Offset F0h[2:1]=10b	RWS	Yes	0_000h
30:20	<b>BAR5 Size</b> Specifies the Address Range size requested by <b>BAR5</b> . 0 = Corresponding <b>BAR5</b> bits are RO bits that always return a value of 0, and Writes are ignored 1 = Corresponding <b>BAR5</b> bits are RWS bits		RWS	Yes	0-0h
31	<b>BAR5 Enable</b> <b>32-Bit BAR</b> 0 = <b>BAR5</b> is disabled 1 = <b>BAR5</b> is enabled as a 32-bit BAR	Offset F0h[2:1]=00b	RWS	Yes	0
	<b>64-Bit BAR</b> 0 = <b>BAR4/5</b> is disabled, all <b>BAR4/5</b> bits read 0 1 = <b>BAR4/5</b> is enabled as a 64-bit BAR	Offset F0h[2:1]=10b	RWS	Yes	0

**Register 16-34. F8h Configuration Address Window  
(Device-Specific Cursor Mechanism)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<i>Reserved</i>	RsvdP	No	0000h
25:16	<b>Register Offset</b>	RW	Yes	0-0h
30:26	<i>Reserved</i>	RsvdP	No	0-0h
31	<b>Interface Select</b> 0 = Access is to the NT Port Link Interface Type 0 Configuration Space register 1 = Access is to the NT Port Virtual Interface Type 0 Configuration Space register	RW	Yes	0

**Register 16-35. FCh Configuration Data Window  
(Device-Specific Cursor Mechanism)**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Register Data</b> Software selects a register by writing into the NT Port Link Interface Configuration Address window, then reads from or writes to that register using this register.	RW	Yes	0000_0000h

The registers detailed in [Section 13.15, “Virtual Channel Extended Capability Registers \(Offsets 148h – 1BCh\),”](#) are also applicable to the NT Port Link Interface, except as defined in [Table 16-7](#) (register map; offsets 178h through 1B4h are **Reserved**), and [Register 16-36](#) through [Register 16-40](#).

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16                      15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

Next Capability Offset (C34h)	Capability Version (1h)	PCI Express Extended Capability ID (0002h)	148h
<i>Reserved</i>		Port VC Capability 1	14Ch
Port VC Capability 2			150h
Port VC Status ( <i>Reserved</i> )		Port VC Control	154h
<i>Reserved</i>		VC0 Resource Capability	158h
VC0 Resource Control			15Ch
VC0 Resource Status		<i>Reserved</i>	160h
<i>Reserved</i>			164h – 1BCh

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>PCI Express Extended Capability ID</b> Program to 0002h, as required by the <i>PCI Express Base r3.0</i> .	RO	No	0002h
19:16	<b>Capability Version</b> Program to 1h, as required by the <i>PCI Express Base r3.0</i> .	RO	No	1h
31:20	<b>Next Capability Offset</b> Next extended capability is the <b>Vendor-Specific Extended Capability 2</b> structure, offset C34h.	RO	No	C34h

**Register 16-37. 14Ch Port VC Capability 1**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
2:0	<b>Extended VC Counter</b> 000b = NT Port Link Interface supports only one Virtual Channel, VC0 001b = <i>Reserved</i> , because the PEX 8748 supports only one VC All other encodings are <i>Reserved</i> .	RO	Yes	000b
3	<i>Reserved</i>	RsvdP	No	0
6:4	<b>Low-Priority Extended VC Counter</b> For Strict Priority arbitration, indicates the quantity of extended Virtual Channels (VCs) (those in addition to VC0) that belong to the Low-Priority VC group for the NT Port Link Interface. 000b = For the NT Port Link Interface, only VC0 belongs to the Low-Priority VC group 001b = <i>Reserved</i> , because the PEX 8748 supports only one VC All other encodings are <i>Reserved</i> .	RO	Yes	000b
7	<i>Reserved</i>	RsvdP	No	0
9:8	<b>Reference Clock</b> <i>Reserved</i>	RsvdP	No	00b
11:10	<b>Port Arbitration Table Entry Size</b>	RsvdP	No	00b
31:12	<i>Reserved</i>	RsvdP	No	0000_0h

**Register 16-38. 158h VC0 Resource Capability**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
1:0	<b>Port Arbitration Capability</b> Bit 0 = 1 – Non-configurable Round-Robin (Hardware-Fixed) arbitration Bit 1 = 1 – Weighted Round-Robin (WRR) arbitration with 64 Phases	RO	No	00b
13:2	<i>Reserved</i>	RsvdP	No	0-0h
14	<b>Advanced Packet Switching</b>	RsvdP	No	0
15	<b>Reject Snoop Transactions</b> Not a PCI Express switch feature; therefore, this bit is Cleared.	RsvdP	No	0
22:16	<b>Maximum Time Slots</b> <i>Not supported</i>	RsvdP	No	000_0000b
23	<i>Reserved</i>	RsvdP	No	0
31:24	<b>Port Arbitration Table Offset</b>	RsvdP	No	00h



**Register 16-39. 15Ch VC0 Resource Control**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>TC/VC Map</b> Defines Traffic Classes [7:0], respectively, and indicates which TCs are mapped into VC0. Traffic Class 0 (TC0) must be mapped to VC0. By default, Traffic Classes [7:1] are mapped to VC0.	RO	No	1
7:1		RW	Yes	7Fh
15:8	<b>Reserved</b>	RsvdP	No	00h
16	<b>Load Port Arbitration Table</b> Hardware writable and software readable.	RW	Yes	0
19:17	<b>Port Arbitration Select</b> Selects the Port Arbitration type for the NT Port Link Interface. Indicates the bit number in the <b>VC0 Resource Capability</b> register <i>Port Arbitration Capability</i> field (offset 158h[1:0]) that corresponds to the arbitration type. 0 = Round-Robin (Hardware-Fixed) arbitration scheme	RW	Yes	000b
23:20	<b>Reserved</b>	RsvdP	No	0h
26:24	<b>VC ID</b> Defines the NT Port Link Interface VC0 ID code. 000b = VC0 (default; VC0 is the only/default VC) All other encodings are <b>Reserved</b> .	RO	No	000b
30:27	<b>Reserved</b>	RsvdP	No	0h
31	<b>VC Enable</b> 0 = Not allowed 1 = Enables the NT Port Link Interface VC0	RO	No	1

**Register 16-40. 160h VC0 Resource Status**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	<b>Reserved</b>	RsvdP	No	0000h
16	<b>Port Arbitration Table Status</b> <i>Not implemented</i>	RO	No	0
17	<b>VC0 Negotiation Pending</b> 0 = VC0 negotiation completed 1 = VC0 initialization is not complete for the NT Port Link Interface	RO	Yes	1
31:18	<b>Reserved</b>	RsvdP	No	0-0h

16.11 NT Port Link Interface Device-Specific Registers (Offsets 1C0h – A74h)

The registers detailed in [Section 13.16, “Device-Specific Registers \(Offsets 1C0h – A74h\),”](#) are unique to the PEX 8748 and not referenced in the *PCI Express Base r3.0*. These registers are also applicable to the NT Port Link Interface, except as defined in [Table 16-8](#) (register map; offsets 1D0h through 1D8h, 200h through 270h, 290h through 46Ch, 600h through 68Ch, 6A0h through 6A8h, 760h through 8FCh, 900h through 93Ch, and 9ECh through A2Ch, and A30h through A74h, are **Reserved**) through [Table 16-10](#), and [Register 16-41](#) and [Register 16-42](#).

Other NT Port Link Interface Device-Specific registers are detailed in:

- [Section 16.12, “NT Port Link Interface Device-Specific Registers \(Offsets B70h – C88h\)”](#)
- [Section 16.14, “NT Port Link Interface Device-Specific Registers \(Offsets F00h – FB0h\)”](#)

*Note: It is recommended that these registers not be changed from their default values.*

Table 16-8. NT Port Link Interface Device-Specific Register Map (Offsets 1C0h – A74h)

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Reserved	1C0h – 1D8h
NT Port Link Interface Device-Specific Registers – Captured Bus Number and Device Number (Offsets 1DCh – 1FCh)	1DCh ... 1FCh
Reserved	200h – 6FCh
NT Port Link Interface Device-Specific Registers – Error Checking and Debug (Offsets 700h – 75Ch)	700h ... 75Ch
Reserved	760h – A74h

16.11.1 NT Port Link Interface Device-Specific Registers – Captured Bus Number and Device Number (Offsets 1DCh – 1FCh)

The registers detailed in [Section 13.16.2](#), “[Device-Specific Registers – Captured Bus and Device Numbers \(Offsets 1DCh – 1E4h\)](#),” are also applicable to the NT Port Link Interface, except as defined in [Table 16-9](#) (register map; offset 1E0h is not *Reserved*; offset 1E4h is *Reserved*), and [Register 16-41](#) and [Register 16-42](#).

Table 16-9. NT Port Link Interface Device-Specific Captured Bus Number and Device Number Register Map

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
NT Captured Bus Number																																1DCh
NT Captured Device Number																																1E0h
Reserved																																1E4h – 1FCh

**Register 16-41. 1DCh NT Captured Bus Number**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Captured Bus Number</b> Captured Bus Number value for the NT Port Link Interface. The value of this field can be overwritten, if bit 31 ( <i>Captured BusDev Number Override</i> ) is Set prior to changing the Captured Bus Number. <i>Note: Overwriting the Captured Bus Number value is not recommended.</i>	RW	Yes	00h
12:8	<b>Captured Device Number</b> Captured Device Number value for the NT Port Link Interface. The value of this field can be overwritten, if bit 31 ( <i>Captured BusDev Number Override</i> ) is Set prior to changing the Captured Device Number. <i>Note: Overwriting the Captured Device Number value is not recommended.</i>	RW	Yes	0-0h
30:13	<i>Reserved</i>	RsvdP	No	0-0h
31	<b>Captured BusDev Number Override</b> 1 = Enables the Captured Bus Number and Device Number to be overridden	RW	Yes	0

**Register 16-42. 1E0h NT Captured Device Number**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
7:0	<b>Captured Device Number</b> NT Port Link Interface Captured Device Number register value. <i>Note: Overwriting the Captured Device Number value is not recommended.</i>	RWS	Yes	00h
31:8	<i>Reserved</i>	RsvdP	No	0000_00h

16.11.2 NT Port Link Interface Device-Specific Registers – Error Checking and Debug (Offsets 700h – 75Ch)

The registers detailed in [Section 13.16.11, “Device-Specific Registers – Error Checking and Debug \(Offsets 700h – 75Ch\),”](#) are also applicable to the NT Port Link Interface, except as defined in [Table 16-10](#) (register map; offsets 700h through 720h, and 72Ch, are *Reserved*).

Table 16-10. Device-Specific Error Checking and Debug Register Map (Offsets 700h – 75Ch)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Reserved																																700h – 720h
Reserved																Gen 3 Framing Error Status																724h
Reserved																Gen 3 Framing Error Mask																728h
Reserved																																72Ch – 75Ch

# 16.12 NT Port Link Interface Device-Specific Registers (Offsets B70h – C88h)

The registers detailed in [Section 13.18, “Device-Specific Registers \(Offsets B70h – DFCh\)”](#) (for offsets B70h through C88h), are unique to the PEX 8748 and not referenced in the *PCI Express Base r3.0*. These registers are also applicable to the NT Port Link Interface, except as defined in [Table 16-11](#) (register map; offsets B80h through C30h are **Reserved**, and offsets C34h through C88h are associated with NT Port-specific registers) and [Table 16-12](#), and [Register 16-43](#) through [Register 16-46](#).

Other NT Port Link Interface Device-Specific registers are detailed in:

- [Section 16.11, “NT Port Link Interface Device-Specific Registers \(Offsets 1C0h – A74h\)”](#)
- [Section 16.14, “NT Port Link Interface Device-Specific Registers \(Offsets F00h – FB0h\)”](#)

*Note: It is recommended that these registers not be changed from their default values.*

**Table 16-11. NT Port Link Interface Device-Specific Register Map (Offsets B70h – C88h)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																
Next Capability Offset 2 (000h)																1h																PCI Express Extended Capability ID 2 (000Bh)																B70h
Device-Specific Registers – Vendor-Specific Extended Capability 2 (Offsets B70h – B7Ch)																																										...	B7Ch					
Reserved																																										B80h –	C30h					
Next Capability Offset 4 (B70h)																1h																PCI Express Extended Capability ID 4 (000Bh)																C34h
NT Port Link Interface Device-Specific Registers – Vendor-Specific Extended Capability 4 (Offsets C34h – C88h)																																										...	C88h					

### 16.12.1 NT Port Link Interface Device-Specific Registers – Vendor-Specific Extended Capability 4 (Offsets C34h – C88h)

The registers detailed in [Section 15.12.1, “NT Port Virtual Interface Device-Specific Registers – Vendor-Specific Extended Capability 4 \(Offsets C34h – C88h\),”](#) are also applicable to the NT Port Link Interface, except as defined in [Table 16-12](#) (register map), and [Register 16-43](#) through [Register 16-46](#).

**Table 16-12. NT Port Link Interface Device-Specific, Vendor-Specific Extended Capability 4 Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																
Next Capability Offset 4 (B70h)												Capability Version 4 (1h)				PCI Express Extended Capability ID 4 (000Bh)																C34h
Vendor-Specific Header 4																																C38h
Memory BAR2 Address Translation Lower																																C3Ch
Memory BAR3 Address Translation Upper																																C40h
Memory BAR4 Address Translation Lower																																C44h
Memory BAR5 Address Translation Upper																																C48h
Reserved																Virtual Interface IRQ Set																C4Ch
Reserved																Virtual Interface IRQ Clear																C50h
Reserved																Virtual Interface IRQ Mask Set																C54h
Reserved																Virtual Interface IRQ Mask Clear																C58h
Reserved																Link Interface IRQ Set																C5Ch
Reserved																Link Interface IRQ Clear																C60h
Reserved																Link Interface IRQ Mask Set																C64h
Reserved																Link Interface IRQ Mask Clear																C68h
NT Port SCRATCH0																																C6Ch
NT Port SCRATCH1																																C70h
NT Port SCRATCH2																																C74h
NT Port SCRATCH3																																C78h
NT Port SCRATCH4																																C7Ch
NT Port SCRATCH5																																C80h
NT Port SCRATCH6																																C84h
NT Port SCRATCH7																																C88h

**Register 16-43. C3Ch Memory BAR2 Address Translation Lower**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<i>Reserved</i>	RsvdP	No	0_0000h
31:20	<b>NT Port Link-to-Virtual Interface BAR2 Base Translation Address</b> Base Translation address when <b>BAR2</b> is enabled ( <b>NT Port Link Interface Memory BAR2 Setup</b> register <i>BAR2 Enable</i> bit (NT Port Link Interface, offset <b>E8h</b> [31]) is Set).	RW	Yes	000h

**Register 16-44. C40h Memory BAR3 Address Translation Upper**

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<i>Reserved</i>	Offset <b>ECh</b> [31]=0	RsvdP	No	0_0000h
	When <b>BAR2/3</b> are used as a 64-bit BAR, bits [31:0] (including bits [19:0]) are used as the upper 32 bits.	Offset <b>ECh</b> [31]=1	RW	Yes	0_0000h
31:20	<b>NT Port Link-to-Virtual Interface BAR3 Base Translation Address</b> Base Translation address when <b>BAR3</b> is enabled ( <b>NT Port Link Interface Memory BAR2/3 Setup</b> register <i>BAR3 Enable</i> bit (NT Port Link Interface, offset <b>ECh</b> [31]) is Set).		RW	Yes	000h

**Register 16-45. C44h Memory BAR4 Address Translation Lower**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<i>Reserved</i>	RsvdP	No	0_0000h
31:20	<b>NT Port Link-to-Virtual Interface BAR4 Base Translation Address</b> Base Translation address when <b>BAR4</b> is enabled ( <b>NT Port Link Interface Memory BAR4 Setup</b> register <i>BAR4 Enable</i> bit (NT Port Link Interface, offset <b>F0h</b> [31]) is Set).	RW	Yes	000h

**Register 16-46. C48h Memory BAR5 Address Translation Upper**

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default
19:0	<i>Reserved</i>	Offset <b>F4h</b> [31]=0	RsvdP	No	0_0000h
	When <b>BAR4/5</b> are used as a 64-bit BAR, bits [31:0] (including bits [19:0]) are used as the upper 32 bits.	Offset <b>F4h</b> [31]=1	RW	Yes	0_0000h
31:20	<b>NT Port Link-to-Virtual Interface BAR5 Base Translation Address</b> Base Translation address when <b>BAR5</b> is enabled ( <b>NT Port Link Interface Memory BAR4/5 Setup</b> register <i>BAR5 Enable</i> bit (NT Port Link Interface, offset <b>F4h</b> [31]) is Set).		RW	Yes	000h



16.13

NT Bridging-Specific Registers  
(Offsets C8Ch – EFCh)

Table 16-13 defines the register map of the NT Port Link Interface NT Bridging-Specific registers.

Table 16-13. NT Port Link Interface NT Bridging-Specific Register Map

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																												15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																														
NT Bridging-Specific Registers – NT Port ID and Requester ID Read Back (Offsets C8Ch – C94h)																																																								C8Ch	...	C94h
Reserved																																																								C98h –	DB0h	
NT Bridging-Specific Registers – Requester ID Translation Lookup Table Entry (Addresses DB4h – DF0h)																																																								DB4h	...	DF0h
NT Bridging-Specific Registers – Lookup Table Enable Link (Offsets DF4h – DFCh)																																																								DF4h	...	DFCh
Reserved																																																								E00h –	EFCh	

### 16.13.1 NT Bridging-Specific Registers – NT Port ID and Requester ID Read Back (Offsets C8Ch – C94h)

The registers detailed in [Section 15.13.1, “NT Bridging-Specific Registers – NT Port ID \(Offsets C8Ch – C94h\),”](#) are also applicable to the NT Port Link Interface, except as defined in [Table 16-14](#) (register map; offset C90h is not *Reserved*), and [Register 16-47](#) through [Register 16-48](#).

**Table 16-14. NT Port Link Interface NT Bridging-Specific NT Port ID and Requester ID Read Back Register Map**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
NT Port ID																C8Ch																
Reserved																Requester ID Read Back																C90h
Factory Test Only																																C94h

**Register 16-47. C8Ch NT Port ID**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	NT Port ID 0 = NT Port 1 = <i>Reserved</i>	ROS	No	0
30:1	<i>Reserved</i>	RsvdP	No	0-0h
31	Link Side Identifier	ROS	No	0

**Register 16-48. C90h Requester ID Read Back**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	Requester ID Read Back Memory Read to this location returns the Requester ID of the last TLP.	RO	No	0000h
31:16	<i>Reserved</i>	RsvdP	No	0000h

### 16.13.2 NT Bridging-Specific Registers – Requester ID Translation Lookup Table Entry (Addresses DB4h – DF0h)

This section describes the NT Port Link Interface NT Bridging-Specific Requester ID Translation Lookup Table (LUT) Entry registers. The NT Port uses these registers for Requester ID translation when it forwards:

- Memory Requests from the NT Port Link Interface to the NT Port Virtual Interface, –or–
- Completion TLPs from the NT Port Virtual Interface to the NT Port Link Interface

If the application needs to send traffic through the NT Port Link Interface, program the registers listed in this group with the corresponding Requester's Requester ID, then Set the *LUT Entry\_n Enable* and *LUT Entry\_m Enable* bits (bits 0 and 16, respectively) for each LUT entry, as needed.

Table 16-15 defines the register and address locations, as they relate to Register 16-49.

**Table 16-15. NT Port Link Interface NT Bridging-Specific Requester ID Translation LUT Entry\_n\_m Register Locations**

ADDR Location	Lookup Table Entry_n_m	ADDR Location	Lookup Table Entry_n_m
DB4h	0_1	DD4h	16_17
DB8h	2_3	DD8h	18_19
DBCh	4_5	DDCh	20_21
DC0h	6_7	DE0h	22_23
DC4h	8_9	DE4h	24_25
DC8h	10_11	DE8h	26_27
DCCh	12_13	DECh	28_29
DD0h	14_15	DF0h	30_31

**Register 16-49. DB4h – DF0h NT Port Link Interface Requester ID Translation**  
**LUT Entry  $_n\_m$  (where  $_n\_m = 0\_1$  through  $30\_31$ )**

Bit(s)	Description		Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>LUT Entry <math>_n</math> Enable</b> 0 = Disables 1 = Enables		RW	Yes	0
1	<b>LUT Entry <math>_n</math> No Snoop Enable</b> If Set, the NT Port Clears the TLP <i>No Snoop</i> attribute bit for the Memory Request, then goes from the NT Port Link Interface to the NT Port Virtual Interface, and re-calculates the ECRC. If the original TLP has an ECRC error, the NT Port corrupts the re-calculated ECRC before transmitting to the other Host domain. The NT Port sets the <i>No Snoop</i> attribute bit when it forwards the Completion TLP from the NT Port Virtual Interface to the NT Port Link Interface if this bit is Set for the corresponding Requester ID entry. This ECRC rule applies to Completion TLPs as well.  0 = Disables 1 = Enables		RW	Yes	0
2	<b>LUT Entry <math>_n</math> Function Number Bit 2</b>		RW	Yes	0
7:3	<b>Requester ID on Link Side</b>	<b>Device Number</b> LUT Entry $_n$ Requester Device Number.	RW	Yes	0000_0b
15:8		<b>Bus Number</b> LUT Entry $_n$ Requester Bus Number.	RW	Yes	00h
16	<b>LUT Entry <math>_m</math> Enable</b> 0 = Disables 1 = Enables		RW	Yes	0
17	<b>LUT Entry <math>_m</math> No Snoop Enable</b> If Set, the NT Port Clears the TLP <i>No Snoop</i> attribute bit for the Memory Request, then goes from the NT Port Link Interface to the NT Port Virtual Interface, and re-calculates the ECRC. If the original TLP has an ECRC error, the NT Port corrupts the re-calculated ECRC before transmitting to the other Host domain. The NT Port sets the <i>No Snoop</i> attribute bit when it forwards the Completion TLP from the NT Port Virtual Interface to the NT Link Virtual Interface if this bit is Set for the corresponding Requester ID entry. This ECRC rule applies to Completion TLPs as well.  0 = Disables 1 = Enables		RW	Yes	0
18	<b>LUT Entry <math>_n</math> Function Number Bit 2</b>		RW	Yes	0
23:19	<b>Requester ID on Link Side</b>	<b>Device Number</b> LUT Entry $_m$ Requester Device Number.	RW	Yes	0000_0b
31:24		<b>Bus Number</b> LUT Entry $_m$ Requester Bus Number.	RW	Yes	00h

### 16.13.3 NT Bridging-Specific Registers – Lookup Table Enable Link (Offsets DF4h – DFCh)

This section describes the NT Port Link Interface NT Bridging-Specific Lookup Table Enable Link registers. [Table 16-16](#) defines the register map.

**Table 16-16. NT Port Link Interface NT Bridging-Specific Lookup Table Enable Link Register Map**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Lookup Table Enable Link	DF4h
Lookup Table No Snoop Enable Link	DF8h
Lookup Table Function Enable Link	DFCh

#### Register 16-50. DF4h Lookup Table Enable Link

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Lookup Table Enable</b> Enable bit per Lookup Table.	RWS	Yes	0000_0000h

#### Register 16-51. DF8h Lookup Table No Snoop Enable Link

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Lookup Table No Snoop Enable</b> No Snoop Enable bit per Lookup Table.	RWS	Yes	0000_0000h

#### Register 16-52. DFCh Lookup Table Function Enable Link

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
31:0	<b>Lookup Table Function Check Enable</b> Function Check Enable bit per Lookup Table.	RWS	Yes	0000_0000h

## 16.14 NT Port Link Interface Device-Specific Registers (Offsets F00h – FB0h)

The registers detailed in [Section 13.20, “Device-Specific Registers – Virtual Switch \(Offsets F00h – F20h\), Virtual Switch Mode,”](#) and [Section 13.22, “Device-Specific Registers \(Offsets F30h – FB0h\),”](#) are unique to the PEX 8748 and not referenced in the *PCI Express Base r3.0*. These registers are also applicable to the NT Port Link Interface, except as defined in [Table 16-17](#) (register map; offsets F30h, F38h, F44h, and F4Ch through F58h are **Reserved**) through [Table 16-19](#), and [Register 16-53](#).

Other NT Port Link Interface Device-Specific registers are detailed in:

- [Section 16.11, “NT Port Link Interface Device-Specific Registers \(Offsets 1C0h – A74h\)”](#)
- [Section 16.12, “NT Port Link Interface Device-Specific Registers \(Offsets B70h – C88h\)”](#)

**Note:** *It is recommended that these registers not be changed from their default values.*

**Table 16-17. NT Port Link Interface Device-Specific Register Map (Offsets F00h – FB0h)**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
NT Port Link Interface Device-Specific Registers – Virtual Switch (Offsets F00h – F20h), Virtual Switch Mode		F00h ... F20h
<b>Reserved</b>		F24h – F6Ch
NT Port Link Interface Device-Specific Registers – Error Checking and Debug (Offsets F70h – FB0h)		F70h ... FB0h

### 16.14.1 NT Port Link Interface Device-Specific Registers – Virtual Switch (Offsets F00h – F20h), Virtual Switch Mode

**Note:** *In Base mode, this entire structure is **Reserved**, RsvdP, not serial EEPROM nor I<sup>2</sup>C writable, and has a default value of 0h.*

The registers detailed in [Section 13.20, “Device-Specific Registers – Virtual Switch \(Offsets F00h – F20h\), Virtual Switch Mode,”](#) are also applicable to the NT Port Link Interface, except as defined in [Table 16-18](#) (register map; offset F20h is **Reserved**).

**Table 16-18. NT Port Link Interface Device-Specific Virtual Switch Register Map (Offsets F00h – F20h) (Virtual Switch mode – VS Upstream Port(s))**

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
VS Failover		F00h
<b>Reserved</b>		F04h – F20h

## 16.14.2 NT Port Link Interface Device-Specific Registers – Error Checking and Debug (Offsets F70h – FB0h)

The registers detailed in [Section 13.22.3, “Device-Specific Registers – Error Checking and Debug \(Offsets F70h – FB0h\)”](#), are also applicable to the NT Port Link Interface, except as defined in [Table 16-19](#) (register map) and [Register 16-53](#).

Other NT Port Link Interface Device-Specific Error Checking and Debug registers are detailed in [Section 16.11.2, “NT Port Link Interface Device-Specific Registers – Error Checking and Debug \(Offsets 700h – 75Ch\)”](#).

**Table 16-19. NT Port Link Interface Device-Specific Error Checking and Debug Register Map**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Power Management Hot Plug User Configuration																																F70h
<i>Reserved</i>																																F74h – FA4h
ACK Transmission Latency Limit																																FA8h
Bad TLP Count																																FACH
Bad DLLP Count																																FB0h

**Register 16-53. F70h Power Management Hot Plug User Configuration**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
0	<b>L0s Entry Idle Count</b> Traffic Idle time to meet, to enter the L0s Link PM state. 0 = Idle condition must last 1 $\mu$ s 1 = Idle condition must last 4 $\mu$ s	RWS	Yes	0
7:1	<b>Factory Test Only</b>	RWS	Yes	0-0h
8	<b>DLLP Timeout Link Retrain Disable</b> Disable Link retraining when no Data Link Layer Packets (DLLPs) are received for more than 256 $\mu$ s. 0 = Enables Link retraining when no DLLPs are received for more than 256 $\mu$ s (default) 1 = DLLP Timeout is disabled	RWS	Yes	0
9	<b>Factory Test Only</b>	RWS	Yes	0
10	<b>L0s Entry Disable</b> 0 = Enables entry into the L0s Link PM state on the NT Port Link Interface when the L0s idle conditions are met 1 = Disables entry into the L0s Link PM state on the NT Port Link Interface when the L0s idle conditions are met	RWS	Yes	0
11	<b>PHY Upstream Port Control Write Enable</b> Enables the <b>Link Control</b> register <i>Retrain Link</i> and <i>Link Disable</i> bits (NT Port Link Interface, offset 78h[5:4], respectively) to be functional on Upstream Port(s). 0 = Disables 1 = Enables	RW	Yes	0
31:12	<b>Reserved</b>	RsvdP	No	0000_0h



## 16.15 NT Port Link Interface Advanced Error Reporting Extended Capability Registers (Offsets FB4h – FDCh)

The registers detailed in [Section 15.14, “NT Port Virtual Interface Advanced Error Reporting Extended Capability Registers \(Offsets FB4h – FDCh\),”](#) are also applicable to the NT Port Link Interface. [Table 16-20](#) defines the register map for the NT Port.

**Table 16-20. NT Port Link Interface Advanced Error Reporting Extended Capability Register Map**

31 30 29 28 27 26 25 24 23 22 21 20																19 18 17 16				15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																
Next Capability Offset (138h)												Capability Version (1h)				PCI Express Extended Capability ID (0001h)																FB4h				
<i>Reserved</i>						Uncorrectable Error Status																								FB8h						
<i>Reserved</i>						Uncorrectable Error Mask																								FBCCh						
<i>Reserved</i>						Uncorrectable Error Severity																								FC0h						
<i>Reserved</i>												Correctable Error Status																FC4h								
<i>Reserved</i>												Correctable Error Mask																FC8h								
Advanced Error Capabilities and Control																																				FCCh
Header Log 0																																				FD0h
Header Log 1																																				FD4h
Header Log 2																																				FD8h
Header Log 3																																				FDCh

**Register 16-54. FB4h Advanced Error Reporting Extended Capability Header**

Bit(s)	Description	Type	Serial EEPROM and I <sup>2</sup> C	Default
15:0	PCI Express Extended Capability ID	ROS	Yes	0001h
19:16	Capability Version	ROS	Yes	1h
31:20	Next Capability Offset Program to 138h, which addresses the <b>Virtual Channel Extended Capability</b> structure.	ROS	Yes	138h

THIS PAGE INTENTIONALLY LEFT BLANK.



## Chapter 17 Test and Debug

### 17.1 Introduction

This chapter describes the following test- and debug-related information:

- [Physical Layer Loopback Operation](#)
- [Built-In Self-Test](#)
- [Using the SerDes Quad x Diagnostic Data Registers](#)
- [PHY Testability Features](#)
- [JTAG Interface](#)
- [Port Good Status LEDs](#)
- [Thermal Sensor Diode](#)

## 17.2 Physical Layer Loopback Operation

### 17.2.1 Overview

Physical Layer (PHY) Loopback functions are used to test the SerDes in the PEX 8748, connections between devices, and SerDes of external devices, as well as various PEX 8748 and external digital logic. The PEX 8748 supports five types of Loopback operations, as described in [Table 17-1](#). Additional information regarding each type is provided in the sections that follow.

[Figure 17-1](#) illustrates the Physical Media Attachment Layer (PMA) and Physical Coding Sublayer (PCS) Loopback paths. [Table 17-2](#) provides additional Loopback information.

**Table 17-1. Loopback Operations**

Operation	Description
<a href="#">Internal Loopback Mode (LB_NES)</a>	This mode connects the SerDes serial Tx output to the serial Rx input. The PRBS generator can be used to create a pseudo-random data pattern that is transmitted and returned to the PRBS checker.
<a href="#">Analog Loopback Master Mode</a>	This mode depends upon an external device or passive connection ( <i>such as</i> a cable) to loopback the transmitted data to the PEX 8748, without SKIP Ordered-Set clock compensation. If an external device is used, it must not include its Elastic buffer in the Slave Loopback data path, because no SKIP Ordered-Sets are transmitted. The PRBS generator and checker should be used to create and check the data pattern.
<a href="#">Digital Loopback Master Mode</a>	This mode also depends upon an external device to loopback the transmitted data. This method is best used with an external device that includes at least its Elastic buffer in the Loopback data path. The PEX 8748 provides a user-definable Test Pattern generator and checker that inserts SKIP Ordered-Sets at the proper intervals.
<a href="#">Analog Loopback Slave Mode (LB_FEP)</a>	The PEX 8748 enters this mode when an external device transmits Training Sets with the <i>Loopback Training Control Bit Set</i> and the <b>Physical Layer Test 0</b> register <i>Analog Loopback Enable</i> bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <a href="#">224h[10]</a> ) is Set. The PEX 8748 automatically performs the necessary sequence of AHB register Writes, to enable this Loopback data path.
<a href="#">Digital Loopback Slave Mode (LB_PIPE)</a>	<p>The PEX 8748 enters this mode when the following conditions are met:</p> <ul style="list-style-type: none"> <li>• <b>Physical Layer Test 0</b> register <i>Analog Loopback Enable</i> bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <a href="#">224h[10]</a>) is Cleared, and</li> <li>• External device transmits Training Sets with the <i>Loopback Training Control Bit Set</i>, –or–</li> <li>• Port's <b>Physical Layer Test 1</b> register <i>Port x Parallel Loopback Path Enable</i> bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset <a href="#">228h[19:16]</a>) is Set</li> </ul> <p>This loopback path includes the Elastic buffer and 8b/10b (Gen 1 and Gen 2 Link speeds) or 128b/130b (Gen 3 Link speed) encode/decode.</p>

Figure 17-1. PMA and PCS Loopback Paths

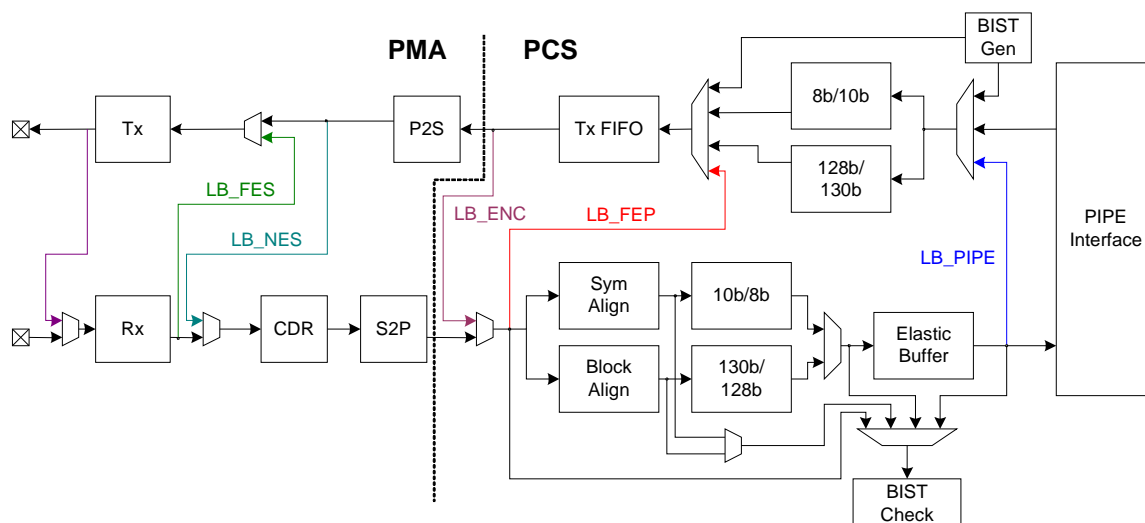


Table 17-2. Loopback

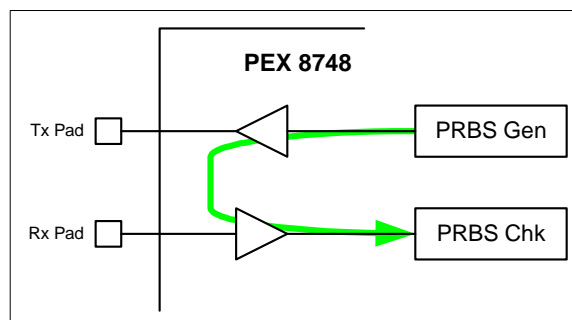
Loopback Name	Loopback Description	AHB Register/PIPE Control	Loopback Direction	Notes
LB_NES	Near-End Serial Loopback	Lane R3[6] = 1 (nes_lb_ena)	Tx -> Rx	Only runs up to 5 GT/s
LB_FES	Far-End Serial Loopback	Lane R1[6] = 1 (fes_lb_ena)	Rx-> Tx	Only runs up to 5 GT/s
LB_ENC	Encoder Loopback	Lane R0[2] = 1 (rx_src) Lane R0[1] = 0 (rx_clk_src)	Tx -> Rx	Requires reset.
LB_FEP	Far-End Parallel Loopback	Lane R1[6:4] = 110b (dmux_txbsel[2:0]) Lane R3[7] = 1 (rxclk_lb_ena) Lane R2[7] = 1 (cktrans_en)	Rx- > Tx	Requires reset and sequencing. Tx FIFO Write clock becomes recovered byte block.
LB_PIPE	PIPE-compliant loopback. Used when the PEX 8748 is a PCI Express Loopback Slave.	Lane R0[7:6] = 01b (dmux_txasel[1:0]) or TxDetRxLpbk && PowerDown = P0	Rx -> Tx	Automatically selected when LTSSM enters LB_ActSlv state.

## 17.2.2 Internal Loopback Mode (LB\_NES)

Figure 17-2 illustrates the Loopback data path when Internal Loopback mode is enabled. The only items in the data path are the SerDes. This Loopback mode is used when the SerDes BIST (Built-in Self Test) is enabled.

SerDes BIST is intended to be overlapped with the serial EEPROM load operation. To achieve this overlap, the **Physical Layer Test 0** register *Factory Test Only* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 224h[11]) is written early in the serial EEPROM load operation. After the bit is Set, the SerDes are placed into Loopback mode, and the PRBS generator is started. The BIST is run for 15 ms, if an error is detected on any of the SerDes, the BIST\_ERROR pin associated with the Station that includes the SerDes in error is asserted. While the SerDes BIST is in progress, the PRBS test data is present on the external PEX\_PER/PETp/nx balls. The continuing serial EEPROM register load has no effect on SerDes BIST.

**Figure 17-2. Internal Loop Back (Analog Near End) Data Path**



### 17.2.3 Analog Loopback Master Mode

Analog Loopback Master mode is typically used for Analog Far-End testing (refer to [Figure 17-3](#)), with a shallow Loopback path Slave device, to determine overall Bit Error rates. However, it can also be used to recreate the BIST described in [Section 17.2.2](#), by looping back the data with a cable. Looping back with a cable includes the internal circuitry, package connections to bond pads, package balls, board traces, and any connectors that might be in the test data path, as illustrated in [Figure 17-4](#). A PRBS pattern is typically used for this mode, because it is appropriate for bit error rate testing. A User Test Pattern (UTP) is *not* recommended for this application.

Figure 17-3. Analog Far-End Loopback

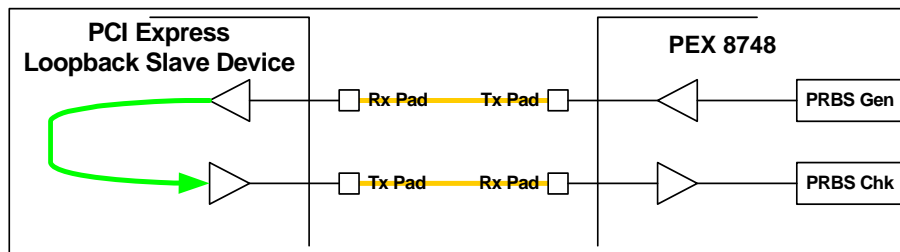
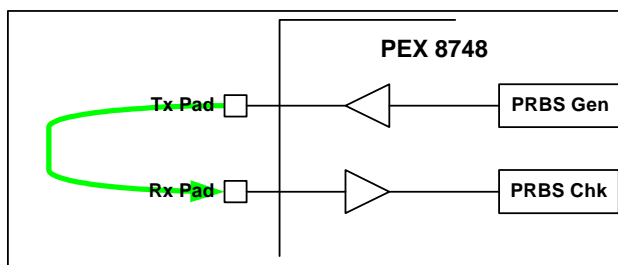


Figure 17-4. Cable Loopback



To cause a PEX 8748 Port to request to become a Loopback Master, the following must occur: After the Link is Up, a Configuration Write to the Port's **Physical Layer Port Command** register *Port x Loopback Command* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 230h[0, 4, 8, or 12]) causes the Port to transition from the L0 Link PM state to the *Recovery* state, and then to the *Loopback* state. If a cable is used for the Loopback, the Port transitions from the *Configuration* state to the *Loopback* state. The cable should be connected only after the Upstream Link is Up and Configuration Writes are possible. If the cable is connected before the Upstream device is able to Set the *Port x Loopback Command* bit, the Link with the cable can reach the L0 Link PM state and **not** go to the *Loopback* state. In this case, the Port must be forced to the *Recovery* state, by asserting the Port's **Link Control** register *Retrain Link* bit (Downstream Ports, offset 78h[5]). The cable length is limited only by the PCI Express drivers and cable properties.

After the Port is in the *Loopback* state the Port's **Physical Layer Port Command** register *Port x Ready as Loopback Master* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 230h[3, 7, 11, or 15]) is Set. At this time, the user can enable the PRBS engine, by Setting the *PRBS Enable* bit that is associated with the SerDes assigned to the Port being tested. The PRBS checker checks the returned PRBS data. Any errors are logged in the **SerDes Quad x Diagnostic Data** register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offsets 238h through 244h) that corresponds to the SerDes quad being tested.



## 17.2.4 Digital Loopback Master Mode

The only difference between Analog and Digital Loopback Master modes is that the external device is assumed to have some of its digital logic in the Loopback data path. Because this includes the Elastic buffer, SKIP Ordered-Sets must be included in the test data pattern, which precludes use of the PRBS engine.

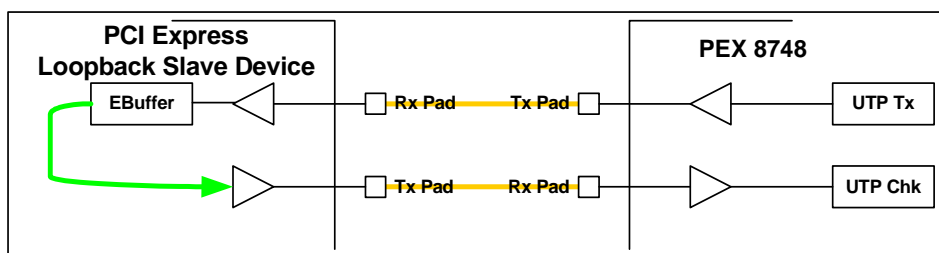
Figure 17-5 illustrates a Far-End Digital Loopback Master connection and data path.

The PEX 8748 provides the User Data Pattern Transmitter for Digital Far-End Loopback testing. After the Loopback Master mode is established, Configuration Writes are used to fill the Test Data Pattern registers. One or more **Physical Layer Test 1** register *SerDes Quad x Pair x User Test Pattern Enable* bits (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 228h[31:24]) is Set, which starts the transmission of the User Data Pattern on all Lanes. If the Port's *Enable by Port* bit is also Set, the test pattern is transmitted on all Lanes of the corresponding Port, regardless of its Link width. If the *Enable by Port* bit is Cleared, the test pattern is transmitted only on the Lanes of the corresponding SerDes quad. SKIP Ordered-Sets are inserted at the interval determined by the **SKIP Ordered-Set Interval** register *SKIP Ordered-Set Interval in Blocks* or *SKIP Ordered-Set Interval in Clocks* field (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 234h[27:16 or 11:0], respectively) value, at the nearest data pattern boundary:

- **Gen 3 (8.0 GT/s)** – Field [27:16]. Default interval is 371 block times.
- **Gen 1 and Gen 2 (2.5 and 5.0 GT/s, respectively)** – Field [11:0]. Default interval is 1,180 symbol times.

The Test Pattern checker ignores SKIP Ordered-Sets that are returned by the Loopback Slave, because the quantity of SKIP symbols received can be different than the quantity transmitted. All other data is compared to the transmitted data. Any errors are logged in the **SerDes Quad x Diagnostic Data** register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offsets 238h through 244h) that corresponds to the SerDes quad being tested.

Figure 17-5. Digital Far-End Loopback

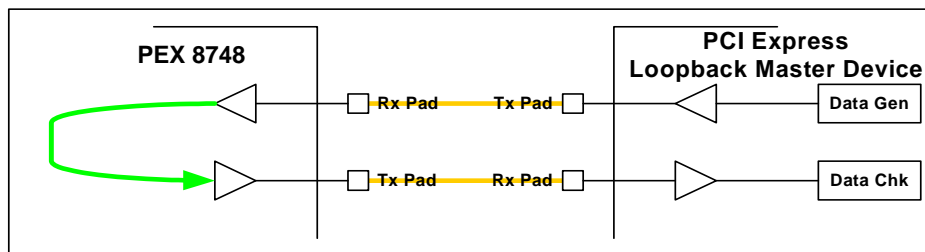


## 17.2.5 Analog Loopback Slave Mode (LB\_FEP)

The PEX 8748 becomes an Analog Loopback Slave (as illustrated in Figure 17-6) if it receives Training Sets with the *Loopback* Training Control Bit Set while the **Physical Layer Test 0** register *Analog Loopback Enable* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 224h[10]) is Set.

While an Analog Loopback Slave, the PEX 8748 includes only the SerDes in the Loopback data path. The Loopback Master must provide the test data pattern and data pattern checking. It is not necessary for the Loopback Master to include SKIP Ordered-Sets in the data pattern.

**Figure 17-6. Analog Loopback Slave Mode**



## 17.2.6 Digital Loopback Slave Mode (LB\_PIPE)

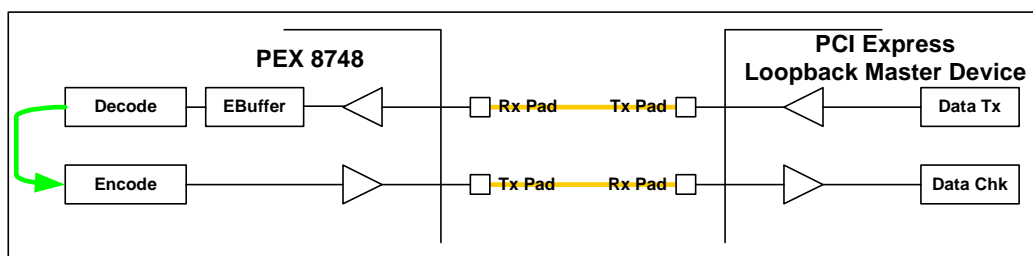
The PEX 8748 becomes a Digital Loopback Slave if it receives Training Sets with the *Loopback Training Control Bit Set*, –or– the Port’s **Physical Layer Test 1** register *Port x Parallel Loopback Path Enable* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 228h[19:16])) is Set, while the **Physical Layer Test 0** register *Analog Loopback Enable* bit (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset 224h[10]) is Cleared.

When a PEX 8748 Port is a Digital Loopback Slave:

- **Gen 1 and Gen 2 Link speeds (2.5 and 5.0 GT/s, respectively)** – Loopback data path includes the Elastic buffer and symbol alignment logic
- **Gen 3 Link speed (8.0 GT/s)** – Loopback data path includes the Elastic buffer and block aligner, as well as the 128b/130b encoder and decoder

The Loopback Master must provide the test data pattern and data pattern checker (*such as* a PEX 8748 User Test Pattern). The Loopback Master must also transmit SKIP Ordered-Sets with the data pattern. The PEX 8748 can return more or fewer SKIP symbols than the switch received from the Master. Therefore, the Master’s data pattern checker must make provisions for this when decoding for errors.

**Figure 17-7. Digital Loopback Slave Mode**



**Note:** 8b/10b (Gen 1 and Gen 2 Link speeds) or 128b/130b (Gen 3 Link speed) encode/decode.

## 17.3 Built-In Self-Test

Each Lane contains a Built-In Self-Test (BIST) generator and checker.

## 17.4 Using the SerDes Quad x Diagnostic Data Registers

Each SerDes quad has its own Diagnostic Data register, per Station. The **SerDes Quad x Diagnostic Data** register (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offsets [238h](#) through [244h](#)) contents reflect the performance of the SerDes selected by the registers' *SerDes Diagnostic Data Select* field [25:24]. This control is specific to this register (which reports results of UTP tests).

When field [25:24] is Cleared, the information in that Diagnostic Data register is for the first SerDes within that SerDes quad. When field [25:24] is programmed to 01b, the information in that Diagnostic Data register is for the second SerDes within that SerDes quad, as illustrated in [Table 17-3](#). Following this pattern, a value of 10b indicates the third SerDes within that SerDes quad, and a value of 11b indicates the fourth SerDes within that SerDes quad.

**Table 17-3. SerDes Quad x Diagnostic Data Register Contents When *SerDes Diagnostic Data Select* Field [25:24]=01b (Base mode – Ports 0, 8, and 16; Virtual Switch mode – Ports 0, 8, and 16, accessible through the Management Port), by Station Port**

Offset	Register	Port 0	Port 8	Port 16
<a href="#">238h</a>	<a href="#">SerDes Quad 0 Diagnostic Data</a>	SerDes 1	SerDes 17	SerDes 33
<a href="#">23Ch</a>	<a href="#">SerDes Quad 1 Diagnostic Data</a>	SerDes 5	SerDes 21	SerDes 37
<a href="#">240h</a>	<a href="#">SerDes Quad 2 Diagnostic Data</a>	SerDes 9	SerDes 25	SerDes 41
<a href="#">244h</a>	<a href="#">SerDes Quad 3 Diagnostic Data</a>	SerDes 13	SerDes 29	SerDes 45

## 17.5 PHY Testability Features

The PEX 8748 includes several Configuration bits to ease PHY testability. Features include:

- Full support of the standard and modified Compliance patterns
- Register controllability of the common block and Lane-specific inputs of the SerDes

Table 17-4 describes the Configuration bits.

**Table 17-4. Configuration Bits to Ease PHY Testability**

Register Bit(s) <sup>a</sup>	Description
<i>SerDes x Mask Electrical Idle Detect</i> <b>Physical Layer Electrical Idle Detect Mask</b> register (offset 204h[15:0])	<b>Never Detect Electrical Idle Mask.</b> When any one of these bits is Set, the Lane's Electrical Idle condition flag does not assert, regardless of the actual presence of Electrical Idle.
<i>SerDes x Mask Receiver Not Detected</i> <b>Physical Layer Receiver Not Detected Mask</b> register (offset 204h[31:16])	<b>Always Detect a Receiver Mask.</b> When any one of these bits is Set, the PHY functions as if the Lane detected a Receiver, regardless of the actual presence of a Receiver.
<i>User Test Pattern</i> <b>Physical Layer User Test Pattern, Bytes x through y</b> registers (offsets 20Ch through 218h)	A 16-byte test pattern can be written to these four registers. When UTP transmission is enabled, Byte 0 of register offset 20Ch is transmitted first and Byte 3 of register offset 218h is transmitted last. (Refer to Section 17.2.4 for further details.) Every byte of the UTP can be a Control or Data character. Illegal Control characters can be specified.
<i>Port x Scrambler Disable Command</i> <b>Physical Layer Port Command</b> register (offset 230h[1, 5, 9, or 13])	<b>Scramble Disable.</b> Unconditionally disables the data scramblers on the Port's Lane(s), and causes the <i>Scramble Disable</i> Training Control Bit to be Set in transmitted Training Sets. There is one bit, per Port, per Station.
<i>Disable Port x</i> <b>Port Control</b> register (offset 208h[3:0])	<b>Unconditional Port Disable.</b> When Set, the LTSSM remains in the <i>Detect.Quiet</i> substate on the Port if it is currently in, or returns to, that substate. Unconditionally disables the Port. This is different from the LTSSM <i>Disabled</i> state, in that the Port does not attempt to enter this state. If the Port is idle, it ceases attempting to detect a Receiver. If the Port is up, it immediately returns to the <i>Detect.Quiet</i> substate and remains there. No Electrical Idle Ordered-Set (EIOS) is sent, which could force any connected device to the <i>Recovery</i> state, and then to an LTSSM <i>Detect</i> state. The Port remains disabled until its <i>Disable Port x</i> bit is Cleared. While the Port is disabled, Receiver termination is disabled and the SerDes that belong to the disabled Port are placed into the L1 Link PM state.
<i>Hold Port x Quiet</i> <b>Port Control</b> register (offset 208h[11:8])	<b>Port Quiet.</b> When Set, the LTSSM remains in the <i>Detect.Quiet</i> substate if it is currently in, or returns to, that substate. These bits do not force the LTSSM to the <i>Detect.Quiet</i> substate. (Refer to the <i>Disable Port x</i> bits.) After entering the <i>Detect.Quiet</i> substate, Receiver termination is enabled and the Transmitters are placed into the L0 Link PM state. This Port can now transmit test patterns (PRBS or UTP), with or without an attached device and without being in the <i>Loopback.Active</i> substate.
<i>Port x Bypass UTP Alignment Pattern</i> <b>Port Control</b> register (offset 208h[19:16])	<b>Bypass Alignment Pattern.</b> When Cleared, the UTP Transmitter continuously transmits the UTP alignment pattern (K28.5 D3.2 D18.2 D13.2), to allow the corresponding symbol alignment logic to find the symbol boundaries before sending the sync pattern (D3.2 D18.2 D13.2 D17.1) and programmed UTP.  When Set, the UTP transmitter sends one UTP alignment pattern (K28.5 D3.2 D18.2 D13.2), one sync pattern (D3.2 D18.2 D13.2 D17.1), and then the programmed UTP.

**Table 17-4. Configuration Bits to Ease PHY Testability (Cont.)**

Register Bit(s) <sup>a</sup>	Description
<i>Port x Test Pattern x Rate</i>  <b>Port Control</b> register (offset 208h[25:24])	<b>Test Pattern Rate.</b> The Port transmits the selected test pattern (PRBS or UTP) at 8.0 GT/s, if the Port's <b>Port Control</b> register <i>Hold Port x Quiet</i> bit (offset 208h[11:8]) is also Set (manual rate selection is enabled only when the <i>Hold Port x Quiet</i> bit is Set).
<b>Physical Layer Error Injection Control</b> register (offset 25Ch)	<b>1-Bit Error Injection.</b> Provides Port-specific controls for 1-bit error injection from the PEX 8748 Transmitters. Far-end devices can use this feature to test their error-handling capability.
<i>x1 Only for Station x</i>  <b>x1 Port Configuration</b> register (offset 304h[23:0])	<b>x1 Only.</b> Forces the Port to linkup with a x1 negotiated Link width, regardless of the quantity of Lanes connected to the Port.
<i>x2 Only for Station x</i>  <b>x2 Port Configuration</b> register (offset 308h[23:8])	<b>x2 Only.</b> Forces the Port to linkup with a x2 negotiated Link width, regardless of the quantity of Lanes connected to the Port.
<i>Port x Receiver Error Counter</i>  <b>Port Receiver Error Counter</b> register (offset BF0h[31:0])	<b>Receiver Error Counter.</b> Contains four 8-bit fields that, when read, return the quantity of Receiver errors that the Port detected. The Error Counter saturates at 255.  The Counter is Cleared with any Write to the corresponding byte in this register; otherwise, this register is RO.

- a. All register bits listed in this table are located, as follows:  
Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port.

## 17.6 JTAG Interface

The PEX 8748 provides a Joint Test Action Group (JTAG) Boundary Scan interface, which is used to debug board connectivity for each ball.

### 17.6.1 *IEEE 1149.1* and *IEEE 1149.6* Test Access Port

The *IEEE Standard 1149.1* Test Access Port (TAP), commonly called the *JTAG Debug Port*, is an architectural standard described in the *IEEE Standard 1149.1-1990*. The *IEEE Standard 1149.6-2003* defines extensions to *1149.1* to support PCI Express SerDes testing. These standards describe methods for accessing internal device facilities, using a four- or five-signal interface.

The JTAG Debug Port, originally designed to support scan-based board testing, is enhanced to support the attachment of debug tools. The enhancements, which comply with the *IEEE Standard 1149.1-1994 Specifications for Vendor-Specific Extensions*, are compatible with standard JTAG hardware for boundary-scan system testing.

- **JTAG Signals** – JTAG Debug Port implements the four required JTAG signals – [JTAG\\_TCK](#), [JTAG\\_TDI](#), [JTAG\\_TDO](#), [JTAG\\_TMS](#) – and optional [JTAG\\_TRST#](#) signal
- **Clock Requirements** – JTAG\_TCK signal frequency ranges from 0 to 15 MHz
- **JTAG Reset Requirements** – Refer to [Section 17.6.4](#)

## 17.6.2 JTAG Instructions

The JTAG Debug Port provides the *IEEE Standard 1149.1-1990* BYPASS, EXTEST, SAMPLE, PRELOAD, CLAMP, and IDCODE instructions. *IEEE Standard 1149.6-2003* EXTEST\_PULSE and EXTEST\_TRAIN instructions are also supported. [Table 17-5](#) lists the JTAG instructions, along with their input codes.

The PEX 8748 returns the JTAG IDCODE values listed in [Table 17-6](#).

**Table 17-5. JTAG Instructions**

Instruction	Input Code	Comments
BYPASS	7EFFh	<i>IEEE Standard 1149.1-1990</i>
EXTEST	7FDFh	
SAMPLE	7FCFh	
PRELOAD	7FCFh	
EXTEST_PULSE	7F9Fh	<i>IEEE Standard 1149.6-2003</i>
EXTEST_TRAIN	7F5Fh	
CLAMP	7FEFh	<i>IEEE Standard 1149.1-1990</i>
IDCODE	1FFFh	

**Table 17-6. JTAG IDCODE Values**

Silicon Revision	Units	Version	Part Number	PLX Manufacturer Identity	Least Significant Bit
BA	Bits	0001b	1000_0111_0100_1000b	001_1100_1101b	1
	Hex	1h	8748h	1CDh	1h
	Decimal	1	34632	461	1
CA	Bits	0010b	1000_0111_0100_1000b	001_1100_1101b	1
	Hex	2h	8748h	1CDh	1h
	Decimal	2	34632	461	1



### 17.6.3 JTAG Boundary Scan

*Boundary Scan Description Language (BSDL)*, IEEE Standard 1149.1-1994, is a supplement to the IEEE Standard 1149.1-1990 and IEEE Standard 1149.1a-1993, *IEEE Standard Test Access Port and Boundary-Scan Architecture*. BSDL, a subset of the IEEE 1076-1993 *Standard VHSIC Hardware Description Language (VHDL)*, allows a rigorous description of testability features in components which comply with the standard. This standard is used by automated test pattern generation tools for package interconnect tests and Electronic Design Automation (EDA) tools for synthesized test logic and verification. BSDL supports robust extensions that can be used for internal test generation and to write software for hardware debug and diagnostics.

The primary components of BSDL include the logical Port description, physical ball map, instruction set, and **Boundary** register description.

The logical Port description assigns symbolic names to the device's signal balls. Each ball includes a logical type of *in*, *out*, *in out*, *buffer*, or *linkage* that defines the logical direction of signal flow.

The physical ball map correlates the device's logical Ports to the physical balls of a specific package. A BSDL description can include several physical ball maps, and maps are provided with a unique name.

Instruction set statements describe the bit patterns that must be shifted into the **Instruction** register to place the device in the various test modes defined by the standard. Instruction set statements also support descriptions of instructions that are unique to the PEX 8748.

The **Boundary** register description lists each cell or shift stage of the **Boundary** register. Each cell has a unique number – the cell numbered 0 is the closest to the **JTAG Test Data Output (JTAG\_TDO)**, and the cell with the highest number is closest to the **JTAG Test Data Input (JTAG\_TDI)**. Each cell includes additional information, *such as*:

- Cell type
- Logical Port associated with the cell
- Logical function of the cell
- Safe value
- Control cell number
- Disable value
- Result value

### 17.6.4 JTAG Reset Input – JTAG\_TRST#

The **JTAG\_TRST#** input is the asynchronous reset input to the JTAG TAP Controller logic. JTAG\_TRST# assertion places the PEX 8748's TAP Controller into the *Test-Logic-Reset* state, which selects the PEX 8748 standard logic path (core-to-I/O), as required for typical functionality of the switch. JTAG\_TRST# de-assertion enables the TAP Controller for test and debug functionality, such as boundary scan.

- If JTAG functionality is required, the JTAG\_TRST# input should be transitioned from Low-to-High once during PEX 8748 bootup, along with **PEX\_PERST#** and/or **VSx\_PERST#** de-assertion. In any JTAG state, the JTAG TAP Controller can be returned to the *Test-Logic-Reset* (initial) state, by holding the **JTAG\_TMS** input High while clocking the **JTAG\_TCK** input at least five times.
- If the design is not going to use the JTAG TAP Controller, an external pull-down resistor is recommended on the JTAG\_TRST# input, because its internal pull-down resistor is weak.

## 17.7 Port Good Status LEDs

The PEX 8748 provides Port Good outputs, PORT\_GOOD[11:0]# (PORT\_GOODx#), that can be used to control external circuitry, *such as* LEDs, to provide visual indication that the PHY of that Port's Link is trained to at least x1 width. These outputs can:

- Default to the PORT\_GOOD output function, when the STRAP\_TESTMODE[2:0] 3-state inputs select the PORT\_GOOD output function (*that is*, Test modes 0, 4, or 8, and additionally for PORT\_GOOD0# only, in Test modes 3, 7, and 11, as well as PORT\_GOOD11#, in Test mode 11; refer to [Section 3.4.4.1, “STRAP\\_TESTMODE\[2:0\] 3-State Input Configurations,”](#) for details), –or–
- Be programmed as a general-purpose I/O, to assume the PORT\_GOOD output function (refer to the **GPIO 0\_x Direction Control** register *Direction Control* bit(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset(s) [600h](#) and/or [604h](#)))

The PORT\_GOODx#/GPIOx I/Os operate from the [VDD18](#) supply, and the maximum input voltage is 2.5V. Because the LED forward voltage drop is typically greater than 1.8V, the LED voltage source will likely be higher than 2.5V. In this case, if the LED is connected directly to the PORT\_GOODx# input, the input voltage level (while the LED is turned Off) would exceed the specification. Therefore, a PORT\_GOODx# signal typically cannot drive an LED directly; however, it can be used to gate a transistor that switches the LED On and Off.

**Notes:** PORT\_GOOD[11:0]# correspond to Ports 19, 18, 17, 16, 11, 10, 9, 8, 3, 2, 1, and 0, respectively. PORT\_GOODx# signals can be re-assigned to any Port, however, by programming the **Port Good Selector x** register(s) (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port), offset(s) [6A0h](#) through [6A8h](#)).

The PORT\_GOOD11#/GPIO11 signal exists only in Virtual Switch mode, when four, five, or six virtual switches are enabled.

[Table 4-1](#) indicates which Ports are enabled/used, and when, based upon the STRAP\_STNx\_PORTCFGx input states and/or serial EEPROM programming.

[Table 17-7](#) describes the LED On/Off patterns when connected to the PORT\_GOODx# signals.

**Table 17-7. PORT\_GOODx# LED On/Off Patterns, by State**

State	LED Pattern
Link is Down	Off
Link is Up, 2.5 GT/s, any negotiated Link width	Blinking, 512 ms On, 512 ms Off (1 Hz)
Link is Up, 5.0 GT/s, any negotiated Link width	Blinking, 256 ms On, 256 ms Off (2 Hz)
Link is Up, 8.0 GT/s, any negotiated Link width	On

Software can determine:

- Which Lanes have completed PHY linkup, by performing a Memory Read of the **Station x Lane Status** register(s) *Lane x Up Status* bits (Base mode – Port 0; Virtual Switch mode – Port 0, accessible through the Management Port, offset(s) 330h[31:0] and 334h[15:0]).
- Whether the Link for each Port has trained, by reading either the Port's **Link Status** register *Data Link Layer Link Active* bit (Downstream Ports, offset 78h[29]) or **VC0 Resource Status** register *VC0 Negotiation Pending* bit (offset 160h[17]). If the *Data Link Layer Link Active* bit is Set, or *VC0 Negotiation Pending* bit is Cleared, the Link has completed Flow Control (FC) initialization.

The **Link Status** register can be read by either a PCI Express Configuration Request or Memory Read. The **VC0 Resource Status** register can be read by either a PCI Express Enhanced Configuration access or Memory Read.

- The negotiated Link width of each Port, by reading the Port's **Link Status** register *Negotiated Link Width* field (offset 78h[25:20]). This register can be read by either a Configuration Request or Memory Read.

## 17.8 Thermal Sensor Diode

The PEX 8748 includes a thermal-sensing diode, to facilitate the measurement of on-die device junction temperature. On the package level, there are two external inputs – [THERMAL\\_DIODEn](#) and [THERMAL\\_DIODEp](#), that connect to anode and cathode terminals of a diode-connected transistor (PNP) implemented on the die.

At the board level, a temperature sensor chip, *such as* the Maxim MAX6657<sup>a</sup>, can be used.

For further details, refer to the manufacturer's data sheet for the device being used.

**Note:** *The PEX 8748 thermal diode produces an offset in the external temperature sensor that results in a temperature reading that is 24°C above the approximate die temperature. Consequently, a temperature sensor chip that measures a maximum temperature of 128°C might not be sufficient, because the maximum temperature reported is effectively reduced to 104°C (128 - 24 = 104). Some sensor ICs support higher temperatures, such as 145°C; however, these devices have not been tested with the PEX 8748 thermal diode.*

---

a. *The MAX6657, MAX6658, and MAX6659 are ±1°C, SMBus-Compatible Remote/Local Temperature Sensors with Overtemperature Alarms,” by Maxim Integrated Products.*



## Chapter 18 Electrical Specifications

### 18.1 Introduction

This chapter provides the PEX 8748 electrical specifications:

- [Power-Up/Power-Down Sequence](#)
- [Absolute Maximum Ratings](#)
- [Power Characteristics](#)
- [Power Consumption Estimates](#)
- [I/O Interface Signal Groupings](#)
- [Transmit Characteristics](#)
- [Receive Characteristics](#)

## 18.2 Power-Up/Power-Down Sequence

The PEX 8748 does not have power sequencing requirements. The power rails can be powered up and powered down, in any sequence.

## 18.3 Absolute Maximum Ratings

**Warning:** *Maximum limits indicate the temperatures and voltages above which permanent damage can occur. Proper operation at these conditions is not guaranteed, and continuous operation of the PEX 8748 at these limits is not recommended.*

**Table 18-1. Absolute Maximum Rating (All Voltages Referenced to VSS System Ground)**

Item	Symbol	Absolute Maximum Rating	Units
I/O Interface Supply Voltage	VDD18	1.98	V
PLL Supply Voltage	VDD18A	1.98	V
Core (Logic) Supply Voltage	VDD09	0.99	V
SerDes Analog Supply Voltage	VDD09A	0.99 <sup>a</sup>	V
Input Voltage (1.8V Interface)	V <sub>I</sub>	2.75	V
Operating Ambient Temperature (Commercial)	T <sub>A</sub>	0 to +70	°C
Storage Temperature	T <sub>STG</sub>	-65 to +150	°C

a. Technically 1.1V; however, most users tie VDD09A to 0.99V.

## 18.4 Power Characteristics

### 18.5 Power Consumption Estimates

**Table 18-2. Operating Condition Power Supply Rails**

Symbol	Parameter	Voltage and Ranges	Min	Typ	Max	Units
VDD09	Digital Core Supply	0.9V -5%, +5.55%	0.855	0.9	0.95	V
VDD09A	Analog SerDes Supply	0.9V -5%, +5.55%	0.855	0.9	0.95	V
VDD18 <sup>a</sup>	I/O Supply	1.8V -5%, +10%	1.71	1.8	1.98	V
VDD18A	Phase-Locked Loop (PLL) Supply	1.8V -5%, +10%	1.71	1.8	1.98	V

a. The PEX 8748 I/Os are 2.5V-tolerant.

**Table 18-3. Power Consumption Estimates (Watts)**

Lanes	Ports	Link Speed (GT/s)	Digital (VDD09)		SerDes Analog (VDD09A)		PLL and I/O (VDD18A + VDD18)		Total	
			Typ	Max	Typ	Max	Typ	Max	Typ <sup>a</sup>	Max <sup>b, c</sup>
48	12	2.5	3.32	8.45	2.09	3.25	0.83	0.98	6.24	12.69
		5.0	3.69	9.19	2.09	3.41	0.83	0.98	6.60	13.58
		8.0	3.81	9.90	2.70	4.13	0.83	0.98	7.34	15.02

a. Typical power based upon 35% traffic, idle Lanes in active L0s PM state, typical power rails (0.9V/1.8V), at 25°C.

b. Maximum power based upon 85% traffic, idle Lanes in active L0s Link PM state, maximum power rails (0.95V/1.98V).

c. Maximum power is at 110°C Junction temperature and Fast/Fast (FF) process corner silicon.

## 18.6 I/O Interface Signal Groupings

**Table 18-4. Signal Group PCI Express Analog Interface**

Signal Group	Signal Type	Signals	Notes
(a)	PCI Express Output (Transmit)	PEX_PETp/nx	Refer to <a href="#">Table 18-6</a> and <a href="#">Table 18-7</a>
(b)	PCI Express Input (Receive)	PEX_PERp/nx	Refer to <a href="#">Table 18-6</a> and <a href="#">Table 18-12</a>
(c)	PCI Express Differential Clock Input	PEX_REFCLKp/n	Refer to <a href="#">Table 18-6</a>
(d)	SerDes External Bias Resistor	REXT_Ax, REXT_Bx, REXT_CMU	3.00K $\Omega$ , $\pm 1\%$ , and refer to <a href="#">Table 18-6</a>

**Table 18-5. Signal Group Digital Interface**

Signal Group	Signal Type	Signals	Note
(e)	Digital Input	STRAP_RESERVEDx <sup>a</sup> , THERMAL_DIODEn/p	Refer to <a href="#">Table 18-6</a>
(f)	Digital Input with Internal Pull-Up Resistor	PEX_NT_PERST#, PEX_PERST#, STRAP_PLL_BYPASS#, STRAP_PROBE_MODE#, STRAP_SERDES_MODE_EN#	
(g)	Digital Input with Internal Pull-Down Resistor	JTAG_TCK, JTAG_TDI, JTAG_TMS, JTAG_TRST#	
(h)	Bidirectional with Internal Pull-Up Resistor (8 mA Drive)	EE_CS#, EE_DO, EE_SK, HP_ATNLED_x#, HP_BUTTON_x#, HP_CLKEN_x#, HP_MRL_x#, HP_PERST_x#, HP_PRSTNT_x#, HP_PWRFLT_x#, HP_PWRLED_x#, PORT_GOOD[11:9, 6:0]#, SHPC_INT#, SPARE[2:0], STRAP_I2C_SMBUS_CFG_EN#, VS0_FATAL_ERR#, VS1_FATAL_ERR#, VS3_FATAL_ERR#, VS4_FATAL_ERR#, VSx_PERST#, VSx_PEX_INTA#	
(j)	Bidirectional with Internal Pull-Down Resistor (8 mA Drive)	HP_PWREN_x, HP_PWR_GOOD_x, PORT_GOOD[8:7]#, SPARE3, VS2_FATAL_ERR#, VS5_FATAL_ERR#	
(k)	Bidirectional Tri-Statable Output with Limited Slew Rate Input	I2C_SCL0, I2C_SCL1, I2C_SDA0, I2C_SDA1, PEX_INTA#	
(m)	Bidirectional Tri-Statable Output	EE_DI, FATAL_ERR#, JTAG_TDO	
(n)	Bidirectional with 3-State Input	I2C_ADDR0, STRAP_GEN1_GEN2, STRAP_I2C_SMBUS_EN, STRAP_MGMT_PORT_EN, STRAP_NT_UPSTRM_PORTSEL[2:0], STRAP_STN0_PORTCFG[1:0], STRAP_STN1_PORTCFG[1:0], STRAP_STN2_PORTCFG[1:0], STRAP_TESTMODE[2:0], STRAP_UPSTRM_PORTSEL[2:0], VS_MODE	

- a. These signals must be pulled or tied High to VDD18 or Low to VSS (Ground). Refer to [Table 3-8](#), “Factory Test Only STRAP\_RESERVED[16, 4:1] Input External Pull-Up/Pull-Down Resistor Requirements,” for specific requirements.

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved



**Table 18-6. Analog and Digital Interfaces (All Signal Groups) – DC Electrical Characteristics**

Symbol	Signal Group	Parameter	Min	Typ	Max	Unit	Conditions
$I_{OL}$	(h) (k)	Output Low Current	10.4	17.9	26.5	mA	$V_{OLmax} = 0.45V$
$I_{OH}$	(h)	Output High Current	9.0	20.2	36.5	mA	$V_{OHmin} = 1.35V$
$V_{IL}$	(e) (f) (g) (h) (k)	Input Low Voltage	-0.3		0.63	V	
$V_{IH}$	(e) (f) (g) (h) (k)	Input High Voltage	1.17		2.75	V	
$V_T$	(e) (g) (h) (k)	Threshold Point	0.78	0.84	0.89	V	
$C_{PIN}$	(a) (b) (c) (d) (e) (f) (g) (h) (j) (k)	Ball Capacitance			5	pF	
$I_{LEAKAGE}$	(m)	Tri-State Leakage			$\pm 10$	$\mu A$	
	(e) (f)	Input Leakage			$\pm 10$	$\mu A$	
$R_{PU}$	(f) (h)	Pull-Up Impedance	38K	57K	92K	$\Omega$	
$R_{PD}$	(g) (j)	Pull-Down Impedance	37K	53K	99K	$\Omega$	

## 18.7 Transmit Characteristics

### 18.7.1 PCI Express Transmitter AC and DC Characteristics

Table 18-7. PCI Express Transmitter (Signal Group a) – AC and DC Characteristics

Symbol	Parameter	Link Speed			Units	Comments
		2.5 GT/s	5.0 GT/s	8.0 GT/s		
UI	Unit Interval	399.88 (min) 400.12 (max)	199.94 (min) 200.06 (max)	124.9625 (min) 125.0375 (max)	ps	The specified UI is equivalent to a tolerance of $\pm 300$ ppm for each REFCLK source. Period does not account for Spread-Spectrum Clock (SSC)-induced variations. Refer to Note 1.
BW <sub>TX-PLL</sub>	Tx PLL Bandwidth for 2.5 GT/s	1.5 (min) 22 (max)	Not specified	Not specified	MHz	Refer to Note 6.
BW <sub>TX-PKG-PLL1</sub>	Tx PLL Bandwidth corresponding to PKG <sub>TX-PLL1</sub>	Not specified	8 (min) 16 (max)	2 (min) 4 (max)	MHz	Second Order PLL Jitter Transfer Bounding function. Refer to Note 6.
BW <sub>TX-PKG-PLL2</sub>	Tx PLL Bandwidth corresponding to PKG <sub>TX-PLL2</sub>	Not specified	5 (min) 16 (max)	2 (min) 5 (max)	MHz	Second Order PLL Jitter Transfer Bounding function. Refer to Note 6.
PKG <sub>TX-PLL1</sub>	Tx PLL Peaking	Not specified	3.0 (max)	2.0 (max)	dB	PLL Bandwidth = 8 MHz (min) at 5 GT/s or Bandwidth = 4 MHz (max) at 8 GT/s. Refer to Notes 6 and 8.
PKG <sub>TX-PLL2</sub>	Tx PLL Peaking	Not specified	1.0 (max)	1.0 (max)	dB	PLL Bandwidth = 5 MHz (min) at 5 GT/s or Bandwidth = 5 MHz (max) at 8 GT/s. Refer to Note 8.
V <sub>TX-DIFF-PP</sub>	Differential Peak-to-Peak Tx Voltage Swing	0.8 (min) 1.2 (max)	0.8 (min) 1.2 (max)	Refer to Table 18-9	VPP	Measured with compliance test load. Defined as $2 \times  V_{TX-D+} - V_{TX-D-} $
V <sub>TX-DIFF-PP-LOW</sub>	Low Power Differential Peak-to-Peak Tx Voltage Swing	0.4 (min) 1.2 (max)	0.4 (min) 1.2 (max)	Refer to Table 18-9	VPP	Measured with compliance test load. Defined as $2 \times  V_{TX-D+} - V_{TX-D-} $ . Refer to Note 9.
V <sub>TX-DE-RATIO-3.5dB</sub>	Tx De-Emphasis Level Ratio	3.0 (min) 4.0 (max)	3.0 (min) 4.0 (max)	Refer to Table 18-9	dB	
V <sub>TX-DE-RATIO-6dB</sub>	Tx De-Emphasis Level Ratio	–	5.5 (min) 6.5 (max)	Refer to Table 18-9	dB	
T <sub>MIN-PULSE</sub>	Instantaneous Lone Pulse Width	Not specified	0.9 (min)	Refer to Table 18-9	UI	Measured relative to rising/falling pulse. Refer to Notes 2 and 10.

NDA CONFIDENTIAL -- Inventec | King Michael

ExpressLane PEX 8748-BA/CA 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Data Book, v1.1

Copyright © 2012 by PLX Technology, Inc. All Rights Reserved

**Table 18-7. PCI Express Transmitter (Signal Group a) – AC and DC Characteristics (Cont.)**

Symbol	Parameter	Link Speed			Units	Comments
		2.5 GT/s	5.0 GT/s	8.0 GT/s		
$T_{TX-EYE}$	Tx Eye Width (including all jitter sources)	0.75 (min)	0.75 (min)	Refer to Table 18-9	UI	Does not include SSC nor REFCLK jitter. Includes $R_j$ at $10^{-12}$ . Refer to Notes 2, 3, 4, and 10. <b>Note:</b> 2.5 and 5.0 GT/s use different jitter determination methods.
$T_{TX-EYE-MEDIAN-to-MAX-JITTER}$	Maximum Time between the Jitter Median and Maximum Deviation from the Median	0.125 (max)	Not specified	Not specified	UI	Measured differentially at zero crossing points, after applying the 2.5 GT/s Clock Recovery function. Refer to Note 2.
$T_{TX-HF-DJ-DD}$	Tx Deterministic Jitter > 1.5 MHz	Not specified	0.15 (max)	Refer to Table 18-9	UI	Deterministic jitter only. Refer to Notes 2 and 10.
$T_{TX-LF-RMS}$	Tx RMS Jitter < 1.5 MHz	Not specified	3.0	Refer to Table 18-9	ps RMS	Total energy measured over a 10-kHz to 1.5-MHz range.
$T_{TX-RISE-FALL}$	Tx Rise and Fall Time	0.125 (min)	0.15 (min)	Not specified	UI	Measured differentially from 20 to 80% of swing.
$T_{RF-MISMATCH}$	Tx Rise/Fall Mismatch	Not specified	0.1 (max)	Not specified	UI	Measured from 20 to 80% differentially. Refer to Note 2.
$RL_{TX-DIFF}$	TX Differential Return Loss (Package + Silicon)	10 (min)	10 (min) for 0.05 to 1.25 GHz 8 (min) for 1.25 to 2.5 GHz	10 (min) for 0.05 to 1.25 GHz 8 (min) for 1.25 to 2.5 GHz 4 (min) for 2.5 to 4 GHz	dB	
$RL_{TX-CM}$	TX Common Mode Return Loss (Package + Silicon)	6 (min) for 0.05 to 2.5 GHz	6 (min) for 0.05 to 2.5 GHz	6 (min) for 0.05 to 2.5 GHz 3 (min) for 2.5 GHz	dB	
$Z_{TX-DIFF-DC}$	DC Differential Tx Impedance	80 (min) 120 (max)	120 (max)	120 (max)	$\Omega$	Low impedance defined during signaling. Parameter is captured for 5.0 GHz by $RL_{TX-DIFF}$ .

**Table 18-7. PCI Express Transmitter (Signal Group a) – AC and DC Characteristics (Cont.)**

Symbol	Parameter	Link Speed			Units	Comments
		2.5 GT/s	5.0 GT/s	8.0 GT/s		
$V_{TX-CM-AC-PP}$	Tx AC Peak-to-Peak Common Mode Voltage (5.0 GT/s)	Not specified	150 (max)	150 (max)	mVPP	For 8 GT/s, no more than 50 mVPP within the 0.03- to 500-MHz range. Refer to Notes 5 and 12.
$V_{TX-CM-AC-P}$	Tx AC Peak Common Mode Voltage (2.5 GT/s)	20	Not specified	Not specified	mV	Refer to Note 5.
$I_{TX-SHORT}$	Tx Short Circuit Current Limit	90 (max)	90 (max)	90 (max)	mA	Total single-ended current the Transmitter can supply when shorted to its Ground. Refer to Note 13.
$V_{TX-DC-CM}$	Tx DC Common Mode Voltage	0 (min) 3.6 (max)	0 (min) 3.6 (max)	0 (min) 3.6 (max)	V	Allowed DC common mode voltage at a Transmitter ball, under any conditions. Refer to Note 13.
$V_{TX-CM-DC-ACTIVE-IDLE-DELTA}$	Absolute Delta of DC Common Mode Voltage during L0 Link PM state and Electrical Idle	0 (min) 100 (max)	0 (min) 100 (max)	0 (min) 100 (max)	mV	$ V_{TX-CM-DC} [\text{during L0}] - V_{TX-CM-Idle-DC} [\text{during Electrical Idle}]  \leq 100 \text{ mV}$ $V_{TX-CM-DC} = DC_{(avg)} \text{ of }  V_{TX-D+} + V_{TX-D-}  / 2 [L0]$ $V_{TX-CM-Idle-DC} = DC_{(avg)} \text{ of }  V_{TX-D+} + V_{TX-D-}  / 2 [\text{Electrical Idle}]$
$V_{TX-CM-DC-LINE-DELTA}$	Absolute Delta of DC Common Mode Voltage between D+ and D-	0 (min) 25 (max)	0 (min) 25 (max)	0 (min) 25 (max)	mV	$ V_{TX-CM-DC-D+} - V_{TX-CM-DC-D-}  \leq 25 \text{ mV}$ $V_{TX-CM-DC-D+} = DC_{(avg)} \text{ of }  V_{TX-D+} $ $V_{TX-CM-DC-D-} = DC_{(avg)} \text{ of }  V_{TX-D-} $
$V_{TX-IDLE-DIFF-AC-p}$	Electrical Idle Differential Peak Output Voltage	0 (min) 20 (max)	0 (min) 20 (max)	0 (min) 20 (max)	mV	$V_{TX-IDLE-DIFFp} =  V_{TX-Idle-D+} - V_{TX-Idle-D-}  \leq 20 \text{ mV}$ Voltage must be high-pass filtered, to remove any DC component and HF noise. The bandpass is constructed from two first-order filters, the high- and low-pass 3-dB bandwidths are 10 kHz and 1.25 GHz, respectively.
$V_{TX-IDLE-DIFF-DC}$	DC Electrical Idle Differential Peak Output Voltage	Not specified	0 (min) 5 (max)	0 (min) 5 (max)	mV	$V_{TX-IDLE-DIFF-DC} =  V_{TX-Idle-D+} - V_{TX-Idle-D-}  \leq 5 \text{ mV}$ Voltage must be high-pass filtered, to remove any AC component. The low pass filter is first-order with a 3-dB bandwidth of 10 kHz.

**Table 18-7. PCI Express Transmitter (Signal Group a) – AC and DC Characteristics (Cont.)**

Symbol	Parameter	Link Speed			Units	Comments
		2.5 GT/s	5.0 GT/s	8.0 GT/s		
$V_{TX-RCV-DETECT}$	Amount of Voltage Change Allowed during Receiver Detection	600 (max)	600 (max)	600 (max)	mV	Total amount of voltage change in a positive direction that a Transmitter can apply, to sense whether a Low-Impedance Receiver is present.  <i>Note: Receivers display substantially different impedance for <math>V_{IN} &lt; 0</math> versus <math>V_{IN} &gt; 0</math>.</i>
$T_{TX-IDLE-MIN}$	Minimum Time Spent in Electrical Idle	20 (min)	20 (min)	20 (min)	ns	Minimum time a Transmitter must be in Electrical Idle.
$T_{TX-IDLE-SET-TO-IDLE}$	Maximum Time to Transition to a Valid Electrical Idle after Sending an Electrical Idle Ordered-Set	8 (max)	8 (max)	8 (max)	ns	After sending the required Electrical Idle Ordered-Set (EIOS), the Transmitter must meet all Electrical Idle specifications within this time. This is measured from the end of the last UI of the last EIOS to the Transmitter in Electrical Idle.
$T_{TX-IDLE-TO-DIFF-DATA}$	Maximum Time to Transition to Valid Differential Signaling after Leaving Electrical Idle	8 (max)	8 (max)	8 (max)	ns	Maximum time to transition to valid differential signaling, after leaving Electrical Idle. This is considered a de-bounce time to the Tx.
$T_{CROSSLINK}$	Cross-Link Random Timeout	1.0 (max)	1.0 (max)	1.0 (max)	ms	Random timeout that helps resolve potential conflicts in the cross-link configuration.
$L_{TX-SKEW}$	Lane-to-Lane Output Skew	500 ps + 2 UI (max)	500 ps + 4 UI (max)	500 ps + 6 UI (max)	ps	Static skew between any two Lanes within a single Transmitter.
$C_{TX}$	AC-Coupling Capacitor	75 (min) 265 (max)	75 (min) 265 (max)	176 (min) 265 (max)	nF	All Transmitters shall be AC-coupled. The AC coupling is required either within the media, or within the transmitting component itself. Refer to Note 14.

**Notes:**

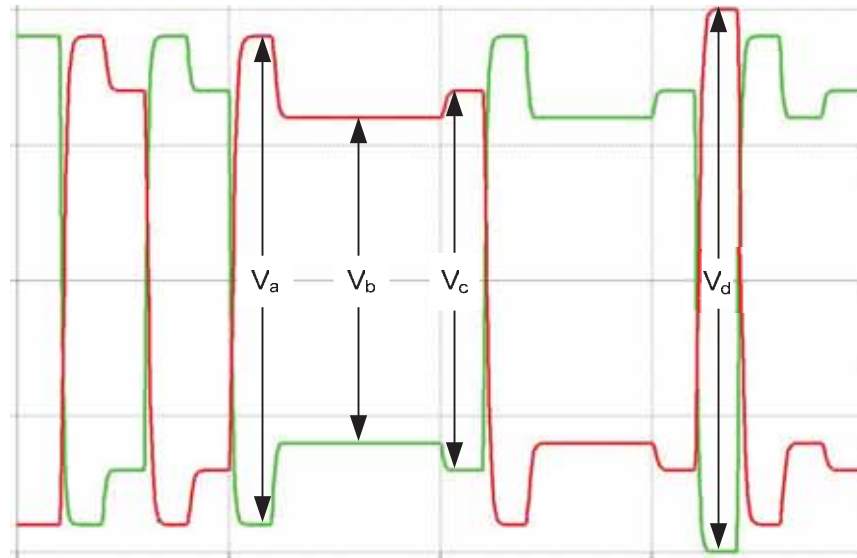
1. SSC permits a +0, -5,000 ppm modulation of the clock frequency, at a modulation rate not to exceed 33 kHz.
2. Measurements at 5.0 GT/s require an oscilloscope with a bandwidth of  $\geq 12.5$  GHz, or equivalent, while measurements made at 2.5 GT/s require an oscilloscope with at least 6.2 GHz bandwidth. Measurement at 5.0 GT/s must de-convolve effects of a compliance test board, to yield an effective measurement at the Tx balls. 2.5 GT/s can be measured within 200 mils of the Tx device's pins, although de-convolution is recommended.
3. Transmitter jitter is measured by driving the Transmitter under test with a low jitter "ideal" clock and connecting the device under test (DUT) to a reference load.
4. Transmitter raw jitter data must be convolved with a filtering function that represents the worst case Clock and Data Recovery (CDR) circuit tracking bandwidth. 2.5 and 5.0 GT/s use different filter functions. After the convolution process is applied, the center of the resulting eye must be determined and then used as a reference point for obtaining eye voltage and margins.
5.  $V_{TX-AC-CM-PP}$  and  $V_{TX-AC-CM-P}$  are defined in the PCI Express Base r3.0, Section 4.3.4.9. Measurement is made over at least  $10^6$  UI.
6. The Tx PLL bandwidth must lie between the minimum and maximum ranges listed in Table 18-7. PLL peaking must lie below the values listed in Table 18-7. The PLL bandwidth extends from zero (0), up to the value(s) specified in Table 18-7.
7. Measurements are made for both common mode and differential return loss. The DUT must be powered up and DC isolated, and its data+/data- outputs must be in the low-Z state at a static value
8. A single combination of PLL bandwidth and peaking is specified for 2.5 GT/s implementations. For 5.0 GT/s, two 20 combinations of PLL bandwidth and peaking are specified, to permit designers to make a tradeoff between the two parameters. If the PLL's minimum bandwidth is  $\geq 8$  MHz, then up to 3.0 dB of peaking is permitted. If the PLL's minimum bandwidth is relaxed to  $\geq 5.0$  MHz, then a tighter peaking value of 1.0 dB must be met. In both cases, the maximum PLL bandwidth is 16 MHz.
9. Low swing output, defined by  $V_{TX-DIFF-PP-LOW}$ , must be implemented with no de-emphasis.
10. For 5.0 GT/s, de-emphasis timing jitter must be removed. This parameter is measured by accumulating a record length of 106 UI while the DUT outputs a compliance pattern.  $T_{MIN-PULSE}$  is defined to be nominally 1 UI-wide, and is bordered on both sides by pulses of the opposite polarity.
11. Root Complex Tx de-emphasis is configured from an Upstream controller. Downstream Tx de-emphasis is Set by way of a command, issued at 2.5 GT/s. For details, refer to the appropriate location in the PCI Express Base r3.0, Section 4.2.
12. Tx common mode (CM) noise for 8.0 GT/s is measured at TP1, without de-embedding the breakout channel. The parameter captures device CM noise only, and is not intended to capture system CM noise. CM noise is measured by applying an LPF with a -3 dB corner frequency at 4 GHz to the raw data. No more than 50 mVPP of the 150 mVPP total can lie within the 0.03- to 500-MHz range.
13.  $I_{TX-SHORT}$  and  $V_{TX-DC-CM}$  stipulate the maximum current/voltage levels that a Transmitter can generate, and therefore define, the worst case transients that a Receiver must tolerate.
14. All Transmitters supporting 8.0 GT/s must implement the 180 to 265 nF  $C_{TX}$  value. Transmitters operating at 2.5 or 5.0 GT/s only may implement over a range of 75 to 265 nF.

## 18.7.2 Tx Preshoot and De-Emphasis

The equalization ratios implemented by the transmitter can be calculated from the PIPE tap-coefficients. These ratios are computed, based upon the voltage levels defined in [Figure 18-1](#).

[Table 18-8](#) defines the **x Preshoot and De-Emphasis Presets**

**Figure 18-1. Definition of Tx Voltage Levels and Equalization Ratios**



where

- De-Emphasis =  $20 \log_{10} (V_b / V_a)$
- Preshoot =  $20 \log_{10} (V_c / V_b)$
- Boost =  $20 \log_{10} (V_d / V_b)$

**Table 18-8. Tx Preshoot and De-Emphasis Presets**

Preset Select <sup>a</sup> (Decimal)	Preshoot (dB)	De-Emphasis (dB)	De-Emphasis <sup>b</sup> (Decimal)	Main <sup>c</sup> (Decimal)	Preshoot <sup>d</sup> (Decimal)	Total
0	0	-6.0	16	47	0	63
1	0	-3.5	11	52	0	63
2	0	-4.5	13	50	0	63
3	0	-2.5	8	55	0	63
4	0	0	0	63	0	63
5	2.0	0	0	57	6	63
6	2.5	0	0	55	8	63
7	3.5	-6.0	13	44	6	63
8	3.5	-3.5	8	47	8	63
9	3.5	0	0	52	11	63
10	0	-9.5	21	42	0	63
11 to 15	<b>Reserved</b>					

- Gen 3 Coefficient Values** register *Preset Select* field (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BD4h[27:24]).
- Gen 3 Coefficient Values** register *Preset 0 Coefficients – De-Emphasis* field (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BD4h[17:12]).
- Gen 3 Coefficient Values** register *Preset 0 Coefficients – Main* field (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BD4h[11:6]).
- Gen 3 Coefficient Values** register *Preset 0 Coefficients – Preshoot* field (Base mode – Port 0, 8, or 16; Virtual Switch mode – Port 0, 8, or 16, accessible through the Management Port, offset BD4h[5:0]).



## 18.7.3 Tx Voltage and Jitter

**Table 18-9. Tx Voltage and Jitter Parameters**

Symbol	Parameter	Value	Units	Notes
$V_{TX-FS-NO-EQ}$	Full Swing Tx Voltage with no TxEq	800 (min) 1,300 (max)	mVPP	Refer to Note 1.
$V_{TX-RS-NO-EQ}$	Reduced Swing Tx Voltage with no TxEq	1,300 (max)	mVPP	Refer to Note 1.
$V_{TX-EIEOS-FS}$	Min Swing during Electrical Idle Exit Ordered-Set (EIEOS) for Full Swing	250 (min)	mVPP	Refer to Note 2.
$V_{TX-EIEOS-RS}$	Min Swing During EIEOS for Reduced Swing	232 (min)	mVPP	Refer to Note 2.
$T_{TX-UTJ}$	Tx Uncorrelated Total Jitter	31.25 (max)	psPP at $10^{-12}$	
$T_{TX-UDJDD}$	Tx Uncorrelated Deterministic Jitter	12 (max)	psPP	
$T_{TX-UPW-TJ}$	Total Uncorrelated PWJ	24 (max)	psPP at $10^{-12}$	Refer to Notes 3 and 4.
$T_{TX-UPW-DJDD}$	Deterministic DjDD Uncorrelated PWJ	10 (max)	psPP	Refer to Notes 3 and 4.
$T_{TX-DDJ}$	Data-Dependent Jitter	18 (max)	psPP	Refer to Notes 4 and 5.
$ps21_{TX}$	Pseudo Package Loss	-3.0 (min)	dB	PP ratio of 64 ones/64 zeros pattern versus 0101 pattern. No Tx equalization. Refer to Note 6.
$V_{TX-BOOST-FS}$	Tx Boost Ratio for Full Swing	8.0 (min)	dB	
$V_{TX-BOOST-RS}$	Tx Boost Ratio for Reduced Swing	2.5 (min)	dB	
$EQ_{TX-COEFF-RES}$	Tx Coefficient Resolution	1/63	—	

**Notes:**

1. Voltage measurements for  $V_{TX-FS-NO-EQ}$  and  $V_{TX-RS-NO-EQ}$  are made using the 64-zeros/64-ones pattern in the compliance pattern.
2. Voltage limits comprehend both full swing and reduced swing modes. The Tx must reject any changes that would violate this specification. The maximum level is covered in the  $V_{TX-FS-NO-EQ}$  measurement which represents the maximum peak voltage that the Tx can drive. The  $V_{TX-EIEOS-FS}$  and  $V_{TX-EIEOS-RS}$  voltage limits are imposed, to guarantee the EIEOS threshold of 175 mVPP at the Rx ball. This parameter is measured, using the actual EIEOS pattern that is part of the compliance pattern, and then removing the ISI contribution of the breakout channel. The Transmitter must advertise a value for LF during TSI at 8.0 GT/s that ensures that these parameters are met.
3. PWJ parameters shall be measured after DDJ separation.
4. Measured with optimized preset value, after de-embedding to the Tx ball.
5. The 18 ps number takes measurement errors into account.
6. The -3.0 dB number takes measurement errors into account. For some Tx package/driver combinations,  $ps21_{TX}$  can be greater than 0 dB.

## 18.8 Receive Characteristics

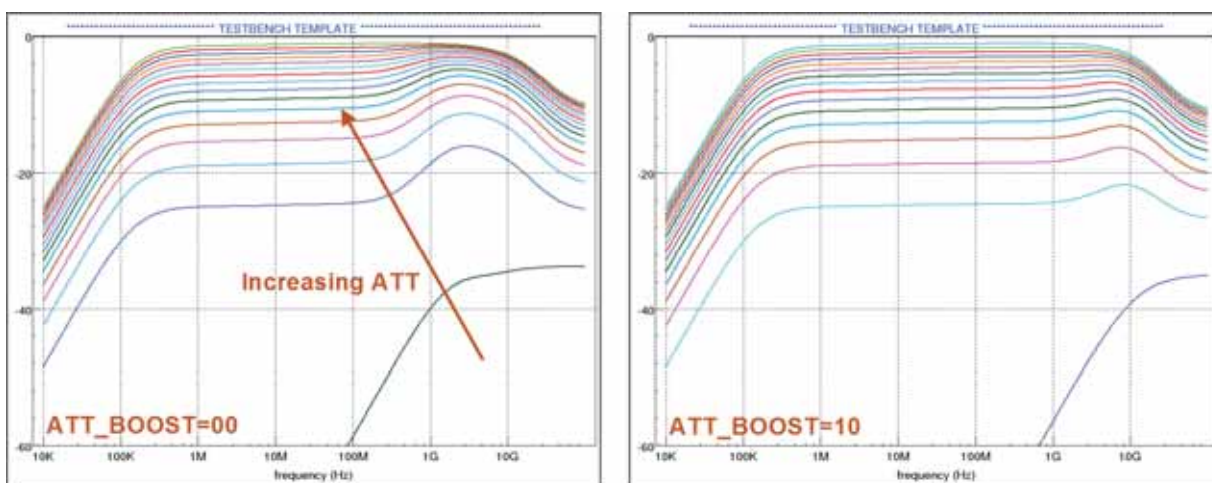
The Receiver circuit includes programmable equalization, to further compensate for the low-pass FR4 loss characteristics of the channel.

### 18.8.1 Receive Equalization

Figure 18-2 illustrates an example frequency response of the attenuator (ATT), which displays nearly 24 dB of programmable attenuation. Also, the ATT is capable of providing up to 9 dB of analog boost at high frequencies, for channel compensation.

The Rx Attenuator can be programmed, using the registers listed in Table 18-10.

**Figure 18-2. ATT Frequency Response for Different ATT Settings**



**Table 18-10. Rx Attenuator Control**

Register	Description
rx_att[3:0]	Rx attenuation value (15 minimum, 0 maximum).
rx_att_boost[1:0]	Rx attenuator analog boost value (2 minimum, 0 maximum, 3 invalid).

Figure 18-3. Increasing Boost

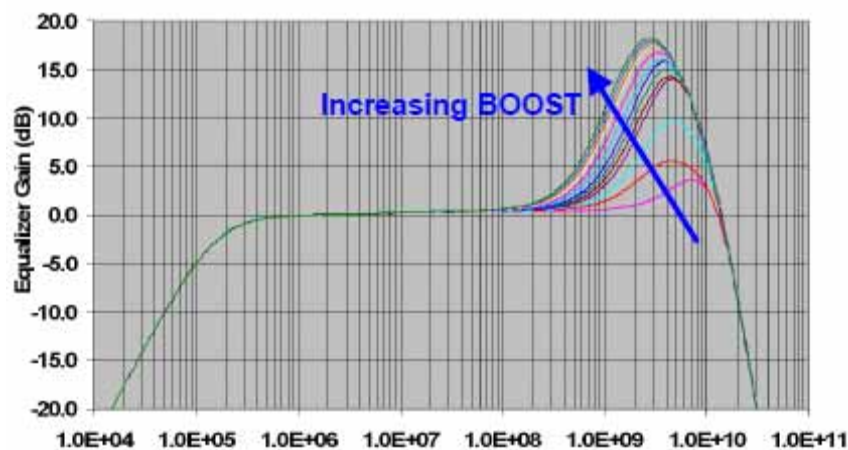


Table 18-11. Boost Settings and Gain

Boost Setting	Gain @ 4 GHz (dB)
0	0.75
1	1.31
2	1.74
3	2.46
4	4.69
5	5.46
6	7.84
7	8.18
8	10.10
9	11.78
10	12.98
11	15.20
12	15.61
13	17.25
14	17.25
15	17.25

## 18.8.2 PCI Express Receiver AC and DC Characteristics

**Table 18-12. PCI Express Receiver (Signal Group b) – AC and DC Characteristics**

Symbol	Parameter	Link Speed			Units	Comments
		2.5 GT/s	5.0 GT/s	8.0 GT/s		
UI	Unit Interval	399.88 (min) 400.12 (max)	199.94 (min) 200.06 (max)	124.9625 (min) 125.0375 (max)	ps	UI does not account for variations caused by SSC.
$V_{RX-DIFF-PP-CC}$	Differential Rx Peak-to-Peak Voltage for Common REFCLK Rx Architecture	0.175 (min) 1.2 (max)	0.120 (min) 1.2 (max)	Refer to <a href="#">Table 18-13</a> and <a href="#">Table 18-14</a>	V	
$V_{RX-DIFF-PP-DC}$	Differential Rx Peak-to-Peak Voltage for Data Clocked Rx Architecture	0.175 (min) 1.2 (max)	0.100 (min) 1.2 (max)	Refer to <a href="#">Table 18-13</a> and <a href="#">Table 18-14</a>	V	
$T_{RX-EYE}$	Receiver Eye Time Opening	0.40 (min)	–	Refer to <a href="#">Table 18-13</a> and <a href="#">Table 18-14</a>	UI	Minimum eye time at Rx balls to yield a $10^{-12}$ Bit Error Rate. Receiver eye margins are defined into a $2 \times 50\Omega$ reference load.
$T_{RX-TJ-CC}$	Maximum Rx Inherent Timing Error for Common REFCLK Rx Architecture	–	0.40 (max)	Refer to <a href="#">Table 18-13</a> and <a href="#">Table 18-14</a>	UI	Refer to Note 1.
$T_{RX-TJ-DC}$	Maximum Rx Inherent Timing Error for Data Clocked Rx Architecture	–	0.34 (max)	Refer to <a href="#">Table 18-13</a> and <a href="#">Table 18-14</a>	UI	Refer to Note 1.
$T_{RX-DJ-DD-CC}$	Maximum Rx Inherent Deterministic Timing Error for Common REFCLK Rx Architecture	–	0.30 (max)	Refer to <a href="#">Table 18-13</a> and <a href="#">Table 18-14</a>	UI	Refer to Note 1.
$T_{RX-DJ-DD-DC}$	Maximum Rx Inherent Deterministic Timing Error for Data Clocked Rx Architecture	–	0.24 (max)	Refer to <a href="#">Table 18-13</a> and <a href="#">Table 18-14</a>	UI	Refer to Note 1.

**Table 18-12. PCI Express Receiver (Signal Group b) – AC and DC Characteristics (Cont.)**

Symbol	Parameter	Link Speed			Units	Comments
		2.5 GT/s	5.0 GT/s	8.0 GT/s		
$T_{RX-EYE-MEDIAN-to-MAX-JITTER}$	Maximum Time Delta between the Median and Deviation from the Median	0.3 (max)	Not specified	Not specified	UI	Only specified for 2.5 GT/s.
$T_{RX-MIN-PULSE}$	Minimum Width Pulse at Rx	Not specified	0.6 (min)	Not specified	UI	Measured to account for worst $T_j$ at $10^{-12}$ Bit Error Rate.
$V_{RX-MAX-MIN-RATIO}$	Minimum/Maximum Pulse Voltage on Consecutive UI	Not specified	5 (max)	Not specified	–	Rx eye must simultaneously meet $V_{RX-EYE}$ limits.
$BW_{RX-PLL}$	Maximum Rx PLL Bandwidth for 2.5 GT/s	1.5 (min) 22 (max)	Not specified	Not specified	MHz	Refer to Note 2.
$BW_{RX-PKG-PLL1}$	Rx PLL Bandwidth corresponding to $PKG_{RX-PLL1}$	Not specified	8 (min) 16 (max)	2 (min) 4 (max)	MHz	Second Order PLL Jitter Transfer Bounding function. Refer to Note 2.
$BW_{RX-PKG-PLL2}$	Rx PLL Bandwidth corresponding to $PKG_{RX-PLL2}$	Not specified	5 (min) 16 (max)	2 (min) 5 (max)	MHz	Second Order PLL Jitter Transfer Bounding function. Refer to Note 2.
$PKG_{RX-PLL1}$	Rx PLL Peaking Limit for PLL1	Not specified	3.0 (max)	2.0 (max)	dB	PLL Bandwidth = 8 MHz (min) at 5 GT/s or Bandwidth = 4 MHz (max) at 8 GT/s. Refer to Note 2.
$PKG_{RX-PLL2}$	Rx PLL Peaking Limit for PLL2	Not specified	1.0 (max)	1.0 (max)	dB	PLL Bandwidth = 5 MHz (min) at 5 GT/s or Bandwidth = 5 MHz (max) at 8 GT/s. Refer to Note 2.
$RL_{RX-DIFF}$	Rx Differential Return Loss (Package + Silicon)	10 (min)	10 (min) for 0.05 to 1.25 GHz 8 (min) for 1.25 to 2.5 GHz	10 (min) for 0.05 to 1.25 GHz 8 (min) for 1.25 to 2.5 GHz 5 (min) for 2.5 to 4.0 GHz	dB	Refer to Note 3.

**Table 18-12. PCI Express Receiver (Signal Group b) – AC and DC Characteristics (Cont.)**

Symbol	Parameter	Link Speed			Units	Comments
		2.5 GT/s	5.0 GT/s	8.0 GT/s		
$RL_{RX-CM}$	Common Mode Return Loss	6 (min)	6 (min)	6 (min) for 0.05 to 1.25 GHz  8 (min) for 2.5 to 4.0 GHz	dB	Refer to Note 3.
$Z_{RX-DC}$	Rx DC Single Ended Impedance	40 (min) 60 (max)	40 (min) 60 (max)	Not specified	$\Omega$	DC impedance limits are needed to guarantee Receiver detect. Refer to Note 4.
$T_{RX-GND-FLOAT}$	Rx Termination Ground Float Time	Not specified	Not specified	500	$\mu s$	Time allowed to float Rx internal Ground in 2.5 or 5.0 to 8.0 GT/s configuration change. Refer to Note 7.
$Z_{RX-DIFF-DC}$	DC Differential Rx Impedance	80 (min) 120 (max)	Not specified	Not specified	$\Omega$	For 5.0 and 8.0 GT/s, covered under the $RL_{RX-DIFF}$ parameter. Refer to Note 4.
$V_{RX-CM-AC-P}$	Rx AC Common Mode Voltage	150 (max)	150 (max)	75 mV (max) (EH <100 mVPP)  125 mV (max) (EH $\geq$ 100 mVPP)	mVP	Measured at Rx balls, into a pair of 50 $\Omega$ terminations into Ground. Refer to Note 5.
$Z_{RX-HIGH-IMP-DC-POS}$	DC Input Common Mode Input Impedance for Voltage >0 during Reset or Power-Down	10K (min)	10K (min)	10K (min)	$\Omega$	Rx DC common mode impedance with the Rx terminations not powered, measured over the range 0 to 200 mV (with respect to Ground). Refer to Note 6.
$Z_{RX-HIGH-IMP-DC-NEG}$	DC Input Common Mode Input Impedance for Voltage <0 during Reset or Power-Down	1.0K (min)	1.0K (min)	1.0K (min)	$\Omega$	Rx DC common mode impedance with the Rx terminations not powered, measured over the range -150 to 0 mV (with respect to Ground). Refer to Note 6.

**Table 18-12. PCI Express Receiver (Signal Group b) – AC and DC Characteristics (Cont.)**

Symbol	Parameter	Link Speed			Units	Comments
		2.5 GT/s	5.0 GT/s	8.0 GT/s		
$V_{RX-IDLE-DET-DIFFp-p}$	Electrical Idle Detect Threshold	65 (min) 175 (max)	65 (min) 175 (max)	65 (min) 175 (max)	mV	$V_{RX-IDLE-DET-DIFFp-p} = \frac{1}{2} \times  V_{RX-D+} - V_{RX-D-} $ Measured at the Receiver's package pins.
$T_{RX-IDLE-DET-DIFF-ENTERTIME}$	Unexpected Electrical Idle Enter Idle Detect Threshold Integration Time	10 (max)	10 (max)	10 (max)	ms	An un-expected Electrical Idle ( $V_{RX-DIFFp-p} < V_{RX-IDLE-DET-DIFFp-p}$ ) must be recognized no longer than $T_{RX-IDLE-DET-DIFF-ENTERTIME}$ to signal an unexpected idle condition.
$L_{RX-SKEW}$	Total Lane-to-Lane Skew	20 (max)	8 (max)	6 (max)	ns	Across all Lanes on a Port. $L_{RX-SKEW}$ comprehends Lane-to-Lane variations due to channel and repeater delay differences.

**Notes:**

1. The four inherent timing error parameters are defined for the convenience of Rx designers, and they are measured during Receiver tolerancing.
2. Two combinations of PLL bandwidth and peaking are specified at 5.0 GT/s, to permit designers to make trade-offs between the two parameters. If the PLL's minimum bandwidth is  $\geq 8$  MHz, then up to 3.0 dB of peaking is permitted. If the PLL's minimum bandwidth is relaxed to  $\geq 5.0$  MHz, then a tighter peaking value of 1.0 dB must be met.

A PLL bandwidth extends from zero up to the value(s) defined as the minimum or maximum in [Table 18-12](#). For 2.5 GT/s, a single PLL bandwidth and peaking value of 1.5 to 22 MHz and 3.0 dB are defined.

3. Measurements must be made for both common mode and differential return loss. In both cases, the DUT must be powered up and DC-isolated, and its D+/D- inputs must be in the low-Z state.
4. The Rx DC single-ended impedance must be present when the Receiver terminations are first enabled, to ensure that the Receiver Detect properly occurs. Compensation of this impedance can start immediately, and the Rx single-ended impedance (constrained by  $RL_{RX-CM}$  to  $50\Omega \pm 20\%$ ) must be within the specified range by the time Detect is entered.
5. Common mode peak voltage is defined by the expression:

$$\max\{|(V_{d+} - V_{d-}) - V_{CMDC}|\}$$

6.  $Z_{RX-HIGH-IMP-DC-NEG}$  and  $Z_{RX-HIGH-IMP-DC-POS}$  are defined, respectively, for negative and positive voltages at the input of the Receiver. Transmitter designers must comprehend the large difference between  $>0$  and  $<0$  Rx impedances, when designing Receiver detect circuits.
7. Defines the time for the Receiver's input pads to settle to new common mode on 2.5 or 5.0 GT/s transition to 8.0 GT/s.



**Table 18-13. Stressed Voltage Eye Parameters**

Symbol	Parameter	Limits at 8.0 GT/s	Units	Comments
$V_{RX-LAUNCH-8G}$	Generator Launch Voltage	800	mVPP	Measured at TP1. Refer to the <i>PCI Express Base r3.0</i> , Figure 4-65. $V_{RX-LAUNCH-8G}$ can be adjusted, if necessary, to yield the proper EH, as long as the outside eye voltage at TP2 does not exceed 1,300 mVPP.
$T_{RX-UI-8G}$	Unit Interval	125.00	ps	Nominal value is sufficient for Rx tolerancing. Value does not account for SSC.
$V_{RX-SV-8G}$	Eye Height at TP2P	25 (-20 dB channel) 50 (-12 dB channel) 200 (-3 dB channel)	mVPP	Eye height at BER = $10^{-12}$ . Refer to Notes 1, 2.
$T_{RX-SV-8G}$	Eye Width at TP2P	0.3 to 0.35	UI	Eye width at BER = $10^{-12}$ . Refer to Note 2.
$V_{RX-SV-DIFF-8G}$	Differential Mode Interference	14 or greater	mVPP	Adjusted to Set EH. Frequency = 2.10 GHz. Refer to Note 3.
$V_{RX-SV-CM-8G}$	Rx AC Common Mode Voltage at TP2P	150 (EH < 100 mVPP) 250 (EH ≥ 100 mVPP)	mVPP	Defined for a single tone at 120 MHz. Refer to Note 3.
$T_{RX-SV-SJ-8G}$	Sinusoidal Jitter at 100 MHz	0.1	UI PP	Fixed at 100 MHz. Refer to Note 4.
$T_{RX-SV-RJ-8G}$	Random Jitter	2.0	ps RMS	Rj is spectrally flat before filtering. Refer to Notes 4, 5
$V_{RX-MAX-SE-SW}$	Maximum Single-Ended Swing	±300	mVP	Refer to Note 6.

**Notes:**

- $V_{RX-SV-8G}$  is tested at three different voltages, to ensure that the Rx DUT is capable of equalizing over a range of channel loss profiles. The test also guarantees that the Rx is capable of operating over a sufficient dynamic range of eye heights. “SV” in the parameter names refers to “stressed voltage”.
- $V_{RX-ST-8G}$  and  $T_{RX-ST-8G}$  are referenced to TP2P, and are obtained after post processing data captured at TP2.  
  
 $V_{RX-ST-8G}$  and  $T_{RX-ST-8G}$  include the effects of applying the behavioral Rx model and Rx behavioral equalization.
- $V_{RX-SV-DIFF-8G}$  measurement is made at TP2, without post processing.  $V_{RX-SV-CM-8G}$  can be made at either TP1 or TP2.  $V_{RX-SV-DIFF-8G}$  voltage might need to be adjusted over a wide range for the different loss calibration channels.
- $T_{RX-SV-SJ-8G}$  and  $T_{RX-SV-RJ-8G}$  measurements are made at TP1, without post processing.
- Rj is applied over the following range. The low frequency limit can be between 1.5 and 10 MHz, and the upper limit is 1.0 GHz. Refer to the *PCI Express Base r3.0*, Figure 4-74, for details.
- $V_{RX-MAX-SE-SW}$  sets the maximum outer, single-ended eye voltage limit in the presence of differential and CM noise applied to the Rx, as observed at TP2 relative to ground with no behavioral RxEq post processing.

**Table 18-14. Stressed Jitter Eye Parameters**

Symbol	Parameter	Limits at 8.0 GT/s	Units	Comments
$V_{RX-LAUNCH-8G}$	Generator Launch Voltage	800 (nominal)	mVPP	Measured at TP1. Refer to the <i>PCI Express Base r3.0</i> , Figure 4-65. Refer to Note 1.
$T_{RX-UI-8G}$	Unit Interval	125.00	ps	Nominal value is sufficient for Rx tolerancing. Value does not account for SSC.
$V_{RX-ST-8G}$	Eye height at TP2P	25 (min) 35 (max)	mVPP	At BER = $10^{-12}$ . Refer to Note 2.
$T_{RX-ST-8G}$	Eye width at TP2P	0.30	UI	At BER = $10^{-12}$ . Refer to Note 2.
$T_{RX-ST-SJ-8G}$	Sinusoidal Jitter	0.1 to 1.0	UI PP	Measured at TP1. Refer to the <i>PCI Express Base r3.0</i> , Figure 4-74. Refer to Note 3.
$T_{RX-ST-RJ-8G}$	Random Jitter	3.0	ps RMS	Rj spectrally flat before filtering. Measured at TP1. Refer to Note 4.

**Notes:**

1.  $V_{RX-LAUNCH-8G}$  can be adjusted to meet  $V_{RX-ST-8G}$ , as long as the outside eye voltage at TP2 does not exceed 1,300 mVPP.
2.  $V_{RX-ST-8G}$  and  $T_{RX-ST-8G}$  are referenced to TP2P, and are obtained after post processing data captured at TP2.

$V_{RX-ST-8G}$  and  $T_{RX-ST-8G}$  include the effects of applying the behavioral Rx model and Rx behavioral equalization.

3.  $T_{RX-ST-SJ-8G}$  can be measured at either TP1 or TP2.
4. While the nominal value is specified at 3.0 ps RMS, it can be adjusted to meet the 0.3 UI value for  $T_{RX-ST-8G}$ . Rj is measured at TP1, to prevent data-channel interaction from adversely affecting Rj calibration accuracy.

Rj is applied over the following range. The low-frequency limit can be between 1.5 and 10 MHz, and the upper limit is 1.0 GHz.



## Chapter 19 Thermal and Mechanical Specifications

### 19.1 Thermal Characteristics

Table 19-1. Sample Thermal Data, Commercial Temperature, with Heat Sink<sup>a</sup>

Maximum Power (Watts) <sup>b</sup>	Air Flow Velocity (m/s)	$\Theta_{JA}$ (°C/W) JEDEC 4-Layer Board	$\Theta_{JA}$ (°C/W) JEDEC 8-Layer Board	$\Psi_{JB}$ (°C/W) JEDEC 4-Layer Board	$\Psi_{JB}$ (°C/W) JEDEC 8-Layer Board	$\Theta_{JC}$ (°C/W) JEDEC 4-Layer Board	$\Theta_{JC}$ (°C/W) JEDEC 8-Layer Board	$\Theta_{JB}$ (°C/W) JEDEC 4-Layer Board	$\Theta_{JB}$ (°C/W) JEDEC 8-Layer Board	Heat Sink
15.02	0.00	6.84	4.68	1.87	1.53	0.62	0.58	2.75	1.95	Aavid Thermalloy, 374324B60023G
	1.00	3.15	2.58	1.26	1.16					
	2.00	2.14	1.82	1.05	0.99					

- a. The Maximum Operating Junction Temperature is 110°C.  
The Maximum Junction Temperature for Reliability is 110°C.
- b. The maximum power value listed assumes the conditions listed in [Chapter 18, “Electrical Specifications,”](#) at Gen 3 (8.0 GT/s), with all Ports enabled.

## 19.2 General Package Specifications

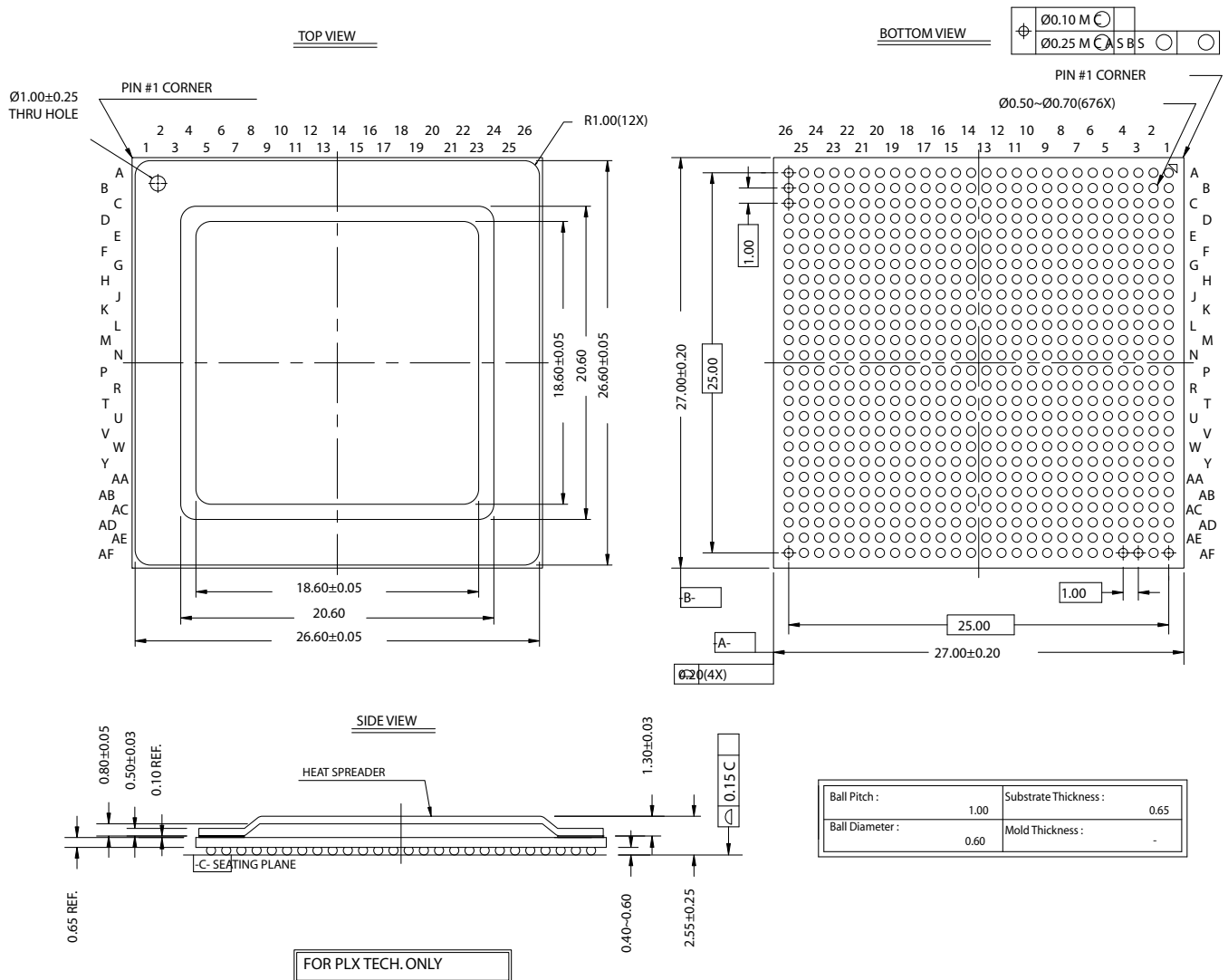
Table 19-2 lists general package specifications. For a more complete list, refer to Figure 19-1.

**Table 19-2. General Package Specifications**

Parameter	Specification
Package Type	Flip-Chip Ball Grid Array with Heat Spreader (HFC-BGA)
Quantity of Balls	676
Package Dimensions	27 x 27 mm <sup>2</sup> (approximately 2.55 ±0.25-mm high)
Ball Matrix Pattern	26 x 26
Ball Pitch	1.0 mm
Ball Diameter	0.60 ±0.1 mm
Ball Spacing	0.40 mm
Ball Pad Diameter	0.58 mm
Solder Mask Opening on Ball Pad	0.48 mm

## 19.3 Mechanical Dimensions

Figure 19-1. Mechanical Dimensions – 27 x 27 mm<sup>2</sup> Package



NDA CONFIDENTIAL -- Inventec | King Michael

THIS PAGE INTENTIONALLY LEFT BLANK.



## Appendix A General Information

### A.1 Product Ordering Information

Contact your local [PLX Sales Representative](#) for ordering information.

**Table A-1. Product Ordering Information**

Part Numbers	Description
PEX8748-BA80BC G PEX8748-CA80BC G  <i>where</i>	PEX 8748 48-Lane, 12-Port PCI Express Gen 3 Multi-Root Switch Lead-Free HFC-BGA Package (27 x 27 mm <sup>2</sup> , 676-ball)  PEX – PCI Express Product Family 8748 – Part Number BA, CA – Silicon Revision 80 – Signaling Rate (8.0 GT/s) B – Flip-Chip Ball Grid Array C – Commercial Temperature G – Lead-free (RoHS 6/6 and Green compliant)
PEX 8748-BA RDK	PEX 8748 Rapid Development Kit with x16 Edge Connector
PEX 8748-CA RDK	PEX 8748 Rapid Development Kit with x16 Edge Connector
Breakout Board-88	Breakout Board with x16 Edge Connector for Additional Fan-Out to Two Slots (x8, x8)
Breakout Board-844	Breakout Board with x16 Edge Connector for Additional Fan-Out to Three Slots (x8, x4, x4)
x1 Adapter	PCI Express x16 to x1 Adapter
x4 Adapter	PCI Express x16 to x4 Adapter
x8 Adapter	PCI Express x16 to x8 Adapter

NDA CONFIDENTIAL -- Inventec | King Michael

## **A.2 United States and International Representatives and Distributors**

PLX Technology, Inc., representatives and distributors are listed at [www.plxtech.com](http://www.plxtech.com).

## **A.3 Technical Support**

PLX Technology, Inc., technical support information is listed at [www.plxtech.com/support](http://www.plxtech.com/support), or call 800 759-3735 (domestic only) or 408 774-9060.